HOCHGESCHWINDIGKEITS-TRANSPORTPROTOKOLLE

- Architekturen und Mechanismen -

Torsten Braun
Ortwin Rose

Institut für Telematik
Universität Karlsruhe
Am Zirkel 2

D-7500 Karlsruhe 1

Der Übergang zu glasfaserbasierten Kommunikationsnetzen leitet eine neue Ära der Telekommunikationstechnologie ein. Im Licht der veränderten Randbedingungen hinsichtlich des unterliegenden Mediums (Datenraten im Bereich einiger Gigabit pro Sekunde - auch im Weitverkehrsbereich - bei Bitfehlerraten der Größenordnung 10^{-13}) und der in solchen Hochgeschwindigkeitsnetzen realisierbar gewordenen neuen Anwendungen stellt sich jedoch immer drängender die Frage, inwiefern traditionelle Protokolle (wie TCP und OSI-TP4) diesen Veränderungen gewachsen sind. Der vorliegende Artikel versucht, dem Leser einen Eindruck über die Konsequenzen der rasch stattfindenden Verbreitung dieser modernen Technologie zu vermitteln.

Innerhalb des ersten Teils werden zunächst die modifizierten Randbedingungen, wie sie in Hochgeschwindigkeitsnetzen vorgefunden werden, dargelegt. Weiterhin werden einige der sich bereits abzeichnenden neuen Anwendungen vorgestellt, welche sich die zur Verfügung stehende extrem große Bandbreite zunutze machen. Anschließend erfolgt eine Diskussion der Schwierigkeiten, die für traditionelle Protokolle in einer solchen Umgebung auftreten, und mögliche Ursachen aufgezeigt.

Der zweite Teil führt den Leser in eine Reihe von Protokollen ein, die speziell im Hinblick auf die modifizierten Anforderungen neuer Anwendungen und Hochgeschwindigkeitsnetze entwickelt worden sind. Anhand dieser Beispiele wird aufgezeigt, mit welchen Methoden die im ersten Abschnitt angesprochenen Probleme gelöst werden sollen. Den Abschluß bildet ein kurzer Ausblick auf die derzeit innerhalb der Normungsgremien hinsichtlich Hochgeschwindigkeitsprotokollen beginnenden Aktivitäten.

Einleitung

Die Glasfasertechnologie erweist sich derzeit als ein Katalysator für neue Anwendungen im Bereich der Telekommunikation. Sie bietet als zugrundeliegendes Übertragungsmedium eine Reihe sehr günstiger Eigenschaften an. Dazu zählen vor allem

- eine um mehrere Größenordnungen höhere Bandbreite als konventionelle Netze. So werden selbst für den Weitverkehrsbereich Datenraten im Bereich einiger Gigabit/s schon in naher Zukunft erwartet .
- eine niedrige Bitfehlerrate im Bereich von etwa 10^{-11} bis 10^{-13}, verglichen mit etwa 10^{-5} bis 10^{-6} in konventionellen Netzen
- das hohe Bandbreiten-Längenprodukt erlaubt die Übertragung von Daten mit einigen Gigabit/s über relativ große Entfernungen, ohne daß eine Signalregenerierung erforderlich ist.
- 'Optische Verstärker' erlauben die Signalregeneration ohne die Notwendigkeit einer Umwandlung der Lichtimpulse in elektrische Signale, der Regeneration auf elektrischem Weg und ihrer Rückwandlung in optische Impulse. Auf diese Weise können zum einen die durch die Umwandlung entstehenden zum Teil erheblichen Verluste eliminiert werden, zum anderen lassen sich mit Optischen Verstärkern alle Lichtbereiche gleichmäßig verstärken. Insbesondere die letzte Tatsache macht den sog. Wellenlängenmultiplex (d.h. die gleichzeitige Übertragung mehrerer Kanäle durch Licht unterschiedlicher Farbe) und damit die gesamte Nutzung der Glasfaser ökonomischer. So wurde in /Bell92/ eine Übertragung mit einer Datenrate von 10 Gbit/s über eine Strecke von 2000 km bei einer Bitfehlerrate von 10^{-13} mit Hilfe Optischer Verstärker vorgeführt. Dabei wurde nur Licht einer Wellenlänge genutzt, so daß sich diese Bandbreite noch um einen gewissen Faktor erhöhen läßt.

Insbesondere die letzten beiden Punkte machen die Glasfaser vor allem für den Weitverkehrsbereich sehr attraktiv. Als Konsequenz dieser Tatsache ist eine sehr rasche Verbreitung der Glasfaser speziell in diesem Sektor zu erkennen, so daß sehr große Bandbreiten auch im Weitverkehrsbereich verfügbar sein werden (so z.B. in verschiedenen 'Gigabit-Testumgebungen' in den USA].

Die Aufgabe, diese enorme Bandbreite des physikalischen Mediums auch dem Benutzer zur Verfügung zu stellen, stellt sehr hohe Anforderungen an die verwendeten Protokolle. Standen zur Bearbeitung eines Paketes (bei einer mittleren Paketlänge von 1024 byte) bei einer konventionellen Übertragung im Weitverkehrsbereich in der Größenordnung von 120 ms und im Bereich Lokaler Netze noch immerhin 1 ms zur Verfügung, so verkürzt sich diese Zeitspanne nunmehr auf wenige Mikrosekunden!

Um Pakete mit dieser Geschwindigkeit bearbeiten zu können, sind Maßnahmen auf mehreren Ebenen notwendig. Auf der Ebene der Protokollimplementierung wird eine Hardware-Realisierung in VLSI-Technik favorisiert. Um diese Vorgehensweise zu erleichtern, werden komplementär vor allem auf den für den Transport zwischen den Endsystemen verantwortlichen Schichten (die den aggregierten Verkehr einer Vielzahl von Verbindungen transportieren müssen) sehr einfache Protokolle verwendet (wie z.B. Frame-Relaying /Rick92/, SMDS /HeRo92/). Diese realisieren einen einfachen Übertragungsdienst, bei dem die Aufgaben der Flußkontrolle und Fehlerkorrektur den Endbe-

nutzern überlassen werden. Eine solche Aufgabenteilung ergibt sich ganz natürlich im Licht der stark gesunkenen Bitfehlerraten, die zeitaufwendige Fehlerkorrekturmaßnahmen über Teilstrecken des Netzes nicht mehr rechtfertigen.

Mit dieser Entscheidung wächst gleichzeitig die Bedeutung der Transportprotokolle, die nun einen Großteil der Aufgaben einer sicheren Ende-zu-Ende Verbindung übernehmen müssen. Besonders problematisch ist aus Sicht dieser Protokolle, daß die sich gleichzeitig im Fluß befindliche Datenmenge proportional zum Anstieg der Datenrate wächst. Um diese Datenrate voll nutzen zu können ist es notwendig, möglichst ohne Pausen Daten zu übertragen. Dazu muß der dem Sender zur Verfügung stehende Kredit so weit erhöht werden, daß ein kontinuierliches Senden möglich ist, bis die erste Quittung des Empfängers eintrifft. Diese Zeitspanne (auch *Umlaufzeit* einer Verbindung genannt) setzt sich aus drei Komponenten zusammen:

- der Bearbeitungszeit eines Pakets durch die an der Übertragung beteiligten Instanzen
- der Wartezeit eines Paketes, d.h. der Zeitspanne in der es innerhalb einer Warteschlange auf die Bearbeitung durch eine Instanz wartet, und schließlich
- der physikalischen Signallaufzeit auf dem unterliegenden Medium.

Während die Bearbeitungszeit mit zunehmender Datenrate geringer wird und die Wartezeit durch ein geeignetes Verfahren möglichst klein gehalten werden sollte, stellt die Signallaufzeit eine physikalische Größe dar, die sich nicht proportional zur Datenrate verändert. Betrachtet man z.B. eine 3000 km lange Übertragungsstrecke, so sind auf dieser bei einer Übertragung mit 64 kbit/s etwa 640 bit allein aufgrund der Signallaufzeit im Transit (die sog. *Pfadkapazität* einer Verbindung, auch Bandbreiten-Delay Produkt genannt). Diese relativ geringe Datenmenge vergrößert sich jedoch bei einer Übertragung mit 500 Mbit/s auf 5 000 000 bit, also etwa 625 kbyte! Es wird somit deutlich, daß die relativ einfachen Verfahren zur Flußkontrolle, wie sie in konventionellen Netzen bei einer kleinen Pfadkapazität durchaus berechtigt waren, in Hochgeschwindigkeitsnetzen durch wesentlich anspruchsvollere Verfahren zu ersetzen sind. Der Bereich der Flußkontrolle wird daher von vielen auch als ein zentrales Problem beim Übergang in diese neuen Umgebungen betrachtet /Part90/.

Weiterhin sehen sich diese Protokolle mit der Entwicklung vollkommen neuer Anwendungen konfrontiert, die aufgrund der neuen Dimensionen hinsichtlich des Datentransports, wie er von glasfaserbasierten Netzen eröffnet wird, überhaupt erst möglich geworden sind. Im Licht dieser Veränderungen wird derzeit sehr intensiv die Frage diskutiert, ob die aktuell verwendeten Transportprotokolle (wie z.B. TCP und ISO-TP4) diesen modifizierten Anforderungen überhaupt gewachsen sind, oder ob neue Protokolle bzw. vollkommen neue Architekturen für diese Hochgeschwindigkeitsnetze geschaffen werden müssen.

Der vorliegende Artikel soll dem Leser einen Überblick über die derzeitigen Aktivitäten im Bereich der Transportprotokolle für Hochgeschwindkeitsnetze geben. Dabei werden zunächst grob die Aufgaben des Transportprotokolls dargestellt, wobei insbesondere auch auf die Anforderungen der neuen, in einer solchen Umgebung möglich gewordenen Anwendungen eingegangen wird. Anschließend wird die Fähigkeit traditioneller Protokolle diskutiert, die geforderten Aufgaben auch unter den neuen Randbedingungen zu erbringen. Den Hauptteil bildet ein Überblick über die Evolution spezieller Protokolle, die (i.a. im Hinblick auf ein spezifisches Einsatzfeld) unter Berücksichtigung von Hochgeschwindigkeitsnetzen entwickelt worden sind. Dabei wird sich zeigen, daß eine recht natürliche Entwicklungslinie aufgezeigt werden kann, nach der - neben der Einführung spezieller neuer Aspekte - neu entwickelte Protokolle durchaus von den Innovationen ihrer Vorgänger profitiert und diese mit integriert haben. Den Abschluß bildet ein kurzer Ausblick auf derzeitige Aktivitäten innerhalb der Normungsgremien, die sich den modifizierten Anforderungen nicht länger verschließen können und derzeit den Bereich der transportorientierten Schichten im Hinblick auf den Einsatz in Hochgeschwindigkeitsnetzen einer Prüfung unterwerfen.

Aufgaben der Transportschicht

Die Transportschicht stellt die vierte Schichte innerhalb des OSI Basisreferenzmodells dar. Sie operiert als erste Schicht zwischen den Endsystemen und bietet dem Benutzer ein einheitliches Interface unabhängig vom speziell unterliegenden Netz an. Die grundlegende Aufgabe der Transportschicht besteht in der der Verwaltung einer Assoziation zwischen den beiden (End-) Teilnehmern. Die ihr übergebenen Daten sollen dabei zuverlässig zwischen den Endsystemen ausgetauscht werden, d.h. die Daten sollen fehlerfrei und i.a. auch in der gleichen Reihenfolge, in der sie vom Sender übergeben worden sind, beim Empfänger abgeliefert werden. Um eine genauere Spezifikation der neu hinzukommenden Anforderungen geben zu können, müssen zunächst auch die in einem Gigabit-Netz vorzufindenden Anwendungen kurz charakterisiert werden.

Die bereits in traditionellen Netzen vorzufindenden Anwendungen wie File-Transfer, Electronic Mail und Terminalzugang werden natürlich auch in den Hochgeschwindigkeitsnetzen der Zukunft eine Rolle spielen. Sie fordern vor allem eine fehlerfreie, reihenfolgebewahrende Übertragung der Daten, stellen jedoch keine oder nur geringe Anforderungen an die maximale Übertragungsdauer.

Neben diesen werden eine Reihe neuer Anwendungen vorzufinden sein, die im folgenden kurz umrissen werden sollen. Es wird dabei kein Anspruch auf Vollständigkeit erhoben, vielmehr soll ein Eindruck über diese neuen Dienste vermittelt werden.

Verteilte Verarbeitung

Aufgrund der weiten Verfügbarkeit leistungsfähiger Workstations und schneller Lokaler Netze ist eine zunehmende Verbreitung Verteilter Anwendungen zu beobachten. Ein gemeinsames Merkmal ist die Anwendung des Client-Server Konzepts zur Interaktion zwischen den beteiligten Instanzen. Verteilte Anwendungen werden dabei auf unterschiedlicher Ebene unterstützt. Manche Ansätze verbergen die Verteilung hinter einem globalen Speichermodell, das die (Haupt-) Speicher aller Rechner eines Verbunds umfaßt /SmFa91/. Die derzeit am häufigsten vorzufindende Realisierungsform baut jedoch auf dem Remote Procedure Call auf /BiNe84/. Bei beiden Modellen spielt eine niedrige Latenzzeit bei der Übertragung der Daten eine sehr zentrale Rolle. Da die Zeitdauer zur Übertragung der eigentlichen Nutzdaten mit zunehmender Datenrate sinkt liegt der Versuch nahe, zeitaufwendige, mit der Datenübertragung nur indirekt verbundene Aktivitäten (wie z.B. den Aufbau einer Verbindung) für diese Anwendungen zu vermeiden. Schließlich sind erweiterte Fähigkeiten (wie Multicast und Broadcast Kommunikation) zur effizienten Unterstützung dieser Modelle notwendig.

Alle diese Eigenschaften sind von einem Transportprotokoll der Zukunft zu fordern, wobei mit zunehmender Ausdehnung Verteilter Anwendungen auch in den Bereich der Metropolitan-Area Networks (mit einer Ausdehnung einiger hundert Kilometer) oder sogar der Weitverkehrsnetze zu rechnen ist.

Bewegtbildübertragung

Die Übertragung von Bewegtbildern wird eine der Hauptaufgaben im Rahmen von Gigabit-Anwendungen sein. Dazu zählt das Bildtelefon in unterschiedlichsten Auflösungen, die Beobachtung von animierten Simulationsmodellen (Computer Visualization /Burn91/) bzw. die darauf aufbauende interaktive Simulation, die entfernte Experimentsteuerung im Bereich der Hochenergiephysik /DFN89/ und schließlich die Verteilung von z.B. Video- oder Fernsehprogrammen. Es ist dabei zu erwarten, daß alle diese Anwendungen aufgrund der hohen Bruttodatenrate über Kompressionsverfahren genutzt werden. Als Resultat der Komprimierung erhält man i.a. sehr stark schwankende Datenraten (etwa um den Faktor 100), die sich im Bereich einiger hundert Kilobit pro Sekunde bis zu einigen hundert Megabit pro Sekunde (je nach Auflösung und Qualität der Bilder) bewegen /ZWK91/. Da Bilder schritthaltend mit der Echtzeit dargestellt werden müssen besteht in der Regel keine Möglichkeit, eine gesicherte Übertragung durch die Wiederholung von Daten zu gewährleisten. Weiterhin darf die Laufzeit der Daten im Netz nicht zu stark schwanken, da ansonsten zu große Puffer zum Auffangen der Schwankungen benötigt werden. Entsprechende Maßnahmen zum Ausgleich dieser Laufzeitschwankungen könnten z.B. von einem Transportprotokoll durchgeführt werden.

Probleme konventioneller Transportprotokolle

Grundlegende Schwierigkeiten erwachsen den traditionellen Protokollen vor allem aus zwei Gründen:

- Sie wurden im Hinblick auf eine ganz spezielle Einsatzumgebung entworfen, die sich häufig nicht mit der in Hochgeschwindigkeitsnetzen vorzufindenden Umgebung deckt.
- Sie wurden zur Unterstützung der traditionellen Dienste entworfen, so daß die speziellen Anforderungen der neuen Anwendungen nicht berücksichtigt sind.

Hinsichtlich der Einsatzumgebung können weitergehend zwei Bereiche unterschieden werden: Die physikalischen Randbedingungen, unter denen Daten zu übertragen sind (d.h. die Eigenschaften des unterliegenden Mediums), und die für die Implementierung der Protokolle antizipierte Umgebung.

Traditionelle Transportprotokolle wurden im Hinblick auf Datenraten im Bereich einiger 10 kbit/s für WANs und etwa 10 Mbit/s für LANs entworfen. Aus diesem Grund war das unterliegende Medium der eigentlich limitierende Faktor, während die Bearbeitungszeit eines Pakets für die Mehrzahl der Fälle durchaus für die Geschwindigkeit der vorhandenen Hardware akzeptabel war. Ein gemeinsames Charakteristikum aller traditionellen Protokolle ist daher der Versuch, die Anzahl zu übertragender Bits zu minimie-

ren. Dabei wurde oft die Komplexität der Pakete zugunsten möglichst kompakter Darstellungen relativ groß gehalten. So findet man häufig bit-gepackte Strukturen, Felder variabler Länge mit vom Pakettyp abhängiger Position etc., die eine Decodierung sehr aufwendig machen.

Da Bitfehlerraten im Bereich von 10^{-5} bis 10^{-6} üblich waren, wurde in der Regel eine Fehlerkorrektur bereits über Teilstrecken durchgeführt. Diese Tatsache in Verbindung mit den niedrigen Datenraten führte zu relativ einfachen Fehlerkorrekturmechanismen in Transportprotokollen, die häufig das Go-back-N Verfahren anwenden. Hier werden in Reaktion auf einen Paketverlust alle auf das verlorene Paket folgenden, bereits übertragenen Pakete vom Sender wiederholt. Diese Datenmenge entspricht somit der Pfadkapazität der Verbindung, die entsprechend dem Beispiel der Einleitung in traditionellen Netzen in der Größenordung weniger Pakete liegt (es ist dabei zu beachten, daß jedes Zwischensystem etwa ein Paket in der Bearbeitung hält, so daß die Pfadkapazität in dieser Umgebung etwa gleich der Anzahl zu traversierender Zwischensysteme ist). In Hochgeschwindigkeitsnetzen dominiert jedoch die Signallaufzeit, so daß die Pfadkapazität hier auf 625.000 byte(!) ansteigt. Wie leicht zu erkennen ist, nehmen diese Datenmengen rasch kritische Werte an, die eine Verwendung einfacher Go-back-N Verfahren in Hochgeschwindigkeitsnetzen zumindest fraglich machen.

Im Hinblick auf die antizipierte Implementierungsumgebung wurde i.a. von einer engen Einbettung in das Betriebssystem des Hostrechners ausgegangen. Die sehr intensive Nutzung von Timern, Prozessen zur Strukturierung der Implementierung und Datenkopiervorgängen führt zu einer sehr starken Belastung der CPU und damit zu einer sehr geringen Leistungsfähigkeit der Implementierung /Svo89, WaMa87/. So wird in /CJRS89/ dargelegt, daß nur ca. 20% der zur Übertragung eines Pakets benötigten CPU-Zeit wirklich zur Protokollbearbeitung genutzt wird. Die restliche Zeit wird zur Realisierung unterstützender Funktionen (i.a. innerhalb des Betriebssystems) verbraucht. Während viele dieser Nachteile durch einen sorgfältigen Entwurf eliminiert werden können, sind einige Kernprobleme schon im Design der Protokolle begründet. So wurden bereits die variablen Datenfelder innerhalb der Pakete genannt, die eine Realisierung in VLSI-Technik erheblich erschweren. Weiterhin wurde beim Protokollentwurf implizit von einer seriellen Verarbeitung der notwendigen Funktionen ausgegangen. Eine parallele Ausführung wird daher oftmals schon durch die konkrete Ausprägung der Protokollautomaten erheblich erschwert. Ein zentrales Entwurfsziel moderner Transportprotokolle muß daher die Realisierbarkeit des Protokolls in VLSI-Technik sein, wobei insbesondere die Verwendung von Parallelität schon durch den Entwurf des Protokollautomaten gewährleistet sein sollte.

Hinsichtlich der Unterstützung neuer Anwendungen sind vor allem folgende Anforderungen beim Design der Protokolle zu berücksichtigen:

- Fehlerkorrekturmaßnahmen müssen auch ohne die Wiederholung von Paketen möglich sein, um den Anforderungen der Bildübertragung zu genügen.
- Daten müssen innerhalb einer begrenzten Zeitspanne beim Empfänger abgeliefert werden können. Die Varianz der Paketlaufzeit muß dabei möglichst klein bleiben.
- Kleine Datenmengen zur Unterstützung des Client-/Server-Betriebs müssen sehr einfach übertragen werden können (keine Notwendigkeit eines vorgeschalteten Verbindungsaufbaus)
- Multicast (zur Unterstützung von Gruppenkommunikation) und Broadcast (zur Lokation von Servern) müssen effizient unterstützt werden.
- Hinsichtlich der unterstützen Dienstformen sollte eine große Flexibilität gewährleistet sein. So müssen zumindest die im Abschnitt 'Entwicklungstendenzen in ISO/OSI' geforderten Formen realisierbar sein.

Nachdem nun die veränderten Randbedingungen innerhalb von Hochgeschwindigkeitsnetzen dargestellt, die dort zusätzlich zu unterstützenden Anwendungen charakterisiert und die Probleme traditioneller Transportprotokolle geschildert sind, sollen im folgenden spezielle Beispiele innovativer Protokolle vorgestellt werden. Dabei wird versucht, die Protokolle in eine Art 'Vererbungsrelation' zu stellen, indem aufgezeigt wird, welche Mechanismen in welchen Protokollen zuerst verwendet worden sind. Dazu wird ausgehend von der intendierten Einsatzumgebung die grundlegende Philosophie eines jeden Protokolls kurz erläutert, und anschließend werden die wesentlichen Innovationen des Protokolls dargestellt.

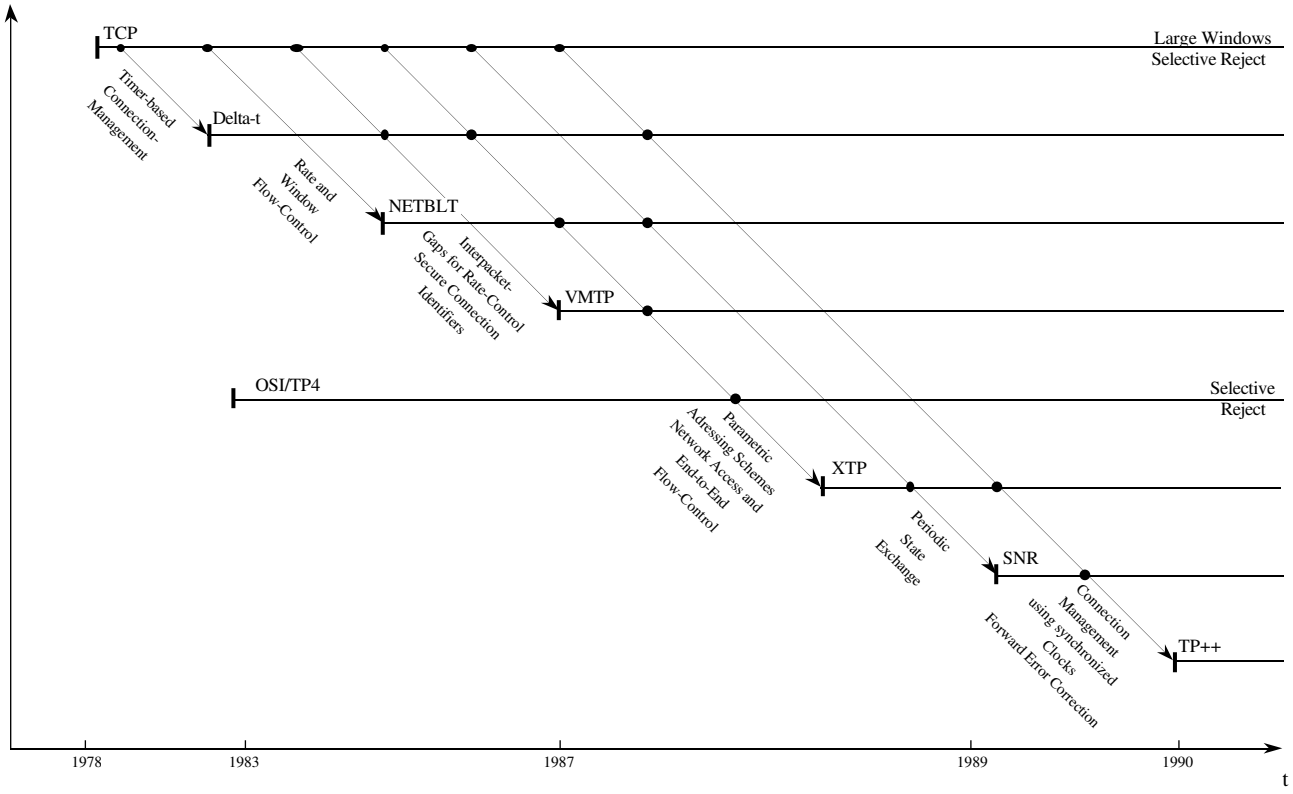


Bild 1: Evolution der Hochgeschwindigkeitsprotokolle (erweitert entsprechend einer Vorlage aus /DDKM90/)

Transportprotokolle: Mechanismen und Beispiele

Die Entwicklung innovativer Protokolle erfolgte häufig nicht primär unter dem Gesichtspunkt ihres Einsatzes in Hochgeschwindigkeitsnetzen. Vielmehr wurde i.a. von einer intendierten Einsatzumgebung ausgegangen, die meist eine bestimmte, von traditionellen Protokollen nicht oder nur ungenügend unterstützte (meist neuartige) Anwendungsform enthielt. Im Hinblick auf die Anforderungen der Anwendung wurden sodann Mechanismen definiert, welche ihre optimale Unterstützung gewährleisten konnten. Da die entsprechenden Anwendungen auch von nachfolgenden Protokollen unterstützt werden sollten, wurden vielfach bewährte Mechanismen übernommen und entsprechend des neuen Bedarfs weitere bei der Definition eines neuen Protokolls hinzugefügt. Auf diese Weise können die Protokolle in einer Art 'Vererbungshierarchie' dargestellt werden (vgl. dazu Bild 1).

In der Abbildung wird als Ausgangspunkt eines Protokolls das Datum der ersten Veröffentlichung genommen. Die diagonal laufenden Pfeile stellen die Vererbung dar, wobei Punkte angeben, von welchen Protokollen Mechanismen übernommen worden sind. Die Beschriftung an den Pfeilen bezeichnen die neuen Innovationen, die von dem Protokoll eingeführt worden sind. Im folgenden werden nun die einzelnen Protokolle kurz vorgestellt, wobei der Schwerpunkt auf der Erläuterung innovativer Mechanismen und ihrer Eignung zur Behebung der im ersten Teil dargestellten Schwierigkeiten traditioneller Protokolle liegt.

Delta-t

Seit Ende der Siebziger Jahre wurde an den Lawrence Livermore Laboratories das Transportprotokoll Delta-t /Wats89/ entwickelt. Delta-t kann auf verbindungslosen Vermittlungsschichtprotokollen aufsetzen und wurde als universelles Transportprotokoll definiert. Um die im Rahmen von Client- / Server-Anwendungen auftretenden kurzen Anfragen und Antworten effizient unterstützen zu können, wurde ein spezieller Schwerpunkt auf die Form der Verbindungsverwaltung gelegt. Ein Ziel hierbei war es, die Mindestanzahl der zur Übertragung einer kurzen Transaktion benötigten Pakete minimal halten zu können. Dazu wurde als vollkommen neues Konzept auf einen expliziten Verbindungsaufbau gänzlich verzichtet.

Verbindungsmanagement

Bei Transaktionen hat ein schneller Verbindungsaufbau eine große Bedeutung. Ein übliches Handshake-Verfahren würde gerade in Hochgeschwindigkeitsnetzen viel länger daueren als der eigentliche Austausch der gewünschten Informationen. Bei dem in der Einleitung erwähnten Hochgeschwindigkeitsnetz benötigt die Etablierung einer Verbindung ca. 20 Millisekunden. In dieser Zeitspanne hätte der Initiator jedoch bereits eine Datenmenge von 10 Mbit übertragen können!

Delta-t verwendet zur effizienten Unterstützung von Transaktionen ein implizites Verbindungsaufbau-Verfahren. Spezielle Verbindungsaufbau-Pakete werden dabei nicht übertragen, vielmehr überträgt der Sender sofort Datenpakete. Das erste empfangene Datenpaket wird vom Empfänger als implizite Aufforderung genommen, eine Verbindung zu eröffnen. Beide Instanzen setzen beim Senden oder Empfangen von Daten einen Timer auf einen Wert der groß genug ist um zu gewährleisten, daß vor dem Ablaufen des Timers alle Übertragungswiederholungen und Quittungen bei der Partnerinstanz ankommen. Nach Ablauf des Timers wird die Verbindung abgebaut. Eine kurze Anfrage kann auf diese Weise sehr rasch übertragen werden, so daß die beim expliziten Verbindungsaufbau unvermeidliche Verzögerung wirksam eliminiert werden kann. Die Verbindung wird allerdings eine ganze Zeit aufrecht erhalten, da ja ggf. eine Fehlerkorrektur durchzuführen ist. Im Normalfall erzielt man allerdings das gewünschte Verhalten.

Bei diesem Verfahren muß allerdings nach dem Verbindungsabbau verhindert werden, daß verzögerte Duplikate einer bereits abgebauten Verbindung zu einem erneuten Verbindungsaufbau führen, oder daß sie mit Paketen einer zwischenzeitlich neu aufgebauten Verbindung verwechselt werden können. Diese Forderungen werden durch die Verwendung von eindeutigen Verbindungsreferenzen erreicht, die nach einem Verbindungsabbau solange nicht weiter verwendet werden dürfen, bis sichergestellt ist, daß keine Duplikate im Netz existieren. Man spricht in diesem Zusammenhang auch vom "Einfrieren von Verbindungsreferenzen".

NETBLT

NETBLT (Network Block Transfer Protocol, /CLZ87/) wurde im Laboratory for Computer Science am Massachusetts Institute of Technology (MIT) insbesondere für den Transport großer Datenmengen mit hohem Durchsatz über Verbindungen mit großer Verzögerung konzipiert, wie sie z.B. bei der Satellitenübertragung vorkommen. Solche Strecken weisen aufgrund der hohen Laufzeit eine große Pfadkapazität auf, wie sie auch in den Hochgeschwindigkeitsnetzen vorgefunden wird. Gerade in dieser Umgebung zeigt sich, daß traditionelle Flußkontrollmechanismen den zu erreichenden Durchsatz stark limitieren können. Die als innovative Komponente im Rahmen von NETBLT entworfenen Mechanismen sind daher auch sehr gut für den Einsatz in modernen Netzen geeignet, weshalb sie von vielen der neueren Protokolle übernommen worden sind.

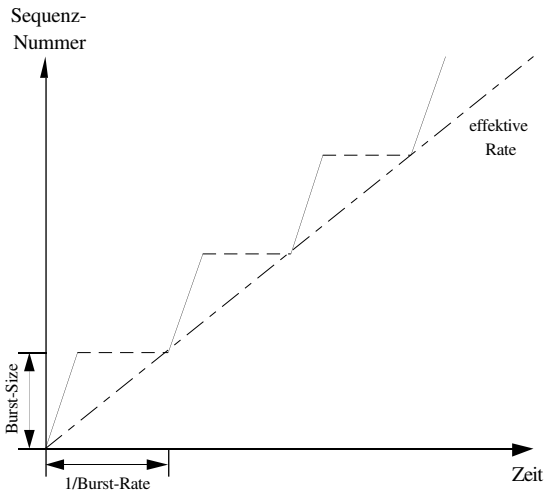
Flußkontrolle

Wird in Hochgeschwindigkeitsnetzen zur Flußkontrolle lediglich ein fensterbasiertes Verfahren verwendet (d.h. der Empfänger erlaubt dem Sender nur die Übertragung von Paketen bis zu einer bestimmten Sequenznummer), so kann ein gleichmäßiger Datenfluß ohne starke Durchsatzschwankungen nur mit kleinen Krediten erreicht werden. Eine solche Limitierung des Kredits führt jedoch dazu, daß die zur Verfügung stehende Übertragungskapazität nicht ausgenutzt werden kann. Die Speicherfähigkeit eines Netzes mit großer Pfadkapazität führt bei kleinem Kredit häufig dazu, daß der Sender durch das Warten auf Quittungen blockiert wird. Bis zum Eintreffen einer den Kredit wieder erhöhenden Quittung kann dann erheblich mehr Zeit vergehen, als der Sender für die Übertragung der restlichen Daten benötigt. Der Sender verbringt in einem solchen Fall eine relativ große Zeitspanne mit dem Warten auf einen neuen Sendekredit, obwohl der Empfänger ggf. zwischenzeitlich alle Daten verarbeitet hat und wieder in der Lage wäre, neue Daten zu empfangen.

NETBLT verwendet in Konsequenz aus dieser Tatsache als erstes Protokoll eine Kontrolle des Datenflusses mit Hilfe einer Übertragungsrate, nicht zuletzt um auch die Fehlerbehebung von der Flußkontrolle zu entkoppeln. Zur Realisierung der Übertragungsratenkontrolle werden beim Verbindungsaufbau die beiden Parameter *Burst-Size* und *Burst-Rate* zwischen den Instanzen ausgehandelt. Die Burst-Size legt dabei fest, wieviele Daten der Sender kontinuierlich übertragen darf. Die Burst-Rate definiert den Mindestabstand zwischen zwei Bursts. Zur Realisierung wird dabei angenommen, daß der Sender zu Beginn eines Bursts einen Byte-Zähler auf den Wert Burst-Size setzt, und einen Timer der Dauer 1/Burst-Rate startet. Beim Senden eines Pakets wird der Zähler um die Paketgröße dekrementiert, wobei eine Übertragung solange zulässig ist, wie dieser Zähler einen positiven Wert enthält; ansonsten wird der Sender keine weiteren Daten übertragen. Sobald der Timer abgelaufen ist, wird der Byte-Zähler wieder neu initialisiert, und der Sender kann mit der Übertragung fortfahren.

Konsequenz dieser Vorgehensweise ist natürlich, daß die Pakete innerhalb eines Bursts mit einer wesentlich höheren Geschwindigkeit als von der durch (Burst-Size • Burst-

Rate) vorgegebenen effektiven Rate übertragen werden können. Mögliche Größen sind z.B. 200 Burst/sec, bei einer Burst-Size von 5000 byte/Burst. Die über mehrere Bursts gemittelte effektive Rate folgt daher zu 200 • 5000 = 1 Mbyte/sec. Ist der Sender jedoch an ein FDDI-Netz angeschlossen und überträgt somit sein Daten mit 100 Mbit/s, so benötigt er zur Aussendung eines Bursts nur etwa 400 µs! Die restliche Zeit wartet er auf den Ablauf des Burst-Timers (vgl. nachfolgendes Bild).



Diese Vorgehensweise wurde ausgewählt, da eine Realisierung von NETBLT in Software vorgesehen war. Aufgrund der beschränkten Auflösung von Software-Timern ist es nicht möglich, Abstände zwischen jeweils zwei Pakete einzulegen und so die effektive Rate genauer zu approximieren. Als Kompromiß wurde daher ein Timer pro Burst gehalten, so daß dieser auf einen höheren Wert eingestellt werden kann.

Zusätzlich zum beschriebenen Übertragungsratenkontrollverfahren verwendet auch NETBLT weiterhin ein fensterbasiertes Verfahren, damit nicht nur die Übertragungsrate kontrolliert wird, sondern sich der Empfänger auch vor einem überlauf seiner Empfangspuffer schützen kann. Aufgrund der Ratenbegrenzung können die Fenstergrößen dann allerdings erheblich größer gewählt werden.

Selektive Quittierung und selektive Übertragungswiederholung zur Fehlerbehebung

Aufgrund der hohen Pfadkapazität ergeben sich in Hochgeschwindigkeitsnetzen desweiteren Probleme bei der Fehlerbehebung. Da innerhalb der Zeitspanne zwischen dem Verlust eines Pakets und seiner Detektion durch den Sender Daten etwa im Umfang der Pfadkapazität übertragen worden sind, müssen diese bei der Go-back-N Technik zur Fehlerbehebung alle erneut übertragen werden. Es ist offensichtlich, daß in solchen Situationen der Durchsatz drastisch reduziert wird. NETBLT verwendet aus diesem Grund eine selektive Anzeige für nicht korrekt erhaltene Pakete und eine kumulative Quittierung für die empfangenen Pakete, d.h. der Empfänger gibt an, bis zu welcher Sequenznummer alle Pakete korrekt empfangen wurden.

Die spezielle Form der Anzeige fehlender Pakete ermöglicht in NETBLT den Einsatz von selektiver Übertragungswiederholung. Im Gegensatz zum Go-back-N Verfahren muß hierbei ein Sender lediglich diejenigen Pakete wiederholen, die der Empfänger selektiv angefordert hat. Auf diese Weise wird viel weniger Übertragungskapazität bei der Fehlerbehebung benötigt. Da bei selektiver Übertragungswiederholung korrekt empfangene Daten beim Empfänger gespeichert werden müssen, erfordert diese Technik allerdings sehr große Pufferkapazitäten.

In NETBLT werden Daten neben den traditionellen Paketen in einer übergeordneten Einheit, *Puffer* genannt, übertragen. Ein Puffer umfaßt dabei eine Reihe von Paketen. Die Größe dieser Puffer wird zwischen Sender und Empfänger ausgehandelt. Trifft das erste Paket eines neuen Puffers beim Empfänger ein, so wird ein Timer gestartet, wobei der Timer so initialisiert wird, daß alle Pakete des Puffers bei fehlerfreier Übertragung vor Ablauf des Timers beim Empfänger eintreffen. Der Timerwert wird dabei vom Empfänger unter Berücksichtigung der vereinbarten Übertragungsrate und der Puffergröße geschätzt. Nach Ablauf des Timers erfolgt die selektive Anforderung der erneut zu übertragenden Daten.

VMTP

VMTP (Versatile Message Transaction Protocol) /Cher88, ChWi89/ wurde an der Stanford University als Transportprotokoll für das verteilte Betriebssystem V entwickelt. Es sollte speziell transaktionsorientierte Anwendungen wie z.B. entfernte Prozeduraufrufe (remote procedure call, RPC) unterstützen. VMTP stellt einen verbindungsorientierten Transportdienst mittels Nachrichtentransaktionen nach dem Client-/Server-Prinzip zur Verfügung. Eine Nachrichtentransaktion besteht aus einer Anforderungsnachricht (request message) an einen oder mehrere Empfänger (multicast) und einer oder mehrerer optionaler Antwortnachrichten (response message). VMTP verwendet u.a. impliziten Verbindungsaufbau, timergesteuerten Abbau, selektive Übertragungswiederholung und Übertragungsratenkontrolle ähnlich NETBLT.

Übertragungsratenkontrolle mit Interpacket-Gaps

Für die Übertragungsratenkontrolle verwendet VMTP ein etwas anderes Verfahren als NETBLT. Sender und Empfänger vereinbaren hier einen Mindestabstand (Inter-Packet Gap), der zwischen dem Absenden von zwei Paketen liegen muß. Diese Vorgehensweise wurde möglich, da VMTP von Beginn an für eine Realisierung in Hardware konzipiert worden ist, und dort Timer mit sehr hoher Auflösung unproblematisch zu realisieren sind. Der Inter-Packet Gap muß während einer Verbindung nicht konstant gehalten werden, sondern kann dynamisch verändert werden. Fordert der Empfänger relativ häufig Übertragungswiederholungen an, so kann der Sender den Abstand vergrößern und damit die Übertragungsrate reduzieren. Andererseits kann der Sender auch versuchen, den Abstand zu reduzieren und damit die Übertragungsrate zu steigern, falls Anforderungen nach Übertragungswiederholungen durch den Empfänger ausbleiben. Der Sender nimmt dann an, daß der Empfänger mindestens mit der aktuellen Übertragungsrate arbeiten kann.

Adressierung

VMTP Instanzen werden mit einer 64-Bit Adresse identifiziert, welche netzweit eindeutig, stabil, sicher und unabhängig von der Adresse des Endsystems ist. Dadurch wird die Migration von Instanzen zwischen Endsystemen wesentlich erleichtert. Diese Erweiterung ist somit als Konsequenz auf die steigende Verbreitung Verteilter Anwendungen zu sehen, die z.B. zum Lastausgleich eine dynamische Migration der beteiligten Instanzen wünschenswert macht. Um die Eindeutigkeit zu gewährleisten, muß eine Adresse vor ihrer Benutzung darauf hin überprüft werden, ob sie von einer anderen Instanz innerhalb des Netzes verwendet wird. Eine Adresse kann dann permanent stabil sein, d.h. sie ändert ihre Bedeutung nicht, oder T-stabil, d.h. vor der Änderung der Bedeutung muß sie für die Zeit T ungültig sein. Um den Wechsel der Bedeutung einer Adresse erkennen zu können, müssen die Instanzen periodisch mit einer Periodendauer < T die Gültigkeit einer Adresse überprüfen.

Implementierung

Eng mit der Entwicklung von VMTP verbunden ist der Entwurf des VMP Network Adapter Boards (NAB) /KaCh89/. Diese Netzwerkadapterkarte umfaßt eine integrierte Implementierung des VMTP-Protokolls und des Netzzugriffsprotokolls. Durch den Einsatz von Hardware und Parallelität soll eine leistungsfähige Implementierung eines kompletten Kommunikations-Subsystems zur Erbringung eines Transportdienstes erzielt werden.

Das Paketformat von VMTP wurde daher so gestaltet, daß Hardware-Implementierungen diverser Protokollfunktionen wie Entschlüsselung, Prüfsummenberechnung u.a. ermöglicht werden. Ein Paket setzt sich aus Empfängeradresse, weiteren Header-Feldern, den Benutzerdaten und der Prüfsumme zusammen. Das Format erlaubt eine fließbandartige Verarbeitung von Netzzugriff, Ent- bzw. Verschlüsselung sowie der Prüfsummenberechnung beim Empfangen bzw. beim Senden. Verschiedene Teile des Pakets können sich dabei gleichzeitig in verschiedenen Stufen des Fließbandes befinden.

XTP

Die Entwicklung von XTP (Xpress Transfer Protocol) /Ches91a, PEI92/ begann bereits im Jahre 1986, die erste Spezifikation von XTP Version 1.0 als damals trailerbasiertes Protokoll lag im Jahre 1987 vor. Kernziel des Projekts ist die Entwicklung eines VLSI-Chipsatzes, der eine Hochleistungsimplementierung des Protokolls möglich machen soll. Derzeit aktuell ist die Version 3.6, nachdem im wesentlichen die Paketformate und weniger die Protokollmechanismen teilweise tiefgreifenden Änderungen unterlagen. Derzeit wird in mehreren Normungsprojekten (ANSI, IEEE, ISO) versucht, das gesamte Protokoll oder zumindest wesentliche Protokollelemente als Standards zu etablieren. Weiterhin existieren bereits eine Reihe von XTP-Implementierungen auf unterschiedlichen Systemen. Nicht zuletzt durch das "Research Affiliate Program" von Protocol Engines Inc. wird XTP an einigen Forschungseinrichtungen und Universitäten zu Forschungszwecken eingesetzt. Ein wichtige Eigenschaft ist die Unterstützung mehrerer Adressierungsschemata. Derzeit werden u.a. IP, ISO, XNS und Netbios-Formate unterstützt, so daß der Einsatz in heterogenen Umgebungen erleichtert wird.

XTP benutzt einige von früheren Protokollen eingeführten Mechanismen wie z.B. den impliziten Verbindungsaufbau, die selektive Übertragungswiederholung, die Übertragungsratenkontrolle und Multicast-Adressierung. XTP ist im wesentlichen eine Sammlung von für Hochgeschwindigkeitsnetze tauglichen Techniken und profitierte sehr stark von früheren Protokollentwicklungen. Verschiedene Strategien und Optionen können wahlweise je nach Anwendung und Netzwerk verwendet werden. Beim Entwurf von XTP wurde weiterhin darauf geachtet, die Funktionen zum Empfangen möglichst einfach zu gestalten, da erfahrungsgemäß gerade der Empfangsteil den Engpaß bei der Protokollverarbeitung bildet. Bestimmte Funktionen wie z.B. die Steuerung des Kontrollnachrichtenaustauschs wurden daher in den Sendeteil des Protokoll verlagert.

Wählbare Fehlerbehebung

XTP bietet nicht nur verschiedene Fehlerbehebungsstrategien an, sondern erlaubt zusätzlich durch ein spezielles Bit im Paketkopf eines Informationspakets die Wahl, ob eine Fehlerbehebung überhaupt erfolgen soll. Der Verzicht auf eine Fehlerbehebung kann bei bestimmten Anwendungen wie Sprach- oder Bewegtbildübertragung sinnvoll sein, wenn die übertragenen Daten für den Empfänger eine gewisse Redundanz besitzen.

Integration von Transport- und Vermittlungsschicht

XTP ist kein reines Transportprotokoll, sondern enthält auch Funktionen der Vermittlungsschicht wie Routing und Weiterleiten von Paketen. Diese Integration hat den wesentlichen Vorteil, daß diverse Funktionen, welche oft sowohl in der Vermittlungsschicht als auch in der Transportschicht auftreten, nur einmal vorkommen. Ferner wird durch diese Zusammenfassung der beiden Schichten verhindert, daß sich der durch die Schichtung verursachte Implementierungs-Overhead hinsichtlich der Leistung negativ bemerkbar macht.

Die Übertragungskontrolle erfolgt nicht ausschließlich zwischen den Endsystemen, die Transitsystem können die entsprechenden Parameter ebenfalls beeinflussen. Wegen der Integration von Transport- und Vermittlungsschicht sind die zwischen zwei Endsystemen ausgetauschten Kontrollpakete für ein Transitsystem nicht transparent. Bei Überlastsituationen kann das Transitsystem die Übertragungsratenparameter in einem zwischen den Endsystemen ausgetauschten Kontrollpaket verändern, um die Übertragungsrate zu verringern. Das Transitsystem kann aber auch selbst ein Kontrollpaket erzeugen und an ein Endsystem senden.

Kontrolldatenübertragung

XTP kennt zwei grundsätzlich verschiedene Paketformate: Informationspakete und Kontrollpakete. Erstere enthalten im wesentlichen Benutzerdaten, während die Kontrollpakete zur Steuerung des Datenflusses vorgesehen sind und beispielsweise Parameter für die Flußsteuerung und Quittierung enthalten. Die Kontrolldaten werden also getrennt (out-of-band) von den Benutzerdaten übertragen. Diese Trennung erlaubt nicht zuletzt eine weitgehend unabhängige bzw. parallele Verarbeitung von Steuerungsinformationen und Benutzerdaten.

Implementierung

Untrennbar mit der Entwicklung von XTP verbunden ist das Protocol Engine - Projekt, in dem ähnlich wie beim VMP NAB mit Hilfe von VLSI-Komponenten und Universalprozessoren ein leistungsfähiges Kommunikationssubsystem implementiert wird /Ches91b/. Diverse Funktionen wie z.B. Fehlererkennung sowie Erzeugung und Überprüfung des Paketkopfes werden direkt in Hardware implementiert oder durch Hardware unterstützt. Dies erforderte jedoch, daß die Pakete ein für die Verarbeitung durch Hardware geeignetes Format (Ausrichtung auf 4-Byte-Worte, Felder fester Länge) besitzen. Auch die Tatsache, daß das Protokoll als eine Menge paralleler Protokollautomaten spezifiziert ist, macht XTP für eine Implementierung auf Mehrprozessor-Architekturen interessant /BSZ92/.

SNR

Hinsichtlich einer effizienten Implementierung sind nicht nur Aspekte bezüglich der Paketformate wichtig. Für eine Implementierung auf parallelen Architekturen ist es von bedeutendem Vorteil, wenn die Protokollarchitektur Parallelität unterstützt. SNR /NRS90/ wurde an den AT&T Bell Laboratories in Murray Hill im wesentlichen hinsichtlich einer parallelen Implementierung auf einer Multiprozessor-Architektur entworfen. Die wichtigste Eigenschaft von SNR ist die Trennung von Kontroll- und Benutzerdaten sowie der regelmäßige, häufige Austausch von Kontrollinformation. Ansonsten werden ähnlich wie bei XTP einige der sich für Hochgeschwindigkeitsnetze etablierten Protokollmechanismen verwendet. Je nach Anforderung durch die Anwendung werden drei Arbeitsmodi angeboten, welche sich vor allem bezüglich der Verwendung von Fehler- und Flußkontrolle unterscheiden.

Periodischer Austausch von Kontrolldaten

Wesentliches Merkmal von SNR ist der häufige, periodische Austausch von Kontrollinformation. Das Senden von Kontrolldaten (wie z.B. Quittungsdaten, Informationen über gesendete Daten und Empfangspufferbelegung) ist nicht von bestimmten Ereignissen wie dem Auftreten von Fehlern abhängig, sondern erfolgt periodisch in sehr kurzen Zeitabständen. Die Zeitabstände liegen dabei i.a. unterhalb des durchschnittlichen Zeitintervalls zwischen dem Senden zweier Pakete, so daß Sender und Empfänger aktuelle Verbindungsdaten besitzen. Durch den periodischen Austausch z.B. von Quittungen werden im Vergleich zur Quittierung auf Anforderung durch den Sender (wie bei XTP) Verzögerungen wesentlich reduziert.

Durch dieses Verfahren wird eine größere Unabhängigkeit zwischen Kontrollfunktionen und Funktionen, welche auf Benutzerdaten operieren, erreicht. Es führt auch zu einem größeren Grad an möglicher Parallelität. SNR unterstützt

Implementierungen auf parallelen Architekturen durch die Aufteilung des Protokolls in endliche Automaten mit unterschiedlichen Aufgaben.

TP++

TP++ ist ein universelles Transportprotokoll, das sich derzeit bei Bellcore (Bell Communications Research) in Morristown in Entwicklung befindet /BCFA92/. Das Protokoll soll für die Unterstützung von Multimedia-Anwendungen in leistungsfähigen Arbeitsplatzrechnern und für den Einsatz in heterogenen Gigabit-Netzen mit hoher Pfadkapazität geeignet sein. So ist auch bei TP++ ein wesentlicher Aspekt, daß möglichst viele Protokollfunktionen in Hardware realisiert werden sollen. Außerdem soll TP++ durch eine modulare Protokollarchitektur so flexibel gestaltet sein, daß es sich leicht für verschiedene Anwendungen und Netze anpassen läßt.

Optimierte timer-basierte Verbindungsverwaltung

TP++ benutzt wie Delta-t timerbasierte Mechanismen zur Verbindungsverwaltung. Sie nutzen hier jedoch als Neuerung netzweit annähernd (d.h. mit einer garantierten maximalen Abweichung) synchronisierte Uhren /Bier92/. Die Synchronisation kann durch spezielle Synchronisationsprotokolle oder beispielsweise durch Funkuhren unterstützt werden. Jedes Paket enthält einen absoluten Zeitwert, zu dem es ungültig wird. Durch die synchronisierten Uhren kann die Gültigkeit eines Pakets relativ präzise festgelegt werden. Die Verbindungsverwaltung wird dadurch erheblich vereinfacht und die Dauer, die ein Verbindungsdateneintrag nach dem Abbau einer Verbindung gespeichert werden muß, kann deutlich geringer gehalten werden. Außerdem wird das Wiederaufsetzen nach einem Zusammenbruch der Verbindungsverwaltung verkürzt.

Vermeiden von Multiplexen, Flußkontrolle

Im Gegensatz zu vielen anderen Transportprotokollen vermeidet TP++ das Multiplexen mehrerer Transportverbindungen auf eine Netzwerkschicht-Verbindung. Dieses Vorgehen folgt im wesentlichen aus dem Wunsch, eine Ende-zu-Ende Flußkontrolle zu vermeiden. Vielmehr sollen die von der Vermittlungsschicht zur Staukontrolle verwendeten Backpressure-Signale zur Flußkontrolle genutzt werden. In Konsequenz aus dieser Forderung muß das Multiplexen innerhalb der Transportschicht unterbleiben, da ansonsten alle auf eine Verbindung der Vermittlungsschicht abgebildeten Transport-Verbindungen gleich behandelt werden.

Bietet die Vermittlungsschicht keine Staukontrolle an, so sieht TP++ eine explizite Ende-zu-Ende Flußsteuerung vor. Dazu wurde eine Sende-Sequenznummer eingeführt, die für jedes zu sendende Paket einschließlich Übertragungswiederholungen inkrementiert wird. Der Empfänger teilt dem Sender jeweils mit, bis zu welcher Sende-Sequenznummer er Pakete senden darf.

Vorwärts-Fehlerbehebung

Manche Anwendungen wie beispielsweise die Bewegtbildübertragung benötigen zwar eine fehlerfreie Übertragung der Daten, können jedoch die bei Verwendung der Übertragungswiederholung entstehenden Verzögerungen nicht tolerieren. Eine auf Übertragungswiederholung gestützte Fehlerbehandlung verzögert im Fehlerfall das Eintreffen der korrekten Daten auf mindestens das Dreifache der Zeit für eine einfache Übertragung der Daten. In der Regel wird die Verzögerung eher höher sein, wenn der Verlust eines Pakets durch das Ablaufen eines Timers erkannt wird. Solchermaßen verspätet eintreffende Daten sind für die Bewegtbilddarstellung jedoch nutzlos, so daß alternative Verfahren genutzt werden müssen.

TP++ bietet für diesen Fall den Mechanismus der Vorwärts-Fehlerbehebung an. Diese wird durch das Senden zusätzlicher, redundanter Information realisiert. Werden zum Senden von k Paketen h redundante Pakete entsprechend einem geeigneten Code erzeugt, so können von den insgesamt (h+k) Paketen h beliebige Pakete verloren gehen. Der Empfänger ist dann immer noch in der Lage, aus den k empfangenen Paketen die korrekten Daten abzuleiten. Zusätzliche Übertragungswiederholungen und die damit auftretenden Verzögerungen werden vermieden.

Entwicklungstendenzen in ISO/OSI

ISO Aktivitäten

Die Entwicklung auf dem Gebiet der Hochgeschwindigkeitsprotokolle beeinflußt seit kurzem auch die Arbeiten der International Standards Organization (ISO) auf dem Gebiet der OSI-Protokolle. Bei den verschiedenen nationalen Standardisierungsorganisationen besteht weitgehend Einigkeit hinsichtlich der Notwendigkeit für neue Transportdienste und -protokolle. Die Aktivitäten werden jedoch nicht allein auf die Transportschicht begrenzt sein, viel mehr wird die Notwendigkeit für Modifikationen und Ergänzungen auf sämtlichen unteren Schichten (2-4) gesehen. Insbesondere sollen auch Vorschläge für neue,

nicht notwendigerweise an die derzeit übliche Schichteneinteilung gebundene Architekturen für die Hochgeschwindigkeitskommunikation erarbeitet werden. Die zu erwartenden neuen Protokollarchitekturen sollen jedoch nicht die derzeit gültigen Standards generell ersetzen, sondern sie lediglich in den Fällen ergänzen, in denen die aktuellen Protokolle die an sie gestellten Anforderungen nicht mehr erfüllen können.

Neue Transportdienste und -protokolle

Die ersten Anregungen für Hochgeschwindigkeits-Kommunikationssysteme betreffen sowohl den Transportdienst als auch die zur Diensterbringung erforderlichen Protokolle und deren Mechanismen. Ein erster Vorschlag für den Transportdienst sieht eine Unterscheidung der verfügbaren Dienstqualitäten in "peer-to-peer connection mode service", "peer-to-peer connectionless mode service", "multipeer connection mode service", "multipeer stream mode service" und "peer-to-peer transaction mode service" vor /ISO91a/. Eine andere Möglichkeit besteht darin, einen allgemeinen Transportdienst mit Optionen und Parametern zur flexiblen Dienstanpassung an die unterschiedlichen Anforderungen durch den Dienstbenutzer anzubieten. Dieser Ansatz erscheint hinsichtlich zukünftiger Erweiterbarkeit für zu erwartende neue Anforderungen durch die Anwendungen sehr sinnvoll.

Die Vorschläge hinsichtlich der Transportprotokollfunktionen umfassen im wesentlichen die Mechanismen, welche durch die in diesem Artikel beschriebenen Protokolle eingeführt wurden /ISO91b/. Neben der Multicast-Fähigkeit und der Unterstützung zusätzlicher Dienstqualitäten, z.B. Verzögerungsschwankungen, steht die Implementierungseffizienz im Vordergrund. Dazu sollte es möglich sein, Protokollfunktionen und deren Parameter entsprechend des zu erbringenden Dienstes zu selektieren. Ein weiterer bereits erwähnter Aspekt ist die mögliche Unterstützung durch VLSI und paralleler Protokollverarbeitung.

Zusammenfassung

Traditionellen Transportprotokollen erwachsen unter den in Hochgeschwindigkeitsnetzen herrschenden Randbedingungen eine Reihe von Problemen. Diese sind vor allem auf zwei Ursachen zurückzuführen:

- dem Design dieser Protokolle im Hinblick auf eine Einsatzumgebung, die sich nicht mit der in Hochgeschwindigkeitsnetzen vorzufindenden Umgebung deckt;
- der natürlicherweise fehlenden Berücksichtigung neuer Anwendungen, die erst aufgrund der neuerdings verfügbaren sehr hohen Bandbreite realisierbar sind.

Neue Transportprotokolle, die speziell für den Einsatz in Hochgeschwindigkeitsnetzen konzipiert sind, sollten daher zumindest folgende Randbedingungen berücksichtigen:

- Sie müssen eine sehr flexible Dienstschnittstelle anbieten, über die Anwendungen speziell auf ihre Bedürfnisse angepaßte Dienste anfordern können.
- Eine dynamische Adaption der Protokollinstanzen an ihre Einsatzumgebung ist sicherlich notwendig, da sich die Anforderungen der Anwendungen einerseits und der vom Netz zur Verfügung gestellt Dienst andererseits während der Dauer einer Verbindung relativ weitreichend ändern können.
- Die Realisierbarkeit in VLSI-Technik sollte berücksichtigt werden, wobei hier insbesondere auch Aspekte der parallelen Verarbeitung eine sehr große Rolle spielen.

Es sollte ferner beachtet werden, daß zur Realisierung einer Anwendung auch die über die Ebene 4 hinausgehenden Schichten mit berücksichtigt werden müssen. Im Hinblick auf Hochleistungsimplementierungen, welche die von der Transportschicht bereitzustellende Datenrate auch an die Anwendung weiterreichen können, wurden bisher jedoch noch sehr wenige Untersuchungen durchgeführt.

Literatur

/BCFA92/ Ernst W. Biersack, C. J. Cotton, David C. Feldmeier, Anthony J. McAuley, W. D. Sincoskie: "Gigabit Networking at Bellcore", IEEE Network Magazine, vol.6, no. 2, March 1992 (to appear)

/Bell92/ Trudy E.Bell: "Telecommunications" IEEE Spectrum, January 1992, pp.36- 38

/Bier91/ Ernst W. Biersack: "Connection Management Using Synchronized Clocks", Andre Danthine, Otto Spaniol (Hrsg.): Proceedings of the 3rd IFIP WG 6.4 Conference on High Speed Networking, Berlin, March 1991, pp. 225- 239

/BiNe84/ A.D. Birell, B.J. Nelson: "Implementing Remote Procedure Calls", ACM Transactions on Computer Systems, vol.2, no.1, February 1984, pp.33- 59

/BSZ92/ T.Braun, B.Stiller, M.Zitterbart: "XTP and VMTP on Multiprocessor Architectures", Proceedings of the International Workshop on Advanced Communications and Applications for High-Speed Networks, München, March 1992

/Burn91/ Mathew Burns: "Scientific Visualization Wars", Supercomputing Review, October 1991, pp.31- 35

/Cher88/ David R. Cheriton: "VMTP: Versatile Message Transaction Protocol", Protocol Specification, Network Working Group, Request for Comments: 1045, Februar 1988

/Ches91a/ Greg Chesson: "The Evolution of XTP", Andre Danthine, Otto Spaniol (Hrsg.): Proceedings of the 3rd IFIP WG 6.4 Conference on High Speed Networking, Berlin, März 1991, pp. 15-24

/Ches91b/ Greg Chesson: "The Protocol Engine Chipset", Transfer, vol. 4, no. 3, May/June 1991, pp. 3- 8

/ChWi89/ David R. Cheriton, Carey L. Williamson: "VMTP as the Transport Layer for High-Performance Distributed Systems", IEEE Communications Magazine, vol. 27, no. 6, June 1989, pp. 37- 44

/CJRS89/ D.D.Clark, V. Jacobsen, R.Romkey, H.Salwen: "An Analysis of TCP processing overhead", IEEE Communications Magazine, vol.27, no. 6, June 1989, pp.23- 29

/CLZ87/ David D. Clark, Mark L. Lambert, Lixia Zhang: "NETBLT: A High Throughput Transport Protocol", Proceedings of the ACM SIGCOMM '87, pp.353-359

/DDKM90/ W. Doeringer, D.Dykeman, M.Kaiserswerth, B.Meister, H.Rudin, R.Williamson: "A Survey of Light-Weight Transport Protocols for High-Speed Networks", IEEE Trans. on Communications, vol.38, no.11, November 1990, pp.2025- 2039

/DFN89/ Deutsches Forschungsnetz - DFN - Projektplan für das Hochgeschwindigkeits-datennetz HDN, Version 1.6, November 1989

/HeRo91/ Ch. Hemrick, J. McRoberts: "SMDS: The Beginning of WAN Superhighways", Data Communications, April 1991, pp.105- 110

/ISO91a/ ISO/IEC JTC1/SC6 N 7070: Contribution from the USA National Body to ISO/IEC JTC/SC6 - Proposed Working Draft for High-Speed Transport Service Definition, November 1991

/ISO91b/ ISO/IEC JTC1/SC6 N 7071: Contribution from the USA National Body to ISO/IEC JTC/SC6 - Work Description in Support of Enhanced Communication Functions for OSI, November 1991

/KaCh88/ Hermant Kanakia, David R. Cheriton: "The VMP Network Adapter Board: High-Performance Network Communication for Multiprocessors", ACM SIGCOMM'88, 1988, pp. 175-187

/NRS90/ Arun N. Netravali, William D. Roome, Krishan Sabnani: "Design and Implementation of a High-Speed Transport Protocol", IEEE Transactions on Communications, vol. 38, no. 11, November 1990, pp. 2010-2024

/Part90/ Craig Partridge: "How Slow is one Gigabit per Second?" ACM Computer Communication Review, vo.20, no.1, January 1990, pp. 44-53.

/PEI92/ Protocol Engines Inc.: XTP Protocol Definition, Version 3.6, PEI-Doc. 92-10

/Rick92/ Neil Rickard: "Frame Relay: The best of both worlds?", Communications International, February 1992, pp.46- 48

/SmFa91/ Jonathan M.Smith, David J.Farber: "Traffic characteristics of a distributed memory system" Computer Communications and ISDN Systems, 22(1991), pp. 143-154

/Svo89/ Liba Svobodova: "Implementing OSI Systems", IEEE Journal on Selected Areas in Communications, vol. 7, no.9, Sept. 1989, pp. 1115-1130

/WaMa87/ Richard W. Watson, Sandy A. Mamrak: "Gaining Effciency in Transport Services by Appropriate Design and Implementation Choices", ACM Trans. on Computer Systems, vol.5, no.2, May 1987, pp.97- 120

/Wats89/ Richard W. Watson: "The Delta-t Transport Protocol: Features and Experience", H. Rudin, R. Williamson (Hrsg.), Protocols for High Speed Networks, Elsevier Science Publishers B. V. (North Holland), IFIP, 1989, pp. 3-18

/ZWK91/ Ya-Qin Zhang, William W. Wu, Kap. S. Kim, Raymond L.Pickholtz, Jay Ramasastry: "Variable Bit-Rate Video Transmission in the Broadband ISDN Environment", Proceedings of the IEEE, vol.79, no.2, February 1991, pp. 214- 221