1. Rechnerarchitekturen

1.3 PCI Express

PCI Express (PCIe) ist der Nachfolger von ISA und PCI. Daher wurde dieser Interconnect anfangs von Intel® auch als 3rd Generation I/O (3GIO) bezeichnet. Ähnlich wie beispielsweise beim Übergang von Parallel ATA zu Serial ATA wird die höhere Datenrate durch aufeinanderfolgende Übertragung der serialisierten PCI-Informationen erreicht. Der Grund für diesen eigentlich paradoxen Zusammenhang liegt in der Notwendigkeit, dass alle parallel versandten Daten innerhalb eines kurzen Zeitfensters beim Empfängerbuffer eingegangen sein müssen. Aufgrund unterschiedlicher Dämpfung und Kabellängen steht dies einer weiteren, starken Taktfrequenzsteigerung in den hohen Megahertzbereich im Wege. Die serielle Übertragung im Gigahertzbereich wird heute jedoch sehr gut beherrscht.

Physikalisch baut jeder Link eine individuelle Punkt-zu-Punkt-Verbindung auf (Lane). PCI Express erreicht pro Verbindung eine Datentransferrate von 2,5-GBit/Sek. Da ein selbsttaktender 8-Bit/10-Bit-Code verwendet wird, ist effektiv ein Übertragungsrate von 250 Mbyte/Sek. möglich. Alle Verbindungen sind vollduplexfähig.

Wie bei vergleichbaren seriellen Verbindungen üblich (z. B. InfiniBand) lassen sich einzelne Lanes bündeln. In den Spezifikationen vorgesehen sind maximal PCI Express 32x. In der Praxis üblich sind PCIe 16x als Alternative zum AGP-Slot, PCIe 8x und 4x finden sich im Serverbereich. Die Slots sind abwärtskompatibel, dementsprechend lässt sich eine 1x Karte auch in einem 8x Slot verwenden.

Alle von PCI bekannten Protokolle werden unverändert eingesetzt. In der Norm sind neben Kupferleitungen auch optische Verbindungen spezifiziert. Prinzipiell erlaubt PCI-Express einen echten HotPlug-Betrieb. Ein- und Ausstecken von Interfacekarten während des Betriebs ist im Bereich der X86 Server allerdings noch selten anzutreffen.

1.3.1 HyperTransport und HTX

HyperTransport (HT) ist ein universelles, bidirektionales Breitband Bussystem, das die aktuellen proprietären Busse ablösen soll. Der HyperTransport Standard ist offen gelegt und wird vom HyperTransport Konsortium herstellerübergreifend weiterentwickelt.

HypertTransport ist softwarekompatibel zu PCI, so dass zur Anbindung von PCI I/O-Karten einfache und effiziente Chipsätze genügen. Auch bei HyperTransport kommen serielle Punkt-zu-Punkt-Verbindungen zum Einsatz. Das elektrische Interface basiert auf LVDS (Low Voltage Differential SCSI) und arbeitet mit 1,2 Volt. Die Taktfrequenz liegt zwischen 200 und 800 MHz. Beim Einsatz einer DDR-Datenübertragung (Double Data Rate, Übertragung einer Information bei der ansteigenden sowie der fallenden Flanke) sind so bis zu 1600-MBit/Sek. pro Link möglich.

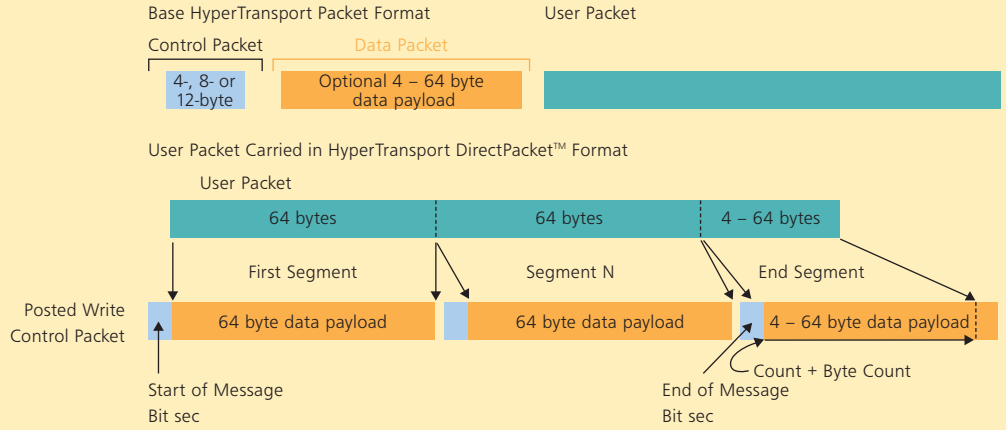
In der Norm ist eine Aggregierung von bis zu 32 Links vorgesehen. In der Praxis kommen aktuell maximal 16 Links vor. Mit diesen werden beim AMD Opteron™ bis zu 6,4 Gbyte/Sek. übertragen.

Zur Vermeidung von Steuer- und Kommandoleitungen wird ein paketbasiertes Datenprotokoll verwendet. Jedes Paket besteht unabhängig von der verwendeten Busbreite grundsätzlich aus einem Satz 32-Bit-Worten. Das erste Wort in einem Paket ist immer ein Kommandowort. Wenn ein Paket eine Adresse enthält, sind die letzten 8-Bit des Kommandowortes mit dem folgenden 32-Bit-Wort verkettet, so dass eine 40-Bit-Adresse entsteht. Die weiteren 32-Bit-Worte in einem Paket sind Nutzlast. Übertragungen werden immer zu einem Vielfachen von 32-Bit aufgefüllt.

HyperTransport wird aktuell von AMD, NVIDIA oder auch Apple eingesetzt. Neben der Kopplung von Prozessoren über einen schnellen Backbone Bus ist auch der Einsatz in Routern oder Switches möglich.

Im HTX-Connector wird der HyperTransport-Bus mit 16 Lanes herausgeführt und beispielsweise für schnelle Interconnects wie InfiniPath zur Verfügung gestellt.

HyperTransport User Packet Handling



1.1.9 Dual/Multi-Core-Prozessoren

Dual-Core-Prozessoren bilden den ersten Schritt beim Übergang zur Multi-Core-Computertechnik. Eine Multi-Core-Architektur hat eine einzige Prozessoreinheit mit zwei oder mehreren „Rechen-Kernen" und bietet in Verbindung mit geeigneter Software eine vollständige Parallelverarbeitung mehrerer Softwareprozesse (Threads). Das Betriebssystem betrachtet jeden dieser Rechen-Kerne als diskreten Prozessor mit allen zugehörigen Ausführungsressourcen.

Diese Mehrkernfähigkeit kann für den Nutzer in Multitasking-Umgebungen zu Verbesserungen führen, besonders wenn mehrere Vordergrundanwendungen mit mehreren Hintergrundanwendungen (z. B. Virenschutz, Datensicherheit, drahtloses Netzwerk, Verwaltung, Datenkomprimierung, Verschlüsselung und Synchronisierung) gleichzeitig aktiv sind. Der Vorteil für den Nutzer liegt auf der Hand: durch Vervielfachung der Prozessorkerne steigern die Prosessorhersteller die Fähigkeiten und Rechenressourcen des PCs erheblich, wodurch sich eine bessere Reaktionsfähigkeit, ein höherer Durchsatz gleichzeitig ablaufender Prozesse und die Vorteile der Parallelverarbeitung in Standardanwendungen ergeben.

Intel arbeitet seit mehr als einem Jahrzehnt an der Parallelverarbeitung, zunächst mit Multiprozessor-Plattformen und später mit der „Hyper-Threading Technology†" (HT-Technik), die 2002 von Intel eingeführt wurde und mit der Prozessoren Aufgaben parallel verarbeiten können, indem mehrere Threads in einem Einkern-Prozessor zusammengeführt werden. Die HT-Technik beschränkt sich auf einen Rechnerkern, dessen vorhandene Ausführungsressourcen sie effizienter nutzt, die Mehrkernfähigkeit bietet dagegen vollständige Ausführungsressourcen in zwei- oder mehrfacher Ausführung, um den gesamten Rechendurchsatz zu steigern. Bestimmte Prozessoren von Intel kombinieren die Vorteile der Dual-Core-Technik mit denjenigen der HT-Technik und unterstützen damit die gleichzeitige Ausführung von vier Threads.

2. Betriebssysteme

Weitere Informationen finden Sie auf unserer Homepage unter
www.transtec.de
www.transtec.at
www.transtec.ch

3. Cluster

3.3 Grid: Entstehung und Zukunft

Die Rechenleistung eines Computers ist eine Ressource. Der Umgang mit Ressourcen muss aus wirtschaftlichen Gesichtspunkten wohl bedacht sein. So können nutzbar gemachte Ressourcen finanziellen Erfolg bringen, während ungenutzte Ressourcen totes Kapital bedeuten. Das verteiltes Rechen über Rechnergrenzen hinweg, wie es heutzutage in Clustern betreiben wird, ist nur der Anfang. Wenn verteilte Rechnen- und Speicherkapazitäten aus „normalen" Bürorechnern oder aus ganzen Clustern zusammengeschlossen an einem Ort und zu einer Zeit nutzbar gemacht werden, können selbst große Rechenprobleme in kürzester Zeit gelöst werden. Heutzutage ist das nächste Ziel die Vernetzung mehrerer Forschungseinrichtungen untereinander um die gemeinsame Rechenkapazität zu bündeln. Dieser Zusammenschluss wird Grid genannt. Der Name Grid ist vom dem Begriff des Power Grid, zu deutsch Stromnetz, abgeleitet. Wie beim Stromnetz der Strom, soll beim Grid die entsprechende Rechnerressource aus der Steckdose erhältlich sein, so der Grundgedanke. Es ist geplant viele Clustersysteme weltumspannend zusammenzuführen.

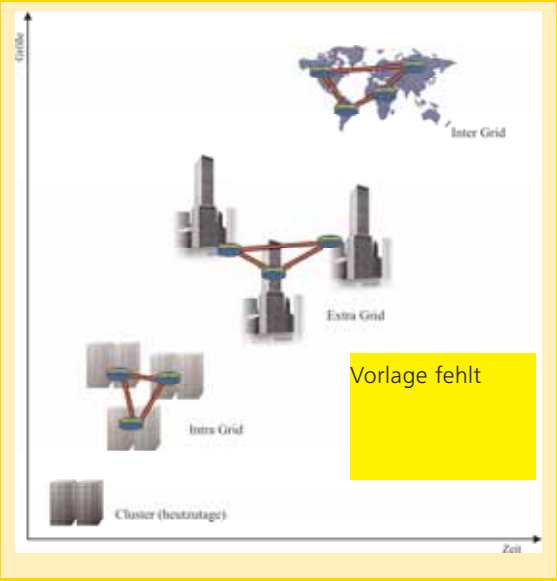
Als endgültiges Ziel ist die Vernetzung sämtlicher Leistungen eines Rechners oder eines Clusters wie die oben genannte Rechenkapazität, die Speicherkapazität, die Informationen und die Applikationen geplant. Zur Erreichung dieses Ziels müssen zuerst die erforderlichen Informationen gesammelt, ausgearbeitet und in Standards niedergeschrieben werden.

3.3.1 Die unterschiedlichen Typen von Gridsystemen

Gridsysteme können auf zwei unterschiedliche Weisen kategorisiert werden. Zum einen ist eine Unterteilung auf Applikationsebene, zum anderen anhand ihrer Größe möglich. In der Applikationskategorie finden sich drei große Untergruppen: Die Compute Grids, die Scavenging Grids und die Data Grids. Bei der ersten, und wohl auch derzeit wichtigsten Untergruppe, handelt es sich im allgemeinen Sinne um zusammengeschlossene Clustersysteme mit dem Zweck der Bündelung der vorhandenen Rechenleistung. Dieser Zusammenschluss erfolgt oftmals schon über die Homogenität der Rechnerarchitekturen hinweg. In der nächsten Kategorie der Scavenging Grids werden die Ressourcengrenzen um weitere, nicht im speziellen für rechenintensive Aufgaben vorgesehene Rechner, wie zum Beispiel normale Bürorechner, erweitert. Der Begriff des Scavenging bedeutet hier „Plünderung", da die ungenutzten Ressourcen von Außen nutzbar gemacht werden. Die dritte Untergruppe der Data Grids vereint die vorhandene Speicherkapazität zu einem gewaltigen Datenaufnahmesystem. Ein solches System kommt im CERN, einem sehr großen Teilchenbeschleunigern zum Einsatz. Bei einer solchen wissenschaftlichen Anwendung fallen in sehr kurzer Zeit sehr viele Daten an.

Eine weitere Kategorie richtet sich nur nach der Größe der Systeme in der evolutionären Entwicklung der Gridsysteme (vgl. Abbildung 1). Hierbei lassen sich vier Stufen voneinander abgrenzen: Die erste Stufe bilden die heutigen Clustersysteme mit meist sehr homogenen Rechnerarchitekturen und geringer räumlicher Ausdehnung. Die nächste Stufe beschreiten die Intra Grids. Diese Stufe fasst mehrere Clustersysteme in Abteilungen oder Unternehmen zusammen. Nomen et omen, so trägt diese Stufe stellvertretend für bereits benötigte Accounting- und Abrechnungsfunktionen bereits den Begriff Grid im Namen.

Abbildung 1: Kategorisierung der Gridsysteme nach Größe



Fasst man nun mehrere Intra Grids aus verschiedenen Unternehmen zusammen, so erhält man die nächste Entwicklunsgsstufe: Die sogenannten Extra Grids. Auf dieser Stufe sind die Rechnerarchitekturen gewiss nicht mehr homogen, weswegen Architekturunabhängigkeit dringend notwendig ist. Inhomogenität ist aber nicht zwangsläufig als Problem anzusehen, sondern bietet für die vernetzten Forschungseinrichtungen eine große Vielzahl an Möglichkeiten. Die vorhandenen

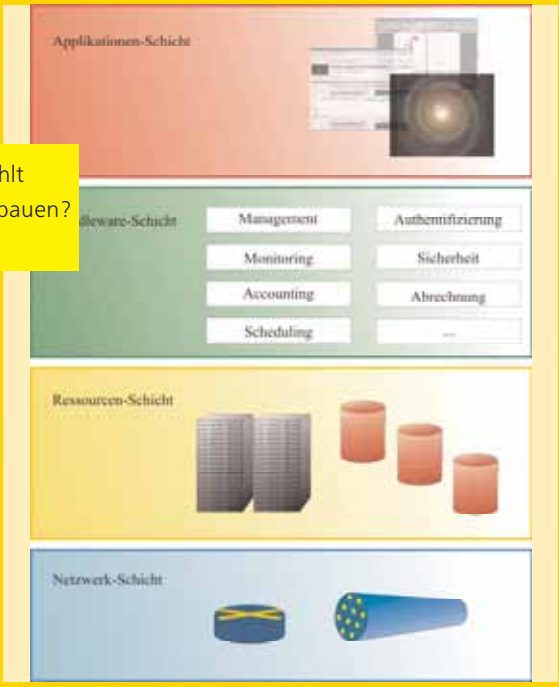
Ressourcen lassen sich so sehr effizient nutzen. Die Inter Grids bilden die letzte Stufe. Der Begriff ist mit Recht an den Begriff des Internets angelehnt. Diese Grids sind umfassend. Jeder Benutzer kann Zugriff auf die vorhandenen Kapazitäten, seien es Rechen- oder Speicherkapazitäten, erhalten. Die Authentifizierung- und Zugriffssteuerung mit Abrechnung und Monitoring stellt eine große Herausforderung an die Managementschicht dar.

Es gibt bereits ehrgeizige Projekte, welche die Idee des globalen Gridsystems verwirklichen wollen. Diese Projekte sind unter anderem das Eurogrid oder das TeraGrid. Die Problematik ist hier nicht nur auf Hard- und Softwareseite verankert, sondern auch an eventuell vorhandenen Interessenkonflikten der verschiedenen beteiligten Parteien.

Architektur eines Grid

Um ein funktionierendes Gridsystem zu entwickeln, bedarf es mehrerer Zwischenstufen. Die reine Vernetzung und Aggregation der Rechenleistung entspräche lediglich einem Clustersystem und nicht einer Gridarchitektur. Daher ist der Aufbau eines Gridsystems in Schichten unterteilt und an das OSI-Modell angelehnt. In der nachfolgenden Abbildung 2 sind der Aufbau und die einzelnen Schichten eines Gridsystems schematisch dargestellt.

Abbildung 2: Grid Architektur Schichtenmodell



Vorlage fehlt oder nachbauen?

Die einzelnen Funktionen der Schichten sind wie folgt: In der obersten Schicht befinden sich die Applikationen und Dienste für den Anwender. So zum Beispiel Entwicklungsumgebungen und Zugangsportale. Die Entwicklungsumgebungen differieren zum Teil sehr stark, abhängig vom jeweiligen Einsatzgebiet. Unter den Diensten versteht man die Management Funktionen, wie zum Beispiel Accounting, Abrechnung- und Monitoringfunktionen. Nur durch diese Funktionen lassen sich die Ressourcen sicher verteilen.

In der Middleware sitzt die eigentliche Intelligenz des Grid. In dieser Schicht werden die Protokolle und Kontrollinstrumente des Grid bereitgestellt.

In der nächst tiefer gelegenen Schicht liegen die Ressourcen, wie Hardware, Speicher und die Rechenkapazitäten. Es kann sich hierbei auch um einen Cluster handeln, bei welchem der Masterknoten die Anfragen und Aufgaben, welche aus dem Grid stammen, annimmt und an seine Rechenknoten weiterverteilt. Diese untere Verteilung ist natürlich abhängig vom Batch Queuing System welches hier Verwendung gefunden hat.

Die unterste Schicht bildet, wie in jedem vernetzten System, das Netzwerk selbst. Diese Schicht, mit ihren Protokollen und der entsprechenden Hardware, ist für die Übermittlung der Pakete zuständig, welche ihr übergeben werden. Auch hier gilt dass die Ausstattung abhängig von der jeweiligen Anwendung ist. Sind sehr niedrige Latenzzeiten erforderlich ist eine andere Netzwerktechnik von Nöten, wie bei der Übermittlung großer Datenpakete. Gerade bei Gridsystemen spielt hier das Internet natürlich eine sehr große Rolle. Aber in diesem Fall ist die Netzwerktechnik und Architektur leider nicht leicht veränderbar. Es wird jedoch von Experten die Meinung geteilt dass Applikationen, welche große Datenmengen untereinander austauschen, sehr selten sein werden. Derweil ist das Problem weniger eine Frage der Netzwerktechnologie als eine Frage von nationalen Interessen und des Budgets. Die entsprechende Netzwerktechnik und auch das Wachstum sind vorhanden.

3.3.3 Anforderungen an eine Grid Information Service Infrastruktur (GIS)

Eine Grid Information Service Infrastruktur kann ein Mitglied oder eine Ressource in einem Gridzusammenschluss sein. Um als Mitglied die entsprechenden Anfragen beantworten zu können muss jedes Gridmitglied bestimmte Fähigkeiten besitzen. Das Global Grid Forum ist eine mehrere tausend Mitglieder umfassende weltweite Organisation. Von dieser Organisation werden die Eckpunkte welche eine Grid Information Service Infrastruktur bieten muss, wie folgt definiert:

- Effizientes Melden von Statusinformationen einer einzelnen Ressource
- Fehlertoleranz
- Verteilte Komponenten für dezentralisierten Zugriff
- Dienste für Zeitstempel und Time-To-Live (TTL) Attribute
- Abfrage und Erhebungsmechanismen
- Starke, sichere Authentifikation

An dieser Stelle folgt nun ein kurzer Vorgriff auf das nächste Kapitel: Die oben aufgeführten Eckpunkte werden am Bespiel der Grid-Middleware Globus Toolkit von der Globus Alliance mit Hilfe der beiden Protokolle Grid Resource Inquiry Protocol (GRIP) und Grid Resource Registration Protocol (GRRP) erlangt. Diese beiden Protokolle sorgen für die Verständigung zwischen dem Grid Index Information Service und dem Grid Resource Information Service.

3.4. Die Grid Middleware

In den nachfolgenden Unterkapiteln wird die Middleware beschrieben, wie sie Ian Foster 2001 in seinem Artikel „The Anatomy of the Grid: Enabling Scalable Virtual Organizations" beschreibt. In der Middleware liegt die eigentliche Intelligenz der Gridsysteme. Ian Foster nennt die Middleware im eigentlichen Sinne „Grid Architektur", da das gesamte Gridsystem und sein Aufbau die Middleware für die Anwendung als Grid zur Verfügung stellt bzw. verknüpft. Diese Gridarchitektur identifiziert Systemkomponenten, spezifiziert sie nach Funktion und Einsatzmöglichkeiten und stellt fest, wie diese mit Komponenten interagieren. Ian Foster als einer der Urväter des Gridcomputings und aktives Mitglied der Globus Alliance erläutert seine Definition der Grid Architektur anhand des Globus Toolkit (GT), welches als Open Source Projekt von der Globus Alliance entwickelt wird.

3.4.1 Interoperabilität

Eine der wichtigsten Forderungen an die Grid Architektur ist die Interoperabilität. Beziehungen zwischen unterschiedlichen Parteien können nur dann dynamisch initiiert werden, wenn Interoperabilität vorliegt. Wäre dies nicht der Fall, könnten keine Ressourcen ver- und geteilt werden, da die Ressourcen nicht kompatibel wären. Aus eben diesem Grund könnten keine virtuellen Organisationen (VO) geschaffen werden. VO sind Zusammenschlüsse (z. T. über Organisationsgrenzen hinweg) von verschiedenen Parteien, welche untereinander Ressourcen teilen.

3.4.2 Protokolle

Zum Erreichen der Interoperabilität werden Protokolle eingesetzt. Protokolle haben den Vorteil, dass vorgegeben ist, wie ein System interagieren und wie Informationen ausgetauscht werden sollen um das gewünschte Verhalten zu erzielen. Das Augenmerk liegt also auf den äußeren Gegebenheiten und nicht auf den inneren.

VOs werden vorhandene Organisationen erweitern und nicht ersetzen. Dadurch ist lediglich von Bedeutung, wie die schon vorhandenen Ressourcen miteinander kommunizieren.

3.4.3 Services (Dienste)

Warum werden Services eingesetzt? Ein Service wird nur durch das Protokoll definiert, dessen Sprache er spricht und demnach er sich verhält. Die Definition von Standard-Services – für den Zugang zu Rechenkapazitäten, zu Daten, Paralleles Scheduling, usw. – erlauben es auf einfachste Weise Details für die Entwicklung von Programmen für VOs zu klären.

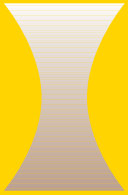
3.4.4 APIs und SDKs

Der Einsatz von Applikation Programming Interface (API) und Software Development Kits (SDKs) erlaubt eine schnelle Entwicklung und einfache Verbreitung von Programmen für das Grid. Benutzer müssen die Möglichkeit haben diese Programmen bedienen zu können. Auch

die Robustheit und Korrektheit eines mit Hilfe eines API entwickelten Programms nimmt zu. Im Gegenzug nehmen die Entwicklungs- und Unterhaltskosten für solche Programme ab.

Ian Foster fasst die oben genannten Bedingung wie folgt zusammen: zuerst müssen Protokolle und Services definiert und anschließend APIs und SDKs entwickelt werden.

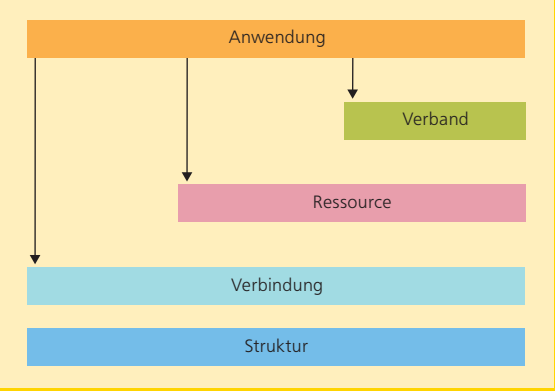
3.4.5 Die Protokoll Architektur



Die Ressourcen- und Verbindungsprotokolle formen die Taille der „Grid-Sanduhr". Für den Aufbau eines Gridsystems eignen sich vorzüglich die Protokolle des Internets. Die sind standardisiert und erprobt. Natürlich unterliegen sie bestimmten Regeln, welche es gilt einzuhalten.

Der größte Vorteil dieser vorhandenen Grundlage ist dass diese Protokolle einen breiten Stapel an Ressourcentypen unterstützen, welche über die Jahre dazu gekommen sind. Die Weiterentwicklung in der untersten Schicht schreitet stets voran und somit kann auf dieses Kollektiv unterschiedlicher Hardware einheitlich zugegriffen werden. In der nachfolgenden Abbildung 3 ist die Grid Protokoll Architektur dargestellt, wie Ian Foster sie spezifiziert.

Abbildung 3: Die Grid Protokoll Architektur



Die Strukturschicht

Die Strukturschicht bietet die Ressourcen an, welche vom Grid benutzt werden können. Diese sind die Rechenkapazitäten, Speichersysteme, Kataloge, Netzwerkressourcen und Sensoren. Diese Ressourcen können einerseits eine logische Einheit, andererseits aber auch ein verteiltes Dateisystem oder ein Cluster sein. Aus diesem Grund kann ein solcher Aufbau durchaus andere, fremde Dienste wie den NFS Dienst beinhalten. Diese gehören indes nicht zur Gridarchitektur.

Die Komponenten der Strukturschicht erhalten die spezifischen Operationen für die spezifische physikalische oder logische Komponente. Bei der Implementierung ist es wichtig darauf zu achten, dass wenige Operationen in die unteren Schichten gelegt werden. Dadurch ist es leichter möglich mehr übergreifende Dienste anzubieten. Zum Beispiel kann ein ökonomisches Reservieren von Ressourcen nur dann stattfinden, wenn der Dienst in einer höheren Schicht angesiedelt ist. Trotzdem gibt es natürlich Ressourcen welche das Reservieren von Ressourcen bereits auf hohem Niveau anbieten, wie zum Beispiel Clustersysteme.

Werden diese Management-Dienste – wie das Job-Queuing-System eines Clusters – in der Ressource angeboten, müssen sie folgenden Anforderungen genügen um für das Grid effizient nutzbar zu sein. Diese Anforderungen sind in nachfolgender Tabelle aufgelistet:

Tabelle 1: Anforderung an Struktur-Komponenten in der Strukturschicht

Typ der Ressource	Anforderungen an die Ressource
Rechenkapazitäten	Es werden Mechanismen benötigt um Programme zu starten und deren Ergebnisse zu überwachen und zu kontrollieren. Eine fortgeschrittene Möglichkeit zur Reservierung von Ressourcen ist nützlich. Untersuchungsfunktionen um vorhandene Soft- und Hardware zu spezifizieren und nutzbar zu machen müssen ebenfalls vorhanden sein.
Speicher-Ressourcen	Mechanismen um Dateien abzulegen und abzufragen müssen vorhanden sein. Möglichkeiten der Performanzsteigerung durch Striping oder mehr Datensicherheit durch Mirroring der Datenbestände ist sinnvoll. Auch hier sind Untersuchungsfunktionen, um die vorhandene Soft- und Hardware zu spezifizieren und nutzbar zu machen, nötig.
Netzwerk-Ressourcen	Management-Funktionen für die Überwachung und die Kontrolle des Netzwerkverkehrs ist sinnvoll (= Priorisierung/Reservierung). Zusätzlich sollten die oben beschriebenen Untersuchungsfunktionen vorhanden sein.
Code-Repository	Code-Repository (zu deutsch, Code-Behältnis) ist eine Art CVS um verschiedene Versionen von Software zu spezifizieren und zu kontrollieren.
Kataloge	Kataloge sind eine besondere Art von Speichersystemen, welche zusätzliche Aktualisierungs- und Suchfunktionen enthalten, zum Beispiel relationale Datenbanken.

Speziell für das Globus Toolkit fasst Ian Foster zusammen, dass das Globus Toolkit entwickelt wurde um bereits bestehende Struktur-Komponenten zu benutzen und auf ihnen aufzusetzen. Sollte die vorhandene Funktionalität nicht ausreichend sein, wird sie vom Globus Toolkit erweitert. So bietet das Globus Toolkit Funktionen wie Untersuchungsfunktionen für die zugrunde liegende Soft- und Hardware, für Speichersysteme und für Netzwerkressourcen.

Die Verbindungsschicht

Die Verbindungsschicht definiert die Protokolle, welche vom Grid benötigt werden um die Kommunikation und Authentifizierung innerhalb der Grids zu ermöglichen. Die Authentifizierung bietet Funktionen basierend auf Kommunikationsdiensten für eine kryptografisch sichere Identifizierung von Personen und Ressourcen. Die Kommunikationsdienste beinhalten Transport, Routing und Naming. Obwohl es hierfür viele Anbieter und Alternativen gibt, werden die Protokolle des TCP/IP Stapels verwendet. Diese sind wie folgt:

- für das Internet: IP und ICMP
- für den Transport: TCP und UDP
- und für Anwendungen: DNS, OSPF, RSVP, etc.

Es ist jedoch nicht gesagt, dass zukünftige Gridanwendungen nicht nach anderen Protokollen verlangen. Ein wichtiger Gesichtspunkt ist die Tatsache, dass die Grundlagen für die Sicherheit im Grid auf Standards basieren, welche bereits erprobt sind, sprich schon häufig verwendet werden. Nur so können Sicherheitslücken ausgeschlossen werden. Diese Sicherheitsaspekte müssen vier Charakteristiken erfüllen.

Tabelle 2: Sicherheitscharakteristiken in der Verbindungsschicht

Charakteristik	Anforderungen/Beschreibung
Einfaches Anmelden	Benutzer müssen befähigt sein sich einmalig einzuloggen und danach Zugriff auf alle ihre entsprechenden Ressourcen zu haben (= Single Sign On).
Bevollmächtigung	Dem Benutzer muss es möglich sein, seine Rechte an ein Programm weiterzugeben, welches auf seine entsprechenden Ressourcen zugreifen kann.
Integration von verschiedenen lokalen Sicherheitslösungen	Jeder Ressourcenprovider hat die Möglichkeit eigene Sicherheitslösungen zu verwenden. Die Gridsoftware muss mit ihnen interagieren können.
Benutzerbezogenes Sicherheitssystem	Das Sicherheitssystem muss Benutzerbezogen sein. Dies bedeutet, dass sich die jeweiligen Ressourcenprovider nicht kennen müssen und keine Konfigurations- und Administrationsinformationen austauschen müssen.

Des Weiteren sollte die Grid Sicherheitslösung ein flexibles Auswählen der Sicherheitsstufe ermöglichen und mit verbindungslosen sowie verbindungsorientierten Protokollen arbeiten können. Für die im Globus Toolkit verwendeten GSI (Globus Security Infrastructure) Protokolle lassen sich folgende verwendete Technologien festhalten: Für die meisten in der oben genannten Tabelle gelisteten Anforderungen wird TLS (Transport Layer Security) verwendet. Genauer gesagt für einfaches Anmelden, Bevollmächtigung, Integration von verschiedenen lokalen Sicherheitslösungen (inklusive Kerberos) und für das benutzerbezogene Sicherheitssystem. Als Zertifikate werden Zertifikate nach dem X.509 Standard verwendet. Lokale Sicherheitslösungen werden mit Hilfe der GAA (Generic Authorization and Access) Kontrollschnittstelle unterstützt.

Die Ressourcenschicht – Zugriff auf eine Ressource

Die Ressourcenschicht baut auf der Verbindungsschicht auf und benutzt deren Kommunikations- und Authentisierungsprotokolle. Kurz gesagt: Die Ressourcenschicht sorgt für die Verteilung einer einzigen Ressource. Von ihr werden folgende Funktionen abgedeckt: Sicherere Übertragung, Verbindungsinitiierung, Überwachung, Kontrollfunktionen, Abrechnung und Verrechnung der in Anspruch genommenen Ressourcen. Diese Ressourcenschicht-Protokolle kümmern sich ausschließlich um die Anforderungen welche an diese eine Ressource gestellt werden und ignorieren somit globale Aspekte der verteilten Architektur.

Es gibt primär zwei Klassen von Protokollen:

Tabelle 3: Protokollklassen der Ressourcenschicht

Protokollklasse	Anforderungen/Beschreibung
Informations-protokolle	Diese Protokolle werden benutzt um Informationen über Struktur und Zustand der Ressource zu sammeln und wiederzugeben.
Management-protokolle	Diese Protokolle werden benutzt um die Anforderungen der Ressource zu regeln, um den Zugang zu steuern und um Operationen auszuführen bzw. ausführen zu lassen. Natürlich müssen diese Operationen mit den Regeln, welche der Ressource zu Grunde liegen, konform sein.

Für das Globus Toolkit werden folgende Protokolle für die obigen Anforderungen verwendet:

- GRIP basiert derzeit auf dem LDAP (Lightweight Directory Access Protocol). Es wird verwendet um ein Standard-Informationsprotokoll zu definieren. Das damit verknüpfte Ressourcen-Registrierungsprotokoll GRRP (Grid Resource Registration Protocol) wird benutzt um Ressourcen an den Grid Index Information Servern zu registrieren.
- GRAM wird benutzt um Rechenkapazitäten in Anspruch zu nehmen und diese zu überwachen.
- GridFTP wird für Dateitransfers und für das Management des Datenzugriff benutzt. Dieser Service beinhaltet die Sicherheitsprotokolle der Verbindungsschicht um so u.a. teilweisen Dateizugriff zur Verfügung zu stellen.
- Zusätzlich wird LDAP für den Zugriff auf die Kataloge verwendet.

Die Verbandsschicht – Zugriff auf mehrere Ressourcen

Im Gegensatz zur Ressourcenschicht kümmert sich die Verbandsschicht um die Verteilung mehrerer Ressourcen. Durch die Abspaltung der Verbandschicht von der Ressourcenschicht kann eine Vielzahl an zusätzlichem Verhalten adoptiert werden, ohne es den einzelnen Ressourcen aufzubürden. Ian Foster führt folgende Beispiele auf:

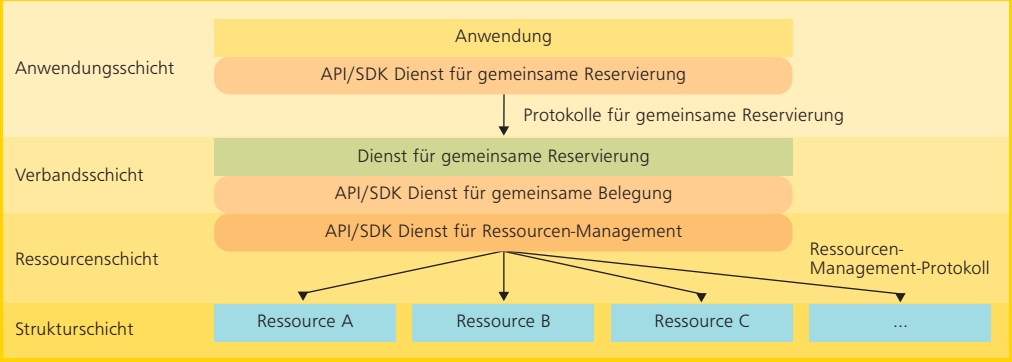
- Verzeichnis Dienste: Benutzer der VO können VO Ressourcen aufsuchen bzw. deren derzeitigen Status abrufen. Die Suche kann anhand eines Namens, des Typs, der Verfügbarkeit oder nach der aktuellen Belastung geführt werden.
- Gemeinsame Belegung von Ressourcen erlauben es dem Benutzer mehrere Ressourcen für einen bestimmten, gemeinsamen Zweck zu reservieren.
- Überwachung und Diagnosefunktionen helfen dem Benutzer ein möglicherweise fehlerhaftes Verhalten einer Ressource oder ein Sicherheitsloch festzustellen.
- Datenreproduzierung dient der Performanzsteigerung beim Zugriff auf gemeinsame Datenbestände.
- Suchdienste für gewünschte Software erlauben es dem Benutzer die optimale Software für seinen Einsatzzweck ausfindig zu machen.
- Abrechnung und Berechnungsdienste dienen dazu die benutzten Ressourcenzeiten zu notieren und abzurechen.
- Gridfähige Programmsysteme erlauben es dem Benutzer „normale" nicht gridfähige Programme auf dem Grid auszuführen.

Die benötigte breite Masse an Verbandsschichtenprotokollen wird durch die verschiedenartigen, obigen Beispiele gerechtfertigt.

Grafisch lässt sich der derzeitig kennen gelernte Aufbau der Gridarchitektur in Abbildung 4 darstellen. Für diese Zeichnung wurde ein System mit einem Verband von Ressourcen gewählt. Natürlich hätte anstatt der Verbandsschicht auch ein Aufbau mit Hilfe der Ressourcenschicht gewählt werden können. Im mittleren Teil befinden sich innerhalb der Verbandschicht ein API und ein SDK für gemeinsame Belegung von Ressourcen, welche mit Hilfe eines Ressourcen-Management-Protokolls auf darunter liegende Ressourcen zugreifen. Aus diesem Grund befinden sich unterhalb wiederum ein API und ein SDK, diesmal für das Management des Ressourcenprotokolls.

Über diesem Teil wird ein Dienst für die gemeinsame Reservierung von Ressourcen implementiert. Dieser wird mit Hilfe eines speziellen Protokolls, welches von einem API und SDK Reservierungsdienst der sich direkt unterhalb der Applikation befindet, gesteuert und angesprochen.

Abbildung 4: Aufbau eines Dienstes für gemeinsame Belegung von Ressourcen

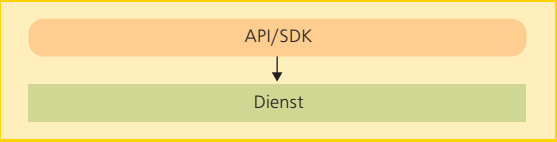


Die anonym aufgezeigten Ressourcen können alle in Tabelle 1 – Anforderung an Struktur-Komponenten in der Strukturschicht aufgeführten Ressourcentypen sein. Ian Foster zeigt anhand des Globus Toolkit folgende Umsetzung der Dienste und Protokoll in die Praxis:

- GIIS sorgt für beliebige Sichtweisen auf Ressourcen
- GRIS wird verwendet um den Ressourcen Status abzufragen
- GRRP wird benutzt um Ressourcen zu registrieren
- DUROC Library wird für die gemeinsame Belegung von Ressourcen verwendet

Als eine Art Zusammenfassung lässt sich festhalten, dass ein API und SDK immer auf einen darunter liegenden Dienst zugreifen. Durch die Trennung in mehrere Schichten können Komponenten einfach ausgetauscht werden. Ian Foster rät die SDKs von dritter Stelle zu verwenden.

Die Anwendungsschicht



Die Anwendungsschicht umfasst die Benutzeranwendungen welche innerhalb einer VO operieren. Diese Anwendungen greifen mit Hilfe von Protokollen entweder auf die Dienste der Verbandsschicht oder auf die Dienste der Ressourcenschicht zu, abhängig davon ob eine oder mehrere Ressourcen gleichzeitig belegt werden müssen. Unter Anwendungen versteht man sinnigerweise Frameworks und Libraries. Diese Frameworks definieren und verwenden andere mögliche Protokolle für die Kommunikation. Auch hier gilt wieder, dass APIs und SDKs auf darunter liegende Dienste zugreifen.

4. Speicherbusse

4.8 SAS und Serial-ATA-2: Die Verbindung von zwei Standards

4.8.1. Geschichte des SCSI- und ATA-Standards

Serial attached SCSI (SAS) stellt die Ablösung der bisherigen parallelen SCSI-Schnittstelle dar. Diese Weiterentwicklung der parallelen SCSI-Schnittstelle wurde notwendig, da mit dem U320-Standard die Grenze des technisch machbaren beinahe erreicht ist. Bei der Verabschiedung des ersten SCSI (Small Computer System Interface)-Standards 1986, waren Übertragungsgeschwindigkeiten, wie sie heute erreicht werden nicht denkbar.

Geschichte des SCSI-Standards

Interconnect	Standard	Year	Speed	Key features
SASI		1979		Shugart Associates
SCSI-1	SCSI-1	1986	~ 2 MB/Sek.	Asynchronous, narrow
SCSI-2	SCSI-2	1989	10 MB/Sek.	Synchronous, wide
SCSI-3	Split command sets, transport protocols and physical interfaces into separate standards			
Fast-Wide	SPI/SIP	1992	20 MB/Sek.	
Ultra	Fast-20 annex	1995	40 MB/Sek.	
Ultra 2	SPI-2	1997	80 MB/Sek.	LVD
Ultra 3	SPI-3	1999	160 MB/Sek.	DT, CRC
Ultra 320	SPI-4	2001	320 MB/Sek.	Paced, Packetized, QAS

Im Laufe der Zeit wurden aber immer höhere Ansprüche an die Datenübertragungsgeschwindigkeit gestellt und es stellte sich heraus, dass die im U640-Standard angedachte Verdoppelung der Busgeschwindigkeit des parallelen Datenbus technisch kaum realisierbar war. Die Geschwindigkeit des Bus muss nämlich so begrenzt werden, dass das langsamste und das schnellste Bit innerhalb eines „Bit-Takt-Zyklus" ankommen. Dieses Problem der unterschiedlichen Signallaufzeiten auf einem parallelen Bus führte zu einer Ablösung der parallelen Busarchitektur durch eine serielle. Die ersten seriellen SCSI-Architekturen wurden 1995 mit Fibrechannel (FCP) und 1996 mit SSA definiert. Während FCP sich, insbesondere nach Einführung des FCP-2 Standards sehr schnell verbreitete, stellt SSA eine proprietäre Architektur von IBM dar, die aus diesem Grund nur eine relativ geringe Marktdurchdringung erreichen konnte. Da auch die ATA-Schnittstelle auf Grund ihrer Architektur, die selben technischen Probleme hat, wurde auch hier mit dem Serial-ATA-Standard der Übergang von einer parallelen Architektur auf einen seriellen Bus definiert.

Geschichte des AT-Attachement(ATA) -Standards

Generation	Standard	Year	Speed	Key features
IDE		1986		Pre-standard
	ATA	1994		PIO modes 0-2, multiword DMA 0
EIDE	ATA-2	1996	16 MB/Sek.	PIO modes 3-4, multiword DMA modes 1-2, LBAs
	ATA-3	1997	16 BM/Sek.	SMART
	ATA/ATAPI-4	1998	33 MB/Sek.	Ultra DMA modes 0-2, CRC, overlap, queuing, 80-wire
Ultra DMA 66	ATA/ATAPI-5	2000	66 MB/Sek.	Ultra DMA mode 3-4,
Ultra DMA 100	ATA/ATAPI-6	2002	100 MB/Sek.	Ultra DMA mode 5, 48-Bit LBA
Ultra DMA 133	ATA/ATAPI-7	2003	133 MB/Sek.	Ultra DMA mode 6

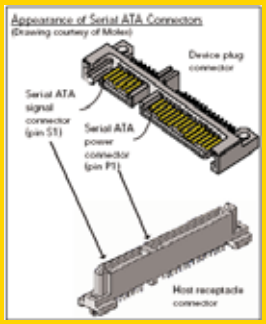
4.8.2. SAS- und Serial-ATA-Standard

Der SAS-Standard stellt eine konsequente Fortentwicklung des SCSI-Standards dar, wobei insbesondere auf eine Abwärtskompatibilität zu den bisherigen definierten SCSI-Protokollen Wert gelegt wurde. Gleichzeitig wurde aber auch auf ein Interoperabilität mit dem Serial-ATA-Standard geachtet. Dieser Punkt ist insbesondere in Hinblick auf „Tiered Storage" und Information Lifecycle Management von Bedeutung. Serial-ATA-Platten können hierbei aber nur in SAS-Devices benutzt werden und nicht umgekehrt.

Speicherbusse

Serial-ATA-Steckverbindern bestehen aus der Logik und der Stromversorgung, mit einer geräteseitigen Lücke dazwischen und einer „Plastiknase" rechnerseitig. SAS benutzt diese Lücke für den zweiten SAS-Link auf der Rückseite des Verbindungsstegs, dies erklärt, warum SAS-Geräte nicht an Serial-ATA-Anschlüssen benutzt werden können, Serial-ATA-Platten jedoch an SAS-Anschlüssen. Bei SAS-Steckverbindern ist im Gegensatz zu den bei Serial ATA definierten Anschlüssen eine Einrastvorrichtung vorgesehen, um sicheren Halt bei Vibrationen zu gewährleisten.

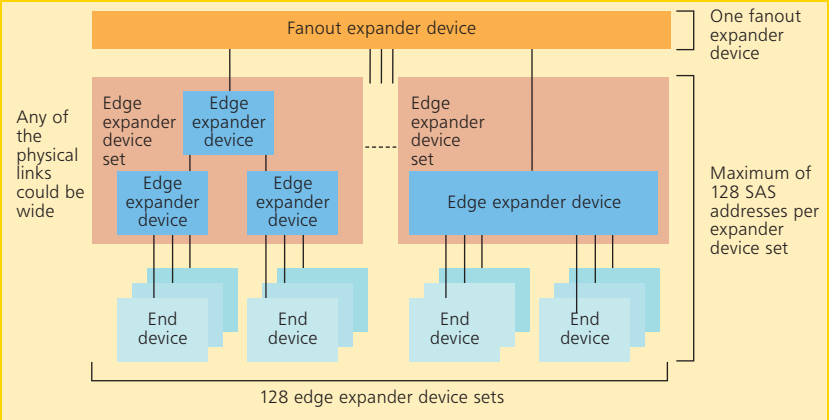
Serial-ATA-Connector



SAS Connector

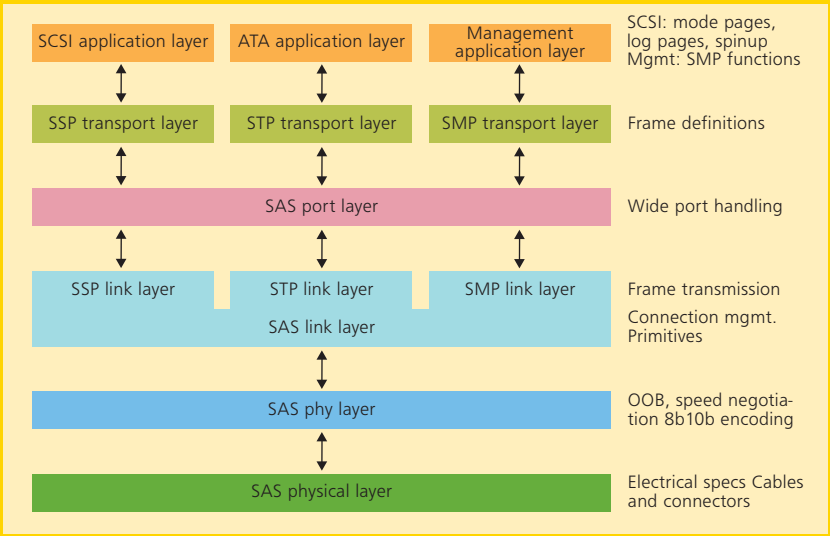
Die Integration des Serial-ATA-Protokolls wird auf der Transportprotokollebene, eines von 6 definierten Layern, des SAS-Standards, definiert. Das SAS-Protokoll unterstützt drei Transportprotokolle: Das Serial SCSI Protokoll (SSP) stellt die Verbindung zu SAS- beziehungsweise SCSI-Geräten dar. Das Serial-ATA-Tuneling-Protocol (STP) ist für die Verbindung zu Serial-ATA-Platten verantwortlich, während das Serial Management Protocol (SMP) die Konnektivität zu Fanout- und Edgeexpandern definiert. Die Funktion der Expander ist ähnlich wie die von Fibrechannelswitches zu sehen; mit ihr ist es möglich, größere SAS-Domains aufzubauen. Es können maximal zwei Edgeexpander in einer SAS-Domain eingesetzt werden. Fanoutexpander sind vergleichbar zu den Directorswitches im Fibrechannelbereich und bilden eine zentrale Vermittlungsstelle. Ein Expander kann maximal 128 SAS-Adressen verwalten, was eine maximale Anzahl von 16.384 Geräteanschlüssen ergibt. In den Expandern sind Routing-Tabellen definiert, wobei mehrere Verbindungen gleichzeitig zwischen zwei Endpunkten aktiv sein können. Ein Serial-ATA-Endgerät darf allerdings nur über eine aktive Verbindung angesprochen werden.

Beispiel einer Edgeexpander mit Fanoutextender SAS-Domain



Auf der physikalischen Ebene werden zum einen die Kabel spezifiziert, die extern den Spezifikationen von Infiniband entsprechen. Für interne Verwendung werden die von Serial ATA bekannten Steckverbindungen benutzt.

Die Layer des SAS-Protokollstandards

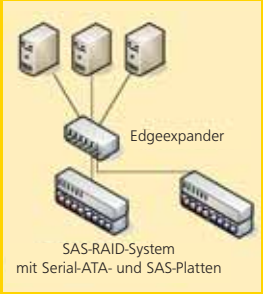


Durch die serielle Punkt-zu-Punkt-Verbindung entfallen auch die bisher bei SCSI notwendigen Terminatoren und – Full-Duplex-Übertragung wird ermöglicht. Ebenso liegt hier die Aushandlung der Geschwindigkeit der Geräte, die in der ersten Stufe auf 3-GBit/Sek. begrenzt ist, was bei einer Full-Duplex-Verbindung Übertragungsgeschwindigkeiten bis zu 600 MB/Sek. ermöglicht. Auch wird hier festgelegt, dass, analog zu Fibrechannel, Platten grundsätzlich dual-ported sind. So lässt sich ein Redundanz relativ einfach aufbauen. Für die Adressierung der Geräte-Anschlüsse werden die beim IEEE registrierten WWNs aus der FC-Technik genutzt. Die oben angesprochene Begrenzung in der Anzahl der Expander ergibt für ein SAS-System maximal 16.384 ansprechbare Adressen. Im Fibrechannel sind es mehr als 16 Millionen Adressen, die theoretisch zum Einsatz kommen können. Ebenso wird das weitverbreitete 8b/10b-Encoding Verfahren, wie es beispielsweise bei FCP-2 Verwendung findet, eingesetzt.

Das Serial-ATA-Protokoll ist insgesamt deutlich einfacher strukturiert, da es direkt auf den ATA/ATAPI-7 -Standards aufsetzt, und diese nur um einige Parameter erweitert (ATA/ATAPI-7 Voulme3). Die ATAPI Spezifikation implementiert die grundlegenden SCSI-Befehle innerhalb des ATA-Standards. Insbesondere mit dem Serial-ATA-2 Standard, der nicht gleichbedeutend mit der Geschwindigkeitserhöhung auf 3-GBit/Sek. zu sehen ist, wurden weitergehende Standards definiert. Beispielsweise soll Native Command Queueing es der Festplatte ermöglichen, Bestätigungen für die Abarbeitung mehrere Befehle hintere inander oder sogar gleichzeitig an den Host zu versenden. Das Laufwerk kann hier mittels eines 8 Byte großen Pakets den Status von 32 Befehlen gleichzeitig übermitteln. Ebenso gehört Click-Connect (Einrastung der Laufwerke in die Steckeranschlüsse, ähnlich SAS) sowie staggered disk spinup und weitere Subsets zur Spezifikation von Serial-ATA-2. Nur die Kombination von mindestens zwei dieser Parameter erlaubt es, ein Gerät als Serial-ATA-2 zu spezifizieren. Die häufige Gleichsetzung von Serial-ATA-2 mit der Geschwindigkeit von 3-GBit/Sek. ist also, ebenso wie 1,5-GBit/Sek. mit NCQ, nicht korrekt.

Die Einführung von SAS erlaubt jetzt, den unkomplizierten Aufbau von mehrstufigen Speicherkonzepten auf einer Plattform. Während SAS-Geräte im Bereich des Onlinespeichers angesiedelt sind, können Serial-ATA-Platten als Nearlinespeicher eingesetzt werden.

Einfaches Beispiel für die Realisierung eine mehrstufigen Speicherkonzepts mit Mischbetrieb von SAS- und Serial-ATA-2 Platten



In der Realisierung solcher mehrstufigen Speicherlösungen mit deutlich einfacheren Mitteln, als Sie momentan gegeben sind, liegt der Vorteil und auch die Zukunft des neuen SAS-Standards in Verbindung mit Serial-ATA-2 Endgeräten.

Quellen: Serial-ATA-Spezifikationen, http://www.sata-io.org/naming-guidelines.asp, www.t13.org, SAS-Spezifikationen, www.t10.org, www.sffcommittee.org, www.scsita.org

5. Festplatten und RAID

5.6 Andere RAID-Level

5.6.1 Aktuelle RAID-Level im Überblick

Angesichts aktueller Festplattensubsysteme, die eine immer größere Anzahl Festplatten wachsender Speicherkapazität in RAID-Sets organisieren können, muss überprüft werden, ob die Rahmenbedingungen einzelner RAID-Level bezüglich Verfügbarkeit, Ausfallsicherheit, Speichereffizienz und Performance noch gegeben sind. Es soll vermittelt werden, wie durch Verwendung alternativer RAID-Level eine an heutige Festplatteneigenschaften angepasste Speicherplatznutzung gestaltet werden kann. Als Einstiegspunkt werden die Schwächen der klassischen RAID-Level 3, 4 und 5 – im Folgenden als Parity-RAID-Level bezeichnet aufgezeigt.

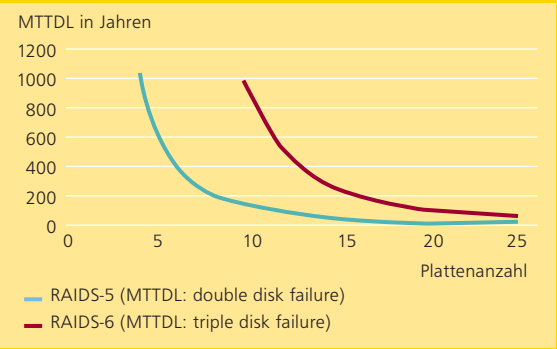
5.6.1.2 Eine Prüfsumme ist nicht sicher genug

Das klingt zunächst wie eine übertriebene Vermutung, wird aber bei genauer Betrachtung der Nachteile klassischer Parity-RAID-Level zur Tatsache. Durch Einsatz dieser RAID-Level soll bei maximaler Speichereffizienz mit minimaler Redundanz eine akzeptable Verfügbarkeit und Ausfallsicherheit geschaffen werden. Genau diese Rahmenbedingung wird durch aktuelle Festplattensubsysteme, die eine große Anzahl Festplatten hoher Speicherkapazität in einem einzigen Geräteverbund zur Verfügung stellen, zunehmend unerfüllbar.

Zur Verdeutlichung betrachten wir ein einziges RAID-Set, das N Festplatten durch RAID-Level 5 zu einem einzigen RAID-Set zusammenfasst. Die prinzipiellen Schwächen klassischer Parity-RAID-Level werden deutlich, sobald eine Festplatte ausgefallen ist (Degraded-Zustand) oder eine ausgefallene Festplatte durch eine neue ersetzt wird (Rebuild-Zustand). In diesen beiden Betriebszuständen ist der Datenbestand des betrachteten RAID-Sets in erhöhtem Maße gefährdet. Es darf weder eine zweite Festplatte ausfallen, noch darf bei Rekonstruktion der ausgefallenen Platte oder bei anderen I/O-Prozessen ein einziger Sektor auf den verbliebenen Festplatten schadhaft oder unlesbar sein.

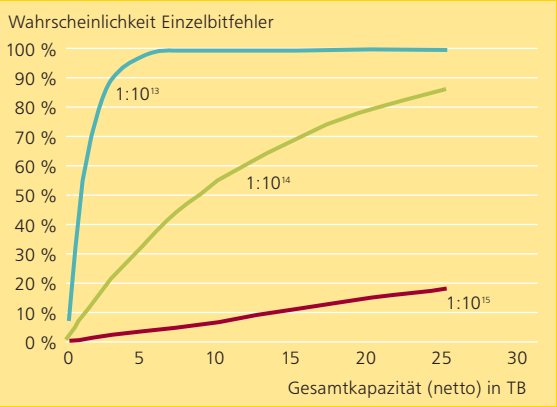
Die Wahrscheinlichkeit für einen zweiten Festplattenausfall hängt von der Gesamtzahl der Festplatten, ihrer durchschnittlichen Betriebsdauer bis zum Fehlerfall (MTTF, mean time to failure) und der Zeitspanne ab, die zur Wiederherstellung der ausgefallenen Festplatte benötigt wird. Die Wiederherstellungszeit (MTTR, mean time to repair) beinhaltet Austausch der Festplatte und die Rebuild-Dauer, die stark von der Einzelplattenkapazität und üblicherweise von der serverseitigen Auslastung des RAID-Systems abhängt. Eine Berechnungsgrundlage für die MTTDL (mean time to data loss) liefert [1]. Ein Berechnungsbeispiel zeigt Abbildung 1.

Abbildung 1: Durchschnittliche Zeit bis zum Datenverlust (MTTDL) für einzelnen RAID-5 und RAID-6-Verbund in Abhänigkeit von Plattenanzahl ohne Berücksichtigung von Bitfehlern auf dem Datenbereich. Berechnungsgrundlage MTTF(disk1)=200000h, MTTF(disk2)=20000h, MTTF(disk3)=2000h, MTTR=36h



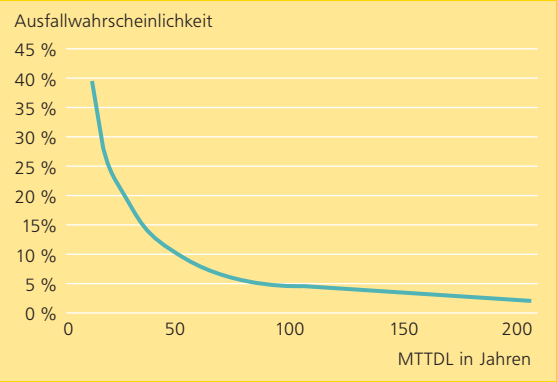
Während der Wiederherstellungszeit muss gewährleistet sein, dass auf den verbliebenen Festplatten alle Sektoren fehlerfrei bearbeitet werden können. Das Risiko eines partiellen Datenverlusts durch defekte Plattensektoren während eines Rebuilds – bei dem alle Sektoren fehlerfrei gelesen werden müssen – kann ebenfalls abgeschätzt werden. Die Wahrscheinlichkeit ist abhängig von der Gesamtkapazität des RAID-Sets und der Bitfehlerwahrscheinlichkeit der verwendeten Festplatten. Bitfehlerwahrscheinlichkeiten werden von Festplattenherstellern in einem Größenordnungsbereich von $1:10^{14}$ bis $1:10^{15}$ angegeben. Eine Abschätzung für die Wahrscheinlichkeit alle Bits fehlerfrei zu lesen liefert [1]. Einen Eindruck vermittelt Abbildung 2. Es zeigt sich, dass eine Variation der Bitfehlerrate der Einzelfestplatte von nur einer Zehnerpotenz gravierende Auswirkungen bei großer Gesamtkapazität des RAID-Sets hat.

Abbildung 2: Bitfehlerwahrscheinlichkeit in Abhängigkeit von Gesamtkapazität eines RAID-Sets und angenommner Bitfehlerhäufigkeit einer einzelnen Festplatte.



RAID-Sets mit klassischen Parity-RAID-Leveln von 5TB und mehr sind mit heutigen Festplattensubsystemen ohne weiteres machbar. Betrachtet man allerdings allein das Risiko von etwa 30 % durch defekte Sektoren, inkonsistente Dateisysteme oder fehlerbehafte Daten zu erhalten, so sind Zweifel an einer akzeptablen Verfügbarkeit, die eigentlich erreicht werden sollte, angebracht. In Kombination mit dem Risiko doppelter Plattenausfälle wird die Verfügbarkeit noch weiter minimiert. Man sollte sich durch MTTDL-Angaben von beispielsweise 100 Jahren, die sich bei entsprechenden Abschätzungen ergeben, nicht täuschen lassen, denn es besteht dabei immer noch ein Restrisiko von 5 % in einer Betriebsdauer von 5 Jahren einen Datenverlust zu erleiden (vgl. Abbildung 3).

Abbildung 3: MTTDL und Ausfallwahrscheinlichkeit für eine Betriebsdauer von 5 Jahren



Als Ausweg bleibt nur der Einsatz alternativer klassischer RAID-Level, wie beispielsweise RAID-10, RAID-30, RAID-40, RAID-50, auf Kosten der Speicherplatzeffizienz. Mehrfache Plattenausfälle sind hier aber nur in bestimmten Plattenkombinationen möglich. Insgesamt gestaltet sich heutzutage die Kompromissfindung zwischen Verfügbarkeit, Fehlertoleranz und Speicherplatzeffizienz allein aufgrund der Festplattenanzahl und der Einzelplattenkapazität zunehmend schwierig.

5.6.1.3 Eine naheliegende Idee ...

Die prinzipiellen Schwächen der klassischen RAID-Level wurden schon früh erkannt. Erste Ansätze für alternative RAID-Level zur Umgehung dieser Schwierigkeiten wurden bereits in den 1990er Jahren entwickelt [2]. Aufgrund der damaligen technischen Möglichkeiten oder der noch nicht vorhandenen Notwendigkeit erfolgte allerdings keine Implementierung. Ein klassisches Beispiel hierfür ist RAID-Level 2, der durch Verwendung von Hamming-Codes zumindest Bitfehler innerhalb der Platten korrigieren kann. Eine kommerzielle Verbreitung erfolgte jedoch nicht.

Die prinzipielle Idee der zunehmend auf dem Markt erscheinenden „neuen" RAID-Implementierungen ist naheliegend: Durch Verwendung einer zweiten unabhängigen Prüfsumme werden zweifache Ausfälle beliebiger Platten möglich. Damit verbunden ist auch ein besserer Schutz vor defekten Plattensektoren, da zusätzliche Korrekturmöglichkeiten verfügbar sind. Man investiert als Kompromiss an die Speichereffizienz in die Plattenkapazität einer zusätzlichen Einzelfestplatte. Dank dieser zusätzlichen Redundanz gewinnt man bessere Fehlertoleranz und Verfügbarkeit durch die Tatsache, dass ein beliebiges paar Festplatten gleichzeitig ausfallen kann.

Abbildung 4: Speichereffizienz (Verhältnis Netto- zu Bruttokapazität)

Festplatten-anzahl	RAID-01, RAID-10	RAID-3, RAID-4, RAID-5	RAID-6, RAID-DP etc.
3	–	66,7 %	–
4	50 %	75,0 %	50 %
5	–	80,0 %	60 %
8	50 %	87,5 %	75 %
10	50 %	90,0 %	80 %

Die weitergehende Verallgemeinerung: Man bildet einen Array von n+m Festplatten. Die Datenpakete werden im Speicherbereich, den n Festplatten bieten, abgelegt. Dann werden m unabhängige Prüfsummen gebildet, die auf der verbleibenden Speicherkapazität, die m Einzelfestplatten entspricht, verteilt werden. Man spricht hier allgemein von RAID-n+m, was deutlich machen soll, dass m Festplatten gleichzeitig ausfallen können. RAID-n+m ist lediglich als Bezeichnung für eine RAID-Level-Klasse zu verstehen, die über die Einzelheiten der verwendeten Algorithmen zunächst nichts aussagt. Diesem Sprachgebrauch folgend sind RAID-6, RAID-DP, RAID 5DP Vertreter der RAID-n+2-Klasse.

5.6.1.4 Unterschiedliche Umsetzungen unter verschiedenen Bezeichnungen.

Verfügbare Implementierungen wie RAID-6, RAID-DP, RAID-5DP, RAID-n und andere unterscheiden sich zunächst im Verfahren, eine zweite oder weitere unabhängige Prüfsummen zu gewinnen. Unabhängige Prüfsummen lassen sich einerseits durch zweifache XOR-Bildung gewinnen oder andererseits durch Verwendung alternativer fehlerkorrigierender Kodierungsverfahren. Ein weiteres Unterscheidungsmerkmal ist die Art und Weise wie Daten- und Prüfsummenpakete auf die Festplatten verteilt werden. Die Verteilung kann analog zu RAID-4 mit dedizierten Prüfsummenplatten oder ähnlich zu RAID-5 mit gleichmäßig über alle Platten verteilten Prüfsummen erfolgen.

5.6.1.5 Zweifaches XOR

Beispiele für dieses Verfahren sind EVENODD [2], RDP (Row Diagonal Parity) [3], RAID-DP und RAID 5DP. RAID-DP von Network Appliance ist eine spezielle Form allgemeiner RDP-Verfahren und steht im NetApp-Betriebssystem zur Verfügung. RAID 5DP (RAID 5 Double Parity) ist die Bezeichnung von HP für einen RAID-Level der VA7000 Serie.

Allgemein entspricht die erste Prüfsumme der üblichen XOR-Prüfsumme, die auch bei RAID-3, RAID-4 oder RAID-5 verwendet wird. Sie wird horizontal über die einzelnen Datenpakete gebildet. Die zweite Prüfsumme wird ebenfalls durch einen XOR-Algorithmus errechnet. Die Unabhängigkeit der beiden Prüfsummen wird durch die Verwendung unterschiedlicher Datenpakete gewährleistet. Die Berechnung der zweiten XOR-Prüfsumme erfolgt daher über diagonal liegende Datenpakete.

Die Daten- und die Prüfsummenpakete können in einer zu RAID-4 analogen Weise mit dedizierten Daten- und Prüfsummenplatten umgesetzt werden, was den Vorteil einer unkomplizierten Absicherung bestehender RAID-4-Sets durch Erweiterung um eine weitere Prüfsummenplatte bietet. Die bekannten Einschränkungen von RAID-4, die durch dedizierte Prüfsummenplatten entstehen, bleiben dabei erhalten. Umgekehrt ist eine Rückkehr zu RAID-4 ebenfalls ohne weiteres durch Abkoppeln der zusätzlichen Prüfsummenplatte möglich. Die von RAID-5 bekannte gleichmäßige Verteilung von Daten- und Prüfsummenblöcken auf alle Platten kann ebenfalls umgesetzt werden, wobei allerdings darauf geachtet werden muss, durch die Umverteilung die Unabhängigkeit beider Prüfsummen nicht zu zerstören.

Abbildung 5: RAID-4

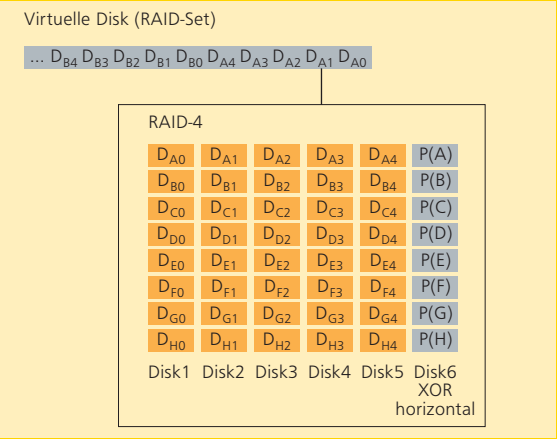
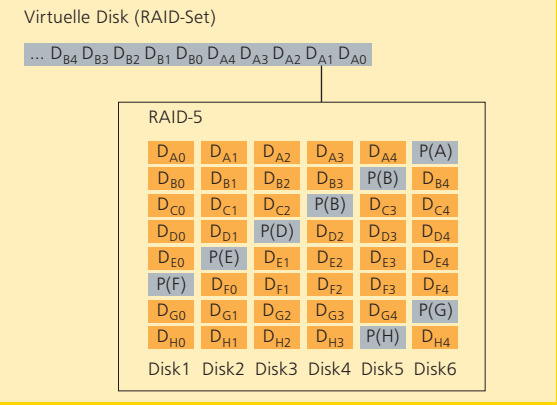


Abbildung 6: RAID-5



Als Beispiele für zweifaches XOR im RAID-4-Stil dienen EVENODD und RAID-DP. Beide Verfahren gemeinsam ist die Tatsache, dass die Unabhängigkeit beider Prüfsummen nur bei Verwendung einer bestimmten Anzahl von Festplatten gegeben ist. Bei EVENODD muss die Anzahl der Datenplatten eine Primzahl sein. Bei RAID-DP die Anzahl der Datenplatten zuzüglich der horizontalen Prüfsummenplatte. Für den Fall einer beliebigen Festplattenanzahl wird der Algorithmus um virtuelle Festplatten, die keine Daten – also nur 0 – enthalten, erweitert, bis das jeweilige Primzahlkriterium erreicht ist. Diese virtuellen Festplatten dienen nur als Platzhalter im Rechenverfahren, um die Unabhängigkeit beider Prüfsummen bei Verwendung einer beliebigen Anzahl von Festplatten zu gewährleisten.

Abbildung 7: EVENODD

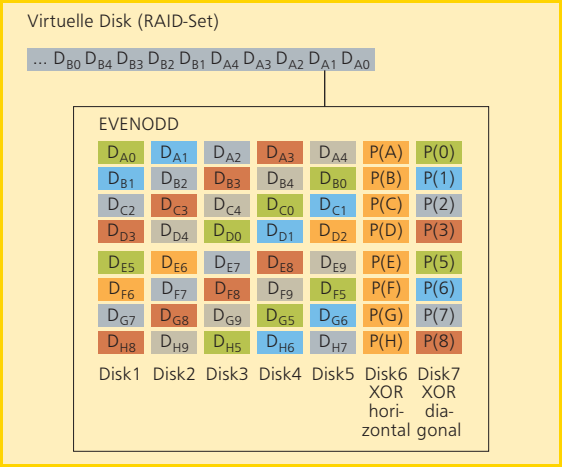
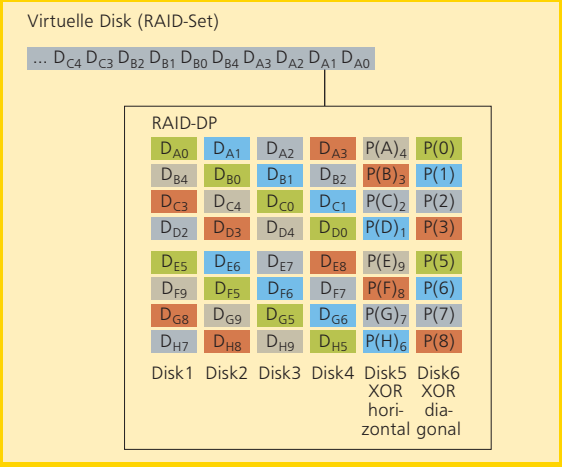


Abbildung 8: RAID-DP



Als weitere Gemeinsamkeit fällt auf, dass nicht alle diagonalen Prüfsummen benötigt werden. Jeweils eine Diagonale – in den Abbildungen weiße Platten – bleibt in Bezug auf diagonale Prüfsummenberechnung unbenutzt. Zudem lassen sich diese Verfahren auf herkömmlicher RAID-Hardware mit XOR-Engine ohne weiteres implementieren.

Bei RAID-DP werden – im Unterschied zu EVENODD – die Daten der horizontalen Prüfsummenplatte in die diagonale Prüfsummenbildung miteinbezogen, was zu einer Verringerung der XOR-Operationen gegenüber EVENODD, insbesondere bei kleiner Festplattenanzahl, führt.

5.6.1.6 Alternative Prüfsummenbildung

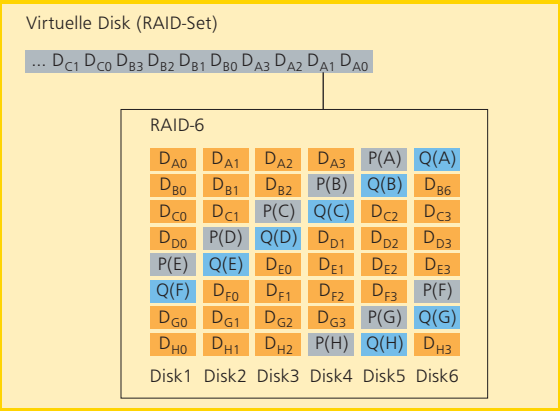
Die langfristige Zielsetzung allgemeine RAID-n+m-Level verfügbar zu machen, kann mit mehrfachen XOR-Algorithmen nicht erreicht werden. Sie sind auf den simultanen Ausfall zweier Platten beschränkt. Nur die Verwendung von allgemeineren fehlerkorrigierenden Kodierungsverfahren führt zum Ziel. Kandidaten sind beispielsweise Reed-Solomon-Codes, Vandermonde-based Reed-Solomon-Codes, Bose-Chaudhuri-Hocquenghem-Codes, die von RAID-2 bekannten Hamming-Codes, Gallager- und Tornado-Codes, wobei die beiden letztgenannten patentrechtlich geschützt sind. Im allgemeinen werden zunächst RAID-n+2-Implementierungen verfügbar sein, deren Kodierungsverfahren aber prinzipiell auf den allgemeinen RAID-n+m-Fall erweiterbar sind. Die zugrunde liegende Mathematik ist durchaus als nicht trivial zu bezeichnen und ist nicht so leicht nachvollziehbar wie die XOR-Prüfsummenbildung. Zudem sind die Verfahren als rechenintensiv zu bezeichnen.

Im Unterschied zu den zweifachen XOR-Verfahren arbeiten diese Kodierungsverfahren im allgemeinen innerhalb eines Stripes, was die zu bearbeitenden Daten während Schreiboperationen im Vergleich zu erstgenannten minimiert. Fraglich bleibt, ob daraus – wegen des im allgemeinen höheren Rechenaufwands – ein Performancevorteil erwächst.

5.6.1.9 RAID-6

Aus den Datenpaketen werden mittels Reed-Solomon-Codes zwei Prüfsummen P und Q – Syndrome genannt – berechnet. Reed-Solomon-Codes sind fehlerkorrigierende Kodierungsverfahren, die bereits in den 1960er Jahren entwickelt wurden. In einer allgemeineren Sichtweise als bei CRC-Verfahren betrachtet man statt Bits Zeichen aus mehreren Bits und konstruiert einen endlichen algebraischen Zahlkörper (Galois-Körper) auf dem eine Addition und Multiplikation definiert werden. Die so definierte Addition und Multiplikation wird zur Berechnung der Prüfsummen benutzt. Das Reed-Solomon-Verfahren ist ein Blockkodierungsverfahren, das jede vorgegebene Anzahl von Fehlern in einem Block korrigieren kann. Eingesetzt wird es u.a. bei Datenspeicherung auf CDs, DVDs und bei DVB-Technologien (Digital Video Broadcasting). Für den Spezialfall zweier Syndrome P und Q operiert man auf dem Zahlkörper GF(2^8) was die Gesamtzahl der Platten auf 256 beschränkt. Die Berechnung von P reduziert sich hierbei auf ein einfaches XOR. Lediglich die Berechnung von Q ist aufwändiger [4].

Abbildung 9: RAID-6



Als Software-Lösung ist RAID-6 seit Kernel 2.6.2 Bestandteil des md-Treibers von Linux. Intel® liefert mit dem IOP333 und IOP331 Hardwarebeschleunigung für RAID-6-Berechnung.

5.6.1.10 RAIDⁿ

RAIDⁿ ist die Bezeichnung für eine Familie höherer RAID-Level der Tandberg-Tochterfirma Inostor. Laut Firmenangaben werden keine Reed-Solomon-Codes benutzt. Es existiert ein US-Patent (Nr. 6.557.123) über nicht näher erläuterte Algorithmen. RAIDⁿ ist wohl hauptsächlich in Hardware-Lösungen von Tandberg und Inostor integriert, obwohl es auch eine Softwarelösung in Form von Linux-Kernelmodulen gibt.

5.6.1.11 RAID-X

Ein zurzeit scheinbar noch in der Designphase befindlicher höherer RAID-Level der Firma ECC Technologies, der durch bitweises Striping unter Verwendung von Reed-Solomon-Codes im Prinzip eine Erweiterung von RAID-3 ist.

Zusammenfassung

Die Notwendigkeit für alternative RAID-Lösungen jenseits der klassischen RAID-Level ist offensichtlich. Aktuell sind einige Geräte wie komplette externe Hardware-Lösungen, PCI-RAID-Controller oder Softwarelösungen mit zeitgemäßen RAID-Implementierungen in Marktreife verfügbar. Bereits der uneinheitliche Sprachgebrauch jedoch zeigt, dass eine Standardisierung noch nicht in Sicht ist. Deshalb dürfte die weitere Entwicklung in diesem Bereich die nächsten Jahre durchaus spannend zu verfolgen sein. Bei Anschaffung entsprechender Geräte lohnt sich ein genauer Blick hinter die Kulissen.

Quellen und Literatur

[1] „RAID: High-Performance, Reliable Secondary Storage" von P.M. Chen, E.K. Lee, G.A. Gibson, R.H. Katz und D.A. Patterson in „ACM Computing Surveys, Vol 26, No. 2, June 1994", pgs. 145 – 185.

[2] „EVENODD: An efficient scheme for tolerating double disk failures in RAID architectures" von M. Blaum, J. Brady, J. Bruck und J. Menon in „Proceedings of the Annual International Symposium on Computer Architecture", pgs. 245 – 254, 1994.

[3] „Row-Diagonal Parity for Double Disk Failure Correction" von Peter Corbett, Bob English, Atul Goel, Tomislav Grcanac, Steven Kleiman, James Leong und Sunitha Sankar in „Proceedings of the Third USENIX Conference on File and Storage Technologies" März/April 2004.

[4] „The mathematics of RAID-6" von H. Peter Anvin; http://www.kernel.org/pub/linux/kernel/people/hpa/

6. Speichernetzwerke

6.5 Network Attached Storage (NAS)

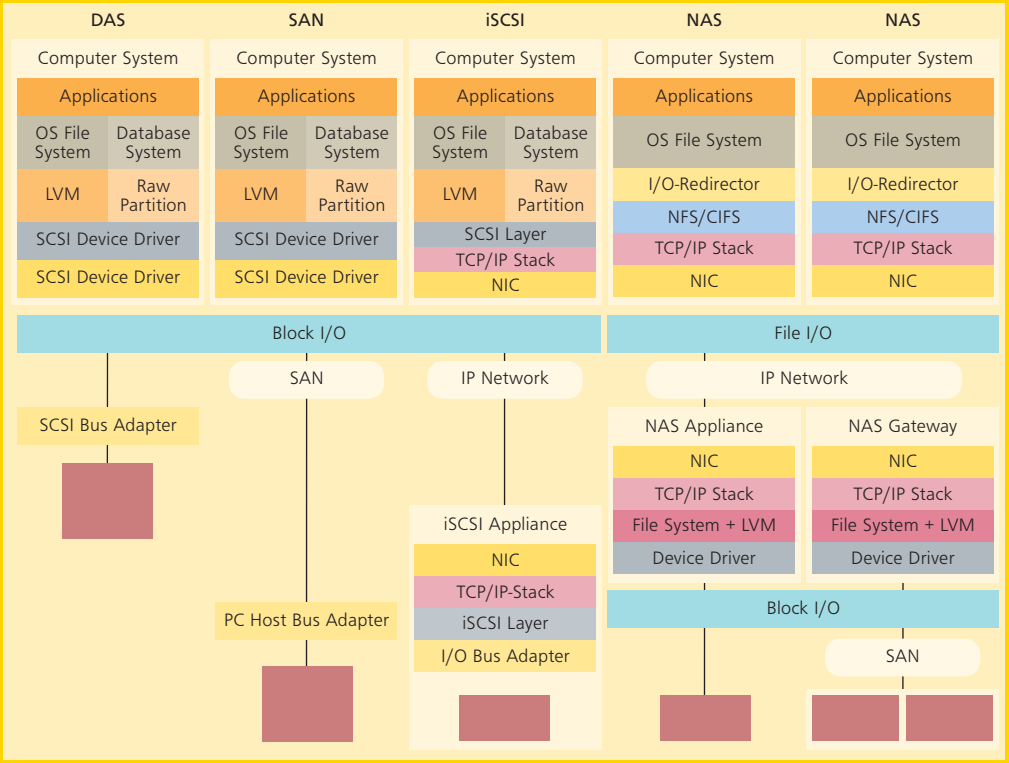
Als Netzwerkfunktionen erstmals in UNIX allgemein verfügbar wurden, mussten sich Benutzer, die Dateien gemeinsam nutzen wollten, im Netz bei einem Zentralrechner einloggen, auf dem sich die gemeinsam genutzten Dateien befanden. Die Zentralrechner waren schnell weit stärker ausgelastet als der lokale Rechner des Benutzers, so dass sich rasch eine immer stärkere Nachfrage nach einer bequemen Möglichkeit für einen gleichzeitigen Dateizugriff von mehreren Rechnern ergab. Als Network Attached Storage (NAS) bezeichnet man Geräte oder Software (z. B. Samba), mit denen man Dateien über IP-Netzwerke mit Hilfe eines oder mehrerer dezentraler Dateisysteme mit mehreren Benutzern gleichzeitig nutzt. Solche Geräte werden daher auch oft Fileserver, Dateiserver oder Filer genannt.

Zum Verständnis von NAS ist es wichtig, die Konzeption eines dateibasierten Speicher-I/O im Unterschied zum blockbasierten Speicher-I/O zu verstehen, wie es von SAN- (iSCSI, FC) und DAS-Speichern verwendet wird. Die nachstehende Abbildung vergleicht zusammenfassend die verschiedenen I/O-Pfade von block- und dateibasierten Speicherzugriffen.

6.5.1 Dateisysteme des Betriebssystems

Alle I/O-Anforderungen eines NAS-Gerätes werden vom Dateisystem seines Betriebssystems bearbeitet. Ein Dateisystem oder File System ist die physikalische Struktur, die ein Betriebssystem für die Speicherung und Anordnung der Dateien auf einem Speichermedium verwendet.

Auf BIOS-Ebene enthält eine Festplattenpartition eine Anzahl nummerierter Sektoren. Jede Partition könnte als ein großer Datensatz betrachtet werden, das aber würde zu einer ineffizienten Belegung des Speicherplatzes führen und die Anforderungen von Software-Applikationen nicht effektiv erfüllen. Zur Verwaltung der Datanorganisation auf der Festplatte nutzt das Betriebssystem eine hierarchische Verzeichnisstruktur. Dateien oder weitere Verzeichnisse, die so genannten Unterverzeichnisse. Die Verzeichnisstruktur und die Methoden zur Organisation der Festplattenpartitionen werden als Dateisystem bezeichnet. Dateisysteme verwalten den Speicherplatz für Daten, die von den Anwendungen kreeiert und verwendet werden. Der hauptsächliche Zweck eines Dateisystems ist die Verbesserung der Datenverwaltung, indem es eine separate Organisation und Verwaltung der verschiedenen Arten von Information ermöglicht.



Das Dateisystem wird durch einen Satz von Betriebssystembefehlen implementiert, die die Anlage, Verwaltung und Löschung von Dateien ermöglichen. Ein Satz Subroutinen ermöglicht den Zugriff auf niedrigerer Ebene, z.B. Öffnen, Lesen, Schreiben und Schließen von Dateien im Dateisystem. Das Dateisystem definiert Dateiattribute (Read-

only-Datei, Systemdatei, Archivdatei usw.) und ordnet den Dateien Namen entsprechend einer Benennungskonvention zu, die für das Dateisystem spezifisch ist. Das Dateisystem definiert auch die Maximalgröße einer Datei und verwaltet den verfügbaren freien Speicherplatz für die Anlage neuer Dateien.

Ein Dateisystem arbeitet jedoch nicht direkt mit der Festplatte. Es arbeitet mit abstrakten logischen Abbildern des Festplattenspeichers, die durch die Volume Manager-Funktion angelegt werden. Mit anderen Worten, die Festplatte kann virtuell oder real sein. Aus der Sicht des Dateisystems spielt dies keine Rolle. Das Dateisystem bewahrt einen Index der Daten, die auf der Festplatte gespeichert sind, einschließlich der virtuellen Laufwerke. Anhand dieses Index findet das Dateisystem den freien Speicherplatz zum Speichern einer Datei. Es wandelt dann die ursprüngliche Datei-I/O-Anforderung in Speicherbefehle um (eine Reihe von Block-I/O-Operationen). Schließlich legt das Dateisystem Metadaten an (Daten zur Dateibeschreibung), die für System- und Speicherverwaltungszwecke verwendet werden und legt die Zugriffsrechte für die Datei fest.

High-End NAS-Geräte von Herstellern wie Network Appliance oder EMC verwenden oft UNIX-Abkömmlinge als Betriebssystem und herstellereigene Dateisysteme, die für maximale I/O-Effizienz und Speicherkapazität konzipiert sind. Preiswertere NAS-Systeme des mittleren und Einstiegssegments verwenden Standard-Betriebssysteme und Dateisysteme wie Windows (NTFS) oder Linux (ext2, ext3, Riser FS).

Eine Festplatte kann Partitionen enthalten, deren Dateisysteme zu verschiedenen Betriebssystemen gehören. Im Allgemeinen ignoriert ein Betriebssystem die Partitionen, deren ID für ein unbekanntes Dateisystem steht. Das Dateisystem ist gewöhnlich eng in das Betriebssystem eingebunden. In Speichernetzwerken kann es jedoch vom Betriebssystem getrennt und auf mehrere dezentrale Rechnerplattformen verteilt werden. Damit wird es möglich, auf ein entferntes Dateisystem (oder einen Teil davon) so zuzugreifen, als wäre es Teil eines lokalen Dateisystems. Weiter unten wird erläutert, wie dies im Network File System (NFS), wie dies am Beispiel des Network File System geschieht.

6.5.2 I/O-Redirector

Ein NAS-Gerät ermöglicht es den Benutzern Dateien über logische Laufwerke und gemeinsame Gruppenverzeichnisse aufzurufen, als befänden sie sich auf dem lokalen Rechner des Benutzers. Wenn ein Benutzer oder eine Anwendung eine Datei-I/O-Anforderung sendet, um eine solche auf einem NAS-System gespeicherte Datei aufzurufen, kann das lokale Dateisystem die I/O-Anforderung nicht verwalten, da es keine Informationen über das Speichermedium besitzt, auf dem die Datei gespeichert ist. Um diese Schwierigkeit zu überwinden, muss die I/O-Anforderung über das Netzwerk an das NAS-Gerät umgeleitet werden.

Ein I/O-Redirector befindet sich im I/O-Pfad des Clients vor dem lokalen Client-Dateisystem (Siehe Abbildung Seite 131). Er enthält eine gemeinsame Ansicht des lokalen Client-Dateisystems und des Dateisystems des entfernten NAS-Geräts. Der I/O-Redirector kennt jedoch nicht die Metadaten der beiden Dateisysteme. Eine I/O-Anforderung an eine Datei, die sich im entfernten NAS-Gerät befindet, wird vom Redirector am Client-Computer abgefangen. Der Redirector stellt dann ein Datenpaket mit den gesamten Anforderungsinformationen zusammen und sendet es über die Netzwerkkarte (Network Interface Card, NIC) des Client und ein Ethernet-LAN/WAN an das NAS-System, auf dem sich die Datei befindet. Da dem Client-System die Eigenschaften des Speichermediums unbekannt sind, auf dem die Datei gespeichert ist, müssen alle umgeleiteten I/Os auf Dateiebene (Bytebereich) erfolgen. Dies wird als File I/O bzw. Datei-I/O bezeichnet.

Da die Netzschnittstelle ein Netzwerkprotokoll verwendet, z.B. TCP/IP-Stacks oder seltener UDP/IP, muss die I/O-Operation mit Hilfe des Netzwerkprotokolls übertragen werden. Nun kommt eines der Netzwerkdateiprotokolle wie NFS (Unix/Linux), SMB/CIFS (Windows), NCP (NetWare) oder AppleTalk (Mac OS) als eine Art Netzwerkgerätetreiber ins Spiel. Tatsächlich befindet sich das Netzwerkdateiprotokoll über der anderen Ebene des Kommunikationsprotokolls. Z. B. TCP/IP das TCP/IP-Protokoll transportiert die umgeleitete I/O-Anforderung durch die Ethernet-Karte in das Netzwerk.

Wenn das NAS-Gerät die umgeleitete I/O-Anforderung erhält, wird sie in der empfangenen Netzwerkkarte von ihren TCP/IP Protokollbestandteilen befreit und zum Netzwerkdateiprotokoll des NAS-Geräts gesandt. Es kontrolliert die Informationen zur Rückverfolgung, um die Antwort an die Netzwerkadresse des richtigen Clients leiten zu können. Nun gelangt die Anforderung zum Betriebssystem des NAS-Geräts, das die I/O-Befehl verwaltet. Ab diesem Punkt wird die I/O-Anforderung mehr oder weniger wie eine lokale behandelt und durch Schreib-/Lesebefehle auf den Blöcken des NAS_Speichermediums abgearbeitet. Schließlich folgt die Antwort auf die I/O-Anforderung derselben oben beschriebenen Route in umgekehrter Richtung.

6.5.3 Network File System (NFS)

Das Network File System (NFS) war das erste kommerziell erfolgreiche und weithin verfügbare Dateiprotokoll für den Zugriff auf entfernte Dateien. Ursprünglich von Sun Microsystems im Jahr 1985 entwickelt und implementiert, wurde die Protokollspezifikation öffentlich zugänglich gemacht. Von Anfang an war das NFS für den Zugriff auf entfernte Dateien und ihre gemeinsame Nutzung durch verschiedene Rechner, Betriebssysteme, Netzwerkarchitekturen und Übertragungsprotokolle konzipiert. Bis heute wird es unter der Aufsicht der Internet Engineering Task Force (IETF) stetig erweitert und standarisiert.

NFS Version 2 wurde als offizieller TCP/IP-Standard niedergelegt, als RFC 1094 im Jahr 1989 veröffentlicht wurde. Im Anschluss daran wurde NFS Version 3 entwickelt und als RFC 1813 im Jahr 1995 veröffentlicht. Es ähnelt Version 2, ist aber leicht verändert und hat einige Funktionen mehr. Dazu gehören die Unterstützung für größere Dateien, die Übertragung größerer Dateien und eine bessere Unterstützung für das Setzen von Dateiattributen sowie mehrere neue Prozeduren für den Dateizugriff und die Dateibehandlung. NFS v2/3 sind immer noch die am weitesten verbreiteten Versionen, während NFS v4 als neuester Standard im Jahr 2000 als RFC 3010 veröffentlicht wurde und praktisch eine komplette Erneuerung von NFS mit zahlreichen Änderungen war.

NFS folgt dem klassischen Server/Client-Modell und besteht aus einem Serverprogramm und einem Clientprogramm. Mit Hilfe des Serverprogramms können Administratoren lokale Laufwerke, Verzeichnisse oder Dateien als shared Ressourcen definieren und sie für den Zugriff von anderen Rechnern über einen so genannten Exportprozess verfügbar machen. NFS-Clients greifen auf freigegebene Dateisysteme zu, indem sie sie von einem NFS-Server mounten. NFS basiert auf einer Architektur aus drei Hauptkomponenten, die seinen Betrieb definieren: Das Mount-Protokoll dient dazu, Ressourcen zu „mounten", und ermöglicht es dem Server, über die Exportkontrolle einer bestimmten Anzahl Clients Zugriffsprivilegien zu gewähren. Der XDR-Standard (External Data Representation) definiert, wie Daten beim Austausch zwischen Clients und Servern dargestellt werden. Das RPC-Protokoll (Remote Procedure Call) schließlich dient als Methode für den Aufruf von Prozeduren auf entfernten Rechnern.

Das NFS-Protokoll wurde zustandslos konzipiert. Der Server braucht keine Informationen über die Clients, die er gerade bedient, oder über die Dateien, die diese gerade geöffnet haben, aufrechtzuerhalten. Da es keinen Zustand gibt, der aufrechterhalten oder wiederhergestellt werden müsste, ist NFS sehr robust und kann auch bei vorübergehendem Ausfall eines Clients oder des Servers in Betrieb bleiben. Allerdings hat das zustandslose Protokoll auch Nachteile im Hinblick auf die Leistungsfähigkeit und das Freisetzen von Dateispeicherplatz.

NFS v2 hat 16 verschiedene RPCs und lief ursprünglich vollständig über das unzuverlässige User Datagram Protocol (UDP). UDP ist zwar schneller als TCP, bietet aber keinen Fehlercheck. NFS sorgte mit Hilfe der eingebauten Retry-Logik der RPCs dafür, dass Anforderungen und Antworten an ihrem Zielort ankommen. Der Client kann die Blockgrößen, die Anzahl der Retry-Versuche und Werte für die Wartezeit angeben, wenn er die Server-Dateisysteme mountet. Bevor ein Client eine RPC-Befehl an den Server absendet, prüft er, ob die gewünschten Daten schon aus einer früheren Anforderung im Cache sind. Wenn die Daten neuer sind als der Timeout-Wert des Cache-Attributs, werden die Daten verwendet, sonst sendet der Client eine Anforderung an den Server, die Änderungszeit seiner im Cache befindlichen Datei mit derjenigen der Server-Datei zu vergleichen. Wenn die Server-Datei neuer ist, erfolgt eine Anforderung, die Daten noch einmal zu senden.

NFS v3 bot einige wesentliche Verbesserungen gegenüber früheren Versionen. NFS ist jetzt auf dem TCP-Protokoll lauffähig. Außerdem unterstützt es nun sichere asynchrone Schreibzugriffe, eine verfeinerte Zugriffskontrolle und größere Dateiübertragungsgrößen mit geringerem Overhead. Da NFS zustandslos ist, muss man darauf achten, dass der Server auch wirklich die Schreibanforderung in einen stabilen Speicherbereich durchgeführt hat, bevor die Bestätigung an den Client erfolgt. Version 3 ermöglicht es, unsichere, asynchrone Schreibzugriffe zuverlässig auf einem stabilen Speichermedium abzulegen. Außerdem wurde die maximale Übertragungsgröße von 8 KB auf 4 GB erweitert, wobei die Rechner die Übertragungsgröße bis zu einer Größe von 64 KB automatisch aushandeln, das zulässige Maximum für UDP und TCP. Das Protokoll, entweder TCP oder UDP, wird von den Rechnern ebenfalls ausgehandelt. Wenn beide Seiten TCP unterstützen, wird standardmäßig TCP eingestellt. Das neue Protokoll ermöglicht nun Datei-Offsets von 64-Bit-Dateien im Gegensatz zur früheren 32-Bit-Beschränkung und unterstützt beliebig große Dateien. Die neue Version ist effizienter, z. B. sendet sie nach jedem Aufruf die Dateiattribute zurück, so dass nicht länger eine seperate Aufforderung für diese Information gesendet werden muss.

Die Kapitel 7 – 14 finden Sie online unter
www.transtec.de
www.transtec.at
www.transtec.ch

15. Local Area Networks – LANs

15.5 Ethernet-Standards

Von Ethernet gibt es eine Vielzahl von Realisierungen, die sich zum grössten Teil in der Geschwindigkeit, den Übertragungsverfahren, der maximalen Segmentlänge sowie Konnektortypen unterscheiden. So gibt beispielsweise die Nomenklatur des Standards 10 BASE-5 nach IEEE802.3 an, dass es sich um ein Ethernet mit 10 MBit/Sek. Basisbandübertragung mit einer maximalen Segmentlänge von 500 m handelt. Im Folgenden werden nun die wichtigsten Standards, die in der Praxis Anwendung gefunden haben und finden, erläutert.

Ethernet-Type	Kabelart	Stecker	Länge (m)
10Base-5	Yellow cable	AUI	500
10Base-2	Coax	BNC	185
10Base-T	CAT3	RJ-45	100
10Base-FL	MM-fibre	ST	2.000
100Base-TX	CAT5	RJ-45	100
100Base-FX	MM-fibre	SC/MT-RJ	2.000
1000Base-T	CAT5	RJ-45	100
1000Base-SX	MM-fibre	LC/SC/MT-RJ	270/550
1000Base-LX	MM/SM-fibre	LC/SC/MT-RJ	550/500

15.5.1 10 BASE-5

Der Standard 10 BASE-5 nach IEEE802.3 gibt an, dass es sich um ein Ethernet mit 10 MBit/Sek. Basisbandübertragung und einer maximalen Übertragungsstrecke von 500 m handelt. 10 BASE-5 ist auch unter dem Synonym Thick Net bekannt, da es als Übertragungsmedium dickes 50 Ohm Koaxialkabel RG8 verwendet. Physikalisch verwendet Thick-Net die Bus-Topologie, wobei die 5-4-3 Repeater-Regel zu berücksichtigen ist. Dies bedeutet, dass in einem Thick Net maximal 5 Segmente mit einer maximalen Segmentlänge von 500 m über 4 Repeater miteinander verbunden werden können. Dadurch ergeben sich maximal 3 Inter Repeater Links (IRL). Pro Segment können maximal 100 Stationen angeschlossen werden, wobei als Konnektortyp AUI (Attachement Unit Interface) verwendet wird; die maximale Länge des AUI-Kabels beträgt 50 m. Der Standard 10 BASE-5 hat in der Vergangenheit Verwendung gefunden, ist jedoch in Neuinstallationen nicht mehr vorhanden.

15.5.2 10 BASE-2

Der Standard 10 BASE-2 nach IEEE802.3a gibt an, dass es sich um ein Ethernet mit 10 MBit/Sek. Basisbandübertragung und einer maximalen Segmentlänge von 185 m handelt. 10 BASE-2 ist auch unter dem Synonym Thin Net oder Cheaper Net bekannt, da es als Übertragungsmedium dünnes 50 Ohm Koaxialkabel RG58 verwendet. Physikalisch verwendet Thin Net die Bus-Topologie, wobei die minimale Segmentlänge zwischen zwei Stationen 0,5 m beträgt und maximal 4 Repeater zwischen zwei Stationen geschaltet werden dürfen. Pro Segment können maximal 30 Stationen angeschlossen werden, wobei als Konnektortyp BNC-T-Adapter verwendet werden. Der Standard 10 BASE-2 hat in näherer Vergangenheit breite Verwendung gefunden, ist jedoch in Neuinstallationen nicht mehr vorhanden.

15.5.3 10 BASE-T

Der Standard 10 BASE-T nach IEEE802.3i gibt an, dass es sich um ein Ethernet mit 10 MBit/Sek. Basisbandübertragung und einer maximalen Segmentlänge von 100 m bei kupferbasierender Verkabelung handelt. Als Übertragungsmedium werden Kupfer Twisted Pair Kabel verschiedener Standards eingesetzt, die als Konnektortyp RJ-45-Konnektoren verwenden. Auf die unterschiedlichen Standards von Verkabelung soll in einem späteren Kapitel noch einmal eingegangen werden. Physikalisch verwendet 10 BASE-T die Stern-Topologie, d. h., es wird eine aktive Komponente zum sternförmigen Konzentrieren der Stationen verwendet; der Konzentrator dient hierbei gleichzeitig als Verstärker. Der Standard 10 BASE-T hat in näherer Vergangenheit breite Verwendung gefunden und hat eine breite installierte Basis, ist jedoch in Neuinstallationen nicht mehr vorhanden.

15.5.4 10 BASE-FL

Der Standard 10 BASE-FL nach IEEE802.23j gibt an, dass es sich um ein Ethernet mit 10 MBit/Sek. Basisbandübertragung und einer maximalen Segmentlänge von 2000 m bei glasfaserbasierender Verkabelung handelt. Als Übertragungsmedium werden Glasfaser-Duplexkabel eingesetzt, die als Konnektortyp häufig ST-Konnektoren verwenden. Auf die unterschiedlichen Standards von Verkabelung soll in einem späteren Kapitel (15.9) noch einmal eingegangen werden. Der Standard ist eine Erweiterung von FOIRL (Fiber Optic Inter Repeater Link) und definiert die Verbindungen zwischen Konzentratoren als auch zwischen Stationen und Konzentratoren. Der Standard 10 BASE-FL hat in näherer Vergangenheit breite Verwendung gefunden und hat eine breite installierte Basis, ist jedoch in Neuinstallationen nicht mehr vorhanden.

15.5.5 100 BASE-TX

Der Standard 100 BASE-TX nach IEEE802.3u gibt an, dass es sich um ein Fast Ethernet mit 100 MBit/Sek. Basisbandübertragung und einer maximalen Segmentlänge von 100 m bei kupferbasierender Verkabelung handelt. Als Übertragungsmedium werden Kupfer Twisted Pair Kabel verschiedener Standards eingesetzt, die als Konnektortyp RJ-45-Konnektoren verwenden. Auf die unterschiedlichen Standards von Verkabelung soll in einem späteren Kapitel (15.9) noch einmal eingegangen werden. Physikalisch verwendet 100 BASE-TX die Stern-Topologie, d. h., es wird eine aktive Komponente zum sternförmigen Konzentrieren der Stationen verwendet; der Konzentrator dient hierbei gleichzeitig als Verstärker. Der Standard 100 BASE-TX hat eine breite installierte Basis und wird sehr häufig in Neuinstallationen verwendet.

15.5.6 100 BASE-FX

Der Standard 100 BASE-FX nach IEEE802.3u gibt an, dass es sich um ein Fast Ethernet mit 100 MBit/Sek. Basisbandübertragung und einer maximalen Segmentlänge von 400 m zwischen Stationen und Konzentratoren und 2000 m ziwschen Konzentratoren handelt. Als Übertragungsmedium werden Glasfaser-Duplexkabel eingesetzt, die als Konnektortyp häufig ST-, SC-, MT-RJ-, LC- oder VF-45-Konnektoren verwenden. Auf die unterschiedlichen Standards von Verkabelung soll in einem späteren Kapitel (15.9) noch einmal eingegangen werden. 100 BASE-FX ist dabei auf dem FDDI (Fiber Distributed Data Interface) aufgesetzt, welcher allerdings nach dem Time-Token-Verfahren arbeitet. Der Standard 100 BASE-FX hat eine installierte Basis und wird ebenfalls für Neuinstallationen im LWL-Umfeld eingesetzt.

15.5.7 1000 BASE-T

Der Standard 1000 BASE-T nach IEEE802.3ab gibt an, dass es sich um ein Gigabit Ethernet mit 1000 MBit/Sek. Basisbandübertragung und einer maximalen Segmentlänge von 100 m im Anschlussbereich handelt. Als Übertragungsmedium werden Kupfer Twisted Pair Kabel verschiedener Standards eingesetzt, die als Konnektortyp RJ-45-Konnektoren verwenden. Auf die unterschiedlichen Standards von Verkabelung soll in einem späteren Kapitel (15.9) noch einmal eingegangen werden. Physikalisch verwendet 1000 BASE-TX die Stern-Topologie, d. h., es wird eine aktive Komponente zum sternförmigen Konzentrieren der Stationen verwendet; der Konzentrator dient hierbei gleichzeitig als Verstärker. Der 1000 BASE-T Standard setzt auf den Standard 100

BASE-T2 und 100 BASE-T4 auf, die Eigenschaften für die Übertragung über Kategorie-3-Kupferkabel spezifizierten und hierbei mehr als zwei Adernpaare nutzen. Um 1000 MBit/Sek. zu erzielen, werden über jedes Adernpaar 250 MBit/Sek. übertragen. Der Standard erfährt eine immer breitere Akzeptanz und wird in Neuinstallationen eingesetzt.

15.5.8 1000 BASE-SX

Der Standard 1000 BASE-SX nach IEEE802.z gibt an, dass es sich um ein Gigabit Ethernet mit 1000 MBit/Sek. Basisbandübertragung über Short Wavelength handelt. Das heisst, es wird mittels einer Wellenlänge von 850 nm gearbeitet, mit der je nach Glasfaserkabel Entfernungen von maximal 275 m bei 62,5/125 Micron Multimodefasern und maximal 550 m bei 50/125 Micron Multimodefasern zu überbrücken sind. Der Standard verwendet eine Punkt-zu-Punkt-Verbindung, also die Nutzung des CSMA/CD. Als Konnektortyp werden in der Regel SC-Konnektoren verwendet, aber auch MT-RJ- oder VF-45-Konnektoren sind im Einsatz. Der Standard erfährt neben seinem Kupfer-Pendant breite Akzeptanz und kommt in LWL-Neuinstallationen zum Einsatz.

15.5.9 1000 BASE-LX

Der Standard 1000 BASE-LX nach IEEE802.3z gibt an, dass es sich um ein Gigabit Ethernet mit 1000 MBit/Sek. Basisbandübertragung über Long Wavelength handelt. Das heisst, es wird mittels einer Wellenlänge von 1300 nm gearbeitet, mit der je nach Glasfaserkabel Entfernungen von maximal 550 m bei 62,5/125 bzw. 50/125 Micron Multimodefasern und maximal 5000 m bei 9/125 Micron Singlemodefasern zu überbrücken sind. Es können auch grössere Entfernungen überbrückt werden, allerdings handelt es sich hierbei in der Regel um herstellerspezifische Lösungen, die nur untereinander kompatibel sind. Der Standard verwendet eine Punkt-zu-Punkt-Verbindung ohne die Nutzung des CSMA/CD. Als Konnektortyp werden in der Regel SC-Konnektoren verwendet. Neben 1000 BASE-SX ist 1000 BASE-LX eine Lösung im LWL-Primär-Bereich der strukturierten Verkabelung und findet in solchen Umgebungen Akzeptanz.

15.5.10 10-Gigabit Ethernet

Im Jahr 2002 wurde Standard IEEE 802.3ae verabschiedet, auch 10-Gigabit-Ethernet, kurz 10GE oder 10GBaseX genannnt. Dieser Ethernet-Standard nutzte erstmalig ausschließlich optische Verbindungsmedien. Im einzelnen sind dies folgende spezifizierte Varianten: 10GBaseSR, 10GBaseLR, 10GBaseER, 10GBaseLX4, 10GBaseSW, 10GBaseLW und 10GBaseEW. Die Buchstaben S, L und E geben dabei Aufschluss über die verwendete Wellenlänge. S (850 nm Wellenlänge), L (1.310 nm Wellenlänge) und E (1.550 nm Wellenlänge).

	10GBase-X LAN 8B/10B	10GBase-R LAN 64B/66B	10GBase-W WAN SONET
Short 850 nm		10GBase-SR	10GBase-SW
Long 1.310 nm	10GBase-LX4	10GBase-LR	10GBase-LW 10GBase-LW4
Extra Long 1.550 nm		10GBase-ER	10GBase-EW

Die verwendeten Wellenlängen geben in Verbindung mit den verwendeten Glasfaserkabeltypen auch gleichzeitig Auskunft über die Distanz, über die eine Datenübertragung möglich ist.

Wellenlänge	LWL-Faser	Bandbreiten-längenprodukt (MHz*km)	Entfernung
850 nm	50 µm	500 MHz	65 m
1310 nm WWDM	62,5 µm	160 MHz	300 m
1310 nm WWDM	9 µm		10.000 m
1310 nm	9 µm		10.000 m
1550 nm	9 µm		40.000 m

10GbE unterstützt auch Sonet/SDH. Es gibt also zwei unterschiedliche physikalische 10GbE-Schnittstellen: eine für den LAN-Bereich mit 10-GBit/Sek. und eine für Weitverkehrsnetze (WAN) mit 9,584640-GBit/Sek., die der Sonet OC-192c/SDH-Stufe VC-4-64c entspricht.
Die Nomenklatur der unterschiedlichen Schnittstellen ergibt sich aus den folgenden Tabellen:

10GBase-LX1 (1.310 nm Seriell LAN)
10GBase-EX (1.550 nm Seriell LAN)
10GBase-LW (1.310 nm Seriell WAN)
10GBase-EX (1.550 nm Seriell WAN)
10GBase-LX4 (1.310 nm WDM LAN)
10GBase-LW4 (1.310 nm WDM WAN)
10GBase-ER (1.550 nm Seriell LAN)
10GBase-EW (1.550 nm Seriell WAN)

Nomenklatur für 10-Gigabit-Ethernet

Version	Klasse	Fenster	Codierung	Typ
10GBase-SR	10GBase-R	850 nm	64B/66B	seriell
10GBase-SW	10GBase-W	850 nm	64B/66B	Sonet/SDH
10GBase-LX4	10GBase-X	1.310 nm	8B/10B	DWDM
10GBase-LW4	10GBase-W	1.310 nm	64B/66B	Sonet/SDH
10GBase-LR	10GBase-R	1.310 nm	64B/66B	seriell
10GBase-LW	10GBase-W	1.310 nm	64B/66B	Sonet/SDH
10GBase-ER	10GBase-R	1.550 nm	64B/66B	seriell
10GBase-EW	10GBase-W	1.550 nm	64B/66B	Sonet/SDH

2004 kamen die Standards IEEE 802.3ai für 10-Gigabit-Ethernet über das IB4X-Kabel dazu mit acht Twinax-Paaren über 15 m (10GBase-CX4) und alternativ für kurze Entfernungen IEEE 802.3ak 10GBase-CX4, eine kupferbasierte Niedrigpreisvariante für kurze Entfernungen.

Im CX4-Standard werden Entwicklungen aus InfiniBand und 10-Gigabit-Ethernet übernommen. So das 10-Gigabit Attachment Unit Interface (XAUI) und das IB4X-Kabel, das auch bei Infiniband Technologien benutzt wird. Einsatzgebiete dieser Verkabelungen ist z. B. der kostengünstige Interconnect zwischen Servern in Rechencentern.

Im Unterschied zu früheren Ethernetstandards sind aber vorhandene Twisted Pair Verkabelungen heute nicht verwendbar.

Aktuell wird am Standard IEEE 802.3an für 10-Gigabit-Ethernet über Twisted Pair Kupferkabel, 10GBase-T gearbeitet. Die Ausdehnung wurde auf 100 m festgelegt. Allerdings steht jetzt schon fest, dass die zumeist installierten Kategorie 5 Kabel diesem Standard nicht genügen werden. Doch selbst mit Kabeln der Kategorie 6 oder 7 erhält man eine kostengünstige Alternative zu teuren Glasfaserkabeln.

15.5.11 Auto Negotiation

Im Rahmen des Fast Ethernet Standards wurde das Auto Negotiation Protocol (ANP) festgelegt, welches zum Einsatz kommt, wenn Geräte wahlweise mit 10 MBit/Sek. oder 100 MBit/Sek. kommunizieren möchten. Das ANP basiert auf dem Nway-Protokoll von National Semiconductor, welches ebenfalls häufig in Beschreibungen zu finden ist. Das Protokoll stellt automatisch die grösstmöglichste Geschwindigkeit ein, die mittels der angeschlossenen Kommunikationspartner zu realisieren ist. Des Weiteren wird die Betriebsart gewählt, also entweder Halbduplex oder Vollduplex, wie im Kapitel Übertragungsverfahren beschrieben. Das ANP ist für 10 BASE-T und 100 BASE-T Komponenten optional. Nicht möglich ist es bei 100 BASE-FX, da aufgrund der unterschiedlichen Wellenlängen bei optischer Übertragung eine Interoperabilität nicht gewährleistet werden könnte. Für 1000 BASE-T ist das ANP vorgeschrieben. Probleme beim ANP können dann entstehen, wenn Stationen auf die gesendeteten Kontrollpakete nicht antworten und somit automatisch der Halbduplex-Betrieb eingestellt wird. Sollte nun der Kommunikationspartner manuell auf Vollduplex eingestellt sein, kann die Verbindung nicht hergestellt werden.

15.6 Sontige LANs

15.6.3 InfiniBand

Überblick

InfiniBand ist eine Verbindungstechnologie, die aus Future I/O und Next Generation I/O hervorgegangen ist. Das Ziel ist einerseits, eine sehr viel höhere effektive Datenübertragungsrate als mit konventionellem Ethernet zu erreichen. Darüber hinaus soll auch eine Ausfallsicherheit erzielt werden, bei der die Verbindungshardware für die Datenintegrität verantwortlich ist. Und das Netzwerk soll leicht zu erweitern sein und dabei hervorragend skalieren.

InfiniBand ist damit prädestiniert für geclusterte Systeme und verteilte Datenbanken. Die erste Spezifikationsversion wurde im Oktober 2000 veröffentlicht. Der Einsatzbereich ist weit gefasst und deckt sowohl externe als auch interne Kanäle ab, so dass InfiniBand nicht nur im Netzwerkverbund sondern auch bei Bussytemen zum Einsatz kommen kann.

Grundlagen

Bei InfiniBand werden serielle Punkt-zu-Punkt Verbindungen aufgebaut. Hierfür sind 2 Leitungspaare notwendig. Der Bus ist bidirektional zu verwenden und kann 2,5-GBit/Sek. brutto übertragen. Aufgrund des Einsatzes einer 8-Bit/10-Bit Codierung zur Verbesserung der Signalqualität werden damit Datenübertragungsraten von 250 MB/Sek. pro Link erreicht.

Die Datenübertragung erfolgt paketorientiert. Ein Datenpaket von 4096-Bit beinhaltet die Adresse und eine Fehlerkorrektur im Header. Ebenso wie bei Fibre Channel sind sowohl Kupfer- als auch Glasfaserkabel definiert, die maximalen Kabellängen betragen 10, respektive 1.000 Meter.

Zur Erhöhung der Datenübertragungsrate sind zwei Verfahren möglich. Einerseits unterstützt InfiniBand neben der Single Data Rate SDR auch die Übertraung mit Double Data Rate DDR sowie Quad Data Rate QDR. Damit kann die Datenübertragungsrate auf 5-GBit/Sek. bzw. 10-GBit/Sek. pro Link gesteigert werden. Zusätzlich ist eine Bündelung von Links möglich. Neben InfiniBand 1x mit 2,5-GBit/Sek. sind InfiniBand 4x mit 10-GBit/Sek. sowie InfiniBand 12x mit 30-GBit/Sek. definiert.

InfiniBand Bandbreiten, brutto/netto

	SDR	DDR	QDR
1 x	2,5/2-GBit/Sek.	5/4-GBit/Sek.	10/8-GBit/Sek.
4 x	10/8-GBit/Sek.	20/16-GBit/Sek.	40/32-GBit/Sek.
12 x	30/24-GBit/Sek.	60/48-GBit/Sek.	120/100-GBit/Sek.

Theoretisch sind also maximal 120-GBit/Sek. mit InfiniBand QDR 12x denkbar. In der Praxis kommt üblicherweise InfiniBand SDR 4x zum Einsatz. Produkte mit SDR 12x sind am Markt verfügbar bzw. Für DDR 4x angekündigt.

Für die reale Datenübertragung ist neben der Bandbreite auch noch die Latenz zu berücksichten. Diese zu Deutsch mit Wartezeit zu übersetzende Größe addiert den Overhead, der jeder einzelnen Übertragung vorausgeht. Der Einfluss der Latenz ist umso bedeutender, je kleiner die durchschnittliche Paketgröße ist. Daher wird in der Praxis vermehrt auch diejenige Paketgröße angegeben, bei der die Hälfte der maximalen Bandbreite erreicht werden kann.

15.6.3.1 Architektur

Die InfiniBand Archtektur besteht aus 4 Hardwareelementen: Dem Host Channel Adapter (HCA), dem Target Channel Adapter (TCA), dem Switch und dem Router.

Ein Host Channel Adapter ist das Interface zwischen einem Server und der InfiniBand-Fabric. Zur Optimierung der Datenübertragung kann er direkt mit dem Prozessor und dem Hauptspeicher kommunizieren. Aufgrund einer ständigen Fehlerkorrektur ist er in der Lage, autonom Übertragungsfehler auszugleichen. Es stehen zwei Remote DMA Verbindungsmodi zur Verfügung, RDMA Write und RDMA Read. In beiden Fällen werden die Daten in den Hauptspeicher eines anderen Knoten übertragen, ohne dass der betroffene Knoten aktiv werden müsste und sein Load Level erhöht würde.

Der Target Channel Adapter ist als erweiterter HCA zu verstehen. Er besitzt einen eigenen I/O Controller, der die Pakete in die Protokolle des angeschlossenen Geräts übersetzt. So können direkt SCSI, Fibre Channel oder Ethernet Targets konnektiert werden, ohne dass ein kompletter Host mit CPU und Speicher notwendig wäre. Projektiert wurde der TCA für Geräte wie Plattensubsysteme oder Backupdevices. InfiniBand unterstützt deshalb die blackorientierte Massenspeicherverwaltung SRP (SCSI RDMA Protocol).

Der Switch regelt im InfiniBand Netzwerk die Punkt-zu-Punkt-Verbindungen gemäß den Anforderungen der angeschlossenen HCAs und TCAs. Über den Local Route Header des einzelnen Datenpakets stellt er das gewünschte Ziel fest und leitet es an die betreffende Komponente weiter.

Ein InfiniBand Router sorgt im Bedarfsfall für den Transport der Datenpakete von dem lokalen Netzwerk zu anderen Subnetzen. Hierfür wertet der Route den Global Route Header aus und übersetzt diese in eine Network Layer Adress nach IPv6. Umgekehrt übersetzt er bei eingehenden Paketen deren Header und baut den betreffenden Local Adress Header auf.

Innerhalb eines InfiniBand Netzwerks sind immer jeweils einzelnen Links miteinander verbunden. Daher kann ein HCA mit IB 4x mit 4 aggregierten Links problemlos mit mehreren Targets gleichzeitig verbunden sein. Auf diese Weise kann auf Hardwareebene für Redundanz gesorgt werden. Die Zuverlässigkeit der Verbindung kann durch eine Reliable Connection erhöht werden, bei der die Hardware für die Datenintegrität zuständig gemacht wird.

15.6.3.2 Randbedingungen

InfiniBand Adapter sind mit PCI-X, PCI Express und HTX Interface verfügbar. Der Einsatz von PCI-X limitiert ein InfiniBand 4x Interface jedoch und erlaubt nur ca. 80 % der maximalen Bandbreite.

Die Treiberunterstützung ist vielfältig. Support gibt es für Windows®, Linux und Mac OS. Seit der Kernelversion 2.6.11 ist InfiniBand nativ in Linux verankert.

Als Message Passing Interface MPI stehen die MVAPICH von der Ohio State University OSU sowie MPICH-VMI vom National Center for Supercomputing Applications NCSA zur Verfügung.

InfiniPath

Eine Sonderstellung nimmt der InfiniBand Adapter der Pathscale ein. Die unter dem Markennamen InfiniPath angebotenen Karten umgehen den klassischen PCI-Bus und verbinden das Netzwerk über den HyperTransport-Tunnel mit dem Server. Dies ist möglich, da in der typischen Dual und Quad Opteron™ Architektur mindestens ein HyperTransport-Anschluss eines AMD Opterons unbenutzt bleibt. Über diese HTX (HyperTransport-Exchange) Schnittstelle sind die Pakete deutlich latenzärmer auszutauschen. Die Protokollverarbeitung übernimmt in diesem Konzept der Prozessor.

16. Metropolitan Area Networks – MANs

17. Wide Area Networks – WANs

18. LAN Core-Lösungen

19. Eingabegeräte

20. Datenkommunikation

21. Terminals

22. Ausgabegeräte

23. Multimedia

24. Unterbrechnungsfreie Stromversorgungen

Die Kapitel 16 – 24 finden Sie online unter
www.transtec.de
www.transtec.at
www.transtec.ch