Institut für Informatik
der Ludwig-Maximilians-Universität München

Diplomarbeit

Erstellung und Anwendung eines Kriterienkataloges zur Klassifikation von Management-Plattformen

Bearbeiter	:	Thorsten Dombach
Aufgabensteller	:	Prof. Dr. Heinz-Gerd Hegering
Betreuer	:	Alexander Keller
Abgabedatum	:	28.05.1997

Ehrenwörtliche Erklärung

Ich versichere, daß ich diese Diplomarbeit selbständig verfaßt und nur die angegebenen Quellen und Hilfsmittel verwendet habe.

München, 30.05.1997

Unterschrift des Diplomanden

Inhaltsverzeichnis

Abbildungsverzeichnis

1 Einleitung

Einführend in die Thematik dieser Diplomarbeit, werden im folgenden das Umfeld und die Motivation sowie die Aufgabenstellung dieser Arbeit beschrieben.

1.1 Umfeld und Motivation der Arbeit

Durch die drastisch fallenden Hardwarekosten haben sich die EDV-Infrastrukturen in Unternehmen verändert. Auf dem Markt läßt sich eine Entwicklung hin zu verteilten Anwendungen beobachten. Wie in Abbildung 1.1 dargestellt, wandeln sich die Mainframe-Strukturen zu verteilten Hardware-Software-Verbunden (HW/SW). Die Komponenten eines solchen Verbundes kooperieren, um eine verteilte Ausführung einer Arbeit zu ermöglichen.

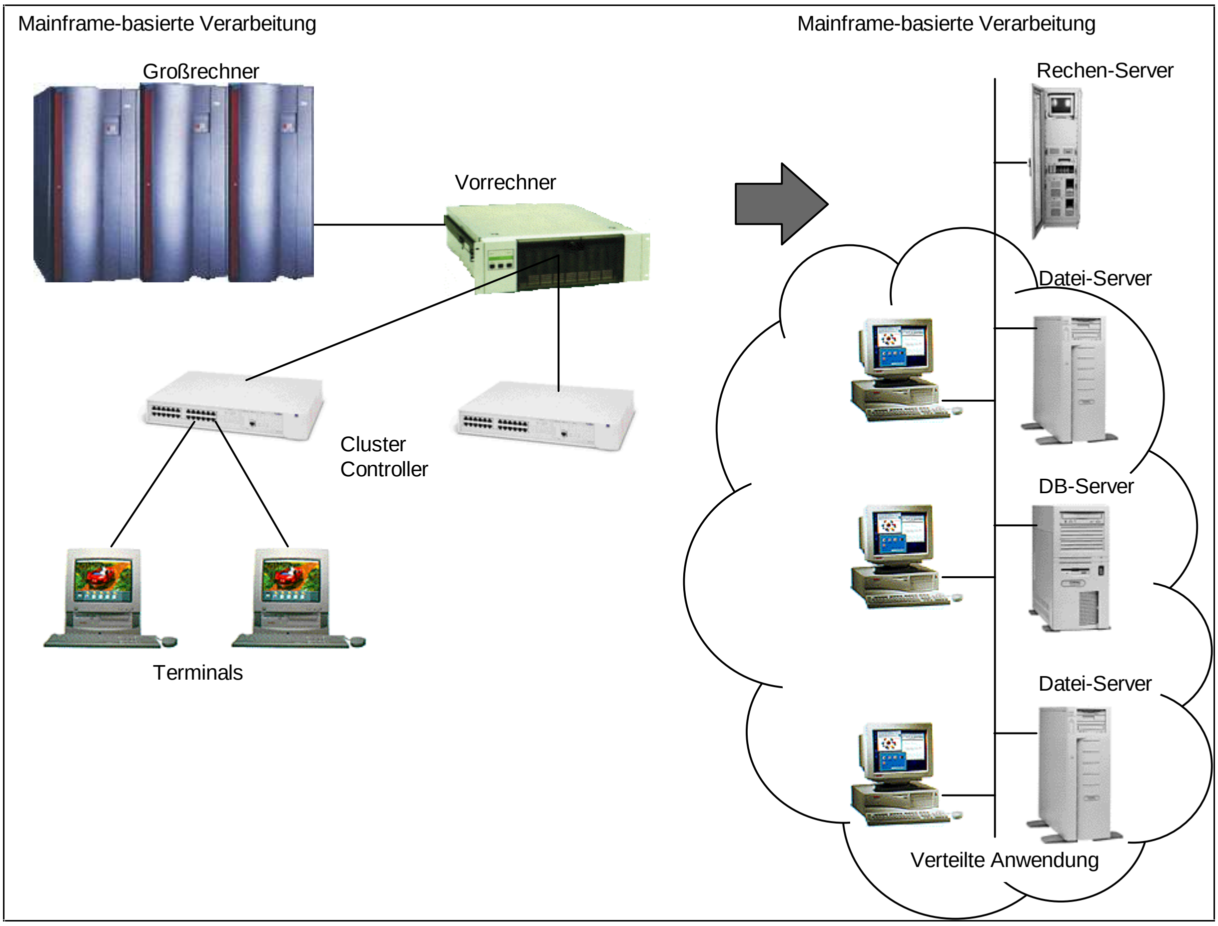

Abbildung 1.1: Neue DV-Versorgungsstrukturen ([DEML95], S. 1)

Solche Anwendungen werden oft nach den Funktionsbereichen Datenhaltung, Datenverarbeitung und Datenpräsentation unterteilt. Die Zusammenarbeit zwischen den Moduln, die die Funktionsbereiche abbilden, werden oft nach dem Client/Server-Prinzip erledigt. Die einzelnen Komponenten können dabei eine oder mehrere Aufgaben erfüllen. Das Client/Server-Prinzip ermöglicht eine flexible Verteilung der Komponenten auf Systeme. Daraus entstand die Anpassung an die speziellen Bedürfnisse eines Unternehmens, was auch „Rightsizing“ genannt wird.

Den Vorteilen, wie günstiger Preis, flexiblere Strukturen und gute Skalierbarkeit, stehen neue Probleme im Netz- und Systemmanagement gegenüber. Bei den alten SNA-Strukturen handelt es sich um ein hierarchisch orientiertes Netz. Den Terminals wird dabei Zugriff auf die Anwendung gegeben, die sich auf dem Mainframe befindet und dort ausgeführt wird. Beim Client/Server-Prinzip hingegen können die Anwendungen über mehrere Rechner im Netz verteilt ausgeführt werden. Dies erfordert neue Wege im Management, um den gestiegenen Anforderungen Rechnung zu tragen.

Die neuen Ansprüche lassen sich in großen EDV-Abteilungen moderner Unternehmen nur rechnergestützt erfüllen. Die Aufgaben sollen jedoch nicht von isolierten Management-Werkzeugen erledigt werden, sondern es soll eine integrierte Managementlösung gefunden werden. Diese soll die Netz-Umgebung mit ihren Anwendungen nach einem einheitlichen Modell überwachen und steuern.

Die Schwierigkeit eines Netzbetreibers besteht in der Auswahl der geeigneten Management-Plattform. Die Vielzahl der angebotenen Produkte sowie die verschiedenen Terminologien der Industrie erschweren die Auswahl zusätzlich.

Um einem Netzbetreiber den Entscheidungsprozeß zu erleichtern, muß eine geeignete Hilfe gefunden werden. Dieses Werkzeug, ein Kriterienkatalog, soll auf die wichtigen Punkte einer Management-Plattform hinweisen und eine objektive Bewertung der einzelnen Produkte ermöglichen.

1.2 Thema der Arbeit

In dieser Arbeit wird ein Kriterienkatalog entwickelt, der einen übersichtlichen Aufbau haben wird. Die Gliederung des Kataloges enthält Kriterien, diese stellen wichtige Anforderungen an ein Management-System dar. Ein Bewertungssystem und Verfahren zum Messen geben die Möglichkeit, mit Hilfe der Kriterien eine Management-Plattform zu bewerten. Durch die Anwendung des Kriterienkataloges auf verschiedene Produkte, werden diese vergleichbar.
Die Funktionalität heutiger Management-Plattformen ändert sich sehr schnell, deshalb wird der Kriterienkatalog eine übersichtliche Gliederung aufweisen. Außerdem wird der Benutzer den Kriterienkatalog um eigene Anforderungen erweitern können. Der Kriterienkatalog wird dann noch durch die Anwendung auf zwei Plattformen getestet.

1.3 Zusammenfassung

Im folgenden Kapitel wird zunächst der Aufbau einer Management-Plattform erläutert und die aktuellen Entwicklungen am Markt vorgestellt. Zur Entwicklung eines Kriterienkataloges ist eine genaue Kenntnis der verschiedenen Management-Architekturen erforderlich.
Die heutige Problematik des Managements soll anhand eines *R/3*-Szenarios dargestellt werden. Hierzu wird die Architektur einer solchen verteilten Anwendung erläutert und die Anforderungen an das Management formuliert, die sich durch das *R/3* ergeben.
Am Ende werden noch die nötigen Grundlagen zum Bewerten und Messen gegeben.

Im darauf folgendem Kapitel wird die Vorgehensweise geschildert, um einen Kriterienkatalog zu entwickeln. Daraus ergibt sich dann die Gliederung des Kriterienkataloges ohne explizite Anforderungen. Durch ein Beispiel soll die Problematik beim Bewerten aufgezeigt werden.

Im vierten Kapitel wird die Anwendung der Bewertungstechniken im Kriterienkatalog erläutert. Der stichwortartige Kriterienkatalog benötigt eine detaillierte Beschreibung mit Beispielen, damit die Anforderungen auch verstanden werden. Außerdem wird die bekannte Struktur des Kriterienkataloges weiter verfeinert.

Kapitel 5 beschreibt die Anwendung des Kriterienkataloges in einer Testumgebung. Als erstes wird die Testumgebung dargestellt und die gestellten Aufgaben erläutert. Danach wird nacheinander der Testverlauf auf die Produkte IBM *NetView* und HP *OpenView* beschrieben, dabei werden die Probleme beim Lösen der Aufgaben geschildert und eine Bewertung der jeweiligen Management-Plattform durchgeführt.

Kapitel 6 faßt die durch diese Arbeit gewonnenen Erkenntnisse zusammen und gibt einen Ausblick auf mögliche Entwicklungen des Kriterienkataloges.

2 Grundlagen und Motivation der Arbeit

Damit die Entstehung des Kriterienkataloges verstanden werden kann, müssen erst einige Grundlagen und Definitionen vermittelt werden.
Deshalb wird in diesem Kapitel auf die wichtigsten Definitionen eingegangen, um für den Leser eine gute Grundlage zu schaffen.
Es werden die verschiedenen Management-System-Architekturen[1] vorgestellt. Anhand eines Management-Szenarios sollen die Probleme, die beim Management entstehen können, verdeutlicht werden.
Aber zuerst folgt eine Erläuterung des Aufbaus einer Management-Plattform.

2.1 Allgemeiner Aufbau einer Plattform

Moderne Management-Plattformen sind heute beim Betreiben eines großen Netzes unerläßlich geworden.
Management-Plattformen stellen eine Architektur bereit, auf der verschiedene Management-Anwendungen ablaufen können. Im nächsten Absatz sind einige „Selbstverständlichkeiten“ von heutigen Produkten formuliert, die von den führenden Herstellern unterstützt werden.

- Die meisten Management-Plattformen laufen heute auf dem offenen Betriebssystem UNIX. Viele Hersteller gehen jedoch dazu über, ihre Management-Plattformen auf das proprietäre System *WindowsNT* zu portieren. Manche Software-Unternehmen gehen sogar soweit und bieten ihre Management-Lösung nur unter *WindowsNT* an.
- Alle Systeme bieten heute eine grafische Benutzeroberfläche. Hier werden verschiedene grafische Oberflächen unterstützt, wie z.B. *Windows* oder *X-Windows/Motif*. Teilweise werden noch textorientierte Oberflächen angeboten.
- Die Plattform sollte alle Ressourcen kontrollieren können, soweit die notwendigen Agenten vorhanden sind, unabhängig davon, ob es sich dabei um Komponenten oder um Software handelt. Theoretisch besteht keine Einschränkung beim Funktionsumfang, allerdings werden gewisse Grenzen durch Überschaubarkeit und Hardware-Ressourcen gesetzt.
- Die Systeme haben einen modularen Aufbau, so daß die Anwendung verteilt auf mehreren Maschinen installiert werden kann. Dadurch ergibt sich auch die Möglichkeit, Management-Aufgaben, z.B. Überwachung einer entfernten Domäne, an Domänen-Manager zu delegieren.
- Ressourcen werden als Objekte modelliert und die Zugriffe auf diese Objekte stellen die Management-Operationen dar.
- Die Management-Plattform stellt standardisierte und definierte Schnittstellen zur Verfügung. Mit diesen Schnittstellen können Informationen anderen Anwendungen zugänglich gemacht werden.

[1] dies sind Szenarien, die den Aufbau eines, eventuell verteilten, Management-Systems beschreiben

Zusammenfassend kann man die Aufgabe einer Management-Plattform in kurzen Worten wie folgt formulieren:
„Eine Managementstation ist ein Trägersystem, auf dem eine Management-Plattform ganz oder in Teilen installiert ist und das für das Management eines Netzes oder verteilten Systems eingesetzt wird“ ([HEAB93], S. 300).
Interessanter als die unterstützten Betriebssysteme oder die vorhandenen Schnittstellen, ist eigentlich die zur Verfügung stehende Funktionalität. Aus diesem Grund soll kurz auf den funktionalen Aufbau einer Management-Plattform eingegangen werden.

2.1.1 Funktionaler Aufbau von Management-Plattformen

Eine Management-Plattform ist in Bausteine aufgeteilt, die verschiedene Funktionen bereitstellen. Diese Bausteine sollen im folgenden beschrieben werden.
Ein Baustein unterteilt sich in einen Funktions- und Konfigurationsteil. Der Funktionsteil übernimmt die Ausführung der eigentlichen Aufgaben und der Konfigurationsteil ermöglicht die Anpassung der Dienste an die individuellen Bedürfnisse des Anwenders.

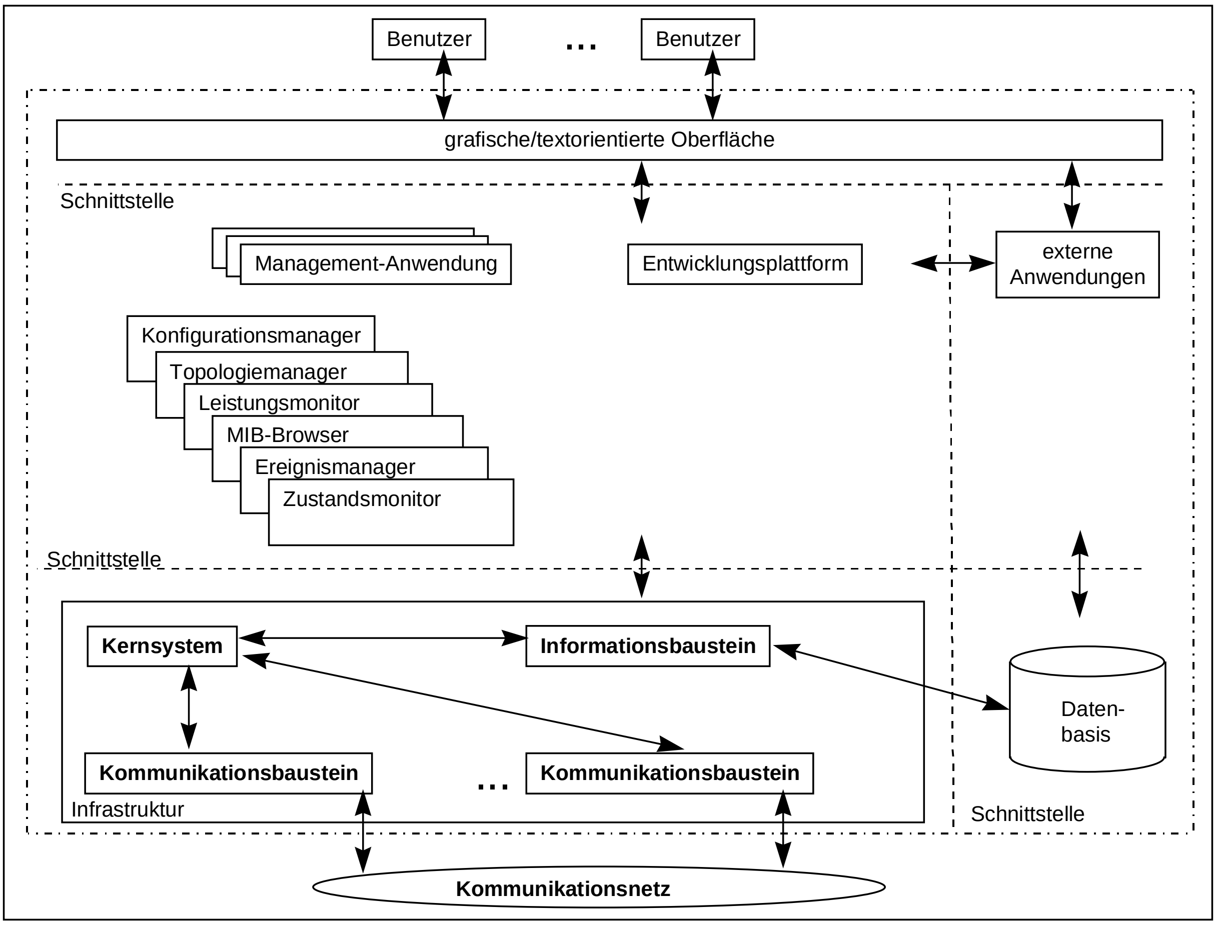

Abbildung 2.1: Funktionaler Aufbau einer Management-Plattform ([HEAB93], S. 301)

Die **Infrastruktur-Einheit** stellt mehrere Dienste zur Verfügung. Alle anderen Managementanwendungen können diese Grunddienste verwenden. Sie werden über eine standardisierte Schnittstelle angesprochen.

Der Baustein **Kernsystem** dient als Koordinator für die anderen Bausteine. Vom Kernsystem wird der Informationsfluß gesteuert, es synchronisiert die anderen Bausteine und überwacht deren Verfügbarkeit. Anfragen werden vom Kernsystem an den zuständigen Baustein weitergereicht.
Der **Kommunikationsbaustein** ermöglicht die Kommunikation mit anderen Systemen, z.B. Komponenten, Systeme, Software. Der Betreiber kann, entsprechend seiner verwendeten Management-Protokolle, Kommunikationsbausteine hinzufügen oder weglassen. Deshalb ist der Baustein in dieser Abbildung zweimal dargestellt.

„Der **Informationsbaustein** implementiert das objektorientierte Informationsmodell der Management-Plattform“ ([HEAB93], S. 302). Er verwaltet und erstellt die Information in einer Datenbasis und übernimmt die Kommunikation mit dieser.

Die **Datenbasis** ist der „Container“ der Management-Plattform für alle Informationen, die über die Ressourcen gesammelt werden. Die Datenbasis wurde in Abbildung 2.1 bewußt außerhalb der Infrastruktur dargestellt. Die Hersteller heutiger Management-Plattformen verwenden die unterschiedlichsten Techniken, um die Management-Informationen zu speichern.
Einige schreiben die Informationen in ASCII-Files auf der Managementstation. Andere verwenden spezielle objektorientierte Datenbanken, die in die Management-Plattform integriert sind. Wieder andere benutzen als Informationsbaustein ein SQL-Anfragesystem und als Datenbank eine relationale Datenbank, die separat erhältlich ist.
Manche Hersteller habe zwei verschiedene Schnittstellen zu ihrer Datenbasis. Eine Möglichkeit bietet der Informationsbaustein, der einer anderen Anwendung immer wieder die gleiche Information bei einer bestimmten Anfrage zur Verfügung stellt. Wird die interne Struktur der Datenbasis geändert, muß der Informationsbaustein eventuell angepaßt werden.
Manchmal können externe Anwendungen direkt mit einer Anfragesprache auf die Datenbasis zugreifen. Somit ist sicherlich eine schnellere Abarbeitung der Anfragen möglich. Bei einer Änderung der Datenbank-Struktur müssen die Anfragen der Anwendung angepaßt werden. Beim Informationsbaustein übernimmt die Anpassung der Hersteller.

Der **Oberflächenbaustein** ist die Schnittstelle für den Benutzer und stellt das Netz grafisch dar. Andere Management-Anwendungen sollten möglichst in der gleichen Oberfläche integriert sein. Moderne Management-Plattformen sollten auch eine textorientierte Schnittstelle zur Verfügung stellen, die natürlich nicht den gleichen Komfort wie eine grafische bieten kann. Dennoch ist sie dazu geeignet, einfache Aufgaben im Batchbetrieb auszuführen.
Zwischen den Oberflächen und den Management-Anwendungen sollte sich eine standardisierte Schnittstelle befinden. Damit soll die Einbindung von unabhängigen Management-Anwendungen in die grafische Oberfläche vereinfacht werden. Manche Management-Anwendungen haben eine eigene Schnittstelle zu externen Anwendungen. Genauso ist aber auch eine einheitliche Schnittstelle zu externen Anwendungen denkbar, durch die ein Datenaustausch möglich ist.

Die **Entwicklungsplattform** ermöglicht die Erweiterung einer bestehenden Management-Plattform um neue Hilfsmittel. Es können neue Anwendungen erstellt werden, die ausgewählte Management-Informationen benutzen oder präsentieren.
Die meisten Management-Anwendungen haben nur die Funktion der Überwachung und grafischen Aufbereitung von Informationen. Aber eigentlich wird nach einer Lösung gesucht, die vorhandene Probleme interpretiert und vorverarbeitet.

Mit der Entwicklungsumgebung wird dem Betreiber ein Werkzeug gegeben, mit dem sich eigene Anwendungen programmieren lassen. So können eventuell für spezifische Problembereiche eigene Lösungen entwickelt werden.

Mit diesen Erläuterungen sollte der schematische Aufbau einer Management-Plattform deutlich sein.
Aber wie sieht der heutige Markt tatsächlich aus? Wie geht die Entwicklung von Management-Plattformen weiter? Welche Management-System-Architekturen gibt es?

2.2 Entwicklungen am Markt

Eine andere Möglichkeit der Betrachtung stellt dieses Kapitel dar. Vor einigen Jahren gab es verschiedene Bereiche im Management, die durch unterschiedliche Produkte abgedeckt wurden. Früher gab es eine Teilung der Funktionalität von Management-Systemen in folgende Bereiche ([DATA95], S. 5010/1):

- Komponenten-Management
- Integriertes Management
- Enterprise-Management
- LAN-Management
- WAN-Management
- Manager-Management

Heute beginnen die Grenzen zwischen diesen Funktionsbereichen zu verschwimmen. Eine wirkliche Trennung zwischen z.B. LAN- und WAN-Management zu vollziehen, ist schwierig. Damit die Unternehmensnetze von heute kontrolliert werden können, ist eine Mischung aus verschiedenen Techniken nötig. Einer der wichtigsten Entwicklungen in den letzten Jahren ist das Zusammenwachsen der Infrastrukturen von LANs, Internets, WANs und Sprachnetzen. Ursprünglich gab es eine strikte Trennung, doch heute kann z.B. die Telefonie und der LAN-Verkehr die gleiche Verkabelung verwenden. Dies bedingt eine neue Generation von Geräten, wie z.B. intelligente Hubs oder die Multifunktionalität von Routern und Brücken.
Diese Multifunktionalität der Geräte ist ein weiterer Grund für die schwierige Trennung der oben genannten Bereiche. Damit die Produkte den Anschein von übergreifender Funktionalität erweckten, wurden zunehmend die Begriffe „integriert“ und „Enterprise“ eingesetzt.
Deshalb ist eine Definition der Begriffe nötig und wird im nächsten Abschnitt beschrieben.

2.2.1 Integriertes oder Enterprise-Management

Mit den fallenden Kosten für Rechenleistung, transparente Verbindungen zu anderen Arbeitsgruppen und Zugriff auf entfernte Unternehmens-Ressourcen hat sich das Netz-Design von hierarchischen zu verteilten Peer-to-Peer-Konfigurationen gewandelt. Das Überwachen von Netzkoppelelementen wird durch diese neuen Anforderungen zur zentralen Bedeutung und ist ein wichtiger Faktor im Enterprise und im verteilten Management. Bei der vollständigen Integration wird das physikalische und logische Management von Netz-Ressourcen in einer Anwendung zusammengefaßt. Dabei wird mit verteilter Information und Korrelation von Ereignissen über mehrere Objekt-Mengen gearbeitet.

Auch die Hersteller gehen neue Wege, um bei den neuen Entwicklungen mithalten zu können. Hersteller von Management-Plattformen ermöglichen Drittanbietern den Zugriff auf ihre Ma-

nagement-Informationsstruktur, so daß diese eigene Anwendungen entwickeln können. Im Gegenzug kaufen viele Plattform-Hersteller neue Management-Anwendungen hinzu, um den Aufwand für eigene Entwicklungen zu minimieren. Einige Hardware-Hersteller gehen dazu über, ihre proprietären Lösungen auf die Management-Plattform von den führenden Herstellern zu portieren. Die Komponenten-Hersteller fangen an, durch Partner-Lösungen gerätespezifische Funktionalitäten in bestehende Lösungen zu integrieren.
Dadurch verringert sich der Aufwand für die Hersteller, sie müssen nicht alle Management-Anwendungen selbst entwickeln. Durch Kooperationen sollen Produkte mit einem großen Umfang an Funktionalität entstehen, die den Ansprüchen der Anwender gewachsen sind.

Plattform-Integration bezieht sich auf das Management von LAN- und WAN-Komponenten von einer einzelnen Plattform aus. Die funktionale Integration bedeutet, daß Management-Information von verschiedenen Komponenten und verschiedenen Domänen koordiniert und bearbeitet werden kann. Zum Beispiel Help-Desk und Inventar-Kontrolle können beide über ein Trouble-Ticket-System (TTS) koordiniert werden ([DATA95], S. 5010/3).

Mit der steigenden Notwendigkeit von Enterprise-Management haben viele Hersteller von LAN- und WAN-Management-Lösungen die Anzahl der unterstützten Netzkoppelelement-Arten erhöht und die Fähigkeiten für Internetworking erweitert. Danach wurden die Produkte mit den Bezeichnungen „integriert oder „Enterprise“ beschrieben. Während integriert noch richtig sein mag, da einige Systeme LANs und WANs betreuen können, so bedeutet Enterprise-Netz nicht nur integriert.
Es bedeutet, daß die Teilbereiche des Managements, wie Komponenten-, Anwendungs- und Systemmanagement, im Kontext des Unternehmens gesehen werden ([HEAB93], S.100).

Ein weiterer Begriff im heutigen Netzmanagement-Markt ist die Management-System-Architektur. Dieser Begriff beschreibt eine Lösung, wie die Management-Leistung durch verteilte Funktionalität erbracht werden kann.

2.2.2 Management-System-Architekturen

Zu der Zeit, als sich das proprietäre Komponenten-Management profilierte, war es nicht ungewöhnlich, wenn es mehrere Management-Systeme gab. Jede Management-Plattform kontrollierte dann eine Gruppe von Geräten eines bestimmten Herstellers.
Der Netzadministrator mußte also mehrere Plattformen beobachten, um das Netz zu kontrollieren. Diese proprietären Lösungen waren in der Regel nicht an das jeweilige Unternehmen angepaßt, womit sich neue Probleme ergaben. Wenn z.B. eine Komponente einen Fehler generierte, erschien dieser auf mehreren Systemen. Der Administrator war dann damit beschäftigt, den tatsächlichen Ursprung des Fehlers zu ermitteln.

Der heutige Ansatz einer Management-Plattform, die viele Ressourcen zentral verwaltet, reicht für große, weit verteilte Netze nicht mehr aus. Damit komplexe Umgebungen effizient kontrolliert werden können, müssen neue Ansätze gefunden werden.
Ein möglicher Weg sind verteilte Management-Szenarien. Die Management-Aufgaben zu verteilen, reicht aber oft nicht aus, denn die Management-Information sollte immer noch an einem zentralen Punkt verfügbar sein. Es ist deshalb mehr Skalierbarkeit nötig, damit eine Lösung für ein Unternehmen gefunden werden kann. Heute gibt es vier verschiedene Management-System-Architekturen ([LCPW89], [COLE96], [HEGE93]):

Die zentrale Architektur

Die zentrale Architektur ist die am häufigsten verwendete. Sie besteht aus einem zentralen, einzelnen Manager, der für das Management des gesamten Netzes verantwortlich ist. Der Manager betreibt die Kommunikation mit den Agenten der Netzkomponenten. Außerdem bietet er zentrale Entscheidungsunterstützung und unterhält eine Datenbank.
Diese klassische Architektur bietet alle Vor- und Nachteile von zentralen Systemen. Der größte Nachteil ist die Unfähigkeit, das Management-System einfach zu skalieren, wenn das Netz komplexer und größer wird. Da es nur einen „Single-Point-of-Failure" gibt, ist die Verfügbarkeit des Systems leichter zu garantieren. Es ist somit sehr einfach, ein vollständiges Ersatzsystem bereitzuhalten. Sind die Management-Aufgaben auf mehreren Systemen verteilt, so muß für jedes System ein Ersatz existieren. Leider sind dadurch keine Maßnahmen für Fehlertoleranz vorhanden. Ein weiterer Nachteil ist der erhöhte Bedarf an Datendurchsatz. Alle Netzknoten müssen mit der zentralen Managementstation kommunizieren, fällt die Leitung zu dieser Station aus, so ist auch keinerlei Management mehr möglich.
Allerdings wird in der Praxis diese Architektur oft den anderen vorgezogen. Die Architektur ist leicht auf ihre Funktionsfähigkeit zu überprüfen, leicht zu installieren und einfach zu handhaben. Ein Beispiel für so ein System ist die Management-Plattform *HP Openview*; wenn sie als einzelne Plattform installiert ist und alle Management-Aufgaben alleine löst.

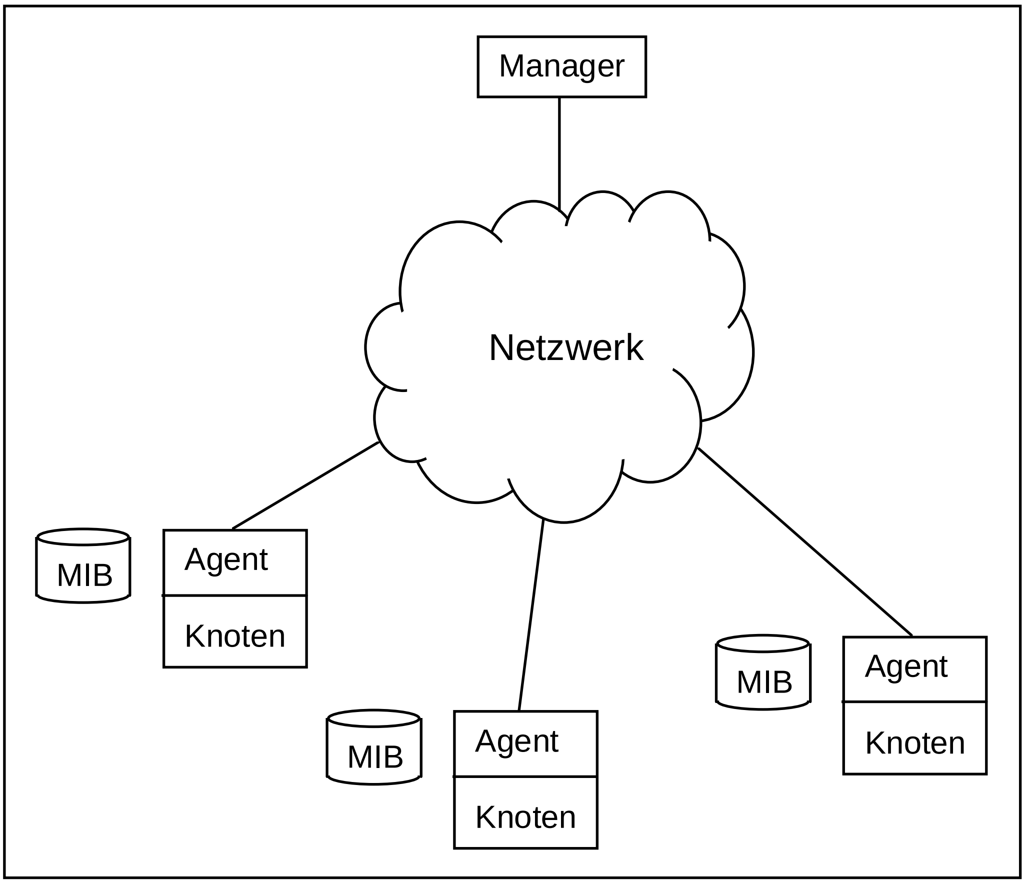

Abbildung 2.2: zentrale Architektur

Die verteilte Architektur

Die verteilte Architektur verwendet mehr als einen Peer-Manager. Diese Architektur ist sehr gut für mehrstufige Domänen-Netze geeignet, denn für jede Domäne ist ein Peer-Manager verantwortlich. Die Domänen-Bildung wird über Attribute der Management-Objekte realisiert. Es können ein oder mehrere Attribute zur Bildung einer Domäne verwendet werden. Denkbar wären z.B. geografische, funktionale oder organisatorische Attribute. Falls Management-Informationen von einer bestimmten Domäne benötigt werden, kommuniziert der Manager mit dem verantwortlichen Peer-Manager, um die benötigte Information zu erhalten. Skalierbarkeit ist der große Vorteil dieser Architektur. Hohe Anforderungen an die Leistung oder mögliche Erweiterungen des Netzes können durch Hinzufügen weiterer Domänen ermöglicht werden.
Ein Nachteil dieser Architektur ist die Steigerung des Kommunikationsaufkommens für Management-Aspekte. Muß ein Problem gelöst werden, das sich über mehrere Domänen erstreckt, so müssen die jeweiligen Peer-Manager intensiv Informationen austauschen, um das Problem zu lösen. Es müssen auch mehrere Manager bemüht werden, um ein übergreifendes Problem zu lösen. Der Administrator hat nicht nur einen Manager, den er konsultieren muß. Beim Ausfall eines Peer-Managers kann das Netz noch teilweise kontrolliert werden.
Eine Implementierung dieser Architektur in ein käufliches Produkt ist nicht bekannt, wahrscheinlich weil der Schritt zur hierarchischen Architektur (siehe später) nur einen geringen Entwicklungsaufwand bedeutet.

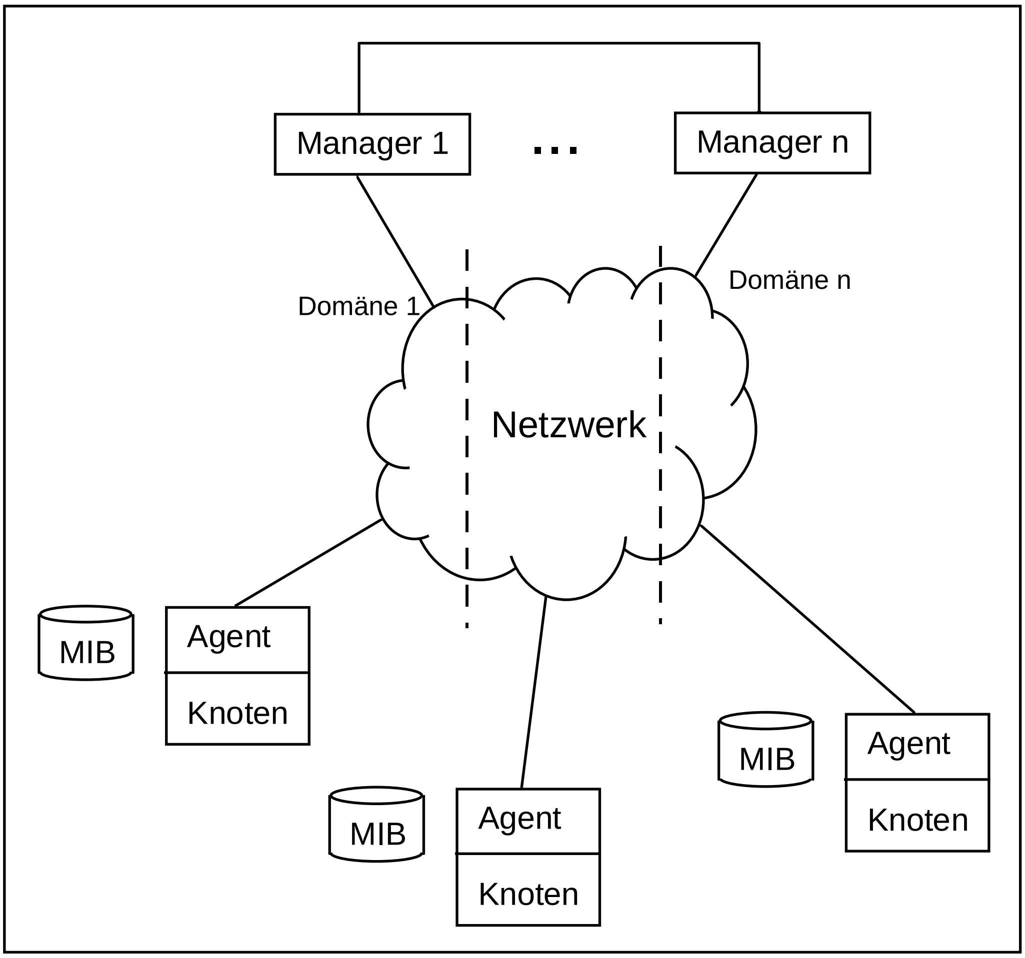

Abbildung 2.3: verteilte Architektur

Die hierarchische Architektur

Die hierarchische Architektur benutzt ebenfalls eine Aufteilung in Domänen für das Management. Jeder Domänen-Manager ist verantwortlich für das Management seiner Domänen, aber nicht für das restliche Netz. Weiterhin gibt es noch einen „Manager-of-Managers“ (MOM). Der MOM arbeitet auf einer höheren Hierarchiestufe und fordert Management-Informationen von den Domänen-Managern an. Aber im Gegensatz zur vorherigen Architektur kommunizieren die Domänen-Manager nicht untereinander. In dieser mußten die Peer-Manager Informationen austauschen und gegenseitig diese Information interpretieren, um ein übergreifendes Problem zu lösen.
Hier fordert der MOM die Informationen von den Domänen-Managern an, diese werden interpretiert und ergeben somit den Gesamtzustand des Netzes. Die hierarchische Architektur kann sehr leicht skaliert werden, und es können mehrere MOMs vorhanden sein. MOMs von MOMs können ebenfalls aufgebaut werden, um eine mehrstufige Architektur zu erhalten. Diese Architektur ermöglicht die Entwicklung von integrierten Anwendungen, die ihre Informationen von heterogenen Domänen erhalten. Durch die MOMs kann eine Schnittstelle für übergreifende Probleme zur Verfügung gestellt werden. Die MOMs sollten also immer Kenntnis über den Zustand ihrer Domäne haben. Damit müssen bei einem Problem nur die MOMs Zustandsinformationen weiterreichen.
Diese Architektur ist aber nicht so leicht zu installieren und zu betreiben wie die zentrale Architektur. Ein erheblicher Verbrauch an Hardware-Ressourcen und Konfigurationsarbeit ist nötig. Sollen bei einem hierarchischen MOM-Szenario heterogene Plattformen eingesetzt werden, so ist zu prüfen, ob die Produkte dafür geeignet sind.
Das Produkt *IBM NetView* bietet eine Möglichkeit einer hierarchischen Architektur mit gewissen Einschränkungen. Hier können Domänen-Manager eingerichtet werden, die eine bestimmte Domäne kontrollieren und bei Auftreten eines Problems eine Nachricht an einen MOM schikken. Allerdings können hier keine MOMs von MOMs aufgebaut werden. Die Möglichkeiten einer Interpretation der Informationen sind gering, dies bleibt weitestgehend dem Benutzer überlassen.

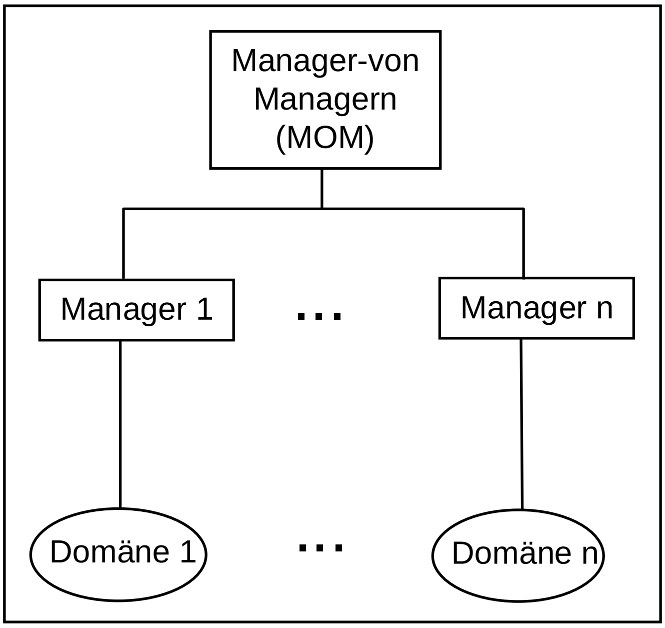

Abbildung 2.4: hierarchische Architektur

Die Netzwerk-Architektur

Die Netzwerk-Architektur ist eine Kombination aus der verteilten und hierarchischen Struktur. In dieser Architektur gibt es zwei Arten von Managern, den Element-Manager (vorher Domänen-Manager) und den Integrated-Manager (vorher MOM). Im Gegensatz zur verteilten oder hierarchischen Architektur sind die Manager in einer vernetzten Struktur verbunden. Durch diesen Aufbau bleibt die Skalierbarkeit vorhanden, die Funktionalitäten können besser verteilt werden und das Management kann auch in ungleichen, nicht-kanonischen Umgebungen durchgeführt werden. Wenn das Netz „wächst“, ist es oft schwierig, ein Management-System zu erweitern. Die Antwortzeiten der Management-Plattform steigen proportional mit der zu überwachenden Netzgröße. Ein weiterer Engpaß ist, daß die Anzahl der Anwendungen, die auf einer Management-Plattform laufen können, einer Grenze unterworfen sind.
Deshalb sollte jeder Manager die Form einer Management-Plattform haben, um eine möglichst hohe Skalierbarkeit zu erreichen. Jeder Manager bietet eine Menge von Management-Diensten an, diese können dann von Anwendungen oder anderen Managern genutzt werden [CPST95].

Der **Integrated-Manager** ist für eine Domäne verantwortlich und bietet detaillierte Informationen oder Zusammenfassungen von Informationen. In dieser Domäne können wiederum Manager enthalten sein.
Ein **Element-Manager** bietet ebenfalls detaillierte und zusammengefaßte Informationen über eine Domäne. Aber in dieser Domäne sind keine weiteren Manager enthalten. Er kann auch nicht für mehrere Domänen verantwortlich sein, im Gegensatz zum Integrated-Manager.
Ein **Anwendungs-Objekt** kann eine lokale oder eine integrierte Anwendung sein. Eine lokale Anwendung läuft auf einem einzelnen Manager. Die integrierte Anwendung verwendet die Dienste von mehreren Managern und ist somit verteilt. Produkte, die diese Architektur verfolgen, gibt es noch nicht zu kaufen. Bisher sind nur Forschungsimplementationen von [GOYE93] bekannt.

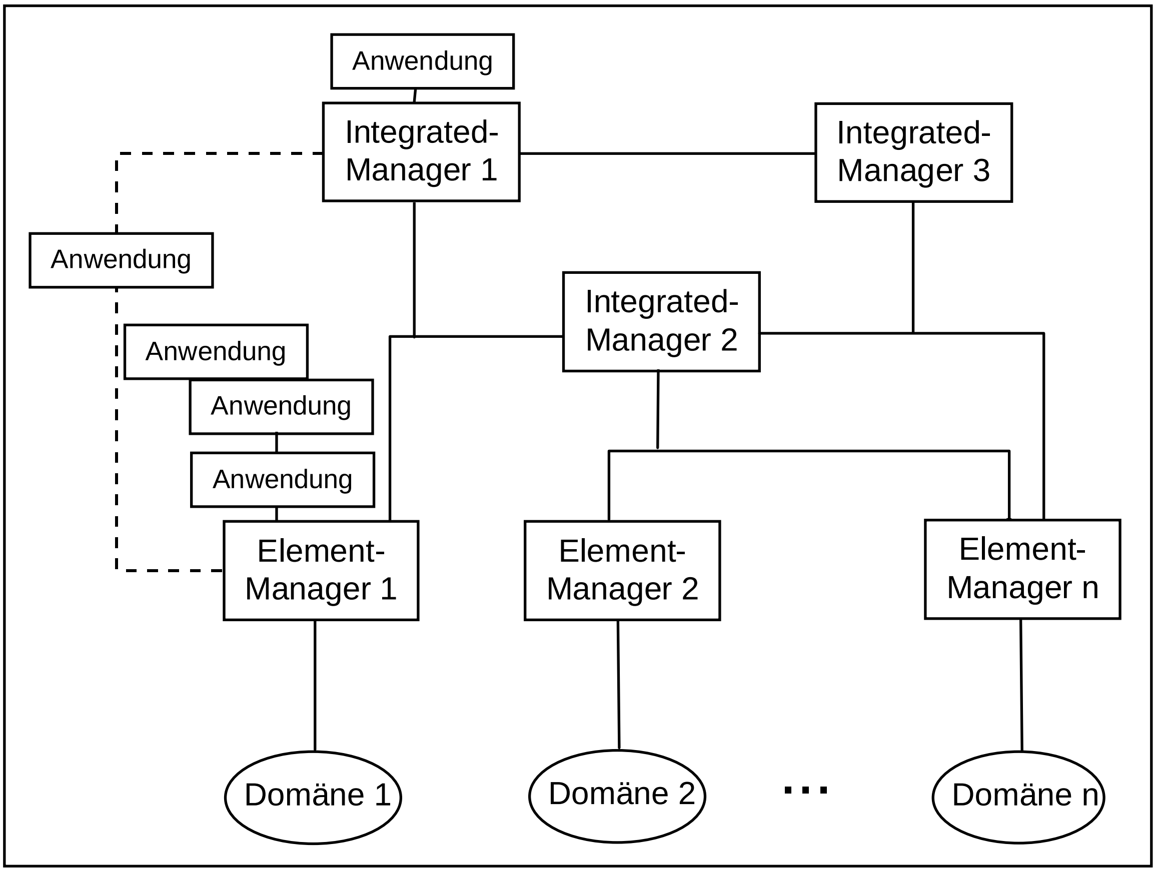

Abbildung 2.5: Netzwerk-Architektur

RMON-MIB

Ein andere Möglichkeit, Management-Funktionalität zu verteilen, ist die Remote Network Monitoring-MIB [RFC1271]. Diese MIB wurde im November 1991 veröffentlicht und erweitert das Management von LANs. RMON erlaubt das Überwachen eines Netzes mit Hilfe eines Zusatzgerätes, der Probe. Eine Probe (LAN-Monitor, LAN-Analysator, engl.: probe) ist ein Gerät, welches in das Netzsegment integriert wird und ständig Statistikdaten über das Netz und die darüber übertragenen Datenpakete sammelt. Diese Daten können zur Auswertung ständig oder nach Ablauf von Schwellwerten (engl.: thresholds) an das Management-System übertragen werden. Dem RMON-Standard ist eine eigene MIB zugeordnet, die RMON-MIB mit dem Eintrag {mib-2 16}. Die RMON-MIB (siehe Abbildung 2.6) gliedert sich in den Ethernet-Standard ([RFC1271]) und spezielle Erweiterungen für Token Ring ([RFC1513]).

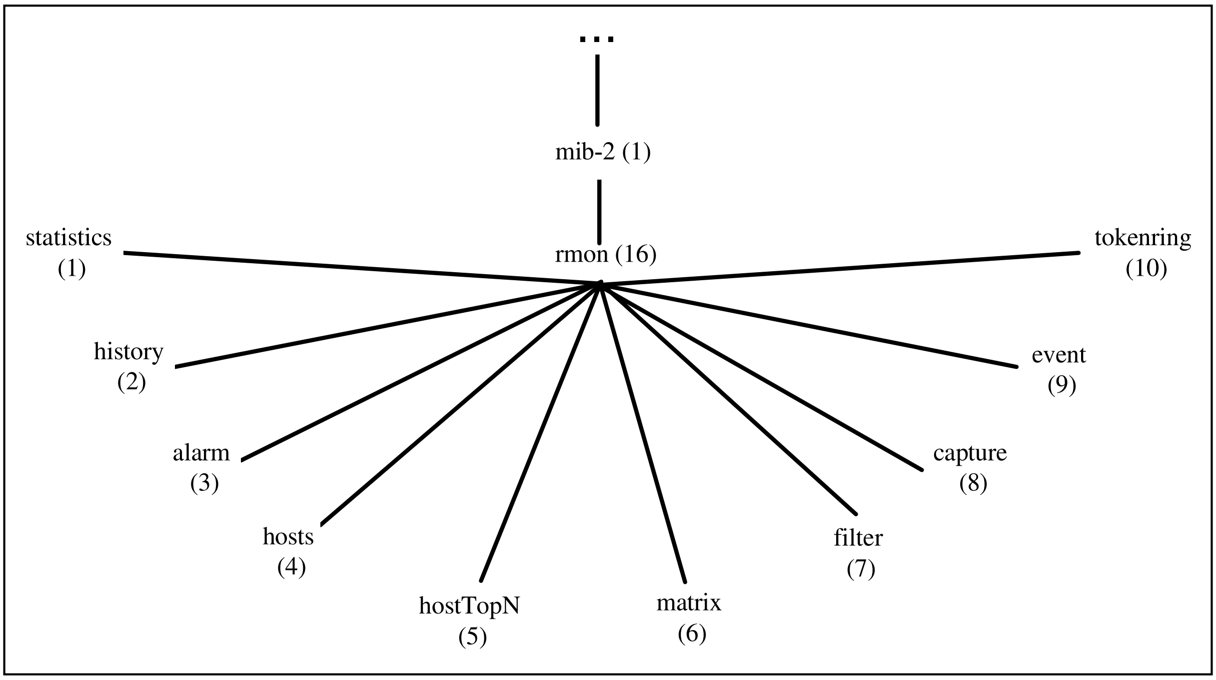

Abbildung 2.6: RMON-MIB

Damit die Möglichkeiten der RMON-MIB klar werden, werden die Erweiterungen 1-9 näher erläutert. Die **Statistics Group** liefert, wie der Name schon andeutet, Statistikdaten über das Netzsegment, wie Anzahl und Größe der übertragenen Pakete und Oktetts (Bytes), Anzahl der Multicasts, Broadcasts und Kollisionen. In der **History Group** werden Daten für historische Analysen der obigen Werte gesammelt. So lassen sich Trendanalysen für einen bestimmten Zeitraum erstellen. Die Beobachtung von Variablen der RMON-MIB und ihr Vergleich mit Schwellwerten wird über die **Alarm Group** realisiert. Die **Host Group** dient der Sammlung von statistischen Informationen (gesendete und empfangene Pakete) aller an dem LAN-Segment angeschlossenen Rechner. Für die Erstellung von Reports, z.B. einer Top10 der Rechner mit der höchsten Auslastung, werden Daten in der **HostTopN Group** gesammelt und nach den jeweiligen Kriterien sortiert. Die **Matrix Group** ist für die Messung des Datenaufkommens zwischen zwei Netzgeräten, angegeben durch ihre MAC[2]-Adressen, zuständig. Dazu erfolgt eine ständige Aktualisierung der Paketzähler. Mit Hilfe der **Filter Group** können über Filterausdrücke bestimmte Pakete aus dem Netzverkehr für eine genauere Untersuchung ausgewählt werden. Die Auswertung der gefilterten Pakete geschieht in der **Capture Group** (auch Packet Capture Group). In der **Event Group** sind die möglicherweise eintretenden Ereignisse bestimmten Aktionen zugeordnet, die der Agent dann auszuführen hat. So ist neben dem Monitoring von Ethernet und Token Ring inzwischen auch eine Analyse von FDDI-Ringen möglich. Oft ist die Funktionalität durch spezielle RMON-MIBs der Hersteller von Probes erweitert.

[2] Medium Access Control

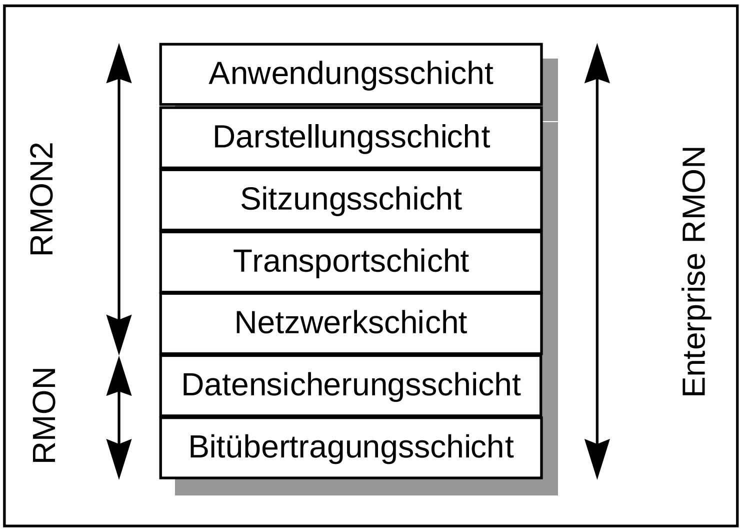

Abbildung 2.7: RMON und OSI

Die Standard-RMON erlaubt nur die Überwachung des Netzsegmentes auf der Physikalischen Schicht (MAC-Layer) des ISO/OSI-Referenzmodells. RMON-Probes können also nicht beobachten, was z.B. mit Protokollen (z.B. FTP ist unabhängig von Ethernet oder Token Ring) oberhalb von Schicht 2 geschieht. Mit dem noch nicht standardisierten RMON2 ist dagegen eine Sicht auf alle sieben Schichten möglich. RMON2 erlaubt somit auch das Kontrollieren der Anwendungsschicht (Schicht 7), womit eine Diagnose von Programmen wie *Microsoft Mail* oder *Sybase* möglich wird. Manche Hersteller haben RMON2 noch um die Möglichkeit der Überwachung einer Client/Server-Umgebung und der Unterstützung von WAN, FDDI, Fast Ethernet und ATM erweitert und bezeichnen dies als Enterprise RMON (oder RMON+). Eine Eingliederung von RMON, RMON2 und Enterprise RMON in das ISO/OSI-Referenzmodell zeigt die Abbildung 2.7.

Die Funktionalität der RMON-MIB stellt eine Möglichkeit zur Verfügung, Management von Ressourcen verteilt zu erbringen. Die gesammelte Management-Information sollte auch in einer Datenbasis gespeichert werden.
Deshalb kann durch neue Entwicklungen bei der Datenbasis die Funktionalität verbessert werden. In diesem Bereich hat sich in letzter Zeit etwas getan, was im folgenden Abschnitt kurz erläutert wird.

2.2.3 Objektorientierte Techniken

Die Computer-Industrie und das Netzmanagement setzen vermehrt auf objektorientierte Techniken. Objektorientierte Datenbanken und Programmiertechniken werden zum Standard der Datenbasis im Management verwendet, um die neuen Netzmanagement-Systeme zu erstellen. Der Vorteil für den Benutzer sind leistungsstärkere und stabilere Management-Systeme, in denen Objekte besser und realitätsnäher dargestellt werden können ([DATA95], S. 5010/5).
Die meisten Hersteller haben eine Entwicklung von einfachen ASCII-Dateien über relationale Datenbanken zu objektorientierten Datenbanken durchlaufen.

Aber zu einer Marktübersicht gehören auch bereits existierende Werkzeuge zur Auswahl von Plattformen. Es gibt bereits solche Hilfsmittel, die teilweise etwas älter sind, aber trotzdem gute Ansätze bieten. Deshalb ein Überblick der Werkzeuge im folgenden.

2.3 Vergleichs-Werkzeuge für Management-Plattformen

Die Auswahl der richtigen Plattform gestaltet sich sehr schwierig. Darum haben auch andere Institutionen die Problematik erkannt und geben Hilfestellung bei der Wahl des Management-Systems.
In diesem Abschnitt werden kurz die bestehenden Auswahlverfahren für Management-Plattformen vorgestellt und kritisiert.

2.3.1 Gesellschaft für Mathematik und Datenverarbeitung mbH

Beschreibung

Die Gesellschaft für Mathematik und Datenverarbeitung mbH (im weiteren GMD genannt) hat zu diesem Thema ein Arbeitspapier mit dem Titel „Ein Vergleich von kommerziellen Netzwerk-Management-Systemen“ veröffentlicht [GMD800].
In diesem Arbeitspapier wurden die drei Plattformen *HP OpenView*, *SunNet Manager*, *DECmcc* verglichen. Allerdings sind die getesteten Produkte sehr umfangreich, deshalb wäre ein Vergleich aller Komponenten zu zeitaufwendig, heißt es in der Einleitung. Deshalb beschränkte sich der Test auf den Bereich Management von IP-Routern mit SNMP. Dies sind auch die Hauptanwendungen der GMD.
Im zweiten Kapitel werden die einzelnen Systeme vorgestellt. Dabei wird die Aufteilung der Software in Module erklärt und die zur Verfügung stehenden Protokolle aufgezählt. Der Aufbau und das Zusammenspiel der Komponenten wird mit Hilfe einer Zeichnung erläutert.
Beim Produkt von HP ist eine Tabelle vorhanden, in der alle Komponenten und Module aufgelistet sind. Man kann ablesen, welche Komponenten in welchen Moduln enthalten, verfügbar oder nicht vorhanden sind.
Im nächsten Kapitel werden die Produkte mit folgenden Informationen genauer spezifiziert:

- Produktname
- Betriebssystem
- Version/Patch
- Maschine

Danach werden hier noch kurz die Router beschrieben, die in der Testumgebung zur Verfügung standen. Außerdem wird explizit darauf hingewiesen, daß die gestellten Anforderungen die Leistungsfähigkeit der Produkte bezüglich dem Management von Routern testen sollen. Bei anderen Anforderungen können auch andere Ergebnisse entstehen.

Im Kapitel vier - Systemkonzepte und -komponenten - werden die Werkzeuge untersucht, die nach Meinung der Autoren dazu dienen: „Das überwachte System in allen seinen Facetten zu definieren und dokumentieren.“
Einen eigenen Schwerpunkt setzten sich die Autoren in den Bereichen:

- grafische Oberfläche
- Dokumentation
- Systemerweiterungen
- Applikationen

Die Kriterien wurden dann teilweise durch das Studium der Handbücher bewertet oder, wenn es vom Aufwand vertretbar war, mit einem Beispiel getestet. Dieses Kapitel wurde dann in folgende Gliederungspunkte unterteilt:

- Netzkarten, Netztopologie
    - Darstellung
    - Funktionen
    - Handhabung
- Datenbank
    - Aufbau
    - Struktur
    - Funktionen
- MIB
    - Format
    - MIB-Browser
    - Erweiterungen
- Dokumentation
    - Handbücher
    - Programmdokumentation
    - Beispiele
    - Online-Hilfe
- Kommandoschnittstelle
    - Schnittstelle
    - Benutzerkommandos

In den obigen Punkten sind dann die eigentlichen Kriterien, die bewertet wurden. Die Anforderungen selbst sind nur als Stichwort beschrieben (z.B. Editieren). Nach dem Stichwort werden in Sätzen die Möglichkeiten der Plattformen bezüglich dieser Anforderung beschrieben.

Im Kapitel Management-Applikationen werden Anwendungen untersucht, die zur Bewältigung der Aufgaben in den verschiedenen funktionalen Bereichen zur Verfügung stehen. Die Schwerpunkte waren hier:

- Konfigurationsmanagement
- Fehlerbehandlung
- Leistungsüberwachung
- Grafik
- Sicherheitsaspekte
- Kostenabrechnung

Bei dieser Einteilung handelt es sich um die fünf Bereiche des OSI-Funktionsmodells [ISO74984], das um den Punkt Grafik erweitert wurde. Die Beschreibungen zu diesen Punkten entsprechen auch den Erwartungen, die von OSI bekannt sind. Im Gliederungspunkt Grafik wurde die Funktionalität bezüglich einer Darstellung von Meßwerten geprüft. Die Kriterien wurden wieder als Stichwort angegeben.

Im letzten Kapitel wurden die vorherigen Ergebnisse zusammengefaßt. Als erstes dient dazu eine Tabelle, in der alle aufgestellten Kriterien unter den zehn Gliederungspunkten zusammengefaßt wurden. Diese einzelnen Anforderungen wurden mit den vier möglichen Werten beurteilt:

- sehr gut
- gut
- ausreichend
- mangelhaft

Zu jeder Plattform wurde also eine Bewertung je Kriterium durchgeführt. Nach dieser Tabelle wurde zu jeder Anforderung eine Begründung für die jeweilige Bewertung gegeben. Die Autoren traffen sogar eine Auswahl, nämlich *HP OpenView*, denn es schien für die Aufgaben in der GMD besonders gut geeignet.
Als Zusammenfassung der Bewertung wurden noch die Stärken und Schwächen der drei Systeme hervorgehoben. Auf den letzten Seiten gaben die Autoren noch einige Hinweise und Erfahrungen, welche Probleme bei der Einführung eines Management-Systems auftreten können.

Beurteilung

Die Autoren beschränken sich von Anfang an auf die Beurteilung von drei konkreten Produkten. Die aufgestellten Kriterien müssen also immer im Kontext zu den verwendeten Produkten und den Themenkreis gesehen werden. Gut dabei ist, daß die Autoren konkret auf diesen Nachteil hinweisen. Vielleicht sollte dieser Kriterienkatalog auch nur für die Auswahl einer Plattform bei der GMD dienen (darauf wird aber nicht hingewiesen) und somit kein allgemeines Entscheidungs-Werkzeug sein.

Sehr gut ist die Beschreibung der Management-Plattformen im zweiten Kapitel. Besonders die Grafiken zum Aufbau ermöglichen einen guten Einblick in die Struktur der Anwendung. Die Tabelle bei *HP OpenView* (siehe oben) macht deutlich, welche Module nötig sind, damit bestimmte Aufgaben gelöst werden können. Warum dieses sehr gute Prinzip nicht für alle Produkte eingeführt wurde, ist unklar.

Die Beschreibung der Testumgebung ist etwas knapp ausgefallen. Die einzelnen Produktinformationen könnten um einige Informationen, wie z.B. Minimalanforderungen an Festplattenspeicher oder Hauptspeicher, erweitert werden. Weiterhin heiß es: „Die Produkte wurden im Rahmen der in der GMD anfallenden Netzwerk-Management-Aufgaben getestet.“ Leider werden die Aufgaben aber nicht näher spezifiziert. Welche Probleme müssen im Umfang dieser Aufgaben konkret gelöst werden? Die Autoren schreiben, daß sie z.B. die Router im GMD-Ebone-Knoten für den Test verwenden. Interessanter wäre aber, um welche Router es sich handelt. Wie sind diese Geräte konfiguriert?
Gerade die Beschreibung der Testumgebung hätte ausführlicher sein müssen. Dann wäre vielleicht auch die gewählte Struktur der Anforderungen verständlicher.

Die fünf Gliederungspunkte vom vierten Kapitel sind gut motiviert und erläutert. Warum aber gerade diese Punkte verwendet wurden und nicht mehrere oder andere, wurde nicht erklärt. Warum sind z.B. keine Kriterien zum Überwachen von Schwellwerten vorhanden?

Die einzelnen Anforderungen unter den Gliederungspunkten prüfen die jeweilige Funktionalität gut ab. Allerdings ist oft nicht verständlich, was man unter dem jeweiligen Stichwort zu verstehen hat. Erst durch die Anwendung des Kriteriums auf die Systeme wird die Semantik des Stichworts klar. Die einzelnen Anforderungen sollten noch mit einer kurzen Beschreibung erläutert werden.
Die Erfüllung der Kriterien wurde in einem fortlaufenden Text beschrieben. Ein Vergleich in der Form Produkt A ist besser als Produkt B gibt es in diesem Kapitel noch nicht.

Im fünften Kapitel wurde zusätzlich zu den OSI-Bereichen der Gliederungspunkt Grafik eingefügt. Die Begründung für diesen Punkt fehlt leider völlig. Der Mangel bei den Anforderungen (siehe vorheriger Absatz) besteht natürlich auch in diesem Kapitel. Manche Bereiche sind auch zu sehr auf einen Teilbereich reduziert, denn beim Konfigurationsmanagement z.B. wurden nur Kriterien im Bereich vom Autodiscovery untersucht. Denkbar wären hier auch Anforderungen zum Ändern der Konfiguration in Ressourcen.

Die Tabelle im letzten Kapitel gibt einen sehr guten Überblick über die erreichten Leistungen der Produkte. Wie die Autoren zu dieser Bewertung kommen, etwa durch Abstimmung oder durch Festsetzung, wird aber nicht deutlich.
In der anschließenden Erklärung wurden kurz die Vorteile des jeweiligen Besten beschrieben. Bei der Netzkarte wurde z.B. die einfache Bedienung von *HP OpenView* hervorgehoben. Wie die Autoren feststellten, daß die Bedienung sehr einfach ist, wurde nicht erklärt. Sehr gut ist die nachfolgende Zusammenfassung der Bewertung, in der eine Entscheidung gefällt wird. Hier begründet man auch, warum man sich dafür entschlossen hat.
Der Abschnitt „Stärken und Schwächen der Systeme“ gibt einen Überblick über den Stand der Technik. Dieser Ausblick zeigt sehr gut, wo noch Entwicklungsbedarf durch die Hersteller besteht. Somit kann man durch einen Vergleich mit diesem Abschnitt sehr schnell erkennen, ob ein Produkt dem aktuellen Stand der Technik entspricht. Ein Beispiel: „Was bei heutigen Systemen fehlt, ist die Überwachung von dynamischen Routing-Protokollen.“
Zum Schluß sind noch sehr gute Tips vorhanden, die man bei der Einführung einer Management-Plattform beachten sollte. Hier wird auch deutlich, daß die Autoren Erfahrungen aus ihrer eigenen Praxis in das Arbeitspapier einfliessen ließen.

Die einzelnen Kriterien sind sehr gut und praxisnah, deshalb werden sie auch in dieser Diplomarbeit verwendet. Da sich die Gliederung größtenteils nicht nachvollziehen läßt, ist die Struktur für eine Verwendung nicht geeignet.

2.3.2 Leibniz-Rechenzentrum

Beschreibung

Die Studie vom Leibniz-Rechenzentrum (im weiteren LRZ genannt) ist eine Untersuchung von Management-Plattformen unter UNIX [VALT91]. Diese Untersuchung zielt auf eine mögliche Einsatzfähigkeit der Produkte am LRZ und im Bayerischen Hochschulnetz (BHN) ab.

Im ersten Kapitel, der Einleitung, wurden kurz die Plattformen genannt, die nicht untersucht werden konnten. Die Begründung dafür ist ebenfalls vorhanden.

Im zweiten Kapitel beschreibt der Autor die Kriterien, die für die Untersuchung verwendet wurden. Dazu wurden die Anforderungen in folgende Teilbereiche gegliedert:

- Ablaufumgebung
- Kernsystem
- Anwendungfunktionalität
- Oberfläche
- Entwicklungsumgebung
- Sonstiges
- Einsatzerfahrung
- Urteil

Diese Teilbereiche werden dann noch weiter untergliedert. Erst unter den Gliederungspunkten der Teilbereiche befinden sich die eigentlichen Kriterien. Zur Verdeutlichung folgendes Beispiel:

- Ablaufumgebung
  - Produkt
  - Hardware/Software
  - Betriebsmittelbedarf
    Plattenplatz
    Arbeitsspeicherausbau

Die einzelnen Teilbereiche wurden also wie folgt gegliedert:

- Ablaufumgebung (sieh oben)
- Kernsystem
  - Kurzbeschreibung
  - Unterstützte Protokolle
  - Unterstützte Objektkataloge
  - Erweiterbarkeit der Objektkataloge
  - Verteilung
  - Anbindung an fremde Management-Systeme
  - Anbindung an andere Systeme
  - Datenbasis
- Anwendungsfunktionalität
  - Abfragen von Variablen
  - Monitoring
  - Langzeitmessung
  - Schreibender Zugriff auf Objekte
  - Ereignismanagement
  - Trap-Behandlung
  - Zustandsüberwachung
  - Fehlermanagement
  - Anwählen von Komponenten
  - Konfigurationsanwendungen
  - Spezielle Anwendungen für bestimmte Komponenten
  - Sonstige Anwendungen
  - Integration zusätzlicher Anwendungen
- Oberfläche
  - Umgebung
  - Aufbau und Verwaltung von Maps
  - Hierarchiestufen

    - Grafische Symbole
    - Realitätsnahe Darstellung
    - Visualisierung von Zuständen
    - Definition von Views
- Entwicklungsumgebung

Eine Besonderheit stellen die Teilbereiche Sonstiges, Einsatzerfahrung und Urteil dar. Im Bereich Sonstiges finden sich noch folgende Gliederungspunkte:
- Sicherheit
- Zeitaufwand
- Verfügbarkeit (hier ist der Kauf des Produkts gemeint)
- Kosten
- Künftige Entwicklungen

Unter Einsatzerfahrung wurden die Probleme und die Erfahrungen mit dem Umgang des Systems geschildert. Beim Urteil wurde kurz auf die Vor- und Nachteile im Kontext zu der jeweiligen Testumgebung (LRZ oder BHN) eingegangen.

Nach jedem Gliederungspunkt wurde eine Frage gestellt, um diesen Punkt näher zu erläutern. Die Kriterien sind dann als Stichworte formuliert. Bei manchen Gliederungspunkten (z.B. Unterstützte Objektkataloge) wurden genauere Erklärungen gegeben, sonst wäre die Bedeutung nicht klar gewesen.
Dieses Kapitel stellt also die Anforderungen in gegliederter Form dar und gibt Erläuterungen zu den einzelnen Kriterien.

Die Kapitel 3-6 sind dann die eigentliche Anwendung der Anforderungen auf die Produkte *HP OpenView*, *Networks Manager NMC 3000*, *SunNet Manager*, *Netlabs DualManager*. Dabei wurde in jedem dieser Kapitel wieder die Struktur vom Kapitel zwei aufgenommen. Statt der Kriterien, wie z.B. Plattenplatz, befinden sich die Angaben (z.B. 90 MB Festplattenplatz) zu den Produkten. Können manche Kriterien nicht erfüllt werden, so ist auch dies vermerkt. Oftmals lassen sich die Anforderungen (z.B. Ereignismanagement) nicht durch Angabe eine Zahl beantworten, dann wurde in freien Worten die Situation erklärt.

Kapitel 7 enthält eine Tabelle, in der verschiedene Eigenschaften aufgelistet sind und alle Management-Plattformen. Diese werden hier durch +, o, - bewertet. Ein Ausschnitt aus der Tabelle:

Eigenschaft	OpenView	NMC3000	SunNet Manager	DualManager
Version	2.0, Sept. 91	1.1, Mai 91	1.1, Mai 91	1.0, Jan. 91
MO-Kataloge	+	-	o	+
Managerkommunikation	-	-	?	-
....				

Im letzten Abschnitt wurden abschließende Bemerkungen zur Auswahl einer Plattform für das LRZ und BHN gemacht. Hier beschreibt der Autor kurz die nächsten Schritte, um eine endgültige Entscheidung für die beiden Einsatzgebiete zu treffen.

Bewertung

Der Übersichtlichkeit der Studie leidet durch das Fehlen eines Inhaltsverzeichnisses. In der Einleitung fehlt auch eine Aufzählung der Produkte, die tatsächlich getestet wurden. Erst durch Lesen der späteren Kapitel wird dies deutlich. Warum sich der Autor für diese Produkte entschieden hat, ist nicht klar. Die Plattformen, die getestet wurden, sollen bezüglich ihrer Tauglichkeit für den Einsatz im LRZ oder im BHN geprüft werden. Welche Aufgaben damit aber verbunden sind, vielleicht gibt es gewisse Schwerpunkte, wurde nicht erklärt.

Der eigentliche Kriterienkatalog im zweiten Kapitel macht dagegen einen sehr ausgereiften Eindruck. Die Notwendigkeit der Kriterien und die Stichworte dazu sind verständlich und nachvollziehbar. Leider besteht aber auch hier der Mangel, wie bei der Studie der GMD, die Gliederung der Kriterien wurde nicht näher erklärt. Warum befindet sich z.B. Sicherheit unter Sonstiges (siehe oben)? Der Anwender dieses Kataloges wird sich schwer tun, neue Anforderung in die vorhandene Struktur einzuordnen. Die Struktur ist aber insgesamt doch sehr gut. Da nur drei Gliederungsebenen vorhanden sind, verliert man auch nicht den Überblick im Kriterienkatalog.

Die nächsten Kapitel mit der Anwendung sind vorbildlich. Die Produkte werden gleich am Anfang genau spezifiziert. Durch die Beschreibung mancher Kriterien durch fließenden Text, können auch komplexe Bereiche verdeutlicht werden. Schade ist, daß der Autor die Möglichkeit nicht nutzt und zu abstrakten Kriterien (z.B. Realitätsnahe Darstellung) persönliche Eindrücke wiedergibt. Die Erfüllung der Kriterien wird hier auch nicht bewertet, sondern es wird nur beschrieben, ob und wie die Anforderungen erfüllt sind.
Interessant ist auch der Gliederungspunkt Einsatzerfahrung. Die Probleme, die dort aufgezählt sind, geben ein sehr gutes Bild über die Schwächen der Produkte in bestimmten Bereichen. Man kann daran erkennen, auf welche Management-Bereiche die Entwickler besonderen Wert gelegt haben.
Das Urteil zum Einsatz im LRZ oder im BHN wäre zwar interessant, aber durch die fehlende Aufgabenstellung in den Einsatzorten, kann das Urteil schwer abgeschätzt werden.

Im Kapitel 7 wurden in einer Vergleichs-Tabelle alle Anforderungen auf 16 Eigenschaften reduziert. Wie diese Eigenschaften entstanden sind, wurde nicht begründet. Was der Leser unter den Symbolen +,-,o,? zu verstehen hat, wurde nicht erklärt. Was ist besser - oder o? Hier hätte auf jeden Fall ein Zusammenhang zwischen den Eigenschaften und den Kriterien hergestellt werden müssen.

Das letzte Kapitel beschreibt den möglichen Einsatz im LRZ oder im BHN. Dies ist für den Leser uninteressant, wenn er nicht die Aufgabengebiete dieser Einsatzorte kennt. Dieses Kapitel scheint nur für den internen Gebrauch bestimmt.

Auch hier sind die Anforderungen sehr gut, obwohl leider Überlappungen mit der Studie des GMD bestehen. Einige Teilbereiche der Gliederungsstruktur wurden im Kriterienkatalog dieser Arbeit übernommen (z.B. Oberfläche). Leider kann das siebte Kapitel aufgrund seiner schlechten Beschreibung nicht verwendet werden.

2.3.3 SunWorld

Beschreibung

Ein Report zu diesem Thema kommt aus einem völlig anderen Bereich. Die Online-Publikation *SunWorld* hat einen Report mit dem Titel „Picking the right network management platform“ veröffentlicht [COOK97]. Dabei handelt es sich also nicht um ein unabhängiges Institut, sondern um eine Publikation von dem Unternehmen *Sun*.
Der Autor Rick Cook geht einen völlig anderen Weg, als die bisher vorgestellten Untersuchungen. Er beschreibt eine Checkliste, die beim Kauf einer Management-Plattform behilflich sein soll.

Dazu ist in der Einleitung eine kurze Übersicht über die heutige Marktsituation gegeben. Wie entwickeln die Hersteller ihre Management-Plattformen weiter und welche Trends sind zu erwarten?

Im Abschnitt „Differences are emphasis, not absolutes“ wurde beschrieben, daß die verschiedenen Hersteller verschiedene Schwerpunkte in ihren Produkten setzen. Außerdem wurde die Problematik erläutert, daß ein Produkt gefunden werden muß, welches möglichst viele Management-Aufgaben des Unternehmens abdeckt.

Das Teilstück „How to select a platform“ beschreibt die Fragen, die sich ein Betreiber beantworten muß, um eine Kauf-Entscheidung zu treffen.

Der nächste Abschnitt beschreibt die Problematik, daß die neue Management-Plattform mit der alten Management-Software zusammenarbeiten soll. Dazu erläutert der Autor die Möglichkeit der Hersteller, Schnittstellen zur Verfügung zu stellen oder Kooperationen mit Drittanbietern einzugehen.

Ein Kapitel ist den Vorteilen von WWW-basierten Schnittstellen gewidmet. Der Autor gibt einige Beispiele, wo solche Schnittstellen hilfreich sind.

Der letzte Abschnitt beschreibt einige Situationen, in denen das Outsourcing[3] aller Management-Aufgaben Vorteile bringt.

Bewertung

Die einzelnen Teilbereiche sind ohne eine erkennbare Ordnung. Der Autor liefert keine Bewertungsverfahren, um Produkte beurteilen zu können. Wie gut die Plattformen Anforderungen erfüllen, bleibt dem subjektiven Eindruck des Anwenders überlassen.
Die Sammlung der Kriterien ist leider sehr dürftig bei diesem Report. Der Autor kommt aus einem praktischen Umfeld, wie man an einigen Stellen erkennen kann. Spürbar ist auch, daß dieser Report durch Sun finanziert ist. So wird z.B. sehr oft auf die Erfüllung von Kriterien durch den *Solaris Enterprise Manager* hingewiesen. Dieser Report ist nur dafür geeignet, um beim Anwender das Bewußtsein für die Problematik beim Kauf einer Management-Plattform zu verdeutlichen. Brauchbare Anforderungen oder Bewertungsverfahren waren in dieser Entscheidungshilfe nicht zu finden.
Doch bevor man sich für eine Plattform entscheidet, muß sich das Unternehmen erst dazu entschließen, ein Management-System zu kaufen. Aber was veranlaßt einen Betreiber eines gro-

[3] Ist die zweckmäßige Aufteilung zwischen Eigen- und/oder Fremdleistung bei den Aufgaben der Informationsverarbeitung

ßen Netzes zum Einsatz einer Management-Plattform? Welche Gründe machen den Einsatz notwendig?
Auf diese Fragen wird im nächsten Abschnitt näher eingegangen.

2.4 Szenario eines Betreibers

Die Frage, wann eine Management-Plattform unerläßlich ist, ist schwierig zu beantworten. Man kann diesen Zeitpunkt nicht exakt bestimmen. Der Kauf einer Plattform ist ab einer bestimmten Zahl von Netzelementen nicht zwingend notwendig. Irgendwann ist die IT-Abteilung so überlastet, daß eine Management-Plattform angeschafft werden muß, um die anfallende Arbeit zu bewältigen.

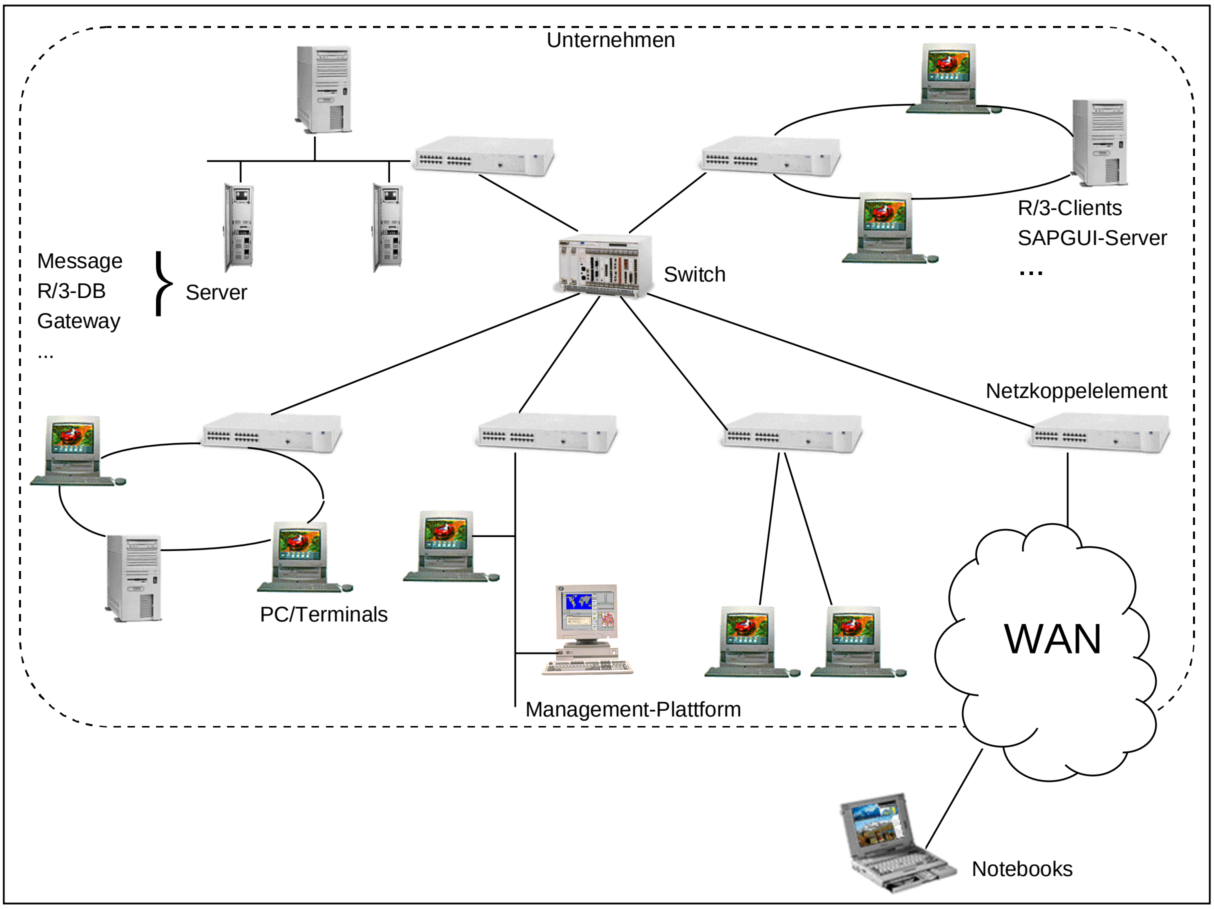

Abbildung 2.8: Infrastruktur komplexer Netze

In Abbildung 2.8 ist schematisch ein komplexes Netz dargestellt. Viele große Netze haben verschiedene Arten von Verkabelungen. Es gibt fast alle erhältlichen Netzkoppelelemente. Schnittstellen zu WAN-Strecken sind ebenfalls vorhanden. Diese Verbindungen müssen genauso wie die In-Haus-Verkabelung kontrolliert werden. Eine große Ausstattung an Hard- und Software ist bei solchen Unternehmen ebenfalls angeschafft worden.
Man kann aus dieser komplexen Struktur erkennen, daß der laufende Betrieb ein schwieriges Problem ist. Die Garantie für die Verfügbarkeit von so vielen Systemen läßt sich nicht nur mit Personal bewältigen.

Es müssen Hilfsmittel gefunden werden, um Aufgaben zu automatisieren. Wichtige Geschäftsprozesse müssen laufend überwacht werden, da heute viele Geschäftsprozesse in Anwendungen implementiert werden.
In der Abbildung sind außerdem Teile der betriebswirtschaftlichen Anwendung SAP *R/3* dargestellt. Diese Anwendung nimmt heute oft eine Schlüsselposition in deutschen Unternehmen ein. In einem *R/3*-System lassen sich Geschäftsprozesse nachbilden und durch den Einsatz von Rechnern automatisieren.
Durch die besondere Struktur dieser Anwendung und durch die fast 100 % gewünschte Verfügbarkeit ergeben sich noch zusätzliche Anforderungen an das Management. Durch die Verteilung der Anwendung auf mehrere Rechner, müssen verschiedene Kontrollpunkte überwacht werden. Diese Information muß durch Korrelation ausgewertet werden. Erst dann ist eine Aussage über den Zustand der Anwendung möglich.
Deshalb soll im nächsten Abschnitt die entstehende Problematik gezeigt werden, die sich beim Management eines *R/3*-Systems ergibt. Dadurch bekommt der Leser einen Eindruck, wie schwierig die Wahl der Kontrollpunkte ist.

Mit diesem Kapitel soll auch verdeutlicht werden, daß sich das Management auf die Überwachung von Anwendungen konzentrieren sollte. Die einzelnen Anwendungen bieten schließlich die Unterstützung für die Mitarbeiter, um die anfallenden Geschäftsprozesse durchzuführen.

2.4.1 Die Client/Server-Architektur von *R/3*

Das Produkt *R/3* der SAP bietet betriebswirtschaftliche Standardlösungen für die gesamte Informationsverarbeitung einer Firma. Das System besteht aus mehreren Modulen, wie z.B. Produktionsplanung (PP), Materialwirtschaft (MM), Personalwirtschaft (HR) und einigen mehr. Neben den Standardlösungen können mit der Entwicklungsumgebung und deren 4GL-Programmiersprache (*ABAP/4*) eigene Erweiterungen erstellt werden. Ohne eigene Daten enthält die Datenbank bereits 4,5 bis 6,5 Gigabyte Daten.

Informationssysteme lassen sich aus hardware-orientierter Sicht in drei Ebenen unterteilen. Dabei sind die Aufgaben, die von den einzelnen Komponenten ausgeführt werden, das Kriterium für die Aufgliederung[4]:

- Präsentationsebene
  In dieser Ebene kommen Präsentationsrechner zum Einsatz, diese stellen mit einer Präsentationsschnittstelle (bei *R/3* SAPGUI genannt, wie SAP Graphical User Interface) die interaktive Verbindung zum Benutzer her. In erster Linie müssen dafür PC oder Workstations benutzt werden, die die geforderte Qualität für die Darstellung der SAP-Oberfläche bieten können. Durch die Präsentationsschnittstelle wird sichergestellt, daß es keinen funktionalen Unterschied in der Dialogführung durch unterschiedliche „Front-Ends“ gibt.

- Anwendungsebene
  In der Anwendungsebene sind die betriebswirtschaftlichen Funktionen und Prozesse im *R/3*-System realisiert. Hier werden auch die Aktionen koordiniert, die für die Benutzersitzung und die Funktionstüchtigkeit des *R/3*-Systems notwendig sind. Im *R/3* übernimmt der Applikations-Server diese Aufgaben, indem auf ihn einer oder mehrere Prozesse laufen.

[4] Siehe Abbildung 2.9

- Datenhaltungsebene
  Das *R/3*-System beinhaltet viele spezielle, umfassende Datenstrukturen. Letztendlich werden diese in SQL und somit in einfache, relationale Tabellen umgesetzt. Diese Tabellen werden auf dem Datenbank-Server einer handelsüblichen Datenbank gehalten.

Das *R/3* gliedert sich ebenfalls in diese Ebenen. Durch genaue Definition von Schnittstellen, die den Datenaustausch zwischen den einzelnen Ebenen regeln, ist es möglich, verschiedene Ausprägungen der Architektur zu wählen.

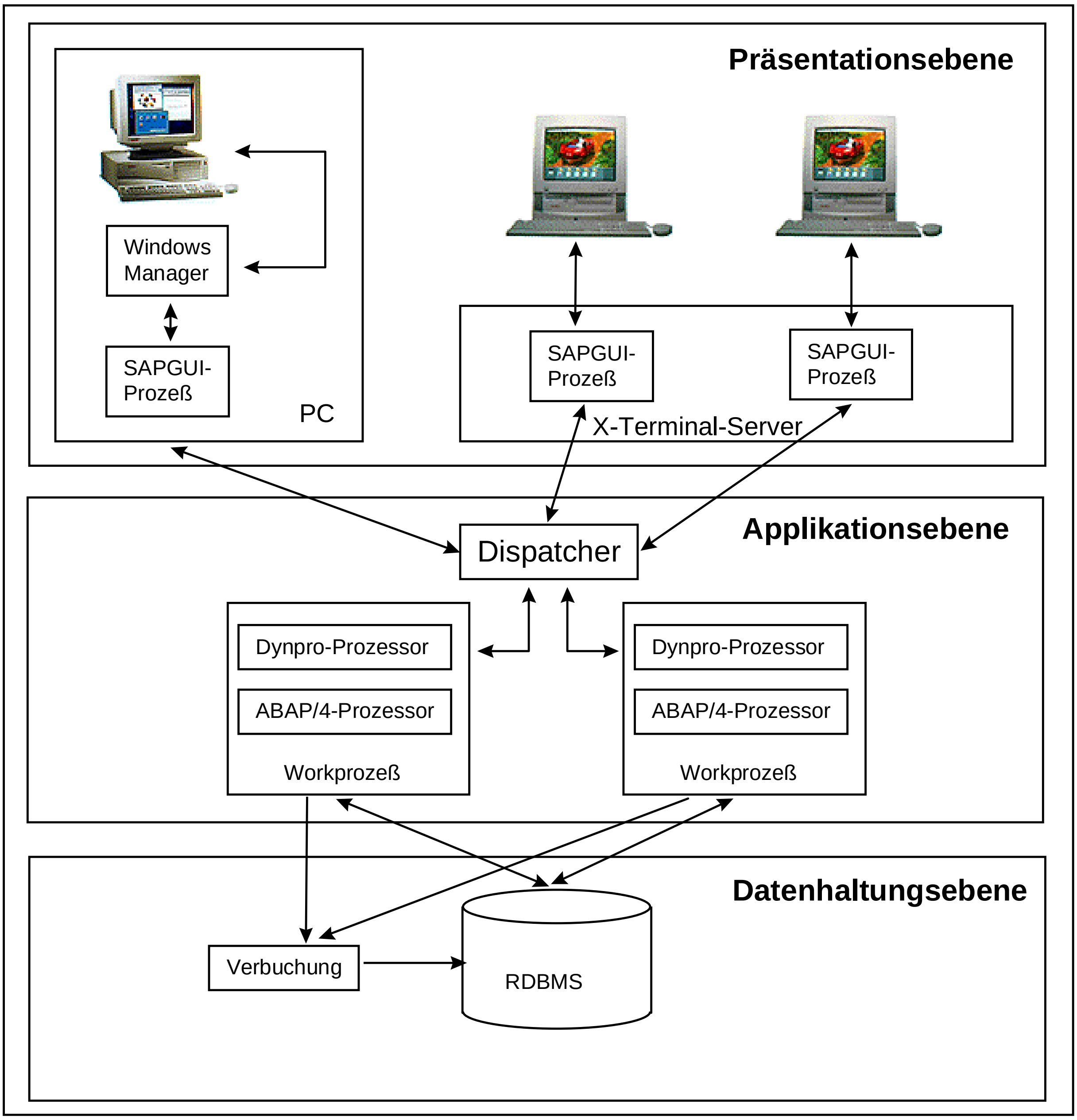

Abbildung 2.9: Drei-Ebenen-Struktur von *R/3*

In einer dreistufigen Client/Server-Architektur ist jede Ebene auf einer eigenen Hardware-Plattform installiert. Dabei sind Mischformen von verschiedenen Herstellern durchaus üblich, um die optimale Hardware-Plattform für das jeweilige Anwendungsgebiet zu finden.

Ein Benutzer kann mit seinem Präsentations-Server je nach Aufgabenstellung auf unterschiedliche Applikations-Server zugreifen. Mehrere Applikations-Server können gleichzeitig auf einen Datenbank-Server zugreifen. Die Anbindung der einzelnen Server untereinander wird über das Netz realisiert. Das *R/3*-System kann sich aufgrund seines Designs allen Varianten der Client/Server-Architektur anpassen. So wäre z.B. auch eine Konfiguration möglich, in der die Applikations- und Präsentationsebene in einem Rechner realisiert sind und das Datenbank-System auf einem anderen Rechner.

2.4.2 Präsentations-Service

Das SAPGUI (SAP Graphical User Interface) ist der SAP-eigene Terminalprozeß, der die Möglichkeiten der jeweils verfügbaren Präsentationssoftware ausnutzt, um die grafische Oberfläche des R/3-Systems auf der jeweiligen Hardware zu realisieren. Es werden die gängigsten Produkte, wie z.B. OSF/*Motif*, Microsoft *Windows* etc., angeboten.
Letztendlich wird auch die Client/Server-Fähigkeit unterstützt. Die Rechner können nicht nur über LANs miteinander gekoppelt werden, sondern auch über WANs, ohne eine schwerwiegende Leistungseinbuße zu befürchten. Dies liegt daran, daß keine komplett aufbereiteten Bilder zum Präsentationsrechner geschickt werden. Vielmehr werden zwischen dem Präsentations- und dem Applikationsrechner generische, plattformunabhängige Beschreibungen der Darstellung übermittelt. Die grafische Darstellung wird erst mit den Möglichkeiten der auf dem Rechner laufenden Präsentationssoftware erstellt. Der dabei entstehende Datenstrom liegt bei etwa einem bis zwei Kilobyte pro Bildwechsel.

2.4.3 Applikations-Service

Hier laufen *ABAP/4*-Programme ab, die die betriebswirtschaftliche Logik des *R/3*-Systems widerspiegeln. Es werden die folgenden Dienste zur Verfügung gestellt:

- **Dialog-Prozeß** - steuert den Benutzerdialog und ermöglicht eine Vorverarbeitung der Eingabe, z.B. Wertbereichüberprüfung
- **Verbuchungs-Prozeß** - ist für die Ausführung der in der Protokollsatz-Datei aufgelisteten Datenbankänderungen verantwortlich
- **Enqueue-Prozeß** - zu sperrende Datenobjekte werden in einer Sperrtabelle verwaltet
- **Batch-Prozeß** - führt zu einem bestimmten Zeitpunkt dialogfreie Programme aus
- **Spool-Prozeß** - erledigt die gepufferte Ausgabe von Daten an Drucker, Faxe etc.

Ein Applikations-Server muß mindestens den Dialog-Prozeß, kann aber zusätzlich eine beliebige Teilmenge der genannten Dienste anbieten. Zu beachten ist, daß jeder Dienst mindestens einmal in einem*R/3*-System vorkommt und der Enqueue-Prozeß nur einmal.

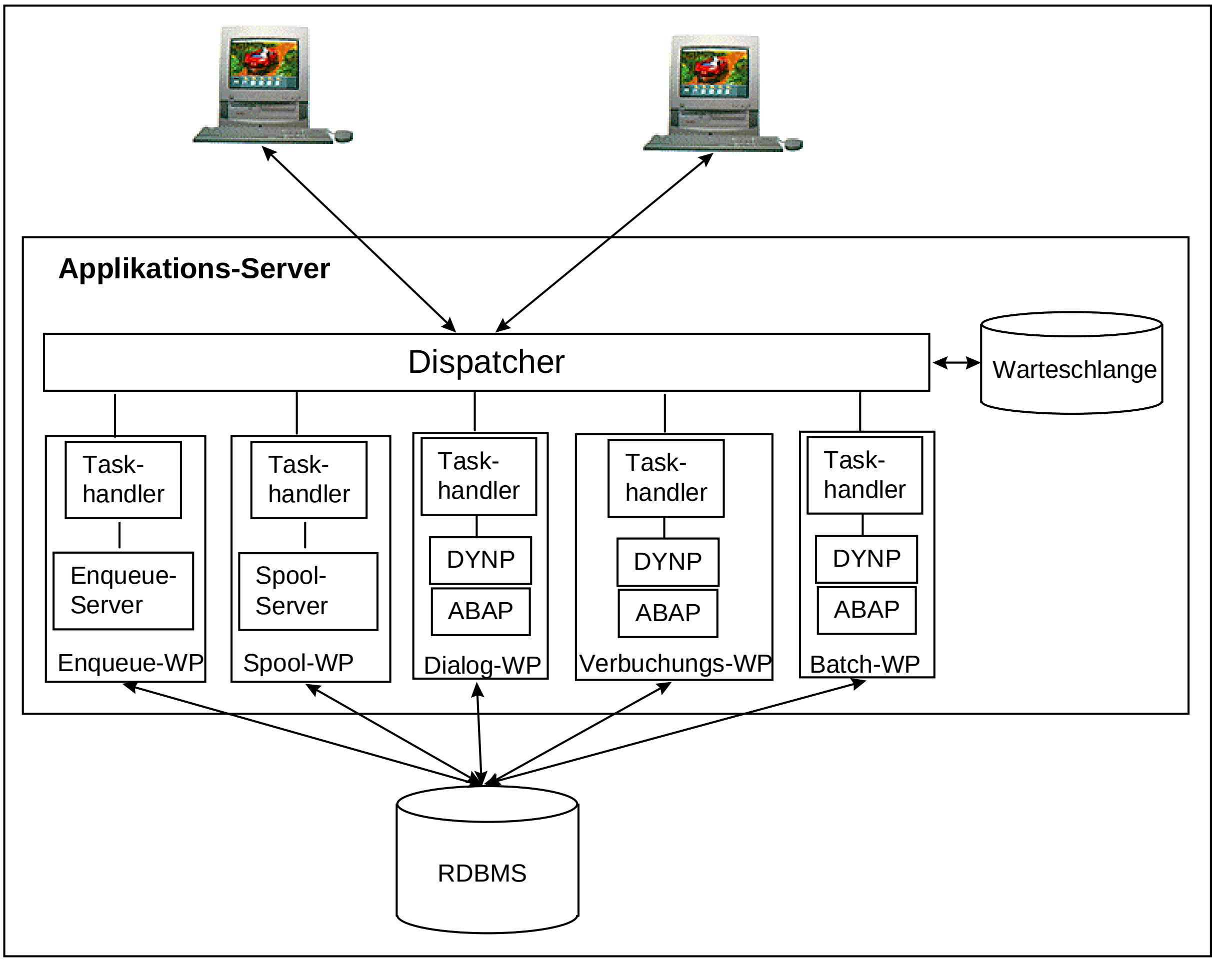

Abbildung 2.10: Applikations-Server

Technisch gesehen sind in einem Applikations-Server nur ein Dispatcher vorhanden und ein oder mehrere Workprozesse (siehe Abbildung). Die obigen Dienste werden durch die Workprozesse[5] erbracht. Der Dispatcher nimmt eine Dienst-Anfrage der *R/3*-Anwendung entgegen und legt sie in einer Warteschlange ab. Sobald ein Workprozeß frei ist, ordnet ihm der Dispatcher eine neue Anfrage aus der Warteschlange zu.

Die Workprozesse bestehen aus folgenden Komponenten:

- **Task-Handler** - übernimmt die Kommunikation mit dem Dispatcher und die Koordination innerhalb des Workprozesses.
- ***DYNP*** - steht für Dynpro-Interpreter, der die Steuerung des Dialogs mit dem Benutzer übernimmt, dazu werden sogenannte. Dynpros (Dynamische Programme) durch einen Interpreter abgearbeitet. Die Programme werden als dynamisch bezeichnet, da sie sich den Änderungen der Datenbeschreibung in der Datenbank anpassen.
- ***ABAP*** - steht für *ABAP/4*-Interpreter, der die Interpretation und Ausführung von *R/3*-Anwendungen ermöglicht.
- **eigener Speicherbereich**

[5] siehe Beschreibung weiter unten

- **Datenbankschnittstelle**

Zur Architektur eines Workprozesses gehören Speicherbereiche zur Zwischenlagerung von Daten über unterschiedlich lange Zeiträume. Speicherbereiche, die länger als einen Dialogschritt aufbewahrt werden müssen, liegen in Hauptspeicherbereichen (Shared Memory Bereiche).

Die Daten der Anwendungsprogramme werden im Paging-Bereich abgelegt. Im Roll-Bereich sind die eher benutzerspezifischen Daten untergebracht. Zu ihnen gehören die Eingabedaten vorhergehender Dialogschritte, Verwaltungsinformationen für den *ABAP/4*- und den *Dynproprozessor*, aber auch Daten zur Charakterisierung des Anwenders, z.B. Zugriffsberechtigung.

Eine Datenbankschnittstelle ermöglicht die Kommunikation mit der zentralen, nicht verteilten Datenbank (verteilte Datenbanken werden bis jetzt noch nicht unterstützt). Dazu baut jeder Workprozeß einen Schattenprozeß zum Datenbanksystem auf.

Bei der Datenbank selbst handelt es sich um Standardsoftware. Hier werden folgende Datenbanken von der SAP unterstützt:

- *ADABAS D*
- *DB2* für AIX
- *DB2/400*
- *Informix Online*
- Microsoft *SQL Server*
- *Oracle*

2.4.4 Anforderungen an ein *R/3*-Management

Eine Möglichkeit der Gliederung von Management-Aufgaben ist die Einteilung in passive Überwachungsaufgaben (Monitoring) und aktive Eingriffe in das Netz (Controlling).

Bei den folgenden Anforderungen wird sich diese Arbeit mit den Aspekten der Verfügbarkeit befassen. Eine Aufzählung der Aspekte, die sich auch mit Konfiguration und Planung beschäftigen, würde den Rahmen dieser Arbeit übersteigen.

Die Management-Aufgaben, die sich mit dem Tuning und der Konfiguration des *R/3*-Systems beschäftigen, sollten besser den proprietären Lösungen der SAP überlassen werden. Das *Computing Center Management System* (*CCMS*) bietet eine Vielzahl von Funktionen und Diensten zur Überwachung eines *R/3*-Systems. Es können z.B. Aufgaben zum Bereich Leistung, Datenbank, Verteilung von Systemressourcen und einiges mehr kontrolliert werden. Das *CCMS* kann durch eine Oberflächenintegration in eine Management-Plattform integriert werden.

Eine verteilte Anwendung besteht aus einzelnen Komponenten, die durch Zusammenarbeit nach einem bestimmten Modell die von der Gesamtanwendung benötigten Dienste realisieren. Die Komponenten können dabei als Diensterbringer und als Dienstnutzer auftreten. Die Dienste, die nötig sind, um die Anwendung zu erbringen, heißen Basisdienste.

Im weiteren werden die Management-Anforderungen der einzelnen Komponenten und des Gesamtsystems beschrieben.

Damit der Zustand einer Anwendungskomponente bewertet werden kann, müssen auch die benötigten **Systemressourcen** überwacht werden. Die erforderlichen Ressourcen und ihre Schwellwerte sind abhängig von der jeweils betrachteten Komponente.
Die folgenden Erfahrungswerte der SAP AG geben eine Übersicht über die notwendigen Systemressourcen, sie wurden in Untersuchen der Testlabors ermittelt [CCMS96], [R3AD93], [SAPS94].

- **CPU**

Die Auslastung der Prozessoren läßt sich durch die Anzahl der wartenden Prozesse und den prozentualen Anteil der Wartezyklen pro Zeiteinheit ermitteln. Es sollten nie mehr als drei wartende Prozesse vorhanden sein und der Anteil der Wartezyklen nicht gegen 0 % gehen. Wichtig ist auch die Anzahl der Prozeßkontextwechsel pro Zeiteinheit, die nicht über 200/Sekunde liegen sollte.

- **Hauptspeicher**

Engpässe im Hauptspeicher liegen vor, wenn der verbleibende physikalische Hauptspeicher geringer als 10 MB ist oder die Maschine Blöcke auslagern muß. Liegt die Anzahl von aus dem Hauptspeicherbereich ausgelagerten Blöcken über 30.000 je Stunde, liegt ein Engpaß vor.

- **Dateisystem**

Das *R/3*-System besitzt sogenannte Puffer auf dem Dateisystem. Die Effizienz solcher Puffer sollte nicht unter 95 % liegen, was aus dem Verhältnis von erfolgreichen und erfolglosen Zugriffen errechnet wird. Andernfalls ist eine Vergrößerung des entsprechenden Puffers ratsam.

- **Syslog**

Ereignisse des Systems und der Anwendungskomponenten werden in der Syslog-Datei protokolliert. Die wichtigen Einträge müssen gefiltert und als Traps weitergeleitet werden. Die Syslog-Datei muß außerdem regelmäßig geleert bzw. umkopiert werden.

- **Netz**

Bei verteilten Systemen trägt die Performance des eingesetzten Netzes ihren Teil zur Gesamtperformance bei. An alle beteiligten Rechner müssen Testdatenpakete versandt und so die Geschwindigkeit der Übertragung ermittelt werden. Mit dem Werkzeug, Netzmonitor der SAP, können von LANProbes auf Basis der RMON-MIB Testdaten versendet und anhand der erhaltenen Information die Netzlast beurteilt werden.

- **Dialog**

Die Antwortzeiten sollten unter 1 bis 2 Sekunden liegen.

- **Batch, Verbuchung**

Diese Dienstgüte ergibt sich aus der Wartezeit der Anforderung in der Warteschlange und der Leistung des DBMS. Für die Überprüfung der Leistungswerte gibt es eine eigene Transaktion (st04) im *R/3*-System. Trotzdem ist die Beurteilung der Leistung einer Datenbank sehr schwierig, denn es werden verschiedene Hersteller unterstützt, die nicht alle die gleichen Meßwerte zur Verfügung stellen. Um die Leistung zu beurteilen, sind deshalb sehr gute Kenntnisse der jeweiligen Datenbank nötig und der physikalischer Aufbau.

- **Spool**

Bei lokalen Druckaufträgen, die länger als 15 Minuten dauern, kann davon ausgegangen werden, daß der Dienst ausgefallen ist. Die Sperrtabelle[6] sollte einen Füllungsgrad kleiner als 95 % haben. Außerdem muß überwacht werden, ob einzelne Tabellen längere Zeit, größer 30 Minuten, gesperrt sind. Dies deutet auf eine Deadlock-Situation hin, die vom Administrator beseitigt werden muß.

- **Basisdienste**

Die Basisdienste müssen immer verfügbar sein, damit die Anwendung erbracht werden kann. Die Überprüfung der Systemressourcen für einen Basisdienst reicht jedoch nicht aus.
Sind die Systemressourcen für ein DBMS und einen Applikations-Server vorhanden, muß der Benutzer nicht automatisch bestimmte Tabellen aktualisieren können. Der Workprozeß Enqueue kann z.B. falsch arbeiten und genau die vom Benutzer benötigten Tabellen sperren. Dies kann durch einen Vergleich der gesperrten Objekte mit den durch die Anwendungsprogramme geforderten Sperrungen erkannt werden.

Deshalb müssen für alle Basisdienste gewisse Leistungsdaten definiert werden. Durch festgesetzte Dienstgüteparameter der Leistungsdaten läßt sich eine Verfügbarkeit der Basisdienste garantieren.

- **Systemkonfiguration**

Des weiteren sind Start- und Stop-Meldungen der einzelnen Komponenten nötig. Nur so hat der Administrator einen Gesamtüberblick über das System. Es kann eine Übersicht der verfügbaren Basisdienste in Abhängigkeit von den architekturspezifischen Gegebenheiten erstellt werden. Diese Konfiguration muß in bestimmten Zeitabständen (abhängig von der notwendigen Verfügbarkeit des Gesamtsystems) überprüft werden.

- **Leistung**

Durch Sammeln von Leistungsdaten der einzelnen Komponenten muß eine Gesamtleistung berechnet werden. Diese muß bei Problemen in den Komponenten oder beim Starten bzw. Stoppen von Komponenten neu berechnet werden. Ein einheitliches Berechnungsschema kann nicht aufgestellt werden, da die Leistung eines *R/3*-Systems von der jeweiligen Ausprägung der Architektur abhängig ist. Es ist auch sehr schwierig die Leistungsfähigkeit eines kompletten Systems zu garantieren. Oftmals reicht der Geschäftsführung die hohe Verfügbarkeit von einzelnen Anwendungen des *R/3*-Systems. Davon ist aber wieder die Berechnung der Leistung abhängig.

Das **Datenbank-System** kann als einzige Komponente nicht mehrfach vorhanden sein. Performance-Einbrüche oder ein Ausfall des DBMS wirken sich deshalb sofort auf den laufenden Betrieb aus.
Allerdings wurden die Management-Anforderungen in dieser Arbeit nicht formuliert. Die Liste der Anforderungen wäre zu ausführlich und die Kriterien sind teilweise vom verwendeten Produkt abhängig. Auch die Hersteller-Werkzeuge erfüllen unterschiedliche Funktionalitäten. So gibt es z.B. von Oracle eine herstellerspezifische Oracle-MIB für das DBMS.
Die aufgeführten Anforderungen an die Ressourcen sind nur Richtlinien der SAP AG. In diesem Bereich gibt es sehr viele konträre Meinungen, mit welchen Parametern die Leistung über-

[6] siehe Beschreibung Enqueue-Prozeß

prüft werden kann. Die Grenzen der obigen Parameter können auch schwanken. Diese sind abhängig von den eingesetzten Modulen, den Instanzen[7] oder die Art der Ausprägung. Manche Firmen setzen auch nicht alle erhältlichen Module ein, wobei die obigen Zahlen davon ausgehen. Eine Anwendung des *R/3*-Systems kann für ein Unternehmen unverzichtbar sein, dann werden oft mehrere Applikations-Server mit diesem Modul installiert. Damit sinkt die Wahrscheinlichkeit des Ausfalls dieser Anwendung und es verändert sich die Anforderung bezüglich der Gesamtüberwachung des Systems. Man kann also erkennen, daß es keine einheitlichen Parameter für die Überwachung eines *R/3*-Systems gibt. Es können lediglich Ansätze gezeigt werden, die die wichtigsten Problemstellen aufzeigen.

Die geeignete Management-Plattform zu finden, mit der sich diese Grenzen überprüfen lassen, ist ebenfalls schwierig. Die Anforderungen an ein *R/3*-System sollten zeigen, wie aufwendig es ist, die Verfügbarkeit einer einzelnen, komplexen Anwendung zu garantieren. Es gibt zahlreiche Parameter, die überwacht und kontrolliert werden müssen. Damit die Überwachung überhaupt möglich ist, ergeben sich eine Vielzahl von Anforderungen an eine Management-Plattform. Ein Unternehmen wünscht sich nicht nur eine Management-Plattform mit vielen modernen Funktionen, sondern ein Management-System, welches die betrieblichen Geschäftsprozesse überwacht und frühzeitig Warnungen generiert.
Allerdings lassen sich die Anforderungen für die Plattform noch schwerer beschreiben als für ein *R/3*-System. Hat sich ein Unternehmen für ein *R/3*-System entschieden, gibt es eine Menge von Diensten, die funktionieren müssen, damit das Gesamtsystem verfügbar ist. Bei der Management-Plattform ist das anders, denn was man mit einem Management-System überwachen will, hat erheblichen Einfluß auf die Anforderungen an dieses Produkt.

Natürlich wird großen Unternehmen oft die Möglichkeit einer Teststellung gegeben, um die Tauglichkeit im Einsatz zu prüfen. Aber bei so vielen Produkten im Management-Bereich lassen sich nicht alle mit einer Teststellung auf ihre Tauglichkeit überprüfen. Dies ließe sich nicht mit dem zeitlichen Rahmen eines Einführungsprojekts vereinbaren.
Man erkennt also, daß ein Test nicht ausreichend ist. Vielmehr müssen die verschiedenen Management-Plattformen bewerten werden, wozu geeignete Techniken nötig sind. Diese Verfahren müssen auch abstrakte Sachverhalte messen können.

Im nächsten Absatz werden deshalb die Methoden erläutert, die im Kriterienkatalog verwendet werden. Diese Techniken dienen dazu, die Management-Plattform vergleichbar zu machen.

2.5 Techniken der Bewertung und Messung

In diesem Abschnitt wird zuerst der Unterschied zwischen Bewerten und Messen erläutert. Im weiteren Verlauf werden die verschiedenen Bewertungsverfahren vorgestellt und jeweils mit einem Beispiel erläutert. Diese Verfahren werden nicht alle in dieser Arbeit Verwendung finden. Da der Kriterienkatalog aber für den Anwender erweiterbar sein soll, muß eine repräsentative Auswahl an Verfahren vorgestellt werden. Somit hat der Benutzer für neue Kriterien die Werkzeuge zur Verfügung, um auch diese bewerten zu können.
Es gibt zwei grundsätzlich verschiedene Verfahren: Das Messen und das Bewerten.

[7] „Eine Instanz ist eine administrative Einheit, der die Komponenten eines *R/3*-Systems zugeordnet sind. Alle Komponenten, die zu einer Instanz gehören, müssen gleichzeitig gestartet und gestoppt werden.“ [R3SY93]

Unter Bewerten versteht man eine Rangfolgebildung für Eigenschaften von Konstrukten. Unter Messen versteht man die Ermittlung eines bestimmten Zustandes, z.B. das Gewicht eines Gegenstandes anhand eines normierten Normals.
Tatsache ist aber, daß sich in der Praxis die Kriterien oft nicht in Messen und Bewerten trennen lassen. Meistens wird nur von Messen gesprochen, obwohl es sich eigentlich um eine Bewertung handeln müßte [MBAU96].

2.5.1 Bewertungstechniken und Verfahren

Manche Anforderungen an eine Management-Plattform können nicht einer Messung unterzogen werden. Die gelungene Gestaltung einer grafischen Oberfläche ist ein derartiges Kriterium. Diese Anforderung kann noch in mehrere Teile zerlegt werden, so daß sich wieder meßbare Einheiten ergeben. Auflösbare Einheiten könnten sein: kann die Schriftart gewechselt werden, Anzahl der verwendeten Ikons etc. Jedoch kann der subjektive Eindruck einer Oberfläche nur bewertet werden.
Bewertungstechniken können den Prozeß der Entscheidungsfindung stark erleichtern und objektivieren. Die nachfolgenden Verfahren sollen als Hilfe bei der Bewertung von subjektiven Eindrücken dienen.

Es gibt in diesem Bereich eine Vielzahl von Techniken. Das Punkteverfahren ist eine Technik, die sich sehr leicht anwenden läßt und die für Bewertung einer Management-Plattform nötig ist.

Punkteverfahren:

- Beschreibung

Jedem Gruppenmitglied wird eine bestimmte Anzahl von Punkten gegeben (üblicherweise zwischen 3-6 Punkten). Diese Punkte können von den einzelnen Teilnehmern auf die Vorschläge verteilt werden. Dabei besteht keinerlei Restriktion bei der Punktevergabe. Im Anschluß daran werden die Punkte pro Szenario addiert und somit kann eine Rangfolge gebildet werden.

- Beispiel

Konstrukt	Bewertung	Summe	Rang
Ereignis-Management	|||||	5	3
grafische Oberfläche	||	2	5
Datenbank	||||| ||	7	2
Konfigurations-Management	||||| |||	8	1
verfügbare Schnittstellen	|||	3	4

Abbildung 2.11: Matrix für das Punkteverfahren

Dieses Beispiel soll verdeutlichen, wie die persönlichen Schwerpunkte eines Unternehmens gefunden werden können. Nicht alle verschiedenen Bereiche, die eine Management-Plattform abdeckt, sind für ein Unternehmen gleichsam interessant. Hat das Unternehmen z.B. schon

viele Datenbanken im Einsatz, so ist es nicht gewillt, noch eine neue Datenbank einzuführen. Somit ist der Bereich Datenbank für das Unternehmen sehr wichtig, der Bereich sollte also bei der Bewertung der Plattform gut abschneiden.
Sind im Unternehmen keine externen Anwendungen für die Anbindung an die Management-Plattform vorhanden und es sind auch keine Eigenentwicklungen geplant, so wird die Firma nicht sehr viel Wert auf die verfügbaren Schnittstellen einer Management-Plattform legen.

- Einsatzmöglichkeit

Dieses Verfahren wird hauptsächlich zur Bestimmung und Strukturierung von Reihenfolgen benutzt. Nicht alle Teilbereiche des Kriterienkataloges sind für jedes Unternehmen interessant. Manche Firmen werden gewisse Management-Aufgaben gar nicht erfüllen, dagegen wird auf andere besonders hohen Wert legen. Das Punkteverfahren ist nicht geeignet, Kriterien des Kataloges zu bewerten. Allerdings kann dieses Verfahren helfen, dem Betreiber und seinen Mitarbeiten die persönlichen Schwerpunkte im Katalog zu finden.
Mit dieser Technik soll z.B. verhindert werden, daß eine Plattform als schlecht bewertet wird, die lediglich im Modul für Telekommunikations-Management schlecht ist. Ein Produkt kann für ein Unternehmen, das kein Telekommunikations-Management benötigt, die Ideallösung darstellen.

2.5.2 Grundlegende Konzepte des Messens

Nach wissenschaftlichen Definitionen muß beim Messen immer ein standardisiertes Maß als Basis verwendet werden.
In der Praxis sieht dies jedoch anders aus. Hier werden oft sehr komplexe Tatbestände (Konstrukte genannt) quantifiziert.
Die Skala ist unmittelbar mit dem Maß verknüpft. Die verschiedenen Typen von Eigenschaften der Messung sollen mit Zuständen in der Realität in Beziehung gesetzt werden. Das Niveau der Skala steht dabei für die Quantität der Konstrukte.
In der nachfolgenden Grafik sind verschiedene Skalen dargestellt. Im späteren Verlauf wird aber nur auf die Skalen eingegangen, die für den Gebrauch im Kriterienkatalog geeignet sind.

Gegenwärtig unterscheidet man folgende Meßskalen:

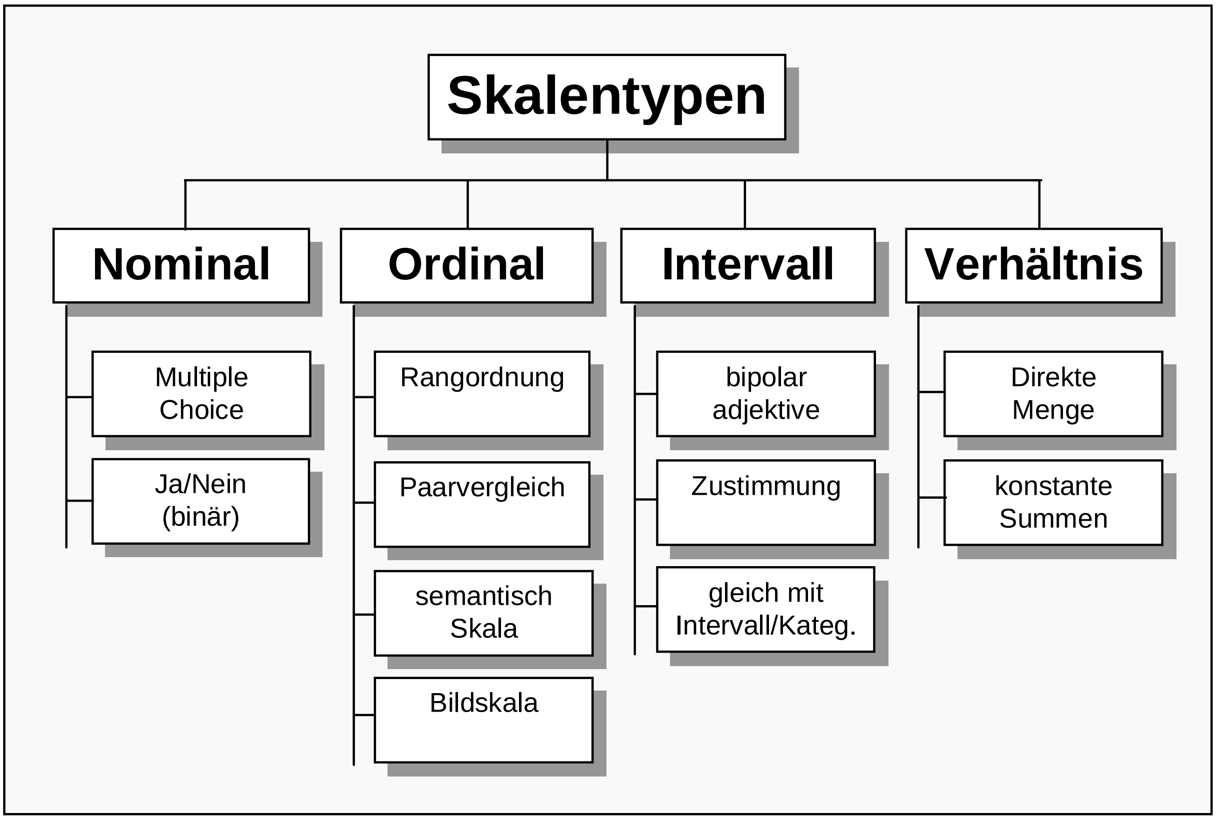

Abbildung 2.12: Skalentypen

Auf der **Ordinalskala** gemessene Eigenschaften sind neben dem Prinzip der Gleichheit/Ungleichheit auch unter dem Prinzip größer oder kleiner vergleichbar. Bei solchen Abbildungen stehen die Zahlen in Größenrelationen zueinander. Dabei ist nicht das Verhältnis, sondern nur eine Größer-als-Beziehung interessant.

Bei dieser Skala gibt es folgende Ausprägungen:
- gezwungene Rangordnung
- paarweiser Vergleich
- semantische Skala
- Bildskala

Für die Messungen im Kriterienkatalog ist jedoch nur eine von Bedeutung.

Gezwungene Rangordnung:

- Beschreibung

Bei der gezwungenen Rangordnung werden die Objekte in eine Reihenfolge gebracht und können dann einzeln „benotet" werden. Dabei muß man mit der Wahl der Reihenfolge der Präsentation vorsichtig sein, das stärkste Objekt sollte nicht unbedingt gleich am Anfang genannt werden, da die Menschen dazu neigen, die meisten Punkte zuerst zu vergeben.

Eine mögliche Form ist die Ordinalskala, die auf Wörtern basiert. Es gibt verschiedene semantische Kategorien, die ein Messen ermöglichen. Der Anwender muß einer gestellten Frage eine bestimmte Kategorie zuordnen. Hat die befragte Person keine Fachkenntnisse mit der Thematik, sollte eine neutrale Kategorie gewählt werden. Dadurch wird verhindert, daß bei willkürlicher Auswahl das Ergebnis verfälscht wird.

- Beispiel

Finden Sie den Befehlssatz der textorientierten Oberfläche ausreichend?

völlig unzureichend (=1)	nicht vollständig (=2)	neutral (=3)	ausreichend (=4)	komplett (=5)

Es handelt sich <u>nicht</u> um eine Intervallskala, weil man keine Aussage über das Verhältnis zwischen den einzelnen Ausprägungen machen kann.

- Einsatzmöglichkeit

Wie das obige Beispiel zeigt, kann dieses Meßverfahren verwendet werden, um den subjektiven Eindruck eines Funktionsumfangs zu messen. Weiter Beispiele sind:
Ist die grafische Oberfläche einfach zu bedienen?
Beschreiben die Handbücher in verständlicher Weise die verschiedenen Funktionen?
Solche Fragen lassen sich nicht durch Messung der Zeit oder anderen Parametern messen. Z.B. läßt sich eine grafische Oberfläche nicht unbedingt einfacher bedienen, nur weil mehr Ikons oder mehr Farben zur Verfügung stehen. Denn diese Merkmale wären einfach zu messen.
Dieses Meßverfahren sollte deshalb sehr oft im Kriterienkatalog angewandt werden, da es viele Anforderungen gibt, die sich nicht durch ein genormtes Maß messen lassen.

Wie der Name schon impliziert, ist bei der **Verhältnisskala** das Verhältnis zwischen den Skalenwerten bedeutend. Die auf dieser Skala gemessenen Daten sind zudem noch unter dem Prinzip eines Verhältnisses zu unterscheiden. Erst jetzt ist es möglich, zu sagen, daß eine Ausprägung doppelt so groß ist wie eine andere.

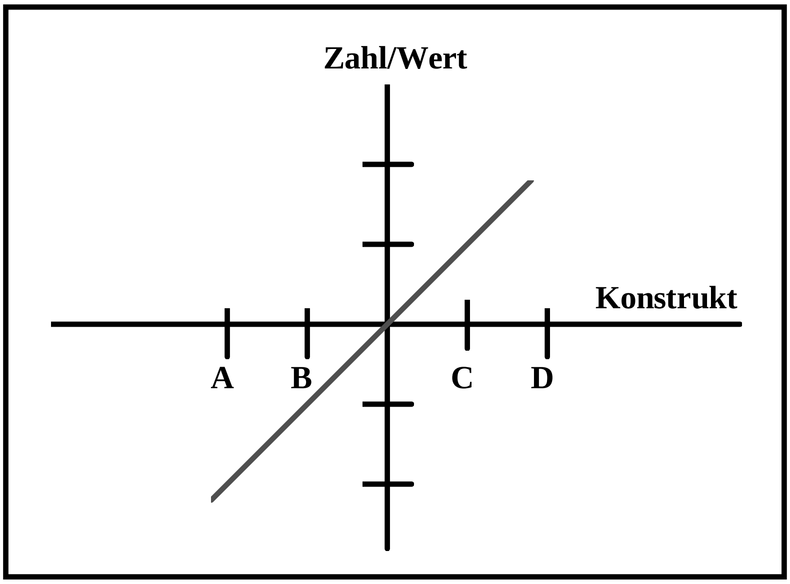

Abbildung 2.13: Verhältnisskala

Folgende Ausprägungen der Verhältnisskala sind möglich:

- direkte Menge
- konstante Summen

Die **direkte Menge** ist für diese Arbeit die einzig interessante Ausprägung.

- Beschreibung

Dabei handelt es sich um eine Messung in einer bestimmten Einheit, um verhältnisskalierte Daten zu erhalten. Der Anwender dieses Verfahrens muß sich vorher genau Start- und Endpunkt der Messung überlegen. Die Meßpunkte müssen bei den anderen Plattformen ebenfalls vorhanden sein, sonst wird die Messung nicht vergleichbar. Hierbei liegt auch das Problem dieses Verfahrens. Denn werden die Start- und Endpunkte der Messungen zu eng gewählt, muß die Messung sehr genau sein und der Aufwand wächst erheblich.

- Beispiel

Wieviel Zeit benötigt die komplette Installation der Management-Plattform?
Antwort: 75 Minuten.
Im Beispiel Installationsdauer würde die Messung verfälscht werden, wenn man Teile mitinstalliert, die nicht bei jeder Plattform vorhanden sind.

- Einsatzmöglichkeit

Dieses Verfahren kann sicherlich auch sehr häufig angewandt werden. Falls es sich bei der Messung um Stückzahlen handelt, ergibt sich auch nicht die oben beschriebene Problematik. Vorsicht ist aber dennoch bei Messungen von Zeiten geboten, denn hier können die Ergebnisse eventuell nicht verglichen werden, weil die Messung nicht auf gleiche Weise bei anderen Plattformen durchführbar ist.

Die beschriebenen Verfahren zum Bewerten und Messen sind teilweise ausführlich beschrieben. Dies liegt an der Notwendigkeit, den Kriterienkatalog ständig weiterentwickeln zu können. Sollten neue Kriterien eingefügt werden, muß man sich erneut Gedanken machen, wie diese gemessen werden können.
Um den Einsatz einiger Techniken zu demonstrieren, soll am Ende des nächsten Abschnitts eine Anforderung beispielhaft bewertet werden. Vorher müssen jedoch die Techniken erläutert werden, die das Finden der Kriterien ermöglichen.

3 Erstellung des Kriterienkataloges

Wie bereits im vorherigen Kapitel beschrieben, gibt es verschiedene Techniken, um auch abstrakte Anforderungen zu bewerten.
Das Vorhandensein solcher Techniken reicht aber nicht aus. Vielmehr müssen noch Kriterien gefunden werden, die dann mit diesen Methoden bewertet werden können.
Um zu diesen Anforderungen zu kommen, genügt es nicht, nur wichtige Kriterien aufzuschreiben. Es muß ein leicht zu erweiterndes Rahmenwerk entworfen werden. Außerdem ist die Sammlung dieser Kriterien so zu strukturieren, daß einzelne Anforderungen nicht vergessen werden.

3.1 Vorgehensweise beim Ausarbeiten der Kriterien

Der erste Schritt in dieser Vorgehensweise war ein Brainstorming. Es ist wohl eine der bekanntesten und am häufigsten angewendeten Methoden der Ideenfindung. Der Vorteil dieser Methode liegt darin, vorzeitige Beurteilungen zu unterdrücken.
Mittels dieser Technik konnten einige einfache Kriterien gesammelt werden. Das Brainstorming wurde mit anderen Kommilitonen und Entscheidern aus Unternehmen durchgeführt.

Eine andere Technik war das Aufstellen von verschiedenen Benutzerrollen. Es wurden verschiedene Personengruppen definiert und für deren Bereich relevante Anforderungen gesucht.

Folgende Benutzerrollen wurden aufgestellt:
- Betreiber
- Systemadministrator
- Netzwerkadministrator
- Plattform-Benutzer
- Plattform-Administrator
- Management-Anwendungs-Entwickler

Aus den obigen Benutzerrollen wurden jeweils Personen aus Unternehmen befragt, um die entsprechenden Kriterien zu finden.
Durch diese Wahl der Personengruppen ergaben sich Überschneidungen bei den neu gefundenen Kriterien. Die Ausbeute an neuen Kriterien war demnach nicht so groß, allerdings entstanden mit dieser Technik einige sehr spezielle Anforderungen (z.B. gibt es bei der Softwareverteilung die Angabe eines Installations-Intervalls).

Eine weitere Möglichkeit, um Informationen zu sammeln, war die Literatur- und Prospektrecherche. Durch dieses Verfahren konnten neue Entwicklungen beobachtet werden, die im Kriterienkatalog berücksichtigt werden können.

Eine Aufzählung von Anforderungen reicht für einen Kriterienkatalog nicht aus. Es müssen weitere Schritte unternommen werden, um die Anforderungen in eine Struktur zu bringen.

3.2 Ausarbeitung der Gliederung des Kataloges

Es wurde versucht, diese Kriterien in verschiedene Strukturen zu ordnen. Der erste Versuch lag darin, das bekannte Schema des OSI-Funktionsmodells [ISO74984] zu verwenden.
Dieses Modell teilt alle Aufgaben des Managements in Funktionsbereiche (Konfigurations-, Fehler-, Leistungs-, Abrechnungs- und Sicherheitsmanagement). Diese Aufteilung unterstützt den modularen Aufbau und das Customizing von Management-Plattformen.
Aber der Einsatz dieses Modells als Gliederung für den Kriterienkatalog scheiterte. Nicht alle gesammelten Kriterien ließen sich bestimmten Funktionsbereichen zuordnen (z.B. Brauchbarkeit der grafischen Oberfläche). Einige wichtige Kriterien betrachteten nur betriebswirtschaftliche Aspekte, z.B. Kaufpreis der Management-Plattform.
Solche Anforderungen finden in der OSI-Einteilung keinen Platz. Ein wichtiger Gliederungspunkt umfaßt also die betriebswirtschaftlichen und nicht-technischen Kriterien.
Allerdings gab es auch einige Anforderungen, die sich nur mit technischen Funktionen befaßten. Somit konnte ein Teil der Idee von OSI übernommen werden. Ein Strukturierungspunkt Funktionen ist Teil des Kriterienkataloges. Dieser enthält die nötigen technischen Anforderungen. Natürlich müssen diese Funktionen noch weiter aufgeteilt werden, dies ist jedoch Thema des nächsten Kapitels.

Da es sich bei modernen Management-Plattformen um verteilte Anwendungen handelt, ist eine Gliederung des Kriterienkataloges nach dem Vorschlag ODP (Open Distributed Processing) von ISO [ISO107462] denkbar.

Das Referenzmodell ODP beinhaltet fünf Ebenen:

- **Enterprise Viewpoint**
  Hier wird versucht, den Kontext mit Blick auf das gesamte Unternehmen zu definieren, in dem die verteilte Anwendung genutzt werden soll. Welche Anforderungen bestehen an die Anwendung? Gibt es z.B. in dem Unternehmen bestimmte organisatorische Vorgaben, die berücksichtigt werden müssen.
- **Information Viewpoint**
  Dieser Punkt umfaßt alle Anforderungen, die die Informationstruktur und den Austausch von Informationen betreffen. ([HEAB93], S. 422) Somit wird das Informations- und Kommunikationsmodell, das der verteilten Anwendung zugrunde zu legen ist, abgedeckt.
- **Computation Viewpoint**
  Dieser Standpunkt beinhaltet die Spezifikationen an die einzelnen Komponenten einer verteilten Anwendung sowie deren Interaktion. Diese Interaktionen werden aber noch völlig systemunabhängig betrachtet.
- **Engineering Viewpoint**

Die Spezifikationen des Computation Viewpoint werden hier verfeinert. Als Resultat ergibt sich eine technische Beschreibung der Infrastruktur, also die technische Umsetzung der vorherigen Vorgaben.

- **Technology Viewpoint**
  Hier werden die Kriterien an die Hard- und Software beschrieben. Sie dienen dann als Grundlage für den Engineering Viewpoint.

Um den Zweck einer verteilten Anwendung zu beschreiben, ist die Summe aller Anforderungen aus diesen einzelnen Standpunkten (engl. viewpoints) nötig.

Dieser Ansatz wurde natürlich überprüft, ob er für Management-Plattformen geeignet ist. Zuerst wurde eine kleine repräsentative Auswahl von Kriterien ausgewählt, die alle denkbaren Eckbereiche des Kriterienkataloges abdeckt.
Dann wurde eine Matrix aufgestellt, in der diese Kriterien von oben nach unten eingetragen wurde und von links nach rechts die einzelnen Viewpoints. Dann wurde versucht, die einzelnen Anforderungen einem konkreten Standpunkt zuzuordnen. Mit dieser Technik wurde untersucht, ob sich alle Kriterien zuordnen lassen (vgl. Abbildung 2.9 und Beschreibung Punkteverfahren).
Ein Kriterium, wie der Preis, läßt sich einfach dem Enterprise Viewpoint zu ordnen. Anforderungen, die die Oberfläche der Management-Plattform betreffen, gehören zum Information Viewpoint.
Der ODP-Vorschlag sieht einer Trennung zwischen Information und Computation Viewpoint vor. Im Information Viewpoint werden Konzepte zur Beschreibung der Informationen der verteilten Anwendung und zum Austausch von Informationen zwischen den einzelnen Modulen dargestellt. Der Computation Viewpoint beschreibt detailliert die Kommunikation zwischen den Komponenten und die dazu benötigte Infrastruktur.
Damit die Leistungsfähigkeit einer Management-Plattform geprüft werden kann, ist eine solche Trennung nicht notwendig. Oftmals hat der Benutzer des Kriterienkataloges nicht die Möglichkeit die zugrunde liegende Infrastruktur zu bewerten.
Ein Beispiel hierfür ist der MIB-Browser. Wichtig ist die Möglichkeit, daß eine Hersteller-MIB geladen werden und der Browser dann die Information der neuen MIB nutzen kann. Wie der MIB-Browser die Information der neuen MIB tatsächlich erhält, ist eventuell nicht wichtig.
Die Einteilung in Engineering und Technology Viewpoint bereitet ebenfalls Schwierigkeiten. Diese Trennung würde beim Aufbau von Management-Plattformen wenig hilfreich sein.
Zum Beispiel läßt sich die Anforderung, daß die Datenbasis als verteilte Anwendung installiert werden kann, schwierig einordnen. Handelt es sich hierbei um einen Engineering Viewpoint, da sich dies aus der Forderung ergibt Management-Verantwortung zu verteilen? Ebenso könnte es sich um ein Kriterium aus dem Technology Viewpoint handeln. Denn schließlich ist eine verteilte Installation der Datenbank eine Anforderung an die Software des Systems.

Als Folgerung müßte der meiste Aufwand bei einer Erweiterung des Kriterienkataloges in die Einordnung von neuen Anforderungen gesteckt werden. Dies würde aber den Zweck des Kataloges verfehlen. Der Katalog muß leicht erweiterbar sein, um sich dem schnellen Wandel der EDV-Branche anpassen zu können.

Anderseits ist die Aufteilung der ODP in einen Technology und Information Viewpoint eine gelungene Strukturierung.

Einige Funktionen verlangen bestimmte Voraussetzungen im Informations- und Kommunikationsmodell. Es ist z.B. wichtig, daß die Objekte Attribute besitzen. Wie die Software der verteilten Anwendung dies ermöglicht, ist unabhängig von der Notwendigkeit, Attribute zu haben. Man kann erkennen, daß eine technische Sichtweise nötig ist, also Anforderungen an die Management-Plattform, die keine Funktionen sind, in dem Sinne, daß sie Informationen bereitstellen. Bei der vorherigen Analyse wurden auch Kriterien gefunden, die das Vorhandensein von bestimmten Elementen fordern oder verbieten (z.B. ein bestimmtes Management-Protokoll, Mindestausbau an Hauptspeicher). Deshalb wurde als neuer Strukturierungspunkt der technische Standpunkt hinzugefügt.

Bei der Recherche in den Firmenbroschüren wurde deutlich, daß sich die Art der Speicherung der Management-Information in eine Datenbasis unterscheidet. Die Hersteller von Management-Plattformen verfolgen unterschiedliche Ansätze bei der Darstellung und Speicherung der Management-Information. Diese Besonderheiten hatten aber weder etwas mit den Funktionen noch mit der technischen Struktur einer Management-Plattform zu tun. Es handelte sich vielmehr um die Struktur der Information.
In der Struktur des Kriterienkataloges muß das Informationsmodell der Management-Plattform festgehalten werden. Wird für das Informationsmodell ein objektorientierter Ansatz verwendet? Welche Attribute können Management-Objekte haben?
Außerdem sind Aspekte der Kommunikation wichtig. Können Management-Anwendungen Steuerinformationen versenden, um Ressourcen zu beeinflussen? Gibt es die Möglichkeit von Statusabfragen oder asynchronen Ereignismeldungen?
Diese Anforderungen zeigen, daß Kriterien aus dem Informations- und Kommunikationsmodell der Management-Plattform wichtig sind. Solche Informationen werden auch im ODP für verteilte Anwendungen beschrieben. Deshalb wurde dann auf die Idee von ODP zurückgegriffen und ein Informations-Standpunkt eingefügt.
Somit ergibt sich als endgültige Gliederung für den Kriterienkatalog:

- **Unternehmens-Standpunkt**

  Anforderung an betriebswirtschaftliche Aspekte und Informationen zur genaueren Beschreibung des Produkts

- **Informations-Standpunkt**

  Anforderungen des Informationsaustauschs und der Speicherung von Informationen

- **Funktions-Standpunkt**

  Anforderungen an die Funktionalität, um das Management zu erfüllen

- **Technik-Standpunkt**

  Anforderungen an die Technik, um eine Management-Plattform zu betreiben

Abbildung 3.1: Gliederung des Kriterienkataloges

Die gewählte Gliederung stellt die Grobstruktur des Kriterienkataloges dar. Die Struktur ist aber noch zu oberflächlich, um die Anforderungen direkt einzufügen.

Im vierten Kapitel werden deshalb die einzelnen Bereiche weiter verfeinert und dann die Kriterien eingefügt.
Ein genaues Vorgehen für die Verfeinerung der Grobstruktur gibt es nicht. Es wurde auf einige bestehende Konzepte zurückgegriffen, doch meistens wurden zusätzliche Gliederungspunkte eingefügt, um thematisch gleichartige Kriterien zusammenzufassen.
Der nächste Abschnitt demonstriert den Einsatz der Techniken anhand eines Beispiels. Nach diesem Kapitel sollte der Gebrauch der Bewertungstechniken klar sein.

3.3 Beispiel: grafische Benutzeroberfläche

Die grafische Benutzeroberfläche (im weiteren GUI genannt) eignet sich sehr gut zur Demonstration der verschiedenen Verfahren (vgl. Kapitel 2.4).
Das GUI ist zweifelsohne ein wichtiges Kriterium beim Kauf einer Management-Plattform. Es muß einfach zu bedienen sein, aussagekräftig und leicht konfigurierbar. Natürlich gibt es noch weitere Anforderungen, vorerst reichen die bisher genannten für dieses Beispiel aus.

Man erkennt an den obigen Beispielen sofort, daß sich solche Forderungen nicht so einfach messen lassen. Auch die Beurteilung durch Fachpersonal ist oftmals nicht der richtige Weg.
Ein Experte mit langer Erfahrung stellt geringere Ansprüche an die Bedienbarkeit, als eine Person die noch nicht lange mit grafischen Oberflächen arbeitet.

Hier muß also ein besonderer Weg gegangen werden. Der erste Schritt ist die Aufsplittung des Kriteriums „gutes GUI“ in leichter quantifizierbare Einzelteile.

Mittels Brainstorming kann das Kriterium „gutes GUI“ zerlegt werden. Der Betreiber ist selbst dafür verantwortlich, die geeigneten Einzel-Anforderungen zu finden. Das Wichtige dabei ist, nicht irgendwelche Kriterien zu finden, sondern solche, die die Anforderungen aus dem Betreiberumfeld widerspiegeln.

Das Sammeln von Einzel-Anforderungen reicht aber nicht aus. Um die Bewertung des GUI zu ermöglichen, müssen diese in eine Rangfolge gebracht werden. Dazu kann die vorher beschriebene Technik des Punkteverfahrens benutzt werden.
Die Mitarbeiter der IT-Abteilung können mit diesem Verfahren eine Rangordnung in die einzelnen Anforderungen bringen. Diese Rangordnung muß dann auf alle GUI der verschiedenen Management-Plattformen angewendet werden. Nur so sind die einzelnen GUI letztendlich vergleichbar.

In diesem Beispiel wurden exemplarisch folgende Einzel-Anforderungen ausgewählt:

- Portierung für Apple Macintosh
- Intuitive Bedienung
- Hauptspeicherbedarf des GUI

Diese Anforderungen demonstrieren die verschiedenen Verfahren zum Bewerten und Messen.

Die erste Anforderung: Gibt es eine Portierung für Apple Macintosh Systeme? Hierbei handelt es sich um ein binäres Nominalskala-Problem. Auf so eine Frage kann nur mit Ja oder Nein geantwortet werden.

Ein Ordinalskala-Problem findet man beim zweiten Kriterium. Das untersuchende Team muß Konstrukte definieren, welche die verschiedenen Kategorien der Fragestellung beantworten. Läßt sich das GUI intuitiv bedienen?
In der Praxis wird bei solchen Messungen oft eine Einteilung von fünf Kategorien verwendet.
Das Testteam kann durch geschickte Wahl der Phrasen versuchen, die Messung des Kriteriums in die richtige Richtung zu lenken.
Z.B. könnten die positiven maximalen Aussagen so lauten:

1. ist sehr intuitiv.
2. kann auch von ungeübten Benutzern bedient werden.

Die erste Aussage drückt aus, daß eine Person ohne Vorkenntnisse mit dem GUI arbeiten kann. Die zweite Aussage verlangt dies aber auch von ungeübten Personen. Dies ist also viel schärfer formuliert.

Die Frage: „Wieviel Hauptspeicher ist nötig, um das GUI auszuführen (minimal, optimal)?“ läßt sich mit der Verhältnisskala beantworten.
Durch eine Auswertung von Datenblättern läßt sich die Frage beantworten. Es handelt sich um eine richtige Messung.

Wurde die obige Zerlegung und Bewertung mit allen relevanten Techniken vom Betreiber und seinen Mitarbeitern durchgeführt, muß wiederum eine weitere Rangfolge gebildet werden.
Nun sollen eigentlich alle Kriterien des Kataloges verglichen werden. Um die geeignete Rangfolge über diese Kriterien mit ihren Einzel-Anforderungen zu finden, wird das Punkteverfahren angewandt. Auch hier gilt, daß bei allen Management-Plattformen dann die gleiche Rangfolge angewendet werden muß.
Nach dieser Ermittlung wird dann Einzel-Anforderung für Einzel-Anforderung zwischen den verschiedenen Ausprägungen der Management-Plattformen verglichen. Somit ergeben sich sofort die Unterschiede. Wo hat die eine Plattform Schwächen, wo Stärken. Durch die Gegenüberstellung können dann Vergleiche zwischen einzelnen Kriterien gezogen werden. Je später Unterschiede auftreten, desto unerheblicher sind sie. Denn sie befinden sich weiter unten in der Rangfolge.

Im Anhang dieser Arbeit befindet sich der eigentliche Kriterienkatalog in Form einer Tabelle. Damit die Darstellung kompakt genug bleibt, wurden in dieser Tabelle nur kurze Stichworte genannt. Teilweise sind zu den Punkten weitere Erklärungen nötig, diese werden im nächsten Kapitel gegeben.

4 Beschreibung Kriterienkatalog

In diesem Abschnitt werden die Teilaspekte des Kriterienkataloges erläutert. Die einzelnen Anforderungen werden detailliert dargestellt und gegebenenfalls mit Bespielen versehen. Jedes Entscheidungsteam muß die für sich relevanten Anforderungen aus dem Kriterienkatalog ermitteln. Deshalb können einige Kriterien in mehreren Abschnitten erscheinen.

Gleich zu Anfang soll der Aufbau des Kriterienkataloges im Anhang A erklärt werden. Der Katalog ist als Tabelle mit vier Spalten aufgebaut.
In der **ersten Spalte** werden die Kriterien mit ihren Gliederungspunkten aufgelistet. Werden neue Kriterien hinzugefügt, muß dies an geeigneter Stelle geschehen. Bei manchen Kriterien sind noch einige Stichworte vorhanden, diese sollen die Arbeit mit dem Kriterienkatalog vereinfachen.

Die **zweite Spalte** dient der Gewichtung der einzelnen Anforderungen. Somit kann der Anwender bei einzelnen Kriterien Schwerpunkte setzen und die für ihn wichtigen hervorheben. Wünscht der Benutzer dies nicht, so kann die Gewichtung auch weggelassen werden. Die Gewichtung muß dann aber bei allen Anforderungen entfernt werden, da sonst das Ergebnis verfälscht würde.
Bei der nachfolgenden Anwendung wurde ein Wertebereich von 1 (nicht wichtig), 2 (normal) bis 3 (wichtig) gewählt. Ein Wertebereich z.B. von 1-10 macht keinen Sinn. Bei so einer Skala kann keine objektive Bewertung mehr erfolgen, da die Wahl der Gewichtung zu schwierig wird. Außerdem können sich zu große Unterschiede in der Gewichtung verfälschend auf das Gesamtergebnis auswirken. Diese Schwankungen würden aber nicht den gewollten Sachverhalt widerspiegeln. Ein letzter Grund für den gewählten Wertebereich ist die Verwendung in der Praxis. „In Bereichen der Marktforschung sind Gewichtungen von 1-3 gebräuchlich“ [Gerd Bögl, Leiter Marketing, Höchst AG].

In der **dritten Spalte** wird die Bewertung der einzelnen Kriterien eingetragen. In dieser Arbeit wurde ein Wertebereich von 1-5 für die Bewertung verwendet (siehe Kapitel 2.4.1 - Gezwungene Rangordnung):

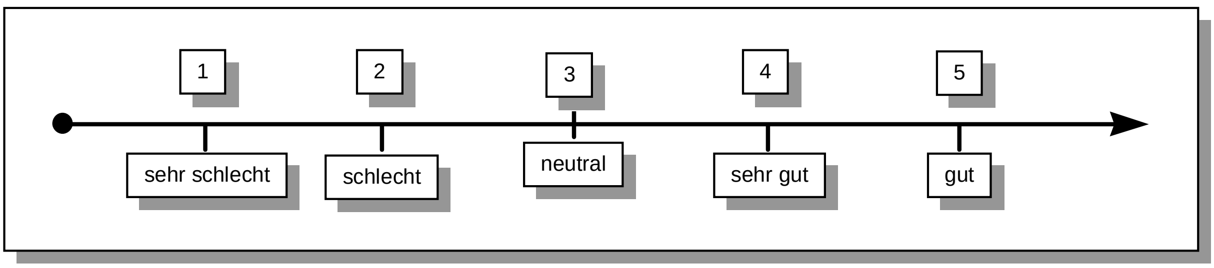

Abbildung 4.1: Bewertungsskala

Ein Grund für diese Skala ist der neutrale Skalenwert drei[1]. Eine feinere Abstufung als 1-3 ist durchaus angebracht, denn eine Anforderung ist nicht nur gut, neutral oder schlecht. Eine

[1] siehe dazu Kapitel 2.4

schlechtere Ausprägung eines Kriteriums als sehr gut, nämlich gut, ist nachvollziehbar. Eine Skala von 1-10 wäre jedoch zu fein, damit könnte die Validität[2] gefährdet sein. Durch die Benutzung von Zahlen in der Skala, bleibt die Möglichkeit der Umrechnung von gemessenen Kriterien gegeben. Als Beispiel ist im folgenden die Installationsdauer verwendet werden.

Wurden z.B. drei Management-Plattformen mit Installationszeiten von 50 Min., 60 Min. und 70 Min. getestet, beträgt die mittlere Installationsdauer (50+60+70):3=60 Min. Eine Installationszeit von 60 Min. entspricht der Bewertung 3. Weiterhin rechnet man die 60 Min. nochmal geteilt durch 3, da 60 Min. der Bewertung 3 entspricht. Man erhält dabei 20 Min. Die nächste Bewertungsstufe 2 würde also bei 80 Min. liegen.

Dieses Beispiel soll zeigen, wie gemessene Werte auf die Bewertungsskala abgebildet werden können. Allerdings stellt dieses Beispiel keine wissenschaftliche Abbildung der Meßergebnisse dar. Dies müßte eigentlich mit der Gaußschen Verteilung gemacht werden. Da bei allen Meßungen aber keine signifikante Testmenge vorhanden ist, kann diese Verteilung nicht angewendet werden.

Ein weiterer Vorteil dieser Bewertungsskala ist die Abbildung von binären Messungen auf diese Skala. Wird eine Anforderung mit Ja beantwortet, bekommt sie die Bewertung 5, sonst 1. Ebenso wird bei Kriterien, die nicht gemessen werden können oder nicht vorhanden sind, der Wert 1 gesetzt. Durch die spätere Division (siehe Beispiel weiter unten) wird ein Einfluß auf das Gesamtergebnis vermieden.

Offen bleibt nun noch die Frage: Warum soll für alle Kriterien bei der Gewichtung und der Bewertung der Wertebereich beibehalten werden?

Dafür gibt es auch mehrere Gründe. Der Kriterienkatalog wird für den Anwender übersichtlicher und einfacher zu bedienen. Ähnliche Kriterien lassen sich dadurch vergleichen. Wird die Bedienbarkeit der textorientierten und grafischen Oberfläche bewertet, so lassen sich z.B. folgende Aussagen machen:

Die textorientierte Oberfläche läßt sich im Verhältnis zu ihren Möglichkeiten genauso gut bedienen, wie die grafische.

Wurden verschiedene Wertebereiche verwendet, ist obige Aussage nicht möglich. Ein weiterer Grund ist, daß ein laufendes Ändern der Wertebereiche keinen nennenswerten Vorteil bringen würde. Man könnte natürlich jedem Kriterium die geeignete Skala zuordnen. Aber die Tests haben gezeigt, daß sich die Kriterien mit der verwendeten Skala gut bewerten lassen.

Ein weiterer Vorteil durch den einheitlichen Wertebereich ist die Möglichkeit der statistischen Auswertung von Teilbereichen z.B. Konfigurationsmanagement. Werden in so einem Bereich verschiedene Wertebereiche verwendet, läßt sich keine statistische Auswertung vornehmen.

Die **letzte Spalte** beinhaltet das Produkt über die beiden anderen Werte. Es wird die Bewertung mit der Gewichtung multipliziert. Am Ende soll die Summe über diese Produkte eine Bewertung des gesamten Produkts ermöglichen.

[2] Gültigkeit eines wissenschaftlichen Versuchs

Die Frage ist nun noch: Welchem Wert auf der Bewertungsskala entspricht z.B. eine Summe von 27?
Dazu eine Rechnung an einem Beispiel:

Kriterium	Gewichtung	Bewertung	Produkt
Kriterium I	3	5	15
Kriterium II	2	4	8
Kriterium III	1	4	4
Summen	6		27

Die Summe der Gewichtung muß ebenfalls gebildet werden. Dann wird noch dividiert:
27:6=4,5

Die Gesamtbewertung ist also 4,5. Durch die günstige Gewichtung von 1-3 kann man sagen, daß entspricht einer glatten 4. Ein Wert unter 4,33 wäre eine 4-.
Mit diesen Ausführungen sollte die Bewertung der Kriterien gut zu verstehen sein. Wird der Kriterienkatalog um neue Anforderungen erweitert, muß man sich erneut Gedanken über den Ablauf der Bewertung machen.
Nachfolgend wird die bereits bekannte Gliederung des Kriterienkataloges aus Kapitel 3.2 weiter verfeinert. Die einzelnen Anforderungen aus dem Anhang A werden erläutert, damit die Stichwörter verstanden werden können.

4.1 Unternehmens-Standpunkt

Wie in Kapitel 3.2 bereits beschrieben sollen im Unternehmens-Standpunkt betriebswirtschaftliche Aspekte eingefügt werden. Deshalb werden am Anfang die Produktinformationen verglichen, die sich normalerweise mit Hilfe eines Prospekts beantworten lassen. Unter den betriebswirtschaftlichen Aspekten finden sich Kriterien, die Anforderungen im unternehmerischen Kontext prüfen. Dem Preis ist ein extra Gliederungspunkt gewidmet, da die Unternehmensführung diesen oft als wichtiges Entscheidungsmittel sieht. Anforderungen, die das Verhältnis zwischen den Vertragsparteien untersuchen, sind unter Lieferanten/Hersteller-Beziehungen zusammengefaßt. Als Abschluß wird in diesem Standpunkt noch die Dokumentation bewertet, da sie in den anderen Standpunkten nicht einzuordnen ist.

4.1.1 Produkt

Am Anfang müssen solche Angaben wie Herstellername, Produktbezeichnung usw. aufgelistet werden ([VALTA91], S. 2). Diese allgemeinen Angaben unterscheiden das Produkt von anderen und sollen lediglich zusätzliche Informationen bieten.
Für heutige Netzmanagement-Plattformen ist meistens eine sehr leistungsfähige Hardware-Umgebung notwendig. Wie groß ist dieser Betriebsmittelbedarf? Dazu gehören Plattenplatz mit Auslagerungsdatei, Arbeitsspeicher und CPU-Leistung ([VALTA91], S. 2). Der Bedarf an Ressourcen ist natürlich stark abhängig von der Konfiguration. Werden einige Module nicht installiert, sinkt sicherlich der benötigte Plattenplatz.
Manche Firmen wollen kein neues Betriebssystem in ihrer Umgebung, sie möchten das neue Produkt auf einem von ihnen bereits eingesetzten Betriebssystem betreiben. Deshalb sind die unterstützten Betriebssysteme der Serverplattform eine wichtige Anforderung.

4.1.2 Betriebswirtschaftliche Aspekte

Moderne Unternehmen erwarten von einer Software, daß sie ein innovatives Produkt ist. Es sollen neue Denkansätze vorhanden sein. Die Innovation kann sich in mehreren Teilbereichen, wie z.B. Oberfläche, Datenbasis, Anwendung, Kommunikation mit den Agenten oder Entwicklungsumgebung ausdrücken. Welche neuen Entwicklungen sind geplant, die bestehende Probleme im Unternehmen lösen können? Andere Anwender verzichten auf Software, die mit Funktionen überladen ist, denn diese Produkte können oft fehleranfällig und instabil sein.
Die meisten Unternehmen setzen EDV-Anlagen ein, um ihr betriebliches Kerngeschäft zu rationalisieren. Auf diesen Anlagen müssen die betrieblichen Prozesse abgebildet und unterstützt werden. Wie wird die Abbildung dieser Prozesse unterstützt?
Sind mehrere tausende von Bildschirmarbeitsplätzen vorhanden, ist es wichtig, die Zufriedenheit der Benutzer zu erfassen. Nur so ist eine vollständige Planung der Neuanschaffungen möglich. Kann die Benutzerzufriedenheit erfaßt werden? Bemerken die Benutzer, daß ihre Probleme mit Hilfe der Plattform schneller gelöst werden können?
Durch Einblenden von Informationsfenstern auf der Arbeitsoberfläche könnte der Benutzer besser und schneller informiert werden. Damit ist gewährleistet, daß fast jeder Benutzer die Nachricht erhält, im Gegensatz zu traditionellen Möglichkeiten (Aushang am Infobrett).
Gibt die Plattform Unterstützung bei der Entwicklung von Kostenumlageschlüsseln? „Die Erfassung der entstandenen Kosten in der EDV ist heute weitestgehend gelöst. Aber die abteilungsbezogene und produktbezogene Umlage (gen. Kostenumlageschlüssel) bereitet noch große Schwierigkeiten“ [Prof. Dr. Kurt Vikas, Universität Graz]. Deshalb müssen Mittel vorhanden sein, um die Kosten transparent und deutlich darstellen zu können.
In welchen Bereich hat der Hersteller sein Produkt positioniert? Sieht er z.B. den Schwerpunkt der Management-Plattform im Telekommunikations-Management oder im Fehlermanagement?

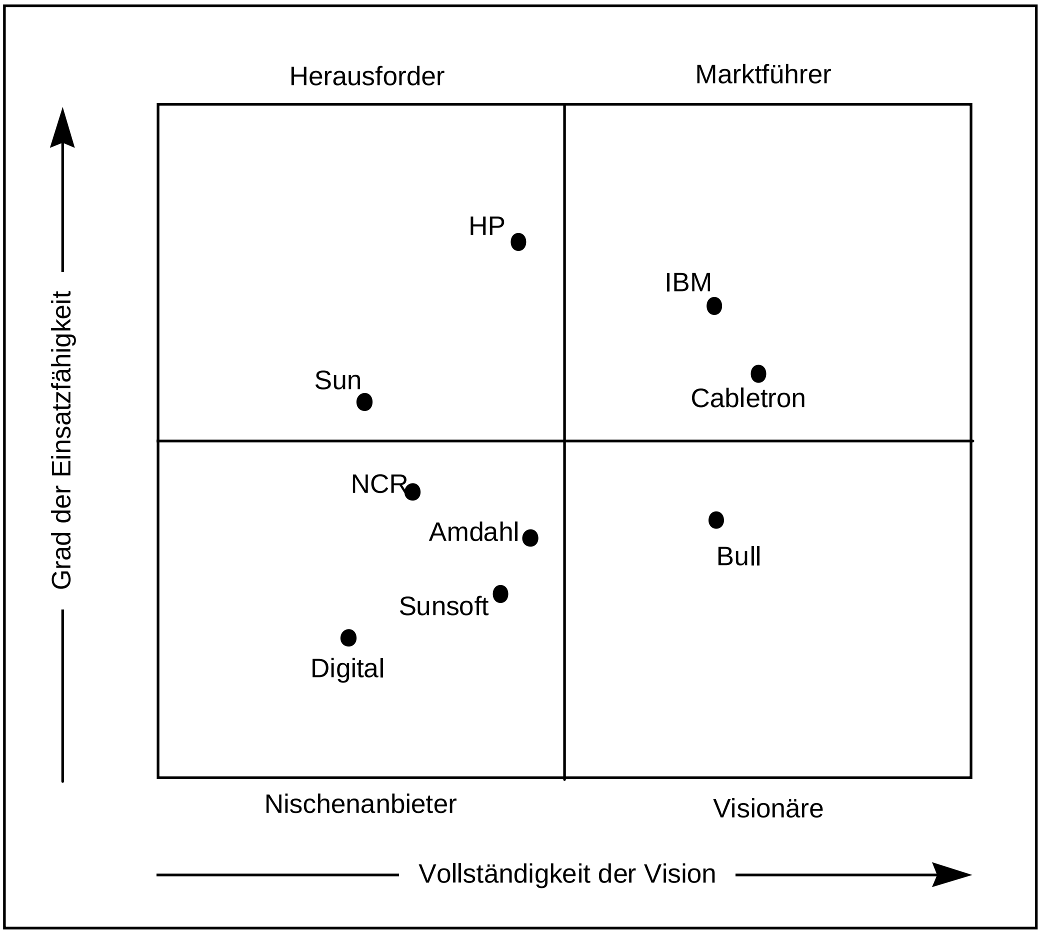

Abbildung 4.2: Enterprise Management - Magisches Quadrat ([GAGR96], S. 1)

Abbildung 4.2 zeigt wie sich die einzelnen Anbieter von Management-Lösungen am Markt positionieren. Hersteller wie IBM und Cabletron streben die Marktführerschaft bei Plattformen an, während z.B. Anbieter wie Digital versuchen, mit Nischenprodukten am Markt zu bestehen.

4.1.3 Preis

Der Preis ist immer noch eine wichtige Anforderung[3], ein Kriterium auf das immer geachtet werden wird. Oftmals bieten die Hersteller mehrere verschiedene Lizenzpolitiken an, z.B. doppelte Anzahl von Lizenzen, aber geringere Kosten pro Lizenz. Eine andere Möglichkeit der Nutzung von Software ist Leasing. Gerade in diesem Bereich sollten gute Kenntnisse von Vertragsabwicklungen vorhanden sein.
Manchmal werden zusätzliche Servicekonzepte angeboten, dies sind oft zusätzliche Leistungen um den Kauf interessanter zu machen, z.B. Unterstützung bei der Installation oder zehn Stunden kostenlose Hotline.
Komplexe Softwareprodukte werden oft durch Wartungsverträge betreut. Solche Angebote sollten ebenfalls geprüft werden.

[3] siehe Zitat in Kapitel 2.2.4 aus [COOK97]

Wünschenswert ist auch ein modularer Aufbau der Software, so daß der Käufer die Module kaufen kann, die wirklich benötigt werden. Damit läßt sich die Management-Plattform besser an die eigenen Bedürfnisse anpassen.

4.1.4 Lieferanten-Beziehungen

In der Computer-Branche gibt es einige unseriöse Unternehmen. Oft werden Produkte umworben und verkauft, obwohl sie noch gar nicht verfügbar sind. Damit der Käufer vor solchen Erfahrungen sicher ist, sollte er eine gute Beziehung zum Lieferanten aufbauen. Wenn der Lieferant dann noch eine kostenlose oder billige Hotline zur Verfügung stellt, sind wichtige Anforderungen vorhanden.
Hat das Unternehmen bereits Geschäftsverbindungen mit dem Hersteller? Vielleicht sind sogar bereits Produkte dieses Herstellers im Haus? Kann die Firma auf die Entwicklung der Software durch Verbesserungsvorschläge Einfluß nehmen?
Die beste Art, ein neues Produkt kennenzulernen, ist eine Teststellung. Ist diese möglich? Wie lange kann getestet werden? Was kostet der Test? Werden diese Kosten beim Kauf angerechnet?
Einfache Aufgaben im Managementumfeld müssen nicht vom Fachpersonal erledigt werden. Dies spart zwar Kosten, fordert aber, daß der Umgang mit der Plattform auch für ungeübtes Personal möglich ist. Sind beim Umgang mit der Management-Plattform Schulungen notwendig? Gibt es Schulungen für verschiedene Benutzergruppen oder können die Schulungen individuell auf das Unternehmen abgestimmt werden? Können die Schulungen im eigenen Unternehmen durchgeführt werden, um auf konkrete Probleme im Unternehmensbereich einzugehen? Auch hier ist die Frage nach den Kosten der Schulung obligatorisch.

4.1.5 Dokumentation

Die wichtigste Frage lautet: Ist eine ausführliche Dokumentation in Form von Handbüchern vorhanden? Besonders im Microsoft *Windows* Umfeld ist in letzter Zeit Software erschienen, die ohne oder nur mit dürftiger Dokumentation ausgeliefert wurde. Ein zusätzlicher Pluspunkt ist, wenn die Handbücher in der jeweiligen Landessprache verfügbar sind. Auch das Nachbestellen von Handbüchern in anderen Sprachen kann für ein internationales Unternehmen von Vorteil sein.
Die Handbücher sollten außerdem verständlich geschrieben und nicht mit zu vielen Fachbegriffen versehen sein. Es müssen gute Beispiele mit erklärenden Bildern angegeben werden ([GMD800], S. 19).
Die verschiedenen Handbücher sollten nach Können und Benutzerrollen (z.B. Administrator, Anfänger ...) gegliedert sein. Ein Inhaltsverzeichnis über alle Handbücher kann bei der Suche nach Informationen sehr hilfreich sein. Der Verweis auf weiterführende Literatur kann den Anfängern den Einstieg erleichtern. Das Vorhandensein einer Online-Dokumentation kann das Suchen von Funktionen während der Arbeit unterstützen ([GMD800], S. 20). In diesem Zusammenhang ist auch die kontextsensitive Hilfe zu nennen, die vor Ort Unterstützung bei Bedienungsproblemen bietet.

4.2 Informations-Standpunkt

Der Informations-Standpunkt unterteilt sich in die Teilbereiche Datenbasis, Oberflächen und Anfrage-System. Im Bereich Datenbasis sollen die Konzepte der Speicherung der Management-Information bewertet werden (siehe Kapitel 2.2.3). Da die Oberflächen die Schnittstellen zu den Benutzern darstellen, müssen in diesem Standpunkt die Anforderungen dafür verfaßt werden. Im Teilbereich Anfrage-System werden Kriterien beschrieben, die die Auswertung von Informationen überprüfen. Hier besteht ein enger Zusammenhang mit dem Ereignismanagement im nächsten Standpunkt.

4.2.1 Datenbasis

Es gibt heute die verschiedensten Arten von Datenbanken. Welche Technik die Datenbasis der Management-Plattform verwendet, bringt immer gewisse Vor- und Nachteile mit sich[4] ([VALTA91], S. 4). Für ein Unternehmen ist es meistens von Vorteil, wenn es auf Standardsoftware zurückgreifen kann. Einige Hersteller von Management-Plattformen bieten deshalb als Datenbasis handelsübliche relationale Datenbanken an.
Wenn die Datenbasis verteilt über mehrere Rechner installiert werden kann, so bleibt der Plattenplatz für die Datenbank besser skalierbar. Kann die Datenbasis nach Belieben gewechselt werden? So können z.B. in objektorientierten Datenbanken Beziehungen zwischen Objekten bestehen. Aber bleibt bei einem Wechsel der Datenbank diese Information erhalten? Auch die Semantik der Information muß von einem Wechsel unberührt bleiben, sonst macht dieser keinen Sinn.
Bei großen Management-Lösungen ist die Information meist unverzichtbar. Hier kann eine Backup-Datenbasis eine zusätzliche Sicherheit gegen Systemausfälle bieten. Auch bei großen Netzen, die über langsame WAN-Verbindungen vernetzt sind, kann eine doppelte Datenhaltung Leitungskosten sparen.
Eine Änderung der Datenbank-Struktur kann bei Fehlern große Probleme im Management erzeugen. Deshalb sollten Datenbank-Änderungen protokolliert werden können. Wenn der Administrator es wünscht, sollte auch die Änderung durch den Informationsbaustein (siehe Abbildung 2.1) der Management-Plattform protokolliert werden können. Hier ist allerdings Vorsicht geboten, da die anfallende Datenmenge sehr groß werden kann.
Zur betriebswirtschaftlichen Inventarisierung ist es wünschenswert, wenn Werkzeuge und Export-Funktionen zum Auslesen der Information aus der Datenbasis vorhanden sind. Eine wichtige Manipulation ist das Löschen von Objekten. Wann werden alle Informationen zu einem Objekt aus der Datenbasis entfernt ([GMD800], S. 16)? Nach vielen Transaktionen auf eine Datenbank, kann eine Pflege nötig sein. Werden dafür Werkzeuge zur Verfügung gestellt ([GMD800], S. 17)? Manchmal kann auch eine direkte Manipulation der Daten ein erhebliche Zeitersparnis bedeuten. Diese Möglichkeit sollte aber durch Authentifizierung ausreichend abgesichert sein. Dazu sind vernünftig dokumentierte Schnittstellen nötig, wie z.B. SQL, C.
Die Struktur der Daten soll verdeutlichen, welchen Informationen wie gespeichert sind. Nur so kann der Anwender die Abhängigkeiten zwischen den einzelnen Objekten verstehen. Die Auswirkung beim Löschen von Informationen kann dann erst abgeschätzt werden. Außerdem ist noch wichtig, durch welche Namengebung ein Objekt in der Datenbasis eindeutig gemacht wird. Gibt es Beschränkungen bei der Namengebung oder wird der Name automatisch vergeben?
Damit die Management-Information möglichst realistisch in der Datenbasis abgebildet werden kann, müssen leistungsfähige Objektkataloge verwendet werden. Welche Objektkataloge werden von der Management-Plattform unterstützt? Es gibt deshalb folgende Objektkataloge ([VALTA91], S.3):

- Internet-MIBs
- Experimental MIBs (z.B. für Token Ring, FDDI ...)

[4] siehe Vorteile in Kapitel 2.2.3

- Privat MIBs (von verschiedenen Herstellern)
- OSI-Objektkataloge
- herstellerspezifische Objektkataloge (wenn der Hersteller ein eigenes Objektekonzept gebraucht)

Wichtig ist auch das interne Format, was zum Laden einer MIB nötig ist ([GMD800], S. 18). Vielleicht sind Umwandlungen von ASN.1-Definitionen in ein anderes Format nötig?
Ebenso wichtig wie standardisierte Objektkataloge sind Erweiterungen der Objekte um zusätzliche Informationen (z.B. Lieferant). Die Information sollten sich nach den Gesichtspunkten des Unternehmens gliedern lassen, so daß im Unternehmen eine einheitliche Informationsstruktur entsteht.
Als Beispiel ist hier die Vergabe von Inventarnummern zu nennen. In großen Unternehmen ist es üblich, allen Geräten eine eindeutige Nummer zu geben, hinter der sich eine bestimmte, historisch gewachsene Information verbirgt. Diese Nummer sollte auch in der Datenbasis gehalten werden können, damit die Information nicht in einem getrennten System verwaltet werden muß.
Sind also Erweiterungen der Objekte möglich ([VALTA91], S. 3)? Welche Methoden zur Erweiterung gibt es und mit welchen sprachlichen Mitteln sind diese durchführbar? Aus den erstellten Objekten neue durch Vererbung abzuleiten, würde die Arbeit zusätzlich erleichtern.
Der MIB-Browser wird hier ebenfalls bewertet, denn schließlich bietet er den Zugriff auf die Informationen der MIBs ([GMD800], S. 18). Bei der Wertung steht nicht so sehr die Bedienung des Browser im Vordergrund, sondern die Art der Informationspräsentation. Vielleicht bietet der Hersteller aber auch andere Anwendungen an, um auf die Informationen der Objektkataloge zuzugreifen.
Oftmals sind mehrere Personen mit dem Management des Netzes beschäftigt, deshalb sollte die Vergabe von getrennten Zugriffsrechten für Objekte, Benutzer und Datenbanken möglich sein.

4.2.2 Oberflächen

Gibt es eine grafische und eine textorientierte Oberfläche ([VALTA91], S. 6? Die grafische Oberfläche ist bei der Präsentation von Daten besser geeignet, da sich Information besser darstellen läßt. Die textorientierte Benutzerschnittstelle ermöglicht aber, Aufgaben im Batchbetrieb, eventuell sogar in der Nacht, ausführen zu lassen. Bei der grafischen Oberfläche ist es noch wichtig, daß alle Funktionen auf der gleichen Oberfläche integriert sind. Die Management-Applikationen von Drittanbietern sollten in dieser Oberfläche integriert und nicht mit einer eigenen Oberfläche zu bedienen sein. Der Hersteller könnte auch Hilfe bei der Integration von Oberflächen geben, in Form von speziellen Werkzeugen ([HEAB93], S. 354f). Innovative Anbieter habe in ihrem Management-System grafische 3-D-Oberflächen integriert und bieten Schnittstelle zum Management auf WWW-Browsern[5].
Damit die grafische Oberfläche Informationen bereitstellen kann, müssen grafische Elemente vorhanden sein. Können diese Ikons verändert werden, wie viele sind vorhanden? Können eigene grafische Elemente hinzugefügt werden ([GMD800], S. 15)?
Die Menüs sollten individuell konfigurierbar und mit einer Zugriffskontrolle versehen sein. Die Zugriffsinformationen sollen nicht nur den Zugriff auf Funktionen einschränken, sondern auch den Zugriff auf Ressourcen und Objekte. Das Konfigurieren der Menüs erleichtert den Umgang für ungeübte Benutzer, so kann dieser nur diese Menüpunkte auswählen, die seinem Wissensstand entsprechen und nicht ungewollt etwas verändern.

[5] Software, mit der auf das Informationsangebot des World Wide Web (WWW) zugegriffen werden kann.

Manche Funktionen müssen für einige Benutzer gesperrt sein (z.B. Entschlüsselung von Protokollen, Paßwortverwaltung). Die Menüpunkte für diese Funktionen sollten dann auch nicht erscheinen. Manchmal möchte die Geschäftsleitung nicht, daß das Personal bestimmte Daten erfährt (z.B. Telekommunikations-Management: Wer hat mit wem telefoniert?).
Man sollte auch benutzerspezifische Sichten anlegen können, die nur die Objekte präsentieren, die der Benutzer auch beobachten will ([VALTA91], S. 7). Allerdings müssen diese Sichten redundante Informationen enthalten können. Besteht die Restriktion, ein bestimmtes Objekt nur auf einer Sicht darstellen zu können, so sind erhebliche Funktionseinbußen hinzunehmen.
Leider können die Sichten oft nur nach Eigenschaften der Objekte erstellt werden (z.B. IP-Adresse, Funktion). Wünschenswert wären aber auch Sichten, die alle Geräte mit einer bestimmten Funktion darstellen (z.B. Druckerspooler). Denkbar wären folgende Attribute, nach denen Sichten generiert werden könnten:

- Dienst
- Attribute der Objekte
- physikalische Gesichtspunkte (z.B. Betriebssystem-Version 12)
- Aufgaben im Unternehmen (z.B. alle Rechner der Entwicklungsabteilung)
- Lokation (z.B. alle Drucker im ersten Stock)
- Geschäftsprozeß (z.B. alle Rechner, die zum Herstellen von Auto 1 benötigt werden)

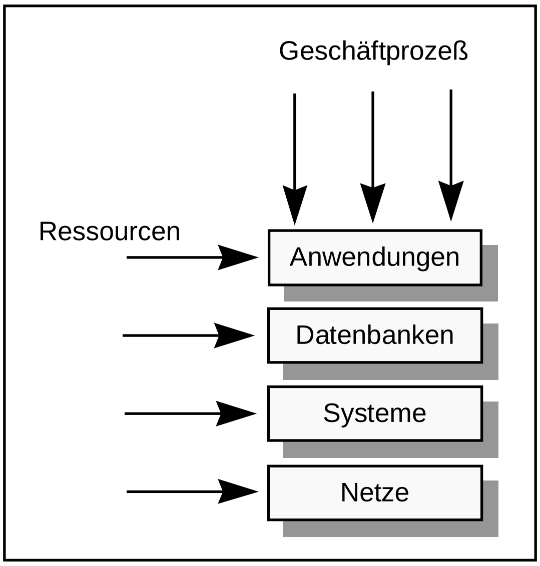

Abbildung 4.3: Geschäftsprozeß-Sicht

Die Geschäftsprozeß-Sicht erlaubt, die Probleme von Ressourcen mit den Geschäftsproblemen in Beziehung zu setzen. Abbildung 4.3 zeigt, daß der Geschäftprozeß die IT-Ressourcen als Ganzes betrachtet. Die traditionelle Sicht blickt auf verschiedene Ebenen jeweils getrennt, wie z.B. Netze, Systeme etc.
Ein schwieriger Punkt in der Darstellung von Management-Informationen ist das Anzeigen von Mobilkomponenten (z.B. Notebooks, Satelliten). Was passiert mit den Objekten, wenn sie momentan nicht im Netz sind? Werden Operationen auf solche Geräte gepuffert und zu einem späteren Zeitpunkt ausgeführt? Wie reagiert die Anzeige, wenn der Standort nicht berechnet werden kann oder wenn das Gerät je nach Lokation eine andere IP-Adresse hat?

Können Informationen und Warnungen auf andere Medien (z.B. Fax, Telefon) geleitet werden ([GMD800], S. 25)? Welche Mechanismen stehen zur Verfügung, um das Propagieren der Zustände über die Übersichtspläne zu steuern ([GMD800], S. 26)? Welche Zustände sind vorhanden, können diese konfiguriert werden? Hilfreich ist es, wenn der Administrator eigene Zustände definieren kann. Somit sind spezielle Reaktionen des Systems bei definierten Zuständen möglich, wenn die Management-Plattform eine Reaktion auf Zustandsänderungen ermöglicht ([VALTA91], S. 5).
Die Bedienbarkeit, die Optik und die intuitive Handhabung müssen ebenfalls ermittelt werden. Hier muß auch die Umgestaltung der Warnungen in akustische Signale oder Farbänderung berücksichtigt werden.

4.2.3 Anfrage-System

Ein wichtiger Aspekt bei der Überprüfung von Daten ist die Möglichkeit, eine Konsistenzprüfung durchzuführen. Bei einer Vielzahl von Geräten ist es nicht immer möglich, die Eindeutigkeit von Parametern netzweit zu garantieren. Zum Beispiel kann eine IP-Adresse mehrfach vergeben werden oder es sind an einem Switch angeblich mehr Maschinen angeschlossen, als Ports verfügbar sind. Dies kann bei folgendem Fall entstehen:
Manche Management-Plattformen erkennen Hubs ohne IP-Adresse nicht. Sind z.B. an einen 8-Port-Switch sieben Computer angeschlossen und ein Hub, an den wiederum drei Computer angebunden sind, dann scheinen zehn Computer direkt an den Switch angeschlossen zu sein.

Ein wichtiges Hilfsmittel bei der Planung von Netzen ist die stochastische Auswertung. Das fängt bei einfachen Anfragen an: Wie viele *3Com* Ethernetkarten wurden bisher installiert? Schwieriger, aber trotzdem nötig, sind Anfragen wie: Wie stark war der Prozessor ausgelastet, bei denjenigen Maschinen, die eine Paketfehlerrate von über 15 % hatten?
In diesem Segment sind natürlich Anfragen denkbar, die das Inventar, die Fehlererkennung oder das Lastverhalten betreffen.
Trendanalysen tragen dazu bei, langsam entstehende Flaschenhälse zu erkennen. Die Trendanalysen sind meist im Bereich des Leistungsmanagements angesiedelt. Eine mögliche Anfrage wäre: Welche Steigerungsraten ergeben sich bei der Netzauslastung zwischen Rechner A und B, gemessen über die letzten drei Monate?
Hier soll das Management-System erkennen, daß sich die Netzlast z.B. jeden Monat verdoppelt hat und daß ein kritischer Wert in ca. zwei Monaten erreicht ist. Diese Vorhersage kann nur wahr werden, wenn das Lastverhalten durch die Benutzer in Zukunft beibehalten wird. So kann die Netzplanung bereits im Vorfeld neue Hardware bestellen.
Eng mit dieser Fähigkeit gekoppelt ist das Erkennen von Engpässen. Oftmals ist nur bekannt, daß zwischen zwei Rechnern ein Engpaß in der Kapazität der Leitung besteht. Sind zwischen den zwei Rechnern verschiedene Leitungsarten (z.B. Ethernet, Modem, Glasfaser) vorhanden, sollte man erfahren, welche Leitung den Engpaß darstellt. Hier sollte die Plattform Unterstützung bieten.
Es gibt noch andere Problemfälle, die bei Netzen auftreten. Manchmal ist bei Leistungseinbrüchen nur eine bestimmte Dienstart betroffen (z.B. FTP funktioniert, HTTP nicht). Dies könnte sein, wenn die Pakete vom Dient FTP einen anderen Weg nehmen als die restlichen Pakete oder wenn z.B. der FTP-Server überlastet ist. In solchen Fällen sollte die Management-Plattform Funktionen bieten, die die Lastermittlung von einzelnen Diensten ermöglicht und die Leistung der Diensterbringer überwacht. Wenn die Plattform dann noch Vorschläge und Unterstützung beim Auslagern oder zeitlichen Verschieben (z.B. Backup) von Diensten bietet, dann ist dies ein zusätzlicher Pluspunkt.

4.3 Funktions-Standpunkt

Der Funktions-Standpunkt beinhaltet Kriterien zum Vergleich der Funktionalität von Management-Plattformen. Es gibt von OSI eine Funktionale Dimension mit den fünf Bereichen [ISO74984]. Diese Einteilung wurde auch für die Untergliederung des Funktions-Standpunktes verwendet. Allerdings sind noch weitere Einteilungen vorhanden. So wurde das Konfigurationsmanagement noch in Inventarisierung und Softwareverteilung unterteilt, weil eine genaue Inventarverwaltung bei einer großen Menge von Maschinen notwendig ist. Da zur Konfiguration eines Rechners auch die installierte Software gehört, wurde noch die Softwareverteilung hinzugefügt. Für Unternehmen, die sehr stark von ihrer EDV-Infrastruktur abhängig sind, ist der totale Ausfall des Rechenzentrums zu gefährlich. Deshalb wurde unter Fehlermanagement noch die Verfeinerung Disastermanagement[6] eingeordnet.
Bei modernen Management-Systemen werden einige Management-Anwendungen mitgeliefert, wie z.B. Ereignismanagement, Topologiemanagement, Netzmanagement und Entwicklungsumgebung, deshalb wurden diese als eigene Punkte im Funktions-Standpunkt eingeführt.
Außerdem wurden noch die Bereiche Speichermanagement[7], Anwendungsmanagement und WWW-Management[8] hinzugefügt. Diese haben heute durch die starke Verbreitung von WWW und SAP *R/3*[9] an Bedeutung gewonnen.

4.3.1 Konfigurationsmanagement

Das Konfigurationsmanagement muß die Möglichkeit bieten, die Konfiguration von Ressourcen zu speichern. Hilfreich ist es, wenn die gespeicherten Konfigurationsdaten bei einem neuen Gerät teilweise oder ganz wiederverwendbar sind. Außerdem sollten die Konfigurationsdaten auf andere Datenträger (z.B. Diskette) exportiert werden können ([VALTA91], S. 6). Somit ist es möglich, die Information an andere Geschäftszweigstellen zu versenden.
Die vorgenommenen Konfigurationsänderungen sollten bei den Ressourcen in Echtzeit vorgenommen oder, wenn gewünscht, zu einem bestimmten Zeitpunkt durchgeführt werden. Bevor die Daten zu den Geräten gesendet werden, sollte mindestens eine Plausibilitätsprüfung (z.B. eine IP-Adresse kann kein String sein) durchgeführt werden. Noch besser wäre es, wenn eine Konsistenzprüfung (z.B. ist die vergebene IP-Adresse schon vorhanden) durchgeführt wird. ([VALTA91], S. 5)
Die Management-Plattform sollte neue Geräte im Netz selbständig bemerken. Daraufhin sollten neue Ikons erscheinen, die mit zusätzlichen Informationen durch den Benutzer ergänzt werden können. Von Vorteil wäre, wenn dieser Vorgang durch einen Batchbetrieb übernommen werden könnte. In welchem Umfang können neue Maschinen erkannt werden und mit welcher Geschwindigkeit? Wird die versprochene Leistung erfüllt? Das Autodiscovery sollte bezüglich seines Funktionsumfangs konfigurierbar sein (z.B. mit welcher Technik werden neue Komponenten gefunden oder kann der Bereich auf IP-Adresse 129.187.214.0 - 129.187.214.50 eingeschränkt werden) ([GMD800], S. 22f). Werden Anwendungen oder Sicherheitssysteme erkannt, wenn dementsprechende Agenten vorhanden sind? Wo werden die neuen Ikons positioniert? Manche Management-Systeme geben die Möglichkeit ein automatisches Layout vorzunehmen ([GMD800], S. 15). Wie nah an der Realität sind die so entstandenen Übersichtspläne?
Nachdem die Komponenten erkannt wurden, sind eventuell noch nachträgliche Änderungen nötig. Einen einheitlichen Konfigurationsparameter (z.B. Kontaktperson) auf mehreren Ma-

[6] darunter versteht man das Verhalten in Extremsituationen (z.B. Brandfall)
[7] darunter versteht man die Sicherung und Verwaltung von Datenträgern
[8] darunter versteht man die Überwachung einer WWW-Server-Umgebung
[9] siehe Kapitel 2.3.1 und folgende

schinen gleichzeitig zu verändern, ist ebenfalls sehr hilfreich. Eigentlich sollten Sichten möglich sein, die Konfigurationsgruppen darstellen. Dies ist ein erster Schritt in Richtung VLANs[10], ein Gerät kann in mehreren dieser Konfigurationsgruppen enthalten sein.
Im wachsenden Windows-Umfeld ist es auch wichtig, Konfigurationen von Windows-PC abzufragen und zu setzen. Wichtig ist auch, wie detailliert die Konfiguration von Rechnern dargestellt werden kann. Läßt die Konfiguration eine Beurteilung des Rechners zu, wenn es zu Problemen kommt? Bei Konfigurationen von Routern ist auch eine grafische Darstellung der Routen sinnvoll.
Zu statistischen Zwecken sollte der Benutzer SQL-artige Abfragen starten können, wie z.B.: Welche Windows-PC benutzen eine 3Com Etherlink XL Netzwerkkarte?
So können einfacher Updates durchgeführt oder betriebswirtschaftliche Statistiken geführt werden. Solche Anfragen sollten auch im Batchbetrieb möglich sein, so daß automatisch Reports generiert werden können.
Das Management-System sollte Konfigurationsmanagement-Anwendungen von Drittanbietern integrieren können. Außerdem sollten diese Anwendungen über eine Schnittstelle auf die Datenbasis der Plattform zugreifen.
In diesem Bereich ist die Erstellung von Reports sehr sinnvoll, um sehr einfach einen Überblick über die Konfigurationen der Komponenten zu erhalten.
Die Inventarisierung und das Konfigurationsmanagement arbeiten sehr eng miteinander, da beiden die vorhandenen Ressourcen bekannt sein müssen.

Inventarisierung

Die vorher gesammelten Konfigurationsdaten sollten deshalb nahtlos der Inventarisierung zur Verfügung stehen. Die Information der Management-Objekte sollte um zusätzliche Informationen (z.B. Lieferantenname) ergänzbar sein ([GMD800], S. 17).
Bei sehr vielen Ressourcen müssen oft zusätzliche Informationen, wie z.B. Kontaktperson, Lieferant, Standort etc., gespeichert werden. Diese Informationen sollten beim jeweiligen Objekt als Attribut speicherbar sein. Ein zusätzliche Datenbank für solche Informationen würde sehr schnell zu Inkonsistenzen führen.
Eng damit gekoppelt ist das Management von Verwaltungsinformationen. Ein großes Unternehmen braucht Unterstützung bei der Verwaltung von Verträgen, Wartungsverträgen, Garantiezeiten etc. Das Management-System kann durch die zusätzlichen Informationen, sofern sie gespeichert werden können, Hilfe bieten.
Bei der Speicherung aller Daten sollte das Unternehmen die Wahl haben. Hier kann eine zentrale Haltung (bei Firmen mit kleinen Außenstellen) oder eine verteilte Speicherung sinnvoll sein.
Das Umziehen von ganzen Abteilungen oder Geräten sollte auf der Plattform im Vorfeld planbar sein. Durch diese Planung können dann Arbeitsaufträge erzeugt werden, die als Anleitung für den Umzug dienen.

Softwareverteilung

Im ersten Schritt müssen alle installierten Software-Komponenten erkannt werden, womit der Zusammenhang zur Inventarisierung besteht. Damit auf einem Rechner neue Software instal-

[10] VLANs (= Virtuellle LANs) sind eine Gruppe von Endgeräten, die auf verschiedenen physikalischen LANs sein können. Die Maschinen können aber kommunizieren, als wenn sie sich auf einem gemeinsamen LAN befinden. [3COM96]

liert werden kann, müssen aber auch die laufenden Prozesse bekannt sein. Sonst könnte es während der Installation zu Konflikten mit der laufenden Software kommen. Die bestehenden Rechte von Diensten und Programmen auf einem Rechner müssen ebenfalls erfaßt werden. Diese Rechte sollten bei der neuen Software wieder vorhanden sein.
Eine Abfragemaschine sollte eine bestimmte Gruppe von Rechnern abgrenzen und eine Installation einer neuen Software nur auf dieser Gruppe ermöglichen (z.B. welche Rechner haben den Grafikkartentreiber Version 123 installiert?).
Auf dieser erhaltenen Menge sollte dann eine Software-Installation möglich sein. Die Attribute für so eine Gruppenbildung sollten frei gewählt werden können. Die Software-Informationen sollten automatisch in Gruppen vorsortiert sein, wie z.B. alle Unix-Rechner oder alle Windows-PC.
Bei modernen Management-Anwendungen ist oft sehr viel Funktionalität nötig und vorhanden, deshalb läßt die Handhabung manchmal zu wünschen übrig. Die Installationsskripten müssen bei der heutigen Vielfalt an Produkten individuell programmierbar sein. Gleichzeitig muß aber eine gute Unterstützung gegeben werden, damit einfache Skripte ohne Probleme geschrieben werden können.
Für den Installationsvorgang müssen Kontrollpunkte frei definierbar sein. An diesen Kontrollpunkten wird dann überprüft, ob die Installation bis dahin richtig abgelaufen ist.
Der Administrator kann den Zeitpunkt der Installation nicht immer genau vorher bestimmen. Es müßten also Installationsintervalle (z.B. frühestens, spätestens) definierbar sein, in denen die Installation möglich ist.
Nach der Installation muß ein detaillierter Bericht erfolgen, was alles ersetzt oder gelöscht wurde.
Vorher sollten Reaktionen auf Fehler definiert werden können. Diese Ereignisse werden dann im Fehlerfall abgearbeitet.
Die neue Software muß auch mit Rechten zur Ausführung ausgestattet werden. Bei einem Administratorfehler darf die Softwaere allerdings keine Rechte zur Ausführung bekommen.
Der Administrator sollte ein Hierarchiekonzept für die Softwareverteilung definieren können. So wird die Software im ersten Schritt auf Abteilungsserver verteilt. Diese verteilen dann zeitgleich zu einem späteren Zeitpunkt die Software auf die Endgeräte. Der Administrator sollte hier jedoch die Wahl haben, wie viele Stufen er bilden möchte.

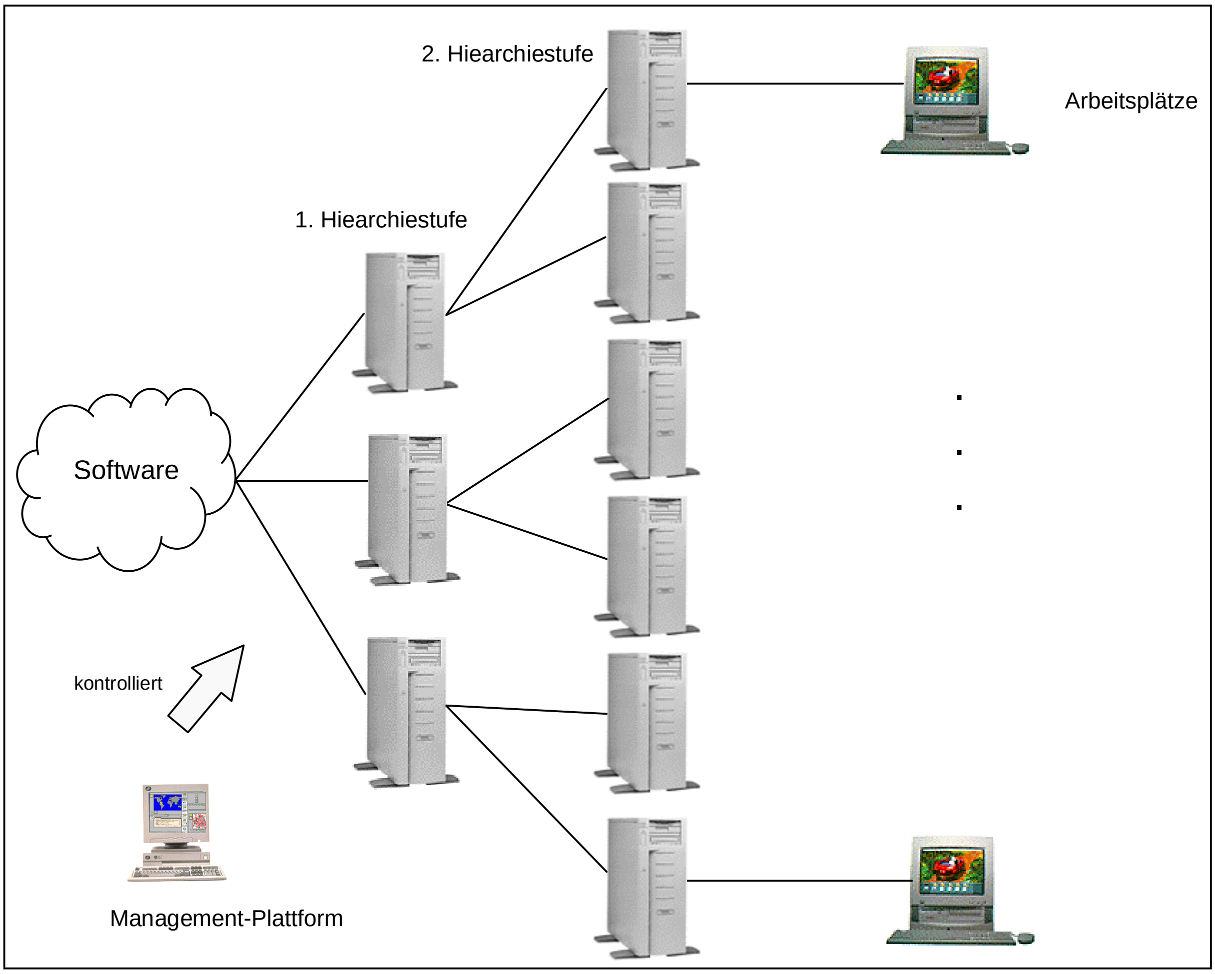

Abbildung 4.4: Hierarchiestufen

In diesem Bild wird eine Softwareverteilung mit zwei Hierarchiestufen dargestellt. Wenn die Software von einer Stufe zur nächsten kopiert wird, sollte dies möglichst zeitgleich geschehen. Die letzte Stufe nimmt dann die eigentliche Installation auf den Benutzer-Rechnern vor.

Besonders bei diesem Konzept ist es wichtig, daß eine große Anzahl von Betriebssystemen unterstützt wird. Der Installationsvorgang sollte zu jedem Zeitpunkt abgebrochen werden können. Nach solch einem Abbruch müssen die bisherigen Änderungen rückgängig gemacht werden können.
Der gesamte Installationsvorgang sollte gleichzeitig zur Informationsrecherche des betreffenden Rechners genutzt werden. Damit kann die Dauer der Beeinträchtigungen des Benutzers durch das Management gering gehalten werden.
Nachdem die neue Software installiert wurde, treten oft Bedienungsprobleme beim Benutzer auf. Eine Fernunterstützung durch den Administrator kann dem Benutzer bei ersten Problemen schnell Unterstützung geben.

4.3.2 Fehlermanagement

Manchmal ist es wünschenswert, daß nicht alle Fehler gemeldet werden. Deshalb sollte der Administrator den Fehlern Gewichtungen zuteilen können und die Art (z.B. Farbe, Akustik,

etc.), wie ein Fehler gemeldet wird, einstellen können. Außerdem sollte der Administrator konfigurieren können, welche Informationen bei Fehlern geliefert werden.
Die Intervalle, in denen Objekte überprüft werden, sollten frei definiert werden können. Die Menge der Objekte, auf die ein bestimmtes Überprüfungs-Intervall angewandt wird, muß der Administrator bestimmen können.
Damit ein Fehlermanagement bei allen SNMP-Komponenten durchgeführt werden kann, sollten Schwellwerte auf SNMP-Variablen gesetzt werden. Eine Fehlermeldung sollte aber auch dann erst entstehen, wenn mehrere Schwellwerte von verschiedenen SNMP-Variablen überschritten sind.
Es gibt Probleme, die sind erst bei längerem Auftreten interessant. Bei ISDN-Multiprimär-Anschlüssen ist z.B. der Anschluß oft für kurze Zeit (1-3 Sekunden - Meldung: Trunk down) nicht verfügbar. Solche kurze Ausfallzeiten muß der Benutzer akzeptieren und hat kaum die Möglichkeit der Beschwerde bei der Telekom. Erst bei mehrmaligem Auftreten oder wenn das Problem über einen längeren Zeitraum besteht, soll eine Warnung erzeugt werden. Es muß also ein Mechanismus im Management-System geben, der das Auftreten von Problemen protokolliert. Dann kann darauf ein Schwellwert gesetzt werden, so daß ab einer bestimmten Häufigkeit eine Fehlermeldung generiert wird.
Zur Lösung von Problemen sollte die Plattform Unterstützung bieten, damit Checklisten erstellt werden können, z.B. dem Problem „Rechner kann sich nicht an Server X anmelden“ könnte eine Checkliste wie folgt aussehen:

- Überprüfung des Ethernetkabels im Rechner, in der Dose
- Rangierfeld im Kabelschrank prüfen
- Kennung und Paßwort prüfen

etc.

Je nachdem wie detailliert die Checkliste ist, können auch einfache Überprüfungen von ungelernten Personen durchgeführt werden.
Damit die Zuständigkeit von Problemen auf mehrere Administratoren aufgeteilt werden kann, muß die Plattform die Probleme vorher in Fehlerklassen einteilen. Sind die Fehler eingeteilt, kann das Management-System eine Vorverarbeitung von Fehlern vollführen und die Fehler an die jeweiligen Administratoren verteilen.
Ist ein Router defekt, ist das dahinter liegende Netz nicht mehr ansprechbar. In diesem Fall sollte die Management-Plattform den defekten Router melden und informieren, daß die dahinter liegenden Rechner nicht erreichbar sind. Meldungen zu jeder Ressource, die in dem nicht mehr erreichbaren Netz liegen, sollten nicht erscheinen.
Eine große Arbeitserleichterung sind eigene Fehlererkennungs-Mechanismen der Plattform. Das Management-System sollte die Möglichkeit bieten, bei bestimmten Fehlern vordefinierte Skripten abzuarbeiten.
Gibt es noch andere spezielle Anwendungen zu diesem Bereich? Ist ein Trouble-Ticket-System integriert ([VALTA91], S. 5)?
Hier sollten auch die Aspekte aus dem Bereich Ereignismanagement eingehen, da diese zwei Bereiche sehr stark verbunden sind.

Disastermanagement

Können Systeme zum Ausgeben/Empfangen von Warnungen angeschlossen werden (z.B. Feuermelder, Routen von Meldungen zur Feuerwehr)? Gibt es einen Mechanismus, der im Brandfall die Routen der Arbeitsstation zum Ausfallrechenzentrum setzt? Es müssen auch Checklisten erstellt werden, die die Arbeit des Personals im Katastrophenfall unterstützen.

4.3.3 Leistungsmanagement

Bei Leistungs-Problemen in einem lokalen Netz ist es schwierig, den Problembereich näher einzugrenzen. Die Management-Plattform sollte z.B. die Leistung von bestimmten Kabelabschnitten gezielt messen können.
Leistungseinbrüche haben ein explosionsartiges Verhalten, die Beeinträchtigung wird oft erst nicht bemerkt. Erst wenn das Problem bewußt wird, ist meist gar kein Arbeiten mehr möglich. Wünschenswert ist die Lastermittlung von Diensten und ganzen Systemen (SAP *R/3*). Bei solchen komplexen Anwendungen reicht die Messung eines bestimmten Parameters an einer Abnahmestelle nicht aus. Es müssen mehrere Punkte definiert werden, wo verschiedene Messungen durchgeführt werden. Die Management-Plattform sollte die Entwicklung solcher komplexen Algorithmen ermöglichen und unterstützen.

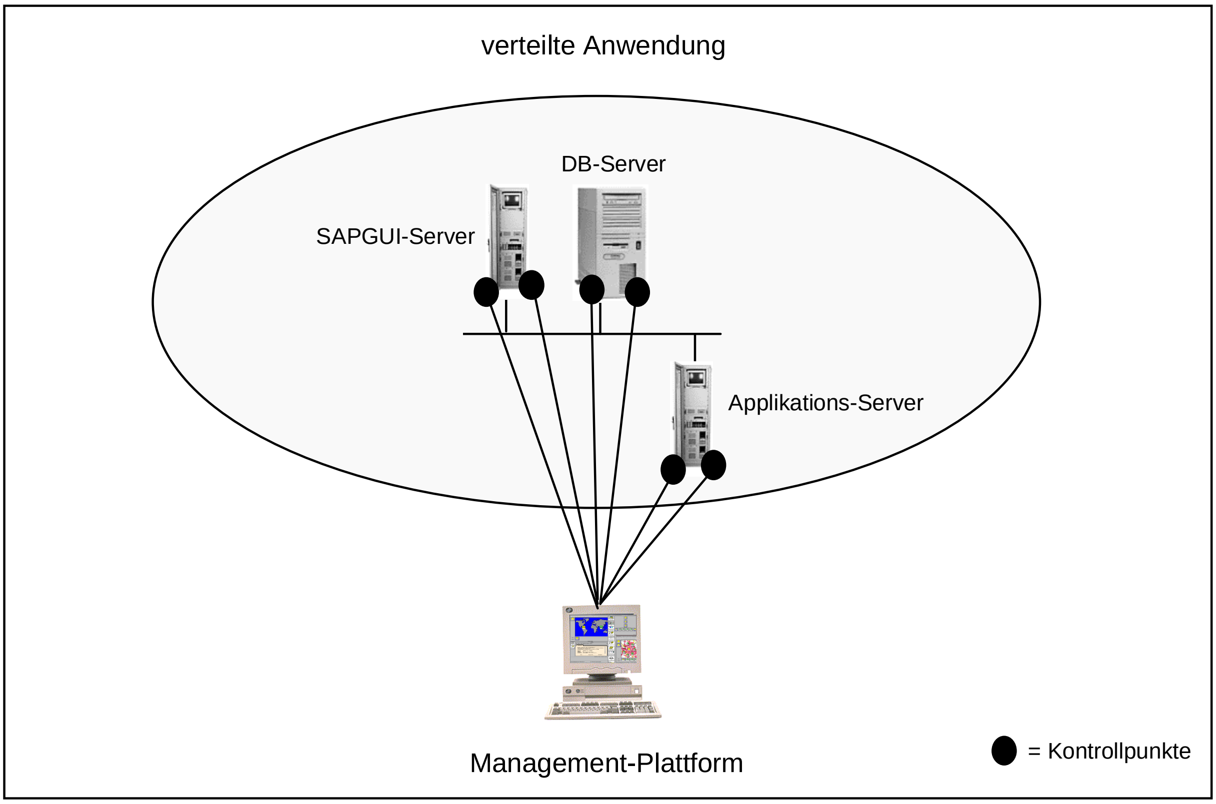

Abbildung 4.5: Kontrollpunkte bei verteilten Anwendungen

Mit der Plattform sollte man Schwellwerte für einzelne Leitungen setzen können. Somit ist das frühe Erkennen von Leistungseinbrüchen möglich. Die Schwellwerte müssen detailliert bezüglich Zeitraum, Intervall der zu messenden Variable und Speicherung konfiguriert werden. Außerdem sollte es möglich sein, daß ein oder mehrere Schwellwerte erreicht sein müssen, bevor ein Ereignis ausgelöst wird ([GMD800], S. 28).

Die Leistung von Netzkoppelelementen läßt sich heute schwer beurteilen. Die Management-Plattform sollte künstlich Last erzeugen können. Wird dazu spezielle Hardware benötigt? Das

Betriebssystem UNIX bietet bereits das Werkzeug *spray*, allerdings können damit nur UDP-Pakete an einen Ziel-Rechner gesendet werden. Fehlerhafte Pakete lassen sich damit nicht erzeugen. Der Administrator kann dann das Verhalten unter Last prüfen.
Bei der Messung von Leistungsparametern wird die Menge der Meßwerte schnell unübersichtlich. Die Messungen müssen grafisch darstellbar sein, um einen besseren Überblick über die Daten zu erhalten. Verschiedene Meßwerte sollten in einer Grafik darstellbar sein und auch in Echtzeit gezeichnet werden können. Meßwerte aus vorherigen Messungen sollten wieder dargestellt und für beliebige Zeit gespeichert werden können. Hier sind dann wieder die Punkte für Speicherung von Daten zu beachten und die Probleme, die dabei entstehen.
Es sollte Anfragemechanismen auf die beobachteten Maschinen geben, um bestimmte Problemfälle zu erkennen, wie z.B. SELECT * FROM isrouter WHERE memutil > 95 %[11]?
Die Plattform sollte eigene Lasttrends erkennen und berichten, wie z.B. Beobachtung der Last einer Leitung, und prüfen, in welchen Abständen sich die Last verdoppelt.

4.3.4 Abrechnungsmanagement

Beim Erfaßen der Verbrauchsdaten sollte der Administrator die Meßeinheit und die Metrik frei bestimmen können.
Nachdem die Kosteneinheiten gesammelt wurden, müssen sie noch auf einen geeigneten Kontenplan umgelegt werden. Die Umlage und die Erstellung von Kontenplänen sollte innerhalb der Management-Plattform möglich sein.
Nachdem die auflaufenden Kosten erfaßt sind, sollten diese durch Schwellwerte überwachbar sein. Die Reaktionen des Systems müssen beim Erreichen eines Schwellwertes definiert werden.
Bei Abrechnungsmanagement müssen einzelne Benutzer sogenannten „Profit Center“ zugeordnet werden können.
Das Management von Wartungs- und Leasingverträgen sollte im Abrechnungsmanagement seine Berücksichtigung finden.
Damit die erfaßten Daten im Controlling weiterverarbeitet werden können, müssen Export-Funktionen zu betriebswirtschaftlichen Anwendungen vorhanden sein. Besteht diese Möglichkeit nicht, sollten Reports generiert werden können.

4.3.5 Sicherheitsmanagement

Bei großen Netzen mit einer Vielzahl von Subnetzen wird es schwierig, die verfolgte Sicherheitspolitik durchzuhalten. Deshalb sollte die Management-Plattform beim Anlegen neuer Domänen oder neuer Subnetze Vorschläge machen, um eine gewünschte Sicherheitspolitik einzuhalten. Die Sicherheitspolitik eines bestimmten Objekts ist schwierig zu ermitteln, deshalb ist ein Werkzeug zur Visualisierung der Sicherheitskonzepte sinnvoll. Gerade die Vertrauensbeziehungen zwischen Netzen können große Sicherheitslücken darstellen, wie es z.B. bei *WindowsNT* üblich ist: Jede Domäne vertraut Benutzerkonten in der anderen Domäne, so daß diese die Ressourcen der Domäne verwenden können. Benutzer können sich von Computern beider Domänen aus an die Domäne anmelden, die ihr Konto enthält.
Falsch angelegte Benutzerkennungen können ebenfalls Lücken für illegale Eingriffe bieten, deshalb ist eine zentrale Benutzerverwaltung nötig. In dieser Anwendung sollen die Kennungen auf allen Rechnern angelegt werden.

---

[11] was bedeutet: Fasse alle Objekte in einer Menge zusammen, die das Attribut Router auf wahr gesetzt haben und deren Hauptspeicherbedarf über 95 % liegt.

Diese Anwendung sollte auch alle Sicherheitsverstöße, die im Netz auftreten, an zentraler Stelle halten. Die Genauigkeit, mit der nach Sicherheitsverstößen gesucht wird, muß vom Administrator einstellbar sein.
Die Unternehmen präsentieren sich im Internet, deshalb muß die Integration von Firewall-Anwendungen möglich sein. Der Administrator soll von der Plattform aus die Firewall überwachen und konfigurieren können.
Damit Sicherheitsverstößen besser nachgegangen werden kann, sollte das Management-System Sicherheitsreports erstellen können. Bei Verstößen können z.B. folgenden Fragen interessant sein: Welche Benutzer haben sich am Rechner X am 01.01.1997 angemeldet (z.B. die Datei „.lastlogin“ im Benutzerverzeichnis)?

Da eine gute Management-Plattform sehr viele Administrationsaufgaben unterstützt und dazu teilweise Administrator-Rechte nötig sind, stellt sie selbst ein erhebliches Sicherheitsrisiko dar. Die Sicherheitskriterien zu beschreiben, die für den Betrieb einer Management-Plattform zu erfüllen sind, ist nicht möglich. Ein solcher Kriterienkatalog würde den Umfang dieser Arbeit übersteigen. Im allgemeinen sollten Unternehmen auf die Angaben der Hersteller vertrauen können. Bei Anwendern, wie z.B. Banken, die ein hohes Bedürfnis an Sicherheit haben, sollte die Sicherheit von einem Institut (in Deutschland macht dies das Bundesamt für Sicherheit in der Informatik) überprüft werden [BSI396].

4.3.6 Ereignismanagement

Bei den Typen und der Anzahl der Ereignisse sollte ein Mittelweg gefunden werden. Das Subsumieren von vielen Ereignissen unter einer Ereignisart macht das Ereignis fast unbrauchbar. Sind jedoch zu viele Ereignisarten vorhanden, verliert der Administrator sehr leicht den Überblick und muß die Erläuterung zum Ereignis im Handbuch nachlesen. Durch eine individuelle Gewichtung der Ereignisse kann der Administrator bestimmten Zuständen eine höhere Priorität geben. Durch eine Wahl der Attribute, an denen die Gewichtung der Ereignisse gemessen wird, kann die Gewichtung feiner abgestimmt werden. Damit die entstehende Arbeit durch Ereignisse verringert werden kann, sollte die Plattform auf bestimmte Ereignisse, oder Ereignisse mit einer bestimmten Gewichtung, selbständig reagieren ([VALTA91], S. 5).
In einem großen Netz werden sehr viele Ereignisse erzeugt. Diese Vielzahl von Ereignissen würde in kürzester Zeit ein riesiges Datenvolumen annehmen. Somit muß ein Kompromiß bei der Speicherung der Daten gefunden werden. Das Management-System sollte es zulassen, daß die Speicherdauer von Ereignissen frei eingestellt werden kann, wobei verschiedene Arten von Schwellwerten möglich sein sollten (z.B. Datum, Speichergröße, etc.). Sind solche Schwellwerte gesetzt, soll das Management-System die Daten löschen. Natürlich muß der Administrator jederzeit die Möglichkeit haben die Löschung zu verhindern oder persönlich auszuführen.
Ein weiteres Werkzeug im Bereich Ereignismanagement ist eine Auswertungsmaschine für die empfangenen Ereignisse. Ist dieses Auswertungswerkzeug leistungsfähig genug? Kann das Abspeichern von unwichtigen Ereignissen damit verhindert werden? Können mehrere Ereignis-Filter gleichzeitig oder hintereinander angewandt werden? Existiert eine dokumentierte Schnittstelle, damit auch Anfragemaschinen von Drittanbietern auf die Ereignisse zugreifen können?
In diesem Bereich gibt es eine Vielzahl von Fragen, die im Umgang mit Ereignissen wichtig erscheinen. Die obigen Fragen sollen nur einen repräsentativen Ausschnitt geben. Hier muß das Unternehmen eigene Wünsche einbringen. Trotzdem sind nachfolgend einige wichtige Anforderungen an diesen Bereich formuliert.

Wurden die wichtigen Ereignisse ermittelt, dann sollten die Ereignisse noch verschiedenen Verantwortungsbereichen zugeordnet werden können. Sinnvoll ist es, wenn die Ereignisse der einzelnen Bereiche den verschieden Administratoren zugeordnet werden können. Natürlich muß eine gewisse Flexibilität bei dem Unterscheidungskriterium der Ereignisbereiche bestehen. Die Abbildung nach der die Ereignisse den Verantwortungsbereichen zugewiesen werden, sollte für den Administrator transparent sein. Auch eine Weiterleitung der Ereignisse an verschiedene Orte oder auf andere Medien (z.B. Fax) sollte möglich sein.
Bei der Vielzahl der Ereignisse bildet die selbständige Korrelation von Ereignissen durch die Management-Plattform eine große Erleichterung. Ein erster Schritt in diese Richtung ist, daß bei einem bestimmten Ereignis automatisch eine vorbereitete Batchdatei abgearbeitet wird. Das eigene Erkennen von Problemvarianten und das Ausführen abgewandelter Lösungsschritte wären wünschenswert.
Damit Probleme im Vorfeld erkannt werden können, ist das Setzen von Schwellwerten auf beliebige Variablen unabkömmlich ([VALTA91], S. 5). Wichtig ist, daß die Schwellwerte frei definiert werden können, ohne das bestimmte Erhöhungsschritte eingehalten werden müssen. Auch das Polling-Intervall für die einzelnen Schwellwerte sollte frei definierbar sein. Das Setzen von Schwellwerten auf eine Gruppe von Geräten sollte ebenfalls möglich sein.
Eine wichtige Hilfe für Trendanalysen ist die Langzeitmessung. Bei dieser Messung ist ebenfalls das Problem des aufkommenden Datenvolumens vorhanden. Es ist deshalb sinnvoll, das Speichern dieser Daten entkoppelt vom Speichern der Objekte durchzuführen. Die Datenbasis einer Langzeitmessung sollte auch entleert oder ausgelesen werden können, ohne die Langzeitmessung zu unterbrechen. Die Dauer einer Langzeitmessung soll konfigurierbar sein und Intervalle sollten ausgespart werden können (z.B. die Messung wird von 21.00-5.00 Uhr unterbrochen). Vielleicht stellt ein Hersteller noch zusätzliche Funktionen bereit, wie z.B. automatischer Neustart der Messung nach einem bestimmten Ereignis ([VALT91], S.5).
Grafiken können Informationen oftmals besser darstellen als Zahlen. Die Plattform sollte die Meßwerte grafisch darstellen können.

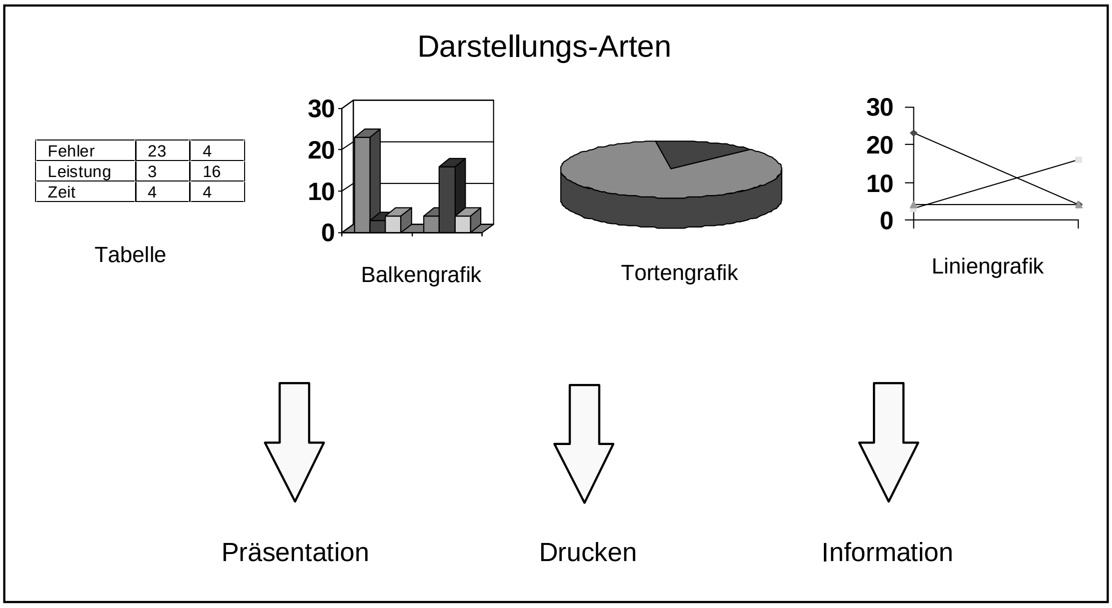

Abbildung 4.6: Meßwert-Darstellung

In der obigen Grafik werden Beispiele für die Meßwert-Darstellung gegeben, diese sollen z.B. helfen, Daten für eine Präsentation zu visualisieren. Außerdem können die Daten in leserlicher Form ausgedruckt werden oder durch die Übersichtlichkeit der Information kann der Administrator einfacher Entscheidungen treffen.
Für statistische Aussagen muß es möglich sein, die Daten korrelieren, kumulieren etc. zu können. Manchmal sind diese Werkzeuge aber nur rudimentär vorhanden, dann sollten aber Export-Funktionen zu statistischen und grafischen Werkzeugen vorhanden sein.

4.3.7 Topologiemanagement

Das Topologiemanagement sollte die Möglichkeit bieten, die Übersichtspläne zu verwalten ([VALTA91], S. 6). Das bedeutet, daß diese aus anderen Anwendungen importiert/exportiert (z.B. drucken) oder vorhandene verändert werden können. Oft findet man nicht gleich die richtige Darstellung realer Sachverhalte. Deshalb ist es sinnvoll, mehrere Versionen der Übersichtspläne zur Verfügung zu haben. So kann bei Bedarf auf eine gespeicherte Version zurück gegriffen werden. Der Administrator hat vielleicht auch die Möglichkeit sogenannte Snapshots zu erstellen ([GMD800], S. 15). Diese dokumentieren einen bestimmten Zustand des Netzes und sind nicht mehr veränderbar.
Ausschlaggebend beim Topologiemanagement ist auch die Qualität der grafischen Darstellung ([GMD800], S. 14). Darunter versteht man z.B. die Überschaubarkeit der Übersichtspläne, die automatische Anpassung der Größe an das Fenster. Werden Verbindungen zwischen Objekten automatisch erstellt und ist ein Beschriftung der Symbole bzw. der Übersichtspläne möglich?
Damit die Darstellung der Komponenten übersichtlich bleibt, muß der Benutzer eine Hierarchie anlegen können ([VALTA91], S. 7). Nützlich ist es, wenn der Benutzer über die Anzahl der Hierarchiestufen frei entscheiden darf ([GMD800], S. 13). Die Art der Hierarchisierung (also welches Gerät ist in welcher Hierarchiestufe) sollte ebenfalls frei wählbar sein.
Die Symbole sollten verwaltet werden können, d.h. Kopieren, Löschen, Verändern und neu Anlegen muß möglich sein ([GMD800], S. 14).
Bei guten Management-Plattformen sollten Funktionen vorhanden sein, um die Verkabelung zu kontrollieren und zu dokumentieren.

4.3.8 Netzmanagement

Es gibt viele Protokolle in den Netzen heutiger Unternehmen. Auch diese Protokolle haben eine bestimmte Konfiguration, so müssen IP-Datagramme beim Transport über FDDI unter anderm in Sub-Network Access Protocol (SNAP) Rahmen eingepackt werden ([RFC1103], S. 2). Für dieses Protokoll gibt es verschiedene Ausprägungen. Wurde bei verschiedenen Rechnern verschiedene Ausprägungen verwendet, kann es zu Fehlern im Netz kommen. Die Plattform muß ermitteln können, wenn sich ein neuer Client nicht an die gegebene Konfiguration hält.
Die verschiedenen Namensdienste (z.B. DNS, Domänen-Kontroller bei *WindowsNT*) müssen kontrollierbar sein.
Ein komplexes Gebiet im Netzmanagement ist das Routing. Speziell in großen Unternehmen sind oft Änderungen nötig. Das Setzen und Ändern von Routen möchte der Administrator durch die Plattform erledigen. Eine grafische Darstellung von Routen kann bei falsch gesetzten Routen sehr hilfreich sein. Bei verteilten Netzen sollten neue Routen auch über Remote-Adminstration gesetzt werden können.
Große Unternehmen sind durch eine Vielzahl von WAN-Verbindungen vernetzt, aber gerade hier können erheblich Probleme entstehen. Diese lassen sich aber nicht so einfach lösen, weil sie ihren Ursprung auf mehreren Leitungen haben kann. Es besteht also die Notwendigkeit,

WAN-Verbindungen zu beobachten und zu kontrollieren. Dazu muß die Management-Plattform neue Protokolle integrieren und mit WAN-Verbindungen umgehen können, die nicht immer zur Verfügung stehen. Die WAN-Verbindungen sollten hinsichtlich der Kosten überwacht werden, deshalb ist eine Schnittstelle zum Abrechnungsmanagement nötig.
Das Datenaufkommen zwischen zwei Stationen im Netz sollte angezeigt werden können. Die Plattform sollte eine Testmöglichkeit haben, in dem der Administrator sehen kann, wie sich neue Erweiterungen auf das Netz auswirken würden. Interessanter ist aber das Datenaufkommen, das durch einen Dienst entsteht. Wie viele Daten wurden z.B. zwischen Rechner A und B durch FTP übertragen?
Das Werkzeug Remote-Ping, bei dem UDP-Pakete zwischen zwei Stationen versendet werden, ermöglicht zumindest die Verfügbarkeit der Leitung zwischen zwei Rechnern zu prüfen.
Wichtig ist auch, ob das Management von Netzen über Router und WAN-Verbindungen hinweg möglich ist. Nicht alle Management-Plattformen können ohne besondere Einstellungen ein entferntes Subnetz kontrollieren. Ein Problem stellen hier auch die Router dar, denn diese können Pakete herrausfiltern, was der Überwachung aber nicht hilfreich ist.
Besondere Probleme entstehen bei kabellosen Verbindungen. Es sollen keine Warnungen generiert werden, wenn die Verbindung zeitweise nicht zur Verfügung steht. Dieser Umstand läßt sich durch individuelle Intervalle zur Überprüfung nicht beseitigen. Muß ein kabelloses Gerät (z.B. Satellit) möglichst immer zur Verfügung stehen, muß die Management-Verantwortung an andere Plattformen weitergereicht werden. Auch die gespeicherte Management-Information muß das andere Management-System erhalten.

Protokollanalysatoren

Die übertragenen Daten sollen über eine bestimmte Zeitperiode ermittelt, gefiltert und in Echtzeit dargestellt werden. Die Daten sollten auch gespeichert werden können.
Die Daten, die von einer bestimmten Station gesendet oder empfangen werden, sollten durch Filtermechanismen kontrollierbar sein. Das Abhören sollte beliebig eingeschränkt werden können, sonst wird die Information schnell unübersichtlich.
Können über eine bestimmte Zeitperiode die prozentuale Netzlast eines Dienstes oder die Anzahl fehlerhafter Bytes erfaßt werden?
Daten, die auf dem Netz gesendet werden, sollten entschlüsselt werden können. Der Protokollanalysator sollte die Information in lesbarer Form darstellen können. Die abgefangenen Datenströme sollten wieder auf das Netz zurück gesendet werden können.
Kabelverbindungen sollten mittels Time Domain Reflectometry getestet werden können.
Der Protokollanalysator muß künstlich Last erzeugen können, um das Netz oder Ressourcen unter Last testen zu können. Gleichzeitig muß die zu untersuchende Ressource überwacht werden.
Die oben genannten Forderungen geben gefährliche Möglichkeiten, um Informationen zu entschlüsseln und Geheimnisse auszuspionieren. Deshalb sind besondere Sicherheitsmaßnahmen nötig. Rudimentäre Paßwortabfragen sollten vom Protokollanalysator zur Verfügung gestellt werden.

4.3.9 Entwicklungsumgebung

Damit eigene Anwendungen programmiert werden können, müssen genügend und gut dokumentierte Schnittstellen vorhanden sein ([VALTA91], S. 7). Das Ansprechen der Schnittstellen sollte anhand von guten Beispielen gezeigt werden.

Programmbibliotheken sollten ebenfalls vorhanden sein, so daß der Entwickler auf eine möglichst vollständige Menge von Grundfunktionen zurückgreifen kann. Diese Bibliotheken müssen natürlich auch gut dokumentiert sein.
Welche Programmiersprachen werden unterstützt ([VALTA91], S. 8)? Ist die verwendete Programmiersprache eine eigene des Herstellers oder werden normierte Sprachen eingesetzt? Wenn es sich um eine eigene Programmiersprache handelt, sollten Schnittstellen zu anderen Sprachen vorhanden sein. Die Entwicklungsumgebung sollte Schnittstellen zu anderen Programmierwerkzeugen bieten, damit ein Unternehmen bereits vorhandene Werkzeuge wiederverwenden kann (z.B. Versionsverwaltung).
Besonders wichtig ist die Möglichkeit, eigene Agenten zu entwickeln ([VALTA91], S. 8). Die Anwender sollten Agenten mit eigener Intelligenz für Betriebssysteme und Anwendungen entwickeln können. Intelligente Agenten können helfen, die Management-Plattform zu entlasten[12] und komplexe Anwendungen können so kontrolliert werden.
Damit die Entwicklung von Software erleichtert wird, sollten Assistenten vorhanden sein, die ein grundlegendes Rahmenwerk zur Verfügung stellen. Bei der Programmierung unter *Windows* gibt es z.B. Funktionsmodule, die die Druckersteuerung übernehmen. Der Programmierer braucht nur dieses Funktionsmodul in seinen Quellcode einzubinden und kann dann mit seiner Anwendung den Drucker steuern.
Sind neue Agenten oder Anwendungen programmiert worden, müssen sie einem Test unterzogen werden. Dazu sollte der Programmierer die Möglichkeit haben, in einer Testumgebung die Software zu prüfen ([VALTA91], S. 8). Ein Einsatz in der realen Umgebung ist gefährlich, da durch falsche Programmierung Fehlfunktionen entstehen können.
Durch Eigenentwicklungen sollte die Release-Fähigkeit der Plattform nicht gefährdet sein. Dies ist natürlich nicht immer möglich, aber der Hersteller sollte nicht mit jeder neuen Version der Management-Plattform auch die Entwicklungsumgebung (dazu gehören Schnittstellen, Programmiersprache, etc.) ändern.

4.3.10 Speichermanagement

Bei großen Datenmengen sind regelmäßige Sicherungen auf andere Medien unverzichtbar. Die Management-Plattform sollte mit möglichst vielen Sicherungs-Systemen zusammenarbeiten. Manchmal sollen Medien verschlüsselt oder komprimiert werden, dies muß vom Management-System gesteuert werden können. Einige Unternehmen setzen Sicherungs-Roboter ein, um eine Sicherung 24 Stunden am Tag zu ermöglichen. Die Steuerung und Überwachung der Roboter sollte vom Administrator durch die Plattform möglich sein.
Weiterhin benötigt man Hilfe beim Einsatz der Sicherungs-Systeme. Sind vollständige oder partielle Sicherungen nötig oder die zeitliche Steuerung der Sicherungs-Jobs muß kontrolliert werden. Die erzeugten Sicherungen sollten in einer Datenbank verwaltet werden und die Archivierung von Daten auf billigere Medien sollte unterstützt werden. Für sensible Daten müssen auch auf den Sicherungs-Medien Zugriffsbeschränkungen eingesetzt werden. Wurden Daten archiviert, sollte ein transparentes Restore dieser Daten möglich sein. Die Daten werden für den Anwender unbemerkt wieder auf schnellere Medien (z.B. Festplatte) kopiert.

Bei Datenbanken oder Betriebssystemen können oft manche Daten nicht gesichert werden, weil die Anwendung gerade benutzt wird. Deshalb sollten für wichtige Anwendungen Agenten zur Verfügung stehen, die eine Sicherung und ein Restore der Daten ermöglichen, ohne die Anwendung zu stoppen.

[12] Siehe Kapitel 2.2.2

Bei verteilten Anwendungen, wie z.B. *R/3*-Systemen, ist eine Sicherung schwierig, da die Komponenten dieser Software auf mehreren Rechnern und Betriebssystemen verteilt sind. Ein anwendungsbezogenes Backup ist in solchen Fällen sehr hilfreich. Die Anwendungs-Hersteller bieten die Informationen, um ein solches System komplett sichern zu können.

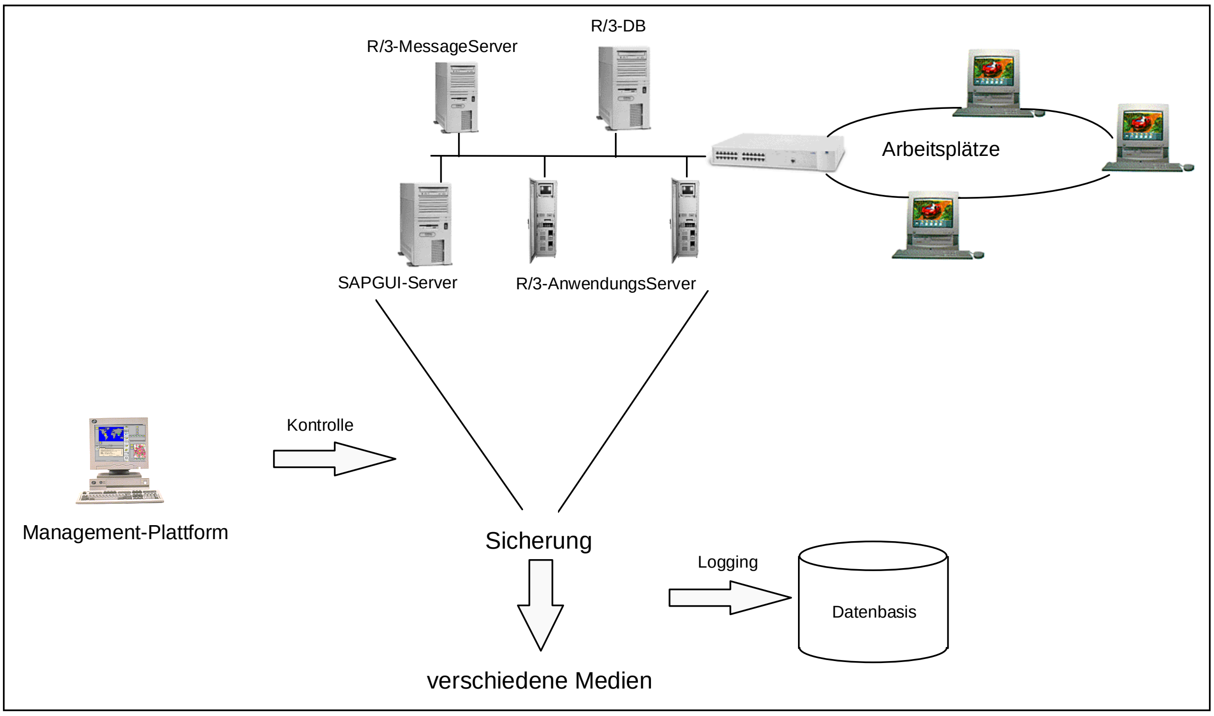

Abbildung 4.7: Anwendungsbezogene Sicherung

In Abbildung 4.5 werden bei der Sicherung automatisch die nötigen Daten von allen verschiedenen Rechnern gesichert. Dabei können die Daten auf auch verschiedene Medien gesichert werden. In einer Datenbasis wird protokolliert, auf welchen Medien welche Daten liegen. Außerdem werden alle Information zu dieser Sicherung unter einem Namen zusammengefaßt. Der Administrator muß also nur noch diese Sicherung auswählen und erfährt sofort, welche Medien und Ressourcen dazu nötig sind.

4.3.11 Anwendungsmanagement

Bei großen Anwendungen gibt es meist eine eigene Benutzerverwaltung, diese bewirkt aber einen hohen Administrations-Aufwand. Die Benutzerverwaltung der Anwendung sollte mit den Zugangsberichtigungen der Server zentral an einem Punkt verwaltet werden. So kann z.B. die Authentifizierung bei ISDN-Routern mit der Software *Radius* mit den Rechten eines UNIX-Rechners (die Datei passwd) gekoppelt sein. Die Benutzer-Kennungen, die eine Anwendung nutzen, sollten angezeigt und informiert werden können. Im R/3-System wird z.B. jede Benutzer-Kennung mit der momentanen Transaktion angezeigt. Bei Bedarf kann der Administrator einem Benutzer eine Nachricht senden. Leider muß dazu aber die R/3-Oberfläche bemüht werden.

Die zentrale Frage beim Anwendungsmanagement: Ist die Anwendung verfügbar? Der Administrator sollte verschiedenen Zustände einer Anwendung definieren können. Die Management-Plattform soll diese Anwendung dann überwachen und den aktuellen Zustand anzeigen. Da die

Meßpunkte für die Verfügbarkeit von Anwendung zu Anwendung variieren, teilweise sogar von Version zu Version, muß der Administrator die Überwachungspunkte vorgeben können.
Die Semantik und die Schwellwerte für solche Meßpunkte müssen ebenfalls vom Menschen vorgegeben werden. Das Ziel ist allerdings, daß der Hersteller dieser Anwendung die Meßpunkte liefert und diese an die Management-Plattform weiterreicht. Die Hersteller von Management-Plattformen könnten aber auch Agenten für die wichtigsten Anwendungen liefern. Der Administrator müßte dann nur noch dafür sorgen, daß der Agent zyklisch abgefragt wird.
Eine Anwendung sollte von der Plattform aus gestoppt und gestartet werden können, besser wäre noch, dies durch einen Zeitplaner automatisch machen zu können. Manche *R/3*-Systeme sollte in der Nacht beendet werden, damit auch die Datenbank gestoppt und gesichert werden kann.
Große Anwendungen können eine erhebliche Last im Netz erzeugen, diese Last sollte ermittelbar sein. Hier müssen Schnittstellen zum Leistungsmanagement bestehen, so daß die Management-Anwendung überprüfen kann, welche Last von einer Anwendung erzeugt wird. Eine Anwendung bietet oft auch mehrere Dienste an, vielleicht kann die Last dieser Dienste ermittelt werden. Durch die erzeugte Last von Anwendern, wird die Leistungsfähigkeit der Anwendung beeinflußt. Kann die entstehende Last von den Benutzern ermittelt werden, kann der Administrator den Ressourcen-Verbrauch solcher Anwendungen besser planen.
Ebenfalls in diesen Bereich gehören Informationen über Ressourcen-Verbrauch oder laufende Prozesse. Es gibt sehr oft Abhängigkeiten zwischen einer Anwendung und anderen Software-Komponenten (z.B. Betriebssystem, Bibliotheken), deshalb ist der aktuelle Versionsstand der Anwendung wichtig. Von Vorteil wäre es, wenn die Management-Plattform diese Abhängigkeiten automatisch überprüfen und Warnungen bei Änderungen generieren könnte.

4.3.12 WWW-Management

Viele Unternehmen betreiben heute einen eigenen WWW-Server, bei manchen ist dieser sogar ein eigener Vertriebskanal.
Da solche WWW-Server vom Internet erreichbar sein müssen, stellen diese eine besondere Anforderung an die Sicherheit. Der Administrator soll mit Hilfe der Management-Plattform die Zugriffsrechte von Dokumenten des WWW-Servers ändern können. Der WWW-Server stellt Dokumente zur Verfügung die vom Internet oder nur vom Intranet erreichbar sind. Diese Trennung sollte sich durch die Plattform dokumentieren und genau steuern lassen, wo die Sicherheitsmaßnahmen greifen sollen. Es besteht auch ein direkter Zusammenhang zwischen der Firewall und dem WWW-Server. Der WWW-Server kann z.B. nur auf einem bestimmten Port-Anfragen entgegen nehmen. Soll er dies auch aus dem Internet tun, muß die Firewall Verbindungen mit diesem Port durchlassen.
Wird vom Unternehmen mit dem WWW Handel betrieben, sind Zugriffs-Statistiken auf Dokumente sinnvoll. Der benötigte Speicherbedarf der WWW-Dokumente sollte in der Management-Plattform auch ersichtlich sein.

4.4 Technologischer Standpunkt

In diesem Standpunkt sind Anforderungen beschrieben, die sich an die Technik einer Management-Plattform richten. Dazu gibt es die drei Teilbereiche Hardware, Software und Installation. Die Mindestvoraussetzungen an die Hardware wurde bereits im Unternehmens-Standpunkt verifiziert (damit am Anfang gleich die Voraussetzungen für eine Installation ersichtlich sind), deshalb sollen hier die Kriterien Ansprüche prüfen, die sich durch zusätzliche Funktionalität ergeben.

Der Gliederungspunkt Installation beinhaltet Anforderungen, die bei der Installation der Plattform notwendig sind. Allerdings ist noch die Verfeinerung **verteilte Installation** vorhanden, hier geht es um die Erbringung einer verteilten Management-Funktionalität.

4.4.1 Hardware

Die Hardware, auf der die Management-Plattform läuft, sollte die Möglichkeit bieten, Protokollanaylsatoren anschließen zu können. Die sonstigen Forderungen (wie Bandlaufwerk, Netzwerkkarten) sollten bei moderner Hardware ohne Probleme integriert werden können.
Außerdem sollte die Maschine möglichst viele verschieden Netzadapter (z.B. Token Ring, Ethernet, FDDI etc.) integrieren können, damit man sich mit dem Management-System direkt (also ohne Netzkoppelelemente) in das Netz einbinden kann.

4.4.2 Software

Bei den heutigen heterogenen Unternehmensnetzen werden viele verschiedene Betriebssysteme eingesetzt. Diesem Trend müssen auch die Plattformen folgen, deshalb sollte die Client-Software, also die „Arbeitsstation“ auf möglichst vielen Betriebssystemen mit voller Funktionalität vorhanden sein.
Der Ressourcenverbrauch dieser Client-Software sollte möglichst gering sein. Denn der Nutzen einer Software, die alle Ressourcen benötigt und somit das Arbeiten mit anderer Software unmöglich macht, ist fraglich.
Allgemein kann man fordern, daß die Management-Plattform möglichst viele Schnittstellen zu anderer Standardsoftware besitzt. Unternehmensdaten werden heute in den verschiedensten Anwendungen weiterverarbeitet, deshalb muß die Plattform diesem Wunsch nachkommen.
Die Administratoren müssen meist Netze mit den verschiedensten Topologien kontrollieren. Deshalb sollten möglichst viele andere Protokolle unterstützt werden. Es dürfen aber nicht nur Protokolle wie TCP/IP oder IPX/SPX[13] sein, sondern es sollten auch verschiedene Management-Protokolle wie SNMP oder CMIP[14] sein([VALTA91], S. 3).

Die Management-Plattform sollte außerdem möglichst viele Agenten zur Verfügung stellen. Diese Agenten sollten erweiterte Funktionen besitzen (z.B. spezielle Agenten für *WindowsNT*-Rechner, die die internen Ereignisse des Betriebssystems an ein Management-System weiterleiten). Außerdem sollten Agenten für wichtige Anwendungen existieren oder sich solche Agenten von Drittanbietern integrieren lassen. Natürlich sollten diese Agenten für möglichst viele Betriebssysteme vorhanden sein. Vielleicht können die Agenten auch eine bestimmte Management-Politik verfolgen, eigene Korrelationen vornehmen, bestimmte Ereignisse filtern und durch eigene Intelligenz auf definierte Ereignisse selbst reagieren.
Bei so viel geforderter Funktionalität besteht aber die Gefahr, daß die Agenten zu viele Betriebsmittel auf dem Ziel-Betriebssystem verbrauchen.
Der Austausch von Informationen zwischen Management-Plattformen von verschiedenen Herstellern sollte auch möglich sein ([VALTA91], S. 4). Welche Protokolle können dafür verwendet werden oder gibt es dazu proprietäre Lösungen? Die Kriterien für eine Verteilung der Management-Funktionalität, die durch mehrere Management-Systeme gleichzeitig erbracht wird, sind im Abschnitt „verteilte Installation“ beschrieben.

---

[13] Internetwork Packet Exchange/Sequenced Packet Exchange sind NetWare-Transport-Protokolle auf Ebene 3 und 4
[14] Common Management Information Protocol ein OSI-Protokoll für den Transport von Management-Informationen

Ein wichtiger Punkt sind die unterstützten Netzmanagement-Architekturen. Die zwei wichtigsten Vertreter sind das OSI-Netzmanagement ([HEAB93], S. 129ff) und das Internet-Management ([HEAB93], S. 157ff), welche zumindest vorhanden sein sollten. Aber die Plattform sollte auch neue oder andere Architekturen (z.B. Telecommunications Management Network von CCITT) integrieren können. Es sollte auch überprüft werden, wie vollständig die Konzepte einer Netzmanagement-Architektur implementiert wurden und ob diese durch eigene Erweiterungen (wobei die Richtlinien der Architektur eingehalten werden müssen) verändert werden können.

Die heutigen Unternehmen betreiben einen sehr großen Aufwand, um all ihre Dokumente zu verwalten. Vor einigen Jahren wurden dies Dokumente deshalb oft auf Mikrofilmen archiviert. Bei dieser Archivierung war es jedoch sehr umständlich auf alte Dokumente zuzugreifen. Heute werden deshalb Datenbanken als Archivierungssysteme mit billigen Datenträgern verwendet. Solche Archivierungssysteme müssen aber auch kontrolliert werden. Vor allem die Schnittstellen zu den anderen Anwendungen und die Zugriffskontrolle benötigen erhöhte Aufmerksamkeit. Vielleicht ermöglicht die Plattform selbst ihre Dokumente (z.B. alte Log-Dateien) zu archivieren? Das Management-System sollte hier eine enge Zusammenarbeit mit dem Speichermanagement ermöglichen, denn schließlich sollten alte Dateien auf billigere Datenträger archiviert werden.
Die Fehler die in einem großen Netz entstehen, müssen ebenfalls verwaltet werden, dazu sollten ein Trouble-Ticket-System oder Schnittstellen vorhanden sein. Steht ein solches System zur Verfügung müssen einige Funktionen, wie z.B. Zuweisung (ein Mitarbeit ist für bestimmte Probleme zuständig), Benachrichtigung, etc. vorhanden sein. Die erfaßten Probleme sollten sich auch statistisch auswerten lassen, wobei hier eine Zusammenarbeit mit anderen statistischen Modulen gefordert ist. Wichtig ist auch, daß dieses System eine Datenbasis für bereits gelöste Probleme darstellt. Es muß für den Benutzer möglich sein, diese Informationen auf einfache und schnelle Weise wieder zu finden.
Große Unternehmen müssen sich heute immer wieder vor Computer-Viren schützen. Dazu gibt es auch einige Produkte für die permanente und die geplante Überwachung. Die Virus-Überwachung sollte netzweit möglich sein und automatisiert ablaufen können. Der Administrator muß die Überwachung durch einen Zeitplan steuern können, um sie mit anderen Diensten (z.B. Sicherung) koordinieren zu können. Da sehr schnell immer wieder neue Viren entstehen, muß sich das Anti-Virus-Modul sehr leicht erweitern lassen.
Vielleicht bietet das Management-System noch zusätzliche Werkzeuge, die hier noch eine Erwähnung finden ([VALTA91], S. 6). Die Entwicklung in diesem Bereich ist sehr rasant, so daß in kürzester Zeit schon völlig neue Ideen verwirklicht werden können.

4.4.3 Installation

Auf welchem Medium wird die Software ausgeliefert und ist damit eine komfortable Installation möglich?
Oftmals sind viele Änderungen im System nötig, diese Änderungsschritte müssen gut beschrieben sein, damit der Administrator diese überwachen kann. Die ganze Installation sollte ohne große Fachkenntnisse möglich sein. Noch komfortabler wäre eine richtiges Installationsprogramm, das die nötigen Fragen stellt und alle Änderungen vornimmt.
Manchmal müssen zur Installation bestimmte Voraussetzungen (z.B. Betriebssystem-Version, installiertes GZIP) erfüllt sein, diese sollten vorher genau beschrieben werden. Auf eine bestimmte Reihenfolge (z.B. zuerst die Datenbank) der Installation sollte auch explizit hingewie-

sen werden. Die nötigen Voraussetzungen sollten automatisch überprüft werden, wie z.B. ist genug Festplattenplatz vorhanden oder ist genug Hauptspeicher installiert
Bei großen Unternehmen werden viele verschiedene Kommunikations-Protokolle verwendet, deshalb sollte die Management-Plattform möglichst viele Protokolle direkt unterstützen. Prinzipiell können auch Proxy-Agenten[15] aufgebaut werden, diese stellen aber einen Flaschenhals dar und bieten einen zusätzlichen Ausfallpunkt im Management. Trotzdem sollte die Möglichkeit bestehen, daß die Plattform als Proxy-Agent arbeiten kann.
Die Dauer einer Standard-Installation und die Anwenderanpassung an die gegebenen Verhältnisse sind wichtige Kriterien. Diese Anpassung wird natürlich von Unternehmen zu Unternehmen variieren. Ein erfahrener Lieferant sollte hier aber eine Durchschnittsangabe machen können. Bei manchen Unternehmen ist der Einsatz einer Management-Plattform dringend, dazu dauert die Anwenderanpassung aber zu lange. Die Anpassung sollte deshalb modulweise oder schrittweise möglich sein. Bei manchen Anwendungen ist die Anpassung sogar so schwierig, daß sie ohne externe Hilfe nicht zu bewältigen ist. Hieraus ergeben sich eine Menge Fragen, z.B. was kostet die Anpassung oder können danach eigene Änderungen vorgenommen werden etc.?
Sollte die Management-Plattform wieder von dem betreffenden Rechner entfernt werden, sollte dies ohne Neuinstallation des Betriebssystems möglich sein. Dies ist bei Rechnern mit *WindowsNT* nicht selbstverständlich, da installierte Bibliotheken eventuell auch von anderen Anwendungen benutzt werden ohne dies zu wissen. Werden solche Bibliotheken entfernt, funktionieren auch die anderen Anwendungen nicht mehr.

Verteilte Installation

Bei verteilten Lokationen, die über Leitungen mit geringer Kapazität verbunden sind, ist eine verteilte Installation der Management-Plattform nötig. In sehr großen Unternehmen reicht eine verteilte Installation nicht aus, vielmehr müssen die Management-Aufgaben verteilt erbracht werden. Es müssen mehrere Plattformen die Funktionalität erbringen.
Ein einfache Lösung ist die RMON-MIB (Remote Network Monitoring-MIB) und die Verwendung von Probes[16]. Welche RMON-MIBs werden unterstützt, RMON-MIB, RMON2-MIB oder Enterprise-RMON-MIBs? Wird der volle Umfang dieser MIBs geliefert oder steht nicht der volle Funktionsumfang zur Verfügung? Hilfreich ist auch, wenn Probes von verschiedenen Hersteller eingebunden werden können. Damit eine echte Verteilung möglich ist, sollten die Probes von der Management-Plattform konfiguriert werden können. Um die Einstellung einer Probe zu ändern, sollte nicht ein Mitarbeiter extra vor Ort sein müssen.

In der Literatur ([LCPW89], [COLE96], [HEGE93])[17] gibt es noch andere Ansätze, um die Management-Funktionalität zu verteilen. Eine Möglichkeit ist die verteilte Management-Struktur, bei dieser sind folgende Anforderungen wichtig:
Wird eine effiziente Technik genutzt, um eine Replikation der Datenbank durchzuführen? Die Management-Informationen sollten in der nächsten Datenbasis gespeichert werden, diese sorgt dann selbständig für eine Replikation der Information in die anderen Datenbanken.

[15] Der Proxy-Agent sammelt die Mangement-Information und stellt sie dem Manager in der von ihm erwart e-ten Weise bereit
[16] siehe dazu Kapitel 2.2.2 - RMON-MIB
[17] siehe dazu Kapitel 2.2

Der Peer-Manager sollte möglichst viel Funktionalität bieten, dabei aber nicht mit unnötigen Funktionen überladen sein. Ein dynamisches Konfigurieren der Funktionen wäre sinnvoll. Vielleicht gibt es aber eine technische maximale Grenze beim Einsatz von Peer-Manager.
Ein Problem ist die Bildung der Management-Domänen. Der Administrator sollte möglichst keinen Einschränkungen unterliegen.

Eine weiteres Modell ist die hierarchische Management-Struktur, sie benötigt folgende Anforderungen:
Obige Kriterien für den Peer-Manager, der hier Domänen-Manager heißt, gelten auch hier. Interessant ist aber auch die Frage, ob eine vollständige Management-Plattform als Domänen-Manager eingesetzt werden kann. Da der MOM[18] und die Domänen-Manager viel Informationen austauschen, sollte zwischen diesen beiden Partnern ein optimiertes, möglichst nicht propritäres Protokoll verwendet werden. Damit das Verkehrsaufkommen weiter verringert werden kann, sollte der Domänen-Manager die Management-Daten vorverarbeiten und interpretieren. Bei sehr vielen Domänen-Managern bedeutet die Konfiguration dieser sehr viel Arbeit. Der MOM sollte die Domänen-Manager konfigurieren können, d.h. die Konfiguration des Netzmanagement-Clients sollte von der Management-Plattform aus eingestellt werden können.
Besteht die Möglichkeit, eine Backup-Datenbank aufzubauen, die erst beim Ausfall der ersten Datenbank zum Einsatz kommt?

Die dritte Art, verteiltes Management durchzuführen, ist eine Netzwerk-Struktur:
Hier gelten ebenfalls die Anforderungen der vorherigen Strukturen. Eine Netzwerk-Struktur muß verteilte Anwendungen über mehrere Manager ausführen können. Da es keine einheitliche Arbeitskonsole gibt müssen alle Funktionen von jeder Station erbracht werden können. Dies bedeutet, daß die Station, an der eine Funktion angefordert wird, diesen Wunsch an die anderen Manager weiterreicht. Ein Manager wird die Funktion erbringen können und sich mit einer entsprechenden Antwort melden, dies kann natürlich auch die erste Station sein.
Die Management-Schnittstelle für die Kommunikation zwischen den Managern sollte standardisiert sein, somit ist der Einsatz von verschiedenen Herstellern als Manager möglich. Ein besonders wichtiger Punkt ist die Anfragemaschine für Management-Informationen. Da die gewünschte Information auf mehreren Managern verteilt sein kann, müssen die Manager eine einheitliche Schnittstelle für dieses Anfragesystem bieten.

Diese ausführlichen Erklärungen sollten den Kriterienkatalog im Anhang A verständlich und anwendbar machen. Damit dieser Kriterienkatalog auf seine Tauglichkeit überprüft werden kann, werden im nächsten Kapitel zwei Plattformen anhand des Kriterienkataloges verglichen. Außerdem soll durch den Test die Anwendung des Kriterienkataloges verdeutlicht werden.

[18] „Manager-of-Managers“ siehe Kapitel 2.2.2

5 Anwendung des Kataloges

Der gesamte Kriterienkatalog ist sehr umfangreich, mit einigen sehr speziellen Gebieten. Trotzdem sollte der Katalog im Rahmen dieser Arbeit bezüglich seiner Anwendbarkeit geprüft werden.
Als Testumgebung wurde das Netz des Lehrstuhls Kommunikationssysteme und Systemprogrammierung verwendet. Es handelt sich dabei um zwei Backbone-Switches, die die Verbindung zum Leibniz-Rechenzentrum ermöglichen. Ausgehend von diesen Switches sind sternförmig weitere Switches und Hubs angeschlossen.

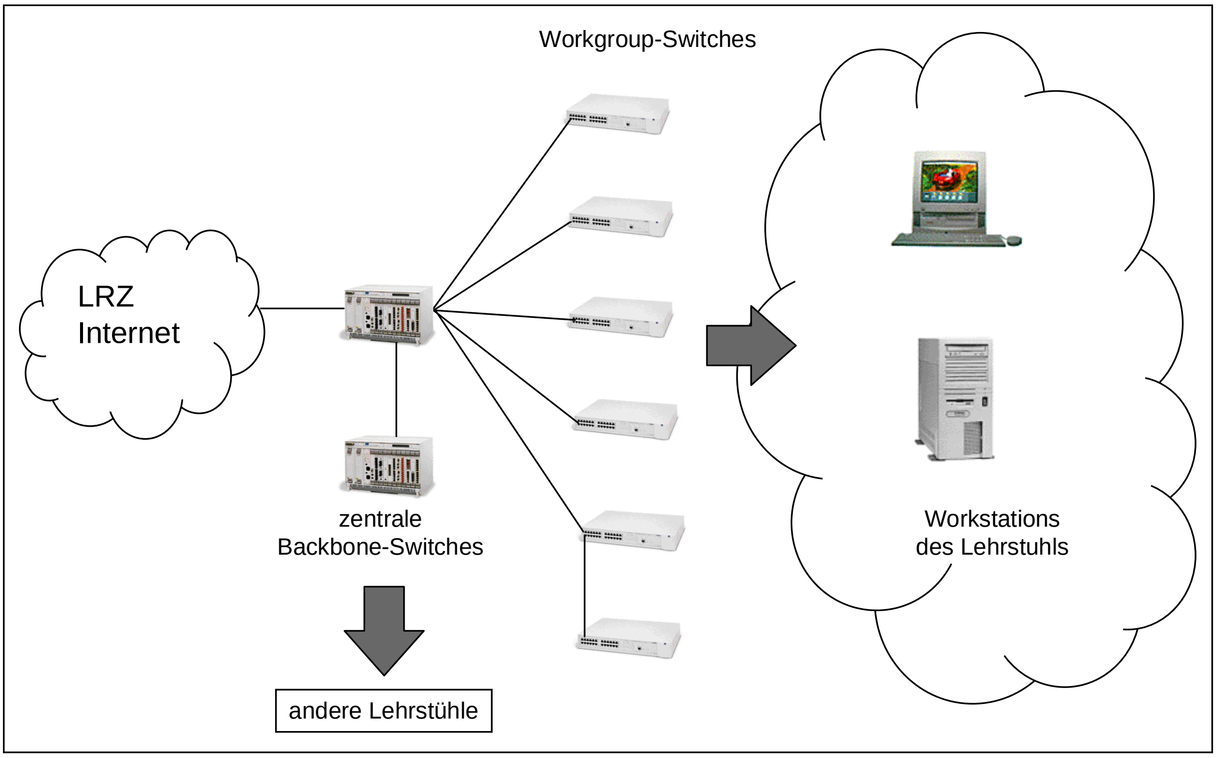

Abbildung 5.1: Lehrstuhlnetz

Das obige Lehrstuhlnetz sollte in den Produkten IBM *NetView* und HP *OpenView* nachgebildet werden. Aufgrund des Katalog-Umfangs und fehlender Testmöglichkeit, konnten nicht alle Module der Plattformen getestet werden. Deshalb mußte die Aufgabenstellung etwas eingeschränkt werden und nur einige Management-Bereiche konnten geprüft werden.

5.1 Aufgaben

Es wurden einzelne Aufgaben gestellt, mit denen die Fähigkeit der Plattformen in verschiedenen Bereichen getestet werden sollte. Für alle getesteten Management-Plattformen waren die Aufgabenstellungen gleich. Des weiteren wurde mit der Dokumentation gearbeitet, um diese auf ihre Vollständigkeit zu untersuchen. Die Aufgaben werden in den folgenden Abschnitten erklärt.

5.1.1 Darstellung

Das Lehrstuhl-Netz sollte mit der Management-Plattform grafisch dargestellt werden, wobei möglichst eine physikalische Sicht (basierend auf IP-Adressen und Verkabelung) und eine logische (basierend auf Arbeitsgruppen) erstellt werden sollte.
Durch diese Aufgaben ließen sich die Fähigkeiten der Plattform beim Darstellen von Objekten überprüfen. Der Umgang mit der grafischen Oberfläche wurden ebenfalls getestet.
Bevor irgendwelche Ressourcen verschoben oder angelegt wurden, konnte anhand der vorhandenen Objekte bestimmt werden, wie leistungfähig die „Autodiscovery-Funktion“ ist.
Beim Darstellen des Lehrstuhl-Netzes sollten folgende Fragen untersucht werden:

1. Wie sind der Umgang und die Handhabung mit Objekten?
2. Kann eine logische Sicht erstellt werden?
3. Ist es möglich, eine dienstbezogene Sicht anzulegen?
4. Sind mehrere Sichten möglich?
5. Welche Möglichkeiten gibt es, eine Hierarchie von Sichten zu bilden?
6. Können die Übersichtspläne den eigenen Bedürfnissen (optische Erscheinung) angepaßt werden?
7. Wie können Übersichtspläne exportiert werden?

Bei dem Test wurde besonders Wert auf die Möglichkeit einer individuellen Darstellung und das Bilden von logischen Sichten gelegt.
Eine übersichtliche Darstellung ist nicht der einzige Zweck einer Management-Plattform. Die vorhandenen Ressourcen sollten auch kontrolliert werden. Deshalb war ein Teilbereich des Tests der Überwachung gewidmet.

5.1.2 Überwachung

Der zweite Teil des Tests befaßte sich mit der Überwachung von Komponenten. Als erstes wurde der Bereich MIB-Unterstützung untersucht. Dazu wurde eine vom/am Lehrstuhl entwickelte SNMP-MIB und eine Enterprise-MIB eines Switches in die Plattform geladen.
Im nächsten Schritt sollte die Management-Plattform einen Trap generieren, wenn ein bestimmter Schwellwert einer MIB-Variable erreicht ist.
Außerdem sollten die Möglichkeiten untersucht werden, aus der Menge der ankommenden Ereignisse, durch ein Regelwerk, nur relevante Ereignisse zu einem bestimmten Thema herauszufiltern. Bietet die Management-Plattform überhaupt Funktionen, damit der Administrator den Überblick über alle Ereignisse erhält?

5.1.3 Installation

In den Handbüchern wurde nach Möglichkeiten gesucht, eine verteilte Architektur mit der Management-Plattform aufzubauen. Gibt es Funktionen, damit die Plattform von einer anderen Plattform kontrolliert und konfiguriert werden kann?

5.1.4 Test-Kriterienkatalog

Für diesen Test wurde der gleiche Kriterienkatalog wie in Anhang A verwendet. Außerdem wurden beim Bewerten die in Kapitel 4 beschriebenen Wertebereiche gebraucht. Bei einigen Anforderungen wurde die gemessene Zeit auf die Bewertungsskala umgerechnet (siehe Beispiel Kapitel 4). Z.B. bei der Oberfläche konnten natürlich nur subjektive Eindrücke als Bewertung genutzt werden.

Die Gewichtung wurde durch das Setzen von eigenen Schwerpunkten und durch die Fragestellungen gewählt (siehe oben). Bei Kriterien, die nicht getestet werden konnten (z.B. Protokollanalysatoren), wurde die Gewichtung und die Bewertung auf eins gesetzt, um einen Einfuß auf das Gesamtergebnis zu vermeiden.

Um einen zu großen Umfang dieser Arbeit zu vermeiden, wurde der Kriterienkatalog hier nicht eingefügt. Im späteren Verlauf befinden sich die Kriterienkataloge zu den beiden Plattformen mit den jeweiligen Werten.

5.2 Anwendung auf IBM *NetView*

Das *NetView for AIX* Version 4 Release 1 Programm ist ein Management-Werkzeug für heterogene Umgebungen basierend auf TCP/IP.
Die Management-Plattform stellt Module für das Konfigurations-, Fehler-, Sicherheits- und Leistungsmanagement zur Verfügung. Es handelt sich um ein offene Plattform, die die Möglichkeit bietet, SNMP- und CMIP-Anwendungen zu integrieren. Außerdem bietet *NetView* Unterstützung für mehrere relationale Datenbanken.

Die IBM verwendet die bekannten Fachbegriffe, die einzige Besonderheit ist der Gebrauch von Symbol und Objekt. Ein Symbol stellt immer höchstens ein Objekt dar. Aber ein Objekt kann durch mehrere Symbole repräsentiert werden. Ein Objekt mit seinen Attributen (in *NetView* Kennzeichen genannt) repräsentiert eine Ressource. In der Datenbasis ist ein Objekt nur einmal vorhanden und wird durch einen Selektionsnamen eindeutig bestimmt. Die Symbole stellen das Objekt auf einem oder mehreren Übersichtsplänen dar. Ein Symbol hat mehrere Attribute. Mit einem Teil der Attribute werden die Eigenschaften des Symbols (z.B. Beschriftung des Symbols) ermöglicht, mit den restlichen wird der Zugriff auf die Kennzeichen der Objekte realisiert. Durch Ändern des Symbols, ändern sich auch die Attribute des Objekts. Durch Zuweisung eines Symbols zu einem Objekt der Datenbank generiert das System automatisch bestimmte Kennzeichen, die teilweise auch Werte bekommen. Bekommt ein Objekt z.B. das Symbol Router, so wird das Kennzeichen isRouter mit dem Wert TRUE erzeugt.

5.2.1 Darstellung

Am Anfang wurde versucht, die Menüstruktur ohne Dokumentation zu verstehen. Die ganzen Funktionen sind sehr logisch unter den einzelnen Menüpunkten zusammengefaßt.
Das Durchwandern der verschiedenen Übersichtspläne ist sehr einfach und intuitiv gemacht. Es können auch beliebig viele gleichzeitig dargestellt werden. Alle Übersichtspläne sind automatisch hierarchisch als Baum angeordnet, der durchwandert werden kann. Außerdem ist ein Fenster „Navigation Tree“ (siehe untere Abbildung) vorhanden, in diesem ist die Hierarchie des Baumes dargestellt. Durch einen Doppelklick auf ein Blatt des Baumes öffnet sich ein Fenster mit diesem Übersichtsplan. Es muß also nicht die gesamte Hierarchie des Baumes durchwandert werden. Für jeden Übersichtsplan, der geöffnet wurde, entsteht genau dann, und nur dann, ein Blatt im „Navigation Tree“-Fenster.

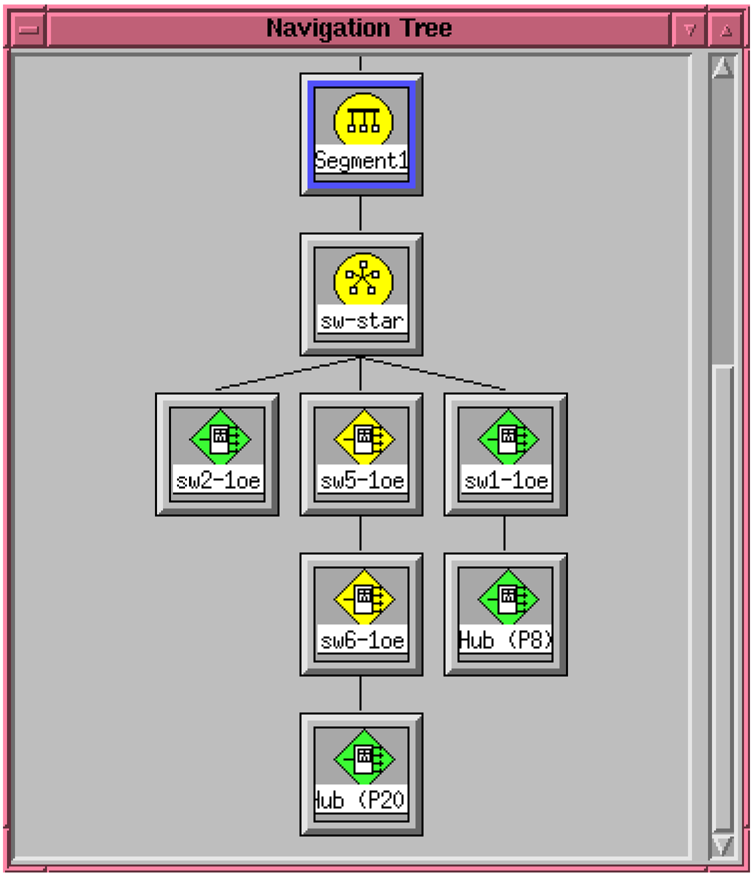

Abbildung 5.2: Navigation Tree

Diese Möglichkeit, mit dem „Navigation Tree"-Fenster in kürzester Zeit die Übersichtspläne zu wechseln, erleichtert die Arbeit sehr. Außerdem wird die Anzahl an geöffneten Fenstern verringert.
Nach dem Start des Programms lag bereits eine „Submap" mit dem entsprechenden Class-C-Netz des Lehrstuhls vor. In diesem Übersichtsplan befanden sich alle erkannten Geräte mit der entsprechenden IP-Adresse.

Leider waren jedoch nicht alle Geräte vorhanden. Alle Switches und Hubs wurden nicht automatisch erkannt, obwohl die Switches eine IP-Adresse haben.
Bei diesem Problem wurde das erste Mal nach Hilfe in der Online-Dokumentation gesucht. Leider ließ sich für dieses spezielle Problem keine schnelle Lösung finden.
Das Problem war, daß NetView neue Ressourcen erkennt, indem es die ARP[1]-Caches der bekannten Objekte nach neuen MAC-Adressen durchsucht. Wird eine solche Adresse gefunden, werden automatisch SNMP-Requests an dieses Gerät gesendet, um weitere Informationen zu erhalten.
Da ein Switch in der Regel (Ausnahme Konfiguration) nicht adressiert wird, erscheint seine MAC-Adresse in keinem ARP-Cache. Deshalb ist kein Switch automatisch erkannt worden. Wurden dem Switch ICMP-Pakete (durch *ping*) gesendet, so erschien das Gerät nach ca. 1-2 Sek. Dies war unabhängig davon, ob der *ping* von der Management-Plattform oder von einer

[1] Address Resolution Protocol

„Shell“ ausgeführt wurde. Ein *ping* auf ein ganzes Subnetz ist jedoch nicht möglich (z.B.in der Form: ping 129.187.214.0), nur ein *ping* auf eine einzige IP-Adresse.
Diese Methode ist bei großen Netzen hilfreich, in denen sehr viele Hubs, Switches und Brükken vorhanden sind, jedoch viel zu aufwendig.
Deshalb gibt es eine „seed“-Datei, in der angegeben werden kann, welche IP-Netze untersucht werden sollen. Außerdem ist es möglich, den Discover-Dämon so zu konfigurieren, daß dieser ICMP-Broadcasts (die Managementstation an unserem Lehrstuhl war aber nicht so konfiguriert) verwendet, um das Netz zu untersuchen. Leider können diese Änderungen nur mit root-Berechtigung vorgenommen werden. Die Änderungen in der seed-Datei können jederzeit vorgenommen werden. Bevor diese Änderungen aber wirksam werden, muß die Plattform die Datei neu auslesen.

Die automatisch generierten Ressourcen erhalten bereits die Informationen, wie IP-Adresse, Rechnername, Gerätehersteller, Betriebssystem, Versionen, Ansprechpartner, Ort. Natürlich sind die Informationen nur vorhanden, wenn sie vorher am Gerät konfiguriert wurden.
Leider ist oftmals das falsche Symbol zugeordnet (siehe Beispiel unter Kapitel 5.2). Die Symbole der Geräte können jedoch ziemlich einfach im nachhinein geändert werden. Aber auch hier ist wieder zu bemerken, daß der Aufwand bei vielen Geräten sehr aufwendig ist. Diese Änderungen ließen sich nicht im Batchbetrieb vornehmen.
Beim Versuch, die Symbole zu ändern, fiel auf, daß nicht alle Befehle in der Menüleiste vorhanden sind. Das Ändern eines Symbols ist nur durch ein Pop-Up-Menü (erhält man durch einen Klick mit der rechten Maustaste) möglich. Dies ist jedoch in keinem Handbuch erwähnt.

Wird ein Symbol manuell angelegt, weil die betreffende Ressource gerade nicht verfügbar ist, muß ein Selektionsname vergeben werden. Durch diesen Namen wird das Objekt in der Datenbank eindeutig bestimmt. Dieser Name muß auf allen Übersichtsplänen eindeutig sein.
Leider wird auf diesen Umstand nicht hingewiesen. Allerdings können mehrere Selektionsnamen für ein Objekt vergeben werden. Der Benutzer sollte sich also vor dem Anlegen von Symbolen ein eindeutiges Namenschema überlegen.
Beim manuellen Anlegen kann keine Information über Hersteller, Version des Betriebssystems etc. angegeben werden. Diese Felder sind bei der Eingabesmaske des Symbols gesperrt. Natürlich können solche Angaben bei Verfügbarkeit der Ressource, mit dem MIB-Browser gelesen und geändert werden. Aber der schreibende Zugriff auf diese Kennzeichen ist durch die Eingabemaske des Symbols nicht möglich. Ob diese Information automatisch vom System, bei Verfügbarkeit der Komponente, nachgetragen wird, konnte mangels Testmöglichkeiten nicht überprüft werden.
Beim Anlegen von Symbolen gibt es fünf Bereiche, in denen Angaben zu machen sind:

- „Capabilities“: z.B. isRouter, isIPXWorkstation
- „General Attributes“: z.B. isSNMPSupported, vendor, isSNMPProxied
- „IP Map“: z.B. Hostname, IP-Adresse
- „Selektionsname“: eindeutige Zeichenfolge
- „Labelname“: Zeichenfolge

Labelname, Selektionsname, General Attributes können jederzeit wieder geändert werden. Die Capabilities können nicht mehr geändert, nur durch Zuweisen eines neuen Ikons können teilweise Änderungen vorgenommen werden. Die Angaben unter „IP Map“ sind nicht mehr änderbar. Sollen die Angaben trotzdem geändert werden, muß das Objekt aus der Datenbank gelöscht und wieder neu angelegt werden. Die Objekte und die Symbole werden in der Daten-

bank getrennt verwaltet. Symbole können nach Belieben gelöscht werden, ohne daß das Objekt beeinflußt wird. Selbst wenn das letzte Symbol, welches ein Objekt repräsentiert, gelöscht wird, bleibt das Objekt vorhanden.
Die Tatsache, daß Objekte gelöscht werden müssen, um ihnen andere IP-Adressen geben zu können, bedeutet einen erheblichen Aufwand. Warum dies im NetView so gelöst ist, wo doch jedes Objekt einen Selektionsnamen als eindeutiges Kennzeichen besitzt, ist unklar.

Ein weiterer Umstand, der jeden *NetView*-Anfänger viel Zeit kostet, ist das Verschieben von Symbolen. Auf jedem Übersichtsplan ist es möglich, das Auto-Layout einzuschalten, dann ist ein Verschieben per Maus nicht mehr möglich. Ist das Auto-Layout ausgeschaltet, so muß immer die CTRL-Taste gehalten sein, damit das Verschieben möglich ist. In der Online-Hilfe wurde 15 Min. gesucht, bis der betreffende Eintrag gefunden wurde. Beim freien Verschieben fehlt dann auch jede Unterstützung, wie z.B. an einem definierten Raster ausrichten zu können.
Das Verschieben und Kopieren von Symbolen in andere Übersichtspläne ist einfach. Das Kopieren von Symbolen ist aber nur sinnvoll, wenn ein Symbol in mehreren Übersichtsplänen dargestellt werden soll. Wird ein Symbol in einen anderen Übersichtsplan kopiert, können Änderungen nur mit den obigen Einschränkungen (siehe oben, Ändern von Attributen der Symbole) vorgenommen werden. Es ist also nicht möglich, Symbole zu kopieren und dann zu editieren, um sich Arbeit beim Anlegen von gleichartigen Objekten (z.B. fünf neue, gleiche Switches sollen ins Netz integriert werden) zu sparen.

Durch das Kopieren kann ein IP-Subnetz in logische Teilnetze gegliedert werden. Im Lehrstuhl-Netz war die zentrale Komponente ein Switch (siehe Abbildung 5.1), an den weitere Switches angeschlossen sind. Für jeden Switch kann ein eigener Übersichtsplan angelegt werden, in den sich alle Geräte befinden, die an dem Switch angeschlossen sind. Beim Anlegen von solchen Übersichtsplänen ist man aber gewissen Einschränkungen unterlegen. Hinter jedem Symbol kann sich nur eine weitere „Submap“ verbergen. Werden mehr „Submaps“ benötigt, müssen zusätzliche Symbole angelegt werden. Das nächste Beispiel soll dies kurz verdeutlichen:
Der zentrale Switch im Lehrstuhl-Netz hat nur eine Einschubkarte, auf der alle Ports für die Verbindung zu anderen Maschinen vorhanden sind. Deshalb ist diese Ressource mit zwei Symbolen dargestellt (ein Symbol für den Switch, eins für die Einschubkarte). Nun möchte man, das sich beim Doppelklick auf einen bestimmten Port ein neuer Übersichtsplan mit dem entsprechenden Gerät öffnet. Dies ist aber bei der momentanen Situation nicht möglich, denn beim Doppelklick auf das Symbol der Einschubkarte, öffnet sich nur **ein** einziger Übersichtsplan. In diesem müßten sich alle Maschinen befinden. Deshalb müßte der Administrator für jeden Port ein eigenes Symbol einführen, worauf sich dann beim Doppelklick auf das Symbol der jeweilige Übersichtsplan öffnet.

Die Darstellung von Symbolen ist bei *NetView* so ausgelegt, daß die reale physikalische Verkabelung dargestellt werden kann. Man kann auch eine logische Darstellung aufbauen, wobei dies nicht einfach ist. Der Benutzer verfällt oft automatisch wieder in die Darstellung der physikalischen Realität.
In *NetView* gibt es vier verschiedene Ebenen:

- Internet
- Netzwerk
- Segment
- Knoten

Der Benutzer hat in der Knoten-Ebene aber nicht die Möglichkeit, IP-Symbole anzulegen. Durch diesen Umstand lassen sich sehr schlecht Hierarchien bilden. Dies soll an folgendem Beispiel kurz gezeigt werden:
In der Netzwerk-Ebene sollen z.B. verschiedene Netze dargestellt werden, die mit Routern über WAN-Strecken verbunden sind. Auf Segment-Ebene müßten dann alle Ressourcen in einer flachen Hierarchie dargestellt werden, obwohl mit kaskadierten Switches oder Hubs durchaus mehrere Ebenen denkbar sind. Die Knoten-Ebene muß den Einschubkarten (Ethernet-Karte) der Geräte vorbehalten bleiben. Damit trotzdem eine Hierarchie gebildet werden kann, müssen zusätzliche Symbole (z.B. Symbol für Raum) eingefügt werden. Somit faßt man Ressourcen wieder unter einem physikalischen „Tatbestand“ (in diesem Beispiel der Raum) zusammen. Diese Problematik wird noch dadurch verstärkt, daß es keine Symbole gibt, die logische Sachverhalte (z.B. Symbol für Arbeitsgruppe) repräsentieren. Bei stark kaskadierten Netzen, was heute bei sogenannten Backplane-Switches durchaus üblich ist, wird die Darstellung deshalb schnell unübersichtlich.

NetView legt beim Start ein Symbol „Segment“ an, durch dieses Symbol kann ein Übersichtsplan geöffnet werde. Dieser stellt das gesamte Netz dar, in dem sich die Plattform befindet. Alle Ressourcen, die auch in diesem Netz sind und erkannt werden, werden als Symbol in diesem Übersichtsplan angezeigt. Jeder Übersichtsplan hat einen bestimmten Verkabelungstap, wie z.B. Ring, Bus, Stern. Beim Test wurde dieser Übersichtsplan mit dem Verkabelungstyp Bus angelegt. Tatsächlich handelt es sich aber um einen Stern. Das nachträgliche Ändern des Verkabelungstyps ist aber nicht möglich.
Alle Symbole müßten auf einen anderen, leeren Übersichtsplan kopiert werden, dann könnte der Übersichtsplan „Segment“ gelöscht werden. Dann kann man den Übersichtsplan „Segment“ neu anlegen und alle Symbole wieder zurückkopieren. Dies dauert für einen geübten Benutzer bei 100 Objekten ca. 1 Std.
Will man ein Symbol auf allen Übersichtsplänen suchen, gibt es eine „Locate“-Funktion. Diese Funktion ist sehr gut realisiert. Man kann nach vielen Arten von Attributen suchen, wie z.B. IP-Adresse, Rechnername, Symbolname, Selektionsname. Bei der Suche kann zwischen einer exakten Übereinstimmung und einer Teil-Zeichenfolge unterschieden werden. Wurden ein oder mehrere Symbole gefunden, so wird dies in einem Auswahlfenster mit Angabe des jeweiligen Übersichtsplans dargestellt. Bei einem Doppelklick auf ein gefundenes Symbol öffnet sich der Übersichtsplan mit dem dann markiertem Symbol in einem neuen Fenster.
Leider gibt es auch Funktionen, die nicht so gut durchdacht sind. Es können mehrere Symbole markiert und ausgeschnitten, aber es kann nur jedes Symbol einzeln gelöscht werden. Wurde ein Symbol aus Versehen gelöscht, so ist es für immer verschwunden. Ein Rückgängigmachen (wie es bei *Windows*-Programmen möglich ist) ist bei *NetView* nicht vorhanden.

Ein Kopieren von Symbolen zwischen verschiedenen Benutzer-Übersichtsplänen ist nicht möglich.
Der aktuelle Stand eines Übersichtsplans kann aber unter einem anderen Namen gespeichert werden. Mit diesem neuen Übersichtsplan kann dann gearbeitet werden. Nimmt man an diesem neuen Übersichtsplan Änderungen vor, so wirken sich diese nicht auf das Original aus. Zu einem späteren Zeitpunkt kann das Original wieder weiterverwendet werden. Die verschiedenen Übersichtspläne sind völlig autark, die Objekte und Symbole, die zu einem bestimmten Übersichtsplan gehören, sind unabhängig von anderen Übersichtsplänen. Ein Kopieren von Symbolen zwischen verschiedenen Übersichtsplänen ist nicht möglich.

Ein weiteres Hilfsmittel ist das Anlegen von Snapshots ([GMD800], S. 15). Ein Snapshot ist eine Kopie eines Benutzer-Übersichtsplans zu einem bestimmten Zeitpunkt. Der Snapshot dokumentiert einen Zustand und kann somit zum Lösen von Problemen beitragen. In einem Snapshot kann der Zustand eines Netzes vor Eintreten eines Problems gezeigt werden.
Der Unterschied zwischen einem Snapshot und den Übersichtsplänen ist jedoch viel zu undeutlich. Lediglich der kleine Vermerk „Snapshot“ am unteren Fensterrand zeigt, daß es sich nicht um einen Übersichtsplan handelt.
Die Einarbeitungszeit für das Konzept der Übersichtspläne von *NetView* dauerte ca. 15 Min., hierzu mußten folgende Bereiche verstanden werden:

- die vier verschiedenen Ebenen (siehe oben) von Übersichtsplänen
- Konfiguration und Attribute der Symbole von Übersichtsplänen
- Beschränkung der Symbole auf bestimmte Übersichtspläne (auf der Knoten-Ebene sind keine IP-Symbole möglich)

Eine eigene „Submap“ erstellen mit Kopieren von 15 Objekten dauerte ca. 20 Min., dazu sind folgende Schritte nötig:

- Attribute für die Symbole bestimmen (z.B. ein bestimmter Ansprechpartner, die in die neue „Submap“ eingefügt werden sollen)
- geeignetes Symbol für die „Submap“ wählen
- Übersichtsplan, von dem aus die „Submap“ geöffnet werden kann, wählen
- „Submap“ anlegen und konfigurieren
- Symbole mit den obigen Attributen suchen und markieren
- Symbole aus den ursprünglichen Übersichtsplänen ausschneiden und in die neue „Submap“ einfügen
- Verbindungslinien zwischen den Symbolen einfügen

Ein weitere Möglichkeit, eigene Sichten zu definieren, bietet der Übersichtsplan „Collections“. Die Möglichkeit, neue Sichten anzulegen, ist sehr komfortabel. Es gibt ein Dialog-Fenster, in dem SQL-artige Anfragen mit und/oder/nicht verknüpft werden können. Hier können mit Maus-Klicks einfache Anfragen generiert werden. Die erstellten Sichten werden dann von *NetView* aktualisiert, wenn neue Ressourcen hinzukommen. Die Einarbeitungszeit für dieses Werkzeug dauerte ca. 15 Min. Außerdem ist ein einfacher Text-Editor vorhanden, mit dem ein geübter Benutzer direkt Befehle eingeben kann. Die Regeln lassen sich aus den Eigenschaften aller Objekte erstellen. Eine möglche Regel ist:
isRouter = TRUE && IP > 129.117.214.0
D.h. alle Geräte, die Router sind, und eine IP-Adresse aus dem Class-C-Netz 129.187.214.0 haben, sollen in dieser Sicht vorhanden sein.
Leider gibt es auch Einschränkungen, so sind z.B. keine Schnittmengen möglich:
IP >129.187.214.0 && IP < 129.187.214.80
Als Ergebnis sollte die Anfrage eine Menge aller Objekte liefern, deren IP-Adresse größer als 129.187.214.0 und kleiner als 129.187.214.80 ist.
Dies schränkt die Funktionalität erheblich ein (siehe dazu Kapitel 5.2.3).
Wurde eine Sicht durch eine Anfrage gebildet, so steht diese Sicht immer unter einem Symbol zur Verfügung. Bei einem Doppelklick auf dieses Symbol öffnet sich ein Fenster mit diesen Symbolen, die Objekte repräsentieren, die der Anfrage entsprechen. Sind neue Objekte seit dem letzten Aufruf hinzugekommen, so werden auch dafür Symbole angezeigt.

Da der Benutzer durch Programmierung eigene Kennzeichen für Objekte erzeugen kann, sind sehr individuelle Sichten möglich. Beim Anlegen von zusätzlichen Kennzeichen sind aber schon sehr tiefe Kenntnisse der Datenbank und der Programmierung erforderlich.
Trotzdem ist die Möglichkeit der Bildung von Sichten ein sehr nützliches Hilfsmittel, was durch seine einfache Benutzerschnittstelle auch für ungeübte Benutzer geeignet ist.

Sind die Übersichtspläne erstellt worden, möchte man diese vielleicht ausdrucken. Export-Funktionen für die einzelnen Übersichtspläne sind aber praktisch nicht vorhanden. Das Layout eines Übersichtsplans wird in einer einfachen Datei und in der Datenbank gespeichert. Eine Export-Funktion für andere Anwendungen gibt es nicht, nur ein Ausdruck ist möglich.
Dazu muß das Fenster, in dem sich der jeweilige Übersichtsplan befindet, selektiert werden. Der Ausdruck ist dann das komplette Fenster mit Menüleiste. Außerdem kann der Ausdruck Postscript-Format abgespeichert und mit Bildbearbeitungsprogrammen (z.B. unter UNIX das Programm *xv*) editiert werden. In solchen Programmen können wenigstens die Menüleisten oder Ränder entfernt werden. Trotzdem ist das Ausdrucken für Dokumentationszwecke völlig unzureichend.

5.2.2 Überwachung

Das Laden von zusätzlichen MIBs funktioniert sehr einfach über einen Menüpunkt. Allerdings wird deutlich, daß der Benutzer Fachkenntnisse benötigt, um mit eigenen MIBs zu arbeiten. Wenn eine MIB geladen wird, so ist dies ein zusätzlicher Teilbaum in der MIB-Baum-Struktur. Deshalb müssen neue MIBs, die von anderen MIBs abhängig sind, in der richtigen Reihenfolge geladen werden. Ist z.B. die IBM-ALERT.MIB von der IBM.MIB abhängig, so muß erst die IBM.MIB und dann die IBM-ALERT.MIB geladen werden.
Das System gibt aber keinerlei Informationen, ob Abhängigkeiten zwischen MIBs bestehen. So muß der Benutzer im Quellcode der MIB nachlesen, ob Abhängigkeiten existieren.
NetView bietet die Möglichkeit, SNMP- und SNMPv2-MIBs zu laden, dies geschieht über zwei verschiedene Menü-Unterpunkte. Wird z.B. eine SNMP-MIB über den falschen Menüpunkt geladen, so führt die Management-Plattform das Laden ohne Fehlermeldung durch. Erst wenn der Benutzer im MIB-Browser versucht, den entsprechenden Teilbaum zu durchwandern, ist dies nicht möglich.
Der Benutzer muß also ebenfalls den Quellcode der MIB durchlesen um zu erfahren, von welcher Version die MIB ist.

Der MIB-Browser ist ein einfach zu bedienendes Werkzeug. Das Einarbeiten ist ohne Hilfe möglich. Eine Verbesserung wäre es, wenn man die Beschreibung einer MIB-Variable nicht explizit anstoßen müßte. Diese Information sollte automatisch zur Verfügung stehen, sobald eine Variable ausgewählt wird. Denn die Information zu den MIB-Variablen wird mit der MIB geliefert und ist deshalb lokal auf dem Rechner gespeichert. Es wird also kein zusätzliches Datenaufkommen erzeugt, um diese Information zu erhalten.

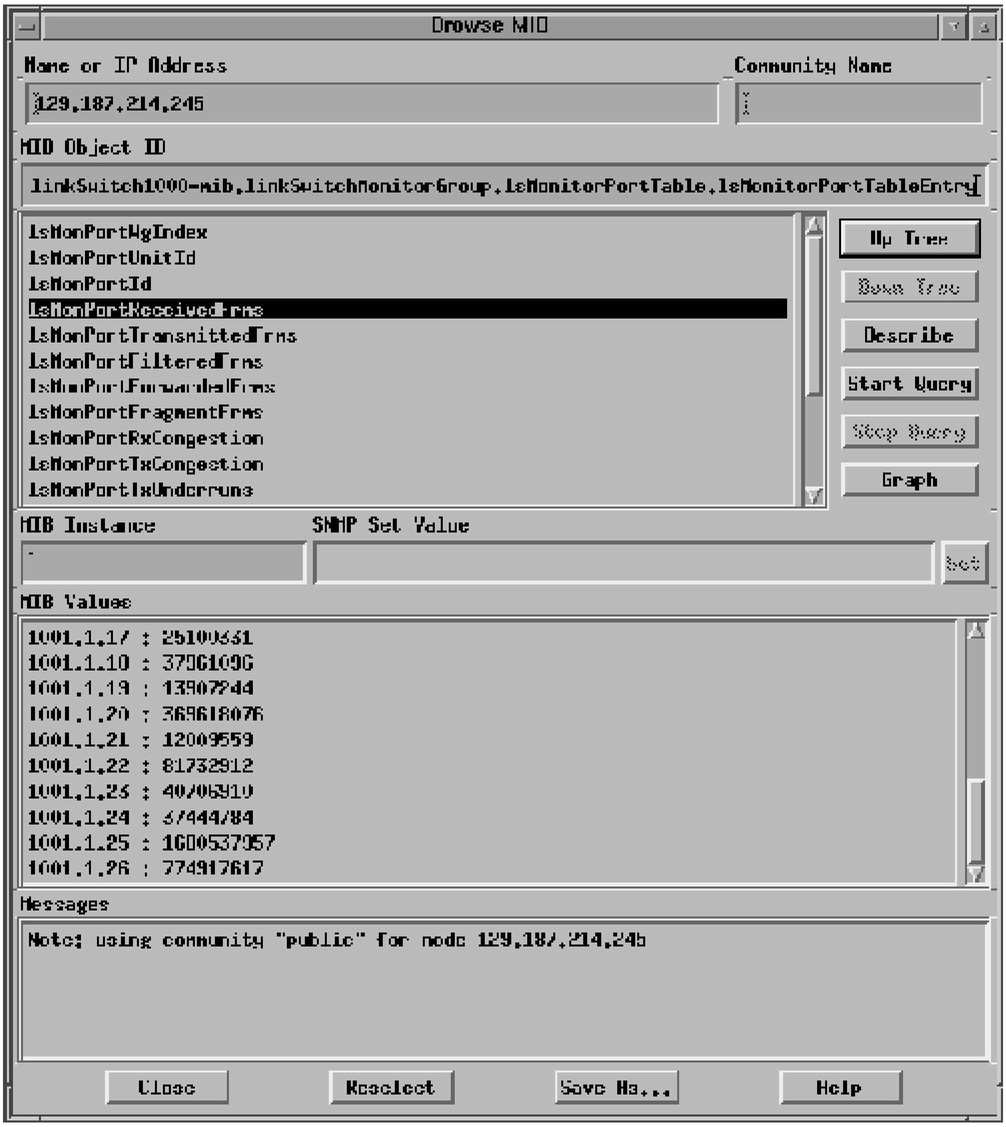

Abbildung 5.3: MIB-Browser

Das nächste getestete Werkzeug, war das „Data Collection & Thresholds“-Werkzeug. Dieses Werkzeug kann auch nur von Fachpersonal effizient benutzt werden. Die Bedienung des Werkzeugs ist einfach, dazu war keine Einarbeitungszeit nötig. Sollen beim Überschreiten des Schwellwerts einer bestimmten Variable Traps generiert werden, so sind gute Kenntnisse im Bereich Internet-Management nötig.
Es können neue Regeln hinzugefügt werden, worauf man im MIB-Browser die zu überwachende MIB-Variable auswählen kann. Als nächstes kann die Menge der Objekte ausgewählt werden, bei denen diese MIB-Variable überprüft werden soll.
Zuletzt muß dann noch der eigentliche Schwellwert, Rearm-Wert, Polling-Intervall etc. gesetzt werden.
Ein großer Nachteil ist, daß dieses Werkzeug anscheinend nicht mit allen MIBs arbeitet. Der Test wurde mit der am Lehrstuhl entwickelten MIB (im weiteren Lehrstuhl-MIB genannt) und einer 3Com-Enterprise-MIB der Switches durchgeführt. Bei der Überwachung konnten die Variablen der Lehrstuhl-MIB nicht gewählt werden. Die Variablen der 3Com-MIB ließen sich überwachen und Schwellwerte konnten gesetzt werden. Da die Lehrstuhl-MIB mit dem MIB-

Browser durchwandert werden konnte, ist wahrscheinlich die Anwendung Data-Collection nicht voll funktionsfähig. Die Anwendung erlaubt auch die Überwachung von MIB-Variablen, die mehrere Instanzen besitzen.

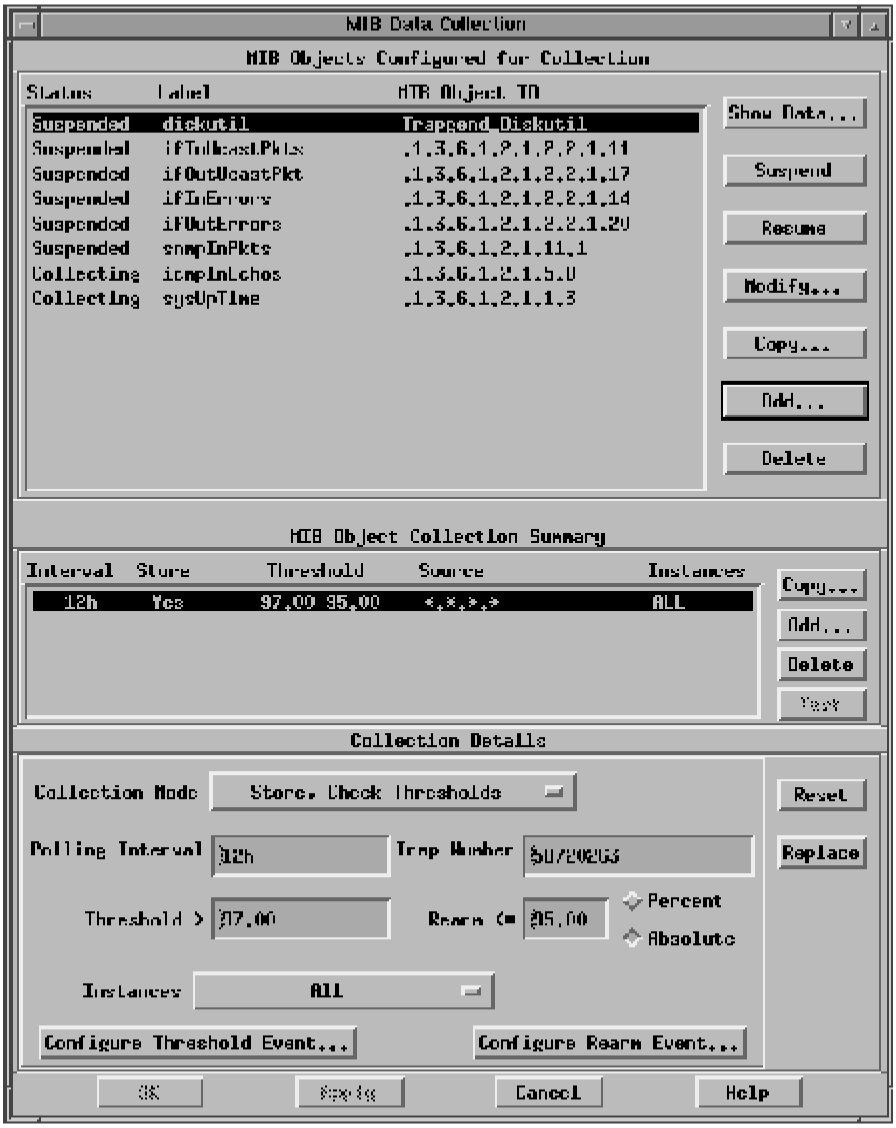

Abbildung 5.4: Data Collection & Thresholds

Als letztes wurde der „Ruleset“-Editor von *NetView* untersucht. Dieses Werkzeug dient dazu, aus der Menge der Ereignisse und Traps die gewünschte Information herauszufiltern. Auch dieses Werkzeug ist nur für den professionellen Benutzer gedacht. Allein die Hilfe dazu fällt sehr dürftig aus. Das Werkzeug unterteilt sich in zwei Bereiche. Es gibt ein Fenster mit den Nodes, dies ist eine feste Anzahl von Funktionen, die zur Verfügung stehen. Im anderen Fenster können die Nodes kopiert und durch Linien verknüpft werden, so daß dann Regeln entstehen.

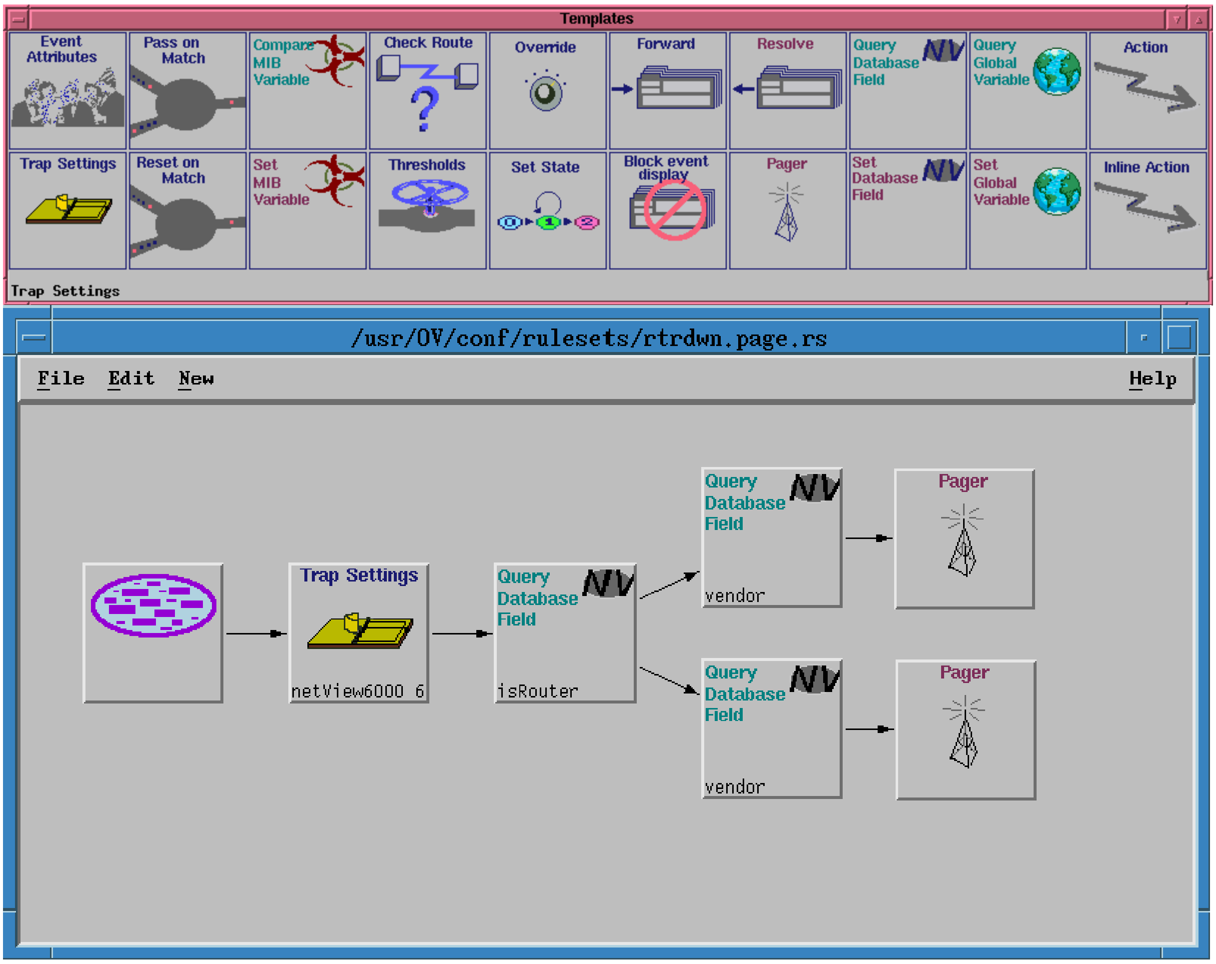

Abbildung 5.5: Ruleset Editor

Nach dem Kopieren sind noch Angaben zu jeder Funktion nötig. Bei dem „Aktion“-Node muß z.B. eine Batchdatei angegeben werden, die dann ausgeführt wird. Die Symbole der Funktionen können dann beliebig verknüpft werden, dadurch ergibt sich der Ablauf der Regel bzw. der Handlung.

Was die einzelnen Nodes für Funktionen bieten, ist in der Hilfe sehr unzureichend und schlecht erklärt. Beispiele sind zwar vorhanden, aber nur die grafische Repräsentation dieser Regeln. Es fehlen Erklärungen, wie die Angaben zu den Funktionen auszufüllen sind.

Eine Regel kann mit bereits abgespeicherten Regeln verknüpft werden. Dadurch lassen sich sehr komplexe Regelwerke erzeugen, wenn die Fenstergröße nicht überschritten wird. Füllt eine Regel die maximale Fenstergröße aus, so können keine neuen Funktionen hinzugefügt werden. Dieser Umstand ist jedoch vertretbar. Bevor die maximale Bildschirmgröße von einer Regel gebraucht wird, ist vorher wahrscheinlich die Regel vom logischen Ablauf nicht mehr zu verstehen. Die Regeln werden als einfache ASCII-Dateien (siehe Abbildung unten) abgespeichert, somit kann der geübte Benutzer direkt diese Dateien editieren.

```
RuleSet2 RuleSet TrapID3 TrapID6
"" 1
TrapID3 TrapID AttrJoin4
netView6000 1.3.6.1.4.1.2.6.3 6 58916867 0 "netView6000    1.3.6.1.4.1.2.6.3" "IBM_NVIDWN_EV    Specific 58916867    " ""
AttrJoin4 AttrJoin ResolvedCorr5
2 0 2 600 0 0 ""
ResolvedCorr5 ResolvedCorr
""
TrapID6 TrapID AttrJoin4.2
netView6000 1.3.6.1.4.1.2.6.3 6 58916866 0 "netView6000    1.3.6.1.4.1.2.6.3" "IBM_NVIUP_EV    Specific 58916866    " ""
```

Abbildung 5.6: Regel als ASCII-Datei

Ist eine Regel geschrieben worden, kann sie in einem neuen „Workspace“ angewandt werden. Dies macht die Arbeit mit Ereignissen in *NetView* sehr komfortabel. Standardmäßig steht dem Benutzer ein Workspace „Event“ zur Verfügung. In diesem werden alle Ereignisse gesammelt und dargestellt. Die Ereignisse können als Listenform und in Karteikartenform durchgelesen werden.
Die Listenform hat den Vorteil, daß auf einen Blick sehr viele Ereignisse zu sehen sind mit Kurzbeschreibung und Herkunft. Wird also ein Ereignis von einem bestimmten Rechner gesucht, kann es so schneller gefunden werden.
Bei der anderen Form ist das Suchen schwierig, dafür sieht der Benutzer auf einem Blick die gesamte Information des Ereignisses. Natürlich kann innerhalb eines Workspace auch nach Ereignissen durch Zeichenfolgen gesucht werden.
Die Ereignisse können durch den Benutzer mit zusätzlichen Informationen ergänzt und die entsprechende MIB-Variable (der aktuelle Stand) des Ereignisses kann durch den MIB-Browser direkt angesehen werden.

Beim Generieren eines neuen Workspace können eine oder mehrere Regeln, die vorher mit dem Ruleset-Editor erzeugt wurden, als Filter definiert werden. Auf alle Ereignisse werden dann vorher diese Regeln angewandt, bevor sie im Workspace angezeigt werden. Der Workspace kann einen aussagekräftigen Namen bekommen und es kann mit mehreren Workspaces gleichzeitig gearbeitet werden.
Folgendes Szenario ist damit denkbar:
Der Benutzer erzeugt mehrere Regeln, eine Regel sammelt alle Ereignisse der Router, die nächste Regel alle Ereignisse der Server und die dritte Regel alle Meldungen der Arbeitsplatzrechner.
Nun werden drei verschiedene Workspaces erzeugt, wobei jeder einer der obigen Regeln unterworfen wird.
Der Benutzer hat in jedem Workspace detaillierte Informationen über eine bestimmte Gerätegruppe, ohne lange in der Flut der Ereignisse nach den wichtigen zu suchen. Weiterhin werden alle Ereignisse noch im Workspace „Event“ festgehalten.
Wichtig dabei ist, daß die Objekte (und damit deren Ereignisse) nach ihren Eigenschaften unterschieden werden, die ihnen vorher durch Symbolauswahl oder Konfiguration zugewiesen wurden. Hat eine Ressource fälschlicherweise ein Router-Symbol, so werden diese Ereignisse der Ressource von der entsprechenden Regel berücksichtigt.

Einen zusätzlichen Vorteil erhalten Benutzer, die *NetView* auf einem IBM *AIX* System (es gibt auch *NetView for OS/2* mit Unterstützung von SNMP-Geräten) installiert oder so einen Rechner im Netz haben. Das Betriebssystem *AIX* von IBM stellt das Werkzeug *SMIT* (System Ma-

nagement Interface Tool) zur Verfügung, welches eigentlich ein Administrator-Werkzeug für das Betriebssystem ist.
Allerdings kann dieses Werkzeug beliebige Ereignisse künstlich erzeugen. Nur für Traps ist root-Berechtigung nötig. Damit lassen sich also Ereignisse an die Plattform (die Plattform kann auch auf einem anderen Rechner installiert sein) senden, und somit können die erstellten Regeln auf ihre Funktionsfähigkeit überprüft werden. SMIT bietet dazu eine einfache Eingabemaske, in der bestimmte Angaben (wie z.B. IP-Adresse der Plattform) gemacht werden müssen. Hier kann dann auch genau mittels einer Auswahlliste das gewünschte Ereignis erzeugt werden.
Da das Erstellen von Regeln sehr schwierig ist, stellt diese Möglichkeit ein sehr gutes Test-Werkzeug für komplexe Regeln dar.

Die gebundene und die elektronische Dokumentation sind beide sehr gut geschrieben. Es gibt einige Stellen, an denen die Dokumentation besser sein könnte. Insgesamt ist sie jedoch verständlich geschrieben, gut mit Bildern und Beispielen ergänzt.
Für unerfahrene Benutzer wird selbst der Umgang mit der Maus und *X-Windows* erläutert. Mit der Suchfunktion über alle Online-Handbücher läßt sich sehr schnell die gewünschte Information finden. Die Hilfe besteht aus Hypertexten, so daß per Doppelklick in ein anderes Dokument gewechselt werden kann. Lobenswert ist, daß die Hilfe gestartet werden kann, ohne die Management-Plattform starten zu müssen.
Der Eindruck wird leider etwas getrübt, da kein Ausdrucken der Online-Dokumentation möglich war. Dies könnte allerdings an einer falschen Konfiguration des Instituts-Rechners oder der Software liegen.
Ein Copy&Paste in einen Texteditor ist aber auch nicht möglich. Wird dies versucht, gehen alle Zeilenumbrüche verloren und der gesamte Text wird verschoben.
Im Test wurde versucht, per Copy&Paste die Funktionen des Ruleset-Editors auszudrucken, dafür war ca. 1 Std. nötig.

5.2.3 Installation

Bei dem Test wurde nach Möglichkeiten gesucht, die Management-Leistung verteilt von mehreren System erbringen zu lassen.
Der wichtigste Konfigurationsparameter bei *NetView* ist die seed-Datei. In dieser Datei kann genau angegeben werden, für welche IP-Bereiche eine NetView Plattform zuständig ist, dabei kann es sich um Netze oder einzelne Rechner handeln.
Diese Datei kann über eine Telnet-Verbindung mit der root-Berechtigung geändert werden, mit der Funktion „Read seed file“ kann die seed-Datei neu ausgelesen werden. Dieser Menüpunkt muß vorher aber konfiguriert werden, bevor er zur Verfügung steht. Ein Administrator muß dazu bei der Menükonfiguration diese Funktion einbinden und den Aufruf der Funktion für andere Benutzer außer root freigeben.

Das zentrale Programm bei einer verteilten Installation sind Netmon und folgende Konfigurationsdateien:

/etc/netnmrc	Die Start-Datei des Netmon
/usr/OV/log/netmon.trace	Die default Trace-Datei
/usr/OV/databases/openview/topo/netmon.lock	Die Lock-Datei des Netmon

Netmon kommuniziert mit SNMP-Agenten, um die Netz-Konfiguration zu ermitteln. Wird eine neue Komponente entdeckt, sendet Netmon einen SNMP-Trap an den „trapd“.

Sind alle Ressourcen im Netz erkannt, sendet der Netmon weiterhin ICMP-Pakete an diese aus, um Konfigurationsänderungen oder Statusänderungen zu bemerken.

Als Beispiel nehmen wir eine Konfiguration mit einer NetView-Plattform als zentrale Managementstation an, mit zwei Subnetzen, in denen sich jeweils eine NetView-Plattform als Submanager befindet.

NetView bietet die Möglichkeit, das Fehler- und das Konfigurationsmanagement von Teilnetzen auf Submanager zu verteilen. Der Submanager kann aber eigenständig keine Aufgaben übernehmen. Die Änderungen im Netz werden an den Zentralmanager gemeldet.

An den Submanager werden keine besonderen Forderungen bezüglich der Installation gestellt. Eine Standardinstallation reicht vollkommen aus. Es bleibt aber noch die Frage: Wie kann die Verantwortung für die Ressourcen auf die Manager verteilt werden?

Die Aufteilung wird mit dem Netmon der zentralen Managementstation konfiguriert. Der Netmon bietet die Möglichkeit, alle möglichen Submanager zu nutzen oder auch nur einen Teil der möglichen Submanager. Dazu muß in der seed-Datei eingetragen werden, daß unter einer bestimmte IP-Adresse ein Submanager erreichbar ist. In der seed-Datei des Submanagers müssen dann die Ressourcen angegeben werden, die dieser Submanager kontrollieren soll. Außerdem besteht die Möglichkeit, daß der Netmon der zentralen Managementstation die Ressourcen automatisch aufteilt (dabei werden alle Ressourcen durch die Anzahl der Manager geteilt und jeder bekommt die gleiche Anzahl zur Überwachung). Verbindungen mit geringem Durchsatz (z.B. Telefonleitungen) werden vom Netmon bei der automatischen Aufteilung nicht berücksichtigt.

Ist der Submanager in Betrieb, prüft er für sein Netz regelmäßig den Status der Ressourcen mittels SNMP-Gets ab. Die Geräte, die nicht in einem kontrollierten Subnetz liegen, werden direkt vom Zentralmanager überwacht. Dieser überwacht auch die Submanager mit dem *ping*-Befehl. Die Polling-Intervalle des Submanagers und des Zentralmanagers können frei konfiguriert werden.

Der Submanager speichert Änderungen in seinem Teilnetz lokal ab und meldet diese Information erst beim nächsten Polling an den Zentralmanager. Stellt der Zentralmanager fest, daß ein Submanager ausgefallen ist, so informiert er die anderen Submanager in diesem Teilnetz. Diese übernehmen dann die Arbeit für den ausgefallenen Submanager. Sind keine anderen Submanager vorhanden, so übernimmt der Zentralmanager die Überwachung.
Steht der ausgefallene Submanager zu einem späteren Zeitpunkt wieder zur Verfügung, wird die Überwachung wieder auf die ursprüngliche Weise vorgenommen.

Die Submanager melden lediglich eine Änderung in der Konfiguration ihres Teilnetzes. Der Zentralmanager ist dann selbst dafür zuständig, daß er die Information beim Submanager abholt. Der Zentralmanager hört in der Grundkonfiguration noch alle Statusmeldungen aller Komponenten mit. Hier ist viel Konfigurationsarbeit nötig, damit der Submanager nur die Probleme sofort meldet und die Konfigurationsänderungen nur einmal am Tag (evt. sogar in der Nacht).

Deshalb stehen im *NetView* noch folgende Funktionen zur Verfügung:

nvCollectionAdd
Fügt eine neue Collection von Objekten in den Collection-Übersichtsplan ein. Dazu müssen ein Name und eine Regel übergeben werden, die die Objekte eindeutig beschreiben.

nvCollectionAddCallback
Mit dieser Funktion können die bestehenden Collections geändert oder gelöscht werden.

nvCollectionIntersect
Diese Funktion überprüft, ob eine Schnittmenge von Objekten zwischen zwei Collections besteht. Als Ergebnis erhält man die Elemente dieser Schnittmenge. Somit läßt sich obiges Problem mit Collections (siehe Beschreibung Collections unter Kapitel 5.2.1) umgehen.

nvCollectionRead
Als Parameter ist ein Socket nötig, über den dann Ereignisse empfangen werden. Es werden nur Ereignisse gesendet, wenn welche vorhanden sind und nur bezüglich einer bestimmten Collection.

nvCollectionUnion
Als Ergebnis dieser Funktion werden die Elemente einer Vereinigung zweier Collections geliefert.

nvFilterDefine
Eine neue Filter-Regel kann hiermit erzeugt oder geändert werden. Diese Funktion ist nötig, damit spezielle Ereignisse, die einer bestimmten Domäne entsprechen, weitergesendet werden können.

nvFilterDelete
Entfernt eine Filter-Regel.

NVisClient
Mit dieser Funktion kann geprüft werden, ob eine bestimmte Anwendung funktionsfähig ist. Damit kann die Verfügbarkeit eines NetView Servers überwacht werden.

Die obigen Funktionen sind nur ein kleiner Ausschnitt. Natürlich gibt es zu jedem Bereich noch viel mehr Funktionen, die aber nur Teilmengen der obigen Funktionen darstellen. Durch diese Funktionen kann ein Submanager sehr detailliert konfiguriert werden.

Für NetView gibt es noch eine Erweiterung, den System Information Agent (SIA), damit kann der Submanager Probleme selbst lösen und nur Status-Meldungen an den Zentralmanager weiterleiten. Ist menschliches Eingreifen nötig, wird der Administrator auch dann vom Submanager informiert. Außerdem steht eine MIB zur Verfügung, mit dieser können der Submanager und die darauf ablaufenden Management-Anwendungen kontrolliert werden.

Am Lehrstuhl war keine zweite NetView Management-Plattform und kein SIA vorhanden, deshalb konnten obige Ausführungen im Test nicht verifiziert werden.

5.3 Anwendung auf HP *OpenView*

Der HP *OpenView Network Node Manager* B.04.01 ist eine Management-Plattform mit Anwendungen für Fehler-, Konfigurations- und Leistungsmanagement in heterogenen TCP/IP-Netzen.

Die Firma IBM hat das vorher erhältliche HP *OpenView* lizenziert und an dem neuen Produkt *NetView* Eigenentwicklungen vorgenommen. Der gemeinsame Ursprung dieser zwei Produkte ist noch immer deutlich zu erkennen. Bei der gegebenen Testreihenfolge war ein Einarbeiten in *OpenView* nicht nötig. Der Test von *OpenView* ist deshalb von den vorherigen Eindrücken durch *NetView* beeinflußt.

Für die Besonderheit bei den Begriffen Symbol und Objekt gilt hier das gleiche wie bei *NetView* (siehe Kapitel 5.2).
Einige Probleme, die im Test mit *OpenView* auftraten, sind völlig identisch mit denen unter *NetView*. Deshalb werden diese Probleme nicht näher erklärt, der interessierte Leser sollte bei Bedarf im Kapitel 5.2 nachlesen.

5.3.1 Darstellung

Die Orientierung unter *OpenView* war kein Problem, da die Menüstruktur die gleiche wie bei NetView ist. Es konnten nur unwichtige Unterschiede im Menüaufbau festgestellt werden.

Im *OpenView* muß für jeden Übersichtsplan ein eigenes Fenster geöffnet werden. Das Durchwandern der Übersichtspläne-Hierarchien ist nicht in einem Fenster wie bei *NetView* möglich. Eine Hilfe beim Durchwandern, wie der „Navigationtree“, ist in *OpenView* ebenfalls vorhanden. Der „Quick Navigator“ ist wie der Dateimanager unter *Windows* aufgebaut. Nach Bedarf können Ebenen angezeigt werden oder nicht. Die Bedienung ist aber etwas umständlich und das Werkzeug ist auch nicht so übersichtlich wie der „Navigationtree“.
Der Benutzer muß explizit angeben, daß ein Übersichtsplan im Quick Navigator aufgenommen wird. Aufgenommen wird dann das übergeordnete Symbol dieses Übersichtsplans. Klickt man dann auf dieses Symbol, wird automatisch der betreffende Übersichtsplan geöffnet. Der Benutzer kann sich im Quick Navigator genau die Symbole zusammenstellen, hinter denen sich die Netze befinden, die er überwachen soll.
Ansonsten ist der Aufbau der Übersichtspläne gleich mit dem Aufbau von NetView. Intelligent ist das Schließen von Fenstern gelöst, hierzu muß im aktiven Fenster lediglich die ESC-Taste gedrückt werden.

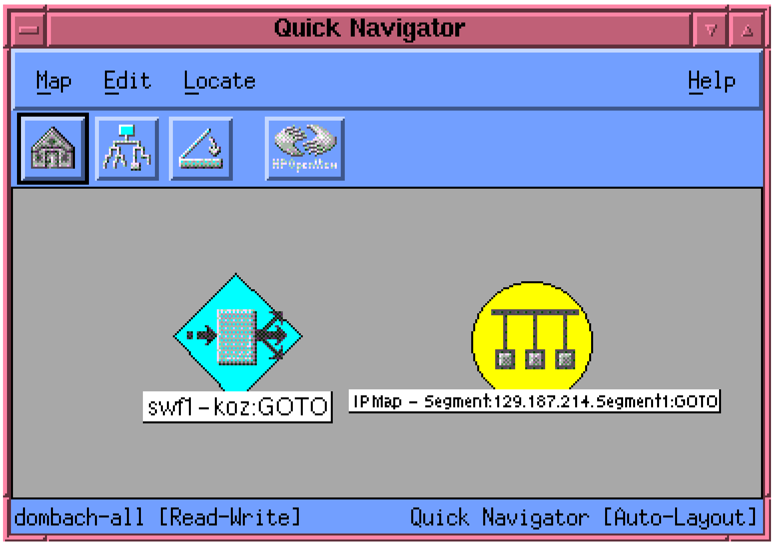

Abbildung 5.7: Quick Navigator

In *OpenView* trat das gleiche Problem wie bei *NetView* auf, nicht alle Ressourcen wurden automatisch erkannt und als Symbol dargestellt. HP verfolgt hier allerdings einen anderen Mechanismus als IBM. Die Plattform sendet SNMP-Anfragen aus, um neue Geräte in den Übersichtsplänen darstellen zu können. Dazu müssen SNMP-Agenten auf den Ressourcen konfiguriert sein. Zusätzlich wird der *ping*-Befehl verwendet.

Allerdings erschienen die fehlenden Geräte auch dann nicht in dem Übersichtsplan, wenn sie „angepingt" wurden. Letztendlich blieb bloß die Möglichkeit, die fehlenden Objekte manuell anzulegen.
Das Konzept einer seed-Datei ist unter *OpenView* vorhanden, allerdings muß die Plattform nach der Konfiguration neu gestartet werden.

Die erkannten Komponenten haben die gleichen zusätzlichen Informationen (z.B. Rechnername, Gerätehersteller, Betriebssystem, etc.) wie unter *NetView* und teilweise wurden ebenfalls falsche Symbole zugewiesen.
Insgesamt ist die Symbol- und Objekt-Verwaltung genauso realisiert wie bei *NetView*. Es werden die gleichen Eingabe-Masken verwendet und es stehen die gleichen Befehle zur Verfügung. Beim manuellen Anlegen von Symbolen stehen aber nicht so viele Symbol-Klassen wie bei *NetView* zur Verfügung.

Etwas eleganter ist das Arrangieren von Symbolen gelöst. Der Benutzer muß zum Verschieben eines Symbols keine Taste gedrückt halten. Auch wenn das Auto-Layout eingestellt ist, kann die Reihenfolge der Symbole (z.B. beim Bus) geändert werden. Sind sehr viele Symbole auf einem Übersichtsplan, gibt es ein Zoom-Fenster „Pan & Zoom", in dem stufenlos Ausschnitte vergrößert oder verkleinert werden können. Dieser Ausschnitt kann beliebig auf dem Übersichtsplan gewählt werden. Diese Technik garantiert, daß der Benutzer immer den Überblick über einen große Übersichtsplan behält.

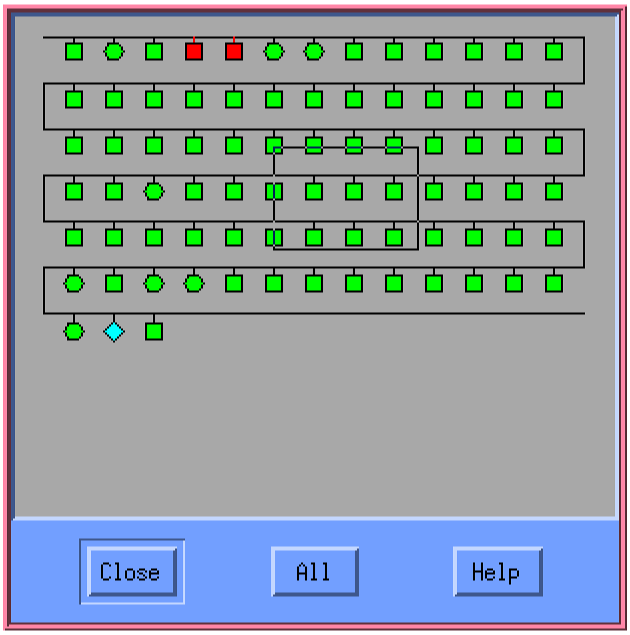

Abbildung 5.8: Pan & Zoom

Die Darstellung von IP-Teilnetzen und das Erstellen von Übersichtsplänen ist den gleichen Einschränkungen unterworfen, wie bei *NetView*. Die vier verschiedenen Ebenen von Übersichtsplänen sind ebenfalls vorhanden. Auch bei diesem Test wurde das Segment des Informatik-Lehrstuhls als Bus angezeigt. Ein nachträgliches Ändern war auch hier nicht möglich.

Die Benutzer-Übersichtspläne werden auf die gleiche Weise wie bei *NetView* verwaltet, Snapshots können auch angelegt werden. Ein Ausdrucken von Übersichtsplänen ist nicht möglich, sicherlich aber ein „Screenshot" mit einem handelsüblichen Grafikprogramm. Benutzerspezifische Sichten können nicht angelegt werden, denn ein ähnliches Werkzeug wie „Collections" unter *NetView* gibt es nicht. Der Benutzer kann nur versuchen, mit den Übersichtsplänen der vier Ebenen seine logische Sicht nachzubilden.

5.3.2 Überwachung

Weitere MIBs können über einen Menüpunkt geladen werden. Allerdings wird unter *OpenView* nicht zwischen SNMPv2-MIBs und SNMP-MIBs unterschieden. Im Test konnte die Lehrstuhl-MIB nicht geladen werden. Der Ladevorgang brach mit einer Fehlermeldung ab, daß in einer Zeile ein Fehler sei. Interessant dabei ist, daß die gleiche MIB in *NetView* ohne Probleme geladen werden konnte. Wurde aber versucht unter *NetView* die Lehrstuhl-MIB mit dem SNMP-MIB Menüpunkt zu laden, so wurde der Vorgang mit der gleichen Fehlermeldung wie bei *OpenView* abgebrochen. Die SNMP-MIB von 3Com konnte ohne Probleme geladen werden.

Das „Data Collection & Thresholds"-Werkzeug wurde von IBM nicht verändert. Bei OpenView handelt es sich um genau die gleiche Eingabe-Maske; alle Fähigkeiten sind hier ebenfalls vorhanden.

Ein Werkzeug wie der Ruleset-Editor ist nicht vorhanden. Mächtige Regeln zum Filtern von Ereignissen gibt es nicht. Beim „Event Browser" von *OpenView* ist von Vorteil, daß die Ereignisse nach Warnungsstufen vorsortiert sind. Außerdem können einfache Filter eingebaut werden, dabei kann nach folgendem gefiltert werden:

- Warnungsstufe
- Kenntnisnahme
- Ereignisse nach einem bestimmten Zeitpunkt
- Ereignisse vor einem bestimmten Zeitpunkt
- Quelle
- Nachrichten-Text
- Ereignistyp

Abbildung 5.9: Event Categories

Der Event Browser ist sehr einfach zu bedienen, dafür sind die Möglichkeiten der Filterung sehr beschränkt. Mehrere Filter hintereinander können nicht eingesetzt werden.
Unangenehm bei der Bedienung ist, daß immer nur eine Kurzbeschreibung des Ereignisses angezeigt wird. Für eine detaillierte Beschreibung muß ein Menüpunkt für jedes Ereignis angewählt werden. Ein Wechsel in eine andere Darstellungsform, wie es bei NetView möglich ist, in der dann jedes Ereignis ausführlich beschrieben ist, gibt es nicht. Ein vergleichbares Werkzeug zu SMIT gibt es auch nicht. Wurden Filter-Regeln geschaffen, können diese nicht überprüft werden, in dem künstlich ein Ereignis erzeugt wird.
Interessant ist auch die Möglichkeit der Kenntnisnahme von Ereignissen. Man kann ein Ereignis auf „Acknowleged“ setzen, als Folge werden die betreffenden Ressourcen nicht mehr farblich in den Übersichtsplänen markiert. Das Ereignis wird deshalb aber noch nicht gelöscht und bleibt noch im Event Browser.

Die Dokumentation ist nicht so gut wie die der IBM. Die gebundenen Handbücher besitzen zwar einen sehr guten Index, allerdings fehlt ein Glossar für die eingeführten Begriffe. Beispiele sind fast gar nicht vorhanden, allerdings werden die einzelnen Kapitel gut mit Screenshots dokumentiert. Eine Einteilung der Dokumentation in Benutzerrollen gibt es nicht.
Die Online-Dokumentation setzt den schlechten Eindruck fort. Eine Suchmöglichkeit über alle elektronischen Handbücher ist vorhanden. Der Suchindex ist jedoch nicht so gut wie bei *NetView*. Die Suchmaschine ist wesentlicher langsamer und das Springen zwischen Informationen ist nicht sehr komfortabel gelöst. Wird von einem Hilfetext zum nächsten gesprungen, ist kein Rücksprung mehr möglich. Das Fenster, in dem sich der Index der Handbücher befindet, ist nicht sehr groß und ein Vergrößern des Fensters ist nicht möglich.
Das Ausdrucken oder Copy&Paste zu einem Editor war auch hier nicht möglich.

5.3.3 Installation

Die Möglichkeit einer verteilten Installation ist bei HP nicht gegeben. In der Dokumentation war folgender Hinweis zu finden ([HPIG95], S. 1-14):

„The HP OpenView network management system works best if the segments on your networks have no more than 200 nodes each.
...
Keep in mind that although you may set up multiple management servers on your network, each management server does his own polling. Therefore the traffic on your network will increase in proportion to the number of managers on your network."

Das HP OpenView System arbeitet am besten, wenn in den Segmenten des Netzes nicht mehr als 200 Knoten vorhanden sind.
...
Beachten Sie, daß Sie mehrere Management-Server einrichten können, aber jeder Server betreibt sein eigenes Polling. Deshalb nimmt der Verkehr überproportional zu mit der Anzahl der Management-Server im Netz.

Dieser obige Ausschnitt sagt aus, daß es zwar möglich ist, mehrere Manager zu installieren, diese dann aber völlig autark arbeiten. Eine zentrale Sammlung der Management-Informationen ist somit nicht möglich. Über die Konfiguration der seed-Datei kann allerdings der Bereich des Autodiscovery konfiguriert werden.
Möglichkeiten für eine verteilte Installation sind somit nicht gegeben.

Im nächsten Kapitel werden die gewonnenen Ergebnisse zusammengefaßt und durch den Kriterienkatalog bewertet.

5.4 Abschließende Bewertung der Produkte

In den nächsten Abschnitten werden die Plattformen durch den Kriterienkatalog bewertet und gegeneinander verglichen. Die Kriterienkataloge werden mit Werten gefüllt, soweit Angaben zu den Anforderungen möglich waren. War eine Bewertung nicht möglich, da wurde als Gewichtung 1 und als Bewertung ebenfalls 1 verwendet.
Ansonsten wurden die Gewichtungen nach den Schwerpunkten, die sich durch die Aufgabenstellung des Tests ergaben, gesetzt.
Die Bewertung wurde nach möglichst objektiver Beurteilung beim Test anhand der beschriebenen Bewertungsskala in Kapitel 4 vorgenommen.
Durch die starke Ähnlichkeit der beiden Produkte, konnten die Zeitmessungen nicht auf eine Bewertung umgerechnet werden. Denn das Meßergebnis von *OpenView* ist immer durch die Kenntnisse von *NetView* beeinflußt gewesen. Deshalb wurden die Zeitmessungen nur als zusätzliche Entscheidungshilfe für die Bewertung verwendet.

5.4.1 Bewertung von *NetView*

In diesem Teil sollen die Erfahrungen durch den Test kurz zusammengefaßt und am Ende die Bewertungen des Test-Kriterienkataloges vergeben werden.
Die Sprache der Dokumentation ist Englisch. Die Autoren haben sich aber bemüht, ein einfaches Englisch zu schreiben. Eine Einteilung der Handbücher in Anfänger, Administrator etc. ist

vorhanden. Die Erklärungen sind verständlich mit guten Beispielen und hilfreichen „Screenshots“. Ein Index über alle Handbücher ist nicht vorhanden, stellenweise könnte der Index mehr Stichworte bieten. In jedem Buch sind Hinweise auf weiterführende Literatur vorhanden.
Für die Online-Dokumentation gilt im Prinzip das gleiche wie für die gebundenen Ausgaben. Außerdem ist eine leistungsfähige Suchmaschine über alle Online-Handbücher vorhanden. Bei fast allen Dialog-Masken gibt es eine kontextbezogene Hilfe.

Die Datenbasis selbst konnte nicht untersucht werden. Jeder Benutzer kann einen eigenen Benutzer-Übersichtsplan haben. Die Objekte, die zu einem bestimmten Benutzer-Übersichtsplan gehören, beeinflussen nicht die Übersichtspläne anderer Benutzer. Durch das Abspeichern des eigenen Benutzer-Übersichtsplans, kann indirekt eine Sicherung aller zugehörigen Objekte durchgeführt werden. Ändert ein Objekt seinen Zustand, so werden auch die Symbole der gerade nicht verwendeten Benutzer-Übersichtspläne aktualsiert.
Echte Vererbung ist nicht möglich. Die verschiedenen Symbole besitzen gewisse Eigenschaften, die auf das Objekt beim Zuweisen übergehen.

Benutzerspezifische Sichten sind möglich, wobei sich die Regel der Sicht auf eine physikalische Eigenschaft stützen muß. Eine Sicht, z.B. mit der Regel: ist Druckerserver, ist nicht möglich. Das Anlegen von solchen Sichten kann jedoch sehr umständlich und zeitraubend sein. Das Anlegen von aufgabenorientierten Sichten ist damit nicht möglich.
Alle Sichten können redundante Informationen enthalten, ein Symbol kann auf beliebig viele Übersichtspläne kopiert werden. Wird ein Objekt aus der Datenbank gelöscht, werden auch die zugehörigen Symbole entfernt.
Warnungen werden auf den üblichen Wegen ausgegeben: Akustik, Farbe, E-Mail. Als Besonderheit ist zu nennen, daß eine Ausgabe auf Pager möglich ist.
Die Bedienbarkeit ist bis auf einige wenige Stellen sehr gelungen und intuitiv.

Die Einteilung der Ereignisse in verschiedene Arten richtet sich nach der üblichen Methode im Internet-Management (normal, critical, etc.).
Das Auswertungswerkzeug, der Ruleset-Editor, bietet einiges an Möglichkeiten, wobei die Bedienbarkeit zu wünschen übrig läßt. Durch dieses Manko wird es auch schwierig, komplexere Regeln zu entwerfen. Positiv ist, daß auf einfache Weise die Ereignisse für ein bestimmtes Objekt gefiltert werden können.
Das Überwachen von Schwellwerten ist sehr komfortabel gelöst. Durch die Möglichkeit des Ruleset-Editor kann der Benutzer erst informiert werden, wenn mehrere Schwellwerte erreicht sind. Das Einrichten der Schwellwert-Überwachung ist ohne viele Kenntnisse möglich. Einziger Nachteil ist, daß die Lehrstuhl-MIB nicht unterstützt wurden.
Beim Überwachen einer MIB-Variable können die Daten gespeichert und später grafisch aufbereitet werden. Das Ausdrucken dieser Grafiken ist ebenfalls möglich. Ob mit diesen Grafiken Aussagen über Verkehrsentwicklungen oder ähnliches möglich sind, konnte wegen mangelnder Testumgebung nicht untersucht werden.

Die Übersichtspläne können jederzeit einfach gespeichert werden. Da ein Benutzer-Übersichtsplan aus einer ASCII-Datei und Informationen in der Datenbank bestehen, kann der Benutzer-Übersichtsplan nicht einfach transportiert werden (z.B. durch Sicherung auf transportable Medien).

Das Hierarchie-Konzept sind die bereits vorher genannten vier Ebenen. Über zusätzliche Symbole lassen sich weitere Hierarchien künstlich bilden. Diese Möglichkeit stößt bei vielen Komponenten sehr schnell an seine Grenzen.
Hilfsmittel zur Darstellung von Verkabelung sind nicht vorhanden.

Die Möglichkeit, Management-Verantwortung zu delegieren, ist anscheinend sehr gut beim NetView unterstützt. Schlecht ist, daß dazu die Installation einer kompletten Management-Plattform nötig ist.
Außerdem basiert die Kommunikation zwischen dem Zentralmanager und dem Submanager auf SNMP. Eine Minderung des Verkehrsaufkommens durch Einsatz eines speziellen, effizienten Kommunikationsprotokolls ist nicht möglich. Der Vorteil der bestehenden Lösung ist der mögliche Einsatz einer Management-Plattform eines anderen Herstellers.
Sehr gut ist das dynamische Verhalten des Netmon, der die Aufgabe je nach Verfügbarkeit verteilt. Bei der Domänenbildung ist die komplette Boolsche Algebra abgebildet, so daß ein mächtiges Werkzeug vorhanden ist. Die Beschreibungen der zusätzlich erhältlichen Produkte machen ebenfalls einen positiven Eindruck.

Nach gründlichen Überlegungen ergaben sich folgende Bewertungen der einzelnen Kriterien:

Unternehmens-Standpunkt

Produktinformationen			
Hersteller	IBM		
Produktbezeichnung	*NetView for AIX*		
Version/Patches	4.1		
Betriebsmittelbedarf Festplatte, Hauptspeicher, CPU-Leistung, Prozesse	64 MB Hauptspeicher 100 MB Festplattenspeicher 192 MB Paging-Speicher		
Betriebssysteme	*AIX* V3R2.5, PTF U428290		
Betriebswirtschaftliche Aspekte	**Gewichtung**	**Bewertung**	**Gesamt**
innovatives Produkt Oberfläche, Datenbasis, Anwendungen, Kommunikation, Entwicklungsumgebung	1	3	3
geplante Entwicklungen	1	1	1
Unterstützung der betrieblichen Prozesse	1	1	1
Benutzerzufriedenheit	1	1	1
Informationen für Benutzer	1	1	1
Kostenumlage	1	1	1
Positionierung des Produkts	1	1	1
Preis			
Lizenzpolitik	1	1	1
Leasing	1	1	1
Servicekonzepte Updates, Installation	1	1	1

Wartungsvertrag	1	1	1
modularer Aufbau	2	3	6
Lieferanten/Hersteller-Beziehungen			
Beziehungen zum Lieferanten, Hersteller	1	1	1
Zuverlässigkeit	1	1	1
Hotline Kosten, Kompetenz, Erreichbarkeit	1	1	1
Einfluß auf Entwicklung	1	1	1
Teststellung Kosten, Dauer, Umfang	1	1	1
Schulung Ort, Preis	1	1	1
Dokumentation			
Sprache	1	1	1
rollenspezifische Unterteilung	2	4	8
Verständlichkeit	3	4	12
Beispiele	3	3	9
Index	3	3	9
Literatur	1	5	5
Online-Ausgabe	3	5	15

Informations-Standpunkt

Datenbasis	**Gewichtung**	**Bewertung**	**Gesamt**
Typ von Datenbasis relational, Text, objektorientiert	2	3	6
Standardsoftware	2	5	10
verteilte Installation	1	1	1
Wechsel der Datenbasis	1	1	1
Backup-Datenbasis	2	3	6
Protokollierung von Datenbank-Änderungen	1	1	1
Werkzeuge Export-Funktionen, Löschen, Pflege, Datenmanipulation	1	1	1
Schnittstellen SQL, C	1	3	3
Struktur der Daten Namensgebung	1	1	1
unterstützte Objektkataloge Internet-MIBs, Experimental MIBs, Private MIBs, OSI-Objektkataloge, herstellerspezifische Objektkataloge, internes Format	2	5	10
Erweiterung der Objekte	1	1	1

Vererbung, Methodik der Erweiterung, Sprachmittel			
MIB-Browser	2	4	8
Zugriffsrechte auf Benutzer, Objekt, Datenbank	1	2	2
Oberflächen			
Arten von Oberflächen textorientiert, grafisch, 2-D, 3-D, WWW, Oberflächenintegration	2	3	6
grafische Elemente Erweiterung, editieren	1	1	1
Menü-Konfiguration	2	2	4
Sicherheitskonzept für die Oberflächen Sperren von Funktionen	2	2	4
Verstecken von Menüpunkten	2	2	4
benutzerspezifische Sichten nach Diensten, Attributen, physikalischen Gesichtspunkten, Aufgabe, Geschäftsprozesse	3	4	12
Sichten mit redundanter Information	3	4	12
Darstellung von Mobilkomponenten	1	1	1
Ausgabe von Zuständen Telefon, Email, Farbe, Audio, Zustandsänderung über Hierarchien, Konfiguration der Verbreitung von Zustandsänderungen	2	4	8
Bedienbarkeit	1	5	5
Anfrage-System			
Konsistenzprüfung	2	3	6
stochastische Auswertung	1	1	1
Trendanalysen	1	1	1
Erkennen von Engpässen	1	1	1
Lastermittlung von einzelnen Diensten	1	1	1

Funktions-Standpunkt

Konfigurationsmanagement	Gewichtung	Bewertung	Gesamt
Speicherung der Konfigurationsdaten Export, Wiederverwendbarkeit	2	3	6
Konfigurationsänderungen Zeitpunkt	1	1	1
Prüfung der Konfigurationsdaten Konsistenz, Plausibiltät	1	1	1
„Autodiscovery“	3	3	9

Umfang, Leistung, Konfiguration, Technik, Anwendungen, Sicherheitssysteme, Positionierung			
Bildung von Änderungsgruppen definierbare Gruppen mit beliebigen Attributen, ein Objekt in mehreren Gruppen	2	2	4
Darstellung der Konfigurationsdaten	1	1	1
Abfrage auf Komponenten	1	1	1
Anwendungen von Drittanbietern Integration, Nutzung der Datenbasis	1	1	1
Reports	1	1	1
Inventarisierung			
Konfigurationsdaten für Inventarisierung	1	1	1
zusätzliche Informationen zum Objekt Lieferant, Kontaktperson, Telefon, Attribute	1	1	1
Management der Ressourcen Verträge, Versicherung, Garantie	1	1	1
Speicherung der Daten	1	1	1
Arbeitsaufträge	1	1	1
Softwareverteilung			
Erkennung der Software-Komponenten	1	1	1
Abfrage der Software-Informationen definierbare Gruppen mit beliebigen Attributen, automatische Gruppierung	1	1	1
Handhabbarkeit	1	1	1
individuelle Installationsskripten	1	1	1
Installationszustände	1	1	1
Installationsintervalle	1	1	1
Lizenzüberwachung	1	1	1
Statusbericht	1	1	1
Definition von Fehlerereignissen	1	1	1
Sicherheitskonzepte für Software	1	1	1
Hierarchie-Konzept mehrstufige Verteilung	1	1	1
unterstützte Betriebssysteme	1	1	1
Notabbruch	1	1	1
„Undo"-Funktion	1	1	1
Informationsrecherche	1	1	1
„Remote"-Unterstützung	1	1	1
Prozesse	1	1	1
Rechte	1	1	1
Fehlermanagement			
Informationsgestalt	2	3	6

Gewichtung, gelieferte Information			
Intervalle	2	4	8
Schwellwerte	2	4	8
Arbeitsaufträge	1	1	1
Fehlerklassen für Zuständigkeit, Behandlungsart	1	1	1
Fehlerfortpflanzung	2	2	4
Algorithmen	1	1	1
spezielle Anwendungen Trouble Tickets	1	1	1
Disastermanagement			
Warnungsweiterleitung	1	1	1
Routenplanung	1	1	1
Arbeitsaufträge	1	1	1
Leistungsmanagement			
Lastermittlung Systemen, Diensten, Leitungen	1	1	1
Schwellwerte Zeit, Intervall, Speicherung	2	4	8
Lasterzeugung Dienste, fehlerhafte Pakete	1	1	1
Algorithmen für Last von Systemen	1	1	1
Darstellung	2	3	6
Anfragemaschine	1	1	1
Antwortzeiten	1	1	1
Last-Trends	1	1	1
Abrechnungsmanagement			
Verbrauchsdatenerfaßung Meßeinheit, Metriken	1	1	1
Kontenplan	1	1	1
Kontingent-Überwachung	1	1	1
Gruppenbildung Attribute	1	1	1
Vertrags-Management Wartung, Leasing, Garantie	1	1	1
Export-Funktionen	1	1	1
Reports	1	1	1
Sicherheitsmanagement			
Vertrauensbeziehungen Darstellung, zentrale Konfiguration	1	1	1

zentrale Benutzerverwaltung über verschiedene Betriebssysteme, Single-Sign-On, Dienste, Paßwort-Dauer, Paßwort-Aufbau, Paßwort-Politik, Paßwort-Sperrung	1	1	1
Sicherheitsverstöße Logging in separate Datei	1	1	1
Trennen des Sicherheitsmanagements Logging bei Änderungen	1	1	1
Firewalls	1	1	1
Reports	1	1	1
Ereignismanagement	**Gewichtung**	**Bewertung**	**Gesamt**
Ereignis-Arten individuelle Gewichtung, Attribute für Gewichtung, Reaktionen bei bestimmter Gewichtung	3	2	6
Speicherdauer individuelle, automatisches Löschen	1	1	1
Auswertungs-Werkzeuge	3	4	12
Verantwortungsbereiche für Ereignisse Verteilung	2	2	4
eigene Intelligenz für Ereignisse	1	1	1
Erkennung von Fehlerfortpflanzung	2	2	4
Schwellwerte	2	4	8
Langzeitmessungen Speicherung, Dauer	2	3	6
Meßwert-Darstellung	2	3	6
Stochastische Funktionen auf Meßwerte	1	1	1
Exportfunktionen für Meßwerte	1	1	1
Topologiemanagement			
Übersichtspläne-Verwaltung Import, Export, verschiedene Versionen, Snapshot	2	3	6
Qualität Übersichtlichkeit,Verbindungen, Beschriftung, Größe	2	3	6
Hierarchie-Konzept Anzahl, freie Zuordnung, Navigation	3	3	9
Symbol-Funktionen zusätzliche Symbole, Attribute, Auflösung	2	3	6
Darstellung der Verkabelung	1	1	1
Netzmanagement			
Protokollkonfiguration	1	1	1
Namensdienste	1	1	1

Routing Darstellung, Fernkonfiguration	1	1	1
WAN-Verbindungen	1	1	1
Datenaufkommen zwischen Netzknoten bez. Dienst	1	1	1
Netzerweiterungen	1	1	1
Netzsegmentierung	1	1	1
kabellose Verbindungen Verantwortung	1	1	1
Protokollanalysatoren			
Datenbearbeitung	1	1	1
gezieltes Abhören	1	1	1
prozentuale Netzlast	1	1	1
Protokollentschlüsselung	1	1	1
Kabelverbindungen	1	1	1
Lasterzeugung	1	1	1
Entwicklungsumgebung			
Schnittstellenbeschreibung	1	1	1
Programmbibliotheken	1	1	1
Programmiersprachen	1	1	1
Schnittstellen zu anderen Anwendungen	1	1	1
Agenten/Proxy-Agenten für Betriebssysteme, Anwendung	1	1	1
Entwicklungsassistent	1	1	1
Testumgebung	1	1	1
Releasefähigkeit	1	1	1
Speichermanagement			
Konfiguration Verschlüsselung, Komprimierung, Roboter	1	1	1
Funktionalität Zeitplan, Datenbank, partielle, volle, Ar chivierung, verschiedene Medien, Zugriffs-schutz, transparentes Restore	1	1	1
Agenten für Online-Backup/Restore	1	1	1
anwendungsbezogenes Backup	1	1	1
Anwendungsmanagement			
Benutzerverwaltung	1	1	1
Zustandsermittlung Definition von Zustandskontrollpunkten, Definition von Schwellwerten, Agenten	1	1	1
Start/Stop automatisiert	1	1	1

Lastermittlung von Diensten der Anwendung Auswirkung durch Anzahl der Benutzer	1	1	1
Betriebsmittelbedarf Prozesse, Ressourcen, Versionen	1	1	1
WWW-Management			
Sicherheit Zugriffsrechte auf Dokumente, Trennung Internet/Intranet, Firewall	1	1	1
WWW-Handel Statistiken, Speicherbedarf	1	1	1

Technologischer Standpunkt

Hardware	**Gewichtung**	**Bewertung**	**Gesamt**
Schnittstellen zu Protokollanalysatoren	2	6	12
Netzadapter	2	6	12
Software			
Betriebssysteme für Oberflächen	2	5	10
Ressourcenverbrauch der Oberflächen	2	3	6
Schnittstellen zu Standardsoftware	1	1	1
unterstützte Protokolle	2	3	6
Agenten mit spezieller Funktionalität, für Anwendungen, Integration von Drittanbietern, für gängige Betriebssysteme, mit Manage ment-Politk, Korrelation, Filterung, Intelli genz, Betriebsmittelbedarf	1	1	1
Management-Plattformen Protokolle	1	1	1
Netzmanagement-Architekturen Integration, Vollständigkeit, Erweiterbarkeit	1	1	1
Dokumenten-Archivierung	1	1	1
Trouble Tickets Zuweisung, Benachrichtigung, Eskalation, Verfolgung, Statistiken, Informationsbasis	1	1	1
Antivirus netzweit, automatisiert, Zeitplan, gute Erweiterbarkeit	1	1	1
besondere Werkzeuge	1	1	1
Installation			
Medium	1	1	1

Installationsbeschreibung	1	1	1
Vorbereitung Voraussetzungen	1	3	3
Installationsprozedur	1	1	1
Protokolle	1	1	1
Gateway-Funktion	1	3	3
Installationsdauer	1	1	1
Anwenderanpassung	1	1	1
De-Installation	1	4	4
verteilte Installation			
RMON-MIB welche, Funktionsumfang, Geräte, Konfiguration	2	4	8
Aufgabenverteilung	2	3	6
verteilte Management-Struktur	3	4	12
hierarchische Management-Struktur	1	1	1
Netzwerk-Management-Struktur	1	1	1
GESAMT	251		525

5.4.2 Bewertung von *OpenView*

In diesem Abschnitt werden die Test-Ergebnisse ovn *OpenView* kurz zusammengefaßt. Am Ende erfolgt die Bewertung durch den Kriterienkatalog.
Auch bei HP ist die Dokumentation in Englisch, leider sind die Handbücher nicht gut geschrieben. Es fehlt sehr oft an Beispielen, schwierige Sachverhalte werden oft zu kurz beschrieben. Eine Einteilung der Handbücher in Themenbereiche (z.B. Installation) ist vorhanden, aber keine rollenspezifische Einteilung der Dokumentation. Der Index ist gut gelungen und mit vielen Stichworten versehen, auf Angabe von weiterführender Literatur wurde verzichtet.
Das Suchen in der Online-Dokumentation dauert sehr lange und die Bedienung ist sehr umständlich. Auch bei *OpenView* gibt es die kontextbezogene Hilfe, allerdings nicht in dem Ausmaß wie bei NetView.

OpenView bietet nicht die Möglichkeit, benutzerspezifische Sichten anzulegen, in denen neue Ressourcen automatisch aufgenommen werden (wie bei Collections unter *NetView*).

Ein komplexes Werkzeug zum Erstellen von Filter-Regeln gibt es nicht, die Möglichkeiten vom Event Browser sind zwar gering, dafür ist die Bedienung sehr einfach und ermöglicht schnelle Erfolge beim Filtern.

Eine verteilte Installation ist nicht möglich und kann damit nicht bewertet werden.

Bei folgenden Punkten sind keine Änderungen gegenüber *NetView*:

- Datenbasis und Übersichtspläne-Verwaltung

- Bedienbarkeit
- Einteilung von Ereignissen

Somit ergaben sich folgende Bewertungen der einzelnen Kriterien:

Unternehmens-Standpunkt

Produktinformationen			
Hersteller	HP		
Produktbezeichnung	*OpenView Network Node Manager*		
Version/Patches	B.04.01		
Betriebsmittelbedarf Festplatte, Hauptspeicher, CPU-Leistung, Prozesse	65 MB Hauptspeicher		
Betriebssysteme	*HP-UX 9.0*		
Betriebswirtschaftliche Aspekte	**Gewichtung**	**Bewertung**	**Gesamt**
innovatives Produkt Oberfläche, Datenbasis, Anwendungen, Kommunikation, Entwicklungsumgebung	1	2	2
geplante Entwicklungen	1	1	1
Unterstützung der betrieblichen Prozesse	1	1	1
Benutzerzufriedenheit	1	1	1
Informationen für Benutzer	1	1	1
Kostenumlage	1	1	1
Positionierung des Produkts	1	1	1
Preis			
Lizenzpolitik	1	1	1
Leasing	1	1	1
Servicekonzepte Updates, Installation	1	1	1
Wartungsvertrag	1	1	1
modularer Aufbau	2	3	6
Lieferanten/Hersteller-Beziehungen			
Beziehungen zum Lieferanten, Hersteller	1	1	1
Zuverlässigkeit	1	1	1
Hotline Kosten, Kompetenz, Erreichbarkeit	1	1	1
Einfluß auf Entwicklung	1	1	1
Teststellung Kosten, Dauer, Umfang	1	1	1
Schulung	1	1	1

Ort, Preis			
Dokumentation			
Sprache	1	1	1
rollenspezifische Unterteilung	2	3	6
Verständlichkeit	3	3	9
Beispiele	3	2	6
Index	3	4	12
Literatur	1	1	1
Online-Ausgabe	3	3	9

Informations-Standpunkt

Datenbasis	**Gewichtung**	**Bewertung**	**Gesamt**
Typ von Datenbasis relational, Text, objektorientiert	2	3	6
Standardsoftware	2	5	10
verteilte Installation	1	1	1
Wechsel der Datenbasis	1	1	1
Backup-Datenbasis	2	3	6
Protokollierung von Datenbank-Änderungen	1	1	1
Werkzeuge Export-Funktionen, Löschen, Pflege, Datenmanipulation	1	1	1
Schnittstellen SQL, C	1	3	3
Struktur der Daten Namensgebung	1	1	1
unterstützte Objektkataloge Internet-MIBs, Experimental MIBs, Private MIBs, OSI-Objektkataloge, herstellerspezifische Objektkataloge, internes Format	2	5	10
Erweiterung der Objekte Vererbung, Methodik der Erweiterung, Sprachmittel	1	1	1
MIB-Browser	2	4	8
Zugriffsrechte auf Benutzer, Objekt, Datenbank	1	2	2
Oberflächen			
Arten von Oberflächen textorientiert, grafisch, 2-D, 3-D, WWW, Oberflächenintegration	2	3	6
grafische Elemente Erweiterung, editieren	1	1	1
Menü-Konfiguration	2	2	4
Sicherheitskonzept für die Oberflächen	2	2	4

Sperren von Funktionen			
Verstecken von Menüpunkten	2	2	4
benutzerspezifische Sichten nach Diensten, Attributen, physikalischen Gesichtspunkten, Aufgabe, Geschäftsprozesse	3	2	6
Sichten mit redundanter Information	3	4	12
Darstellung von Mobilkomponenten	1	1	1
Ausgabe von Zuständen Telefon, Email, Farbe, Audio, Zustandsänderung über Hierarchien, Konfiguration der Verbreitung von Zustandsänderungen	2	2	4
Bedienbarkeit	1	6	6
Anfrage-System			
Konsistenzprüfung	2	3	6
stochastische Auswertung	1	1	1
Trendanalysen	1	1	1
Erkennen von Engpässen	1	1	1
Lastermittlung von einzelnen Diensten	1	1	1

Funktions-Standpunkt

Konfigurationsmanagement	**Gewichtung**	**Bewertung**	**Gesamt**
Speicherung der Konfigurationsdaten Export, Wiederverwendbarkeit	2	3	6
Konfigurationsänderungen Zeitpunkt	1	1	1
Prüfung der Konfigurationsdaten Konsistenz, Plausibiltät	1	1	1
„Autodiscovery“ Umfang, Leistung, Konfiguration, Technik, Anwendungen, Sicherheitssysteme, Positionierung	3	2	6
Bildung von Änderungsgruppen definierbare Gruppen mit beliebigen Attributen, ein Objekt in mehreren Gruppen	2	1	2
Darstellung der Konfigurationsdaten	1	1	1
Abfrage auf Komponenten	1	1	1
Anwendungen von Drittanbietern Integration, Nutzung der Datenbasis	1	1	1
Reports	1	1	1
Inventarisierung			
Konfigurationsdaten für Inventarisierung	1	1	1
zusätzliche Informationen zum Objekt	1	1	1

Lieferant, Kontaktperson, Telefon, Attribute			
Management der Ressourcen Verträge, Versicherung, Garantie	1	1	1
Speicherung der Daten	1	1	1
Arbeitsaufträge	1	1	1
Softwareverteilung			
Erkennung der Software-Komponenten	1	1	1
Abfrage der Software-Informationen definierbare Gruppen mit beliebigen Attributen, automatische Gruppierung	1	1	1
Handhabbarkeit	1	1	1
individuelle Installationsskripten	1	1	1
Installationszustände	1	1	1
Installationsintervalle	1	1	1
Lizenzüberwachung	1	1	1
Statusbericht	1	1	1
Definition von Fehlerereignissen	1	1	1
Sicherheitskonzepte für Software	1	1	1
Hierarchie-Konzept mehrstufige Verteilung	1	1	1
unterstützte Betriebssysteme	1	1	1
Notabbruch	1	1	1
„Undo“-Funktion	1	1	1
Informationsrecherche	1	1	1
„Remote“-Unterstützung	1	1	1
Prozesse	1	1	1
Rechte	1	1	1
Fehlermanagement			
Informationsgestalt Gewichtung, gelieferte Information	2	2	4
Intervalle	2	4	8
Schwellwerte	2	4	8
Arbeitsaufträge	1	1	1
Fehlerklassen für Zuständigkeit, Behandlungsart	1	1	1
Fehlerfortpflanzung	2	2	4
Algorithmen	1	1	1
spezielle Anwendungen Trouble Tickets	1	1	1
Disastermanagement			
Warnungsweiterleitung	1	1	1
Routenplanung	1	1	1
Arbeitsaufträge	1	1	1

Leistungsmanagement			
Lastermittlung Systemen, Diensten, Leitungen	1	1	1
Schwellwerte Zeit, Intervall, Speicherung	2	3	6
Lasterzeugung Dienste, fehlerhafte Pakete	1	1	1
Algorithmen für Last von Systemen	1	1	1
Darstellung	2	2	4
Anfragemaschine	1	1	1
Antwortzeiten	1	1	1
Last-Trends	1	1	1
Abrechnungsmanagement			
Verbrauchsdatenerfaßung Meßeinheit, Metriken	1	1	1
Kontenplan	1	1	1
Kontingent-Überwachung	1	1	1
Gruppenbildung Attribute	1	1	1
Vertrags-Management Wartung, Leasing, Garantie	1	1	1
Export-Funktionen	1	1	1
Reports	1	1	1
Sicherheitsmanagement			
Vertrauensbeziehungen Darstellung, zentrale Konfiguration	1	1	1
zentrale Benutzerverwaltung über verschiedene Betriebssysteme, Single-Sign-On, Dienste, Paßwort-Dauer, Paßwort-Aufbau, Paßwort-Politik, Paßwort-Sperrung	1	1	1
Sicherheitsverstöße Logging in separate Datei	1	1	1
Trennen des Sicherheitsmanagements Logging bei Änderungen	1	1	1
Firewalls	1	1	1
Reports	1	1	1
Ereignismanagement	**Gewichtung**	**Bewertung**	**Gesamt**
Ereignis-Arten individuelle Gewichtung, Attribute für Gewichtung, Reaktionen bei bestimmter Gewichtung	3	2	6
Speicherdauer	1	1	1

individuelle, automatisches Löschen			
Auswertungs-Werkzeuge	3	2	6
Verantwortungsbereiche für Ereignisse Verteilung	2	1	2
eigene Intelligenz für Ereignisse	1	1	1
Erkennung von Fehlerfortpflanzung	2	2	4
Schwellwerte	2	4	8
Langzeitmessungen Speicherung, Dauer	2	3	6
Meßwert-Darstellung	2	2	4
Stochastische Funktionen auf Meßwerte	1	1	1
Exportfunktionen für Meßwerte	1	1	1
Topologiemanagement			
Übersichtspläne-Verwaltung Import, Export, verschiedene Versionen, Snapshot	2	3	6
Qualität Übersichtlichkeit, Verbindungen, Beschriftung, Größe	2	3	6
Hierarchie-Konzept Anzahl, freie Zuordnung, Navigation	3	2	6
Symbol-Funktionen zusätzliche Symbole, Attribute, Auflösung	2	3	6
Darstellung der Verkabelung	1	1	1
Netzmanagement			
Protokollkonfiguration	1	1	1
Namensdienste	1	1	1
Routing Darstellung, Fernkonfiguration	1	1	1
WAN-Verbindungen	1	1	1
Datenaufkommen zwischen Netzknoten bez. Dienst	1	1	1
Netzerweiterungen	1	1	1
Netzsegmentierung	1	1	1
kabellose Verbindungen Verantwortung	1	1	1
Protokollanalysatoren			
Datenbearbeitung	1	1	1
gezieltes Abhören	1	1	1
prozentuale Netzlast	1	1	1
Protokollentschlüsselung	1	1	1
Kabelverbindungen	1	1	1
Lasterzeugung	1	1	1

Entwicklungsumgebung			
Schnittstellenbeschreibung	1	1	1
Programmbibliotheken	1	1	1
Programmiersprachen	1	1	1
Schnittstellen zu anderen Anwendungen	1	1	1
Agenten/Proxy-Agenten für Betriebssysteme, Anwendung	1	1	1
Entwicklungsassistent	1	1	1
Testumgebung	1	1	1
Releasefähigkeit	1	1	1
Speichermanagement			
Konfiguration Verschlüsselung, Komprimierung, Roboter	1	1	1
Funktionalität Zeitplan, Datenbank, partielle, volle, Ar chivierung, verschiedene Medien, Zugriffs-schutz, transparentes Restore	1	1	1
Agenten für Online-Backup/Restore	1	1	1
anwendungsbezogenes Backup	1	1	1
Anwendungsmanagement			
Benutzerverwaltung	1	1	1
Zustandsermittlung Definition von Zustandskontrollpunkten, Definition von Schwellwerten, Agenten	1	1	1
Start/Stop automatisiert	1	1	1
Lastermittlung von Diensten der Anwendung Auswirkung durch Anzahl der Benutzer	1	1	1
Betriebsmittelbedarf Prozesse, Ressourcen, Versionen	1	1	1
WWW-Management			
Sicherheit Zugriffsrechte auf Dokumente, Trennung Internet/Intranet, Firewall	1	1	1
WWW-Handel Statistiken, Speicherbedarf	1	1	1

Technologischer Standpunkt

Hardware	**Gewichtung**	**Bewertung**	**Gesamt**
Schnittstellen zu Protokollanalysatoren	2	6	12
Netzadapter	2	6	12
Software			
Betriebssysteme für Oberflächen	2	5	10
Ressourcenverbrauch der Oberflächen	2	3	6
Schnittstellen zu Standardsoftware	1	1	1
unterstützte Protokolle	2	3	6
Agenten mit spezieller Funktionalität, für Anwendungen, Integration von Drittanbietern, für gängige Betriebssysteme, mit Manage ment-Politk, Korrelation, Filterung, Intelligenz, Betriebsmittelbedarf	1	1	1
Management-Plattformen Protokolle	1	1	1
Netzmanagement-Architekturen Integration, Vollständigkeit, Erweiterbarkeit	1	1	1
Dokumenten-Archivierung	1	1	1
Trouble Tickets Zuweisung, Benachrichtigung, Eskalation, Verfolgung, Statistiken, Informationsbasis	1	1	1
Antivirus netzweit, automatisiert, Zeitplan, gute Erweiterbarkeit	1	1	1
besondere Werkzeuge	1	1	1
Installation			
Medium	1	1	1
Installationsbeschreibung	1	1	1
Vorbereitung Voraussetzungen	1	1	1
Installationsprozedur	1	1	1
Protokolle	1	1	1
Gateway-Funktion	1	3	3
Installationsdauer	1	1	1
Anwenderanpassung	1	1	1
De-Installation	1	1	1
verteilte Installation			
RMON-MIB welche, Funktionsumfang, Geräte,	2	4	8

Konfiguration			
Aufgabenverteilung	2	1	2
verteilte Management-Struktur	3	1	3
hierarchische Management-Struktur	1	1	1
Netzwerk-Management-Struktur	1	1	1
GESAMT	251		458

5.4.3 Interpretation der Ergebnisse

In diesem letzten Abschnitt der Anwendung, sollen die Ergebnisse durch den Kriterienkatalog interpretiert werden.
Als Gesamtwert der Gewichtungen erhielt man 251. Dieser Wert ist in dieser Form noch nicht sehr aussagekräftig, deshalb wird der Mittelwert der Gewichtungen gebildet.
Der Mittelwert läßt sich wie folgt bilden:

$$M_{Gew} = \frac{1}{n} \cdot \sum_{1}^{n} g$$

Was bedeutet, daß die Summe der Gewichtungen durch die Anzahl der Kriterien dividiert werden muß. Bei diesem Kriterienkatalog ergibt sich also:

$$M_{Gew} = \frac{1}{193} \cdot \sum_{1}^{193} g = \frac{251}{193} = 1{,}30$$

Da bei dieser Anwendung des Kriterienkataloges alle Anforderungen, die nicht getestet wurden (im weiteren 1-Kriterien genannt), auf Gewichtung 1 gesetzt sind, ist es wichtig, diesen Mittelwert zu errechnen. Daran erkennt man, ob die Anzahl der 1-Kriterien überwiegt, wie es in diesem Fall durch die geringe Differenz zwischen der 1 und dem Mittelwert zu erkennen ist. Tatsächlich wurden 55 Kriterien bewertet, die restlichen konnten nicht getestet werden.
Doch nun sollen noch die Ergebnisse der Bewertungen errechnet werden (siehe dazu Kapitel 4), um einen Vergleich zwischen *NetView* und *OpenView* zu ermöglichen. Dies geschieht nach folgender Gleichung:

$$E_{Ges} = \frac{\sum_{1}^{n} p}{\sum_{1}^{n} g}$$

Um die Gesamtbewertung auf die Bewertungsskala umzurechnen, muß die Summe aller Produkte von Gewichtung und Bewertung eines jeden Kriteriums gebildet werden. Diese Summe wird dann durch die Summe aller Gewichtungen dividiert. So errechnet man für *NetView*:

$$E_{NV} = \frac{\sum_1^n p}{\sum_1^n g} = \frac{525}{251} = 2{,}09$$

Bei *OpenView* erhält man:

$$E_{OV} = \frac{\sum_1^n p}{\sum_1^n g} = \frac{458}{251} = 1{,}83$$

Dies entspricht in Worten ein **schlecht** für *NetView* und ein **sehr schlecht** für *OpenView* auf der Bewertungsskala.
Trotz der teilweise gut bis sehr guten Fähigkeiten von *NetView* ist die Gesamt-Bwertung so schlecht ausgefallen. Dies liegt an der sehr hohen Beteiligung von 1-Kriterien, die nicht bewertet werden konnten. Die gewählte Bewertungstechnik geht von einer großen Verwendung von getesteten Kriterien (im weiteren g-Kriterien genannt) aus. Die beschriebenen Techniken in Kapitel 2.5 sollte der Betreiber nutzen, um diejenigen Kriterien zu ermitteln, die getestet werden können und von Interesse für das Unternehmen sind. Bei einer Bewertung, in der die Anzahl der 1-Kriterien größer als die Anzahl der g-Kriterien ist, wird immer das Gesamt-Ergebnis verfälscht sein.
Damit trotzdem ein Vergleich zwischen den Management-Plattformen möglich ist, wurde der Einfluß der 1-Kriterien herausgerechnet.
Dazu mußte die Anzahl der g-Kriterien ermittelt werden, dies waren in diesem Test 55. Jetzt muß noch der Einfluß der 1-Kriterien aus der Gesamtsumme entfernt werden. Dazu müssen von den 193 Anforderungen 55 subtrahiert werden, somit ergibt die Anzahl der 1-Kriterien 138. Es wurde also 138 auf die Summe der Kriterien addiert. Die Gesamtsummen lauten dann:

$$E_{NV} = 525 - 138 = 387$$
$$E_{OV} = 458 - 138 = 320$$

Ebenso muß der Einfluß aus den Gewichtungen entfernt werden:

$$g_n = 251 - 138 = 113$$

Nach obiger Gleichung kann nun die Bewertung berechnet werden:

$E_{NV} = \frac{387}{113} = 3{,}42$ entspricht einer glatten 3, also **neutral**

$E_{OV} = \frac{320}{113} = 2{,}83$ entspricht einer 2+, also **schlecht**

Auf diese Weise läßt sich der Einfluß durch die 1-Kriterien entfernen und eine gültige Entscheidung treffen. In diesem Test ist *NetView* in bezug auf die gestellten Aufgaben, die bessere Wahl. Werden andere Kriterien bewertet oder andere Schwerpunkte gesetzt, könnte auch *OpenView* der Sieger sein.

Natürlich ist der beschriebene Einfluß durch die 1-Kriterien durchaus gewollt. Denn werden Anforderungen bei so einem Test aufgenommen, weil sie von Interesse sind, können sie bei der Durchführung aber doch nicht überprüft werden (z.B. aufgrund von technischen Problemen, nicht funktionierender Treiber). Dann sollen diese trotzdem einen Einfluß bei der Bewertung haben. Der Anwender des Kriterienkataloges muß also immer einen Kompromiß schließen. Welche Kriterien läßt er weg, weil diese nicht überprüft werden können, und welche fügt er hinzu, weil sie wichtig sind, auch wenn sie nicht getestet werden können.

Im nächsten Kapitel wird ein Ausblick auf mögliche Entwicklungen beim Kriterienkatalog gegeben.

6 Abschließende Betrachtung

Von den vielen Anforderungen, die an ein Management-System gestellt werden können, wurden in dieser Arbeit nur die wichtigesten Kriterien betrachtet. Durch die Integration von bestehenden Konzepten bei der Strukturierung, wurde ein leicht zu erweiterndes Rahmenwerk entworfen. Der Anwender kann auf einfache Weise neue Themenbereiche mit den dazugehörigen Kriterien einfügen.
Bei der Betrachtung des Betreiber-Szenarios wurde deutlich gemacht, daß die tatsächlichen Management-Lösungen noch weit von den Wünschen der Betreiber entfernt sind. Es besteht also noch akuter Handlungsbedarf, um die neuesten Trends der verteilten Verarbeitung auch überwachen zu können.
Außerdem wurden einige Bewertungstechniken vorgestellt, um eine Grundlage für die Bewertung der Kriterien zu ermöglichen. Hier besteht allerdings noch großer Handlungsbedarf. Die Unterschiedlichkeit bei der Beantwortung von Anforderungen

- Ja/Nein
- Messung mit verschiedenen Einheiten
- subjektiver Eindruck
- Dokumentationsangaben

erschwert die Bewertung zusätzlich. Könnte eine einheitliches Bewertungsschema auf alle Kriterien mit gleicher Qualität angewendet werden, würde dies die Transparenz des Ergebnisses erheblich verbessern.
Die Anwendung der Bewertung muß zusätzlich vereinfacht werden, damit diese schnell und einfach von jedem durchgeführt werden kann.

Vielleicht versuchen aber auch die Hersteller ein Entscheidungs-Werkzeug für Management-Plattformen zu entwickeln. Ein Test-Werkzeug ähnlich einem Benchmark, der an definierten Stellen die Leistung eines Management-Systems bei bestimmten Rahmenbedingungen testet, wäre ein erster Schritt in diese Richtung. Natürlich kann die Entwicklung zu unabhängigen Management-Anwendungen gehen, die in beliebige Rahmenwerke integriert werden können. Diese Rahmenwerke stellen die Ablösung der klassischen Management-Plattform dar. Sie bieten eine große Anzahl von Schnittstellen, über die dann alle Management-Anwendungen ihre Informationen austauschen, um die Management-Leistungen zu erbringen.

Anhang A Kriterienkatalog

Unternehmens-Standpunkt

Produktinformationen			
Hersteller			
Produktbezeichnung			
Version/Patches			
Betriebsmittelbedarf Festplatte, Hauptspeicher, CPU-Leistung, Prozesse			
Betriebssysteme			
Betriebswirtschaftliche Aspekte	**Gewichtung**	**Bewertung**	**Gesamt**
innovatives Produkt Oberfläche, Datenbasis, Anwendungen, Kommunikation, Entwicklungsumgebung			
geplante Entwicklungen			
Unterstützung der betrieblichen Prozesse			
Benutzerzufriedenheit			
Informationen für Benutzer			
Kostenumlage			
Positionierung des Produkts			
Preis			
Lizenzpolitik			
Leasing			
Servicekonzepte Updates, Installation			
Wartungsvertrag			
modularer Aufbau			
Lieferanten/Hersteller-Beziehungen			
Beziehungen zum Lieferanten, Hersteller			
Zuverlässigkeit			
Hotline Kosten, Kompetenz, Erreichbarkeit			
Einfluß auf Entwicklung			
Teststellung Kosten, Dauer, Umfang			

Schulung Ort, Preis			
Dokumentation			
Sprache			
rollenspezifische Unterteilung			
Verständlichkeit			
Beispiele			
Index			
Literatur			
Online-Ausgabe			

Informations-Standpunkt

Datenbasis	**Gewichtung**	**Bewertung**	**Gesamt**
Typ von Datenbasis relational, Text, objektorientiert			
Standardsoftware			
verteilte Installation			
Wechsel der Datenbasis			
Backup-Datenbasis			
Protokollierung von Datenbank-Änderungen			
Werkzeuge Export-Funktionen, Löschen, Pflege, Datenmanipulation			
Schnittstellen SQL, C			
Struktur der Daten Namensgebung			
unterstützte Objektkataloge Internet-MIBs, Experimental MIBs, Private MIBs, OSI-Objektkataloge, herstellerspezifische Objektkataloge, internes Format			
Erweiterung der Objekte Vererbung, Methodik der Erweiterung, Sprachmittel			
MIB-Browser			
Zugriffsrechte auf Benutzer, Objekt, Datenbank			
Oberflächen			
Arten von Oberflächen textorientiert, grafisch, 2-D, 3-D, WWW, Oberflächenintegration			
grafische Elemente Erweiterung, editieren			
Menü-Konfiguration			

Sicherheitskonzept für die Oberflächen Sperren von Funktionen			
Verstecken von Menüpunkten			
benutzerspezifische Sichten nach Diensten, Attributen, physikalischen Gesichtspunkten, Aufgabe, Geschäftsprozesse			
Sichten mit redundanter Information			
Darstellung von Mobilkomponenten			
Ausgabe von Zuständen Telefon, Email, Farbe, Audio, Zustandsänderung über Hierarchien, Konfiguration der Verbreitung von Zustandsänderungen			
Bedienbarkeit			
Anfrage-System			
Konsistenzprüfung			
stochastische Auswertung			
Trendanalysen			
Erkennen von Engpässen			
Lastermittlung von einzelnen Diensten			

Funktions-Standpunkt

Konfigurationsmanagement			
Speicherung der Konfigurationsdaten Export, Wiederverwendbarkeit			
Konfigurationsänderungen Zeitpunkt			
Prüfung der Konfigurationsdaten Konsistenz, Plausibiltät			
„Autodiscovery“ Umfang, Leistung, Konfiguration, Technik, Anwendungen, Sicherheitssysteme, Positionierung			
Bildung von Änderungsgruppen definierbare Gruppen mit beliebigen Attributen, ein Objekt in mehreren Gruppen			
Darstellung der Konfigurationsdaten			
Abfrage auf Komponenten			
Anwendungen von Drittanbietern Integration, Nutzung der Datenbasis			
Reports			
Inventarisierung			

Konfigurationsdaten für Inventarisierung			

zusätzliche Informationen zum Objekt Lieferant, Kontaktperson, Telefon, Attribute			
Management der Ressourcen Verträge, Versicherung, Garantie			
Speicherung der Daten			
Arbeitsaufträge			
Softwareverteilung			
Erkennung der Software-Komponenten			
Abfrage der Software-Informationen definierbare Gruppen mit beliebigen Attributen, automatische Gruppierung			
Handhabbarkeit			
individuelle Installationsskripten			
Installationszustände			
Installationsintervalle			
Lizenzüberwachung			
Statusbericht			
Definition von Fehlerereignissen			
Sicherheitskonzepte für Software			
Hierarchie-Konzept mehrstufige Verteilung			
unterstützte Betriebssysteme			
Notabbruch			
„Undo"-Funktion			
Informationsrecherche			
„Remote"-Unterstützung			
Prozesse			
Rechte			
Fehlermanagement			
Informationsgestalt Gewichtung, gelieferte Information			
Intervalle			
Schwellwerte			
Arbeitsaufträge			
Fehlerklassen für Zuständigkeit, Behandlungsart			
Fehlerfortpflanzung			
Algorithmen			
spezielle Anwendungen Trouble Tickets			
Disastermanagement			

Warnungsweiterleitung			
Routenplanung			
Arbeitsaufträge			
Leistungsmanagement			
Lastermittlung Systemen, Diensten, Leitungen			
Schwellwerte Zeit, Intervall, Speicherung			
Lasterzeugung Dienste, fehlerhafte Pakete			
Algorithmen für Last von Systemen			
Darstellung			
Anfragemaschine			
Antwortzeiten			
Last-Trends			
Abrechnungsmanagement			
Verbrauchsdatenerfaßung Meßeinheit, Metriken			
Kontenplan			
Kontingent-Überwachung			
Gruppenbildung Attribute			
Vertrags-Management Wartung, Leasing, Garantie			
Export-Funktionen			
Reports			
Sicherheitsmanagement			
Vertrauensbeziehungen Darstellung, zentrale Konfiguration			
zentrale Benutzerverwaltung über verschiedene Betriebssysteme, Single-Sign-On, Dienste, Paßwort-Dauer, Paßwort-Aufbau, Paßwort-Politik, Paßwort-Sperrung			
Sicherheitsverstöße Logging in separate Datei			
Trennen des Sicherheitsmanagements Logging bei Änderungen			
Firewalls			
Reports			

Ereignismanagement	**Gewichtung**	**Bewertung**	**Gesamt**
Ereignis-Arten individuelle Gewichtung, Attribute für Gewichtung, Reaktionen bei bestimmter Gewichtung			
Speicherdauer individuelle, automatisches Löschen			
Auswertungs-Werkzeuge			
Verantwortungsbereiche für Ereignisse Verteilung			
eigene Intelligenz für Ereignisse			
Erkennung von Fehlerfortpflanzung			
Schwellwerte			
Langzeitmessungen Speicherung, Dauer			
Meßwert-Darstellung			
Stochastische Funktionen auf Meßwerte			
Exportfunktionen für Meßwerte			
Topologiemanagement			
Übersichtspläne-Verwaltung Import, Export, verschiedene Versionen, Snapshot			
Qualität Übersichtlichkeit, Verbindungen, Beschriftung, Größe			
Hierarchie-Konzept Anzahl, freie Zuordnung, Navigation			
Symbol-Funktionen zusätzliche Symbole, Attribute, Auflösung			
Darstellung der Verkabelung			
Netzmanagement			
Protokollkonfiguration			
Namensdienste			
Routing Darstellung, Fernkonfiguration			
WAN-Verbindungen			
Datenaufkommen zwischen Netzknoten bez. Dienst			
Netzerweiterungen			
Netzsegmentierung			
kabellose Verbindungen Verantwortung			

Protokollanalysatoren			
Datenbearbeitung			
gezieltes Abhören			
prozentuale Netzlast			
Protokollentschlüsselung			
Kabelverbindungen			
Lasterzeugung			
Entwicklungsumgebung			
Schnittstellenbeschreibung			
Programmbibliotheken			
Programmiersprachen			
Schnittstellen zu anderen Anwendungen			
Agenten/Proxy-Agenten für Betriebssysteme, Anwendung			
Entwicklungsassistent			
Testumgebung			
Releasefähigkeit			
Speichermanagement			
Konfiguration Verschlüsselung, Komprimierung, Roboter			
Funktionalität Zeitplan, Datenbank, partielle, volle, Archivierung, verschiedene Medien, Zugriffsschutz, transparentes Restore			
Agenten für Online-Backup/Restore			
anwendungsbezogenes Backup			
Anwendungsmanagement			
Benutzerverwaltung			
Zustandsermittlung Definition von Zustandskontrollpunkten, Definition von Schwellwerten, Agenten			
Start/Stop automatisiert			
Lastermittlung von Diensten der Anwendung Auswirkung durch Anzahl der Benutzer			
Betriebsmittelbedarf Prozesse, Ressourcen, Versionen			

WWW-Management			
Sicherheit Zugriffsrechte auf Dokumente, Trennung Internet/Intranet, Firewall			
WWW-Handel Statistiken, Speicherbedarf			

Technologischer Standpunkt

Hardware	**Gewichtung**	**Bewertung**	**Gesamt**
Schnittstellen zu Protokollanalysatoren			
Netzadapter			
Software			
Betriebssysteme für Oberflächen			
Ressourcenverbrauch der Oberflächen			
Schnittstellen zu Standardsoftware			
unterstützte Protokolle			
Agenten mit spezieller Funktionalität,für Anwendungen, Integration von Drittanbietern, für gängige Betriebssysteme, mit Management-Politik, Korrelation, Filterung, Intelligenz, Betriebsmittelbedarf			
Management-Plattformen Protokolle			
Netzmanagement-Architekturen Integration, Vollständigkeit, Erweiterbarkeit			
Dokumenten-Archivierung			
Trouble Tickets Zuweisung, Benachrichtigung, Eskalation, Verfolgung, Statistiken, Informationsbasis			
Antivirus netzweit, automatisiert, Zeitplan, gute Erweiterbarkeit			
besondere Werkzeuge			

Installation			
Medium			
Installationsbeschreibung			
Vorbereitung Voraussetzungen			
Installationsprozedur			
Protokolle			
Gateway-Funktion			
Installationsdauer			
Anwenderanpassung			
De-Installation			
verteilte Installation			
RMON-MIB welche, Funktionsumfang, Geräte, Konfiguration			
Aufgabenverteilung			
verteilte Management-Struktur			
hierarchische Management-Struktur			
Netzwerk-Management-Struktur			
GESAMT			

Abkürzungsverzeichnis

ABAP	Advanced Business Application Programming
AIX	Advanced Interactive Executive
API	Application Programming Interface
ARP	Address Resolution Protocol
ASCII	American Standard Code for Information Interchange
ASN1	Abstract Syntax Notation 1
CCMS	Computing Center Management System
CMIP	Common Management Information Protocol
CMIS	Common Management Information Service
CMOT	Common Management Information Services and Protocol over TCP
DB	Datenbank
DCE	Distributed Computer Enviroment oder Data circuit terminating equipment
DME	Distributed Management Enviroment
DNS	Domain Name System
Dynpro	Dynamisches Programm
FDDI	Fiber Distributed Data Interface
FTP	File Transfer Protocol
GUI	Graphical User Interface
GMD	Gesellschaft für Mathematik und Datenverarbeitung mbH
HP	Hewlett Packard
IBM	International Buisness Machines
ICMP	Internet Control Message Protocol
IEEE	Institute of Electrical and Electronics Engineers
IETF	Internet Engineering Task Force
IP	Internet Protocol
IPX	Internetwork Packet Exchange
ISDN	Integrated Services Digital Network
ISO	International Organization for Standardization
IT	Informations Technologien
LAN	Local Area Network
MAC	Medium Access Control
MAN	Metropolitan Area Network
MIB	Management Information Base
MOM	Manager-of-Managers
ODBC	Open Database Connectivity
OSI	Open Systems Interconnection
PC	Personal Computer
PING	Packet Internet Groper
R/3	Runtimesystem 3
RDBMS	Relational Database Management System
RFC	Remote Function Call
RMON	Remote Network Monitoring
SAP	Systeme, Anwendungen, Produkte
SMIT	System Management Interface Tool
SNA	Systems Network Architecture

SNMP	Simple Network Management Protocol
SPX	Sequenced Packet Exchange
SQL	Structured Query Language
TCP	Transmission Control Protocol
TELNET	Telnet Virtual Terminal Protocol
TTS	Trouble Ticket System
VLAN	Virtual LAN
WAN	Wide Area Network
WWW	World Wide Web

Literaturverzeichnis

[3COM96] 3Com, (1996). The Virtual LAN Technology Report. http://www.3com.com/0files/strategy/537VLAN.thml, 3Com Corpertation.

[MBAU96] Bauch, M. (SS 1996) Ermittlung von betriebswirtschaftlichen Erfolgsfaktoren und Kennzahlen für ein Managementinformationssystem. Diplomarbeit. FH-München, Fachbereich Feinwerk- und Mikrotechnik, Produktion und Automatisierung.

[BSI396] Bundesamt für Sicherheit in der Informatik. (März 1996) Sicherheitsanforderungen an Internet-Firewalls.

[CSCG93] Cabletron Systems, Inc. (June 93) Concepts Guide. 9030647E3. P.O. Box 5005. Rochester. NH 03867-0505.

[CSSU93] Cabletron Systems, Inc. (August 1993) System User's Guide. 9030263-02E11. P.O. Box 5005. Rochester. NH 03867-0505.

[CSSA93] Cabletron Systems, Inc. (August 1993) System Administrator's Guide. 9030264-01E10. P.O. Box 5005. Rochester. NH 03867-0505.

[CRZI91] Carter, E., Dia, J. Evaluating Network Management Systems: Criteria and Observations. Krishnan, I., Zimmer, W. (1991) Integrated Network Management II. North-Holland Punlications. Amsterdam. New York. p. 213-223.

[RFC1451] Case, J., McCloghrie, K., Rose, M. T., (April 1993) Manager-of-Manager Management Information Base. (RFC 1451) Internet Activies Board.

[LCPW89] Cassel, L., Patridge, G., Westcott, J. Network Management Architectures and Protocols: Problems and Approaches. IEEE JSAC, Volume 7, Nr. 7, Sept. 89.

[CPST95] Chiotis, T., Maglaris, B., Stamatelopoulos, F. A Scaleable, Pltform-Based Architecture for Multiple Domain Network Management. Accepted for presentation at the IEEE International Conference on Communications '95, Juni 95

[COLE96] Conroy, K. F., Leinwand, A. (1996). Network Management, A Practical Perspective. (2. Aufl.). Massachusetts: Addison-Wesley Publishing Company.

[COOK97] Cook, R. (01.1997). Picking the right network management platform. http://www.eu.sun.com/sunworldonline/swol-01-netmanagement.html.

[DATA95] Datapro (10.1995) Network Management, Network Management Systems. Delran NJ 08075. The McGraw-Hill Companies.

[DEML95] Deml, S. (15.05.1995) Modellierung der verteilten Anwendung SAP R/3 für ein integriertes Management. Diplomarbeit. TU-München. Institut für Informatik.

[GMD800] Eisenblätter, U., Hommes, F., Pless, E. McCloghrie, K. Rose, M. T., (Oktober 1993) Ein Vergleich von kommerziellen Netzwerk-Management-Systemen. Arbeitspapiere der GMD 800, Gesellschaft für Mathematik und Datenverarbeitung mbH.

[GARB91] Garbe, K. (1991) Management von Rechnernetzen. Teubner Verlag.

[GOYE93] Goldszmidt, G., Yemini, Y. (1993) Evaluation Management Decisions via Delegation. In *The Third International Symposium on Integrade Network Management*, San Francisco, April 1993

[GUHE95] Gutschmidt, M., Hegering, H.-G., Neumair, B. (1995) Cooperative Computing und integriertes Systemmanagement. Springer-Verlag. Informatik-Spektrum.

[HEAB93] Hegering, H.-G., Abeck, S. (1993). Integriertes Netz- und Systemmanagement. (1. Aufl.). Bonn: Addison-Wesley.

[HEGE93] Hegering, H.-G., Yemini, Y. (1993), Intergrated network management, III. Amersterdam: North Holland.

[HPUN95] Hewlett Packard Co. (10.1995) HP OpenView, Using Network Node Manager. (1. Aufl.) HP Part No. J1169-90002.

[HPIG95] Hewlett Packard Co. (10.1995) HP OpenView, Network Node Manager Products, Installation Guide. (1. Aufl.) HP Part No. J1169-90000.

[INVA93] IBM (1993). NetView/6000 and NetView for AIX. http://www.raleigh.ibm.com/nv6/nv6over.html.

[IALD94] IBM (1994, 13, September). Announcement Letters (US) - Document '294-545'. http://www.ibmlink.ibm.com.

[ISO107462] International Organization for Standardization (September 1995), Open Distributed Processing - Reference Model - Part 2: Foundations. (ISO/IEC DIS 10746-2).

[JAHE92] James, H. (Juni 1992) Distributed Network Management. Data Communications International. McGraw-Hill.

[JAMA91] Jander, M. (Nov. 1991) Users Rate SNMP Multivendor Network Management System. Data Communications International, McGraw-Hill.

[RFC1103] Katz, D. (Juni 1989) Proposed Standard for the Transmission of IP Datagrams over FDDI Networks. (RFC 1103) Internet Activities Board.

[KAUF89] Kauffels, F. J. (1989), Network Management Series. Bergheim: DATACOM Buchverlag GmbH.

[KAUF92] Kauffels, F. J. (1992), Netzwerk-Management: Probleme, Standard, Strategien. (2. Aufl.) DATACOM Buchverlag GmbH.

[LIPI95] Lipinski, K. (1995), Lexikon der Datenkommunikation (2. Aufl.), Bergheim: DATACOM Buchverlag GmbH.

[RFC1213] McCloghrie, K., Rose, M. T., (März 1991) Management Information Base for network management of TCP/IP-based internets: MIB-II. (RFC 1213) Internet Activities Board.

[ISO74984] Open Systems Interconnection (1989) Basic Reference Model Part: 4: Management Framework. (International Standard ISO 7498-4) International Organization for Standardization.

[ISO74982] Open Systems Interconnection (1988) Basic Reference Model - Part 2: Security Architecture. (International Standard ISO 7498-2) International Organization for Standardization.

[GAGR96] Paquet, R. (24.06.1996) Enterprise Network Management. (Research Note, Markets, M-NMF-280) Gartner Group.

[R3AR93] SAP AG (1993), Architektur System R/3. Walldorf: SAP AG.

[CCMS96] SAP AG (1996), Computing Center Management System: BC120. Walldorf: SAP AG.

[R3AD93] SAP AG (1993), R/3-Administration: BC20. Walldorf: SAP AG.

[SAPS94] SAP AG (1994), R/3-Systemverwaltung. Walldorf: SAP AG.

[R3SY93] SAP AG (1993), R/3-Systemarchitektur: BC10. Walldorf: SAP AG.

[TERA92] Terplan, K. (1992), Communication networks management (2. Aufl.) Prentice Hall.

[TERA93] Terplan, K. (1992), Effective Management of Local Area Networks: Functions, instruments, and people. (1. Aufl.) Mc Graw-Hill.

[TERA94] Terplan, K. (1993), Netzmanagementanwendungen als Praxisanforderung. DATACOM Buchverlag GmbH.

[VALT91] Valta, R. (25.09.1991), Netzmanagementstationen - Vergleichende Darstellung. München: Leibniz-Rechenzentrum.

[RFC1271] Waldbusser, S. (November 1991) Remote Network Monitoring Management Information Base (Ethernet). (RFC 1271) Internet Activities Board.