

Zeitschrift: Bulletin des Schweizerischen Elektrotechnischen Vereins, des Verbandes Schweizerischer Elektrizitätsunternehmen = Bulletin de l'Association suisse des électriciens, de l'Association des entreprises électriques suisses

Herausgeber: Schweizerischer Elektrotechnischer Verein ; Verband Schweizerischer Elektrizitätsunternehmen

Band: 84 (1993)

Heft: 3

Artikel: Sieben OSI-Schichten sind oft zuviel : Rechnernetzmittel Buskopplern, Teil 1

Autor: Stein, Erich

DOI: <https://doi.org/10.5169/seals-902663>

Nutzungsbedingungen

Die ETH-Bibliothek ist die Anbieterin der digitalisierten Zeitschriften. Sie besitzt keine Urheberrechte an den Zeitschriften und ist nicht verantwortlich für deren Inhalte. Die Rechte liegen in der Regel bei den Herausgebern beziehungsweise den externen Rechteinhabern. [Siehe Rechtliche Hinweise.](#)

Conditions d'utilisation

L'ETH Library est le fournisseur des revues numérisées. Elle ne détient aucun droit d'auteur sur les revues et n'est pas responsable de leur contenu. En règle générale, les droits sont détenus par les éditeurs ou les détenteurs de droits externes. [Voir Informations légales.](#)

Terms of use

The ETH Library is the provider of the digitised journals. It does not own any copyrights to the journals and is not responsible for their content. The rights usually lie with the publishers or the external rights holders. [See Legal notice.](#)

Download PDF: 19.03.2025

ETH-Bibliothek Zürich, E-Periodica, <https://www.e-periodica.ch>

Bei der Kommunikation zwischen Rechnern stehen heute lokale Netzwerke (LAN) und grossflächig ausgedehnte Netzwerke (MAN, WAN) im Vordergrund. Eine weitere Möglichkeit ist die direkte Kopplung der Bussysteme verschiedener Rechner über Buskoppler. Diese bietet im Vergleich zu den erstgenannten Möglichkeiten spezifische Vorteile, aber auch Einschränkungen. In diesem Beitrag wird zunächst die direkte Buskopplung im übergeordneten Zusammenhang der Nutzung und Realisierung der Rechnerkommunikation beleuchtet. In einem zweiten Teil wird dann ein Überblick über den Aufbau und die Anwendung von Buskopplern gegeben. Zum Abschluss werden einige Anwendungen näher betrachtet.

Sieben OSI-Schichten sind oft zuviel

Rechnerkommunikation mittels Buskopplern, Teil 1

■ Erich Stein

Die Kommunikation zwischen Rechnern wird häufig über lokale Netzwerke (LAN) realisiert. Hier soll eine alternative Realisierung der Rechnerkommunikation durch die direkte Kopplung von Bussystemen betrachtet werden. Die Bedeutung dieser Kopplung ist durch die zunehmende Verbreitung verschiedenartiger Bussysteme bei gleichzeitig wachsenden Anforderungen an die Bandbreite der Kommunikationspfade gestiegen. Die direkte Buskopplung stellt häufig ein elegantes Mittel dar, Systeme mit speziellen Anforderungen auf einfache Weise zu realisieren.

Im folgenden wird zunächst kurz dargestellt, wozu die Rechnerkommunikation genutzt werden kann und welche grundsätzlichen Möglichkeiten zu ihrer Realisierung es gibt. Eine Charakterisierung der wichtigsten Bussysteme leitet über zur Diskussion von Situationen, in denen man Bussysteme direkt miteinander verbinden möchte. Abschliessend werden Aufbau und Eigenschaften von Buskopplern und die zu ihrem Betrieb erforderliche Software systematisch dargestellt.

Grundsätzliches zur Nutzung der Rechnerkommunikation

Verteilte Verarbeitung

Bei einer abgegrenzten Aufgabenstellung, beispielsweise bei der Informationsverarbeitung in einem Unternehmen, könnte man zu-

nächst versuchen, alle Aufgaben auf einem einzigen Rechner zu lösen. Zu Beginn der elektronischen Datenverarbeitung hat man das auch so gemacht, und entsprechend grosse Mainframe-Computer eingesetzt. Bei zunehmendem Umfang der Aufgaben traf man jedoch auf Probleme wie hohe Komplexität, begrenzte Ausfallsicherheit, Grenzen der Rechnerleistung und des Speichervolumens. Zur Lösung dieser Probleme wurde die Alternative, Teilaufgaben auf mehrere Rechner – genauer gesagt auf mehrere Rechnerknoten in einem Netzwerk – zu verteilen, konzipiert und in verschiedene Richtungen entwickelt. Man spricht deshalb von verteilten (Rechner)Systemen oder einfach von verteilter Verarbeitung. Wedekind [1] unterscheidet folgende Ansätze zur Verteilung von Aufgaben:

- Verteilung von Funktionen
- Verteilung von Daten
- Verteilung von Lasten

Das Bild 1 versucht, diese drei Fälle anhand eines Rechnernetzwerkbeispiels zu veranschaulichen.

Verteilung von Funktionen

Die Verteilung von Funktionen hat beispielsweise dazu geführt, dass Applikationen von Mainframe-Computern auf Workstations oder gar auf Personal Computer ausgelagert wurden. Dieser Vorgang wird als Downsizing oder neuerdings auch als Rightsizing (sinnvolles Downsizing) bezeichnet. Allerdings konnte damit der Mainframe-Computer nicht überflüssig gemacht werden.

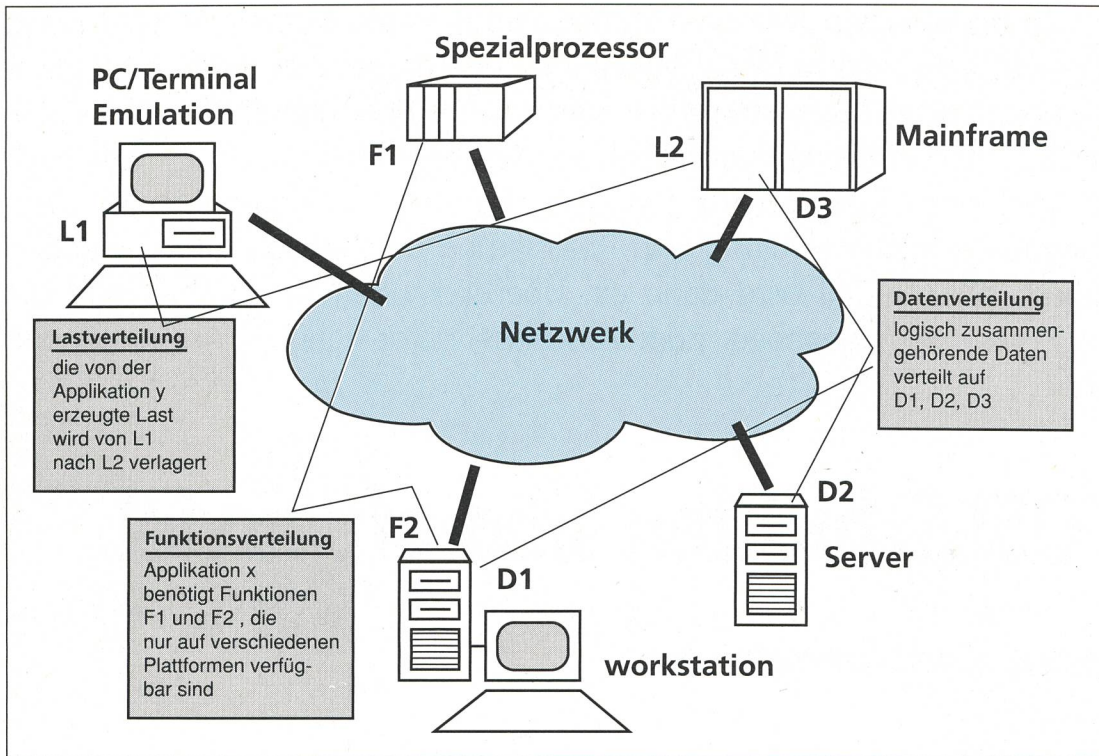


Bild 1: Nutzung der Rechnerkommunikation

Es muss deshalb in diesem Zusammenhang auch die Client-Server-Architektur diskutiert werden, bei der Teile einer Applikation auf verschiedenen Rechnern – meist verschiedener Grösse – laufen und zusammenarbeiten. Clients nennt man die Arbeitsplatzcomputer, Server die übergeordneten Computersysteme, welche die verschiedenen Clients bedienen.

Verteilung von Daten

Die Verteilung von Daten ist heute in Workstation- und PC-Netzwerken gängige Praxis. Die Daten werden auf verschiedenen Fileservern gehalten, welche dank eines verteilten Filesystems logisch als ein einheitlicher, grosser Speicher betrachtet werden, der allen Netzwerkknoten zur Verfügung steht. Eine Weiterentwicklung dieses Konzepts führt zu verteilten Datenbanken. Bei diesen sind Daten, die zu einer einzigen logischen Datenbank gehören, auf mehreren Netzwerkknoten verteilt gespeichert, ohne dass der Benutzer die Tatsache oder die Art der Verteilung kennen muss. Das bedeutet, dass Datenbankabfragen auf genau dieselbe Weise wie bei einer nichtverteilten Datenbank möglich sind.

Die Verteilung von Daten führt fast zwangsläufig zur Replikation (Mehrfachspeicherung) von Daten, sei es, weil man dadurch die Ausfalltoleranz erhöhen möchte, sei es, weil man Zugriffe beschleunigen und Kommunikationskosten senken möchte. Eine Konsequenz der Replikation ist das Problem der Datenkonsistenz. Wenn Daten verändert werden, muss sichergestellt werden,

dass alle Replikas nachgeführt werden. Dies ist für grosse Netzwerke (viele Knoten, grosse Anzahl von Replikationen) und unter der realistischen Annahme, dass Knoten ausfallen und Kommunikationspfade unterbrochen sein können, ein nichttriviales Problem.

Verteilung von Lasten

Bei der Verteilung von Lasten geht es darum, Verarbeitungsaufträge der Benutzer (die

Lasten) so auf die vorhandenen Mittel zu verteilen, dass ein vereinbartes Qualitätskriterium, zum Beispiel Antwortzeit, Verfügbarkeit oder Durchsatz erfüllt wird. Viele Betriebssysteme enthalten ausgefeilte Algorithmen, welche die Lastverteilung in Abhängigkeit von Parametern, zum Beispiel Prioritäten, ausführen. Ein aktuelles Lastverteilungsmodell [2] wurde von seinen Entwicklern Ecology of Computation genannt. Es geht von der folgenden Beobachtung aus. In ausgedehnten Netzwerken mit Workstations ergibt sich häufig die Situation, dass ein wesentlicher Teil der Workstations nicht aktiv ist, sondern auf den Benutzer wartet. Andererseits sind einige Workstations mit grossen Programmen beschäftigt und haben entsprechend lange Antwortzeiten. Für diese Workstations wäre es interessant, die brachliegende Rechenzeit anderer Maschinen zu nutzen. Entsprechende Konzepte, die analog zum Marktmechanismus auf der Basis von Angebot und Nachfrage eine Lastverteilung bewirken, wurden vorgeschlagen und untersucht.

Glossar

Buskoppler: Oberbegriff für Anordnungen, die verschiedene Busse (Backplanes) miteinander verbinden.

Verteilte System: Rechnersysteme, bei denen Anwendungen durch mehrere, kooperierende Prozessoren – auch als (Prozessor)Knoten bezeichnet – abgearbeitet werden.

Multiprozessorsystem: ein System mit mehreren Prozessoren, die untereinander gekoppelt sind. Multiprozessoren sind ein Beispiel für verteilte Systeme.

Shared Memory System: Multiprozessorsystem, dessen Prozessoren über einen gemeinsamen Speicherbereich miteinander kommunizieren. Gegensatz: Message Passing System.

Message Passing System: Multiprozessorsystem, dessen Prozessoren getrennte Speicher besitzen und mittels Nachrichtenaustauschs miteinander kommunizieren.

Anforderungen an die Kommunikation

Alle Ansätze der verteilten Verarbeitung stellen Anforderungen an die Kommunikation der einzelnen Rechner untereinander. Global kann man sagen, dass mit dem Grad der Verteilung auch die Anforderungen an die Kommunikation steigen. Dadurch wird die Weiterentwicklung der Rechnerkommunikation angetrieben. Aktuelle Prognosen gehen beispielsweise davon aus, dass in wenigen Jahren grossflächige Netzwerke (MAN, WAN) die gleichen, hohen Datenraten anbie-

ten werden wie Hochgeschwindigkeits-LAN. Die verbesserte Kommunikation fordert jedoch neue Anwendungsmöglichkeiten heraus, so dass im Sinn einer positiven Rückkopplung die Entwicklung beschleunigt wird und schliesslich völlig neuartige Konzepte der verteilten Verarbeitung entstehen, beispielsweise die oben skizzierte Ecology of Computation.

Grundsätzliches zur Realisierung der Rechnerkommunikation

Aus der Aufgabenstellung der Rechnerkommunikation ist ersichtlich, dass nicht Rechner miteinander kommunizieren sollen, sondern Applikationen (Anwendungsprogramme), die auf den beteiligten Rechnern laufen¹. Die Realisierung verlangt deshalb nicht nur Hardware für die physikalische

Übermittlung von Informationen, sondern auch Kommunikationssoftware zur Anbindung der Anwendungsprogramme.

Topologien

Auf der physikalischen Ebene steht eine Reihe von Übertragungsmedien zur Verfügung (vgl. zweite Kolonne der Tab. I), die hier nicht weiter behandelt werden sollen. Für den Aufbau von Netzwerken existieren darüber hinaus verschiedene Topologien (Bild 2) [3]. Die eindimensionalen Topologien werden bevorzugt in LAN eingesetzt, also dort wo mittlere Distanzen von einigen 10 m bis zu einigen 100 m überbrückt werden müssen. Zweidimensionale Topologien findet man häufig dort, wo einfache Prozessorelemente in grosser Anzahl bei der Lösung spezieller Aufgaben, wie beispielsweise von numerischen Berechnungen, parallel arbeiten. Die Realisierung bietet sich insbesonde-

re auf der Ebene der Silizium-Chips (LSI, VLSI) und der Silizium-Wafer (Wafer Scale Integration) an. Die überbrückte Distanz liegt bei einigen 10 mm. Dreidimensionale Topologien sind für allgemein einsetzbare Multiprozessorsysteme von Interesse. Die Distanzen betragen hier bis zu einigen Metern, da schon die einzelnen Prozessoren ganze Racks füllen können.

Kommunikationssoftware und OSI-Modell

Die logische Ebene, die Kommunikationssoftware, gibt Probleme auf, insbesondere wenn man eine zuverlässige Kommunikation in heterogenen Netzen (Netze, die Produkte verschiedener Hersteller, also unterschiedliche Rechner- und Betriebssysteme enthalten) sicherstellen möchte. Die Komplexität dieser Aufgabe legt eine Zergliederung in Teilaufgaben nahe, die beispielsweise im ISO/OSI-Modell² der Kommunikation in offenen Systemen durchgeführt wurde. In diesem Modell repräsentiert Schicht 1 die physikalische Ebene, die oben bereits angesprochen wurde. Die Schichten 2-6 befassen sich mit zunehmend allgemeineren Funktionen der Kommunikation. Schicht 7 ist die Anwendungsschicht, welche die Schnittstelle zum Anwender oder zum Anwendungsprogramm bildet. Für diese Schicht ist von ISO eine Reihe von Kommunikationsfunktionen – sogenannte Dienste – beschrieben worden, die durch standardisierte Protokolle implementiert werden. Die Tabelle II umschreibt die wichtigsten Dienste [4]; diese sind heute auf vielen auf dem Markt verfügbaren Rechnerplattformen bereits verfügbar.

Anhand der lokalen Netzwerke (LAN) soll die Bedeutung und Vielfalt der Kommunikationssoftware illustriert werden. Die betrachtete Situation ist idealisiert in Bild 3 dargestellt. Man betrachtet zwei Rechner³ mit jeweils einem eigenen Betriebssystem, die durch das Kommunikationsnetzwerk, das ein LAN oder ein LAN/WAN-Internetwork sein kann, verbunden sind. Man unterscheidet netzwerkfähige Betriebssysteme und Netzwerkbetriebssysteme (Network Operating System, NOS). Erstere können nur Nachrichten austauschen, ohne dass eine Kooperation stattfindet, während letztere miteinander kooperieren. DOS ist ab Version 3.1 ein netzwerkfähiges Betriebssystem, während Networkware, LAN Manager und Vines als Netzwerkbetriebssysteme zu verstehen sind.

Kommunikationsmittel	Übertragungsmedium	Distanz, Datenrate 1)	Protokolle, Standards	Funktionalität, Dienste
Buskoppler	Mehrleiter-Kabel	10 m 20 MByte/s		Funktionalität der beiden Busse + Zusatzfunktionen (vgl. Tab. III, IV)
	Optische Faser	2 km 20 MByte/s		
Spezialisierter Parallelbus	Mehrleiterkabel	20 m 1 MByte/s	IEEE 488 für Messgeräte, ANSI X3.131 (SCSI) für Massenspeicher, Peripheriegeräte	Standard-Commands für bestimmte Anwendungsbereiche
Serielle Verbindungen (Seriellbusse, Punkt zu Punkt, Stern)	Zweidrahtleitung	300 m, 1 Mbit/s	RS-232 RS-485	Feldbusse, z.B. Bitbus, Canbus, Profibus, MMS 2)
	Koaxialkabel	einige km, einige 100 kbit/s		
	Optische Faser			
LAN (Local Area Network)	Zweidrahtleitung (UTP)	500 m, 1 Mbit/s 100 m, 10 Mbit/s	IEEE 802.3 1 BASE 5 10 BASE T	Funktionalität des Netzwerk-Betriebssystems und zusätzlicher Kommunikationssoftware
	Koaxialkabel	500 m, 10 Mbit/s 3)	Ethernet (IEEE 802.3), Token Ring (IEEE 802.5)	
	Optische Faser	2 km, 100 Mbit/s 4)	FDDI (ANSI X3T9.5)	
MAN (Metropolitan Area Network)	Optische Faser	einige 10 km bis ca. 150 Mbit/s	DQDB (IEEE 802.6) B-ISDN	
WAN (Wide Area Network)	Telefonnetz inkl. Richtfunk, optische Fasern, Satellitenstrecken	Weltweit bis ca. 2 Mbit/s	CCITT	

Tabelle I Grundsätzliche Möglichkeiten zur Realisierung der Rechnerkommunikation

- 1) Anhaltswerte. Kompromiss zwischen Distanz und Datenrate beachten
- 2) MMS (Manufacturing Message Service) auf Schicht 7 von MAP-Netzwerken (Manufacturing Automation Protocol)
- 3) pro Segment
- 4) zwischen benachbarten Stationen

¹ Die Mensch-Maschine-Kommunikation und die über Rechnernetze hergestellte Kommunikation zwischen Personen, beispielsweise durch Electronic Mail, Voice Mail oder Groupware, wird hier nicht betrachtet.

² ISO International Standardization Organization, OSI Open System Interconnection

³ Reale LAN umfassen in der Regel bis 100 Rechner.

Dienstelement bzw. Dienst	Funktion
Association Control Service Element (ACSE)	Elementare Verbindungen zwischen Anwendungsinstanzen
Commitment, Concurrency and Recovery Service Element (CCRSE)	Verteilte Transaktionsverarbeitung
Reliable Transfer Service Element (RTSE)	Zuverlässiger Datentransfer (Transfer vollständig und genau einmal)
Remote Operations Service Element (ROSE)	Entfernter Operationsaufruf für die asymmetrische Zusammenarbeit zwischen Client und Server
File Transfer, Access and Management (FTAM)	Entfernter Dateizugriff / Dateiverwaltung
Job Transfer and Manipulation (JTM)	Initiiert, beobachtet und steuert die Verarbeitung auf einem entfernten Rechner
Virtual Terminal (VT)	Zugriff eines lokalen Terminals auf eine entfernte Anwendung ohne gegenseitige Kenntnis der Interna
Message Handling Service (MHS)	Austausch von Meldungen zwischen Anwendungsprozessen
Manufacturing Message Service (MMS)	Definiert Format und Bedeutung von Meldungen für die Fabrikautomation
Directory Service (DS)	Verzeichnisdienst zur Abbildung von Namen auf Adressen

Tabelle II Dienste der Anwendungsschicht

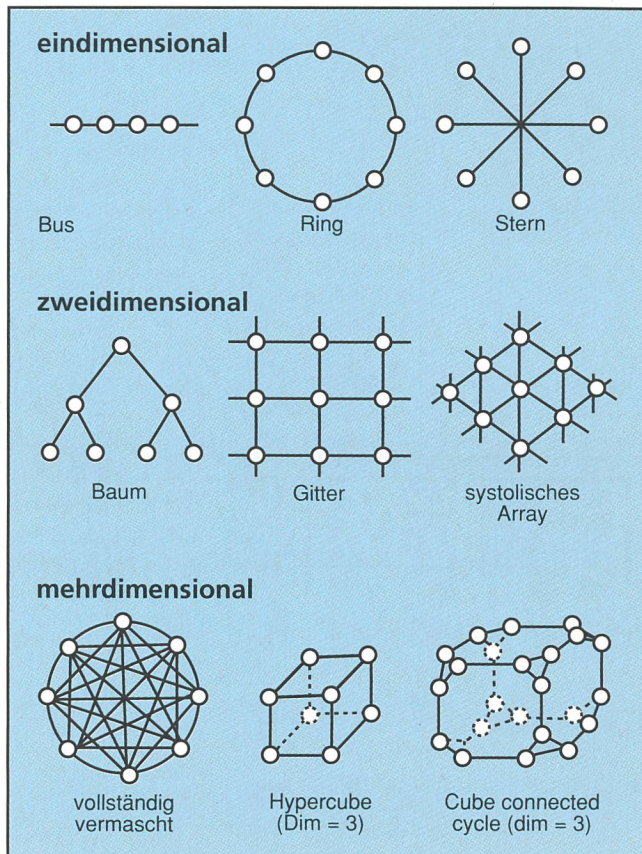
Abbildung von Netzwerksoftware-Produkten auf das ISO/OSI-Schichtenmodell

Zum Verständnis und zur Einordnung verschiedener Netzwerksoftware-Produkte ist es sinnvoll, diese auf das ISO/OSI-Schichtenmodell abzubilden⁴. Bild 4 versucht die wichtigsten Konzepte zusammenfassend darzustellen. Die Angaben sind als Beispiele zu verstehen, erheben keinen Anspruch auf Vollständigkeit und enthalten keinerlei Wertung. Details zu einigen wichtigen Netzwerksoftware-Produkten finden sich in [6] und [7]. Auf der horizontalen Achse werden zunächst zwei Anwendungsbereiche unterschieden: einerseits der Büro- und der Ingenieurbereich und andererseits der Bereich Leittechnik und Prozessautomatisierung. Eine feinere Unterteilung orientiert sich zweckmässig an der eingesetzten Plattform. Hier werden DOS (umfasst hier als Oberbegriff auch MS-Windows und OS/2), Unix (auf Workstations wie auf PC), Real-Time-Plattformen und weitere unterschieden. Die vertikale Achse umfasst die sieben Schichten des ISO/OSI-Referenzmodells der Kommunikation in offenen Systemen. Im folgenden werden die Blöcke (1-6) kurz erläutert. Block (1) enthält die wichtigsten Alternativen für die physikalische und die Sicherungsebene. Dies sind die weitverbreiteten Konzepte wie Ethernet, Token Ring und FDDI. DQDB wird vom IEEE als Standard für die Realisierung von MAN (Metropolitan Area Networks) vorangetrieben. Der Token Bus ist hauptsächlich für die Fabrikautomatisierung (Manufacturing Automation Protocol, MAP) als Transportmedium vorgesehen. In Block (2) wird DOS als Plattform eingesetzt. Hier bietet Netbios (Network Basic Input Output System) die Funktionalität der Schichten 3-5. Das Betriebssystem DOS selbst ist gemäss [7] der Schicht 6 zugeordnet. Neben DOS wird noch der Redirector – auch als Netzwerkschale bezeichnet – benötigt. Er erkennt, welche Anforderungen an das Betriebssystem lokal erfüllt werden und welche an das Netzwerk weiterzuleiten sind. Der Schicht 7 sind die File Server Software und Hilfsprogramme zugeordnet. Das heisst, der File Server ist als DOS-Applikation realisiert. Seine Leistungsfähigkeit ist jedoch infolge der bekannten Einschränkungen von DOS (nur Single Task, kleiner Adressraum, ineffizientes Filesystem usw.) eingeschränkt. Deshalb sind Netzwerkbetriebssysteme wie Novell Netware von Grund auf auf die Serverfunktionalität ausgerichtet. Sie überdecken damit die Schichten 3-7.

Die TCP/IP-Protokollfamilie Block (3) wurde auf Veranlassung des Department of

⁴ Für die Darstellung des Aufbaus des eigentlichen Betriebssystems ist das ISO/OSI-Modell nicht geeignet. Für diesen Zweck gibt es andere Schichtenmodelle wie zum Beispiel [5].

Bild 2 Topologien für die Rechnerkommunikation



Defense entwickelt. Die offene Kommunikation in WAN (Wide Area Networks) mit heterogenen Knoten war von Anfang an eine zentrale Zielsetzung. TCP/IP ist ab der Unix-Version BSD 4.2 in Unix integriert, steht heute aber auf praktisch allen Plattformen zur Verfügung. Gegenüber der OSI-Protokollfamilie, die dieselbe Zielsetzung verfolgt, hat TCP/IP einen zeitlichen Vorsprung von mehreren Jahren und eine wesentlich geringere Komplexität.

Auf Block (4), der Plattformen mit Real-Time-Betriebssystemen umfasst, kann hier nicht weiter eingegangen werden. Immerhin

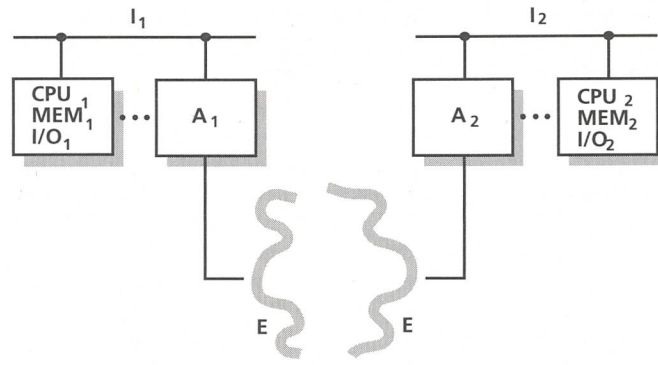


Bild 3: Generelle Anordnung für die Rechnerkommunikation

- I Rechnerinternes Kommunikationssystem (Bussystem, Backplane)
- E externes Kommunikationssystem
- A Kommunikationsadapter
- Fall Busadapter: $I_1 \neq I_2$, E = direkte, feste Verbindung
- Fall Busrepeater: $I_1 = I_2$, E = direkte, feste Verbindung

		Büro- und Ingenieurbereich			Leittechnik, Prozessautomatisierung	
		②	③	④	⑤	⑥
Schicht		DOS als Plattform	Unix als Plattform	Realtime-BS als Plattform	Prozessautomatisierung (Bsp: Profibus)	Buskoppler
7 Anwendung		File Server, Utilities		MMS, ...	FMS LLI DIN 19245 Teil 2	Treiber (vgl. Tab. IV)
6 Darstellung		DOS, Redirector			—	—
5 Session				Realtime-Betriebssysteme (OS/9, VRTX, PDOS, LynxOS, ...)	—	—
4 Transport		Netbios	Telnet, Remote Terminal Protocol, File Transfer Protocol, Simple Mail Transfer Protocol, Name Server Protocol		—	—
3 Netzwerk			TCP Transmission Control Prot., UDP		—	—
2 Sicherung		ISO 8802.2 LLC (Logical Link Control)		LLC	LLC MAC	Treiber
1 Bitübertragung		ISO 8802.3 Ethernet 1)	ISO 8802.4 Token Bus 1), ISO 8802.5 Token Ring 1), IEEE 802.6 DQDB 1)	ISO 9314 FDDI 1)	DIN 19245 Teil 1 (RS-485)	Busadapter

1) physikalische Ebene und MAC (Media Access Control) ①

Bild 4 Einordnung von Netzwerksoftware in das ISO/OSI-Modell

- UDP User Datagram
- MMS Manufacturing Message Specification
- LLI Lower Link Interface
- MAC Media Access Control
- FMS Fieldbus Message Specification

ergeben sich bezüglich der eingesetzten Protokolle und der Kommunikationssoftware keine grundsätzlichen Abweichungen gegenüber den Blöcken (2) und (3). Bei der Prozessautomatisierung auf Fertigungszellen- und Feldebene Block (5) möchte man ausser den Schichten 1 und 2 nur noch die Schicht 7 haben, um ein möglichst gutes Real-Time-Verhalten zu erreichen. Die Transportschicht nach ISO entfällt, weil sie die benötigte Broadcast- und Multicast-Adressierung nicht erlaubt. Bei den Buskopplern Block (6), die im zweiten Teil dieser Arbeit behandelt werden, wird in der Regel nur Software für die Schicht 2 zur Verfügung gestellt, das heisst diese Schicht hat ein direktes Interface zur Applikationssoftware. Es gibt jedoch zunehmend auch Toolsets, die Funktionen höherer Schichten umfassen (vgl. Teil 2 dieses Beitrags).

Die in Bild 4 eingetragenen Beispiele sind nach ihrem Ursprung teilweise als herstellerspezifische und teilweise als betriebssystemspezifische Protokollfamilien zu verstehen. Heute ist diese Abgrenzung nicht mehr wesentlich, und in Zukunft wird sie belanglos werden, denn die wichtigsten Protokollfamilien werden von vielen Herstellern auf den gängigen Plattformen angeboten.

Nachteile des ISO/OSI-Modells

Dem Vorteil der offenen Kommunikation nach dem ISO/OSI-Modell und den zugehörigen ISO-Normen steht eine Reihe von Nachteilen gegenüber. So ist die Abarbeitung der Protokolle aufwendig, was die Datenrate verringert. Die vorgegebene Menge von Standardfunktionen schränkt die Freiheit der Kommunikation ein, das heisst nicht alle vorstellbaren Kommunikationsbeziehungen sind möglich. Da ein Computer das Innenleben des anderen nicht kennt (es ist ja gerade das Ziel der normierten Protokolle, die Details des Systemaufbaus zu verdecken und damit für die Kommunikation belanglos zu machen), spricht man von einer lockeren Kopplung der Rechnersysteme.

Direkte Buskopplung als Alternative

Eine Alternative zu den genannten Verbindungen ist eine direkte Kopplung der Bussysteme verschiedener Computer (Bild 3). Im Idealfall ist die Datenrate auf der Busverbindung gleich der Datenrate der einzelnen Bussysteme. Die Funktionalität einer Bus-Bus-Kopplung ist höher, da auf ihr auch Adressen und Steuersignale ausgetauscht werden. Man spricht hier von einer engen Kopplung, da die beteiligten Computer gegenseitig auf ihre Adressräume – oder zumindest Teile davon – zugreifen können. Der Einsatz von Buskopplern ist nicht auf die Verbindung zweier Rechnersysteme beschränkt. Vielmehr können – unter Beachtung der Distanzgrenzen infolge der Art der physikalischen Übertragung – Topologien nach Bild 2 realisiert werden. Auf diese Weise können ausserordentlich leistungsfähige Multiprozessorsysteme aufgebaut werden. Man unterscheidet locker und eng gekoppelte Multiprozessorsysteme [8]. Bei der lockeren Kopplung haben die einzelnen Prozessoren private Speicher, die Kommunikation erfolgt durch den Austausch von Nachrichten (Message Passing). Eng gekoppelte Mul-

tiprozessorsysteme kommunizieren über gemeinsame Speicherbereiche (Shared Memory).

Der zweite Teil dieses Beitrags folgt in Heft 9/93.

Literatur

[1] H. Wedekind: Grundbegriffe verteilter Systeme aus der Sicht der Anwendung. Informationstechnik 30(1988)4, S. 263-271.
 [2] B.A. Huberman (ed.): The Ecology of Computation. Studies in Computer Science and Artificial Intelligence, vol. 2. Amsterdam: North-Holland, 1988.
 [3] U. Raabe u.a.: Verbindungsstrukturen für Multiprozessoren. Informatik-Spektrum 11(1988)4, S. 195-206.
 [4] B. Plattner u.a.: Datenkommunikation und elektronische Post. Bonn: Addison-Wesley, 2. Aufl., 1990.
 [5] E. Stein: Offene modulare Rechnersysteme. Technische Rundschau 1991, H. 49, S. 36-41.
 [6] M.A. Miller: Internetworking – A Guide to Communications LAN to LAN, LAN to WAN. Redwood City, CA: M&T Publishing Inc., 1991.
 [7] F.-J. Kauffels: Personalcomputer und lokale Netzwerke. Haar b. München, Markt & Technik Verlag, 6. Aufl., 1992.
 [8] F. Hertweck: Vektor- und Parallel-Rechner: Vergangenheit, Gegenwart, Zukunft. Informationstechnik 31(1989)1, S. 5-22.

Sept couches OSI sont souvent trop
Communication entre ordinateurs par couplage de bus, partie 1

La communication entre ordinateurs est souvent réalisée par des réseaux locaux (LAN). On veut ici considérer une réalisation alternative de la communication entre ordinateurs par couplage direct de systèmes de bus. L'importance de ce couplage augmente du fait de la banalisation croissante de systèmes de bus variés, alliée à des exigences grandissantes à la largeur de bande des voies de communication. Le couplage direct de bus est souvent un moyen élégant pour réaliser facilement des systèmes qui doi-

vent répondre à des exigences spéciales. Dans cette première partie, on expose succinctement à quoi la communication entre ordinateurs peut être utilisée, et les possibilités fondamentales de sa réalisation qui existent. Après une caractérisation des différents systèmes de bus on discute des situations où l'on désire relier directement des systèmes de bus entre eux. Dans une deuxième partie on présentera systématiquement la structure et les propriétés des coupleurs de bus et le logiciel nécessaire à leur fonctionnement.