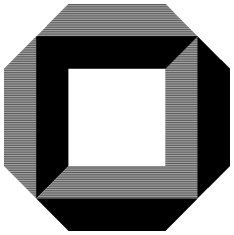


Universität Karlsruhe

Fakultät für Informatik

Informatik-Rechnerabteilung
Institut für Betriebs- und Dialogsysteme

Seminar
Netzwerkmanagement

Wintersemester 1992/93

Betreuer und Herausgeber:

Dipl.-Inform. G. Schreiner

Prof. Dr. W. Zorn

Einleitung

Die Datenkommunikation ist in den letzten Jahren wesentlicher funktionaler Bestandteil einer Datenverarbeitungsanlage geworden, so daß fast jedes Rechensystemen in ein Netzwerk eingebunden wird. Während die theoretischen Grundlagen einer DV-Anlage in Bezug auf ihr Inneres in Forschungsbereichen wie Betriebssysteme u.ä. ergiebig untersucht wurde, blieb der Bereich der Netzverwaltung längere Zeit unbetrachtet.

Aufgrund der geringen Teilnehmerzahl wurden innerhalb dieses Seminars drei spezielle Aspekte des Netzwerkmanagements in den Blick genommen: Datenstromüberwachung, Router-Konfigurationen und Ansätze zu verteiltem Netzwerkmanagement.

Der Herausgeber

Die verspätete Veröffentlichung dieser Zusammenfassung beruhte zum Einen auf dem Weggang einer der beteiligten Mitarbeiter und zum Anderem auf unzureichenden Dokumentationsrichtlinien, die das Zusammenbinden in eine Berichtsform erschwerten. Die Nachfrage nach insbesondere einem der behandelten Themen bewirkte den Ausschlag zur Veröffentlichung als interner Bericht.

Inhaltsverzeichnis

Abbildungsverzeichnis

Kapitel 1

Monitoring von Datenströmen (RMON)

Carsten Voigt

1.1 Einleitung

Im Laufe der letzten Jahre sind viele der klassischen innerbetrieblichen Netze, auch Local Area Network (LAN) genannt, entstanden, wobei die Datenströme der LANs mit einer einzelnen Managementstation kontrollierbar waren. Doch die isolierten LANs wuchsen in größere, meist öffentliche zusammenhängende Netzwerke MANs und WANs hinein, so daß sich nun die Frage nach deren Kontrolle stellte.

1.1.1 Der SNMP Standard

In der Internet-Welt, unter Verwendung des TCP/IP-Protokolls, hat sich mittlerweile ein Standard durchgesetzt, um Daten in heterogenen Netzwerken zu sammeln und zu analysieren. Dieser Standard ist das Simple Network Management Protocol (SNMP). Durch diesen Standard ist nun das Management unabhängig von den einzelnen Herstellern möglich, sofern diese das SNMP benutzen. Somit besteht die Möglichkeit auf alle Beteiligten des Netzwerkes einzugehen.

1.1.2 Definitionen

Bei SNMP und später bei RMON werden folgende Definitionen verwendet:

RFC-1155, welches das SMI definiert. Mechanismen zur Beschreibung und Benennung von Objekten; RFC-1212 ist eine Erweiterung.

RFC-1156, welches die MIB-I (Management Information Base) definiert: die Menge der managed objects definiert durch ASN.1. RFC-1213 ist eine Erweiterung aufgrund der Erfahrungen mit MIB-I.

RFC-1157, welches das SNMP (Simple Network Management Protocol) definiert: Beschreibung des Netzwerkzugriffes.

RFC-1271, welches die Remote Network Monitoring Management Information Base (RMON) definiert: Erweiterung des SNMP.

1.1.3 Ziele des Remote Network Monitoring

Die Stationen in den Segmenten sammeln die Netzwerkdaten und senden diese zur zentralen SNMP-Managementstation. Selbst in einer einzelnen Firma sind auf einer zentralen Station unterschiedliche Tools installiert, entsprechend ihrer Funktionen und Aufgaben.

Eine Datenübertragung kann nur sinnvoll sein, wenn die beteiligten Stationen und deren Tools sich "verstehen", d.h. ein Standardformat benutzen. Mit RMON MIB können einheitliche Schnittstellen zu anderen Tools definiert werden (Graphische Ausgabe, Datenbanken, Expertensysteme).

Da in den Segmenten die Netzwerkdaten gesammelt werden, wird ein Großteil der Auswertung in die Segmente verlegt.

$\Rightarrow$ geringere Netzbelastung

$\Rightarrow$ geringere Auslastung der zentralen Station

1.2 Möglichkeiten von RMON

Die Remote Network Monitoring Management Information Base (RMON-MIB) realisiert einen Standard, um Systemen, die SNMP benutzen, die Möglichkeit zu geben, die Netzbelastung und die Datenströme auszuwerten. Hierzu hat die Internet Engineering Task Force (IETF) beschrieben,

wie die Daten gesammelt und gespeichert werden müßten, um sie an andere Managementstationen weiterzugeben. Die RMON MIB bringt einigen Nutzen, dessen wichtigster es ist, daß Stationen in verschiedenen Segmenten mit einer oder mehreren zentralen SNMP-Managementstationen zusammenarbeiten können.

Die Zusammenarbeit aller Stationen sieht folgendermaßen aus: Eine Managementstation kann aktiv werden, um Auswertungen vorzunehmen. Dazu fordert sie durch Aufruf einer Funktion die nötigen Daten über das Netz bei dem Agenten an. Erst dadurch reagiert der Agent und wird aktiv. Er behält diese Aktivität, bis entweder die Funktion abgeschlossen ist oder von der Managementstation unterbrochen wird. Eine Ausnahme ist, wenn im Netzwerk Fehler oder Probleme auftreten, die der Agent aufgrund seiner Konfigurierung erkennt. Dann wird der Agent selbständig aktiv.

1.2.1 Offline-Operationen

Es gibt Situationen, bei denen ein permanenter Kontakt zwischen den Monitoren nicht möglich ist, aufgrund von Fehlern im Netz oder der zu hohen Kosten z.B. in einem WAN. Der RMON-Agent kann so konfiguriert werden, daß er selbständig Auswertung und Speicherung der statistischen Daten off-line ausführen kann, eben genau dann, wenn die Kommunikation nicht möglich oder nur sehr ineffizient ist. Diese Daten werden dann später zur zentralen Station gesendet.

1.2.2 Vorausschauendes Monitoring

Es ist hilfreich, die Auswertungen und die Auslastungen des Netzwerkes und der beteiligten Stationen fortlaufend zu protokollieren. Die historischen Daten und Fehler werden abgespeichert, um sie bei Bedarf erneut anzuschauen und daraus die Entwicklungen der Zukunft abzuleiten. Damit ist allerdings auch ein großer Speicheraufwand notwendig, der sich bei geringer Speicherkapazität der Station nachteilig auswirken kann. Ohne diesen Speicheraufwand wäre ein Netzwerkmanagement nur unzureichend möglich, da keine Entwicklungen der Netzwerke, sondern nur Zeitpunkte beobachtet werden können.

1.2.3 Das Problem der Fehlererkennung und des Reports

Der RMON Agent kann so konfiguriert werden, daß er die Netzwerkprobleme und -fehlermöglichkeiten kennt und diese fortlaufend kontrolliert. Er informiert daraufhin die zentrale Station. Wird ein Fehler erkannt, informiert der Agent daraufhin die zentrale Station in Form eines Reports. Dieser Report wird je nach Priorität sofort oder erst auf Abruf zur zentralen Station gesendet (siehe Alarme).

1.2.4 Hinzufügen der Auswertungsergebnisse

Neben dem einfachen Sammeln der statistischen Daten können Auswertungsergebnisse diesen Daten hinzugefügt werden, indem z.B. Fehler besonders hervorgehoben werden und es der zentralen Station leichter gemacht wird, denn die zentrale Station braucht die statistischen Daten nicht noch einmal auswerten, dadurch wird ihr Arbeitsaufwand verringert und im Fehlerfall kann die zentrale Station schneller darauf reagieren. Hierbei muß man beachten, daß die zentrale Station mehrere Agenten verwalten muß und schon dadurch einen erhöhten Aufwand hat.

1.2.5 Abstimmung zwischen mehreren Managern

Eine Organisation kann durchaus mehrere zentrale Managementstationen haben, entsprechend ihrer speziellen Aufgaben. Der Entwurf des Remote Network Monitoring stimmt auch mehrere zentrale Managementstationen aufeinander ab (siehe Kontrolle der Managementstationen).

1.3 Funktionen der RMON MIB

In dem RMON-MIB-Standard sind neun verschiedene Funktionen definiert, die ein Auswerten der Netzauslastung ermöglichen.

1.3.1 Die statistischen Daten der Netzauslastung

Diese Gruppe umfaßt die Daten, welche von den Agenten in ihren Segmenten gesammelt wurden. Sie können dem Operator auf dem Bildschirm ausgegeben werden, damit Aussagen über die Funktionstüchtigkeit des Netzes gemacht werden können. Dies umfaßt im wesentlichen die CRCs (cyclical

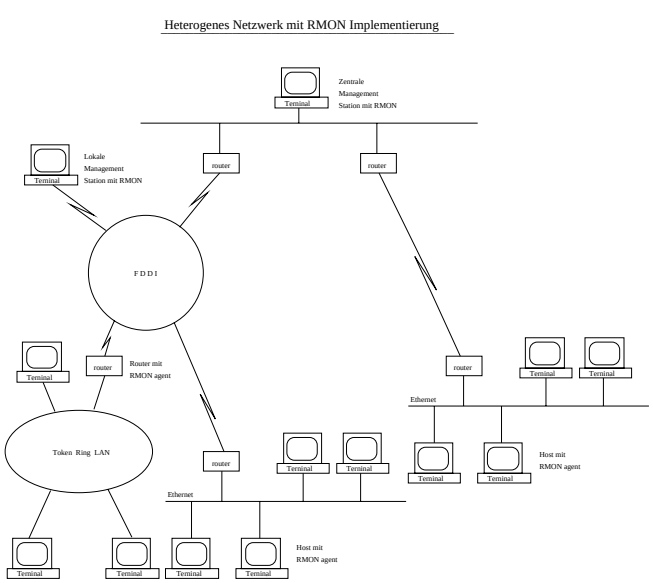


Abbildung 1.1: RMON im heterogen Netzwerk

redundancy checks), Kollisionen und Verfälschungen eines Ethernets. Die Daten sind enthalten in der *etherStatsTable*. Eine spätere Erweiterung des RMON wird auch den Token-Ring und FDDI berücksichtigen.

1.3.2 Die historischen Daten

In der Gruppe der historischen Daten werden in bestimmten Intervallen die Bedingungen in einem Netzwerk zu einem gegebenen Zeitpunkt gespeichert, um sie u.a. von der zentralen Managementstation zu einem späteren Zeitpunkt erneut anzuschauen und zu analysieren. Die Intervalle sollten immer über den gleichen Zeitraum gewählt werden, um eine benutzerfreundliche Darstellung zu ermöglichen, um z.B. einen Vergleich zwischen zwei verschiedenen Tagen zu erleichtern, bei einem Intervall von einem Tag. Diese historischen Daten werden aus den statistischen Daten gewonnen. Ein Agent sollte sinnvollerweise nur ein Netzwerk betrachten, an dem er angeschlossen ist, da fremde Daten vom Koppelbaustein (Bridge oder Router) dieses Segmentes gefiltert werden. Er speichert in der *historyControlTable* Einträge zur Konfiguration, d.h. Schnittstellen, Abfrageintervalle und andere Parameter. Für kurze Polling Perioden werden 30 Sekunden und für lange 30 Minuten vorgeschlagen. Dadurch werden die Statistiken genormt, die sich daraus gewinnen lassen und bieten eine bessere Vergleichsmöglichkeit. Die Daten der historischen Gruppe sind spezifisch für das Medium eines Ethernet in der *etherHistoryTable* gespeichert, parallel zu den Einträgen in der *historyControlTable*.

1.3.3 Alarme

In dieser Gruppe werden verschiedene Schwellwerte angegeben, die in periodischen Intervallen geprüft werden, um bei Überschreitung der Schwellwerte automatisch einen Alarm an die zentrale Managementstation zu senden. Hierbei wird die Implementierung der Ereignisgruppe vorausgesetzt, die die Ereignisse, darunter einen Alarm generieren kann. Beim Senden eines Alarms werden Prioritäten vergeben, die je nach Dringlichkeit zu einem Vermerk in einem Logfile oder zur Alarmierung des Operators führt. Diese Gruppe steht in der *alarmTable*. Die Generierung eines Ereignisses obliegt der Ereignisgruppe. Für die Prioritäten können Bandbreiten angegeben werden. Betrachten wir hierzu ein Beispiel:

Wählen wir das Beispiel der Netzauslastung (**alarmSampleType** = "Netzauslastung"):
Angenommen eine Netzauslastung von 15% ist bei seltenem Auftreten wenig bedenklich, aber wenn dies häufiger vorkommt, muß darauf reagiert werden. Zudem erfordern 40% eine sofortige Reaktion, da kaum noch Daten übertragen werden können. Dann können mit RMON zwischen 15% und 40% ein Alarm mit geringer und über 40% mit hoher Priorität generiert werden, bei folgender Variablenbelegung:

alarmRisingThreshold = 15% ⇒
Alarm mit Eintrag in Logfile
alarmRisingThreshold = 40% ⇒
SNMP-Trap

alarmFallingThreshold = 10% ⇒
Bereitschaft zum Senden eines neuen Alarms

Wie die Netzauslastung definiert ist, bestimmt der Manager des Netzwerkes. RMON vergleicht nur die ihm vorgegebenen Parameter. Die **alarmFallingThreshold** ist notwendig, damit es nicht in jeder Periode zu einem Alarm kommt, d.h. bewegt sich der Wert dauerhaft über 10%, nachdem es einen Alarm gegeben hat, dann besteht keine Bereitschaft mehr, einen Alarm zu senden, erst wieder bei Unterschreiten der 10%.

1.3.4 Hosts

Der RMON Agent führt eine Liste der ihm bekannten angeschlossenen Rechner. Wird ein neuer Rechner am Segment angeschlossen, so aktualisiert der Agent seine Liste, indem er die Hostadresse den Datenpaketen des Netzwerkes entnimmt. In der Liste, die aus der *hostControlTable*, *hostTable* und der *hostTime Table* besteht, stehen zudem die statistischen Angaben zu den am Netzwerk aktiven Rechner.

1.3.5 Die größten N Hosts

Die Gruppe der "host top N" erwartet die Implementierung der Host Gruppe. Damit läßt sich eine Liste der kritischsten Hosts erstellen. Die Managementstation setzt die erforderlichen Parameter (hier in Klammern geschrieben). Zum Beispiel steht in dieser Gruppe eine Liste der 10 (**hostTopNRequestedSize** = 10) größten Knoten, die an einem Tag (**hostTopNTimeRemaining** = 86400 Sekunden) die meisten Daten übertragen haben. Diese Gruppe besteht aus der *hostTopNControlTable* und der *hostTopNTable*, wobei die Letztere erst am Ende des Beobachtungszeitraumes erstellt wird.

1.3.6 Der Datenfluß im Netzwerk

Diese Matrixgruppe beinhaltet die Kommunikationsdaten von Adressen auf MAC-Ebene (Media Access Control = Schicht 2). Aufgrund der in diesen Tabellen verzeichneten Übertragungsraten ist sofort der Kommunikationsaufwand zwischen den Hosts ersichtlich. Diese Daten stehen in der *matrixSDTable* und der *matrixDSTable*. Parallel dazu werden in der *matrixControlTable* der Index, Meßzeitraum usw. festgehalten. Zum Beispiel wird die Datenübertragung zu einem ftp-server vielfach genutzt und in der Matrix-Gruppe angezeigt. Der Netzwerkmanager kann entsprechend darauf reagieren.

1.3.7 Die Filterung der Pakete

Die "Packet Filter Group" erlaubt die Übertragung von Paketen, mit allen unterschiedlichen spezifischen Formaten, um Daten oder Ereignisse aufzufangen oder zu übertragen. Die Formate müssen in der *filterTable* bekannt gemacht werden. Die Kanäle entsprechen logischen Datenströmen. Mit zunehmender Anzahl der Filter, bzw. der aktiven Filter wird sich die Belastung des Agenten steigern, denn er wird versuchen, ein ankommendes Paket in entsprechend viele Kanäle zu schicken. Man sollte sich bei der Einrichtung der Filter auf die Formate beschränken, die in den Segmenten verwendet werden. Um die Kommunikation zwischen der Managementstation und dem Agenten zu ermöglichen, müssen beide ihre passenden Filter aktiv halten. Diese Gruppe steht in der *filterTable* und der *channelTable*.

1.3.8 Auffangen der Pakete

In dieser Gruppe werden die Pakete aufgefangen, die den Filtermatch der Filtergruppe bestanden haben. Die *bufferControlTable* kontrolliert die aufgefangenen Pakete aus einem Kanal (Kennummer des Kanals, Größe des Datenstromes usw.). Diese Pakete werden in die *captureBufferTable* eingetragen. Sie enthält eine Liste der aufgefangenen Pakete aus einem Kanal, parallel zu den Einträgen in der *bufferControlEntry*. Diese Gruppe benötigt die Filterung der Pakete.

1.3.9 Die Ereignis Gruppe

Es werden alle Auswertungsergebnisse und Daten in der *eventTable* und der *logTable* festgehalten, die der RMON Agent an eine oder mehrere zentrale Managementstation gesendet hat. Zudem steht dem RMON Agenten die Möglichkeit offen, alle Ereignisse zu speichern. Es besteht die Möglichkeit auf ein Ereignis in folgender Weise zu reagieren:

1. Ereignis ignorieren,
2. Eintrag ins Logfile,
3. Senden eines SNMP-Trap, oder schließlich
4. SNMP-Trap und Logfile-Eintrag.

In der *eventCommunity* steht die Senke (z.B. Management Station), zu der der Trap gesendet werden soll. Dabei sind folgende Traps definiert :

- *rising alarm*,
- *falling alarm*,
- *packet match*.

1.4 Die Kontrolle der RMON Stationen

Aufgrund der komplexen Struktur aller möglichen Funktionen, benötigen diese in den meisten Fällen eine Konfiguration durch den Benutzer. Solche Operationen funktionieren erst dann, wenn deren Parameter gesetzt wurden zum Sammeln der Daten. Im weiteren Verlauf werden die Parameter durch die Auswertungsergebnisse verändert. In den oben beschriebenen Funktionen gibt es die Menge der Kontrollparameter im ReadWrite–Mode einerseits und andererseits Readonly–Mode diejenige, wo die Daten und Ergebnisse gespeichert werden. An diesem Punkt stellt sich die Frage, wie eine gute Kontrolle gewährleistet wird, wenn viele Managementstationen im Netz existieren. Eine Funktion beansprucht meist Ressourcen der Auswertung und Speicherung. Nun müssen diese Ressourcen auf mehrere Managementstationen verteilt werden.

1.4.1 Teilung der Ressourcen

Wenn viele Managementstationen eine RMON-Funktion benutzen möchten und diese nur auf eine beschränkte Menge der Ressourcen zurückgreifen kann, ist eine Teilung und Verwaltung notwendig.

Dies beinhaltet zum Beispiel folgende Konfliktsituationen :

1. Zwei Managementstationen wollen gleichzeitig eine Ressource belegen, welches zusammen die Kapazität des Gerätes übersteigt (CPU, Datenleitung.)
2. Über einen längeren Zeitraum werden die Ressourcen von einer Managementstation blockiert.
3. Blockierte Ressourcen werden nicht mehr freigegeben, z.B. durch einen Absturz der Managementstation oder Unterbrechung der Datenleitung. Diese Ressourcen können nicht mehr von Anderen benutzt werden.

Durch die RMON-MIB werden Mechanismen zur Verfügung gestellt, mit denen diese Konflikte vermieden werden, oder zumindest deren Behebung unterstützt wird. Hierzu bekommen die Funktionen einen Label mit dem der Besitzer identifiziert werden kann. Dieser Label wird gesetzt, um für die folgenden Möglichkeiten zu sorgen :

1. Eine Managementstation "erinnert" sich an seine Ressource und entscheidet über deren weiteren Gebrauch.
2. Ein Netzwerk-Operator findet den Besitzer zu einer belegten Ressource und kann diese, falls notwendig, wieder freigeben.
3. Ab der Initialisierung wird von der Managementstation jede Ressource protokolliert, die bisher belegt wurde und kann zudem über deren weitere Verwendung entscheiden.

Es wäre wünschenswert, wenn die Managementstation ein einheitliches Format im Label der Ressource hinterläßt, zum Beispiel,

> IP-Adresse, Managementstationsname, Netzwerkmanagementname, Ort oder Telefonnummer.

Mit diesen Informationen kann eine Ressource dem Benutzer besser zur Verfügung gestellt werden. Gelegentlich können die Geräte selbständig aktiv werden, u.a. durch ein "Set up". Daraufhin bekommt der Label den Vermerk "Monitor". Wird eine Funktion durch eine Managementstation angefordert, sollte erst in der Kontrolltabelle nachgeschaut werden, welche Ressourcen zur Verfügung stehen und ob deren Funktionen von mehreren Managementstationen gleichzeitig verwendet wird. Dadurch muß verhindert werden, daß eine Ressource bei gleichzeitigem Gebrauch geändert oder gelöscht wird.

1.4.2 Konfigurierung der Funktionen

Um Funktionen neu zu konfigurieren oder zu erweitern, werden in den Funktionstabellen neue Zeilen hinzugefügt. Hierbei löst die Managementstation die Operation aus und der Agent prüft die neuen Parameter mit dem ihm vorgegebenen Einschränkungen der MIB. Hier gibt es zwei Möglichkeiten:

Die Managementstation sendet die Parameter ...

- einzeln, oder
- im Block.

Tritt bei der Prüfung ein Fehler auf, so sendet der Agent eine Fehlermeldung. Bei einer einzelnen Versendung der Parameter wird von der Managementstation sofort der fehlerhafte Parameter erkannt, im Gegensatz zur Blockversendung, wo der ganze Block wiederholt werden muß. Wenn die Parameter nur gemeinsam eingetragen werden dürfen, ist eine Blockversendung notwendig, um Inkonsistenzen zu vermeiden. Wollen mehrere Managementstationen gleichzeitig einen Tabelleneintrag ändern, so setzt die Erste ein Flag und die Nachfolgenden bekommen eine Fehlermeldung.

1.5 Aussicht in die Zukunft

Da die RMON-MIB in RFC1271 nur für das Ethernet definiert wurde, ist es notwendig diese auch für Token-Ring und FDDI zu entwickeln, denn die meisten großen Netzwerke beinhalten Token-Ring oder FDDI.

1.6 Literaturverzeichnis

[Wal91] [WNH92]

Kapitel 2

Router, Brücken: Funktion und Management

Alexander Hofmann

2.1 Einführung

Die maximale physikalische Ausdehnung eines LANs ist durch verschiedene Einschränkungen, wie Güte des Mediums, Anzahl Verstärker usw. bestimmt. Ferner besteht oft die Notwendigkeit des Zusammenschlusses mehrerer LANs, welche historisch bedingt von unterschiedlicher Technologie sind. Zum Beispiel können in verschiedenen Institutionen LANs vom Typ Ethernet und Token-Bus vorliegen, welche es zu koppeln gilt. Die Schaffung eines logischen Gesamt-LANs und die Überbrückung großer Distanzen führt zum Begriff des **Internetworkings**. Es existieren 4 Internetworking Gerätetypen zur Realisierung der oben genannten Aufgaben, die zusammenfassend in Bild 2.1 dargestellt sind:

a) ein Repeater (z.B. Ethernet-Ethernet),

b) eine Brücke (z.B. Ethernet-Token-Bus),

c) ein Router (z.B. Token-Bus-X.25),

d) ein Gateway.

2.2 Der Repeater

Der **Repeater** arbeitet bitweise, meist bidirektional auf OSI Schicht 1. Repeater sind nicht intelligent, d.h. sie besitzen keine Softwaresteuerung und daher erfolgt ein blindes Kopieren ankommender Daten. Ein Repeater vergrößert die Reichweite von physikalischen Signalen und muß nicht konfiguriert werden. Der Nachteil des Repeaters besteht darin, daß eine Verbindung nur möglich ist, wenn identische LAN vorhanden sind.

2.3 Das Gateway

Wenn die Struktur der zu verbindenden LANs ab Schicht 3 (bzw. aufwärts) unterschiedlich ist, kann man eine Kopplung nur durch Gateways vornehmen. Da es maximal 7 Schichten von jedem LAN (in Summe also 14) besitzt, ist das **Gateway** relativ langsam und aufgrund seiner Komplexität eigentlich eine Station.

2.4 Die Brücke

2.4.1 Allgemeines

Eine einfache Brücke dient zur Verbindung zweier LANs, so daß alle Ressourcen auf allen angeschlossenen Netzwerken zur Verfügung stehen, ist der OSI Schicht 2 zuzuordnen und arbeitet paketweise. Eine längenmäßige Erweiterung eines LANs durch eine Brücke ist unbegrenzt. Sie ist protokolltransparent, d.h. der Inhalt der Pakete spielt keine Rolle. Ihre Leistung wird gemessen in der Weiterbeförderungsrate und der Filterrate. Als Gründe für den Einsatz einer Brücke wären etwa zu nennen, die Aufteilung eines LANs ist erforderlich, um die Belastung zu verteilen, die Entfernung der LANs ist groß, Sicherheitsanforderungen (Filterung von Paketen) usw.

2.4.2 Funktionsprinzip

Wie in Abbildung 2.2 verdeutlicht, will Host A ein Paket versenden. Das Paket fällt in die LLC (Logical Link Control) Teilschicht hinab und erhält einen LLC Nachrichtenkopf. Es erfolgt ein Weiterfallen in die MAC (Medium Access Control) Teilschicht und ein Anhängen eines 802.3 Nachrichtenkopfes. Jetzt

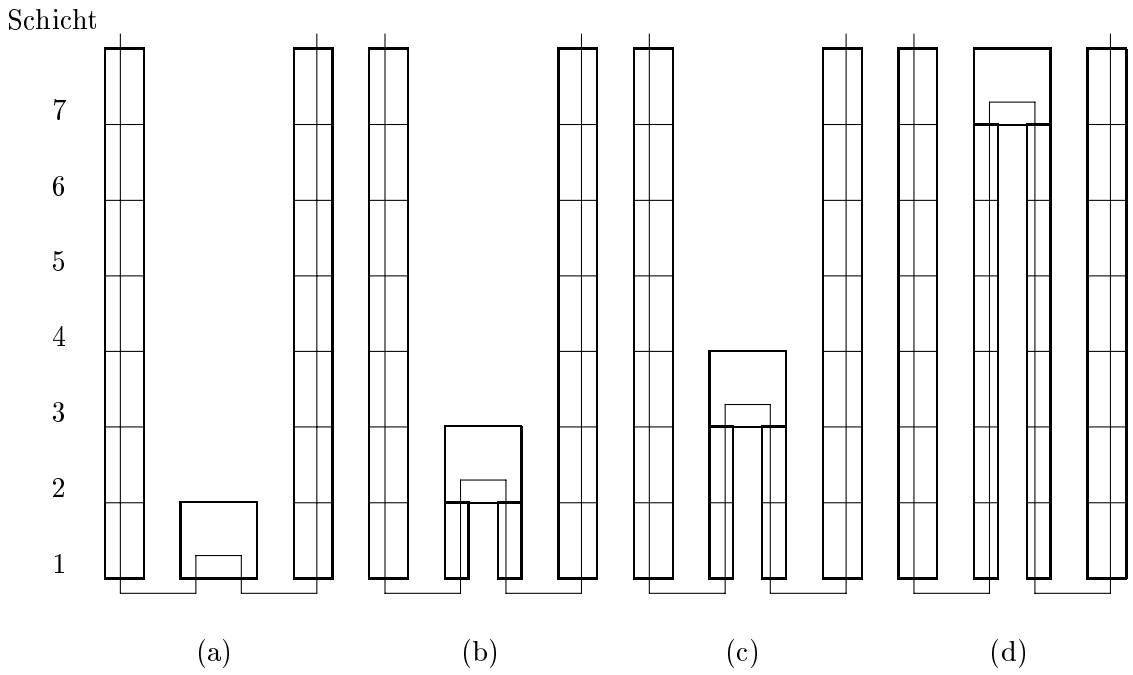


Abbildung 2.1: Internetworking Geräte

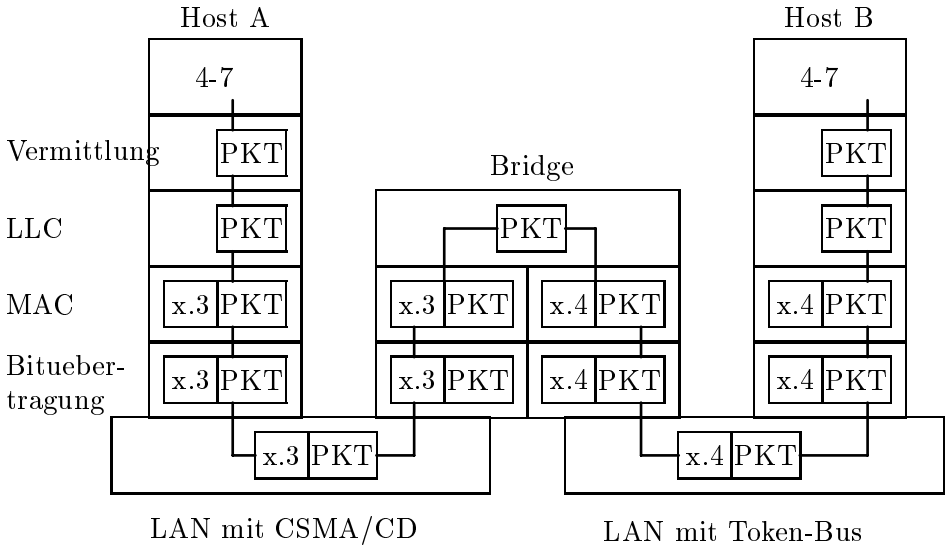


Abbildung 2.2: Funktionsweise einer LAN-Brücke von 802.3 nach 802.4 (x=802)

geht das Paket über die Bitübertragungsschicht auf das Kabel und kommt in der MAC Teilschicht der Brücke an. Dort wird der 802.3 Nachrichtenkopf entfernt und das Paket wird an die LLC der Brücke übergeben. Nun erfolgt ein Hinunterfallen des Pakets in die MAC Teilschicht der Brücke und ein Anhängen eines 802.4 Nachrichtenkopfes usw. Grundlegend für die Funktionsweise der Brücke ist, daß wenn sie k unterschiedliche LANs verbindet, muß sie auch k verschiedene MAC und Bitübertragungsschichten besitzen. Bei der Kopplung müssen jedoch Probleme wie unterschiedliche Rahmenformate bzw. Rahmengrössen, verschiedene Übertragungsraten usw. beachtet werden.

2.4.3 Formen von Brücken

Transparente Brücken

Das Anliegen bei der Entwicklung dieses Brückentyps lag darin, völlige Kompatibilität zu erreichen und eine Installation überflüssig zu machen. Die Brücke besitzt eine Tabelle (meist Hash-Tabelle), worin alle Ziel-LANs enthalten sind. Nach dem Einbau der Brücke ist diese Tabelle leer und sie muß nun alle Ziele im Netzwerk lernen. Dafür steht der Flutalgorithmus (beruht auf Weitergeben des Pakets an alle angeschlossenen LANs bei Nichtkennen des Ziels) oder der Algorithmus des spannenden Baumes zur Verfügung. Hierbei überträgt jede Brücke aller paar Sekunden ihre Kennung. Aus der so entstehenden Menge von Kennungen können alle Brücken im Netz alle Ziele lernen.

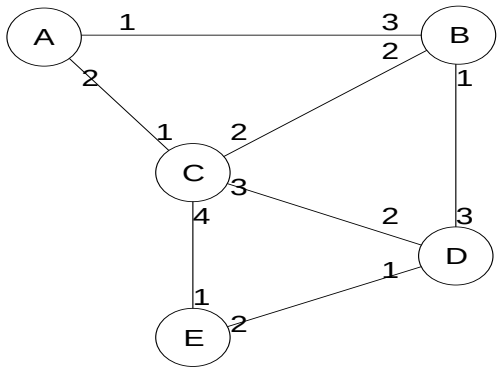
Brücken mit Quellenleitwegbestimmung

Voraussetzung hierbei ist, daß jeder Sender eines Rahmens weiß, ob das Ziel in seinem eigenen LAN liegt, oder nicht und den genauen Leitweg zum Ziel kennt. Diese Leitwege werden mit sog. Rundsenderahmen ermittelt, welche periodisch durch das Netz geschickt werden.

Vergleich der Brückenformen

Der Vorteile einer transparenten Brücke liegen darin, daß keine Installation notwendig ist, daß ein Selbstlernen der Zieladressen erfolgt und daß eine volle Kompatibilität zu allen 802.x LANs gegeben ist. Ein Nachteil wäre hierbei etwa, daß eine Dopplung einer Identifikationsnummer schwer erkennbar ist. Im Gegensatz zur transparenten Brücke nutzt die Brücke mit Quellenleitwegbestimmung den optimalsten Leitweg und es kann eine Lastenaufteilung durch eine Parallelschaltung erzielt werden. Als Nachteile wären hier etwa die Nichttransparenz und die Inkompatibilität zu nennen.

2.5 Der Router



Zielknoten	zu wählender Ausgang
A	3
B	3
C	2
D	–
E	1

Abbildung 2.3: Topologie eines Netzes und Routing-Tabelle für Knoten “D”

2.5.1 Allgemeines

Der Router ist wie die Brücke ein bidirektionaler Empfänger-Sender jedoch der OSI Schicht 3 zuzuordnen. Es erfolgt eine Interpretation der Adreßangaben innerhalb des Pakets zu Routingzwecken.

2.5.2 Funktionsprinzip

Genauer wird der Nachrichtenkopf von jedem Paket interpretiert, herausgenommen und aus den Routingtabellen (Abb.3) ein völlig neuer Schicht 2 Nachrichtenkopf zusammengebaut und dem Paket hinzugefügt. Jetzt erfolgt ein Weiterschicken des Pakets in Richtung Ziel-LAN.

Der ankommende Schicht 2 Nachrichtenkopf kann sich hierbei völlig von dem weitergeschickten unterscheiden, je nach dem, welche Protokolle auf Quell- und Ziel-LAN vorliegen. Ein durchschnittlicher Router ist in der Lage, 5 - 20 verschiedene Protokolle zu erkennen. Zur Verdeutlichung der Funktionsweise eines Routers möchte ich hier ein kleines Beispiel aus dem täglichen Leben anführen.

Man stelle sich vor, A möchte einen Brief an B schicken. Nach dem Schreiben bringt A den Brief

zur Post und der Postbeamte bestimmt die beste Route für den Brief, ohne jedoch den Inhalt zu lesen. Daraufhin legt der Postbeamte den Brief in einem speziellen Postsack ab und schickt ihn zum nächsten Postamt usw., bis der Brief in seinem Bestimmungspostamt ankommt. Der Brief wird ausgeliefert und *B* erhält ihn, ohne die genaue Route des Briefes zu kennen. Ebenso funktionieren Router, welche man in diesem Beispiel mit den Postämtern gleichsetzen muß.

Woher weiß nun der Router, zu welchem Router er das Paket weiterleiten soll ? Dies führt uns zur Untersuchung der Leitwegbestimmungsalgorithmen von Routern.

2.5.3 Leitwegbestimmungsalgorithmen

Leitwegbestimmungsalgorithmen (LBA) für Router teilen sich in adaptive (es erfolgt eine ständige Anpassung an Änderungen der Netztopologie) und in nicht adaptive (alle Leitwege werden beim Hochfahren des Netzes berechnet). Die flexiblere Form der LBAs stellen sicher die adaptiven Algorithmen dar, welche in Form von zentralisierten und isolierten Algorithmen realisiert werden.

Die **zentralisierte Leitwegbestimmung (LB)** basiert auf einem RCC (Routing Control Center), welches ab und zu Informationen über die aktuellen Nachbarn, Warteschlangen und den verarbeiteten Datenverkehr an alle Router im Netz sendet. Das RCC berechnet alle optimalen Leitwege und verteilt sie dann wie oben beschrieben an alle Router. Probleme können hierbei auftreten, wenn das RCC abstürzt, was fatale Folgen haben kann.

Die **isolierte LB** geht davon aus, daß wenn ein Paket mit unbekanntem Ziel auftaucht, es so schnell wie möglich an einen Ausgang mit kurzer Warteschlange geschoben wird (**Hot Potato-Technik**). Ist das Ziel bekannt, wird das Paket korrekt an den jeweiligen Router weitergeleitet. In diesem Prozeß merkt sich der Router die Quellen aller ankommenden Pakete, um sie später als Ziele verwenden zu können. Es erfolgt somit ein Rückwärtslernen aller Leitwege. Die Form der Abspeicherung der Leitwege erfolgt wie bei Brücken mittels Hash-Tabellen.

2.5.4 Formen von Routern

Es gibt bei Routern die oben beschriebene einfache Form und die Mischform des **Brouters**, bei dem die Vorteile der Brücke und des Routers vereinigt wurden. Der Brouter arbeitet auf den Schichten 2 und 3 und ist in der Lage ein Paket, welches nicht geroutet werden kann, mit den Mitteln einer Brücke weiterzuleiten.

2.6 Zusammenfassung

Das Internetworking gewinnt immer mehr an Bedeutung, da die Anzahl der LAN-Installationen ständig zunimmt. Nur eine strenge Normung von der Kommunikationsumgebung gewährleistet eine hohe Zuverlässigkeit des Gesamtnetzes. Anhand der OSI Schichten werden Internetworking-Geräte in 4 Klassen eingeteilt (Abb. 2.1).

2.7 Literaturverzeichnis

[Com88] [Mar92] [Her92]

Kapitel 3

Verteiltes Netzwerkmanagement

Torsten Auerbach

3.1 Kurzübersicht

Im Zuge der Entwicklung zu immer komplexeren Netzwerken und verteilten Applikationen[1] ist das Konzept einer zentralen Netzwerkverwaltung an seine Grenzen gestoßen. Die naheliegende Lösung liegt in einem verteilten Netzwerkmanagement (NM), dessen Prinzip in der Aufteilung der Management-Software und -Daten auf mehrere Management-Server entsprechend den Anforderungen des Verarbeitungsprozesses besteht. Diese Ausarbeitung soll einen Überblick über den Grundaufbau eines verteilten NM-Systems, über die herstellerübergreifenden Standardisierungsbemühungen der Open Software Foundation (OSF) und über die aktuellen Systeme am Markt geben.

3.2 Einleitung

Während das erste verteilte NM-System Openview von HP, das 1988 herauskam, allgemein noch als Vorgriff auf die Zukunft bewertet wurde, brachten die anderen führenden Netzwerkanbieter bereits kurze Zeit später eigene verteilte NM-Systeme auf den Markt (siehe Inhaltsverzeichnis). Wem würde es gelingen, die Standards für die 90er Jahre zu setzen? Ein Hersteller allein wäre wohl kaum in der Lage, die wichtigsten Forderungen an ein modernes verteiltes NM-System, wie allseitige Kompatibilität, Herstellerunabhängigkeit, Flexibilität, Transparenz und vor allem das Setzen und Unterstützen von Standards, zu erfüllen. Deshalb befaßt sich ein mächtiges Firmenkonsortium, die Open Software Foundation (OSF), die ursprünglich mit dem Ziel gegründet wurde, eine Industriestandardversion von UNIX zu entwickeln, mit der Entwicklung und Festlegung von Standards für verteiltes NM und mit der Definierung entsprechender verteilter System- bzw. Managementumgebungen, die diese Standards implementieren.

[1] z.B. Anwenderprogramme

3.3 Grundaufbau verteilter NM-Systeme

Moderne verteilte NM-Systeme sind meist in einer Client[2]-Server-Architektur aufgebaut (siehe Abbildung 3.1). Die einzelnen Komponenten und ihre Funktionen:

Management-Server Hauptkomponente; speichern Management-Daten und arbeiten den Großteil der Management-Software ab; tauschen Daten mit Clients, Applikationen und Agents [3] aus;

verwendetes Kommunikationsverfahren: Remote Procedure Call (RPC); [4]

RPC ist eine erweiterte Form des normalen Unterprogrammaufrufes, wie er in den meisten modularen Programmsystemen verwendet wird. In einer RPC-Umgebung können sich Programmodule, die sich in verschiedenen Rechnern befinden, gegenseitig aufrufen sowie Argumente und Resultate einander übergeben.

Vorteile des RPC: Ressourcen können leicht und ohne Umcodieren lokalisiert werden. Die Abarbeitung von Applikationen wird verteilt durchgeführt. Mit RPC wird das verteilte Management-System optimal in das zu verwaltende verteilte System integriert. Konventionelle Management-Protokolle wie CMIP (Common Management Information Protocol)

[2] Benutzerzugang zu den Management-Diensten ermöglichender Rechner,z.B. PC als Endgerät

[3] in den zu verwaltenden Netzressourcen gespeicherte Managementsoftware-Module

[4] Fernprozeduraufruf

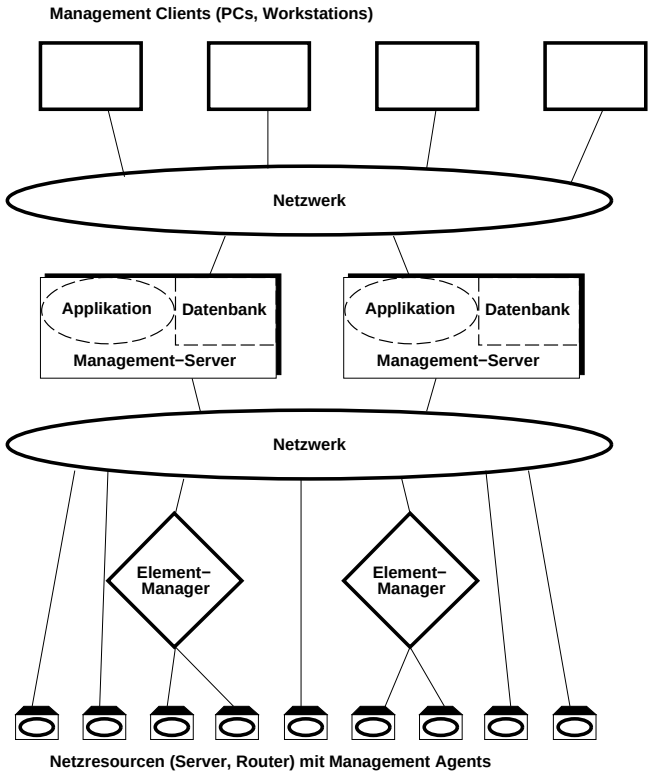


Abbildung 3.1: Verteiltes NM auf einen Blick

oder SNMP (Simple Network Management Protocol) werden durch Unterprogrammaufrufe zwischen den Agents und dem Rest des Systems ersetzt. RPC schränken den Netzverkehr ein, d.h. die verwalteten Geräte (z.B. Router) werden weniger mit der Abwicklung der Kommunikation belastet und können mehr Leistung erbringen als mit einfacheren Management-Protokollen. Problematisch ist bei RPC das Fehlen von Industriestandards.

Ein weiterentwickeltes Kommunikationsverfahren stellt das Message Passing dar (anstelle von oder in Verbindung mit RPC).

Es wurde speziell für objekt-orientierte Software entwickelt. Im wesentlichen kommunizieren die Objekte durch hin- und hersenden von Nachrichten, die Routinen (sogenannte Methoden) beinhalten. Nach Meinung der OSF ist Message Passing das zukunftsträchtigste Kommunikationsverfahren in einer objektorientierten Umgebung.

In manchen Systemen können Server nur über zwischengeschaltete, herstellerspezifische Elementmanager mit den Netzressourcen in Verbindung treten. Das ist das Haupthindernis für die Schaffung von Multivendorsystemen [5].

Management-Clients ermöglichen den Zugang zu den Management-Diensten für Benutzer und Applikationen; können im günstigsten Falle PCs oder Workstations mit unterschiedlichen Betriebssystemen (UNIX, OS/2, DOS, Macintosh) und von verschiedenen Herstellern sein; Sie bieten eine graphische Benutzeroberfläche (Graphical User Interface—GUI), die meist auf X-Windows basiert.

Management-Agents i.a. Softwaremodule, die in den verwalteten Netzressourcen gespeichert sind; Sie sammeln Management-Daten und übertragen sie paketweise auf der Basis von SNMP, CMIP oder dem fortschrittlichen RPC zu den Management-Servern.

verteilte Directories werden benötigt, damit das verteilte Management-System den Überblick über seine zahlreichen Komponenten behält; ermöglichen eine eindeutige Zuordnung der Komponenten zu *einem* Namen, der im gesamten Netz gültig ist und damit einen Dateizugriff von überallher im Netz

[5] Systeme,die aus Geräten und Software verschiedener Hersteller bestehen oder die zu Systemen anderer Anbieter kompatibel sind

Datenbank speichert und aktualisiert sämtliche Daten über Konfiguration und Bestand des Netzes und stellt sie den Management-Servern zur Verfügung; Meist werden relationale Datenbanken verwendet, zukunftsträchtiger sind jedoch objekt-orientierte Datenbankmanagement-Systeme. Problem: Es ist zwar möglich, ein vollständiges Datenbanksystem auf mehrere Server zu kopieren, aber Datenbanken auf verschiedenen Servern können sich noch nicht gegenseitig durch Änderungsmeldungen aktualisieren, d.h. relationale Datenbanken können Daten nicht partiell kopieren. Jedesmal muß die gesamte Datenbank übertragen werden. Sie stellt zur Zeit also die Achillesferse des verteilten NM dar, denn die gesamte Datenbank muß sich in einem Server befinden. Dieser Server kann bei Wide Area Networks (WANs) schnell zu einem Leistungsengpaß werden.

Sicherheit Verhinderung des unbefugten Zugangs zum Netz und Sicherstellung der Vertraulichkeit der Daten

1. Benutzerauthentifikation an den Clients
   Passwordverfahren haben sich für hohe Sicherheitsanforderungen als unzureichend erwiesen. Besser geeignet sind kryptographische Verfahren, wie das vom Massachusetts Institute of Technology (MIT) entwickelte System *Kerberos*.
2. Serverauthentifikation
   soll verhindern, daß mit einem unbefugten Rechner ein Server simuliert werden kann; Hacker könnten sonst die Kontrolle über sensible Netzbereiche erlangen.
3. Sicherheit der Kommunikation zwischen Management-Servern und Agents
   erforderlich für vertrauliche Daten und um zu verhindern, daß Agents von Unbefugten manipuliert werden, um verwaltete Geräte zu beeinflussen, z.B. um deren momentane Operation zu unterbrechen

Das Hauptziel besteht darin, Hardware und Software verschiedener Hersteller zu einem verteilten NM-System vereinen zu können, die Management-Software aufzusplitten und auf mehrere vernetzte Server zu verteilen.

Verteiltes NM bietet eine Reihe von Vorteilen:

- mehrere vernetzte Workstations und PCs stellen mehr Leistung zu niedrigeren Kosten zur Verfügung als Zentralrechner
- besser an veränderte Anforderungen anpaßbar
- Portierbarkeit der Management-Software und der Applikationen
- erheblich gesteigerte Ausfallsicherheit
- an den Clients steht eine einheitliche graphische Benutzeroberfläche zur Verfügung
- Management-Daten und Softwarewerkzeuge sind überall im Netz gleichermaßen verfügbar
- die Entwicklung kompatibler Systeme und Geräte wird zur Notwendigkeit und damit beschleunigt
- Zwang zur Standardisierung der Komponenten und Verfahren

3.4 Wer setzt die Standards?

Bisher wurden Industriestandards, wenn überhaupt, von den größten Unternehmen der Branche (z.B. IBM) festgelegt und basierten auf OSI[6]. Mittlerweile beschäftigen sich große Firmenkonsortien, wie die Open Software Foundation (OSF) und – in geringerem Maße – UNIX International (UI) mit dieser Aufgabe. Die OSF entwickelte die im folgenden vorgestellten Umgebungen DCE und DME. UI arbeitet an einer Spezifikation für verteiltes Management als Teil des Systems *Atlas*.In Fachkreisen vermutet man, daß es ähnliche Merkmale wie DME besitzen wird. Ob das Atlas-System allerdings eine ebenso große Unterstützung durch die Industrie erfahren wird, ist ungewiß.

3.4.1 Distributed Computing Environment (DCE)

DCE stellt Applikationen und Benutzern ein umfassendes und konsistentes System von Kommunikationsdiensten zur Verfügung. Die Umgebung ist herstellerunabhängig gegenüber Betriebssystem und Netzwerk. Besonders hervorzuheben ist die Kompatibilität zu den meisten existierenden Benutzerumgebungen und verteilten Systemen, wodurch eine der Hauptforderungen der Industrie, nämlich die Unterstützung bestehender und die Festlegung neuer Standards, erfüllt wird. Deshalb hat sich eine

[6] Open Systems Interconnection

Reihe von Firmen, unter anderen IBM, DEC, HP, Groupe Bull (F) und Siemens, für DCE entschieden.

Architektur

DCE ist nach einem Bottom-up-Schichtenmodell aufgebaut (siehe Abbildung 3.2). Für die Applikationen als oberste, aufgesetzte Schicht erscheinen die vielgestaltigen Dienste als ein homogenes logisches System. Die physikalische Komplexität der Netzumgebung wird verborgen. Alle Dienste können sowohl einzeln, als auch kombiniert benutzt werden. Sicherheit und Management umrahmen die übrigen Dienste und stehen ihnen zur Verfügung. Für die Kommunikation zwischen Applikationen wird RPC verwendet.

Die Dienste sind in 2 Kategorien unterteilt:

1. fundamentale verteilte Dienste: bieten Programmierwerkzeuge für die Entwicklung von Benutzerdiensten und Applikationen

2. Data-Sharing-Dienste:[7] bauen auf den fundamentalen Diensten auf und ermöglichen eine bessere Datennutzung

Merkmale und Vorteile der fundamentalen Dienste

RPC Grundprinzip: einzelne Prozeduren einer Applikation laufen auf einem beliebigen Rechner im Netz, RPC dezentralisiert also die Ausführung der Applikationen; RPC garantiert mehr Leistung, Einfachheit, Netzwerkunabhängigkeit und Portabilität der Software sowie Unabhängigkeit vom verwendeten Transportdienst. Sowohl verbindungsloser als auch verbindungsorientierter Transport werden unterstützt. RPC ermöglicht die effiziente Bewältigung großer Datenmengen durch die Zulasssung unbegregrenzter Variablengrössen. Die Internationalisierung wird durch Datentypen für sehr umfangreiche Zeichensätze, wie Japanisch, Arabisch und Chinesisch, gefördert, entsprechend den Standards der ISO.

Darstellungsdienste verbergen die Unterschiede in der Datenrepräsentation bei verschiedenen Rechnern; Dadurch laufen Programme in heterogenen Systemen überall.

Threads sind Teilprozesse, die einem Prozeß Prozessorzeit zuteilen und eine quasiparallele Verarbeitung ermöglichen, da sie unabhängig voneinander ablaufen. Beispielsweise kann während der eine Thread einen RPC ausführt, ein anderer eine Benutzereingabe verarbeiten. Der Threads-Dienst beinhaltet Operationen für die Erzeugung und Steuerung mehrerer Threads in einem Prozeß und für die Synchronisierung des Zugriffs auf globale Daten innerhalb einer Applikation. Dadurch kann ein Server-Prozeß gleichzeitig mehrere Clients bedienen und andererseits auch ein Client mit mehreren Servern zusammenwirken. Dieser Dienst unterstützt damit einen transparenten Mehrprozessorbetrieb.

Zeit Dienst zur präzisen, fehlertoleranten Synchronisierung der Systemuhren aller Rechner im Netz; Viele Applikationen benötigen eine Zeitbasis, um Operationen zeitlich zu koordinieren und in eine Reihenfolge zu bringen. Durch diesen Dienst bekommt also jedes Gerät im Netz Informationen darüber, ob seine eigene Systemzeit korrekt ist. Die Zeit für die Übertragung dieser Daten wird vernachlässigt. Um das Network Time Protocol (NTP) zu unterstützen, ist auch eine Verwendung von Zeitsignalen externer Quellen vorgesehen.

Naming unterstützt verteilte Directories; Sämtliche Betriebsmittel innerhalb der verteilten Umgebung erhalten einen eindeutigen und nach einem einheitlichen Muster festgelegten Namen. Über diesen Namen können die Ressourcen eindeutig identifiziert werden und der Zugriff auf sie kann ohne Kenntnis der eigentlichen Position erfolgen. Das verteilte Directorysystem basiert auf X.500, das Naming auf dem Distributed Name System (DNS) von Digital Equipment.

Sicherheit In den meisten konventionellen Time-Sharing-Systemen übernimmt das Betriebssystem die Bestätigung der Identität eines Benutzers und der Zugangsberechtigung. In einer verteilten Umgebung wird dafür ein eigenständiger Dienst benötigt, der vom gesamten Netz aus aufgerufen werden kann. Die Authentifikation[8] basiert auf dem *Kerberos*-System des MIT, einem kryptographi-

[7] Datenverbunddienste

[8] Bestätigung der Identität eines Benutzers oder eines Dienstes

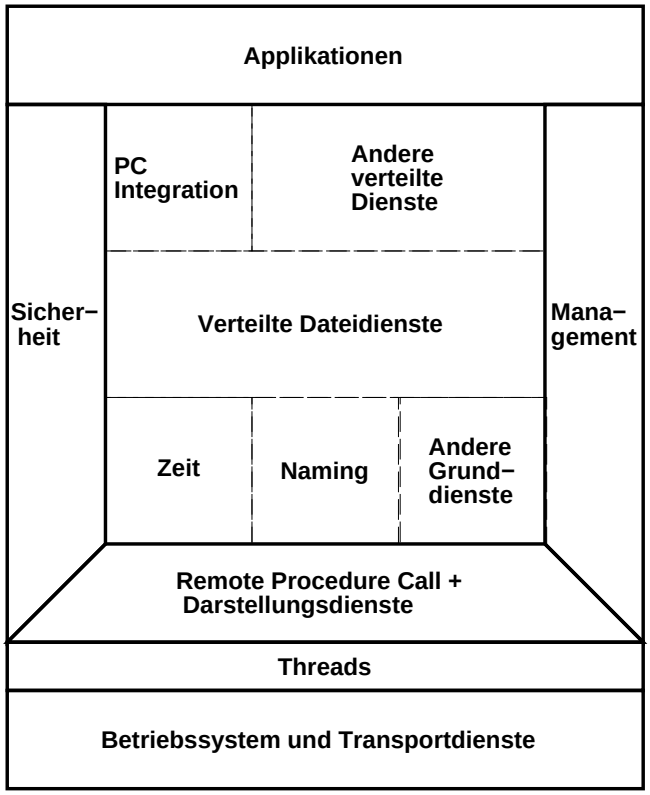


Abbildung 3.2: Architektur der Distributed Computing Environment

schen Verfahren, das eine höhere Sicherheit als Passwörter gewährleistet. Für die Regelung des Zugriffs auf die Ressourcen stehen Autorisations-Tools zur Verfügung, die auch die entsprechenden Kontrolldaten verwalten. Das User-Account-Management wird von der sogenannten User Registry übernommen, ein System, das eine Datenbank mit den Benutzerzugangsdaten verwaltet und die Eindeutigkeit und Genauigkeit der Benutzernamen und Codewörter überall im Netz gewährleistet. Sichere RPCs ermöglichen die Erkennung von Nachrichtenverfälschungen und sichern die Vertraulichkeit der Daten.

Merkmale und Vorteile der Data-Sharing-Dienste

Distributed File System [9] verbindet die Dateisysteme einzelner Rechner über eine einheitliche Schnittstelle; ermöglicht einen ebenso leichten globalen wie lokalen Dateizugriff; Ein protokollgestütztes physisches Dateisysytem erhöht durch die Möglichkeit einer schnellen Wiederherstellung nach einem Serverausfall die Zuverlässigkeit. Dateien und Directories werden unmittelbar auf mehrere Rechner kopiert, was permanente Verfügbarkeit der gespeicherten Daten im Netz, selbst bei Ausfall einzelner Geräte, ermöglicht. Die Sicherheit wird durch Access Control Lists (ACL), die den Zugriff auf Dateien überwachen, und einen sicheren RPC-Dienst gewährleistet. Das Problem des Zugriffs auf entfernte Dateien wird vom Distributed File System unabhängig von deren Größe, Position im Netz und Typ der verwendeten Hardware gelöst. Für den Benutzer stellt es sich wie ein lokales Dateisystem dar. Der Dateizugriff kann über einen einheitlichen Dateinamen vom gesamten Netz aus erfolgen. Es ist kompatibel zum Network File System (NFS) von Sun Microsystems.

Diskless Support Unterstützung von Workstations ohne Festplatte

PC-Einbindung Unterstützung von Desktop-Geräten; PCs unter den Betriebbsystemen DOS, OS/2, DOS mit Windows und Macintosh-PCs können in UNIX-Systeme und andere DCE-kompatible Systeme integriert werden. Weiterhin stehen den Software-Entwicklern Application Program Interfaces (APIs) zur Verfügung, die die Entwicklung

[9] verteiltes Dateisystem

verteilter Applikationen für PCs erleichtern. Die DCE-Directory-Dienste ermöglichen auch PC-Benutzern einen einfachen Zugang zu verteilten Daten und Rechenkapazitäten im Netz.

Standards Folgende Standards werden unterstützt:

1. OSI-Standards
2. ISO[10]-Standards: u.a. ROSE (Remote Operations Service Element), ACSE (Association Control Service Element), Directory System CCITT X.500, Sitzungs- und Darstellungsdienste
3. Internet-Standards: TCP/IP-[11]Transport- und Netzwerkprotokolle
4. X/Open-Richtlinien für Directory- und Transportdienste:
   XTI (X/Open Transport Interface)
5. offen für die Implementierung neuer Standards

DCE wird als Quellencode in Standard-C geliefert und ist damit auf unterschiedliche Systemumgebungen (z.B. VMS) portierbar. Einen Nachteil besitzt die Umgebung jedoch: Es wird eine neue, kompliziertere Ebene von Problemen beim Management verteilter Systeme geschaffen. Speziell für diesen Anwendungsbereich entwickelte die OSF die im folgenden vorgestellte Distributed Management Environment (DME).

3.4.2 Distributed Management Environment (DME)

Mit der DME hat die OSF einen wichtigen, wenn nicht sogar den wichtigsten Standard der 90er Jahre für das effektive Management offener verteilter Systeme geschaffen. Bereits heute erfährt DME eine breite Unterstützung durch die Industrie.

Architektur der DME

Die DME basiert auf DCE und ist größtenteils als objekt-orientierte Software implementiert, was zu modularen flexiblen Lösungen bei den Applikationen führt und die Integrierung von Management-Applikationen verschiedener Hersteller erleichtert. Daten und Operationen sind in Objekte klassifiziert. Alle DME-Managementoperationen können über die Kommunikation mit Objekten ausgeführt werden. In der DME gibt es im wesentlichen zwei Arten von Objekten (Managed Objects): herstellerkompatible Objekte (z.B. Server) und herstellerspezifische Objekte (z.B. eine spezielle Softwareversion für Router X). Diese Menge von Objekten kann beliebig erweitert werden.

Management Request Broker (MRB) ermöglichen die Kommunikation zwischen Applikationen, Agents und Objekt-Servern; nehmen Anforderungen an Objekte an, lokalisieren die Objekte und geben die Anforderungen weiter;
2 Typen von MRBs:
Typ 1 unterstützt SNMP und CMIP und sammelt Management-Daten der Netzressourcen; bietet eine Standard-Programmierschnittstelle zu SNMP und CMIP (CM-API [12] bzw. XMP von X/Open);

Typ 2 unterstützt ein auf RPC basierendes Managementprotokoll; bietet eine objektorientierte Umgebung zur Implementierung von Management- Applikationen und Basis-APIs, die deren Zugriff auf die DME-Dienste ermöglichen;

Objekt-Server enthalten die Objekte, aktualisieren und verwalten sie; ermög lichen den Zugriff auf die in den Objekten gespeicherten Daten;

DME-Dienste und Tools

Ereignis-Management informiert den Administrator bzw. Applikationen über Probleme und Veränderungen im Netz; Dafür können Filter programmiert werden, die die Merkmale einer Ereignismeldung analysieren und entsprechende Reaktionen auslösen.

Daten-Management Sicherung der in den Objekten gespeicherten Daten auf Platte; Durch Protokollierung ist ein roll-back nach Systemabsturz möglich, wodurch die Zuverlässigkeit der Datenspeicherung erhöht wird.

Benutzeroberfläche basiert auf OSF Motif; Die mit einem Objekt verknüpfte Benutzeroberfläche und die auf ihm möglichen Operationen können mittels einer Dialogsprache definiert werden. Diese Definition ist im verwalteten Objekt gespeichert und wird nach Anforderung vom Display Manager interpretiert.

[10] International Organization for Standardization
[11] Transmission Control Protocol/Internet Protocol
[12] Consolidated Management-API

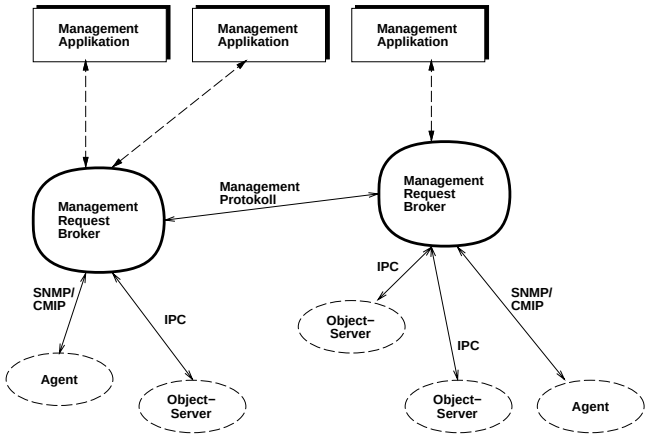


Abbildung 3.3: Architektur der Distributed Management Environment

Vorteilhaft sind die Darstellung der verwalteten Objekte durch Icons, die dadurch mögliche übersichtliche Visualisierung der Systemstruktur und die Konsistenz in der Bedienung quer durch die Systeme und Managementdienste. Für die Benutzung sind keine Programmierkenntnisse erforderlich.

Management flexibel und an sehr unterschiedliche Größenbereiche anpaßbar; Zerlegung des Management in kleinere Teilbereiche;unterschiedliche Arten der Systemverwaltung möglich; stellt kein festgefügtes Modell dar, sondern bestimmt Standards für die Entwicklung neuer Systeme;

Entwickler-Tools Objekt-orientierte APIs, basierend auf ANSI-C bzw. C++, vereinfachen die Entwicklung von Management-Applikationen für die MRBs, insbesondere von graphischen. Für die Erstellung von Benutzeroberflächen, für Ereignismanagement und Objektimplementierung stehen Tools in höheren Programmiersprachen zur Verfügung.

PC-Unterstützung ermöglicht vollständige Integrierung und das Management von DOS-PCs in heterogenen verteilten Umgebungen

Vorteile der DME

Besoders hervorzuheben ist die Kompatibilität der DME zu den meisten existierenden Management-Standards. Damit ist man ein gutes Stück vorangekommen auf dem Weg zu einem echten Multivendor-System. Die Benutzerfreundlichkeit wurde durch die Konsistenz innerhalb der zu verwaltenden Umgebung, die unkomplizierte Bedienung und die bessere Übertragbarkeit von Nutzerkenntnissen vom Einzelsystem auf umfangreiche Netzwerke und Supercomputer bedeutend verbessert. Für Applikationen hat DME den Markt wesentlich erweitert, da aufgrund genormter Schnittstellen und Dienste und umfassender Entwickler-Tools Drittfirmen eigene Entwicklungen anbieten können.

3.5 Die 5 führenden Systeme– Jeder gegen jeden?

3.5.1 SunNet Manager von Sun Microsystems

Die Firma Sun Microsystems versucht es auf dem Gebiet des verteilten NM noch im Alleingang. Dahinter stecken vermutlich Ängste, man könnte seine Vormachtstellung im UNIX-Bereich durch eine zu enge Zusammenarbeit mit den großen Konkurrenzfirmen bald einbüßen. Gegenwärtig nimmt das SunNet-Manager-System den Spitzenplatz auf dem Markt ein, was diese Strategie bestätigt. Ob das Konzept auch in Zukunft aufgehen wird, erscheint jedoch fraglich.

SunNet Manager ist in der Lage, verteilte Multivendornetze und -systeme zu verwalten, deren Größe und Komplexität stark variieren kann (bis zu 10^4 Netzknoten. Das Management von TCP/IP-Netzen erfolgt mittels SNMP, das von DECnet- oder FDDI-Netzen unter Zuhilfenahme von SunConnect.

Merkmale

- graphische Benutzeroberfläche und graphische Hilfsprogramme für Netzkonfigurierung, Überwachung, Steuerung, Zukunftsanalyse und Reports (z.B. Grapher Tool für die graphische

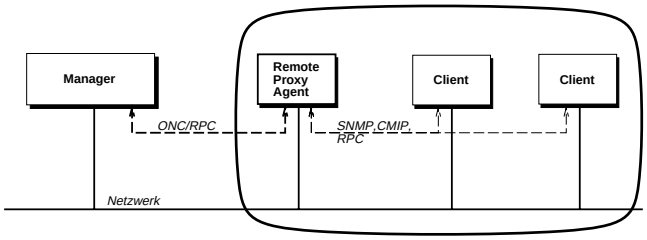


Abbildung 3.4: Proxy Agents übersetzen das ONC/RPC-Protokoll in das Protokoll, das die zu verwaltenden Geräte verstehen

Darstellung von aktuellen Netzparametern)

- einfache transparente Architektur
- Konzept der Remote Proxy Agents[13] zur Überbrückung großer Entfernungen in Weitverkehrsnetzen (WANs) (siehe Abbildung 3.4)
- Kommunikation zwischen SunNet Manager und Remote Proxy Agent mittels verteilter ONC-Dienste (Open Network Computing) auf der Basis von RPC
- Remote Proxy Agents übersetzen das RPC-Protokoll in ein Protokoll, das die verwalteten Ressourcen "verstehen"; Sie stellen damit die Kompatibilität zu SNMP und CMIP her.
- Remote Proxy Agents sammeln und selektieren Daten, um nur die wichtigsten Informationen paketweise über das WAN zum SunNet Manager zu übertragen.
- dadurch Reduzierung der Management-Abfrageaktivitäten und des Datenverkehrs im Netz
- Ereignisse und Traps[14] automatisieren das Management und verteilen die Arbeitsbelastung von Konsole und Manager (siehe Abbildung 3.5)
- Nachteil der Remote Proxy Agents: keine Weiterverzweigung vom Proxy aus möglich, d.h. es ist keine Verbindung zwischen zwei Proxies zulässig (immer nur über den Manager) (siehe Abbildung 3.6)
- Umfangreiche Zusammenstellung von NM-Lösungen in einem Solutions-Portfolio
- Drittfirmen entwickeln innerhalb des SunConnect-Partners Programmes Lösungen für NM-Probleme; diese Entwicklungen werden in dem Portfolio beschrieben
- zur Erleichterung von Fremdentwicklungen stehen genau definierte APIs zur Verfügung
- die Sicherheit wird durch sichere RPCs gewährleistet (Sicherheitsgruppen im Netz, Zugriffskontrolle, Authentifikation)

Insgesamt bietet SunNet Manager keine so umfangreiche Funktionalität und Ausbaufähigkeit wie OSF DME. Sun weigert sich sogar, die DME zu unterstützen. Wahrscheinlich ist man bei Sun der Meinung, die Abkürzung OSF stehe in Wirklichkeit für " Oppose Sun Forever". Über eventuelle Pläne von Sun, SunNet Manager auf den Stand von DME zu bringen, ist noch nichts bekannt, was ja nicht heißen muß, daß keine existieren.

3.5.2 OS/2 Distributed Systems Manager (DSM) von IBM

IBMs DSM spiegelt das Grundmodell eines Systems für offenes verteiltes NM wider. Ein DSM-Server implementiert die graphische Benutzeroberfläche, das Basisbetriebssystem (OS/2), die DSM Systemsoftware, APIs sowie Applikationen aus eigener und Fremdentwicklung (siehe Abbildung 3.7). In einem System können mehrere Server vorhanden sein. DSM unterstützt folgende Standards: CM-API, SNMP, CMOT[15] und CMOL[16]. Mit dieser Neuentwicklung, die eine breite Kompatibilität aufweist und für die IBM große Anstrengungen unternommen hat, will der blaue Riese einen großen Marktanteil bei Clients, Servern, Bridges usw. erringen. CM-APIs erleichtern zudem die Entwicklung von DSM-kompatiblen Applikationen durch

[13] entfernte 'stellvertretende' Agents
[14] asynchrone Ereignismeldungen eines Agents
[15] CMIP over TCP/IP
[16] CMIP over Logical Link Control–ein neuer LAN-Standard der IEEE

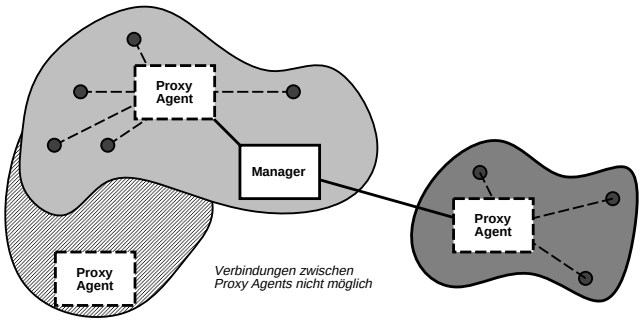


Abbildung 3.5: Manager-Proxy-Agenten-Relationen

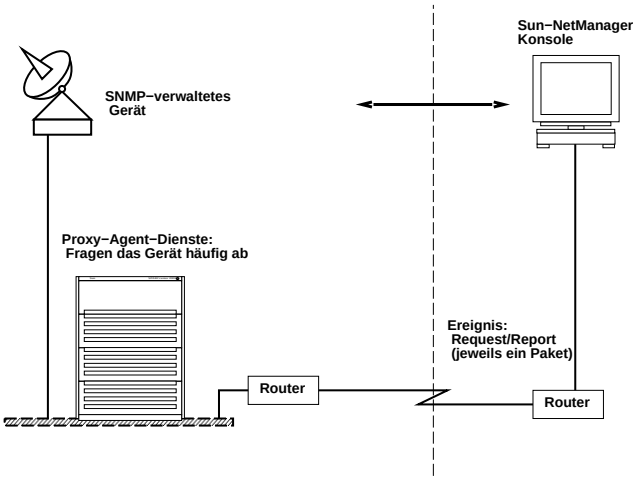


Abbildung 3.6: Beispiel für eine Topologie des SunNet Managers

Drittfirmen. IBM arbeitet auch an einer abgespeckten Version, die als Agent in OS/2-Clients und -Servern für geschäftliche Applikationen dienen wird.

Die Kommunikation zwischen den DSM-Servern basiert noch auf CMIP, später sollen dann RPC und Message Passing integriert werden. Mittels des Common User Access Interface (CUA-Interface) bekommen die Clients Zugang zu den DSM-Servern. Dadurch wird eine verteilte Lösung für Clients und Server möglich, bei der eine beliebige Anzahl intelligenter OS/2-, DOS-, DOS/Windows- oder UNIX-Clients auf die DSM-Managementsoftware, die auf den Servern läuft, zugreifen kann. DSM unterstützt ein Standard-SQL-[17]Interface für den Zugriff auf eine relationale Datenbank. Weiterhin kann es Daten mit Netview austauschen, einem zentralen Managementsystem von IBM. Zweifellos wird der Erfolg von OS/2 DSM stark davon abhängen, inwieweit sich OS/2 als Standard-Betriebssystem durchsetzen wird. Die Chancen stehen jedoch nicht schlecht.

3.5.3 Openview von Hewlett-Packard (HP)

Von allen Herstellern hat HP das Konzept einer offenen verteilten Management-Umgebung als erster und am intensivsten vorangebracht. Openview war die erste verteilte NM-Lösung auf dem Markt und bildete den Kern, aus dem später die DME der OSF entwickelt wurde. Folglich unerstützt HP maßgeblich die DME. Als Multivendorsystem konzipiert, läuft es unter dem Betriebssystem UNIX auf Workstations von HP und Sun und auf Kleinrechnern von HP. Natürlich steht eine graphische Benutzeroberfläche, basierend auf X-Windows bzw. OSF Motif, zur Verfügung.

Das System besteht aus den Komponenten:

- Kommunikatonsprotokolle
- Datenspeicherdienste mit SQL-Datenbank
- Benutzeroberflächendienste
- Objektmanager
- Management-Applikationen

Alle Komponenten sind untereinander über die sogenannte DCI (Distributed Communication Infrastructure) verbunden. Als Protokoll wird eine Variante von CMIP verwendet (CMOT), später soll RPC verwendet werden. Kompatibilität besteht auch zu SNMP. Openview erfreut sich einer weiten Verbreitung und unterstützt Drittentwickler durch CM-APIs.

3.5.4 DECmcc, eine Implementierung der Enterprise Management Architecture (EMA) von Digital Equipment (DEC)

Seit 1988 entwickelt und vertreibt DEC EMA-Software für eigene Netzwerkprodukte. Dabei wurde besonders darauf geachtet, daß Neuentwicklungen weiterhin EMA unterstützen. DECmcc, eine Implementierung der EMA, ist eine offene Management-Plattform, die für die Interprozeßkommunikation eine Kombination aus RPC und der DEC-Entwicklung Distributed Name Service (DNS) verwendet. Damit stellt EMA auch die komplizierteste verteilte Management-Architektur aller Hersteller dar. APIs ermöglichen es, 3 Typen von Applikationsmodulen zu installieren:

1. Zugriffsmodule
2. Funktionsmodule
3. Darstellungsmodule

Zugriffsmodule steuern die Verbindung zwischen Server und Agents. Funktionsmodule sichern die Verarbeitung der Management-Informatonen. Darstellungsmodule enthalten die Benutzeroberfläche und übernehmen die Report-Verarbeitung.

Einzigartig an DECmcc ist die Verwendung des DNS, um Module zu lokalisieren und dynamische Verbindungen zwischen Modulen aufzubauen. Alle anderen Hersteller verwenden hier primitivere Tabellen und Dateien, um die Positionen der verschiedenen Komponenten ihrer Systeme für die Wiederauffindung zu vermerken. Das hat den Nachteil, daß die Tabellen und Dateien jedesmal von Hand geändert werden müssen, wenn das Netz verändert wird.

Auch bei der verwendeten Datenbank-Technik bildet DECmcc eine Ausnahme: Es wird keine relationale Datenbank wie bei den anderen Herstellern verwendet, sondern ein objekt-orientiertes Datenbank-Managementsystem (DBMS). Obwohl die Zukunftsaussichten dieses Systems aufgrund seiner innovativen Technologie nur positiv bewertet werden können, hat DEC gegenwärtig einige

[17]Structured Query Language = strukturierte Abfragesprache

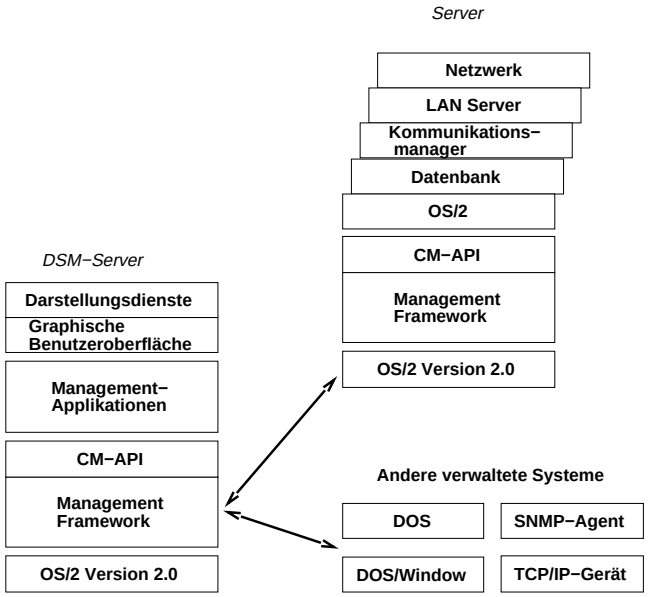


Abbildung 3.7: Architektur des OS/2 Distributed Systems Management von IBM

Schwierigkeiten, eine breite Akzeptanz und Unterstützung von DECmcc durch Drittfirmen zu erlangen, besonders in den USA. In Europa stellt sich die Situation durch die Zusammenarbeit mit Alcatel und Olivetti etwas günstiger dar.

3.5.5 Network Management System (NMS) von Novell

Novell definierte diese offene Management-Plattform speziell für das Management von PC-LANs, die unter Netware laufen, und um Drittfirmen die Entwicklung von Applikationen für dieses Einsatzgebiet zu ermöglichen. NMS verwendet entweder SNMP oder ein Novell-spezifisches Protokoll (oder beide) für den Zugriff auf die Management-Agents. Herstellern, die kompatible Agents für ihre eigenen PC-LAN-Produkte entwickeln wollen, stehen APIs zur Verfügung.

Aufbau und Funktion:

- Client-Server-Architektur
- Clients und Server können sowohl unter DOS/Windows als auch unter OS/2 laufen
- Server: arbeitet Management-Software ab und steuert den Zugriff auf Gateways zu anderen Management-Systemen, z.B. Netview
- Client: implementiert eine graphische Benutzeroberfläche und ermöglicht den Zugriff auf gesammelte Management-Daten

Bisher existiert noch keine allgemeinere Version, die RPC anstelle von SNMP bzw. des Novell-Protokolls benutzt. Da Novell der OSF angehört, dürfte die Integrierung von RPC in Entwicklung sein. Die Zielgruppe von NMS besteht eher aus kleineren Firmen oder Forschungsgruppen, die fast ausschließlich mit Netware oder kompatiblem Equipment arbeiten. Die Anpassung von NMS an das Management größerer Netze könnte Probleme bereiten. Novell beschränkt sich lieber auf spezielle, kleinere Einsatzbereiche und überläßt die komplexeren Aufgaben den anderen Herstellern. Wenn sich die Entwicklung im PC-Sektor mehr zum Netzwerk- und Systemmanagement als zu eigenständigen Einzelgeräten vollziehen sollte, steht Novell auf jeden Fall besser da als Microsoft, das sich offensichtlich mehr auf Client-Software konzentriert und die Server und verteilten Architekturen lieber DEC überläßt. Beide Firmen verbindet eine Produktentwicklungs-Kooperation.

3.6 Vergleich der Konkurrenten

3.6.1 IBM

Produkt OS/2 Distributed Systems Management (DSM)

Entwicklungsstand lieferbar; viele herstellerunabhängige Interfaces und Applikationen

Server-Betriebssystem OS/2

Client-Betriebssystem OS/2

Management-Protokoll SNMP, CMOT, CMOL

Interprocess Communication (IPC) CMIP

Datenbank SQL

API CM-API

Kommentar bedeutsamer Vorstoß eines Zentralrechner-Giganten in das verteilte NM

Ausblick gute herstellerunabhängige Applikationen zu erwarten; Erfolg abhängig von der Durchsetzung von OS/2

3.6.2 DEC

Produkt DECmcc

Entwicklungsstand verwaltet alle DEC-Produkte; einige herstellerunabhängige Interfaces und Applikationen

Server-Betriebssystem VMS, Ultrix, OSF/1

Client-Betriebssystem X-Windows

Management-Protokoll SNMP, NICE, CMIP

Interprocess Communication (IPC) DEC RPC

Datenbank DEC-spezifisch, objekt-orientiert

API DEC-spezifisch; volle OSF-Unterstützung angekündigt, aber kein Zeitraum angegeben

Kommentar einziges System mit voll ausgereiftem Directory-Dienst

Ausblick gute Zukunftsaussichten mit OSF APIs; gegenwärtig noch einige Akzeptanz-Schwierigkeiten

3.6.3 HP

Produkt Openview

Entwicklungsstand lieferbar; verwaltet alle HP-Produkte

Server-Betriebssystem HP-UX

Client-Betriebssystem OSF Motif

Management-Protokoll SNMP, CMOT, CMIP

Interprocess Communication (IPC) CMIP-Variante

Datenbank SQL

API CM-API; volle OSF-Unterstützung angekündigt, aber kein Zeitraum angegeben

Kommentar vielleicht als erster Hersteller in der Lage, ein DME-kompatibles System herauszubringen

Ausblick weit verbreitet, mehr herstellerunabhängige Applikationen zu erwarten

3.6.4 Novell

Produkt Network Management System (NMS)

Entwicklungsstand lieferbar; viele herstellerunabhängige Applikationen versprochen

Server-Betriebssystem OS/2, Windows

Client-Betriebssystem OS/2, Windows

Management-Protokoll SNMP, Novell-spezifische Protokolle

Interprocess Communication (IPC) keine, da keine mehrfachen Management-Server

Datenbank Novell Btrieve

API Novell-spezifische APIs inkl. Novell Hub Management Interface; keine Unterstützung der OSF

Kommentar komplizierteste Management-Architektur aller PC-LAN-Hersteller

Ausblick hervorragende Perspektive; Novell ist einer der wenigen Hersteller, die noch einseitig De-facto-Standards setzen können. NMS könnte einer werden.

3.6.5 Sun

Produkt SunNet Manager

Entwicklungsstand lieferbar; verwaltet alle Sun-Produkte; viele herstellerunabhängige Applikationen

Server-Betriebssystem SunOS

Client-Betriebssystem X-Windows

Management-Protokoll SNMP, Sun RPC

Interprocess Communication (IPC) Sun ONC (RPC)

Datenbank keine

API Sun-spezifisch; keine Unterstützung der OSF

Kommentar einziges Management System, das vollständig RPC nutzt

Ausblick Bisher führend, aber die langfristigen Aussichten verschlechtern sich durch die Weigerung, die OSF und offene Interfaces zu unterstützen.

3.7 Zusammenfassung und Zukunftsszenario

Nachdem man es noch vor wenigen Jahren den zentralen Management-Systemen durchaus zutraute, den Anforderungen für die nächsten Jahre gewachsen zu sein, vollzog sich die Entwicklung bei Netzwerken und verteilten Systemen so rasch,

daß die eigentlich für die Zukunft gedachten verteilten Management-Systeme schneller als geplant benötigt wurden. Die derzeitige Situation auf dem Gebiet des verteilten NM verdeutlicht, daß ein Hersteller allein kaum mehr in der Lage ist, in der Kürze der zur Verfügung stehenden Zeit ein allgemein akzeptiertes und anforderungsgerechtes System zu entwickeln, daß dann auch noch allseits kompatibel ist. Solange keine umfassenden Industriestandards für verteiltes NM festgelegt sind, kann die Forderung nach Kompatibilität nur unzureichend erfüllt werden.

Alle diese Tatsachen sprechen für ein Konsortiums wie die OSF, das die entscheidenden Impulse in Entwicklung und Standardisierung geben kann und die Interessen und Potentiale der einzelnen Hersteller so produktiv wie möglich zusammenfassen soll. Wenn das Gerüst definiert ist, können bei einem modularen Aufbau individuelle Anforderungen der Anwender umso besser erfüllt werden, da verschiedene Teillösungen auch zusammenpassen. Für die einzelnen Hersteller sinkt dann auch das Risiko, den Anschluß zu verlieren, weil das eigene System sich nicht durchsetzen konnte.

Die DME wird aller Voraussicht nach zum Standard für das verteilte NM der 90er Jahre werden und die Grundlage für weitere Entwicklungen sein. Begründet wird diese Annahme durch ihren modularen Aufbau , der die Ausbaufähigkeit garantiert, die breite Unterstützung der wichtigsten Hersteller, das offene Konzept für die Entwicklung von Applikationen, die Definierung vernünftiger Standards und die Unterstützung bestehender und die Kompatibilität zu den wichtigsten Betriebssystemen und Netzwerkprodukten.

Die Ausnahmerolle von Sun könnte schon bald zu einer Außenseiterrolle werden, wenn sich die Mitglieder der OSF einig sind.

3.8 Literaturverzeichnis

[Her92] [DCE92] [DME92]

Anhang A

Abkürzungsverzeichnis

ACL Access Control List

ACSE Association Control Service Element

ANSI American National Standards Institute

API Application Program Interface

CM-API Consolidated Management API

CMIP Common Management Information Protocol

CMOL CMIP over Logical Link Control

CMOT CMIP over TCP/IP

CUA Common User Access

DBMS Database Management System

DCE Distributed Computing Environment

DME Distributed Management Environment

DNS Distributed Name Service (System)

DSM Distributed Systems Management

EMA Enterprise Management Architecture

FDDI Fiber Distributed Data Interface

GUI Graphical User Interface

IPC Interprocess Communication

ISO International Standards Organization

LAN Local Area Network

MIB Management Information Base

MRB Management Request Broker

NFS Network File System

NICE Network Information and Control Exchange

NM Netzwerkmanagement

NMS Network Management System

NTP Network Time Protocol

ONC Open Network Computing

OSF Open Software Foundation

OSI Open Systems Interconnection

ROSE Remote Operation Service Element

RPC Remote Procedure Call

SE Service Element

SNMP Simple Network Management Protocol

SQL Structured Query Language

TCP/IP Transmission Control Protocol / Internet Protocol

UI Unix International

WAN Wide Area Network

XMP X/Open Management Protocol

XTI X/Open Transport Interface

Literaturverzeichnis

[Com88] Douglas Comer. *Internetworking with TCP/IP:principles, protocols and architecture*. Prentice-Hall, 1988. ISBN 0-13-470154-2.

[DCE92] Distributed computing environment overview. *OSF White Papers*, Juni 1992.

[DME92] Distributed management environment overview. *OSF White Papers*, Juni 1992.

[Her92] James Herman. Distributed network management. *DATA COMMUNICATIONS*, Juni 1992.

[Mar92] Martin. Building backbones with bridges and routers. *Networking Management Europe*, Mai/Juni 1992.

[Wal91] S. Waldbusser. Remote network monitoring management information. *RFC-1271*, November 1991.

[WNH92] S. Waldbusser, M. Nair, and M. Hoerth. SNMP management goes down to the wire. *DATA COMMUNICATIONS*, November 1992.