



„Entwurf eines verteilten Bild-Archivs mit ATM Netzwerk  
Backbone am Beispiel von Planetendaten“

## **Dissertation**

zum Erlangen des akademischen Grades  
*Doktor der technischen Wissenschaften*

**an der  
Technischen Universität Graz**

vorgelegt von

Dipl. Ing. Herwig Rehatschek

**Betreuer:** o. Univ. Prof. Dipl. Ing. Dr. Franz Leberl

Graz, im Juni 1997



# Inhaltsverzeichnis

<b>1. KAPITEL: PROBLEMDEFINITION UND ZIELSETZUNG</b>	<b>5</b>
1.1 Hintergrund	5
1.2 Relevanz	6
1.3 Zielsetzung der Arbeit	7
1.4 Gliederung	8
1.5 Publikationen	9
1.6 Literatur und Referenzen	10
<b>2. KAPITEL: ÜBER BESTEHENDE BILDARCHIVE UND ATM NETZWERKE</b>	<b>11</b>
2.1 Geo-Daten bezogene Systeme	12
2.1.1 Spacepicture	12
2.1.2 GISIS	15
2.1.3 Das Arno-Projekt: „A distributed multimedia information system designed for the Arno Project“	18
2.1.4 VISTA	21
2.1.5 CEO / EWSE	25
2.1.6 NASA's EOSDIS	29
2.1.7 CORE SW ImageNet	35
2.1.8 AVHRR CD-Browser	37
2.1.9 Gesamtwertung und Gegenüberstellung	40
2.2 Planetarische Systeme	42
2.2.1 NASA/JPL PDS; IPA	42
2.2.2 NASA/JPL PDS; PIA	44
2.2.3 Zusammenfassung und Wertung	44
2.3 ATM Pilot-Netzwerke	46
2.3.1 Canarie Netz	46
2.3.2 Magic Net	49
2.3.3 Bay Area Gigabit Netzwerk	52
2.3.4 Europäischer ATM Pilotversuch	55
2.3.5 Zusammenfassung und Diskussion	59
2.4 Literatur und Referenzen	60
<b>3. KAPITEL: GDSS SYSTEMENTWURF</b>	<b>63</b>
3.1 Funktionalität	63
3.1.1 Rasterorientiertes Browsinginterface	65
3.1.2 Suche nach Suchpunkt und Suchregion	66
3.1.3 Überdeckungsabfragen	66
3.1.4 Suche nach Metainformationen	67

3.1.5	Client-Server Architektur	67
3.1.6	Unterstützung von existierenden internationalen Datenaustausch-Standards	68
3.1.7	Intelligentes lokales Caching	71
3.1.8	Batchjob Abarbeitung	71
3.1.9	Datenvorhersage-Algorithmus des lokalen Server	71
3.1.10	Usergruppenverwaltung	72
3.1.11	Netzwerksicherheit / Verrechnungsstrategien	72
3.1.12	Netzwerkmanagement	73
3.1.13	Remote Data Processing	74
<b>3.2</b>	<b>GDSS Architektur</b>	<b>75</b>
3.2.1	Gesamtdesign	75
3.2.2	Spezifikation des zentralen Servers	78
3.2.3	Spezifikation des Lokalen Servers	80
3.2.4	Spezifikation der Archive	82
<b>3.3</b>	<b>GDSS System Design</b>	<b>83</b>
3.3.1	Design des zentralen Servers	83
3.3.2	Design des lokalen Servers	88
<b>3.4</b>	<b>Österreichweites GDSS Szenario mit mehreren Servern</b>	<b>94</b>
<b>3.5</b>	<b>Anwendbarkeit des GDSS Konzeptes auf MISSION</b>	<b>96</b>
<b>3.6</b>	<b>Zusammenfassung und Wertung</b>	<b>96</b>
<b>3.7</b>	<b>Literatur und Referenzen</b>	<b>98</b>
<b>4.</b>	<b>KAPITEL: ATM NETZWERK BACKBONE</b>	<b>101</b>
<b>4.1</b>	<b>Netzwerk Anforderungsanalyse</b>	<b>101</b>
4.1.1	Anforderungsprofil	102
4.1.2	Allgemeine Auswahlkriterien	107
4.1.3	Finanzielle Auswahlkriterien	108
4.1.4	Technische Auswahlkriterien	109
4.1.5	Netzwerkmanagement	109
4.1.6	Internet und GDSS	110
4.1.7	Schlußfolgerungen und Konsequenzen	111
<b>4.2</b>	<b>Asynchronous Transfer Mode (ATM)</b>	<b>112</b>
4.2.1	Technischer Überblick	113
4.2.2	Migration zu ATM am Bsp. der TUG	114
4.2.3	Aufbau des TUG OC-3 Pilotnetzwerkes	119
4.2.4	Kostenverrechnung in ATM Netzwerken	125
4.2.5	Wirtschaftliche Aspekte	128
4.2.6	Zusammenfassung und Bewertung	128
<b>4.3</b>	<b>Alternativen zu ATM</b>	<b>129</b>
<b>4.4</b>	<b>Begründung der Wahl von ATM</b>	<b>132</b>
<b>4.5</b>	<b>Literatur und Referenzen</b>	<b>133</b>

---

<b>5. KAPITEL: GDSS IMPLEMENTIERUNG</b>	<b>135</b>
<b>5.1 Datenbankmodell</b>	<b>136</b>
5.1.1 Wahl des Datenbankmodells	136
5.1.2 Details zur Magellan FBIDR Aufzeichnung	140
5.1.3 Datenbankschema für die Magellan FBIDR Metadaten	141
5.1.4 Implementationsdetails	146
5.1.5 Inbetriebnahme des Oracle Enterprise Servers	150
5.1.6 Einbringung von BLOBs in die Oracle Datenbank	152
<b>5.2 Java Programme und Applets</b>	<b>153</b>
5.2.1 Java Applets	154
5.2.2 Sicherheitsaspekte	155
5.2.3 Kritische Bewertung	156
<b>5.3 Client-Server Kommunikation unter Verwendung von JDBC</b>	<b>157</b>
5.3.1 Datenbankkommunikation eines offenen Systems	157
5.3.2 Implementierung einer multitier Lösung	161
5.3.3 Alternativen und zukünftige Lösungen	162
5.3.4 Kritische Bewertung und Erfahrungen	163
<b>5.4 FBIDR Query Tool - Ein GDSS Klienten Prototyp</b>	<b>163</b>
5.4.1 Programmstruktur	164
5.4.2 Bedienung und Funktionalität	169
5.4.3 Versuchsszenario	170
5.4.4 Probleme und Erfahrungen	171
<b>5.5 Zusammenfassung und Schlußfolgerungen</b>	<b>173</b>
<b>5.6 Literatur und Referenzen</b>	<b>175</b>
<b>6. KAPITEL: ERGEBNISSE</b>	<b>177</b>
<b>6.1 Aufbau des ATM Netzwerk Backbones</b>	<b>177</b>
<b>6.2 Performance Messungen am ATM Netzwerk</b>	<b>178</b>
6.2.1 Overhead der physischen Schicht	179
6.2.2 Overhead der ATM Schicht	179
6.2.3 Overhead der AAL Schicht	180
6.2.4 Meßergebnisse	181
6.2.5 Bewertung der Ergebnisse	188
<b>6.3 Video Konferenz ATM Pilotversuch mit dem VCPC</b>	<b>190</b>
6.3.1 Versuchsablauf	190
6.3.2 Bewertung des Ergebnisses	193
<b>6.4 ATM Pilotversuch mit dem VCPC: Application Sharing</b>	<b>194</b>
6.4.1 Versuchsablauf	194
6.4.2 Bewertung des Ergebnisses	195

---

6.5	GDSS Prototyp zum Abfragen von FBIDR	195
6.6	Literatur und Referenzen	203
<b>7. KAPITEL: ZUSAMMENFASSUNG UND ZUKÜNFTIGE ARBEIT</b>		<b>205</b>
7.1	Zusammenfassung	205
7.2	Bewertung des Systems	209
7.3	Schlußfolgerungen	211
7.3.1	Schlußfolgerungen aus dem Systementwurf	212
7.3.2	Schlußfolgerungen aus dem Netzwerk Backbone	213
7.3.3	Schlußfolgerungen aus dem GDSS Prototypen, Java, JDBC und DBMS	215
7.4	Mögliche Verbesserungen und zukünftige Arbeit	217
7.5	Literatur und Referenzen	219
<b>ANHANG A : NETZWERK</b>		<b>221</b>
<b>Einführung ins Netzwerkmanagement</b>		<b>221</b>
	Internet Management Protokolle SNMP und CMIP	222
	Management Architektur	226
	Management Funktionalität	227
	Schritte zur Anschaffung eines Netzwerkmanagement Systems	231
	Kommerzielle Netzwerkmanagement Systeme	235
<b>Internet Grundlagen</b>		<b>235</b>
	Historisches	236
	Internetprotokolle	237
	Die Vernetzung von Österreich und Europa	238
	Internetstatistiken	242
<b>Technische Grundlagen von ATM</b>		<b>246</b>
	Das Protokoll Referenz Modell	246
	ATM Dienste	251
<b>Wirtschaftliche Aspekte von ATM</b>		<b>253</b>
	Zukünftige Entwicklungen und Trends	258
<b>Literatur und Referenzen zum Anhang A</b>		<b>261</b>
<b>ANHANG B : DATENBANK</b>		<b>265</b>
	Erfahrungen und Probleme bei der Installation vom Oracle Enterprise Server	265
<b>ANHANG C : JAVA</b>		<b>267</b>
<b>Historisches</b>		<b>267</b>

<b>Literatur und Referenzen zum Anhang B</b>	<b>268</b>
<b>ANHANG D : AKRONYME</b>	<b>269</b>



## Vorwort

*„Till we find our place,  
on the path unwinding ...“*

Tim Rice,  
from „The Lion King“

Im Oktober 1994 trat ich in das Institut für Computergraphik (ICG)<sup>1</sup> von Prof. Leberl ein, mit dem Plan, ein Dokorratsstudium zu absolvieren. Meine Hauptinteressen lagen im Bereich Netzwerke, Internet und Datenbanken. Da das Institut im September 1994 den 512 GB Bilddaten umfassenden Magellan Datensatz erhalten hatte, war mein Dissertations Projekt sehr schnell definiert: es wurde dringend Konzept und ein Prototyp benötigt, diese riesige Datenmenge zu verwalten und einer größeren Gemeinschaft von Wissenschaftlern zugänglich zu machen. Für die Verwaltung der Daten benötigt man ein Systemkonzept, für den Transport eine flexible und hohe Bandbreite liefernde Netzwerktechnologie, für die Speicherung ein geeignetes Datenmodell und eine Datenbank und für den Zugriff benötigt man ein Benutzerinterface welches leicht zu verstehen und möglichst plattformunabhängig ist. Dies waren die Anfangsgründe des vorliegenden Dissertationsprojektes, welches sich über 2½ Jahre erstreckte und an dessen Ende ein funktionierender Prototyp steht.

Graz am 27. April 1997

---

<sup>1</sup> ab Winter 1996: „Institut für Maschinelles Sehen und Darstellen“

## Danksagung

Dieses Projekt wurde durch das Forschungsprogramm „Theory and Applications of Digital Image Processing and Pattern Recognition“ Projekt Nr. 1 / Task 4 (Parallel Processing Strategies For Large Image Data Sets) des Österreichischen „Fond zur Förderung wissenschaftlicher Forschung“ (FWF) finanziell unterstützt.

Ich möchte an dieser Stelle allen jenen Personen von ganzem Herzen danken, die diese Dissertation ermöglicht haben. An erster Stelle meinen Eltern und meiner Freundin Vanessa Keitel, die mir das ganze Projekt hindurch mit ihrer Liebe und seelischen Unterstützung stets eine sichere Basis und Rückhalt gewährt haben. An zweiter Stelle danke ich Prof. Leberl für seine Geduld, Hilfe und sein Engagement. Ohne sein Mitwirken wäre dieses Projekt niemals zustande gekommen. Ebenfalls ein ganz besonderes Dankeschön gilt meinem Gruppenleiter Rainer Kalliany, der die Gruppe die gesamte Zeit über gemanagt und mir zahlreiche Reisen zu interessanten Kongressen und Vorträgen ermöglicht hat. An dieser Stelle auch ein warmes Dankeschön an meine Freundin Vanessa, die mich stets durch ihre Liebe und Geduld sowie dem Korrekturlesen zahlreicher Publikationen tatkräftig unterstützte.

Last but not least möchte ich noch meinem Kollegen und Partner Wolfgang Walcher ein herzliches Danke sagen - seine Dissertation ist eine Zwillingsarbeit zu der vorliegenden, da wir uns das Gesamtprojekt GDSS in zwei Teile aufgetrennt haben. Damit beeinflusst seine Arbeit die meinige maßgeblich. Weiters war sein großes Wissen der Bildverarbeitung stets eine große Hilfe.

Graz am 27. April 1997

## Zusammenfassung

Zahlreiche gerade stattfindende und auch bereits erfolgreich abgeschlossene planetarische und Geo-Satellitenmissionen haben die Entstehung von großen Bilddatenarchiven bewirkt, welche auf der gesamten Welt verstreut sind. Die Suche nach spezifischen Daten gestaltet sich derzeit als sehr schwierig, da kaum internationale Abfrage und Katalog Standards verwendet werden, langsame Netzwerkverbindungen das Laden von großen Bilddaten unterbinden, keine einheitlichen graphischen Benutzerinterfaces verwendet werden und plattformunabhängige Klient Software nicht existiert. Das in dieser Arbeit beschriebene GDSS (Graz Distributed Server System) zeigt in drei Schritten den System Entwurf eines globalen Fernerkundungs-Datenarchivs, den Aufbau einer geeigneten Netzwerk Infrastruktur sowie die Implementierung eines Prototypen. GDSS benutzt als Netzwerk Backbone die ATM (Asynchronous Transfer Mode) Hochgeschwindigkeits-Technologie. Der Prototyp benutzt die plattformunabhängige Sprache Java sowie das JDBC (Java Data Base Connectivity) Modul um mit Hilfe von ANSI SQL auf jede ANSI SQL kompatible relationale Datenbank mit demselben Kode und Benutzerinterface zugreifen zu können. Der Prototyp ermöglicht den Fernzugriff auf den NASA Magellan Datensatz der Venus, der als erster Testdatensatz des GDSS fungierte.

## Abstract

On-going and completed planetary and Earth satellite missions have spawned large image data archives distributed all over the world. The searching for specific data suffers from lack of using international query and catalogue standards, slow network connections, no graphical user interface (GUI) and no platform independent client software. The proposed GDSS (Graz Distributed Server System) project outlines the system design, the setup of a high-speed ATM (Asynchronous Transfer Mode) backbone and the prototype implementation. The GDSS prototype uses Java and JDBC (Java Data Base Connectivity) in order to access any ANSI SQL compatible relational database system with the same set of SQL statements. The client software provides remote access to search and retrieve data from a planetary meta-data set. The 500 GByte NASA Magellan data set from planet Venus was used as a first test dataset. However, the system is not restricted to the given dataset. We expect data management concepts, network technology, Java client software and database connectivity based on international standards illustrated within the GDSS project to also be applicable to Earth based remote sensing data.



# 1. Kapitel

## Problemdefinition und Zielsetzung

### 1.1 Hintergrund

Anlaß für diese Arbeit war der Bedarf an Bearbeitungs- und Managementkonzepten für Fernerkundungsdaten, insbesondere den Datensatz des Satelliten Magellan, der am 12. Oktober 1994 eine mehr als fünf Jahre dauernde Mission beendet hat. Am 9. September 1994 wurde zwischen der NASA und dem Institut für maschinelles Sehen und Darstellen (ICG) der TU-Graz ein Kontrakt abgeschlossen, der das ICG zum kostenlosen Erhalt des gesamten Datensatzes ermächtigte. Im Gegenzug verpflichtet sich das ICG, diese Daten und Fachwissen über die Daten allen wissenschaftlichen europäischen Institutionen bei Bedarf anzubieten, sowie entwickelte Software und Datenmanagement-Konzepte der NASA zur Verfügung zu stellen.

Die Verwaltung großer Datenmengen in der Fernerkundung ist derzeit nicht zufriedenstellend gelöst. Die zunehmende Datenflut erfordert immer größere (und teurere) Speichermedien sowie äußerst leistungsfähige Hardware, die komplizierte Berechnungen in vernünftiger Zeit durchführen kann. Als besonders schwierig erweist es sich in der Fernerkundung Bilder nach geographischen Kriterien zu suchen und in Originalauflösung zu bekommen. Meist sind die Bilder in Katalogen nach Überflugzeit und Zyklus sortiert. Sucht man zu einem bestimmten Gebiet (region/point of interest) Bilder, oder hat gar weitere Suchkriterien (etwa: alle Bilder von einem bestimmten Gebiet der Venus zwischen dem 1. und 6. Jänner 94), muß man mühevoll manuell suchen. Dies wird durch die relativ kleinen und unübersichtlichen Quicklooks, wie sie in Katalogen zu finden sind, noch zusätzlich erschwert. Die Washington University St. Louis bietet zu diesem Zwecke zwar großformatige Poster an, es ist jedoch sofort einsichtig, daß diese „Lösung“ alles andere zufriedenstellend ist (alleine schon wegen der hohen Druckkosten).

Das gewünschte Bildmaterial muß dann in Folge meist auf dem Postwege bestellt werden (auf CD-ROM oder Band), was ebenfalls einen langwierigen Prozeß darstellt. So mußten wir für die Magellan Daten etliche Wochen in Kauf nehmen, ein Zeitraum, der in der Praxis inakzeptabel ist, vor allem im Zeitalter der globalen Vernetzung.

Das zweite sich sofort aufwerfende Problem ist der Mangel an verfügbaren Abfragestandards. Es existieren mittlerweile Archive auf der ganzen Welt, die sowohl planetarische als auch terrestrische Bilddaten speichern. Ein Benutzer muß jedoch, um nach Daten zu suchen, erstens die genaue Position/Adresse des Archivs kennen und zweitens auch die Abfragesprache erlernen. Da es keinen genormten Zugang auf Archive gibt, ist es sehr wahrscheinlich, daß jedes einzelne Archiv einen

unterschiedlichen und daher neu zu erlernenden Abfragemechanismus aufweist.

Der Entwurf eines globalen Systemkonzeptes für Fernerkundungsdaten sowie der dringende Bedarf einer Standardisierung der Abfrage ist somit vollkommen einsichtig und braucht nicht mehr weiter argumentiert zu werden.

## 1.2 Relevanz

Heute existiert bereits eine große Anzahl von Datenarchiven, so etwa EROS (Earth Resource Observation Systems) Datenzentrum, ImageNet, NASA PDS (Planetary Data System) - es besteht aber noch immer kein einheitlicher Zugriff auf die dort angebotenen Datenmengen. GDSS (Graz Distributed Server System) soll vorhandene „Bilddateninseln“ mit einem Netzwerk zusammenfassen und den Benutzern einen einheitlichen, plattformunabhängigen Zugang zu Fernerkundungsdaten ermöglichen

Als Testdaten sollen zunächst die Bilddaten des Satelliten Magellan vom Planet Venus verwendet werden, grundsätzlich sollten die vorgestellten Konzepte und Ideen jedoch auch auf andere Daten anwendbar sein. Das Projekt soll einen Beitrag zum PDS (Planetary Data System) der NASA liefern, dem das Institut für Maschinelles Sehen und Darstellen seit September 1994 offiziell als der „European Magellan Data Node“ angehört.

Dieses Projekt kann aber auch als Beitrag zum EU Projekt CEO (Center for Earth Observation), für das international Projekt „Mission to Planet Earth“ (US) sowie für das österreichische Projekt MISSION (Multi-Image Synergistic Satellite Information for the Observation of Nature) gesehen werden. In allen drei Projekten werden Methoden zur intelligenten Verwaltung von Giga oder sogar Tera Byte an Daten sowie die parallele Suche und Bestellung von hoher Wichtigkeit sein. Vor allem im Netzwerkbereich und mit dem funktionstüchtigen Prototypen kann GDSS wertvolle Bausteine für diese Projekte liefern.

GDSS läßt sich prinzipiell auch für terrestrische Daten verwenden, ein mögliches österreichisches Szenario wäre eine Verteilung der Bezirksamt Bilddaten auf die Bundesländer mit gleichzeitiger Berücksichtigung einer leistungsfähigen Vernetzung. Hier ist ebenfalls ein flexibles Konzept nötig, wie man mit möglichst billiger lokaler Hardware trotzdem in vernünftiger Zeit österreichweit Bilddaten suchen und in Originalauflösung zur Verfügung stellen kann. Ein mögliches Szenario wird in dieser Arbeit dargestellt.

Alle Szenarien haben das gemeinsame Ziel, Bilddaten möglichst verteilt und nicht doppelt zu halten. Damit wird nicht nur ein hoher Speicherbedarf reduziert, sondern vor allem die Hoheit über gewisse Daten gewahrt. Die Verantwortung für Langzeitverwaltung liegt an einer zentralen Stelle, die Anwender sind verteilt. Dies entspricht einem föderativen Datenmodell, d.h. es muß in geeigneter Weise gesorgt werden, daß die am häufigsten benötigten Daten lokal vorhanden sind. Sollten

dennoch andere Daten benötigt werden, muß das aber auch in vernünftiger Zeit möglich sein (suchen und holen).

Voraussetzung für die Funktion eines solchen Systems sind:

- ein geeignetes Systemkonzept,
- eine leistungsfähige Netzwerktechnologie,
- eine wohlorganisierte Datenbank.

Damit besteht die Notwendigkeit, ein geeignetes System zu entwerfen. Dies ist Gegenstand dieser Arbeit.

### 1.3 Zielsetzung der Arbeit

*Das Zusteuern auf ein Ziel,  
gleicht dem Wandern auf einem endlosen Weg.  
Sicher, es gibt Meilensteine - aber das Ende  
des Weges erreicht der Mensch mit seiner  
stetigen Unzufriedenheit niemals.*

(H.R.)

Das System soll als verteilte Client/Server Applikation entworfen werden, die den Namen GDSS (Graz Distributed Server System) trägt. Die prinzipielle Architektur des Systems, der Netzwerkbackbone sowie die Implementierung eines funktionstüchtigen Prototypen werden in dieser Arbeit behandelt. Die Ergebnisse stellen eine Art „Skelett“ dar, dem „Fleisch“ später durch Projekte (an einer Universität durch Diplomarbeiten, EDV-Projekte, usw.) zugefügt werden kann. Entscheidend ist ein objektorientierter Entwurf. Als Vorbild kann Hyper-G Kappe, 95] gelten, welches ebenfalls von einer Person konzipiert, über mehrere Jahre hindurch von einem Team an der TU-Graz entwickelt wurde und jetzt kommerziell genutzt wird.

Drei Ziele dieses Projektes sind:

- ◆ Erstellung eines theoretischen Konzeptes zur Verwaltung großer Bildmengen aus der Fernerkundung
- ◆ Begründung der gewählten Netzwerktechnologie sowie Aufbau eines Netzwerkbackbones in Verbindung mit Pilotversuchen
- ◆ Erstellung eines funktionierenden GDSS Prototypen

Demzufolge beinhaltet die Dissertation auch drei Hauptabschnitte, die in den Kapitel 3,4 und 5 behandelt werden.

Es gilt ein Konzept zur Vereinigung von vorhandenen „Bilddateninseln“ via einem flexiblen Netzwerk zu entwickeln, welches man mit einem leistungsfähigen verteilten Server System der Öffentlichkeit zugänglich machen kann. Die

Hauptaugenmerke sollen auf einem einheitlichen Benutzerinterface, Plattformunabhängigkeit, Verwendung von internationalen Standards sowie einem flexiblen Netzwerk liegen. Originaldaten sollen in den Archiven verweilen, Benutzer sollten über einen Gesamtindex auf die Archive in geeigneter Form zugreifen können.

ATM soll als geeignete Netzwerktechnologie für das GDSS System hervorgehen. Es gilt einen Vergleich mit Alternativen zu prüfen. Die Wahl von ATM muß sowohl durch einen praktischen Versuchsaufbau und entsprechende Pilotversuche als auch durch wirtschaftliche Abwägungen untermauert werden.

Schließlich soll ein funktionstüchtiger GDSS Prototyp entwickelt werden. Dieser muß einerseits in das entwickelte Konzept passen, andererseits mit der gewählten ATM Technologie verträglich sein. Weiters müssen die Forderungen auf Erfüllung eines internationalen Abfragestandards, auf ein einheitliches graphisches Benutzerinterface und auf Plattformunabhängigkeit erfüllt werden. Eine geeignete Programmiersprache ist zu wählen bzw. ein Datenmodell ist zu erstellen. Als Testdaten sind die Metadaten der Magellan FBIDR (Full Resolution Basic Image Data Records) zu verwenden. Das zu erstellende Programm sollte zu einer Interessensregion die zugehörigen FBIDRs gruppiert nach Aufnahmezyklen liefern.

## 1.4 Gliederung

Im 1. Kapitel erfolgte die Motivation dieses Projektes. Die Problemstellung an sich, Hintergrund und Relevanz wurden erläutert. Weiters wird dargelegt, welche Ergebnisse von dieser Arbeit erwartet werden bzw. welchen primären Zielsetzungen sie dient.

Der erste Baustein ist die Aufarbeitung des Wissensstandes, der im 1. Projektjahr berücksichtigt wurde. Im 2. Kapitel erfolgt eine umfangreiche Analyse vorhandener Fernerkundungs-Systeme und ATM Netzwerke und deren Funktionalität. Jeder Abschnitt endet mit einer Bewertung, das Kapitel wird mit einer Gesamtbewertung und Gegenüberstellung der vorhandenen Systeme abgeschlossen.

Auf diesem Wissen aufsetzend wird im 3. Kapitel eine Anwender-orientierte Definition des Systems gegeben. Dies geschieht in drei Unterkapitel: im ersten Teil wird die Frage „Was soll das System leisten?“ beantwortet, im zweiten Teil wird ein die Anforderungsanalyse erfüllendes Gesamtsystemkonzept vorgeschlagen, die Funktionalität der Hauptkomponenten wird erläutert. Im dritten Teil wird das Design ein wenig verfeinert, indem die Komponenten in einzelne Module aufgeteilt und diese beschrieben werden. Hier wird das „Wie?“ angedeutet. Diesem Konzept, welches so allgemein wie möglich gehalten ist, wird in den folgenden Kapiteln (4-6) in Top Down Manier schrittweise Leben eingehaucht. Das Ergebnis ist ein funktionstüchtiger GDSS Prototyp.

Die erste Komponente des Systems, die zugrunde liegende Netzwerkstruktur, wird

im 4. Kapitel behandelt. Das Kapitel ist in vier Unterkapitel aufgeteilt, wobei im ersten Abschnitt über Anforderungen an das Netzwerk, über allgemeine, finanzielle bis hin zu technischen Anforderungen berichtet wird. Der nächste Abschnitt beschäftigt sich mit der für das GDSS vorgeschlagenen Breitband Netzwerktechnologie ATM. Basierend auf der ATM Technologie wird nun einerseits die Migration des TUG Netzes andererseits der Aufbau des ATM Testnetzwerkes zwischen dem Institut für maschinelles Sehen und Darstellen (ICG) und dem Institut für Allgemeine Informations- und Kommunikationstechnologie (IAIK) beschrieben, um die Migrationsmöglichkeiten zu ATM an einem praktischen Beispiel zu demonstrieren. Den Abschluß des Kapitels bilden eine Gegenüberstellung von ATM und Alternativtechnologien. (konkret wird ATM mit Frame Relay, FDDI/FDDI II, DQDB, Fibre Channel und SMDS verglichen) sowie eine Begründung der Wahl von ATM.

Im 5. Kapitel wird der für dieses Projekt entwickelte, auf JAVA basierende GDSS Prototyp vorgestellt. Zu Beginn erfolgt eine Vorstellung des Schemas und eine Beschreibung des Datenbankmodells, welches mit Oracle realisiert wurde. Dem folgt eine Beschreibung der Magellan Metadaten. Im zweiten Abschnitt wird der JAVA Klient beschrieben, wobei in einer kurzen Übersicht auf die Programmiersprache JAVA eingegangen wird. Der dritte Teil beschäftigt sich mit der Klient/Server Kommunikation, mit der Verwendung von JDBC (Java Data Base Connectivity) sowie mit 2 und 3 Tier Lösungen. Der vierte und letzte Teil des Kapitels behandelt schließlich das Programm selbst und erklärt seine Funktionsweise.

Das 6. Kapitel faßt die erzielten Ergebnisse dieser Arbeit zusammen, wobei in fünf Unterabschnitten der Aufbau des TUG Clusters, die Performance Messungen am ATM Cluster, die beiden DIANE Pilotversuche sowie der GDSS Prototyp vorgestellt werden. Die Ergebnisse sind durch Erfahrungsberichte ergänzt.

Im 7. und letzten Kapitel werden Schlußfolgerungen, eine Zusammenfassung sowie zukünftige Arbeit dieses Projektes präsentiert.

Im Anhang befinden sich Grundlagen- und Hintergrundwissen über Netzwerke (vor allem ATM Technologie - Anhang A), Datenbanken (Anhang B) und Java (Anhang C) sowie ein Akronym Verzeichnis (Anhang D) mit kurzen Erläuterungen der wichtigsten in der Telekommunikation verwendeten Abkürzungen.

## 1.5 Publikationen

Im Laufe des Projektes entstanden 6 Publikationen, wovon ich bei drei Co-Autor, von drei Haupt-Autor war. Alle Publikationen sind zumindest als PS Version verfügbar - einige sogar als HTML und PS Version - und sind von [Rehat, 97] abrufbar. Die Referenzen der Publikationen lauten im einzelnen:

Walcher W., H. Rehatschek: "The Design of a Distributed Remote Sensing Data Base or the Magellan Data

Set from Planet Venus".IGARSS95, IEEE Proceedings catalog number 95CH35770, Vol II,pp.1610-1612, Firenze, Italy (1995).

Rehatschek H.: "Preliminary Design of a Distributed Planetary Image Data Archive Based on an ATM Network". Proc. of Visual Conference , 5-6 Feb. 1996, Melbourne, pp. 68-78.

Rehatschek H.: "A Concept for a Network-Based Distributed Image Data Archive". Proc. of ISPRS Conference , 9-19 July 1996, Vienna, Vol. 31, II, pp. 327-332.

Kofler M., Rehatschek H., Gruber M.: "A Database for a 3D GIS for Urban Environments Supporting Photo-Realistic Visualization". Proc. of ISPRS96 Conference, 9-19 July 1996, Vienna, Vienna, Vol. 31, II, pp. 198-202.

Rehatschek H. "Design of a Distributed Planetary Image Data Archive Based on an ATM Network". Submitted to Springer Verlag 'Visual Information Systems', Jan. 1997

Raffer P., Rehatschek H. "Die digitale Vernetzung Österreichs", Telematik Ingenieur Verband 4/96, Seiten 7-10

## 1.6 Literatur und Referenzen

[Kappe, 95] Kappe Frank, „A Scaleable Architecture for maintaining Referential Integrity in Distributed Information Systems“, JUCS, Vol.1 No.2, Feb. 28, 1995

[Rehat, 97] Rehatschek H., Homepage: Basis Informationen, Publikationen und Forschung, Jän. 97, <URL: <http://www.icg.tu-graz.ac.at/wigl>>

## 2. Kapitel

### Über bestehende Bildarchive und ATM Netzwerke

*Zum Lichte des Verstandes  
können wir immer gelangen;  
aber die Fülle des Herzens  
kann uns niemand geben.*  
(Johann Wolfgang von Goethe)

Dieses Kapitel beschreibt bereits existierende oder noch im Entwicklungsstadium befindliche Bildkatalog-Systeme und ATM Netzwerke. Die Stärken und Schwächen dieser Applikationen sowie deren grundsätzliche Funktion und Architektur sollen durchleuchtet werden. Ziel ist es, die Schwächen der vorgestellten Systeme zu erkennen und in GDSS zu eliminieren bzw. die Stärken zu kombinieren und in weiterer Folge noch zusätzliche, nicht existierende Merkmale einzubringen.

Das Kapitel ist in drei Abschnitte unterteilt, wobei sich der erste Teil mit terrestrischen Bildsystemen befaßt, der zweite mit planetarischen Systemen und im dritten Teil werden einige ATM-Pilotnetzwerke und deren Applikationen vorgestellt. Die Trennung zwischen planetarischen und Geo-Daten ist deswegen sehr wichtig, da der Suchvorgang für planetarische Daten anders organisiert sein muß als jener für Geo-Daten. Dies ergibt sich allein schon aus der Tatsache, daß es auf Planeten - im Vergleich zur Erde - nur sehr wenige benannte Merkmale gibt<sup>1</sup>. Weiters genießt man auf der Erde den Vorteil, daß etwa zwei Drittel der Oberfläche mit Wasser bedeckt sind und daher ein Drittel der Oberfläche im Detail, die anderen zwei Drittel aber anders dargestellt werden. Für einen Suchprozeß in terrestrischen Daten reicht oft eine vektororientierte Darstellung der Oberfläche aus. Für planetarische Daten existieren obige Vorteile nicht, eine Speicherplatz sparende vektororientierte Darstellung der Oberfläche kommt damit eigentlich nicht in Frage.

Für die Verwaltung von Erddaten existiert eine ganze Reihe von bereits voll funktionstüchtigen Applikationen. Sie weisen zum Teil sogar bereits dreidimensionale GUIs (Graphical User Interfaces) auf, aber auf Grund der oben erwähnten Tatsachen beruhen sie fast ausnahmslos auf vektororientierten Suchinterfaces. Für planetarische Daten gibt es nur sehr wenige bestehende Systeme, obwohl auch auf diesem Sektor gearbeitet wird, wie Projekte bei der NASA, der EU und der ESA beweisen.

Um die vorgestellten Systeme zu vergleichen, ist in den ersten beiden Unterkapitel an jede Systembeschreibung eine allgemeine Bewertung angeschlossen. Für die Bewertung selbst werden Punkte von 0 (unbefriedigend, nicht vorhanden) bis 3

---

<sup>1</sup> Auf der Erde kann man z.B. politische Grenzen, Städte, Flüsse, ... auf der Karte als Orientierungshilfe eintragen

(sehr gut, vorhanden) vergeben. Die Gesamtwertung ist die Summe der Punkte. Es werden z.B. beim User Interface Übersichtlichkeit, Funktionalität, Interaktivität und Bedienungskomfort extra bewertet. Die Gesamtwertung für einen Aspekt ergibt sich aus dem arithmetischen Mittel der Teilaspekte. Die maximal zu erreichende Punktezahl ist 39 (entspricht 13 Kriterien).

## 2.1 Geo-Daten bezogene Systeme

### 2.1.1 Spacepicture

Das System „Spacepicture“ wurde 1991 fertiggestellt und bei der Deutschen Luft und Raumfahrtforschung (DLR) in Oberpfaffenhofen, Deutschland installiert. Space Picture sollte innerhalb des ISIS Projektes (siehe nächstes Kapitel) die Visualisierung von hochauflösenden (6000 x 6000 Pixel) Satellitenbildern der Erdoberfläche ermöglichen.

Spacepicture basiert auf dem Hyper Picture Toolkit, das vom ZGDV (Zentrum für graphische Datenverarbeitung) in Darmstadt im Auftrag von Daimler-Benz für den Einsatz auf deren Optical Disk (OD) Jukeboxen entwickelt wurde. Das DFD (Deutsches Fernerkundungs Datenzentrum) hatte 1990 vom Daimler-Benz Software Systemhaus Debis eine solche Jukebox gekauft, und sich vom ZGDV die Software für den Zugriff auf ihre Satellitendaten modifizieren lassen, daher der Name Spacepicture.

Diese Entwicklung entstand im Rahmen des 1. GISIS Konzepts (siehe Kapitel GISIS), welches OD-Jukeboxen als Speichermedium, Spacepicture als Such und Visualisierungstool, und ein von der DLR KI-Gruppe geschriebenes User-Guide System zum Inhalt hatte. Dieses System wurde auf der ISY-Tagung in München 1992 vorgestellt. Der kommerzielle Erfolg scheiterte an den Produktkosten, obwohl die Kommerzialisierung durch Debis und ZGDV versucht wurde.

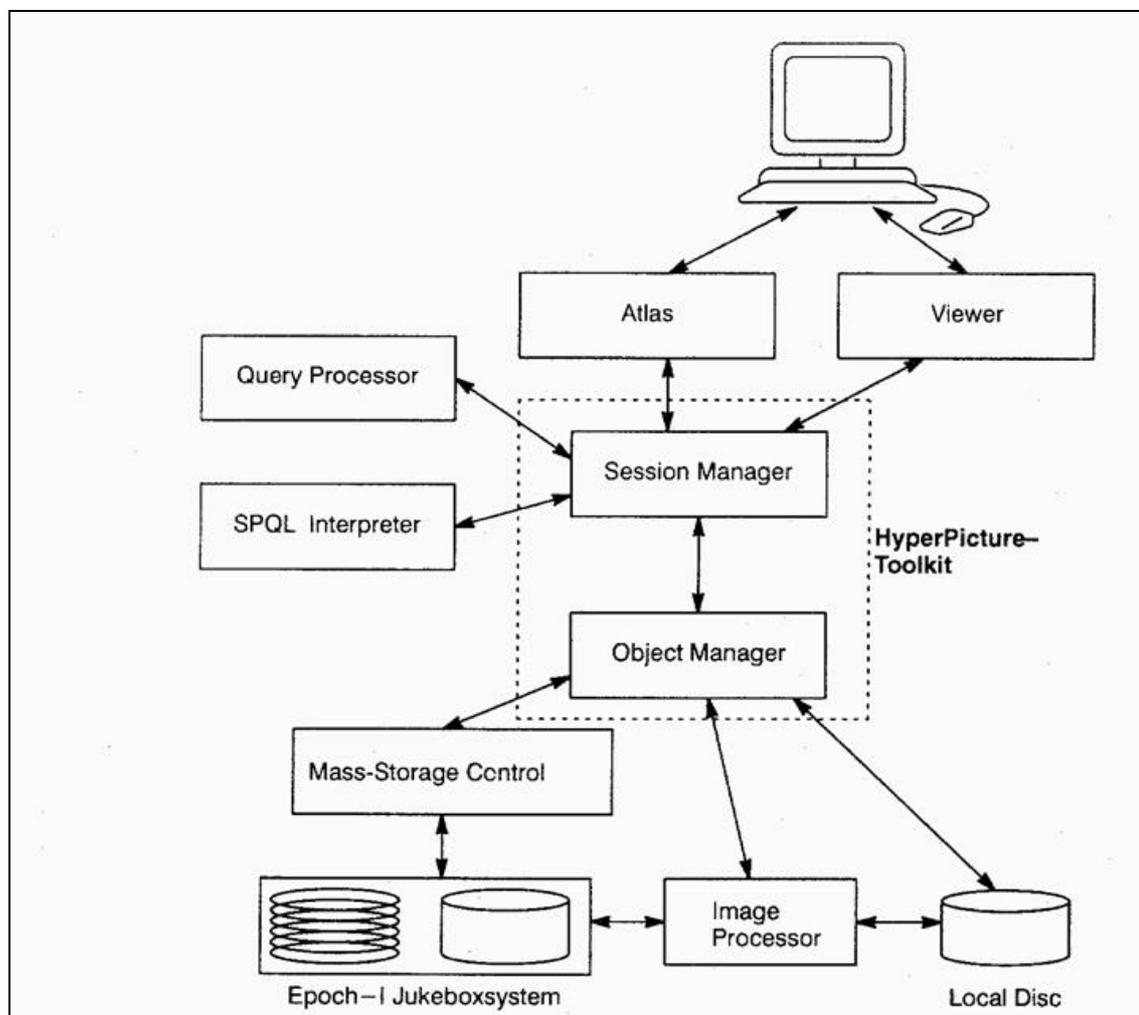
Daraufhin stellte DLR das Konzept um, und entwickelte mit den Erfahrungen und der Funktionalität von Spacepicture in eigener Regie ein Gesamtsystem. Damit entstand die jetzige graphische Schnittstelle von GISIS. Spacepicture wurde intern im DFD noch bis Mai 1996 für den Zugriff auf die geokodierten Landsat und SPOT Daten genutzt. Es wird aber nicht mehr gewartet. Sobald diese Daten auf das neue Robotarchiv migrieren, wird das System endgültig stillgelegt.

Spacepicture (siehe Abbildung 2.1 für die prinzipielle Architektur) ist ein historisch wichtiges System zur Archivierung und Organisation von großen Satelliten Bilddatenmengen. Es basiert auf einem hypermedialen Datenmodell und unterstützt besonders den interaktiven Suchprozeß mit Hilfe der hypermedial verlinkten Datenstruktur. Spacepicture verfügt über ein komfortables GUI, wobei die Suche sowohl durch Navigation in der Hyperlinkstruktur als auch durch geographische Abfragen erfolgen kann [Kirste, 93]. Die geographische Suche erfolgt nach einer

Interessensregion, wobei als Zusatzkriterien Datum und Sensortyp fungieren können.

Das zugrunde gelegte Datenbanksystem ist der HyperPicture Toolkit, ein hypermediales Datenbanksystem das den objektorientierten Ansatz voll unterstützt. Spacepicture ist vollkommen objektorientiert aufgebaut, alle gespeicherten Daten werden mit Hilfe wohldefinierter Objekte repräsentiert.

Spacepicture verfügt über eine eigene Abfragesprache (SPQL, SpacePicture Query Language), die eine Datenmanipulation von einem einfachen ASCII Terminal erlaubt.



**Abbildung 2.1:** Prinzipielle Architektur von Spacepicture - Darstellung der Hauptmodule und der Kommunikationspfade zwischen den einzelnen Komponenten (Quelle: Kirste, 93).

Merkmalsbeschreibung	Detailbeschreibung	Detailwertung	Gesamtwertung
Graphical User Interface	Übersichtlichkeit	3	3
	Funktionalität	3	
	Interaktivität	3	
	Bedienungskomfort	3	
Suchmechanismen	point of interest	3	2.25
	region of interest	3	
	Überdeckung	0	
	Meta Informationen	3	
HS Netzwerk Backbone	> 1.5 Mbit/sec (T1)	0	0
	> 44.736 Mbit/sec (T3)	0	
	> 155 Mbit/sec (OC-3)	0	
Plattformunabhängigkeit	SUN	3	0.5
	SGI	0	
	HP	0	
	IBM	0	
	PC	0	
	Apple	0	
Client-Server Konzept		0	0
Kostenmanagement		0	0
Verteiltes Datenkonzept		1	1
Offenes System	CIP	0	0
	IDN mit DIF	0	
	Z39.50	0	
	SQL	0	
	TCP/IP	0	
	SNMP	0	
Kommerziell erhältlich		0	0
Globales Management		0	0
Lokales Management		0	0
Keine spezielle Client Software notwendig		0	0
Modulares System		2	2
Systemwertung:			9.25

Tabelle 2.1: Gesamtwertung für SpacePicture

Folgende Einschränkungen gelten für das System:

- keine Multi-User Fähigkeit,
- keine Standardabfrage-Sprache,
- kein Management,
- keine Kostenverrechnung,
- spezifisches Datenbanksystem (HyperPicture Toolkit) ohne relevante Marktpräsenz,
- Datenintegrität und Datensicherheit nicht berücksichtigt,
- kein Versionsupdate.

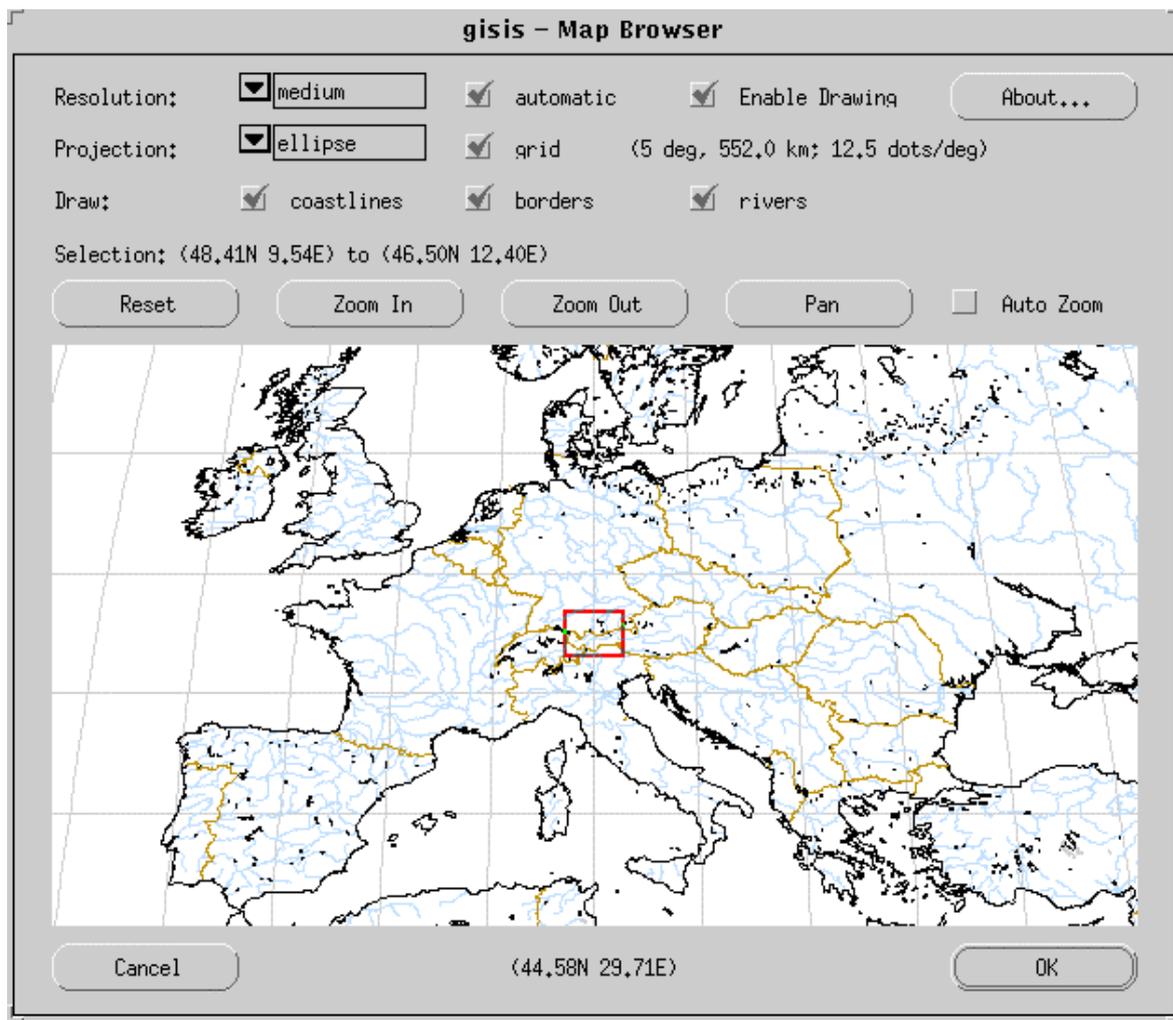
Tabelle 2.1 ist ein Versuch, das System zu bewerten.

### 2.1.2 GISIS

Das Deutsche Fernerkundungsdatenzentrum der DLR hat das auf Erddaten spezialisierte Informationssystem GISIS (Graphisches Intelligentes Satelliten Informationssystem) entwickelt und betreibt es als zentrale Benutzerschnittstelle zu seinem Daten- und Informationssystem [Lotz-Iwen, 95].

Um eine Reihe verschiedener Hardware Plattformen zu unterstützen, kann GISIS sowohl über eine reine ASCII-Schnittstelle als auch über eine graphische Schnittstelle via eine vektororientierte Karte für PCs und SUN Workstations bedient werden (siehe Abbildung 2.2).

GISIS beruht auf einer Client/Server Architektur (siehe Abbildung 2.3) unter Ausnutzung vorhandener Netzwerke (ISDN, X.25, Modemzugriff, Internetzugang) - das zugrunde gelegte Datenbankmodell ist ein relationales, wobei allerdings eine objektorientierte Zwischenschicht eingefügt worden ist. Die notwendige Client-Software kann kostenlos vom Anonymous-ftp-Server der DFD (SUN-Version) oder über das GISIS-Infoboard (PC-Version) kopiert werden, ein Upgrade auf höhere Versionen erfolgt automatisch beim Zugriff auf GISIS per Netzwerk.



**Abbildung 2.2:** Das User Interface von GISIS - Der vektororientierte Browser von GISIS unterstützt eine komfortable Suche nach Interessensregionen, gestattet die Suche nach Metainformationen sowie die individuelle Gestaltung der Darstellung

GISIS bietet zum Stichtag 12/96 folgende Dienste an:

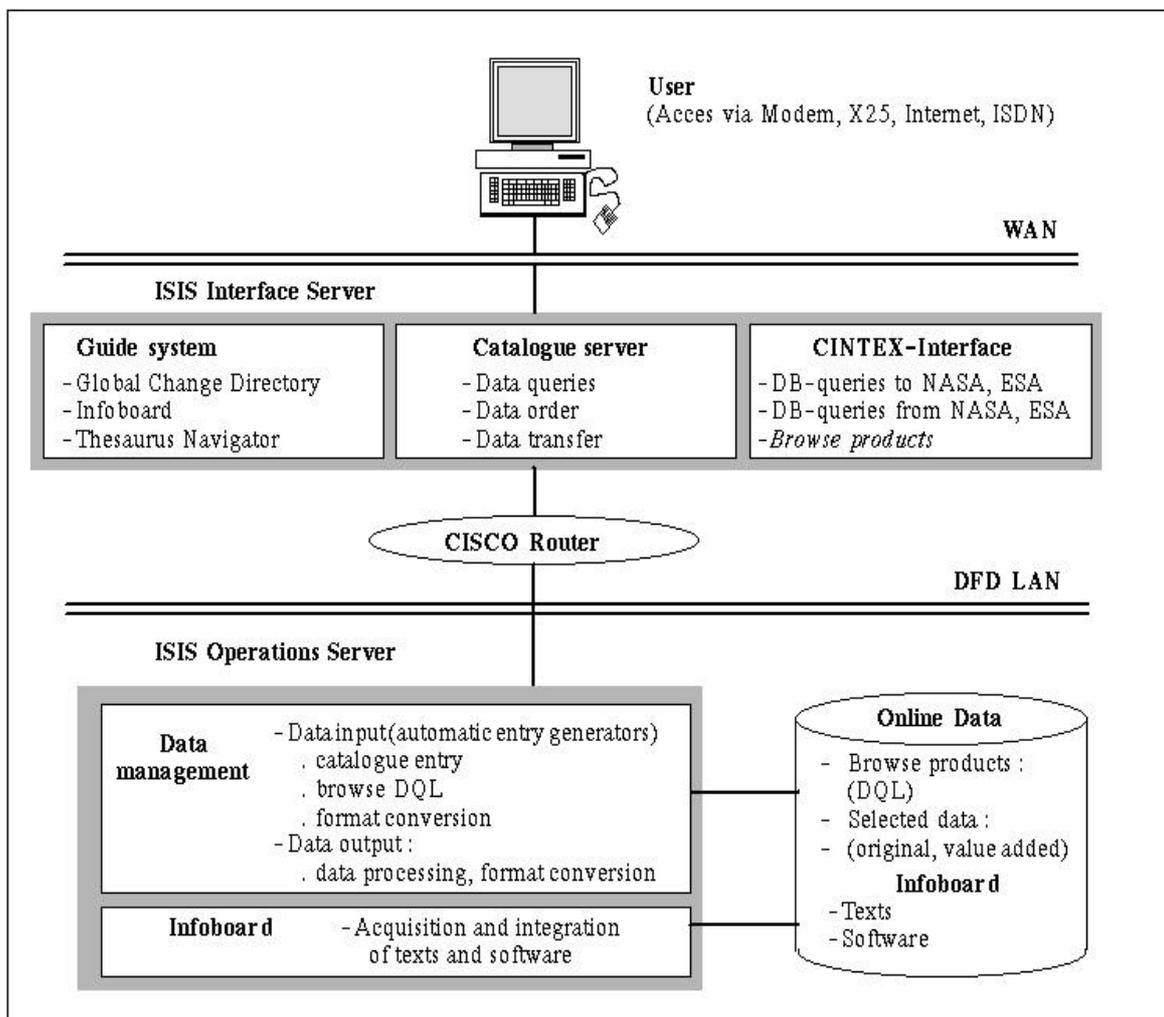
- Katalogabfrage<sup>1</sup> in verteilten Archiven, unterstützt durch einen Map Browser und eine geographische Namensdatenbank,
- Übertragung digitaler Quicklooks über Netzwerke zur Vorab-Visualisierung,
- Transfer von ausgewählten Daten (komplette Datensätze oder interaktiv definierte Teildatensätze) und Durchführung von online Datenbestellungen,
- Zugang zum International Directory Network IDN (Global Change Master Directory),

<sup>1</sup> über 30.000 digitale Bilder von 14 verschiedenen Sensoren und Produkten

- Weiterleitung von Katalogabfragen zum NASA-IMS und ESA-UIT durch das CINTEX-Interface von CEOS (zur Zeit nur für Pilotnutzer möglich),
- Bereitstellung eines Infoboards mit ergänzenden Informationen in Textform und Software Tools,
- Unterstützung der Informationsabfrage durch einen Thesaurus Navigator,
- Online Eingabe von Nutzerkommentaren.

Neuere Entwicklungen konzentrieren sich auf eine zusätzliche WWW-Schnittstelle und die Einbindung zukünftiger Daten von GOME (Ozonmessung) und der russisch-deutschen PRIRODA-Mission.

Das GISIS System weist derzeit folgende Einschränkungen auf:



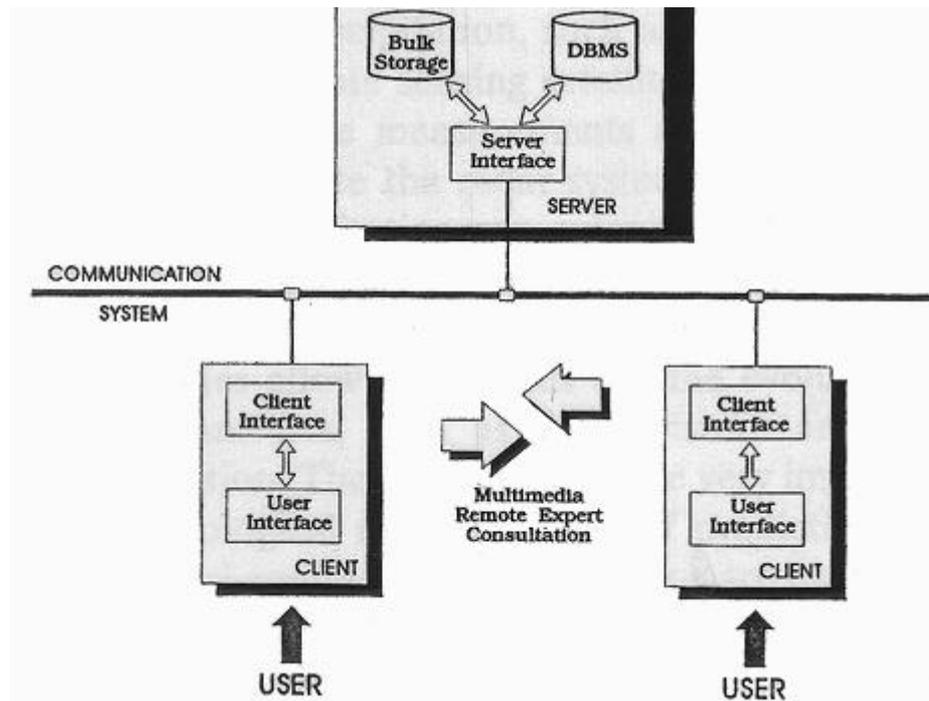
**Abbildung 2.3:** Die Systemarchitektur von GISIS Man erkennt die beiden Hauptserver des Systems, den Interface Server zur Anpassung an die jeweiligen Protokolle (samt CINTEX Interface) sowie den Operations Server, der die Daten bereitstellt sowie ein Infoboard über existierende Metadaten bereitstellt. (Quelle: <http://isis.dlr.de>)

- kein globales/lokales Management,
- keine Verrechnungsstrategien,
- ein fehlendes High Performance Netzwerk erschwert die Bestellung von Originaldaten,
- ungeeignet für planetarische Daten, da ausschließlich vektororientierte Darstellung.

Eine Gesamtbewertung des Systems wird in Tabelle 2.2 versucht. Insgesamt ergibt sich eine höhere Bewertung als für Spacepicture.

### 2.1.3 Das Arno-Projekt: „A distributed multimedia information system designed for the Arno Project“

Das Arno System ist ein multimediales, auf die Sektoren Meteorologie und Hydrologie gerichtetes Informationssystem, welches sich zunächst als eine spezifische Applikation zum Sammeln und Organisieren von meteorologischen und hydrologischen Daten präsentiert.



**Abbildung 2.4:** *Generelles Arno Layout* - Prinzipielle Architektur des Arno Systems zur Speicherung von meteorologischen und hydrologischen Daten (Quelle: [Nativi, 95])

Merkmalsbeschreibung	Detailbeschreibung	Detailwertung	Gesamtwertung
Graphical User Interface	Übersichtlichkeit	3	3
	Funktionalität	3	
	Interaktivität	3	
	Bedienungskomfort	3	
Suchmechanismen	point of interest	3	2.25
	region of interest	3	
	Überdeckung	0	
	Meta Informationen	3	
HS Netzwerk Backbone	> 1.5 Mbit/sec (T1)	0	0
	> 44.736 Mbit/sec (T3)	0	
	> 155 Mbit/sec (OC-3)	0	
Plattformunabhängigkeit	SUN	3	1
	SGI	0	
	HP	0	
	IBM	0	
	PC	3	
	Apple	0	
Client-Server Konzept		3	3
Kostenmanagement		0	0
Verteiltes Datenkonzept		1	1
Offenes System	CIP	0	1.29
	IDN mit DIF	3	
	CINTEX Interface	3	
	Z39.50	0	
	SQL	0	
	TCP/IP	3	
	SNMP	0	
Kommerziell erhältlich		0	0
Globales Management		0	0
Lokales Management		0	0
Keine spezielle Client Software notwendig		0	0
Modulares System		1	1
Systemwertung:			12.54

**Tabelle 2.2:** Gesamtwertung des GISIS Systems

Die vorgestellten Konzepte sind aber durchaus so allgemein, daß sie auch auf die Entwicklung von verteilten Informationssystemen in der Fernerkundung angewandt werden können. Im speziellen wurde das System für die Verwaltung und Observation von Multisensordaten aus relevanten meteorologischen Ereignissen entwickelt. Das System wurde im Jahre 1994 an der Universität von Florenz entwickelt und fertiggestellt [Nativi, 95].

Das System beruht auf einer Client/Server Architektur (siehe Abbildung 2.4), die ebenfalls 1994 an der Universität von Florenz entwickelt wurde, sowie einem leistungsfähigen Netzwerkbackbone. Das Netzwerk des spezifischen Systems besteht grundsätzlich aus mehreren LANs (Ethernet, 10 Mbit/sec) und einem MAN (QPSX<sup>1</sup>, 140 Mbit/sec) welches die LANs verbindet.

Der Klient verfügt über ein übersichtliches GUI, das eine komfortable Suche gestattet. Die graphisch eingegebenen Abfragen werden vom Klienten in ein SQL Statement übersetzt und an den Server geschickt. Dieser interpretiert die Anfrage und leitet via einem relationalen DBMS (Data Base Management System) eine Suche ein. Die erhaltenen Daten werden in ikonisierter Form (Quicklook) an den Klienten zurückgeschickt. Der Benutzer hat nun die Möglichkeit, auf die gewünschten Daten zu klicken, worauf der Datentransfer der Bilder in Originalauflösung eingeleitet wird. Weiters verfügt der Klient über Bildbearbeitungstools (statistische Filter, Faltungsoperationen, Pseudofarben, Zooming ...) - diese kann man auf die erhaltenen Bilddaten anwenden.

Ein besonderes Merkmal am Rande ist ein integriertes Videokonferenz-System, das eine beliebige Klient-Klient Kommunikation in Bild und Ton gestattet. Dies erleichtert die Zusammenarbeit einer örtlichen Gruppe ungemein, da über das System auch Daten (Bildschirmhalte) ausgetauscht werden können. Für die Zusammenstellung dieses heterogenen Soft- und Hardwaresystems erwies sich die Verwendung von Standards (SQL, TCP/IP, POSIX, ...) als besonders wichtig.

Das System weist eine Reihe von Einschränkungen auf:

- kein Accounting,
- kein Usermanagement,
- keine Verteilung der Daten,
- mit der Wahl von QPSX, einem veralteten Standard wurde die Bandbreite auf 150 Mbit/sec beschränkt,
- kein zentrales Netzwerkmanagement,
- die Bildbearbeitungstools laufen lokal, daher relativ teure und aufwendige lokale HW/SW nötig,

---

<sup>1</sup> QPSX (Queued Packet and Synchronous Switching) ist jene Netzwerktechnologie, aus der sich letzten Endes DQDB (Distributed Queue Dual Bus) entwickelt hat.

- zur Speicherung von planetarischen Daten - wegen der vektororientierten Darstellung - nicht geeignet,
- keine Trennung zwischen Archiven (Bilder in voller Auflösung) und Quicklooks; somit wird das Einbringen von neuen Archiven erschwert.

Die Gesamtwertung des Systems in Tabelle 2.3 reiht es zwischen Spacepicture und GISIS.

#### 2.1.4 VISTA

Das Projekt wurde 1990 unter dem Titel „Earth View“ vom Backgrounds Data Center (BDC) der Space Science Division am Naval Research Lab, Washington D.C. in der Absicht entwickelt, mit einem zwei- bzw. dreidimensionalen GUI gewisse Daten aus Erdsatelliten zu suchen und zu visualisieren. VISTA (Visual Interface for Space and Terrestrial Analysis) [Snyder, 93 / Snyder, 94] unterstützt sowohl die Suche als auch die Visualisierung und die Analyse der Daten, wobei für letzteres Feature eine Schnittstelle zu dem als Freeware erhältlichen Bildverarbeitungs-Werkzeug Khoros/Cantata angeboten wird.

Weiters lassen sich auch die Flugbahn des Satelliten und der Aufnahmebereich visualisieren. Die Applikation wurde auf einer UNIX Plattform unter dem X11 Window System (OSF/Motif) vollständig in C entwickelt. Der erste Prototyp verwendete auch einige SGI spezifische GL (Graphics Library) Routinen für das GUI, da er auf einer SGI Workstation entwickelt worden war. In der letzten Version von VISTA (2.0) wurden die meisten dieser Routinen bereits in OpenGL portiert, das auch auf andere Plattformen portierbar ist.

Die Architektur von VISTA besteht aus vier prinzipiellen Schichten (siehe Abbildung 2.5) deren Funktion im folgenden beschrieben werden:

##### **User Interface Schicht:**

Diese Schicht kümmert sich um alle graphischen Ausgaben, steuert also das GUI (siehe Abbildung 2.6) und ist somit für die User Interaktion verantwortlich. Das User Interface benutzt das X-Windows System (X11) und SGI's (Open) GL.

##### **Session Management Schicht:**

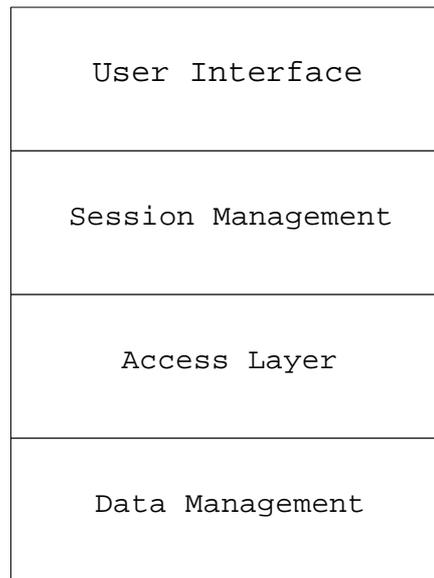
Hier werden User Identifikation, Zugriffsrechte, Metadaten File Position und Zeiger zu den System Files verwaltet.

##### **Access Schicht:**

Diese Schicht verwaltet zwei Hauptobjekte, die sogenannten „User Management Objects“ und die „Query Objects“. Die User Management Objects sind die Menge aller Dateien, die ein Benutzer während einer Sitzung (session) kreiert hat, während die Query Objects die User Abfragen sichern und die Verbindung zu den Bild- und Metadaten bilden.

Merkmalsbeschreibung	Detailbeschreibung	Detailwertung	Gesamtwertung
Graphical User Interface	Übersichtlichkeit	3	3
	Funktionalität	3	
	Interaktivität	3	
	Bedienungskomfort	3	
Suchmechanismen	point of interest	3	2.25
	region of interest	3	
	Überdeckung	0	
	Meta Informationen	3	
HS Netzwerk Backbone	> 1.5 Mbit/sec (T1)	3	1
	> 44.736 Mbit/sec (T3)	0	
	> 155 Mbit/sec (OC-3)	0	
Plattformunabhängigkeit	SUN	3	0.5
	SGI	0	
	HP	0	
	IBM	0	
	PC	0	
	Apple	0	
Client-Server Konzept		3	3
Kostenmanagement		0	0
Verteiltes Datenkonzept		0	0
Offenes System	CIP	0	0.85
	IDN mit DIF	0	
	CINTEX Interface	0	
	Z39.50	0	
	SQL	3	
	TCP/IP	3	
	SNMP	0	
Kommerziell erhältlich		0	0
Globales Management		0	0
Lokales Management		0	0
Keine spezielle Client Software notwendig		0	0
Modulares System		3	3
Systemwertung:			10.6

**Tabelle 2.3:** Gesamtwertung des Distributed Information Systems für das Arno Projekt



**Abbildung 2.5:** Die vier Schichten von VISTA

### **Data Management Schicht:**

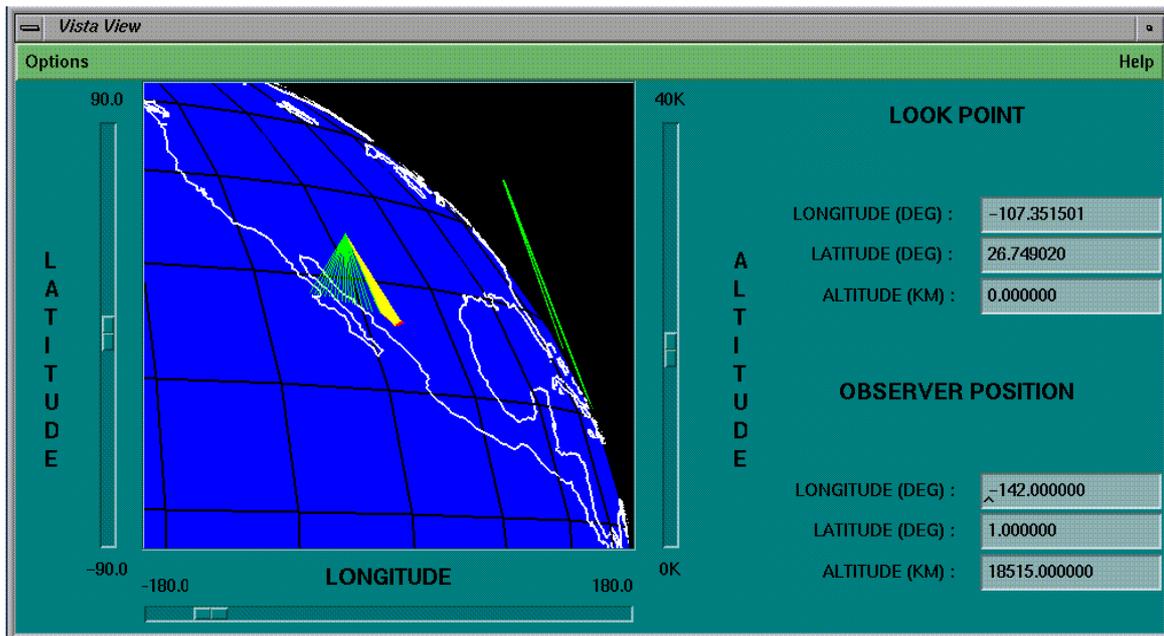
Dieser Layer ist in zwei separate Sublayers gegliedert, dem „Data Structure Flat File Management System“ und dem „RDBMS (Relationales Datenbank Management System)“. Die ausgereifere Version von VISTA ist jene mit dem Flat File DBMS, die Entwickler wollten jedoch eine offene Schnittstelle vorsehen. Somit bleibt im Idealfall dem Benutzer überlassen, welches RDBMS er benutzen möchte. In diesem Sinne zielt die aktuelle Forschungs- und Entwicklungsrichtung von VISTA darauf hin, einen Standard für ein verteiltes DBMS VISTA zu kreieren.

VISTA weist eine große Anzahl von wünschenswerten und unverzichtbaren Merkmalen auf, hat jedoch auch einige Einschränkungen:

- das Einbringen von neuen Daten erweist sich auf Grund des (noch) fehlenden RDBMS als schwierig (Metainformationen, geokodierte Bilder lokalisieren etc.);
- das GUI des Systems ist weitestgehend an die Verwendung von (Open)GL gebunden - dies wird zwar auf bereits vielen Plattformen angeboten, jedoch verfügt ein billiger PC nun einmal nicht über die graphischen Möglichkeiten einer teuren UNIX Workstation;
- VISTA unterstützt kein Beschneiden der Daten - Daten müssen in Originalauflösung (und -größe) bestellt werden;

Merkmalsbeschreibung	Detailbeschreibung	Detailwertung	Gesamtwertung
Graphical User Interface	Übersichtlichkeit	3	2.75
	Funktionalität	2	
	Interaktivität	3	
	Bedienungskomfort	3	
Suchmechanismen	point of interest	3	3
	region of interest	3	
	Überdeckung	3	
	Meta Informationen	3	
HS Netzwerk Backbone	> 1.5 Mbit/sec (T1)	0	0
	> 44.736 Mbit/sec (T3)	0	
	> 155 Mbit/sec (OC-3)	0	
Plattformunabhängigkeit	SUN	0	0.5
	SGI	3	
	HP	0	
	IBM	0	
	PC	0	
	Apple	0	
Client-Server Konzept		3	3
Kostenmanagement		0	0
Verteiltes Datenkonzept		0	0
Offenes System	CIP	0	0.43
	IDN mit DIF	0	
	CINTEX Interface	0	
	Z39.50	0	
	SQL	0	
	TCP/IP	3	
	SNMP	0	
Kommerziell erhältlich		0	0
Globales Management		0	0
Lokales Management		2	2
Keine spezielle Client Software notwendig		0	0
Modulares System		2	2
Systemwertung:			13.68

**Tabelle 2.4:** Gesamtwertung für das System „VISTA“



**Abbildung 2.6:** Graphische Benutzerschnittstelle von VISTA - Das 3D GUI von VISTA mit der Möglichkeit, nach Eingabe des Beobachtungspunktes Abfragen zu formulieren und die retournierten Daten zu analysieren.

- die Bildanalyse Tools von VISTA stützen sich auf die Funktionen die Khoros zur Verfügung stellt - Khoros ist ein Freewaresystem, professionelle Wartung und Weiterentwicklung sind nicht sichergestellt;
- das verteilte System verfügt über keinen geeigneten Netzwerkbackbone;
- es gibt kein leistungsfähiges Netzwerkmanagement.

Die Weiterentwicklung des Vista System wurde in seiner derzeitigen Form zu Beginn des Jahres 1996 eingestellt. Seit diesem Zeitpunkt wird Vista auf ein objekt-orientiertes System umgestellt<sup>1</sup>. Die Gesamtwertung in Tabelle 2.4 reiht Vista höher als die bisher besprochenen Systeme ein.

### 2.1.5 CEO / EWSE

Im Jahre 1992 begann die EC (European Commission) über das Joint Research Center mit einem Projekt, das die effektivere und kostenreduzierende Arbeit mit Erddaten gewährleisten soll. Das gesamte Projekt ist eigentlich mehr als „Marktinitiative“ zu sehen. Bisher entstand ein Prototyp, der sogenannte EWSE (European Wide Service Exchange) Browser als Vorschau auf die Entwicklung von EEOS (European Earth Observation System), eines koordinierten, dezentralen Erdbeobachtungs-Datennetzes. Anbei liefert die ESA Daten und die

<sup>1</sup> laut einer Auskunft von Bryan Dorland (Entwickler im National Research Laboratory)

Vorbearbeitung. Der Beitrag der EC ist das CEO (Center of Earth Observation). CEO wird für die Entwicklung von Applikationen sorgen, die die Useranforderungen befriedigen.

EEOS soll standardisierte Schnittstellen zu internationalen Initiativen wie dem NASA EOSDIS (Earth Observation System Data and Information System) und dem japanischen Satellitendaten Management und Informationssystem EOIS aufweisen.

Die prinzipiellen Phasen des Projekts sind wie folgt:

Tätigkeit	Zeitraum
Anforderungsanalyse	1992-93
Pathfinder Phase	1993-95
Design and Implementation Phase	1996-98

Zum Stichtag 8/1996 befand sich das Projekt in der sogenannten Pathfinder Phase. Sie ist für die Produktion eines Projektplanes für die Design- und Implementierungsphase verantwortlich. Hierzu sind umfangreiches Wissen über das System und die Useranforderungen erforderlich um CEO letzten Endes zum Leben zu erwecken. Der EWSE spielt dabei die Rolle eines Übungs- oder Laborsystems.

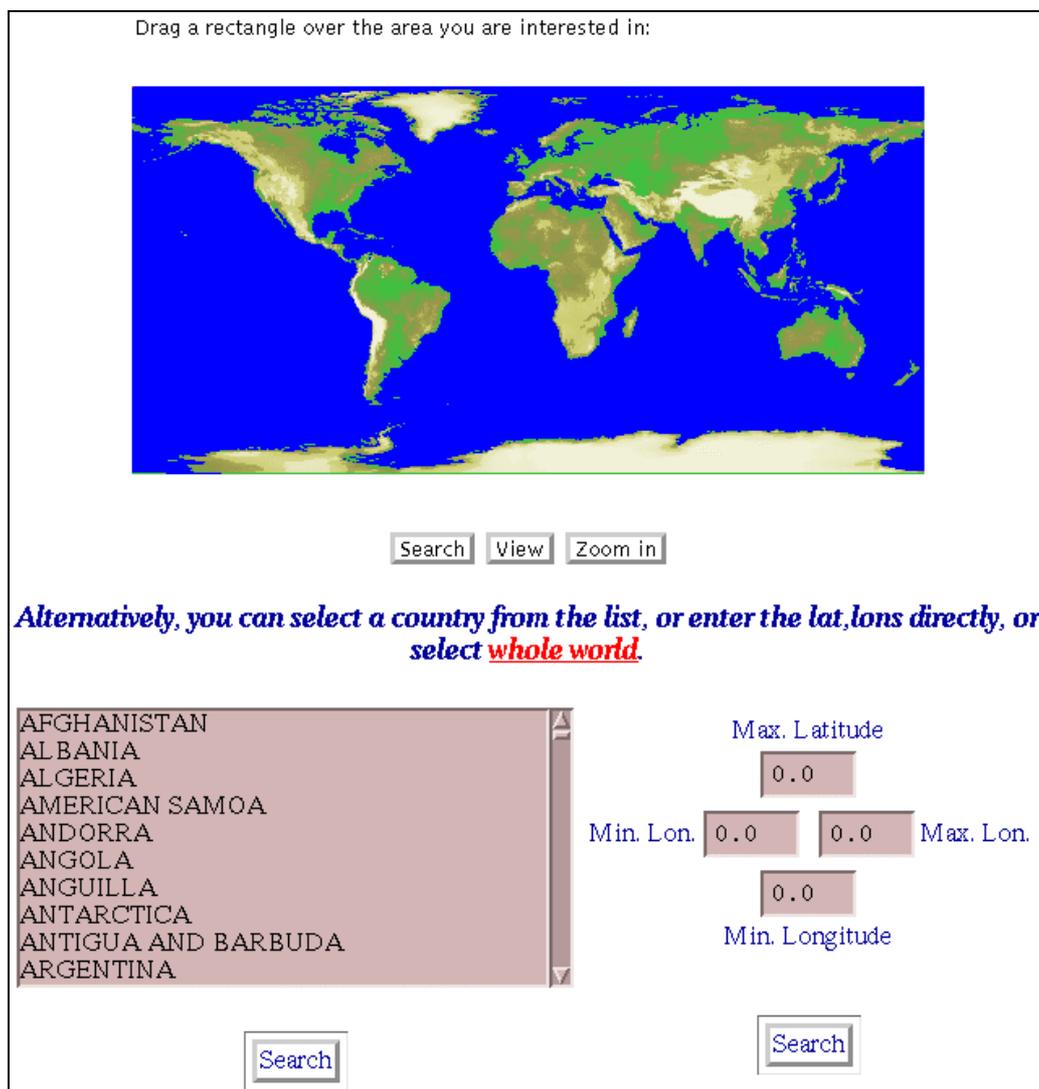
Die prinzipiellen Entwurfskriterien von CEO lassen sich wie folgt aufzählen:

- **Teilnehmer** sind alle Organisationen, die Erddaten in professioneller Art und Weise benutzen bzw. davon Nutzen haben. Dies sind u.a. Regierungen, Universitäten und Forschungszentren, Firmen, internationale Konsortien.
- **Autonomie der Organisationen** bedeutet, daß die Leistungen und Ressourcen von allen teilnehmenden Organisationen kostenlos benutzt werden dürfen. Die Organisationen sollen dadurch die Möglichkeit bekommen, einerseits die Implementierung von CEO-Services entsprechend der Anforderungsanalyse überprüfen zu können bzw. die Entwicklung von eigener Infrastruktur und die Erweiterung des eigenen Wissens voranzutreiben.
- **Dezentralisierter Ansatz** soll die Verringerung von zentralen Diensten bewirken. Die zentrale Koordination soll ausschließlich in der Initialisierungsphase bestehen, später jedoch völlig abgeschafft werden.
- **Standards** in SW und HW sind Grundvoraussetzungen, um EEOS einer breiten Öffentlichkeit zugänglich machen zu können.
- **Formate.** Um die zahlreichen Bildformate sinnvoll zu unterstützen, muß EEOS Konvertierungs- und Visualisierungssoftware zur Verfügung stellen.
- **Expertenrat.** Ein nicht unwesentlicher Dienst ist die Archivierung und

Bereitstellung von Expertenrat für Erdbeobachtungsdaten. EEOS hat für einen einfachen Such-, Speicher- und Bereitstellungsmechanismus Sorge zu tragen.

- **Flexibilität** ist bei den stets kürzer werdenden Innovationszyklen und in Anbetracht der sechsjährigen Projektdauer ein wesentliches Merkmal. EEOS muß ein offenes System bzgl. neuer Applikationen, Dienste und Netzwerktechnologien sein.

Der erste Prototyp, der EWSE, ist nun am Internet verfügbar und über WWW weltweit zugänglich. Der einfachste Weg, Benutzer und Anbieter in Kontakt zu



**Abbildung 2.7:** Graphische Benutzerschnittstelle des EWSE Browsers - Das GUI gestattet eine graphische Suche durch das Markieren von geographischen Regionen samt Zoomen. Weiters kann man auch manuell entweder durch Koordinateneingabe oder durch Auswahl von politischen Grenzen nach Bilddaten und Anbietern suchen (Quelle: **Lotz-Iwen, 95**).

bringen, ist die Bereitstellung eines „Clearing-Houses“. EWSE operiert als eine Art BBS (Bulletin Board Server), der es Benutzern erlaubt, Daten und Dienste zu finden bzw. umgekehrt gelingt es Anbietern, Benutzer für ihre Dienste zu finden.

Primär offeriert EWSE eine dynamische On-Line Datenbank, die stets mit den neuesten eingebrachten Informationen erneuert wird. Gemäß den verschiedenen Wünschen der Benutzer und Anbieter existieren mehrere Sichten auf diese Datenbank, die da wären:

- **Trade Shop.** Wie in einem richtigen Kaufhaus werden die Dienste in Regalen und geordnet nach Zusammengehörigkeit angeboten. Die unterschiedlichen Dienste werden - analog in einem Kaufhaus die Abteilungen - durch verschiedene Stockwerke repräsentiert.
- **Geographische Suche.** EWSE stellt ein 2D Interface zur komfortablen Suche von Satellitendaten zur Verfügung (siehe Abbildung 2.7). Benutzer können in einfacher Art und Weise in verschiedenen Auflösungsstufen ihre geographische Suchregion auswählen und sich eine Liste der zur Verfügung stehenden Daten anzeigen lassen. Diese Liste wird in Form von Yellow Pages ausgegeben. Die gesamte Suche ist mit Hilfe eines Java Applet implementiert. Dies garantiert Plattformunabhängigkeit (Netscape ist so gut wie auf allen Plattformen verfügbar und gratis) sowie ein inkludiertes Update Service, da die neueste Version des Applets stets bei der Verbindungsaufnahme geladen wird.

Registrierte Benutzer<sup>1</sup> können weiters über folgende Dienste verfügen:

- **Individualisierung des GUI.** Registrierte Benutzer erhalten einen eigenen Account, in dem persönliche Daten (z.B. Homepage, Adresse, ...) sowie das Layout des Benutzerinterfaces gespeichert werden. So kann man etwa festlegen, ob man bei jeder Auswahl eines Dienstes die ausführlichen Hilfetexte haben will oder lieber gleich Aktionen veranlaßt.
- **Technische Diskussionen, Hilfeofferierung.** Weiters erlaubt eine Registrierung die Teilnahme an technischen Diskussionen (ähnlich Newsgroups) sowie die Anfrage oder Bereitstellung von Hilfe.
- **Spezielle Interessensgruppen, Newsletter, Suche nach Daten und Anbietern.** Sofort nach der Anmeldung stehen leistungsfähige Datenbankdienste zur Verfügung, die z.B. eine Suche nach ganz bestimmten Anbietern bzw. auch eine Auflistung aller derzeit vorhandenen erlauben.
- **Kommerzielle Datenanbieter** können nach der Registrierung selbst ein

---

<sup>1</sup> Man kann sich bei EWSE kostenlos und ohne weitere Verpflichtungen als Benutzer registrieren lassen. Hierzu bekommt man ein Passwort und eine Benutzer ID, die in Zukunft bei jedem Login ins EWSE angegeben werden kann.

Kaufhaus eröffnen und ihre Dienste anbieten. Dies inkludiert die Möglichkeit einer gratis Werbung für das eigene Unternehmen. Aber auch die Möglichkeit neue Benutzer zu finden bzw. ihre Anforderungen zu analysieren sind gegeben.

Das gesamte Konzept von EEOS beruht auf dem Schlüsselement „Service“. Service ist definiert als eine Aktivität, die von einem Benutzer ausgeführt wird. Somit ist eine Funktion, die eine Liste von Services hervorbringt, ebenfalls ein Service. Services können von jedem Teilnehmer eingebracht werden, also von Geo-Datenanbietern, von Universitäten und Forschungszentren, aber auch von individuellen Benutzern. Jedem Anbieter steht frei die Konditionen seiner Dienste festzulegen, es sei denn sie widersprechen den Entscheidungen eines relevanten Benutzerkomitees.

Das EWSE System, das eher einem Diskussionsforum als einem professionellen Such- und Archivierungssystem gleicht, weist etliche Einschränkungen auf:

- Keine Suche nach Überdeckung möglich,
- Bestellung von Daten bzw. Kostenverrechnung über das Netzwerk ist nicht implementiert,
- keine Hochgeschwindigkeits Netzwerkunterstützung,
- kein Client Server Konzept,
- weder globales noch lokales Management,
- keine Unterstützung von Abfrage Standardprotokollen.

Aus Tabelle 2.5 ist zu entnehmen, daß das EWSE System die bisher höchste Bewertung erhält.

### 2.1.6 NASA's EOSDIS

Das EOS (Earth Observing System) [EOSDIS, 96] ist ein Teil des NASA Projektes „Mission to Planet Earth“ (MTPE) [MTPE, 96], Das MTPE Projekt ist der größte von der NASA gelieferte Beitrag zum GRCP (Global Change Research Program). Die DIS (Data Information System) Komponente wurde als verteiltes System zur Unterstützung von Archivierung und Verteilung von Geo-Daten zwischen mehreren Archiven konzipiert. Archive werden durch ein IMS (Information Management System) verbunden, welches Benutzern die Suche und die Bestellung von Daten gestattet.

Merkmalsbeschreibung	Detailbeschreibung	Detailwertung	Gesamtwertung
Graphical User Interface	Übersichtlichkeit	3	2
	Funktionalität	1	
	Interaktivität	1	
	Bedienungskomfort	3	
Suchmechanismen	point of interest	0	0.67
	region of interest	3	
	Überdeckung	0	
	Meta Informationen	3	
HS Netzwerk Backbone	> 1.5 Mbit/sec (T1)	0	0
	> 44.736 Mbit/sec (T3)	0	
	> 155 Mbit/sec (OC-3)	0	
Plattformunabhängigkeit	SUN	3	3
	SGI	3	
	HP	3	
	IBM	3	
	PC	3	
	Apple	3	
Client-Server Konzept		3	3
Kostenmanagement		0	0
Verteiltes Datenkonzept		3	3
Offenes System	CIP	0	0.43
	IDN mit DIF	0	
	CINTEX Interface	0	
	Z39.50	0	
	SQL	0	
	TCP/IP	3	
	SNMP	0	
Kommerziell erhältlich		0	0
Globales Management		0	0
Lokales Management		0	0
Keine spezielle Client Software notwendig		3	3
Modulares System		2	2
Systemwertung:			17.1

**Tabelle 2.5:** Gesamtwertung für das EWSE System

Die Version 0 wurde im September 1994 freigegeben, sie war eine Co-Produktion zwischen dem Goddard Space Flight Center (IMS Team) und Teams aus existierenden Daten-Zentren. Das System basiert auf einem relationalen DBMS (Sybase), welches um eine „spatial option“ erweitert wurde. Eine Migration zu dem von Informix aufgekauften ORDBMS (Object Relational Data Base Management System) Illustra ist in der Release B vorgesehen [Simpson, 95]. Dieses neue System wird auf dem gerade im Standardisierungsprozeß befindlichen SQL/3 Standard basieren [SQL/3, 96]. Dadurch ist eine allgemeine Performance Steigerung zu erwarten, da das gesamte EOSDIS in C++ programmiert wurde und durch ein ORDBMS eine Abbildung auf das relationale Datenmodell entfallen kann [EOSDIS/B,96]. Die Idee von EOSDIS besteht darin, möglichst viele Erfahrungen und existierende Applikationen/Schnittstellen von den DAACs (Distributed Active Archive Centers) zu verwenden [Ryan, 94].

Das EOSDIS ist ein von der NASA zur Verfügung gestelltes wissenschaftliches System, für die Benutzung sind keinerlei Gebühren zu entrichten. Die Beta Version 1.3 des IMS Gateways unterstützt die folgenden Punkte:

- Umfangreiche Suche (Parameter, Sensortyp, Datensatz, Datenarchiv, Zeitangabe, region of interest),
- Integrierter Browser für das Ansehen von quicklooks (wo vorhanden),
- Datenbestellung über ein Formular, das an das jeweilige Archiv weitergeleitet wird, möglich,
- Integrierte On-line Hilfe (User Manual),
- Benutzer Kommentare können abgesetzt werden,
- Browser Selbsttest - Test ob der verwendete Netzworlbrowser mit dem IMS Gateway verwendet werden kann,
- Bereits durchgeführte Abfragen können gespeichert und zu einem späteren Zeitpunkt wiederholt werden (individuelle User Präferenzen können ebenfalls gespeichert werden).

Das EOSDIS System verfügt über spezielle Client Software, die für SGIs und SUNs verfügbar ist (Stand Juni 1996). Die Client Software besitzt ein sehr funktionelles und komfortables Benutzer Interface, siehe hierzu auch Abbildung 2.8, welches ein Abfrage Formular zur Verfügung stellt. Das User Interface ist sehr gut gegliedert und ist übersichtlich und funktionell gestaltet.

The screenshot shows the 'EOSDIS Client Search Form' with the following sections and annotations:

- Search Type:** Inventory, Directory, Guide
- Geographic Area:** Global Search
- Geographic Coordinates:** N. Latitude, W. Longitude, E. Longitude, S. Latitude, Latitude, Longitude, Northernmost, Easternmost, Southernmost, Westernmost.
- Date/Time:** Continuous Time Range, Start Date/Time, End Date/Time.
- Parameters:** Campaign/Project, Source/Platform, Sensor, Parameters, Processing Level, Dataset ID, Data Center ID, Text String, Day/Night, Browse Only Granules, Number of Granules returned per Dataset.
- Buttons:** Execute Search..., Result Status..., Save Search, Retrieve Search, Clear Field, Close.

Annotations with arrows pointing to specific fields:

- Enter geographic location criteria.** Points to the Latitude and Longitude input fields.
- Enter time criteria.** Points to the Start and End Date/Time input fields.
- Enter criteria by typing in the entry fields or selecting from a valid list.** Points to the 'List...' buttons next to the parameter fields.
- To access a list of selectable entries, the user should click on a List button.** Points to a 'List...' button.

**Abbildung 2.8:** Suchformular des EOSDIS Klienten. Man erkennt die umfangreichen Möglichkeiten nach Daten zu suchen. Prinzipiell ist das Formular in eine Eingabe eines geographischen Suchgebietes (rechts oben), in eine Eingabe eines Zeitraumes (rechts unten) sowie in die Eingabe von zahlreichen Parametern (Sensortyp, Datenarchiv, ...) aufgeteilt.

In Tabelle 2.6 sind die derzeit angeschlossenen Datenzentren samt den angebotenen Daten aufgelistet.

Für die Suche nach Daten stehen vier verschiedene Möglichkeiten zur Verfügung:

- Character Terminal über telnet,
- GUI X-Windows Interface über telnet (mindestens 40 kBit/sec Link notwendig),
- Herunterladbarer Klient derzeit für SGIs (IRIX) und SUNs (Solaris),
- WWW Interface mit der Möglichkeit graphisch die „region of interest“ auszuwählen.

Datenzentrum	Art der Daten
ASF DAAC -Alaska SAR Facility	Polar Prozessierung, SAR Produkte
EDC DAAC -EROS Data Center	Land Prozessierungs Daten
CIESIN-SEDAC -The Consortium for International Earth Science Information Network	Sozioökonomisches Datenzentrum - menschliche Interaktionen in der Umwelt
GSFC DAAC -Goddard Space Flight Center	Obere Atmosphäre, globale Biosphäre, atmosphärische Dynamik und Geophysik
JPL DAAC -Jet Propulsion Lab	physische Ozeonographie
LARC DAAC -Langley Research Center	Strahlung, Chemie in der Troposphäre, Wolken
MSFC DAAC -Marshall Space Flight Center	Hydrologische Zyklen
NOAA SAA -National Oceanic and Atmospheric Administration - Satellite Active Archive	Satelliten Daten (Atmosphäre, Land, Ozean, Erdforschung, Fernerkundung)
NSIDC DAAC -National Snow and Ice Data Center	Schnee und Eis, Cyrosphäre und Klima Daten
ORNL DAAC -Oak Ridge National Lab	Bio-, Geochemische Dynamiken

**Tabelle 2.6:** Die Tabelle faßt alle derzeit über das IMS des EOSDIS verfügbaren Datenzentren zusammen

Die einzelnen DAACs sind über T3 Leitungen verbunden, was einer Bandbreite von 44,736 Mbit/sec entspricht.

Folgende Einschränkungen weist das System in der derzeitigen Beta-Version 1.3 vom 11. April 1996 auf:

- Es wird kein Standard Abfrage Protokoll (wie etwa Z39.50 oder CIP) verwendet,
- Plattformunabhängigkeit nur bedingt, WWW Interface weist bei weitem nicht die Komfortabilität des GUI auf. GUI derzeit nur für SUN/SGI,
- Schwaches lokales Systemmanagement (lediglich Plattenplatz wird getestet bevor ein telnet Benutzer angenommen wird),
- Kein globales Systemmanagement,
- Minimale Modularität - als Prototyp entwickelt, Hauptaugenmerk auf Funktionalität,
- Bestellung von (kostenpflichtigen) Daten nicht möglich und auch nicht geplant. Alle angebotenen Daten sind derzeit gratis,
- Keine Hochgeschwindigkeits-Netzwerk Unterstützung geplant.

Merkmalsbeschreibung	Detailbeschreibung	Detailwertung	Gesamtwertung
Graphical User Interface	Übersichtlichkeit	3	3
	Funktionalität	3	
	Interaktivität	3	
	Bedienungskomfort	3	
Suchmechanismen	point of interest	3	2.25
	region of interest	3	
	Überdeckung	0	
	Meta Informationen	3	
HS Netzwerk Backbone	> 1.5 Mbit/sec (T1)	3	2
	> 44.736 Mbit/sec (T3)	3	
	> 155 Mbit/sec (OC-3)	0	
Plattformunabhängigkeit	SUN	3	1
	SGI	3	
	HP	0	
	IBM	0	
	PC	0	
	Apple	0	
Client-Server Konzept	3	3	3
Kostenmanagement	0	0	0
Verteiltes Datenkonzept	2	2	2
Offenes System	CIP	0	0.43
	IDN mit DIF	0	
	CINTEX Interface	0	
	Z39.50	0	
	SQL	0	
	TCP/IP	3	
	SNMP	0	
Kommerziell erhältlich		0	0
Globales Management		0	0
Lokales Management	1	1	1
Keine spezielle Client Software notwendig	2	2	2
Modulares System	0	1	1
Systemwertung:			17.6

**Tabelle 2.7:** Gesamtwertung für das NASA EOSDIS Geo-Daten Abfragesystem



**Abbildung 2.9:** *Graphisches WWW Interface von ImageNet* - Das WWW Interface von ImageNet. Via dem bekannten und kostenlos erhältlichen Netscape kann man per Zooming und markieren einer Region of Interest Quicklooks anzeigen und hochauflösende Daten endgültlich bestellen.

Tabelle 2.7 ist die Gesamtwertung und vergibt eine geringfügig höhere Beurteilung als EWSE.

Zusammenfassend kann man sagen, daß EOSDIS für ein rein wissenschaftliches System überraschend viel Komfort und Funktionalität bietet. Um eine Kommerzialisierung allerdings tatsächlich durchführen zu können, müßte noch einige Arbeit in Themen wie globales/lokales Management, Plattformunabhängigkeit, Verwendung von Standardabfrageprotokollen sowie Bestellung kostenpflichtiger Daten über das Netz investiert werden.

### 2.1.7 CORE SW ImageNet

CORE Software Technology (CST) Inc. [ImageNet, 94] ist eine auf Geo-Daten und Bilddatenmanagement spezialisierte Firma mit Hauptsitz in Pasadena, Kalifornien. CST stellt mit ihrem ImageNet die erste kommerziell erhältliche On-Line Datensuche und Bestellung per remote access zur Verfügung. ImageNet inkludiert das SW Paket TheCore, welches mittels GUI die Anzeige und Manipulation der bestellten Daten ermöglicht. An Manipulationen werden prinzipielle Basisfunktionen wie Zooming, Drehen, Histogramm Anzeige, Beschriftung und Vektorüberlagerung geboten. Weiters bietet TheCore auch ein leistungsfähiges Filemanagement, um die heruntergeladenen Daten vernünftig verwalten zu können.

Generell ist die Architektur von ImageNet darauf ausgelegt, verteilte Daten Archive zu unterstützen - im Moment befinden sich jedoch alle angebotenen Daten in einem zentralen Archiv im CST Hauptquartier in Kalifornien.

ImageNet stützt sich prinzipiell auf eine Client/Server Architektur, wobei zwischen den folgenden Komponenten unterschieden wird:

- ImageNet Client
- ImageNet Server
- TheCore

Der „ImageNet Client“ ist praktisch eine Ergänzung zur Applikation „TheCore“. Der Klient stellt die Query Formulare und die auf TCP/IP Sockets beruhende Kommunikation zur Verfügung, während TheCore für die Raster bzw. Vektoranzeige und Manipulation der bestellten Daten verantwortlich ist. Der ImageNet Server lauscht an den jeweiligen Ports nach Klienten-Kommandos (SQL Kommandos) und verteilt sie per Kommandozeilen Schnittstelle an ein RDBMS welches SQL versteht. In der RDB werden nur Metainformationen gespeichert, Bilddaten werden extra - mit einem eindeutigen Filenamen als Zeiger - abgelegt. Bevor hochauflösende Daten heruntergeladen werden, schickt der Server eine Reihe von JPEG komprimierten „thumbnails“, die das Suchresultat repräsentieren. Der ImageNet Server kümmert sich auch um Dinge wie Autorisierung und Verrechnung. Nur berechtigte Benutzer, in der Regel der DB Administrator, können Daten einbringen, verändern oder löschen.

In Hinblick auf das exponentiell wachsende Internet und des boomenden WWW Dienstes bietet CST auch ein WWW Interface via Netscape Browser an (siehe Abbildung 2.9). Der ImageNet Server wurde in jener Hinsicht erweitert, daß er nun Netscape als vollwertigen ImageNet Client unterstützt. Somit wurde die Verfügbarkeit des Klienten von ursprünglich nur UNIX auf einfache Art und Weise auf eine große Anzahl von Plattformen<sup>1</sup> und Betriebssystemen erweitert. Die neue Java Programmiersprache zum Erzeugen von Applets wird aber noch nicht verwendet. Das SW Paket TheCore kann in Netscape als „Helper Application“ integriert und aufgerufen werden um die Bildmanipulationsalgorithmen und das Filemanagement nicht missen zu müssen.

Derzeit verfügt das CORE ImageNet über Bilddaten von EOSAT (Landsat 4,5), Interbranch Association SOVINFORMSPUTNIK (russische Fernerkundungsdaten), SPOT (Spot 1,2,3), ACRES (Australia Center for Remote Sensing), EINET (Eurimage Corporation für Europa - bietet Zugang zu über 40 Anbietern in Europa und Nordafrika) und ETAK (bietet digitales Kartenmaterial für die USA, Europa

---

<sup>1</sup> PC und Macintosh

und Asien an).

Das CORE ImageNet System macht zwar einen sehr ausgereiften Gesamteindruck, dennoch gibt es eine Reihe von Einschränkungen:

- keine Unterstützung von Hochgeschwindigkeits-Netzwerken, der Zugang fordert vor allem beim Laden von Graphiken enorme Geduld vom Benutzer,
- keine Unterstützung von Standard Abfrage Protokollen,
- das Netscape WWW Interface bietet gegenüber der speziellen Client Software nur unzureichende Funktionalität (keine Bildmanipulation). Wünschenswert wäre die Verwendung von Java Applets,
- keine Bestellung der Daten über das Netz und somit auch kein Kostenmanagement. Es wird lediglich das Senden eines Bestellwunsches per E-mail an den Anbieter unterstützt,
- kein globales Systemmanagement,
- Software relativ teuer, Server Software 15.000\$, Client Software 500\$,

Die Gesamtwertung des Core ImageNet Systems ist in Tabelle 2.8 mit der bisher höchsten Bewertung zu finden.

### 2.1.8 AVHRR CD-Browser

Der AVHRR (Advanced Very High Resolution Radiometry)<sup>1</sup> CD-Browser wurde speziell für das Anzeigen und Bestellen von NOAA (National Oceanographic and Atmospheric Administration) Satellitenbildern entwickelt [Mungo, 94]. Der Suchvorgang ist lokal, die Daten werden der CD entnommen. Es gibt jedoch die Möglichkeit die Daten auch über das WWW mit Hilfe eines speziellen Interfaces zur Verfügung zu stellen. Die Bilddaten stammen aus dem ESA ECTN (European Coordinated Tiros Network) und bestehen auf der CD aus 2000 Quicklooks. Dies sind in der Auflösung reduzierte Bilder der rund 50.000 hochauflösenden Bilder des ESA Archivs, das bei der ESRIN (European Space Research Institute) in Frascati weilt.

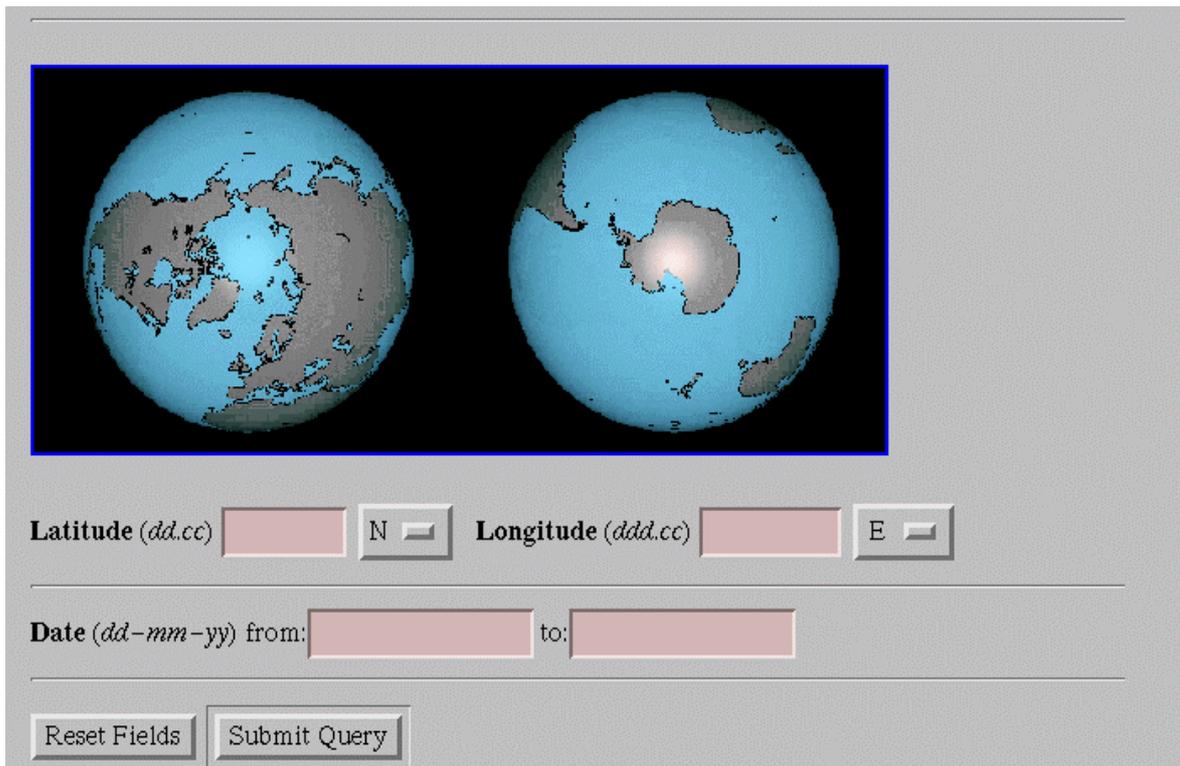
Der CD-Browser unterstützt IEF (Inventory Exchange Format) für den Katalog und den GIF (Graphics Interchange Format) Standard bei den Quicklooks. Jeder Quicklook zeigt mit dem bei der ESA/ESRIN eigens dafür entwickelten Algorithmus in Pseudofarben Land, Wasser, Schnee/Eis Oberflächen, Feuer und Wolken. Diesen Algorithmus kann man wie folgt beschreiben:

---

<sup>1</sup> Die Auflösung beträgt 1km/Pixel

Merkmalsbeschreibung	Detailbeschreibung	Detailwertung	Gesamtwertung
Graphical User Interface	Übersichtlichkeit	3	2.5
	Funktionalität	3	
	Interaktivität	1	
	Bedienungskomfort	3	
Suchmechanismen	point of interest	3	2.25
	region of interest	3	
	Überdeckung	0	
	Meta Informationen	3	
HS Netzwerk Backbone	> 1.5 Mbit/sec (T1)	0	0
	> 44.736 Mbit/sec (T3)	0	
	> 155 Mbit/sec (OC-3)	0	
Plattformunabhängigkeit	SUN	3	2
	SGI	3	
	HP	3	
	IBM	3	
	PC	0	
	Apple	0	
Client-Server Konzept		3	3
Kostenmanagement		0	0
Verteiltes Datenkonzept		3	3
Offenes System	CIP	0	0.43
	IDN mit DIF	0	
	CINTEX Interface	0	
	Z39.50	0	
	SQL	0	
	TCP/IP	3	
	SNMP	0	
Kommerziell erhältlich		3	3
Globales Management		0	0
Lokales Management		3	3
Keine spezielle Client Software notwendig		0	0
Modulares System		3	3
Systemwertung:			22.18

**Tabelle 2.8:** Gesamtwertung für das Core ImageNet



**Abbildung 2.10:** *Graphisches WWW Interface des AVHRR Browsers* - Das WWW Interface des AVHRR Browsers (Ionia Query Page) erlaubt die Suche nach poi (Point Of Interest) mit zusätzlicher Angabe des Zeitraumes

- man bekommt einen prinzipiellen Eindruck von der Bildqualität des hochauflösenden Bildes,
- funktioniert für jedes von einem NOAA Satelliten aufgenommene Bild, unabhängig von Aufnahmehöhe und Aufnahmezeitpunkt, spezifische interessante Oberflächencharakteristiken werden bei den Suchresultaten hervorgehoben (z.B. Wolkenhöhe, Wassertemperatur, Vegetationsdichte, Feuer, Schnee und Eis, ...),
- der Browser unterstützt auch ein WWW Interface (siehe Abbildung 2.10).

Weiters unterstützt der Browser die Suche nach Metadaten wie etwa: Aufnahmedatum, Koordinateneingaben, Satellit, Tag/Nachtaufnahme, Bearbeitungsstation, Grad der Wolkenbedeckung in Prozent. Sobald die gewünschten Daten ausgewählt worden sind, werden die in Frage kommenden Quicklooks als „thumbnails“ mit Zusatzinformationen (Koordinaten, Aufnahmezeitpunkt, Satellit, ...) auf dem Schirm dargestellt. Durch Klicken erhält man das vollauflösende Quicklook sowie Informationen zur Bestellung der Daten per Post (Band, CD-ROM). Die originalauflösenden AVHRR Daten werden kommerziell von EURIMAGE vertrieben.

Der ESA CD-ROM Browser weist folgende Einschränkungen auf:

- keine Unterstützung von Accounting, Daten müssen per Post bestellt und bezahlt werden,
- kein Usermanagement,
- keine Netzwerkverbindung, für neue Satellitendaten muß jedesmal eine neue CD-ROM umständlich per Post bestellt werden,
- keine Bildbearbeitungsalgorithmen,
- Suche nach roi (region of interest) wird nicht unterstützt,
- auf einen speziellen Satelliten- (NOAA) und Datentyp (Erddaten) fixiert und spezialisiert,
- verwendet für seine Kataloge keinen internationalen Standard (IEF anstatt IDN - International Directory Network [IDN, 93] bzw. CINTEX - Ceos INterchange EXperiment).

Eine Gesamtwertung des Systems in Tabelle 2.9 ergibt den bisher geringsten Wert.

### **2.1.9 Gesamtwertung und Gegenüberstellung**

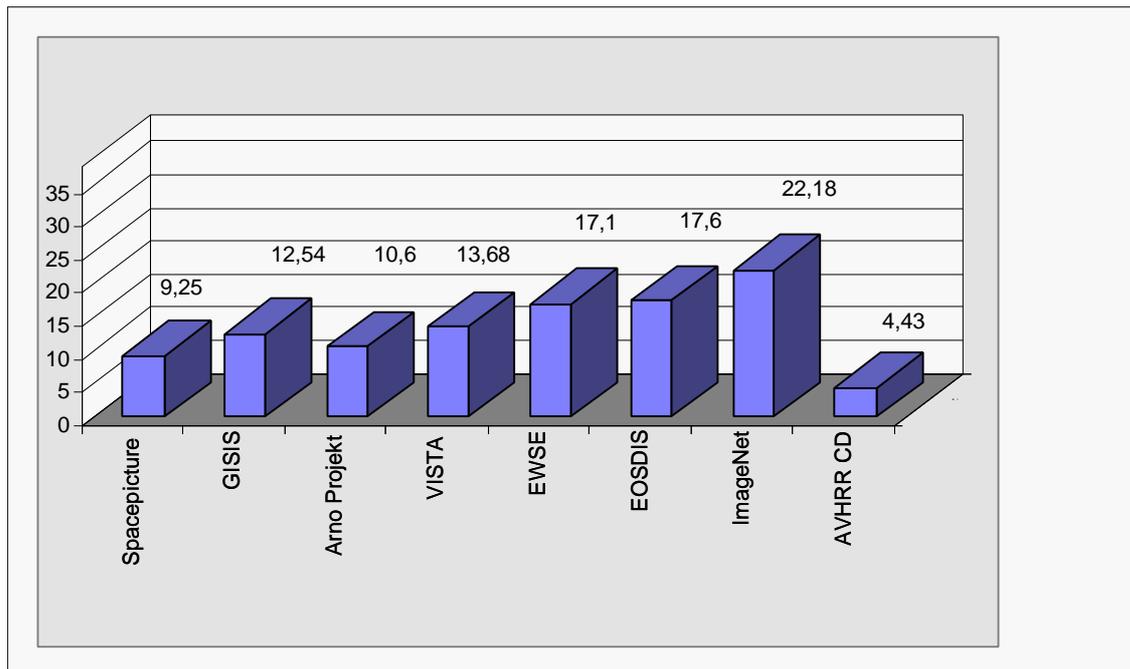
Eindeutiger Punktesieger (siehe Diagramm 2.1) ist erwartungsgemäß das erste kommerziell erhältliche System zum Suchen und Bestellen von Bilddaten aus existierenden Archiven von CORE Systems Inc. Dennoch ist es mit einer Gesamtwertung von 22.18 Punkten noch relativ weit von der möglichen Höchstwertung von 39 Punkten. Die hohe Wertung ist vor allem durch die kommerziellen Erhältlichkeit (und somit einer damit verbundenen professionellen Wartung), dem lokalen Management und der Modularität zu erklären.

Dies sind auch Dinge, die CORE Systems den beiden engsten Verfolgern EOSDIS (17.6 Punkte) und EWSE (17.1 Punkte) voraus hat. Beim EWSE Browser ist besonders die Verwendung eines Java Applets hervorzuheben, welches die Funktionalität eines Programmes bei gleichzeitiger Plattformunabhängigkeit über das WWW garantiert. Außerdem werden jegliche Update Probleme hinfällig, da jeweils die neueste Version des Applets beim Aufruf des Browsers geladen wird.

Im Mittelfeld befinden sich VISTA (13.68 Punkte), GISIS (12.54 Punkte) und das Arno Projekt System (10.6 Punkte). Die herausragenden Merkmale von VISTA sind sicherlich das gute GUI, das zum Teil vorhandene lokale Management sowie die Modularität. GISIS glänzt vor allem durch seine zwei Interfaces zu IDN (DIF) und dem CINTEX Zugang. GISIS ist auch das einzige von den untersuchten Systemen, welches eine Standardschnittstelle zu internationalen Datenanbietern unterstützt.

Merkmalsbeschreibung	Detailbeschreibung	Detailwertung	Gesamtwertung
Graphical User Interface	Übersichtlichkeit	3	2
	Funktionalität	1	
	Interaktivität	1	
	Bedienungskomfort	3	
Suchmechanismen	point of interest	3	1.5
	region of interest	0	
	Überdeckung	0	
	Meta Informationen	3	
HS Netzwerk Backbone	> 1.5 Mbit/sec (T1)	0	0
	> 44.736 Mbit/sec (T3)	0	
	> 155 Mbit/sec (OC-3)	0	
Plattformunabhängigkeit	SUN	3	0.5
	SGI	0	
	HP	0	
	IBM	0	
	PC	0	
	Apple	0	
Client-Server Konzept		0	0
Kostenmanagement		0	0
Verteiltes Datenkonzept		0	0
Offenes System	CIP	0	0.43
	IDN mit DIF	0	
	CINTEX Interface	0	
	Z39.50	0	
	SQL	0	
	TCP/IP	3	
	SNMP	0	
Kommerziell erhältlich		0	0
Globales Management		0	0
Lokales Management		0	0
Keine spezielle Client Software notwendig		0	0
Modulares System		0	0
Systemwertung:			4.43

**Tabelle 2.9:** Gesamtwertung für den AVHRR Browser



**Diagramm 2.1:** Vergleichspunktwertung aller untersuchten Geo-Datensysteme. Die höchste zu erreichende Punktzahl war 39. Das erste kommerzielle System, ImageNet, erreicht mit 22,18 Punkten den ersten Platz. Details zur Wertung siehe Text.

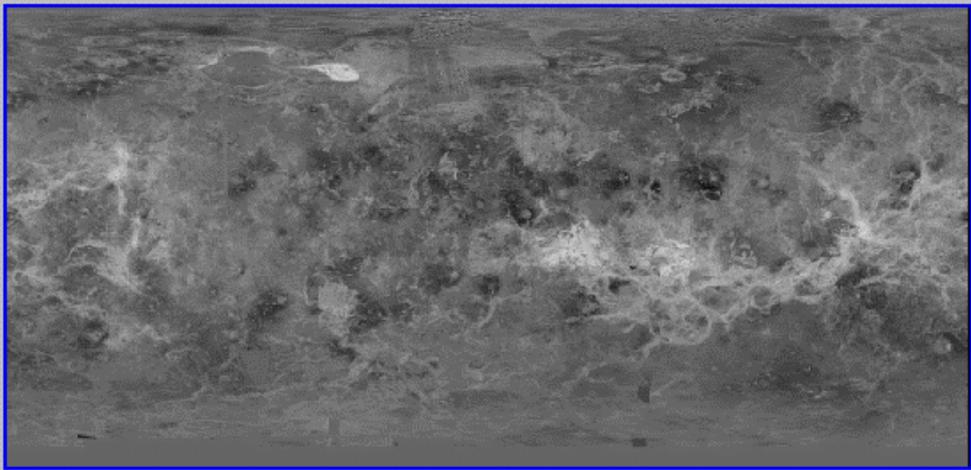
Beim Arno Projekt ist vor allem die zu den anderen Systemen vergleichsweise hohe Bandbreite von 140 Mbit/sec zu erwähnen. Es ist das einzige System, bei dem ein entsprechender Netzwerkbackbone berücksichtigt wurde, über den sogar On-line Videokonferenzen geführt werden können.

Im Schlußfeld bewegen sich der GISIS Vorläufer Spacepicture (9.25 Punkte) und der AVHRR CD Browser (4.43 Punkte). Beide Systeme weisen entschiedene Schwächen in allen untersuchten Punkten auf und sind für eine professionelle Verwendung kaum geeignet.

## 2.2 Planetarische Systeme

### 2.2.1 NASA/JPL PDS; IPA

Der IPA (Interactive Planetary Atlas) wird gemeinsam vom CalTech (California Institute of Technology), JPL (Jet Propulsion Laboratory) und dem USGS (United States Geological Survey) entwickelt. Gesponsert wird dieses Projekt von der NASA. IPA soll Such-, Zugriffs-, Analyse- und Anzeigemethoden für CD-ROM Datenprodukte zur Verfügung stellen. Darin ist auch der Zugriff auf die PDS (Planetary Data System) [NASAPDS, 94] CD-ROMs enthalten. Zum Zeitpunkt 1/1997 wurde eine Alpha Version bei CalTech getestet. Der Zugriff erfolgt mittels WWW und Netscape Interface (siehe Abbildung 2.11).



The above map is available in other sizes as well: [small](#) (46K) | [large](#) (409K)

To see a close-up of an area on the above image of Venus, click on that area or provide a latitude and longitude below.

---

Please select one of the following [data types](#):

F-MIDR (5° x 5°)     C1-MIDR (15° x 15°)  
 C2-MIDR (45° x 45°)     C3-MIDR (120° x 80°)

---

Please select a [cycle](#):

Cycle 1     Cycle 2     Cycle 3

---

You may either enter latitude and longitude coordinates below and click on the "Submit Query" button, or click on an area in the above image.

Latitude:  Longitude:

To reset the form, press this button:

**Abbildung 2.11:** *IPA WWW Query Formular* - Ein Abfrage Formular sowie es im Rahmen vom IPA per WWW und Netscape zur Verfügung gestellt wird. Hier handelt es sich um die Magellan Daten vom Planet Venus, einfach an den für die Magellan Daten individuellen Abfrage Optionen (Zyklen 1-3 und FMIDRs usw.) zu erkennen.

Seit März 1996 ist das IPA Projekt vom SSV (Solar System Visualization) Projekt abgelöst worden, Inhalt und Zweck blieben gleich [SSV, 96].

In Tabelle 2.10 ist eine kurze Übersicht, welche Daten über das PDS online verfügbar sind.

<b>Mission</b>	<b>Planet</b>	<b>Ressourcen<sup>1</sup></b>	<b>CD-ROM</b>	<b>Image Browser</b>
<b>Cassini</b>		Ja	Nein	Nein
<b>Clementine</b>	Mond	Ja	Ja	Nein
<b>Galileo</b>		Ja	Ja	Nein
<b>Magellan</b>	Venus	Ja	Ja	Ja
<b>Mariner 9</b>		Nein	Ja	Nein
<b>Mariner 10</b>		Nein	Ja	Nein
<b>Mars Pathfinder</b>	Mars	Ja	Nein	Nein
<b>Viking Lander</b>	Mars	Ja	Ja	Ja
<b>Viking Orbiter</b>	Mars	Ja	Ja	Nein
<b>Voyager</b>		Ja	Ja	Nein

**Tabelle 2.10:** Übersicht der PDS online Ressourcen

### 2.2.2 NASA/JPL PDS; PIA

Das PIA (Planetary Data Access) System soll die Suche nach speziellen NASA Satelliten Bilddaten unterstützen. Dies geschieht mit einer Top-down Hierarchie. Sucht man z.B. Daten zum Neptunmond Triton, so muß man zuerst das Planetensystem Triton auswählen um anschließend zu einer spezifischeren Seite gelangen zu können. Auf dieser Seite kann man nun die Suchkriterien verfeinern. Generell ist PIA (im Gegensatz zu IPA) dafür ausgelegt, Bildprodukte zu finden und nicht spezielle Bilder daraus. Es ist allerdings möglich, durch die verfügbaren Bilder zu blättern. Eine entscheidende Schwäche von PIA ist die oft geringe Bearbeitungsgeschwindigkeit, dies verursacht auch einen nicht zu vernachlässigenden Kostenfaktor.

### 2.2.3 Zusammenfassung und Wertung

Zusammenfassend kann gesagt werden, daß alle derzeit existierende planetarischen Suchsysteme eklatante Schwächen aufweisen und es insgesamt weitaus mehr akzeptable Lösungen für Geo-Daten als für planetarische Daten gibt. Die Gesamtwertung der vorgestellten Systeme ist in Tabelle 2.11 zusammengefaßt. Da alle Systeme von ein und derselben Organisation entwickelt wurden, ist die Bewertung auch auf eine Tabelle beschränkt. Weitere Systeme bzw. ausführlichere Beschreibungen sind der Parallel- Arbeit von Walcher zu entnehmen [Walcher, 97].

---

<sup>1</sup> Unter Ressourcen verstehen sich Missionsdaten, technische Daten des Satelliten, Aufnahmeverfahren, Art des Bildmaterials, Verfügbarkeit der Daten, etc.

Merkmalsbeschreibung	Detailbeschreibung	Detailwertung	Gesamtwertung
Graphical User Interface	Übersichtlichkeit	2	1.75
	Funktionalität	1	
	Interaktivität	1	
	Bedienungskomfort	3	
Suchmechanismen	point of interest	0	0.75
	region of interest	0	
	Überdeckung	0	
	Meta Informationen	3	
HS Netzwerk Backbone	> 1.5 Mbit/sec (T1)	0	0
	> 44.736 Mbit/sec (T3)	0	
	> 155 Mbit/sec (OC-3)	0	
Plattformunabhängigkeit	SUN	0	0
	SGI	0	
	HP	0	
	IBM	0	
	PC	0	
	Apple	0	
Client-Server Konzept		0	0
Kostenmanagement		0	0
Verteiltes Datenkonzept		3	3
Offenes System	CIP	0	0.43
	IDN mit DIF	0	
	CINTEX Interface	0	
	Z39.50	0	
	SQL	0	
	TCP/IP	3	
	SNMP	0	
Kommerziell erhältlich		0	0
Globales Management		0	0
Lokales Management		0	0
Keine spezielle Client Software notwendig		3	3
Modulares System		0	0
Systemwertung:			8.93

**Tabelle 2.11:** Gesamtwertung für das NASA PDS, das PIA und das IPA System

## 2.3 ATM Pilot-Netzwerke

Ein nicht unwesentlicher Teil dieser Arbeit beschäftigt sich mit ATM Netzwerktechnologie. Dieses Kapitel leitet zu diesem Thema über. In den nächsten vier Unterabschnitten werden die derzeit größten existierenden ATM Versuchsnetzwerke vorgestellt. Eine detaillierte Vorstellung aller bekannten ATM Testnetzwerke würde auf Grund der großen Anzahl den Rahmen dieser Arbeit sprengen. Ausführliche Informationen über alle bekannten ATM Testnetzwerke sind aber über begleitende Webseiten unter [Rehat, 96] über das Internet zu beziehen.

Die folgenden 4 Unterkapitel stellen vier nach den Kriterien Größe und Relevanz ausgesuchte ATM Versuchsnetze vor.

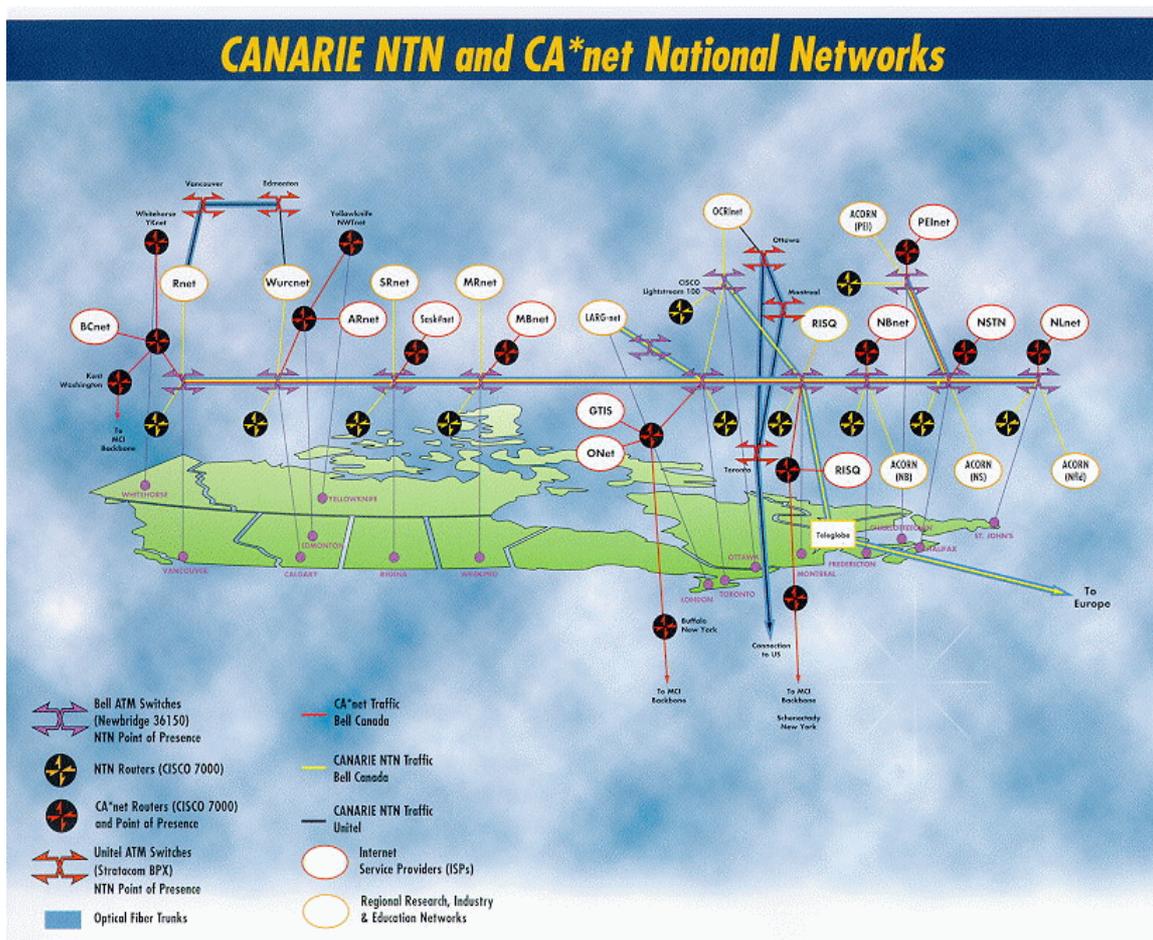
### 2.3.1 Canarie Netz

Canarie (Canadian Network for the Advancement of Research, Industry and Education) ist das derzeit größte ATM Versuchsnetzwerk der Welt [Canarie, 96]. Es erstreckt sich zwischen St. Johns, Neufundland bis nach Vancouver, British Columbia. Das ist eine Distanz von mehr als 6000 km. Über den Backbone, der in Zusammenarbeit mit dem größten Telekommunikationsunternehmen in Canada, der Bell Canada errichtet wurde, stehen sowohl DC-3 (45 Mbit/sec) als auch OC-3 (155 Mbit/sec) zur Verfügung. Außerdem gibt es noch Hochgeschwindigkeitszugänge nach Europa (durch Teleglobe) sowie eine ATM Satellitenanbindung (von Telesat Canada). Eine Darstellung des NTN (National Test Networks) ist Abbildung 2.12 zu entnehmen.

Das Projekt ist in 3 Phasen angesetzt, wobei die erste Phase im März 1995 zu Ende gegangen ist. Die erste Phase hatte drei Hauptziele:

1. Infrastruktur zur Verfügung stellen (das Canarie Netz wurde auf T1 Links aufgerüstet),
2. Aufbau eines Hochgeschwindigkeits Netzwerkes; das NTN (National Test Network) wurde ins Leben gerufen, zahlreiche Tests fanden statt.
3. Entwicklung von Applikationen - die Entwicklung von ONPS (Operational Network Products and Services) wurde propagiert und motiviert.

Die Finanzierung des Projektes geschieht zu etwa einem Viertel aus öffentlichen Mitteln sowie zu ca.  $\frac{3}{4}$  aus privaten Geldern. Bis 1999 hat die kanadische Regierung 106 Millionen \$ investiert, die mehr als 150 privaten Firmen leisteten einen Beitrag von mehr als 400 Millionen \$.



**Abbildung 2.12:** CANARIE Nationales Test Netzwerk - Zu sehen ist das nationale Backbone Netzwerk, das 11 kleinere Hochgeschwindigkeitsnetzwerke in ganz Kanada verbindet. (Quelle: <http://www.canarie.ca/eng/main.html>)

Folgende Applikationen sind in der Entwicklung:

**CanGuide.** Hinter diesem Projektnamen verbirgt sich die Entwicklung einer zweisprachigen Datenbank für das Anbieten von öffentlichen Dokumenten. Eine Bestellfunktion ist ebenfalls inkludiert. Die Datenbank wird schließlich über ein Gateway an das Internet angebunden [CanGuide, 96].

**Camel.** Der kanadische Server für mathematische Wissenschaftler stellt Services und Informationen für kanadische Mathematiker und Lehrbeauftragte [Camel, 96].

**Crafts Information Service.** Hinter diesem Projekt steht die Entwicklung einer Datenbank die Informationen über Ausrüstung, Werkzeuge, Dienste und Expertenrat für die Fahrzeugindustrie anbietet [Crafts, 96].

**CultureNet.** Die Universität von Calgary plant in Zusammenarbeit mit der „Canadian Conference of the Arts“ und dem kanadischen Institut für Theater

Technologie einen Informationsserver für kulturelle Ereignisse aufzubauen. Der Server wird unter anderem Informationen über Aufführungen, Spielpläne, Kritiken haben aber auch Kommunikationsmöglichkeiten via E-mail bzw. Konferenztools bieten [Culture, 96].

**Global Internet Seminar.** Das „Centre de Recherche informatique de Montreal“ hat ein kostenfreies halbtägiges Seminar in Englisch und Französisch zusammengestellt, um eine globale Einführung ins Internet zu geben. Um mit dem Internet vertraut zu werden, wird das Seminar über das Netzwerk angeboten [Global, 96].

**HEC/Net.** Innerhalb dieses Projektes wird eine zweisprachige (teils auch 3 sprachig mit spanisch) Datenbank aufgebaut, um einen hochqualifizierten Informationspool für internationale Management Lehre anzubieten. Dies umfaßt Posten und Austausch von Kommentaren zu Courseware, leiten von Konferenzen und Foren, On-Line Benutzung von Management Simulationsspielen [HecNet, 96].

**IEEE Canada Information Services.** Hier geht es um den Ausbau der bestehenden Fax Kommunikation zwischen der IEEE und ihren kanadischen Mitgliedern. Die Kommunikation soll in Zukunft über das Netz erfolgen, alle Mitglieder sollen stets mit den neuesten Standards und technischen Informationen versorgt werden [IEEECan, 96].

**InfoTree.** Info Tree ist das Nachfolgeprogramm zu Archie, dem berühmten Suchprogramm für FTP Server auf dem Internet. Info Tree soll einen umfassenden Index über FTP Server, einen Index über Gopher Server sowie eine hilfreiche Ansammlung von Material für Interneteinsteiger. Ein besonderes Augenmerk wurde auf das Navigieren innerhalb der InfoTree Datenstruktur gelegt, dies geschieht mit Hilfe von graphischen Symbolen [InfoTree, 96].

**Music Library of the Future.** Dieses Projekt wird neue Technologien entwickeln um auf eine Multimediale Datenbank zugreifen zu können, die aktuelle Informationen über kanadische Komponisten enthält. Die Datenbank wird außerdem ausgewählte Geschäfte, Musikaufnahmen, Photos, Graphiken, MIDI Daten, Klangbeispiele sowie technisch/analytische Beschreibungen der Arbeit enthalten [Music, 96]

**NSTN Cyber Mall.** Hiermit soll das erste existierende Programm um über das Internet einkaufen zu können erneuert und verbessert werden. Der erste Klient war noch auf Gopher basierend, der neue wird mit Navigationstools und WWW Zugang ausgerüstet [NSTN, 96].

**Ntnet.** Innerhalb dieses Projektes wird ein Netzwerk aufgebaut, das die Gemeinden in den nordwestlichen Regionen mit dem Rest der Welt verbindet [NTNet, 96].

**Promotion of Tourism through the Internet.** Im Zusammenhang mit der nordischen Ski Olympiade wurde ein Internet Informationssystem entwickelt,

welches Graphik, Audio und Textmaterial zur Verfügung stellte. Weiters stellte es auch Kommunikationskanäle für die Athleten zur Verfügung sowie Modem Anschlußmöglichkeit während der Spiele [Tourism, 96].

**Yukon Net.** Dieses Projekt initiiert den Aufbau eines Netzwerk Links für das Yukon Gebiet mit dem Rest der Welt [Yukon, 96].

**CA Domain Registration.** Die Universität von British Columbia wird einen DNS (Domain Name Server) für die Domain Kanada (ca) aufbauen und in Betrieb nehmen. Die Universität wird Namen innerhalb der Domain registrieren, warten und weitergeben. Die gesamte Koordination wird hiermit zentralisiert [CaDomain, 96].

**Science, Eh?.** Es wird eine interaktive Umgebung geschaffen, um den Information-Highway zu entdecken und kennenzulernen. Als Metapher wird ein Magazin benutzt, das jedoch über zahlreiche Multimediale Fähigkeiten besitzt. Weiters soll das Einbringen von eigenen Applikationen möglich sein [Science, 96].

Nach dem erfolgreichen Abschluß der ersten Phase ist seit April 1995 die zweite Phase begonnen worden. Die Hauptziele der Phase II sind:

- Technologie und Entwicklungsprogramm - Innovative Projekte werden finanziert,
- Das National Test Network steht zum Test von neuen Technologien und Applikationen zur Verfügung, die für zukünftige Breitbandapplikationen benötigt werden,
- Weitere Aufrüstung des Canarie Netzes,
- Rahmen Programm, welches die kanadischen Bürger auf die neuen Technologien aufmerksam machen soll bzw. Publikation der Ergebnisse.

Die 2. Phase wird im Jahre 1999 abgeschlossen sein, bis dahin wird das gesamte Canarie Projekt voraussichtlich mehr als 1 Billion \$ an wirtschaftlichen Umsatz initiiert haben sowie 20.000 Mann-Jahre an Anstellungen [Canarie, 96].

### 2.3.2 Magic Net

Das MAGIC (Multidimensional Applications and Gigabit Internetwork Consortium) Netz Projekt wurde 1993 in den USA ins Leben gerufen [Magic, 96] um sich folgenden Herausforderungen zu stellen:

- Sicherstellung der Interoperabilität zwischen heterogenen Netzwerkgeräten
- Koordination von mehreren, dasselbe Ziel habenden, Datenströmen
- Umgang mit stoßartigem (bursty) und kontinuierlichem (stream oriented) Datenverkehr
- Kompensation der Netzwerkverzögerung und Fehler Effekte durch einen hohen Datendurchsatz

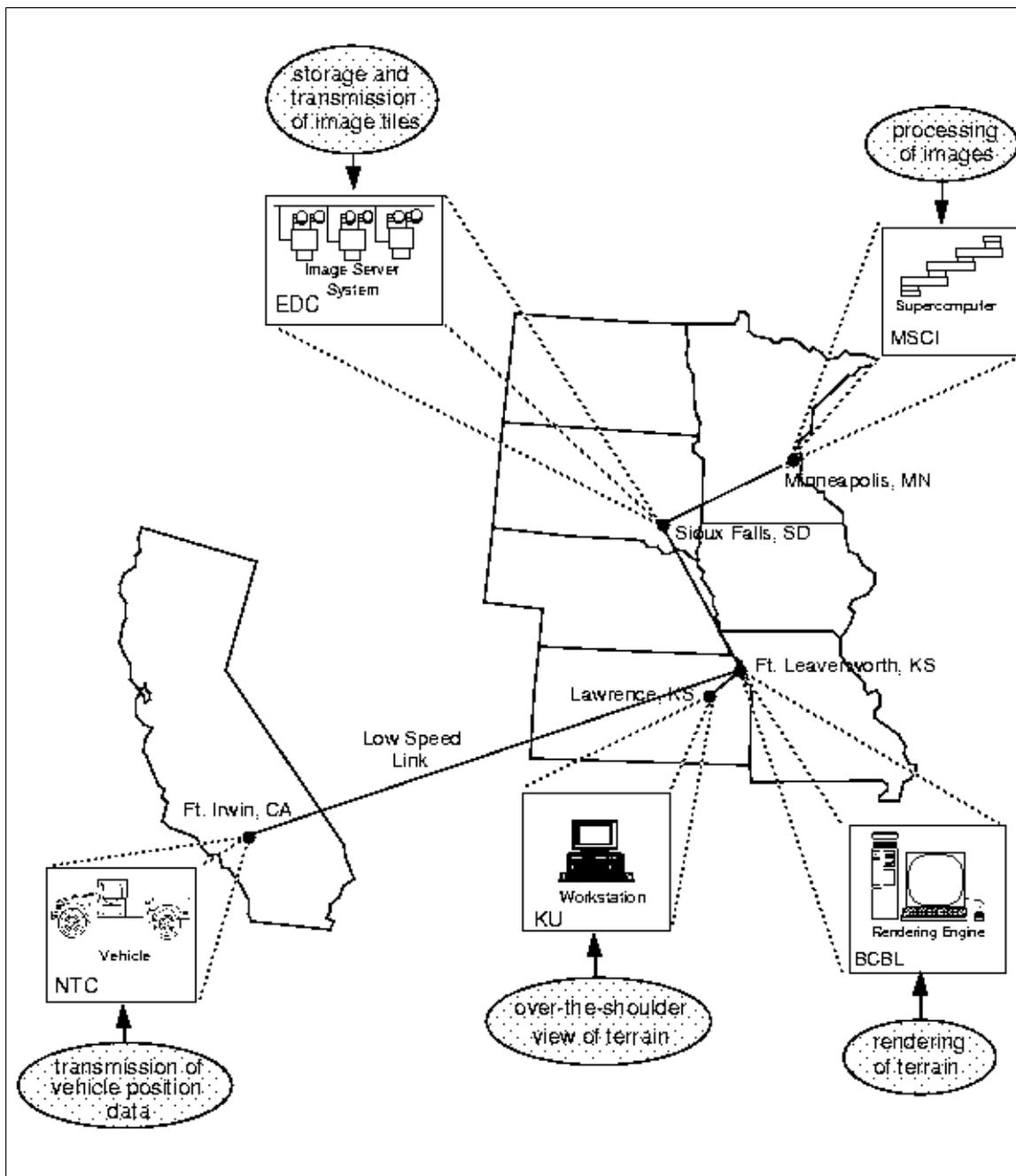
Am MAGIC Konsortium nehmen sowohl öffentliche als auch akademische und industrielle Organisationen teil. Die Teilnehmer werden auch teilweise von der ARPA (Advanced Research Projects Agency) unterstützt. Die Teilnehmer sind im einzelnen:

- Earth Resources Observation System (EROS) Data Center, U.S. Geological Survey
- Lawrence Berkley Labaratory (LBL), U.S. Department of Energy
- Minnesota Supercomputer Centre (MCSI)
- SRI International
- University of Kansas (KU)
- Army High-Performance Computing Research Centre (AHPCRC)
- Battle Commande Labaratory, U.S. Army Combined Arms Command
- Digital Equipment Corporation (DEC)
- Northern Telecom, Inc./Bell Northern Research
- Southwestern Bell Telephone
- Splitrock Telecom
- Sprint
- US West Communications Inc.

Folgende Ziele sollen mit Hilfe des Magic Netzwerkes erreicht werden:

- Aufbau eines Gigabit Netzwerkes zur Unterstützung von Entwicklung und Forschung
- Demonstration des Nutzens eines Hochgeschwindigkeits Netzwerkes für militärische Echtzeitapplikationen
- Erforschung und Lösung der Probleme die sich aus der Nutzung einer verteilten Hochgeschwindigkeits Rechner- und Speicherarchitektur ergeben

Im Prinzip wurde das Magic Netzwerk für den Test der militärischen Schlachtfeldsimulation „Terravision“ aufgebaut. Terravision ist eine Landschafts-Simulations-Software, die es erlaubt, in verschiedenen Auflösungsstufen durch einen Landstrich zu navigieren. Die Bilder werden hierzu von einem Image Server System (ISS) [Tierney, 95] geliefert, das hunderte von Kilometern von der Applikation entfernt ist. Dieser Datenverkehr kann sowohl „stream“ orientiert sein (wenn der Benutzer langsam durch die Landschaft gleitet) als auch „bursty“ (wenn er den Standort sprunghaft ändert). Zusätzlich zu den Bilddaten werden auch noch in Echtzeit sich bewegende militärische Objekte in die Szene eingeblendet. Die benötigte Bandbreite ist enorm, die Anforderungen an die ISS Architektur rigoros. Eine Darstellung des gesamten Magic Netzes samt den funktionellen Charakteristiken ist in Abbildung 2.13 zu sehen. Der aktuelle Testdatensatz ist ein Höhenmodell von Fort Irwin.



**Abbildung 2.13:** das MAGIC Netz - eine geographische und funktionelle Darstellung des Versuchsnetzwerkes für die Landschaftsdurchflugs Simulationssoftware Terravision. (Quelle: <http://www.magic.net>)

Das MAGIC Netz besteht aus vier Hochgeschwindigkeits LANs, die durch ein Gigabit WAN verbunden sind. Das gesamte Netzwerk basiert auf SONET und ATM Standard. Der Backbone weist eine Geschwindigkeit von OC-48 (2.4 Gbit/sec) auf, wobei die Interfaces jeweils den für ATM derzeit höchsten Übertragungsstandard OC-12 (622 Mbit/sec) aufweisen.

Das Projekt ist in zwei Phasen gegliedert, wobei in der ersten Phase das Netzwerk sowie ein Prototyp der Geländevisualisierungssoftware aufgebaut wird. In der zweiten Phase werden dann durchgreifende Leistungs Verbesserungen durchgeführt. Weiters ist geplant die Anzahl der ISS Server zu erhöhen. Folgende Forschungsziele sollen innerhalb des Projektes durchgeführt werden [Magic, 96]:

- Erstellung einer interaktiven Landschaftsvisualisierungssoftware,
- Design der Architektur eines parallelen verteilten Speichersystems mit Hochgeschwindigkeitsnetzzugang,
- Test und Überwachung des ATM Netzes,
- Entwicklung eines Gigabit LAN/WAN Gateways mit dynamischer Bandbreitenallozierung,
- Steigerung der Netzwerk Performance,
- Entwicklung von Hochgeschwindigkeitsprotokollen für den Transport von Bilddaten.

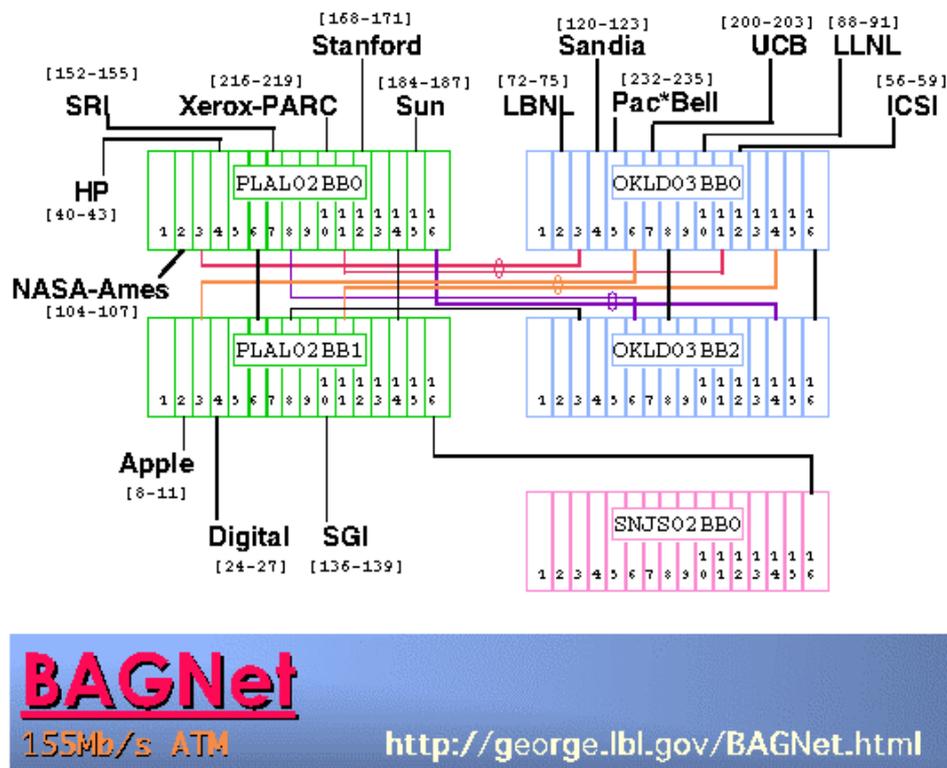
Phase 1 des Projektes ist erfolgreich beendet worden, ein Prototyp von Terravision läuft. Ergebnisse bzw. Videos sind auf dem WWW zu finden [Magic, 96]. Die zweite Phase ist zum Stichtag 1/1997 im Gange.

### 2.3.3 Bay Area Gigabit Netzwerk

Das im San Francisco (Kalifornien/USA) Bay angelegte BAGNet (Bay Area Gigabit Network) wurde zu Beginn 1994 von Pacific Bell in Betrieb genommen [BagNet, 96]. Die Projektdauer war auf 2 Jahre begrenzt, die kostenlose Nutzung des Testnetzwerkes wurde im Mai 1996 eingestellt. Das Netz ist auf dem OC-3 (155 Mbit/sec) Standard aufgebaut und benutzte IP über ATM. Weiters verfügte BAGNet über eine JPEG komprimierte multicast Videoübertragungsmöglichkeit. BAGNet hatte insgesamt 15 Teilnehmer aus dem wissenschaftlichen Bereich sowie aus der Industrie. Eine schematische Darstellung des Netzwerkes ist in Abbildung 2.14 zu finden.

Die Versuchsteilnehmer von BAGNet waren im einzelnen [BagNet1, 96]:

- Apple Computer , Advanced Technology Group
- Digital Equipment Corporation, Palo Alto Systems Research Center, BAGNet
- Hewlett-Packard Laboratories ,HPL Internaut Web Server
- International Computer Science Institute



**Abbildung 2.14:** BAGNet - Schematische Darstellung des vollvermaschten Bay Area Gigabit Network im San Francisco Bay. BAGNet hat mit OC-3 (155 Mbit/sec) Links insgesamt 15 Teilnehmer aus Industrie und Forschung verbunden. Im May 1996 wurde der kostenlose Testbetrieb eingestellt.

- Lawrence Berkeley National Laboratory, Imaging and Distributed Computing Group, LBNL BAGNet Status Page
- Lawrence Livermore National Laboratory, LLNL BAGNet Page
- NASA Ames Research Center, Ames BAGNet Page
- Pacific Bell , Broadband Strategy and Product Development Group
- Sandia National Laboratories - California, BAGNet Page
- Silicon Graphics, Inc.
- SRI International, SRI BAGNet Page
- Stanford University
- Sun Microsystems, Inc.

- University of California, Berkeley , Computer Science Dept. - Tenet Group
- Xerox Palo Alto Research Center , PARC BAGNet Page

Herausragende Merkmale des BAGNet waren [BagNet1, 96]:

- Multicast Unterstützung von IP über ATM
- Hochgeschwindigkeits JPEG komprimierte Video Übertragung (320x240 bei 30 Bildern/sec) über Multicast
- Verbindung von heterogenen Hardware Komponenten (es wurden insgesamt Switches von 5 verschiedenen Herstellern, sowie Hostadapter und Treibersoftware von 8 verschiedenen Herstellern verwendet).
- Das Netz wurde mit PVCs (Permanent Virtual Circuits) vollvermascht. Insgesamt wurden 60 Hosts an 15 Orten verbunden. Dies entspricht  $(60 * 59)/2 = 1770$  unidirektionalen Verbindungen. Insgesamt wurden doppelt so viele aufgebaut, da jeder der 60 Hosts eine voll duplex Verbindung zu jedem anderen hatte.

Folgende Applikationen und Tests wurden in den 2 Jahren durchgeführt:

**Teleseminar.** Die Hauptanwendung des BAGNet lag in der Übertragung von hochqualitativen Videoseminaren. Stanford übertrug 2 Semester lang jede Woche ein Seminar, Apple, UCB and Xerox-PARC haben auch nach Beendigung der Testphase mit dem Senden von Videoseminaren fortgesetzt. Der Video Strom hatte 320x240 Pixel pro Bild Auflösung bei einer Bild Rate von 20-24 Bilder/sec. Die benötigte Bandbreite betrug 1 - 2,5 Mbit/sec. Der PCM (Pulse Code Modulation) kodierte Audio Teil benötigte 78 kb/sec an Bandbreite.

**ATM Verkehrsuntersuchungen.** Pacific Bell und Bellcore waren äußerst an der Auslastung des Netzwerkes interessiert und haben zahlreiche Analysen durchgeführt. Alleine zwischen dem 11. September 1995 und dem 6. Oktober 1995 wurden 400 GByte an Daten über das Netz geschickt.

**Vereinigung von örtlich getrennten wissenschaftlichen Arbeiten.** RIACS/NASA und Sandia haben Programme entwickelt um konferenzmäßige Echtzeit Kommunikation zwischen örtlich getrennt arbeitenden Wissenschaftlern zu ermöglichen.

**Fernlernkurse.** Das Zentrum für Telekommunikation in Stanford beschäftigte sich mit dem zur Zeit sehr heiß diskutierten Aspekt des Asynchronous Distance Education Project (ADEPT). Ziel ist es einen On-demand Service für den Zugang zu Lehrveranstaltungen sowie zu Studenten auf dem Campus aufzubauen. Derzeit ist die Quote zwischen Studenten die im Kampusgelände bzw. zu Hause sind 50/50. Eine solche Applikation würde also großen Nutzen versprechen, denkt man vor allem an die gesparten Ressourcen.

**Hochgeschwindigkeits Datenverkehr Protokoll.** RIACS/NASA-Ames entwickelt ein neues Datentransfer Protokoll für den Transport von großen Bilddaten. Das Protokoll basiert auf UDP, da Bildübertragung im Prinzip gegenüber einer kleinen Bit-Fehlerrate sehr tolerant ist. Die Implementierung ist darauf ausgelegt, Übertragungsfehler kontrolliert zu managen (Definition eines Quality of Service), da die Fehlertoleranz in übertragenen Bildern natürlich Applikationsabhängig ist. So werden in medizinischen Applikationen bestimmt keine Fehler geduldet, bei der Übertragung von WWW Bildern aber sehr wohl.

**Terravision - Interaktive Landschafts Visualisierung.** Die im Kapitel Magic Net beschriebene Terravision Applikation samt dem Berkley ISS (Image Server System) wurde auch erfolgreich im BAGNet getestet.

**Performance Messungen.** Es wurden mit Hilfe des netperf<sup>1</sup> Tools von HP Performance Messungen am Netz durchgeführt. Zahlreiche Tests wie Multicast Performance, UDP/IP Transaktionen (zur Messung von NFS Performance), TCP/IP Performance und UDP/IP Performance wurden durchgeführt. Für den TCP/IP Durchsatz wurden nochmals getrennt eine große Anzahl von Experimenten durchgeführt.

**Öffentliches ATM Netzwerk für ein medizinisches Bildverarbeitungssystem.** Es wurde ein System zur Speicherung, Analyse, Verteilung und Lieferung von medizinischen Bildern und Videosequenzen entwickelt. Neben dem reinen Datentransport hat man sich auch mit Dingen wie flexible verteilte Speichermöglichkeiten und Zugriff auf Analyse Systeme unabhängig vom Standort beschäftigt.

**Remote Zugriff auf den NASA Wind Tunnel.** NASA Ames experimentierte mit dem Hochgeschwindigkeitsnetz um Remote Kontrolle über Wind Tunnel Experimente durchführen zu können. Dies ermöglicht Wissenschaftler mit Netzzugang eine Überwachung der Experimente ohne physisch präsent zu sein.

BAGNet wurde nach den 2 Jahren Projektdauer aus Mangel an Endbenutzer Interesse eingestellt. Insgesamt hat der 2 jährige Versuch jedoch eine Menge an neuen Erfahrungen für die 15 beteiligten Unternehmen gebracht, alle Ergebnisse wurden veröffentlicht und sind unter [BagNet, 96] am WWW zu finden.

### 2.3.4 Europäischer ATM Pilotversuch

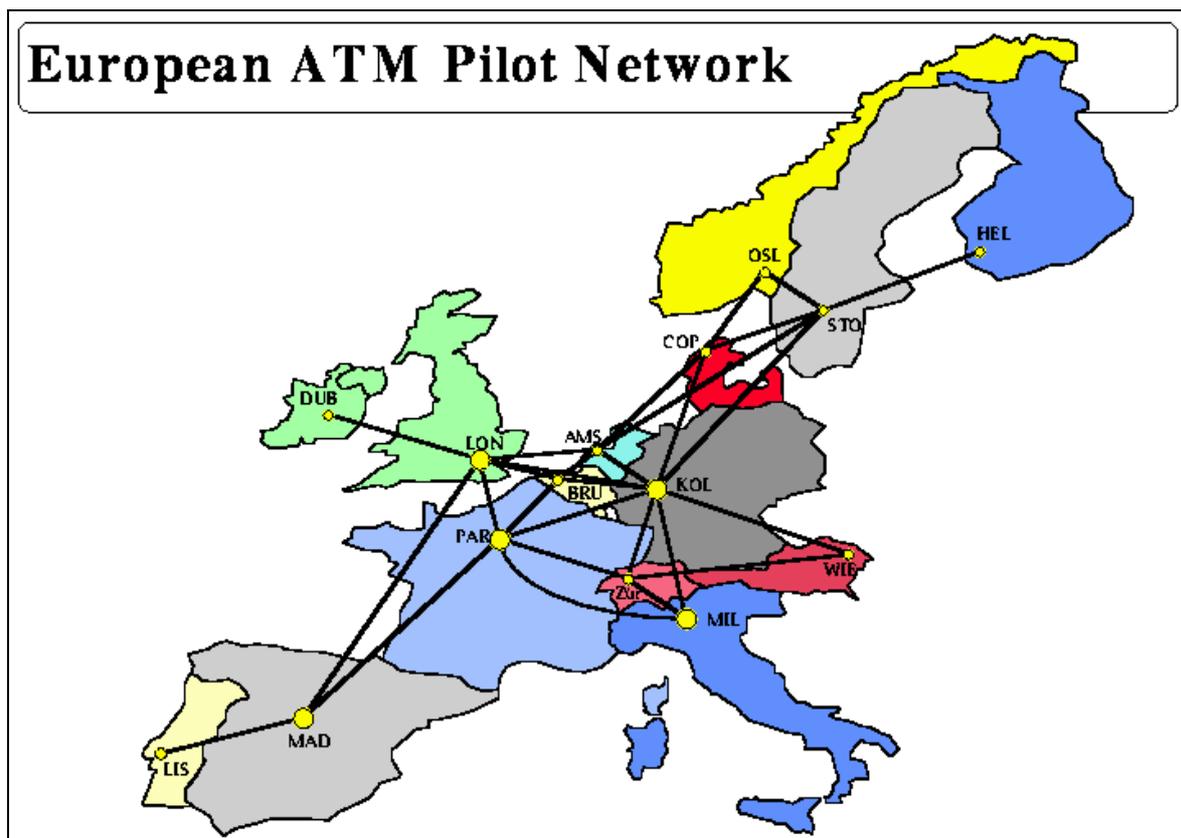
Sechzehn nationale Telekommunikationsunternehmen haben sich im Juli 1994 vorerst für ein Jahr zusammengeschlossen, um mit Hilfe der ATM Technologie einen europäischen Super Information Highway aufzubauen [Europ, 95].

---

<sup>1</sup> Dieses Tool wird auch in dieser Arbeit für Performance Messungen am lokalen TUG ATM Cluster verwendet

Hauptzweck des Versuches ist das Testen der neu angekündigten ETSI (European Telecommunication Standard Institution) und ITU-T (International Telecommunication Union) Standards auf Interoperabilität. Sobald die Operation der Standards bewiesen ist, kann mit der Entwicklung von heterogener ATM Hard- und Software begonnen werden. Dies ist dann der Grundstein für kommerziell anbotbare ATM Dienste der einzelnen Telekom Unternehmen. Das aktuelle ATM Pilotversuchsnetzwerk besteht aus einem nationalen Knoten in jedem teilnehmenden Land. Die Verbindung erfolgt derzeit über E3 (34 Mbit/sec) PDH (plesiosynchrone digitale Hierarchie) Links. Jeder Knoten muß die VP (virtual path) Crossconnect Standards der ITU-T sowie die ETSI Standards unterstützen. Weiters muß jeder Knoten zumindest 256 VPs pro Port unterstützen. Die einzelnen Verbindungen sind Abbildung 2.15 zu entnehmen.

Die Teilnahme von anderen Organisationen (Universitäten, Forschungseinrichtungen, Regierung und private Firmen) wird mit Hilfe von EU Projekten unterstützt. Die Bewerbung erfolgt auf nationaler Ebene über sogenannte „National Host“ Projekte. Der National Host ist eine Vereinigung, die zwar nicht direkt Geldmittel zur Verfügung stellen kann, dafür aber in sehr rascher Art und



**Abbildung 2.15:** *Europäisches Versuchsnetzwerk* - Die Abbildung zeigt den ATM Versuchsaufbau, der im Herbst 1994 in Betrieb genommen wurde. Alle Verbindungen basieren auf dem E3 (34 Mbit/sec) Standard. Insgesamt haben sich 16 Telekommunikationsunternehmen aus ganz Europa an dem Versuch beteiligt.

Weise mit den richtigen Leuten in Brüssel Kontakt aufnehmen kann und eine Finanzierung organisieren kann. Eine genaue Übersicht der einzelnen National Host Projekte sowie deren Aktivitäten ist unter [Rehat1, 96] zu finden. Eine genaue Beschreibung aller einzelnen Projekte würde den Rahmen dieser Arbeit sprengen.

Das Versuchsnetzwerk stellt a priori folgende Dienste zur Verfügung:

- SMDS (Switched Multimegabit Data Service) zum Austausch von Daten mit variabler Paketlänge sowie CBDS (Connectionless Broadband Data Service), das ETSI Äquivalent zu SMDS,
- Frame Relay stellt PVCs (Permanent Virtual Circuit), die hauptsächlich für LAN Verbindungen gebraucht werden, zur Verfügung,
- CBR (Constant Bit Rate) zur Emulation von verbindungsorientierten Telefonleitungen. Es werden u.a. 2 Mbit/sec, G 703 und Nx64 kbit/sec zur Verfügung stehen.

Im Jahre 1994 wurden 11 Projekte von der EU akzeptiert und in 2 Phasen analysiert bzw. implementiert [Rankin, 96]. Die Projekte waren:

- MEDIATOR, Vergleich von Videos, Bildern, Text und Graphiken um schnell Zeitungskolumnen produzieren zu können
- VISINET, 3D Visualisierung über Netzwerke unter Benutzung von Remote Präsentationen und Video Konferenzen
- FASHION-NET, Audiovisuelle und multimediale Dienste in Zusammenhang mit der Modeindustrie
- HIM, medizinische Informationsverarbeitung und Bildbearbeitung
- RETAIN, Teleworking mit medizinischen Bildern für lebensrettende Sofortmaßnahmen
- STEN, Wissenschaftliches transeuropäisches Netzwerk
- ARTLINE
- CITY TEN, Koordinierung von Verkehrsinformationen, Bildung und Videokommunikation
- E=MC2, Kooperation zwischen den Supercomputer Zentren in Europa
- IDEA, kooperative Zusammenarbeit unter Benutzung von shared Audio, CAD und Video
- M-NET, Touristinformationssystem und europaweite Koordination

Die erste Phase wurde im Mai 1995 abgeschlossen, die Ergebnisse rechtfertigen einen Einsatz eines Hochgeschwindigkeits ATM Backbone. So wurde u.a.

festgestellt, daß Interaktivität mit den gegenwärtigen Netzwerkverbindungen nur unter äußerst günstigen Umständen gewährleistet werden kann. WAN Anwendungen im verteilten Rechnerbereich erfordern in jedem Falle ein Hochgeschwindigkeitsnetzwerk - nur so kann man die verteilten Rechner als einen „Metacomputer“ sehen [Rankin, 96]. Der Start der zweiten Phase erfolgte im Herbst 1995 mit noch größerer Beteiligung. Diesmal schlossen sich 18 europäische Netzbetreiber zur Realisierung von „Joint ATM Experiments on European Services“ (JAMES) zusammen um eine Entwicklungs- und Testplattform für Europäische Breitbandtelekommunikationsprojekte zu schaffen. Hier sind Projekte wie National Host, TEN34 und G7-Ministerkonferenzen beteiligt. Die österreichische PTA ist an dem 2 jährigen Projekt mit einem Gesamtvolumen von 111 Mannmonaten beteiligt (insgesamt stehen 2048 zur Verfügung) und 14,5 Millionen ECU (~19,3 Millionen Schilling). Dies ist bis zum vorläufigen Vertragsende am 1.4.1997. Für die Verlängerung<sup>1</sup> stehen der PTA weitere 596 kECU (~7,87 Millionen Schilling) zu. Somit ist der weitere ATM Infrastrukturausbau in Österreich gewährleistet.

Im Rahmen des National Host - Österreich wird auch das DIANE (DIstributed ANnotation Environment) [DIANE, 96] ACTS EU Projekt abgewickelt. DIANE soll in der Endphase einer verteilten Benutzergemeinschaft ein Werkzeug zum einfachen Annotieren von Multimedialen Daten zur Verfügung stellen. In diesem Projekt ist auch das ICG beteiligt, die ATM Karte der SGI Power Challenge wurde dadurch finanziert. Pilotversuche mit dem X-wedge Application Sharing Tool werden im Kapitel 6 beschrieben.

Speziell von der Post und Telekom Austria (PTA) wurden anlässlich der Ars Electronica 1996 in Linz ATM Pilotversuche durchgeführt. Diese umfaßten:

- Übertragung von Live Sendungen des ORF,
- Übertragung von Filmmaterial zur Nachbearbeitung und Archivierung,
- interaktive Multimedia- und Filmapplikationen (Video on Demand).

Die PTA steuerte für diese Versuche die Infrastruktur, bestehend aus Glasfaserkabel, ATM Knoten und entsprechende B-ISDN Dienste bei.

Im Zuge eines Betriebsversuches mit Billa und Libro wurden vorerst 20 Stück 2 Mbit/sec ATM Anschlüsse in Wien und Wienumgebung zwischen Billa und Libro Filialen im Juli 1996 hergestellt und in Betrieb genommen. Dieser Pilotversuch soll Kunden die Möglichkeit geben, Videoclips von Billa Filialen aus abzurufen und in der Folge gleich zu bestellen.

---

<sup>1</sup> Projektdauer ist 2 Jahre

### 2.3.5 Zusammenfassung und Diskussion

Das derzeit größte nationale Testnetzwerk ist das kanadische CANARIE Netz mit seinen mehr als 150 zahlenden Mitgliedern sowie 15 Großprojekten. Die Ausdehnung des Gesamtnetzwerkes von mehr als 6000 km beweist letztendlich auch die WAN Fähigkeit des ATM Standards.

Das von militärischen Einrichtungen ins Leben gerufene MAGIC Netz konzentriert sich auf eine verteilte Applikation Terravision. Der Schlachtfeldsimulator wurde verteilt aufgebaut, wobei das ISS (Image Server System) im EROS Datenzentrum installiert ist, die Prozessierung der Bilder im Minnesota Supercomputer Zentrum abläuft, die Visualisierung hingegen im LBL (Lawrence Berkley Labaratory) stattfindet. Einzigartig am MAGIC Netz ist der Gigabit Netzwerk Backbone. Er setzt sich aus vier OC-12 (622 Mbit/sec) Links zusammen, was einer Gesamtbandbreite von OC-48 (2,4 Gbit/sec) entspricht. Die größte überbrückte Distanz macht mehr als 1100 km aus.

Das auf 2 Jahre beschränkt gewesene BAGNet Projekt hat sich hauptsächlich mit Videokonferenzsystemen und der Übertragung von Seminaren beschäftigt. Die insgesamt 15 Teilnehmer, hauptsächlich aus der Industrie, konnten während der 2 Jahre wertvolle Erfahrungen über Installation, Betrieb und Interoperabilität zwischen heterogenen Switch Herstellern sammeln. Das Versuch ging im Frühling dieses Jahres zu Ende, Ergebnisse und Erfahrungen wurden publiziert.

Der europäische ATM Versuch wurde im Herbst 1994 gestartet. 16 Telekom Unternehmen haben sich zusammengeschlossen um primär die Interoperabilität von ITU-T und ETSI Standards zu testen. Zusätzlich wurden auf nationaler Ebene sogenannte „National Host“ Projekte gestartet, um auch anderen interessierten Institutionen (öffentlicher und privater Art) den Zugang zum Netz zu ermöglichen sowie die Entwicklung von Applikationen mittels EU Projekten zu fördern.

Diese vier ATM Versuchsnetzwerke waren nur einige relevante Beispiele. Weltweit ist eine Unzahl von Versuchsnetzwerken im Aufbau. Erwähnung verdienen auch die Netzwerke in Hongkong und Thailand, wo ebenfalls eifrige Aktivitäten zu beobachten sind<sup>1</sup>.

Zusammenfassend kann man sagen, daß die durchgeführten Experimente sowie die entwickelten Applikationen durchaus die Verwendung von ATM als ein weltweites Backbone Netzwerk rechtfertigen. Die präsentierten Resultate lassen darauf schließen, daß ATM nach Abschluß der ITU-T Standardisierung die integrierende Netzwerktechnologie der Zukunft wird.

---

<sup>1</sup> siehe hierzu [Rehat, 96]

## 2.4 Literatur und Referenzen

- [BagNet, 96] BAGNet, Bay Area Gigabit Network at San Francisco Bay, Homepage and papers, Juli 1996, <URL: <http://george.lbl.gov/BAGNet.html>>
- [BagNet1, 96] Wiltzius D., Berc L., Devadhar S. „BAGNet - Experiences with an ATM metropolitan-area network“, ConneXions, Vol. 10 No. 3, März 1996, <URL: <http://www.llnl.gov/bagnet/article.html>>
- [CaDomain, 96] CA Domain Registration Projekt/Canarie Netz, Kontakt Person: Hubert Lai, Tel: (604) 822-8580
- [Canarie, 96] Canarie, Canadian Network for the Advancement of Research, Industry and Education, Homepage, Juli 1996, <URL: <http://www.canarie.ca/eng/main.html>>
- [Camel, 96] Camel Projekt/Canarie Netz, Kontakt Person: H. Campbell, email: eddy@mast.queensu.ca, Tel: (613) 545-2390 (0) 2964, homepage, <URL: <http://camel.cecm.sfu.ca/home.html>>
- [CanGuide, 96] CanGuide Projekt/Canarie Netz, Kontakt Person: Ruth Marks, email: rmarks@oise.on.ca, Tel: (416) 923 6441 X 2380
- [CEO, 94] ESA/ESRIN EEOS Workshop on „European Data Networks and Earth Observation User Information Services“, Marino (Rome), 13-15 December 1994, CEO/130/1994 ESRIN EEOS-WS1-001-FP-1.0
- [CEO, 95] CEO (Center of Earth Observation), Homepage, Dez. 1996, <URL: <http://ceo-www.jrc.it>>
- [Crafts, 96] Crafts Projekt/Canarie Netz, Kontakt Person: Peter Rukavina, email: peter@crafts-council.pe.ca, Tel: (902) 566-1584
- [Culture, 96] CultureNet Projekt/Canarie Netz, Kontakt Person: Ken Hewitt, email: kdhe Witt@acs.ucalgary.ca, Tel: (403) 220-4900
- [DIANE, 96] DIANE (Distributet Annotation Environment) über ein ATM Netzwerk, Homepage, Jän 1997 <URL: <http://www.vcpc.univie.ac.at/activities/projects/DIANE.html>>
- [EOSDIS, 96] Nasa, Earth Observing System Data Information System. Homepage, Jän. 1997, <URL: [http://www-v0ims.gsfc.nasa.gov/v0ims/eosdis\\_home.html](http://www-v0ims.gsfc.nasa.gov/v0ims/eosdis_home.html)>
- [EOSDIS/B,96] Nasa, Earth Observing System Data Information System Release B, Homepage, Dez. 1996, <URL: <http://edhs1.gsfc.nasa.gov/>>
- [Europ, 95] European ATM Trial, Homepage, Juli 1996, <URL: <http://www.com21.com/pages/eap.html>>
- [EWSE, 95] The European Wide Service Exchange, <http://ewse.ceo.org>
- [Global, 96] Global Internet Seminar Projekt/Canarie Netz, Kontakt Person: Didier Clerc, email: cleric@crim.ca, Tel: (514) 398-2660
- [HecNet, 96] HecNet Projekt/Canarie Netz, Kontakt Person: Guy Stanley, Tel: (514) 340-6188
- [IDN, 93] IDN (International Directory Network), Homepage, Dez. 1996, <URL: <http://gcmd.gsfc.nasa.gov/ceos-idn.html>>
- [IEEECan, 96] IEEE Canada Information Services Projekt/Canarie Netz, Kontakt Person: Dr. J. Chrostowski, email: Chrostowski@iit.nrc.ca, Tel: (613) 993-7908
- [ImageNet, 94] CORE Software Technology, Inc., „ImageNet“, Homepage, Dez. 1994, <URL: <http://coresw.com>>

- [InfoTree, 96] InfoTree Projekt/Canarie Netz, Kontakt Person: Peter Deutsch, email: peterd@bunyip.com, Tel: (514) 875-8611
- [Johnston, 95] Tierney B., Johnston W., Feuquay J., Butzer T., „A Distributed Parallel Storage Architecture and its Potential Application within EOSDIS“, NASA Mass Storage Symposium, March 1995
- [Kirste, 93] Kirste T. „Spacepicture - An interactive Hypermedia satellite image archival system“, ZGDV - Darmstadt, 1993
- [Lotz-Iwen, 95] Lotz-Iwen H.J., "Earth Observation User Information Services in Germany"; EEOS Workshop on European data networks and earth observation user information services; proceedings; Document refs: CEO/130/1994; March 1995, pp 359-363.
- [Magic, 96] Magic (Multidimensional Applications and Gigabit Internetwork Consortium), Homepage, Dez. 1996, <URL: <http://www.magic.net>>
- [MTPE, 96] Nasa, Mission to Planet Earth, Homepage, Dez. 1996, <URL: <http://www.hq.nasa.gov/office/mtpe/>>
- [Mungo, 94] Mungo F., Arino O., „AVHRR CD-Browser Ionia 1km net browser“, proceedings of the 6th international symposium on physical measurements and signatures in remote sensing, ISPRS, Val d'Isere, 17-21 January, 1994
- [Music, 96] Music Library of the Future Projekt/Canarie Netz, Kontakt Person: Dr. Bruce Pennycook, email: brp@sound.music.mcgill.ca, Tel: (514) 398-4548 X 5653
- [NASAPDS, 94] Nasa, Planetary Data System, Central Node, Homepage, Dez. 1996, <URL: <http://pds.jpl.nasa.gov>>
- [Nativi, 95] Nativi S., Giuli D., Pellegrini P.F. „A distributed multimedia information system designed for the Arno Project“, ISPRS Journal, Vol. 50 Nr.1, 1995, pp. 12-22
- [NSTN, 96] NSTN Cyber Mail Projekt/Canarie Netz, Kontakt Person: Michael Martineau, email: martinea@hawk.nstn.ns.ca, Tel: (902) 481-6786
- [NTNet, 96] NTNet Projekt/Canarie Netz, Kontakt Person: Jim France, Tel: (403) 873 7613
- [Rankin, 96] Rankin R., Sage P. „Experiences using the European ATM Pilot Network“, Proc. of High Performance Computing and Networking, Brussels, April 1996, pp. 441-446
- [Rehat, 96] Rehatschek H., ATM Webseiten, Informationen-Papers-Hersteller-Software, Jan. 1997, <URL: <http://www.ICG.tu-graz.ac.at/herwig/Research/ATM/ATM.html>>
- [Rehat1, 96] Rehatschek H., ATM Webseiten, Informationen über Gigabit Testnetzwerke auf der ganzen Welt, <URL: <http://www.ICG.tu-graz.ac.at/herwig/Research/ATM/testbedsites.html>>
- [Ryan, 94] Ryan P. „An Overview of the EOSDIS V0 Information Management System: Lessons Learned from the Implementation of a Distributed Data System“, Proceedings of the Scientific Information Management & Data Compression Workshop
- [Science, 96] Science, Eh? Projekt/Canarie Netz, Kontakt Person: Brent Sauder, Tel: (604) 689-0551
- [Simpson, 95] Simpson G., „International Catalogue Developments“, EEOS Catalogue Workshop, VO/1115/116/SB, 5. - 6. Dez. 1995
- [Snyder, 93] Snyder W., „Visual Interface for Space and Terrestrial Analysis“, NASA book publication of the AGU Special Session on Visualization, May 1993
- [Snyder, 94] Snyder W. „Metadata management and the VISTA System“, HICSS-27 Conference of System Sciences, DSS/Knowledge-Based Systems, Vol.III, 418-427, Jan.1994

- [SQL/3, 96] SQL/3 Standardisierungs Übersicht, Dez. 1996, homepage,  
<URL: [http://www.jcc.com/sql\\_std.html](http://www.jcc.com/sql_std.html)>
- [SSV, 96] Solar System Visualization, SSV NASA/JPL homepage,  
<URL: [http://www-ssv.jpl.nasa.gov/~srl/SSV7SSV\\_home.html](http://www-ssv.jpl.nasa.gov/~srl/SSV7SSV_home.html)>
- [Tierney, 95] Tierney B., Johnston W., Chen L., Herzog H., Hoo G., Jin G., Lee J. "Using High Speed Networks to enable Distributed Parallel Image Server Systems". Proc. Supercomputing 94, Nov. 94, LBL-35437,  
<URL: <http://www-itg.lbl.gov/ISS/papers/ISS-paper.LBL-report.fm.html>>
- [Tourism, 96] Promotion of Tourism through the Internet Projekt/Canarie Netz, Kontakt Person: Don Waton, Tel: (807) 343-8354
- [Walcher, 97] Walcher W., Design Aspects of Information Systems for Planetary Image Data. PhD thesis, Institute for Computer Graphics and Vision, University of Technology, Graz-Austria, Jan. 1997
- [Yukon, 96] Yukon Projekt/Canarie Netz, Kontakt Person: Jim Tousignant, Tel: (403) 667-5383

## 3. Kapitel

### GDSS Systementwurf

*„Ich will nicht sagen, es sei unmöglich,  
der Maschine intuitive Fähigkeiten zu geben,  
doch es wäre einfach unwirtschaftlich,  
sie auf etwas anzusetzen, was der Mensch  
viel besser kann.“*

Norbert Wiener

Im folgenden wird die Frage „**Was** soll das System leisten?“ bearbeitet. Dies ist in erster Linie eine Anwender-orientierte Sichtweise. Eine exakte Definition der Funktionalität und eine prinzipielle Systemarchitektur sind festzulegen. Die Klärung der Frage „**Wie** soll das System die geforderten Funktionen erfüllen?“ wird im 5. Kapitel (Implementierung) exemplarisch mit der Umsetzung der Konzepte in einen Prototypen behandelt. Das Ergebnis ist ein Laborversuchssystem mit der Bezeichnung GDSS (Graz Distributed Server System).

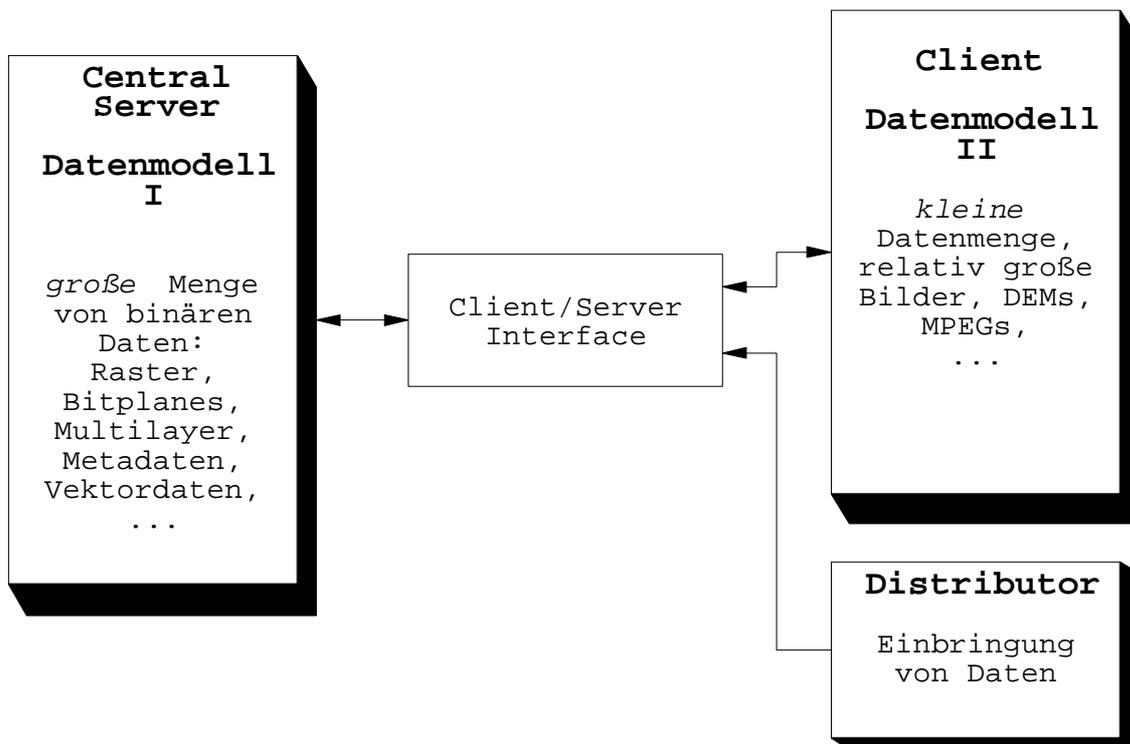
#### 3.1 Funktionalität

Das angestrebte Ziel ist die für den Benutzer transparente Integration vorhandener Bilddaten über ein Netzwerk. Die Daten sollten primär geographisch gespeichert werden, die Bilddatenbank muß solcherart optimal sein, daß der Klient Zugang zu den Daten weitestgehend interaktiv<sup>1</sup> ablaufen kann. Als Testdaten sollen die Magellandaten der Venus dienen. Hierbei gilt es, für bereits vorhandene Datenbanken ein leistungsfähiges Schema zu erstellen, bzw. einen Fernzugriff (remote access) zu realisieren. Grundsätzlich soll jede am Markt befindliche netzwerk- und multiuserfähige relationale Datenbank, die Standard ANSI SQL unterstützt, in GDSS integrierbar sein.

In dieser Arbeit wird nur die logische Speicherung der Daten berücksichtigt, die physikalische Speicherung (samt Integrität) ist Aufgabe des jeweiligen Datenbank Managementsystems (DBMS). Es soll keine neue Datenbanksoftware entwickelt, sondern eine bereits vorhandene verwendet werden.

---

<sup>1</sup> sofern ein entsprechender Netzwerkbackbone zugrunde liegt



**Abbildung 3.1:** *Verschiedene Datenmodelle von GDSS* - In der linken Hälfte müssen Gigabytes an heterogenen Daten effizient gespeichert werden, so daß eine Suche nach Überdeckung, Interessenspunkt und -region rasch durchgeführt werden kann. In der rechten Hälfte ist die Klienten Seite dargestellt, die sich zwar mit relativ wenig verschiedenen Daten, dafür aber u.U. mit der Speicherung und dem Zugriff auf sehr großen Bilddaten auseinandersetzen muß. Beide Ansätze erfordern eine gesonderte Behandlung durch ein geeignetes DBMS.

In bezug auf die Verwaltung einer großen Bilddatenmenge geht es um eine optimale Abstimmung der beiden Pole: Effizienz bezüglich Geschwindigkeit  $\Leftrightarrow$  Effizienz bezüglich Speicherung. Einerseits möchte man einen möglichst schnellen Zugriff erreichen (somit ist es z.B. nötig, ein und dasselbe Bild in einer Art Auflösungspyramide mehrfach zu speichern, um den Zeitaufwand für einen interaktiven Suchvorgang mit Zooming gering zu halten), andererseits aber so wenig Speicherplatz wie möglich verbrauchen (es wird dazu verlustbehaftete Kompression verwendet).

Um die beiden Pole in optimaler Weise abzustimmen, sollen im GDSS zwei Datenmodelle verwendet werden. Das erste Datenmodell besteht im zentralen Server, wo es darum geht, eine verhältnismäßig große Bilddatenmenge effizient zu verwalten und den Zugriff darauf nach gewissen Kriterien zu optimieren. Das zweite Datenmodell besteht auf der Klienten Seite, wo es wenige aber u.U. relativ große Bilder zu verwalten gilt (bzw. auch andere Daten wie DEMs, MPEGs usw.).

Zwischen dem Klienten und dem Server muß es eine Konvertierungsschicht geben, die die Daten in der jeweils gewünschten Form rasch an den Klienten bereit stellt bzw. durch die man letzten Endes auch Daten in den zentralen Server einbringen kann<sup>1</sup> (siehe hierzu auch Abbildung 3.1).

Der Hauptaugenmerk dieses (Teil-) Projektes liegt auf dem zentralen Server, das Klienten Datenmodell wird in einem Parallelprojekt [Walcher, 97] behandelt. Die folgenden Unterkapitel beschreiben die Entwurfskriterien von GDSS und geben gegebenenfalls auch einen Überblick über derzeit bereits vorhandene Komponenten sowie Realisierungshinweise.

### 3.1.1 Rasterorientiertes Browsinginterface

GDSS muß, im Gegensatz zu den auf Geo-Daten spezialisierten Systemen (siehe Kapitel 2), ein rasterorientiertes Benutzer-Interface verwenden. Die Begründung für den erhöhten Aufwand gegenüber einer vektororientierten Darstellung ist: auf Planeten existieren im Gegensatz zur Erde nur sehr wenige benannte Merkmale<sup>2</sup>, weiters werden auf der Erde rund 2/3 der Oberfläche von Wasser eingenommen, was die zu repräsentierende Fläche entscheidend verringert. Auf Planeten, wie der Venus, existieren weder Städte noch Ozeane. Die einzig sinnvolle Möglichkeit, die Oberfläche darzustellen, liegt also in einem rasterorientierten Ansatz. Die Daten sollen in verschiedenen Detailstufen zur Verfügung stehen, wobei der Benutzer interaktiv suchen und Abfragen formulieren kann. Um die Suchergebnisse übersichtlicher darzustellen, bietet sich eine Überlagerung der rasterorientierten Anzeige mit Vektorgraphik an. (z.B. Stereo Überdeckung als grünes Polygon, Rot sind Höhenschichtenlinien usw.).

Auf Geo-Daten bezogen, ist eine Rasterdarstellung genau dann sinnvoll, wenn obig genannte Vorteile (benannte Merkmale, 2/3 Wasser) wegfallen, wenn man also gerade im Gebiet des Ozeans nach Fernerkundungsdaten der festen Erde (Klima, Atmosphäre, Wasser ...) suchen möchte. Ist die Auflösung der Fernerkundungsdaten zu groß (z.B. 4m/Pixel bei MISSION<sup>3</sup> Projekt), dann ist ebenfalls eine rasterorientierte Darstellung notwendig, da derzeit keine existierende vektororientierte Karte diese Auflösung erreicht.

---

<sup>1</sup> Im Central Server gilt es unterschiedlichste Daten einheitlich, optimal zu speichern. Dies sind u.a. Bitplanes, Multilayerdaten, Multichannel Bilder, Metadaten (Merkmale, Aufnahmedatum, etc.), Vektordaten, Rasterdaten, ...

<sup>2</sup> So existieren derzeit z.B. auf der Venus 1468 benannte Merkmale

<sup>3</sup> Nähere Information über das MISSION (Multi-Image Synergistic Satellite Information for the Observation of Nature) Projekt kann man unter <http://www.icg.tu-graz.ac.at/MISSION> beziehen.

### 3.1.2 Suche nach Suchpunkt und Suchregion

Den Daten<sup>1</sup> muß eine räumliche Datenstruktur übergeordnet sein, sodaß Fragen nach Suchpunkten/-regionen besonders rasch durchgeführt werden können. Für Suchpunkte sollen die Suchangaben in geographischen Längen- und Breitengraden erfolgen, wobei dies entweder durch manuelle Eingabe oder einen Mausklick an die entsprechende Kartenstelle geschehen kann. Als Suchregionen sind Rechtecke (oder allgemeiner: Polygone) und Ellipsen denkbar. Wiederum kann die Eingabe der Region entweder manuell (bei der Ellipse Mittelpunkt in geographischen Koordinaten und Halbachsen in Metern; beim Polygon die Eckpunkte in geographischen Koordinaten) oder durch interaktives Markieren am Bildschirm erfolgen.

Als Suchergebnis sollte GDSS i.a. einen Bildstapel an Quickviews aller sich im definierten Suchgebiet befindlichen Bilddaten liefern. Quickviews sind verlustbehaftete hochkomprimierte (>1:30) Bilder der hochauflösenden, speicherintensiven Äquivalente.

### 3.1.3 Überdeckungsabfragen

GDSS muß Anfragen bezüglich Merkmal- und Stereoüberdeckung behandeln können. Diese Anfrage erfolgt als zusätzliche Angabe bei einer Regionssuche. Speziell für den Magellan Datensatz sind auch noch Abfragen nach den drei Aufnahmezyklen denkbar. Prinzipiell sollte es ein für alle Daten gültiges Abfrageformular geben und eines, welches speziell auf die jeweilig gewünschten Daten zugeschnitten ist. Denn jede Mission hat u.U. eigene Besonderheiten. Die Antwort des Systems soll auf zwei Arten erfolgen:

1. Quickviews aller geokodierten Bilder, die sich sowohl in der vom Benutzer markierten Region befinden als auch die Zusatz Überdeckungskriterien erfüllen
2. Visualisierung der Überdeckung als vektororientierte Überlagerung zur Rasterdarstellung. Hierzu können verschiedenfarbig schraffierte Flächen verwendet werden.

Mit der zweiten Art der Darstellung unterstützt GDSS eine rasche Übersichtsgewinnung. Der Benutzer kann rasch und effektiv feststellen, auf welchen Teilen des Planeten sich für ihn relevantes Bildmaterial befindet, ohne gleich MBytes an Quickviews herunterladen zu müssen. Mit Hilfe einer Regionssuche kann ein Laden der Quickviews erfolgen.

---

<sup>1</sup> Daten sind prinzipiell einheitlich gespeichert, es wird jedoch schon an dieser Stelle auf die große Vielfalt der Daten hingewiesen. Dies können sein: Rasterdaten, Vektordaten, Metadaten (Datum, Zyklus, ...), Bitplanes, Multilayerdaten, ... Diese Datenvielfalt motiviert die Verwendung einer räumlichen Datenbank.

### 3.1.4 Suche nach Metainformationen

Zu einer Bildabfrage sollen neben den geographischen Koordinaten und den Überdeckungskriterien weitere Zusatzkriterien angegeben werden können. Die wichtigsten sind:

- Überflugszyklus,
- Aufnahmezeitpunkt / -datum,
- Sensorart,
- Satellit,
- Auflösung (radiometrisch, geometrisch),
- Art der Daten<sup>1</sup>,
- Wellenlängenbereich (Infrarot, Thermalkanal, ...),
- Wellenlänge (bei Radarbildern),
- Aufnahmegeometrie (Flughöhe, Aufnahmedauer, Aufnahmegeschwindigkeit),
- Wetterdaten während der Aufnahme (wichtig bei Interferometrie),
- tatsächliche physikalische Auflösung (versus 2D Darstellung).

Für jede individuelle Mission kann es natürlich weitere Kriterien geben. Zusätzlich zu den Quickviews könnte man sich zum Suchergebnis auch noch die sogenannte „Radar History“ (Radargeschichte) anzeigen lassen. Diese beinhaltet umfangreiche Informationen über die Aufnahmeart, die Auflösung und die zahlreichen Prozessierungsalgorithmen, die auf das SAR Bild angewandt worden sind, um es zu geokodieren.

### 3.1.5 Client-Server Architektur

Das zugrundeliegende Konzept von GDSS soll eine verteilte Client-Server Architektur [Vaughn, 94] [Mull, 94] sein, deren prinzipielle Komponenten sind:

- Bild Archive,
- Lokaler Server,
- Zentraler Server,
- Klient,
- Netzwerk (Backbone).

Die Funktionalität dieser Komponenten wird im zweiten Teil dieses Kapitels noch

---

<sup>1</sup> dies ist von Planet zu Planet verschieden und hängt vom Aufnahmeverfahren ab - bei der Magellan Mission gibt es etwa FBIDR (Full Resolution Basic Data Records), FMIDR (Full Resolution Mosaicked Data Records), C-MIDR (Compressed FMIDRs). Die Auswahl muß hier über eine von Planet zu Planet individuelle Liste erfolgen.

näher erläutert. Als Backbone soll eine flexible, leicht skalier- und wartbare Netzwerntechnologie agieren, die auch auf WAN Basis eine hohe Bandbreite (z.B. OC-3 Standard 155 Mbit/sec oder höher) zur Verfügung stellen kann. Diese Bandbreite ist notwendig, um Interaktivität mit großer Wahrscheinlichkeit zu garantieren [Minoli, 94]. Weiters sollte diese Technologie möglichst auf bestehenden physikalischen Protokollen aufsetzen (SDH, PDH, SONET, DQDB, etc. ) sowie TCP/IP unterstützen. Die Wahl fiel hierbei auf die sich gerade im Endstandardisierungsprozeß befindliche Asynchronous Transfer Mode (ATM) Technologie<sup>1</sup>. Sie erfüllt alle für GDSS notwendigen Voraussetzungen. Diese sind:

- Garantie eines interaktiven Suchvorganges durch entsprechende Bandbreite,
- Transparenz (die heterogene Netzwerkstruktur bleibt dem Benutzer verborgen),
- Zuweisung von fixen Bandbreiten für eine Sitzung (Qualität des Dienstes),
- bestmögliche Bandbreitenausnutzung durch asynchrones Senden,
- unterstützt verbindungsorientierte Dienste.

Der letzte Punkt bedeutet, daß ATM bereits beim Verbindungsaufbau feststellt, ob die gewünschten Benutzerparameter (z.B. durchschnittliche Bandbreite, Zellenverlustrate, Zellenverzögerungsrate) erfüllt werden können. Falls nicht, wird die Verbindung verweigert. Speziell für GDSS ist ein leistungsfähiges Netzwerk besonders wichtig, da aufgrund einer erhofft hohen Benutzeranzahl und der zu transferierenden Bilddaten ein hoher Netzwerkverkehr zu erwarten ist.

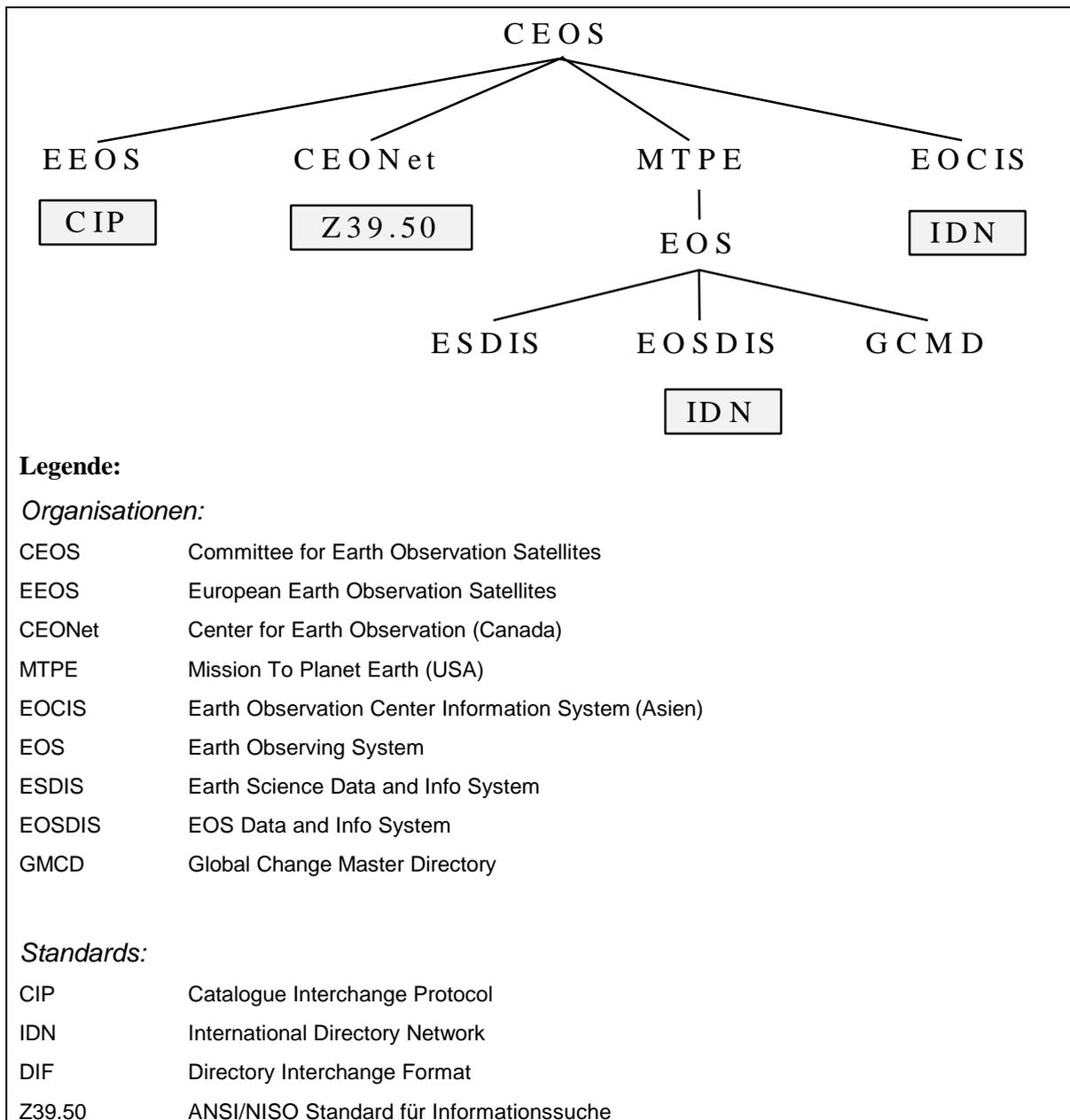
### 3.1.6 Unterstützung von existierenden internationalen Datenaustausch-Standards

Die Verwendung von Datenaustauschstandards ist für ein globales System unabdingbar. Im Rahmen des CEOS (Committee for Earth Observation Satellites) Programmes werden auf der ganzen Welt Standards entwickelt und erprobt, einige davon - wie z.B. Z39.50<sup>2</sup> [ANSI, 92; ANSI , 95; Kunze, 95] - sind bereits standardisiert, einige sind noch in der Testphase, wie etwa das von CEO entwickelte CIP (Catalogue Interchange Protocol) [CEOS, 95]. Anfang 1996 gibt es noch keinen weltweit eingeführten und akzeptierten Standard. Abbildung 3.2 ist ein Überblick über die derzeitigen Entwicklungen, Stand 4/1996.

---

<sup>1</sup> es sei erwähnt, daß auch andere Netzwerntechnologien (FDDI I+II, DQDB, Frame Relay, ISDN) auf ihre Tauglichkeit geprüft werden, dies wird in Kapitel 4 ausführlich behandelt

<sup>2</sup> Z39.50 ist das Protokoll der WAIS (Wide Area Information System) Server, die am Internet im Gopher Sektor wertvolle Suchdienste zur Verfügung stellen



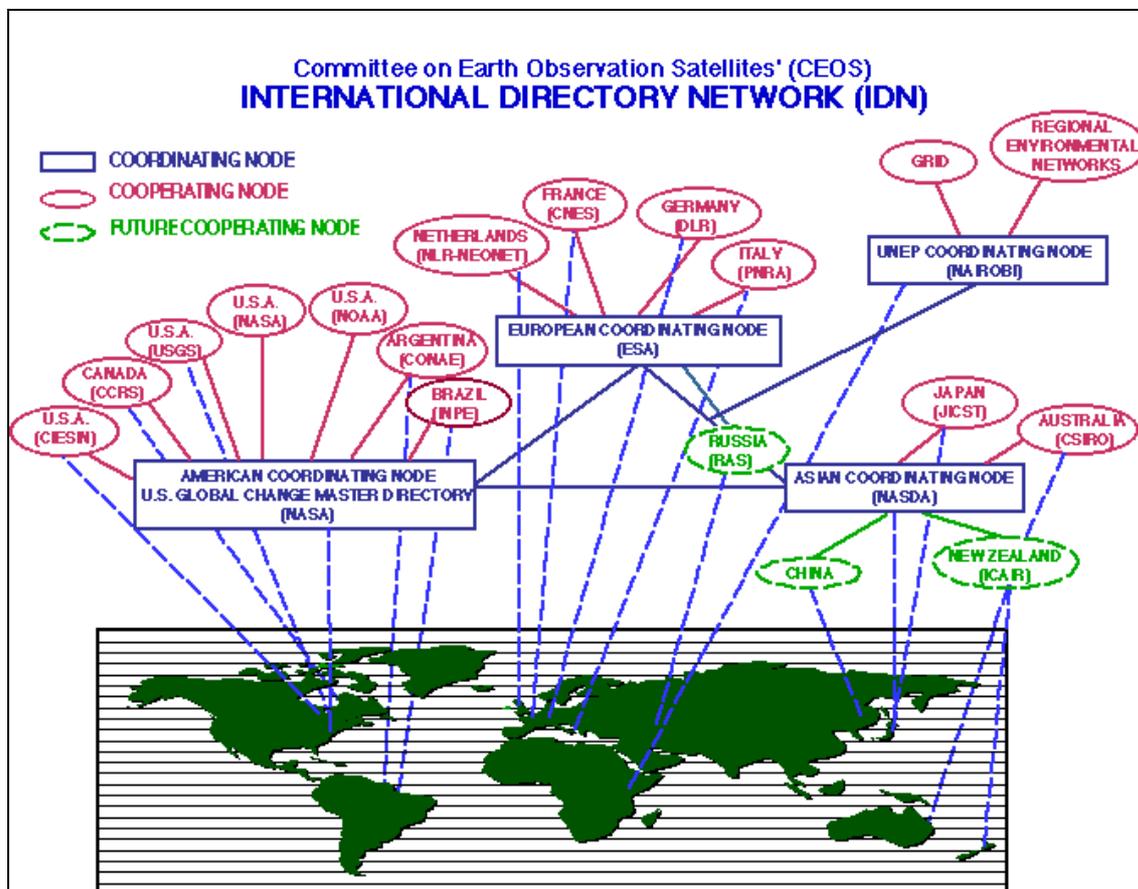
**Abbildung 3.2:** *Internationale Datenaustauschstandards* - Übersicht über internationale Projekte und ihre verwendeten Datenaustauschstandards, Stand April 1996

CIP (Catalogue Interoperability Protocol) ist eine Initiative zur Entwicklung eines Protokolls, einer Datenstruktur sowie einer System Architektur für ein verteiltes Bilddatenbanksystem. CIP basiert auf dem ANSI Z39.50 Protokoll, weist aber gegenüber selbigem einige wichtige Erneuerungen auf [CEOS, 95]. Die Hauptentwicklung geschieht bei der ESA (European Space Agency), das Projekt ist aber Teil der CEOS-Initiative.

CINTEX (Catalogue INTerchange Experiment) [CINTEX, 96] ist ein von CEOS

durchgeführtes Experiment, um etliche Datenzentren in den USA, in Canada, in Japan und in Europa über das vom NASA EOSDIS/IMS (siehe Kapitel 2) verwendeten Protokoll zu verbinden. In Zukunft sollen die Forschungsergebnisse von CIP in das CINTEX Projekt einfließen. Es erfolgt bereits jetzt eine sehr enge Zusammenarbeit zwischen der CINTEX und der CIP Initiative<sup>1</sup>.

IDN (International Directory Network) [IDN, 96] ist ein Netzwerk, um in strukturierter Weise Meta-Geo-Daten zu speichern und einer möglichst großen Anzahl von Benutzern weltweit zugänglich zu machen. IDN verwendet das DIF (Directory Interchange Format), um die Daten strukturiert in ASCII Text zu beschreiben. Es werden keine Originaldaten, sondern nur Verweise auf dieselben gespeichert. IDN ist ein Verbund von drei Primärknoten, die sich in Europa (ESA), Amerika (NASA) und in Japan (NASDA) befinden. An diese sind Subknoten, sogenannte „cooperating nodes“ angeschlossen. Das Netz mit Stand Juli 1996 ist in



**Abbildung 3.3:** IDN Netzwerk. Netzwerk der IDN Knoten auf der Welt samt den Kooperationsknoten. Die drei Hauptknoten befinden sich in Europa (ESA), Amerika (NASA) und in Japan (NASDA).

<sup>1</sup> Chairman des Projektes ist Richard Goebel von der DLR in Deutschland.

Abbildung 3.3 dargestellt.

Zusammenfassend kann man sagen, daß sich alle diese Standards im wesentlichen auf einen einfach Austausch von Metainformationen beschränken, über den Zugriff auf tatsächliche Bilddaten sowie ein einfaches Herunterladen derselben wird jedoch nicht weiter eingegangen. Für den GDSS Prototyp wurde ein Standard verwendet, der genau dies gestattet. Dies ist in Kapitel 5 näher ausgeführt.

### **3.1.7 Intelligentes lokales Caching**

Um den Netzwerkverkehr so gering wie möglich zu halten, soll der lokale Server, an den die lokale Benutzergruppe angeschlossen ist, lokales Caching unterstützen. Caching Strategie und Cachegröße können vom örtlichen Systemadministrator konfiguriert werden. Das größte Problem eines jeden Cachingalgorithmus, das Update Problem, wurde bereits im an der TU Graz entwickelten Hyper-G System [Kappe, 93; Pani, 93] befriedigend gelöst. Hier findet der p-flood Algorithmus [Kappe, 95] Verwendung - in GDSS könnte eine daran angelehnte Strategie zur Anwendung kommen.

### **3.1.8 Batchjob Abarbeitung**

Um zeitkonsumierende Standardprozeduren (z.B. Laden von hochauflösenden Bildern, rechenzeitintensive Anfragen an den zentralen Server) ohne Benutzeraufsicht in Zeiten, in denen die Auslastung des Gesamtsystems gering ist, ablaufen lassen zu können, soll eine Batchjobschnittstelle angeboten werden. Diese gestattet es, Befehle vorzuformulieren, die Abarbeitung derselben jedoch zu einem späteren Zeitpunkt automatisch durchführen zu lassen. Dieser Zeitpunkt kann z.B. eine Kostenminimierung bewirken (Übertragung während der Nacht oder am Wochenende, hier sind i.a. die Tarife günstiger), aber auch einen besseren Ausnutzungsgrad des zentralen Servers bei rechenintensiven Serviceanfragen bewirken, woraus wiederum erhöhte Fairneß gegenüber den anderen Benutzern im System resultiert.

### **3.1.9 Datenvorhersage-Algorithmus des lokalen Server**

Der zentrale Server von GDSS soll die Oberflächenkarten aller verfügbaren Planeten in unterschiedlichen Auflösungsstufen in Form einer Auflösungspyramide (resolution pyramid) speichern. Jede dieser Karten ist aus einer Anzahl von speziellen rechteckigen Datenstrukturen, sogenannten „Map-Tiles“ (Kacheln), zusammengesetzt. Während eines Suchvorgangs lädt der Klient gemäß der Benutzernavigation die benötigten Kacheln nach. Interaktivität würde nun bedeuten, daß sowohl bei einem Zoomvorgang als auch beim Navigieren in der selben Auflösungsstufe die notwendigen Kacheln in Bruchteilen von Sekunden verfügbar und am Bildschirm darstellbar sind. Unter Berücksichtigung, daß auch noch andere Benutzer mit dem System arbeiten, ist es daher notwendig, eine Art

Datenvorhersage durchzuführen. Der lokale Server registriert den aktuellen Standpunkt jedes einzelnen Benutzers und könnte eine Vorhersage folgendermaßen durchführen: er lädt zuerst alle Kacheln, die für einen Schritt in jede Himmelsrichtung notwendig sind, danach alle die für ein Zoom in/out notwendig wären in den lokalen Cache. Bewegt man sich nun auf der Oberfläche, wird dieser Vorgang entsprechend für den neuen Punkt, an dem sich der Benutzer nun befindet, wiederholt. Einmal geladene Kacheln verweilen im lokalen Cache des Servers und werden entsprechend der eingestellten Caching-Strategie und Cachegröße nach gewisser Zeit mit aktuelleren Daten überschrieben.

### **3.1.10 Usergruppenverwaltung**

Der lokale Server ist der zentrale Verbindungspunkt für die Nutzer einer Lokalität. Jeder lokale Server soll einer an ihn angeschlossenen Anzahl von Benutzern die Möglichkeit bieten, die Dienste von GDSS zu nutzen. Daraus resultiert die Notwendigkeit eines Benutzergruppenmanagements. Der lokale Administrator soll hierbei mindestens über die folgenden Möglichkeiten verfügen:

- Aufbau von lokalen Arbeitsgruppen, die zu beliebigen Unterdatenmengen Zugriff haben,
- Vergabe von individuellen und gruppenweisen Benutzerrechten (Autorisierung von Benutzern zur Inanspruchnahme von kostenpflichtigen Diensten, Schreib-Leserechtvergabe lokal/global),
- Vergabe von Prioritäten (wissenschaftliches Personal könnte in etwa eine höhere Priorität bekommen als Studenten, die an einer Übung teilnehmen)

### **3.1.11 Netzwerksicherheit / Verrechnungsstrategien**

Datenmaterial und Rechenzeit an einem Supercomputer sind im allgemeinen nicht gratis verfügbar. Daher muß das System, um eine kommerzielle Verwendbarkeit sicherzustellen, Identifikation, Autorisierung und Verrechnung unterstützen. Diese Transaktionen (Paßwörter, Benutzerrechte, Preise) erfordern auch ein hohes Maß an Übertragungssicherheit. Eine verschlüsselte Übertragung von geheimem Bildmaterial sollte ebenfalls möglich sein. Diese könnte durch ein symmetrisches Schlüsselsystem sichergestellt werden, welches durch Standard-Hardwareverfügbarkeit auch für eine entsprechende Geschwindigkeit sorgt und die Gesamtleistung des Systems nur geringfügig belastet. Für die Identifikation und Autorisierung kann man ein - da nur einmal pro Sitzung angewandt - hierfür besonders gut geeignetes Public Key System verwenden.

Es besteht die Möglichkeit, daß gewisse Bilder nur unter der Angabe des Vertriebers verwendet werden dürfen. Zur Wahrung des Urheberrechtes sollte dem entsprechenden Bildmaterial - einem Stempel gleich - eine Copyright-Nachricht

aufgeprägt werden können.

Unter Mißbrauch wird die beabsichtigte oder unbeabsichtigte Zerstörung bzw. Veränderung von fremden Daten verstanden, dem sollte GDSS aktiv durch ein ausgefeiltes Benutzermanagement (Benutzerrechte Vergabe) und dem Führen eines Systemjournals entgegen treten. Der in Graz entwickelte Hyper-G Server - welcher nun als Hyperwave [HyperW, 97] kommerziell vertrieben wird - erfüllt die Anforderungen nach Benutzerrechten und dem Vorhandensein eines Benutzermanagements bereits sehr gut. Der lokale Server könnte somit auf einem durch entsprechende Module erweitertem Hyperwave Server bestehen.

### 3.1.12 Netzwerkmanagement

Um eine hohe Verfügbarkeit zu gewährleisten, sollte ein globales, heterogenes System über ein ausgeklügeltes, flexibles Netzwerkmanagement verfügen. Hierzu soll GDSS eine ganze Reihe von leistungsfähigen Administratorwerkzeugen anbieten, die eine einfache, zentrale Verwaltung ermöglichen. Folgende Funktionen werden von der OSI für ein Netzwerkmanagementtool empfohlen:

- Leistungsüberwachung,
- Netzwerksicherheit (Identifikation, Autorisierung, Verrechnung, verschlüsselte Übertragung),
- Monitoring (aktive Verbindungen, aktuelle Benutzer, ..),
- flexible Fehlerbehandlung (Diagnose, Fehlerdatenbank, trouble ticket Generation, Systemjournal ...),
- Konfigurationsmanagement.

GDSS soll so fehlertolerant wie möglich sein und über eine intelligente Fehlerbehandlung verfügen. Das Management läßt sich in eine lokale und eine globale Komponente unterteilen: Globale Strategien befassen sich mit dem Hergang:

- wenn eine Netzwerkverbindung zusammenbricht?
- wenn eine neue Komponente (Archiv, Lokaler Server, ...) eingefügt werden soll?
- wenn eine Verbindung permanent überlastet ist?
- wenn mit gewisser Wahrscheinlichkeit ein Hackversuch detektiert wurde?

Lokale Strategien finden ausschließlich im Bereich des Lokalen Servers (lokaler Server und Klienten) Verwendung und beschäftigen sich mit:

- Usermanagement,

- Softwarefehlern,
- lokalen Netzwerkproblemen,
- lokales Systemjournal,
- Performance Monitoring (Cache, Netzwerkauslastung, etc.).

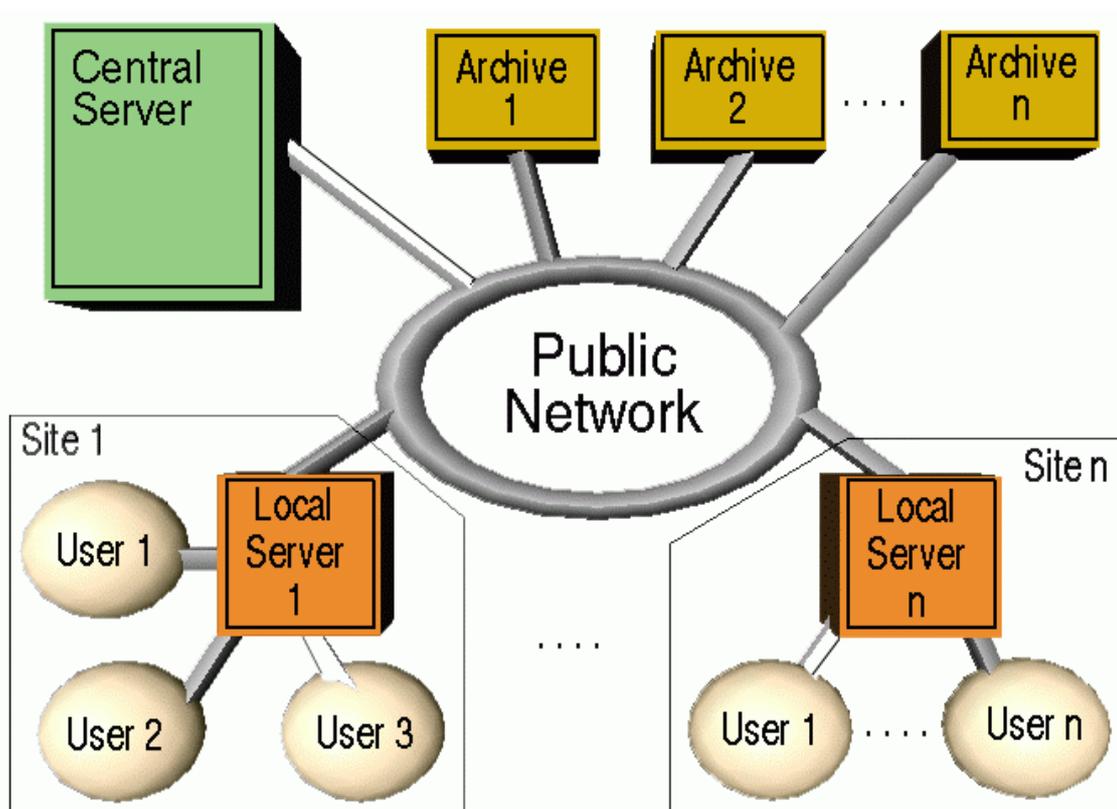
Auftretende Fehler sollen sowohl im lokalen als auch im globalen Management in einer Fehlerdatenbank mitprotokolliert werden. Mit jedem Fehler soll eine Aktion verknüpfbar sein (z.B. eine E-mail an den Systemadministrator schicken oder ein Shellscript starten). Alle Systemkomponenten sollten weiters „Simple Network Management Protocol“ (SNMP) fähig sein und folgende Einträge in ihrer MIB (Management Information Base) haben: Herstellername, Installationsdatum, genaue technische Spezifikationen, sowie eine Kontaktadresse im Fehlerfall. Dies ermöglicht eine zügige Fehlerbehebung bzw. die rasche Durchführbarkeit von Maßnahmen.

### 3.1.13 Remote Data Processing

Mit Hilfe des GUI von GDSS sollen auch Bildmanipulationen ausgewählt werden, wie sie z.B. moderne Fernerkundungs Software (Beispiel: ER-Mapper) gestatten. Dazu gehören Standardprozeduren wie Histogramm-Manipulationen, Grauwertkorrekturen, (Auto-) Klassifizierung oder Filterung, die lokal ausgeführt werden aber auch komplexere nichtstandard Operationen wie etwa Shape-from-Shading oder Matching die remote ausgeführt werden. Jeder dieser Services ist i.a. kostenpflichtig und bedarf einer Autorisierung. Es ermöglicht Benutzern mit billiger Hard- und Software die Dienste von äußerst kostspieliger Fernerkundungssoftware in Anspruch zu nehmen<sup>1</sup>. Hierbei könnte man an einer Lokalität (in der Regel beim Standort des Zentralen Servers) eine Anzahl von Lizenzen kaufen, die dann per Remote Access und dem leistungsfähigen ATM Backbone von GDSS einer globalen Benutzerschicht zur Verfügung gestellt werden. Die Transparenz von GDSS und der leistungsfähige Netzwerk Backbone gewährleisten die Unerheblichkeit der Distanz zum Benutzer, welcher im wesentlichen alle Dienste so benutzen kann, als wären sie lokal verfügbar.

---

<sup>1</sup> z.B. beträgt für den ER-Mapper der Preis für eine Lizenz 300.000 Schilling (Stand Sept. 1995)



**Abbildung 3.4:** *Generelle Architektur von GDSS* - Der „Central Server“ speichert Quicklooks und Oberflächenkarten in verschiedenen Auflösungen aller von Archiven angebotenen Bilddaten. Benutzer Klienten greifen auf den Central Server mittels eines „Local Server“ zu. Für jede an GDSS teilnehmende Organisation wird ein Local Server aufgesetzt, an den eine Reihe von Benutzern angeschlossen werden können. Der gesamte Such- und Bestellprozeß ist für den Benutzer transparent, d.h. er braucht weder zu wissen in welchem Archiv seine Daten verweilen noch wo sich dieses Archiv physikalisch befindet. Bis zur Bestellung verweilen die hochauflösenden Daten in den Archiven, die von kommerziellen Anbietern oder öffentlichen Organisationen betrieben werden.

## 3.2 GDSS Architektur

Die folgenden vier Unterkapitel beschreiben die GDSS Architektur, wobei eine Top-down Strategie verfolgt wird. Ausgehend von einem Gesamtdesign werden nacheinander die einzelnen Komponenten des Systems vorgestellt.

### 3.2.1 Gesamtdesign

In Abbildung 3.4 ist die generelle Architektur von GDSS zu sehen. Hierbei handelt es sich um ein im höchsten Grade verallgemeinertes Konzept, das in den folgenden Kapitel in Top-down Manier schrittweise verfeinert wird.

GDSS besteht aus fünf Hauptkomponenten:

- Bildarchive (Image Archives)

- Lokaler Server (Local Server)
- Benutzer Klient
- Zentraler Server (Central Server)
- Netzwerk

**Bildarchive** werden von kommerziellen Anbietern oder öffentlichen Institutionen (Regierung, Universitäten, etc.) betrieben. Das verteilte Konzept der Bildarchive soll einerseits die bereits bestehenden Archive zu einem globalen System integrieren und somit Benutzern weltweit einheitlichen Zugriff gewährleisten, andererseits soll der Mehrfachspeicherung von Terabyte an Daten aktiv entgegengetreten werden. Der Aufwand zur Anpassung der Archive an GDSS ist dabei zu minimieren. Die genauen Voraussetzungen werden im Kapitel 3.2.4 (Spezifikation der Archive) näher erläutert.

Ein **lokaler Server** wird an jeder sich an GDSS beteiligenden Institution aufgesetzt. Im Hinblick darauf, daß er als Anschlußpunkt für eine lokale Benutzergruppe an den ATM Netzwerk Backbone dient, kann man ihn auch als Konzentrator sehen. Der lokale Server nimmt Aufgaben wie lokales Benutzermanagement, Verrechnung, Batchjobprozessierung, Caching und Datenvorhersage wahr. Er erhält den Kontakt zum zentralen Server aufrecht und soll die individuelle Benutzerverrechnung abwickeln. Genauere Spezifikationen des Lokal Server werden im Kapitel 3.2.3 (Spezifikation des Lokalen Servers) beschrieben.

Der **Benutzer Klient** ist mit seinem GUI für eine plattformübergreifende, einheitliche Interaktion mit den Benutzern verantwortlich. Diese Interaktion ist einerseits in den Abfrage Dialog gegliedert, der mit Hilfe geeigneter Formulare eine komfortable Komposition von Abfragen zulassen soll, andererseits in das interaktive Browsing durch Oberflächenkarten der Planeten. Diese Karten werden in verschiedenen Auflösungsstufen rasterorientiert angeboten, der Benutzer kann mit ihnen interaktiv agieren. Dies umfaßt einfaches Ein-/Auszoomen sowie das Markieren von Interessensregionen. Um Suchergebnisse geeignet visualisieren zu können, sollen die Rasterdaten mit Vektordaten überlagert werden können, die etwa Konturlinien, Ausdehnung und Namen von Merkmalen, Grenzen von geokodierten Bildern oder persönliche Notizen repräsentieren. Ergebnisse von Überdeckungsabfragen (z.B. „Gib mir alle Gebiete die „same side Stereo“ Aufnahmen haben“) könnten mit Hilfe von Farbkodierungen verwirklicht werden. So können etwa Gebiete mit Stereoüberdeckung in einer Farbe schraffiert werden. Die Entwicklung eines komfortablen Benutzer Klienten ist der Arbeit von [Walcher, 97] zu entnehmen.

Diese Arbeit konzentriert sich auf den Aspekt, einen plattformübergreifenden und einheitlichen Zugriff auf die Daten zu gewährleisten, da dies für ein globales System von entscheidender Wichtigkeit ist. Geeignete Konzepte und Methoden werden erörtert. Die Implementierung von Speziallösungen für jede an GDSS

teilnehmende Plattform wirft eine große Anzahl von Problemen auf. Neben der finanziellen Frage der unterschiedlichen Implementierungen, der schwierigen Behandlung von Neuzugängen, dem komplizierten Management der heterogenen Komponenten stellt sich hier auch die Frage des Update Problems. Diese Vorgehensweise ist also von vornherein nicht erstrebenswert. Eine geeignete Lösung dieses Problems wird in Kapitel 5 (Implementierung) vorgestellt.

Der **zentrale Server (ZS)** ist neben dem Netzwerkbackbone die wichtigste Komponente im GDSS. Hier werden sowohl Quicklooks und Oberflächenkarten in verschiedenen Auflösungsstufen als auch Metadaten und räumliche Informationen aller im GDSS vorhandenen Bilddaten gespeichert. Die hochauflösenden Daten verweilen bis zur Bestellung in den Archiven. Zu den Aufgaben des ZS zählen:

- die Gewährleistung eines interaktiven Suchvorganges für mehrere lokale Server und ihre Benutzer durch das Liefern der gerade benötigten Karteninformationen (sogenannte „Map-tiles“, Bildkacheln - eine rechteckige Datenstruktur mit deren Hilfe die Bildschirmdarstellung erfolgt);
- die Behandlung von Such- und Überdeckungsabfragen;
- die Verrechnung von Bilddaten sowie das „Cropping“ (Herausschneiden jener Daten, die der Benutzer benötigt);
- Verarbeitung von rechenintensiven Bildbearbeitungsalgorithmen die von Benutzern per Fernzugriff in Auftrag gegeben werden. Ergebnisse werden an den lokalen Server retourniert.

Für eine detailliertere Spezifikation sei auf das Kapitel 3.2.2 (Spezifikation des zentralen Servers) verwiesen.

Das **Netzwerk** ist die alles verbindende Struktur im GDSS. Der primäre Vorsatz lautet hier: Dezentralisierung von teuren Ressourcen (Fernerkundungssoftware, Speichermedien, Supercomputer, ...) und Verbindung dieser mittels einem leicht skalierbaren Backbone, welcher letzten Endes natürlich auch die erforderliche Bandbreite zur Verfügung stellen muß. Ohne zusätzlichen Aufwand ergeben sich hieraus folgende Vorteile:

- Verminderung der Redundanz teurer Hardware,
- bessere Auslastung von Supercomputern - Amortisierung der Anschaffung,
- Vernetzung der Wissenszentren weltweit zum Austausch von „Know-how“,
- ein Datenendgerät schafft mit einheitlichem Benutzerinterface Zugang zu vielen unterschiedlichen Diensten,
- evolutionäres Wachstum,
- Zuverlässigkeit,

- Verfügbarkeit,
- geographische Flexibilität.

Selbstverständlich wirft die Vernetzung von heterogener Hard- und Software (und GDSS besteht weitestgehend aus heterogenen Komponenten) u.U. auch große Probleme auf. Entscheidend ist es daher, sich soweit wie irgendwie möglich auf weltweite Standards zu stützen und keine Speziallösungen zu verwenden. Die Wahl fiel hierbei auf den von der CCITT 1991 vorgestellten internationalen ATM Standard. Das gesamte 4. Kapitel befaßt sich mit dem Netzwerk Backbone, von der Anforderungsliste bis hin zum Aufbau und konkreten Lösungsansätzen. Weiters wird die Wahl von ATM argumentiert und ein Vergleich mit Alternativtechnologien angestellt.

### 3.2.2 Spezifikation des zentralen Servers

Die wichtigsten Merkmale des zentralen Servers sind:

- Speicherung von Oberflächenkarten in mehreren Detailstufen für alle im System vorhandenen Planeten,
- Metadatenbank für Überdeckungsabfragen,
- Quicklook Datenbank,
- Methodendatenbank,
- Identifikation,
- Autorisierung,
- Archivkommunikation,
- zentrales Management.

Unter der Methodendatenbank versteht man ein Pool von rechenintensiven Funktionen, die Benutzer remote am Server ausführen lassen können.

Das im gesamten GDSS einheitlich verwendete Protokoll ist eine der wichtigsten Software Komponenten des Systems. Die gesamte für GDSS benötigte Funktionalität wird von keinem derzeit bekannten international standardisierten Protokoll unterstützt. Deswegen wird ein neues, abstraktes Protokoll, das sogenannte GQP (General Query Protocol) vorgeschlagen. GQP ist ähnlich dem Internet Protokoll kein einzelnes Protokoll, sondern eine Protokoll-Menge. Unter „GQP“ ist die Gesamtheit aller Protokolle zu verstehen, die benötigt wird, um eine einheitliche Kommunikation zwischen den GDSS Systemkomponenten zu gewährleisten. Die Teilprotokolle müssen - soweit möglich - auf internationalen Standards basieren und, um eine globale Kommunikation zu gewährleisten, auf

TCP/IP aufsetzen. Insgesamt soll GQP vier Kategorien von Aufgaben wahrnehmen, die in Tabelle 3.1 zusammengefaßt sind.

Die wichtigste Aufgabe des GQP ist es, über ein im gesamten System einheitliches Benutzerinterface den Zugang zu Fernerkundungsdaten herzustellen. Der Benutzer darf also keinesfalls in die Verlegenheit kommen, jedesmal wenn er ein anderes Bilddatenarchiv durchsucht, neue Kommandos lernen zu müssen oder mit einem völlig anders gearteten Benutzerinterface arbeiten zu müssen. Gegebenenfalls müssen Konvertierungen vorgenommen werden, alle Benutzer des Systems sehen stets dasselbe graphische Interface und können unabhängig von den gesuchten Daten immer dieselben einheitliche Kommandos verwenden.

<b>Kategorie</b>	<b>Funktionalität</b>
<b>I Räumliche Suchanfragen</b>	Anfragen nach Daten um einen Punkt / in einer Region, Anfragen nach Überdeckung und Anfragen zur Filterung (bzw. detaillierterer Information) mit Angabe von Zusatzkriterien (Datum/Zeit, Berechnungsalgorithmus, Radargeschichte, Prozessierungsparameter, Typ des Sensors, Satellit)
<b>II Daten Wartung</b>	Einbringung von neuen bzw. Auffrischung von vorhandenen Bilddaten
<b>III Dienste</b>	Die Verwendung von rechenintensiven Bildbearbeitungsalgorithmen sowie die Funktionalität teurer Fernerkundungssoftware Pakete können über ein Fernzugriff Interface beim Klienten angeboten werden. Die Berechnung kann auf einer massiv parallelen Maschine innerhalb von GDSS erfolgen, Resultate werden zum Benutzer geschickt. Der Vorgang soll für den Benutzer unsichtbar sein, die Dienste werden in Rechnung gestellt.
<b>IV Administrative Anfragen</b>	Wichtige Funktionen für das globale Systemmanagement. Statusanfragen, Performance Überwachung, neue Verbindungen aufbauen, Ressourcen Management, ...

**Tabelle 3.1:** Die vier Kategorien des abstrakten GQP Protokolls. Siehe Text für nähere Erklärung.

Dies ist ein äußerst wichtiges Design Kriterium, man bedenke nur die lange Einarbeitungszeit, um immer wieder neue Systeme bedienen zu lernen. Ganz zu schweigen vom Schaden, den ein Unternehmen erleidet, wenn ein Mitarbeiter, der

es gelernt hat viele Systeme zu bedienen, den Betrieb verläßt. Es ist also von großer Relevanz, daß das aus heterogenen Komponenten bestehende GDSS eine homogene Benutzerschnittstelle anbietet.

GQP soll soweit wie möglich auf dem bereits existierenden internationalen Datenaustausch und Datenbankabfrage Standard Protokoll Z39.50 [ANSI, 92; ANSI, 95] aufgebaut werden, auf dem auch das von CEOS entwickelte CIP (Catalogue Interoperability Protocol) [CEOS, 95] basiert. Nur die nichtunterstützten Kategorien sollen neu implementiert werden. Internationale Standards sind von entscheidender Wichtigkeit, da GDSS keine „Insellösung“ sondern ein globales System sein soll.

Z39.50 ist ein Client/Server Protokoll, das eine abstrakte Abfrage, Sichtweise und Selektion von Daten gestattet. Z39.50 ist plattformunabhängig und unterstützt alle Datenbanken (per Interface). Es unterstützt somit eine homogene Abfrage einer heterogenen Datenbankwelt. Der Benutzer muß sich nur einmal mit den Kommandos und dem Benutzerinterface vertraut machen - braucht sich also nicht jedes Mal wieder beim Konsolidieren eines neuen Bildarchivs mit einer neuen Syntax vertraut machen.

Derzeit nicht vorhandene Komponenten von GQP - das sind im wesentlichen die Kategorien II - IV - sollten ähnlich dem WWW HTTP (Hyper Text Transfer Protocol) [Ragetti, 95] ein einfaches ASCII Protokoll sein. Ein ASCII Protokoll weist Vorteile der leichten Portierbarkeit auf verschiedene Plattformen und der einfachen Erweiterbarkeit auf, da leicht verständlich.

### 3.2.3 Spezifikation des Lokalen Servers

Folgende Aufgaben obliegen dem lokalen Server:

- lokales Caching der Daten,
- Identifikation,
- Autorisierung,
- Usergruppenverwaltung,
- individuelle Verrechnung,
- Datenverkehrsmanagement - Anbindung an einen „Daten Highway“,
- lokales Management.

Das lokale Caching der Daten ergibt sich aus der einfachen Tatsache, daß innerhalb des GDSS MBytes an Bilddaten transferiert werden können. Caching trägt zu einer deutlichen Netzwerkentlastung bei, zumal pro lokalen Server ohnehin meist am selben Datensatz gearbeitet wird (z.B. Graz: Magellan Daten, Innsbruck: Ötztal Daten, usw.). Aber auch für das schnelle Browsen der Oberfläche ist ein lokales

Caching vor allem bei schlechten Netzwerkverbindungen unerlässlich.

Identifikation, Autorisierung, individuelle Verrechnung der Daten sowie Usergruppenverwaltung sind vor allem für ein kommerzielles System unerlässlich. Um die Verwaltung der Kostenverrechnung zu dezentralisieren und somit den zentralen Server zu entlasten soll das GDSS in einer solchen Weise organisiert sein, daß der zentrale Server nur eine Verrechnung pro lokalem Server vornimmt und der lokale Server die Aufgabe der Benutzer-individuellen Verrechnung wahrnimmt. Dieses Konzept ist in Kapitel 3.3.2 (Design des lokalen Servers) etwas näher ausgeführt.

Der lokale Server ist weiters der zentrale Einstiegspunkt in einen Daten Highway - vorzugsweise zu einem ATM Backbone. Eine Ankoppelung eines Hochgeschwindigkeits-Netzwerkes bis zum Benutzer wäre aus heutiger Sicht zu teuer und ist auch für GDSS nicht unbedingt notwendig. Die Regelung der einzelnen Verkehrsströme, vor allem die Aufteilung der Bandbreite derselbigen, obliegt dem lokalen Server.

Um Interaktivität in GDSS zu gewährleisten, kann man nicht ohne weiteres das existierende Internet verwenden. TCP/IP, das Internet Protokoll, teilt die vorhandene Bandbreite so fair wie möglich zwischen den Teilnehmern auf. Dies hat jedoch zur Folge, daß bei einer Überlastung keiner mehr arbeiten kann, da das Netz einfach vollkommen überlastet ist. Lange Wartezeiten, Verbindungsunterbrechungen durch „Time outs“ und dadurch frustrierte Benutzer sind die Folge. Besser wäre es, wenn der Benutzer sagen könnte, welche Bandbreite er benötigen würde und das Netz entsprechend der Vorgaben diese garantieren oder eben zurückweisen würde. Entsprechende Forschung für ein neues IP Version 6, auch als RSVP bezeichnet (ReSerVation Protocol), ist im Gange [Mitzel, 96], [Berson, 96], [Demirtjis, 96], [Herzog, 95]. Eine erste funktionstüchtige Spezifikation und Software sind seit November 1996 erhältlich [Braden, 96]. Die RSVP Forschung ist jedoch noch sehr jung, praktische Ergebnisse sind noch kaum vorhanden. Eine funktionierende Version des RSVP stand während der Durchführung dieses Projektes noch nicht zur Verfügung. Ein prinzipielles Muß für das RSVP ist allerdings das Vorhandensein einer entsprechenden Kontrollfunktion für die Benutzer - es ist nämlich zu befürchten, daß ansonsten Benutzer mehr Ressourcen als tatsächlich benötigt reservieren. Diese Kontrollfunktion ist derzeit im RSVP nicht vorhanden, man muß sich also auf die Fairneß der anderen Teilnehmer verlassen können.

Innerhalb von GDSS sollte es demnach einen Mechanismus geben, der für die Dauer eine Sitzung gewisse Vorgaben für eine Verbindung - etwa: Bandbreite, Zellenverlustrate, Zellenverzögerung - mit einer gewissen hohen Wahrscheinlichkeit garantieren kann. Können diese Parameter nicht erfüllt werden, sollte die Verbindung einfach verweigert werden. Dieser Mechanismus sollte möglichst mit

dem Standard Internet Protokoll zusammenarbeiten, um eine globale Vernetzung garantieren zu können. Dies schließt vorläufig die Verwendung des oben erwähnten RSVP Protokolls mangels an Verbreitung aus. Sobald es sich einigermaßen flächendeckend etabliert, wäre es aus heutiger Sicht eine gute Wahl, da es u.a. die Reservierung von Ressourcen entlang einer Verbindung unterstützt.

Jeder lokale Server soll über ein lokales Systemmanagement verfügen. Dieses wird mit dem zentralen Management verbunden und muß Informationen wie Komponentenausfälle, Sicherheitsverletzungen, Netzwerkausfälle etc. an dieses weiterleiten. Die Kombination lokales und zentrales Management ergibt ein globales Management System, welches in verteilter Weise eine sinnvolle Verwaltung des GDSS ermöglicht.

### 3.2.4 Spezifikation der Archive

Für die Datenarchive wird keine spezielle Architektur vorgeschlagen, da sie ohnehin bereits überall auf der Welt bestehen. Eine Reorganisation der Daten bzw. ein kompletter Austausch der Hardware Komponenten wäre inakzeptabel. Statt dessen wird dem verteilten Server Konzept von GDSS Rechnung getragen, indem die Datenorganisation und die bereits existierende Hardware bestehen bleiben. Die Archive müßten lediglich mit einem GQP Standardinterface ausgestattet werden, das die Befehle der Kategorien I (Räumliche Anfragen), II (Datenwartung) und IV (Administrative Anfragen) beherrschen muß (siehe auch KapitelSpezifikation des zentralen Servers). Weiters müssen alle Archive die folgenden Voraussetzungen erfüllen:

- Das Archiv muß 24 Stunden am Tag/7 Tage die Woche verfügbar sein, um Anfragen des zentralen Servers jederzeit abhandeln zu können;
- Anfragen müssen weitestgehend vollautomatisch behandelt werden können, die Anwesenheit eines Operaten ist nur in äußersten Notfällen erforderlich;
- Neu hinzugekommene Archive müssen dem zentralen Server eine zylindrisch projizierte Oberflächenkarte in mehreren Detailstufen (Auflösungspyramide) sowie Quicklooks aller geokodierten Bilder zur Verfügung stellen. Die Quicklooks müssen samt Metadaten an die Datenbank des zentralen Servers weitergeleitet werden.
- Falls neue Daten hinzugekommen sind, muß das Archiv den zentralen Server darüber informieren und gegebenenfalls die Oberflächenkarten und/oder die Quicklooks erneuern (Update).

---

<sup>1</sup> etwa wenn selten gebrauchte Bilddaten gelesen werden müssen, die Daten daher auf Band liegen und kein vollautomatisches Robot Archiv lokal verfügbar ist

Die Verrechnung der Daten erfolgt über Kommandos der Kategorie IV direkt mit dem zentralen Server, welcher wiederum die Verrechnung mit den einzelnen lokalen Servern vornimmt.

Das Interface soll, wie bereits in Kapitel 3.2.2 (Spezifikation des zentralen Servers) erwähnt, weitestgehend auf dem internationalen Standard Protokoll Z39.50 basieren. Dieses Protokoll gewährleistet neben einer standardisierten Datenbankabfrage und Suche auch Plattformunabhängigkeit.

### 3.3 GDSS System Design

Nach einer allgemeinen Spezifikationsbeschreibung widmet sich der folgende Abschnitt nun einer etwas detaillierteren Beschreibung der wichtigsten Komponenten des Systems. Das sind neben dem Netzwerk der zentrale und der lokale Server. Die zwei folgenden Unterabschnitte beschreiben diese beiden Komponenten.

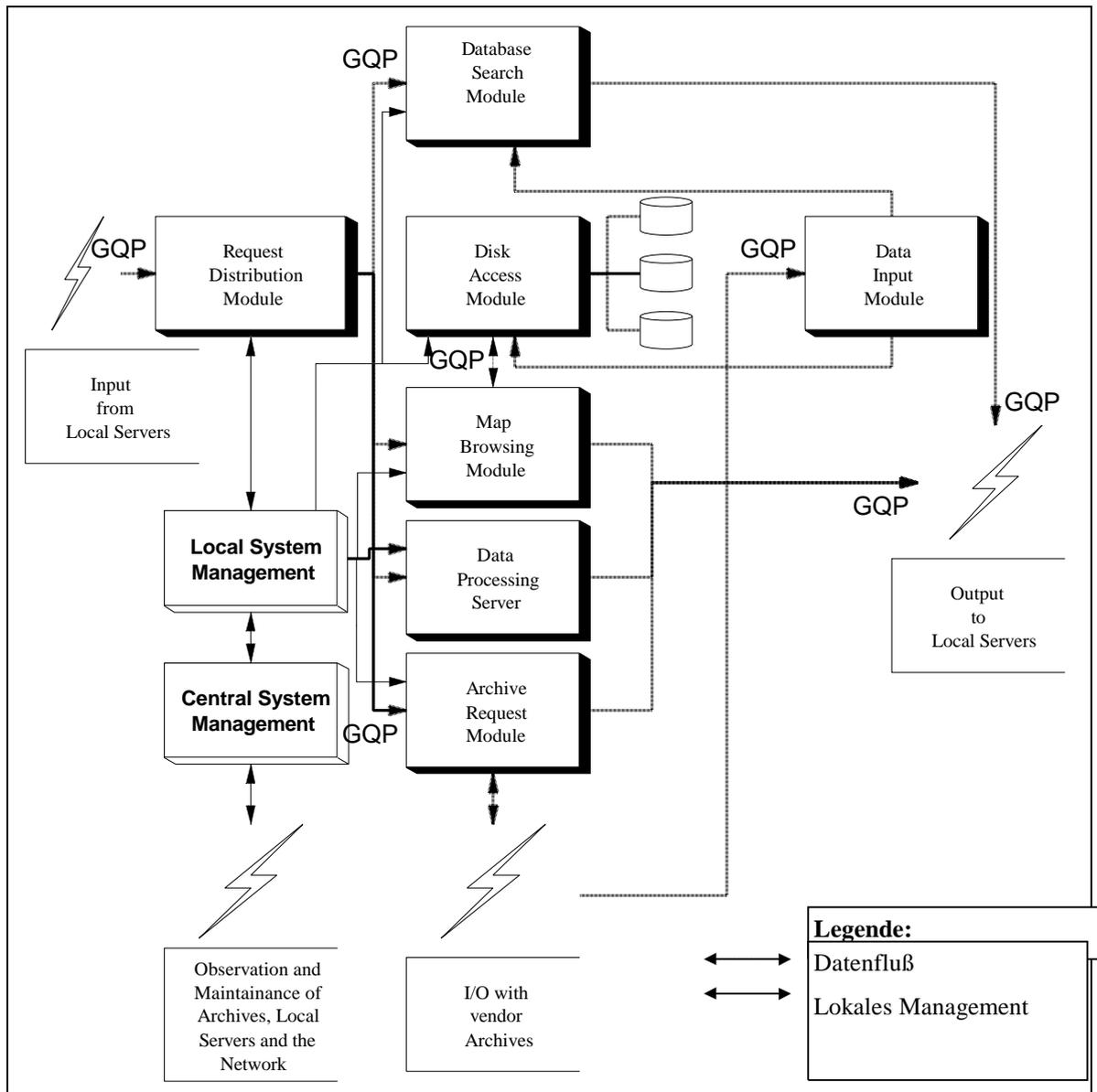
#### 3.3.1 Design des zentralen Servers

In Abbildung 3.5 ist der prinzipielle Aufbau des Zentralen Servers dargestellt. In der Abbildung kann man die Hauptmodule, die Kommunikationspfade und den lokalen Managementfluß derselben erkennen. Im folgenden Abschnitt werden die einzelnen Module näher durchleuchtet.

**Abfrageverteilungs-Modul (Request Distribution Module):** Dieses Modul soll im wesentlichen zwei Aufgaben wahrnehmen: (1) Identifikation des jeweiligen lokalen Servers und (2) die effiziente<sup>1</sup> Aufteilung der eingehenden Anfragen an die jeweils zuständigen Module. Diese sind das Datenbanksuch-Modul (Database Search Module) für räumliche Abfragen, das Oberflächenbrowsing-Modul (Map Browsing Module) für interaktives Visualisieren der Oberfläche, das Archivabfrage-Modul (Archive Request Module) für das Bestellen von hochauflösenden Daten und der Datenprozessierungs Server (Data Processing Server) um Zugriff auf Bildbearbeitungsalgorithmen und komplexe Fernerkundungssoftware zur Verfügung zu stellen.

---

<sup>1</sup> mit effizient ist gemeint, daß bestimmte Module auch mehrfach implementiert sein können und somit die Möglichkeit besteht, Benutzeranfragen parallel zu exekutieren.



**Abbildung 3.5:** Prinzipielle Architektur des Zentralen Servers - Das Diagramm stellt die Hauptmodule des Zentralen Servers, den Daten- bzw. Kommunikationsfluß und die Kommunikationsverbindungen in die Außenwelt dar.

Die Identifikation soll hierbei einem verteilten Konzept unterliegen. Das bedeutet, daß der zentrale Server keine Rücksicht auf individuelle Benutzer an den jeweiligen Lokalitäten nehmen soll, sondern lokale Server identifiziert und autorisiert. Details wie Benutzeridentifikation und -autorisierung, sowie individuelle Verrechnung der in Anspruch genommenen kostenpflichtigen Dienste liegen in der Verantwortung des lokalen Servers bzw. des lokalen Systemadministrators. Dies ist eine konsequente Umsetzung der Entwurfsbestimmungen lokale und globale Management Strategien (siehe Kapitel 3.1.12).

**Datenbanksuch-Modul (Database Search Module) :** Dieses Modul muß GQP Anfragen der Kategorie I (räumliche Anfragen) unterstützen. Primär sollten Anfragen nach Metadaten/Quickviews um einen Interessenspunkt oder eine Interessensregion beantwortet werden können. Weiters soll das Modul die Suche nach Metainformationen unterstützen. Die wichtigsten wurden bereits in Kapitel 3.1.4 definiert.

Da Metainformationen auch missionsspezifisch sein können (z.B. Magellan: Gliederung der Daten nach den drei Überflugszyklen), muß auch eine Art generisches Konzept vorgesehen werden. Je nach in der Abfrage involvierten Missionen kann das Abfrageformular also neben den Stammeinträgen auch noch zusätzliche Felder enthalten.

Als dritte und letzte Aufgabe müssen noch Überdeckungsabfragen behandelt werden. Diese Art der Abfrage liefert auf die Benutzerkriterien zutreffende Gebiete auf der Oberfläche zurück. Zum Beispiel könnte ein Anfrage lauten: „Zeige mir alle Gebiete auf dem Planet Venus, die ‘same side stereo’ aufweisen.“ Das Datenbanksuch-Modul soll mit einer Menge von Arealen antworten, auf die dieses Kriterium zutrifft. Die Visualisierung kann beim Benutzer etwa durch Schraffur erfolgen.

**Oberflächenbrowsing-Modul (Map Browsing Module):** GDSS soll über ein rasterorientiertes Anzeigefenster verfügen, welches am Bildschirm die Planetenoberfläche in unterschiedlichen Auflösungsstufen darstellt. Die Aufgabe des Moduls besteht nun darin, einer Anzahl von Benutzern die jeweils benötigten Bildschirmausschnitte rasch zur Verfügung zu stellen. Dies ist ein äußerst zeitkritischer Prozeß, da Interaktivität<sup>1</sup> gewährleistet sein muß. Hinzu kommt, daß gewöhnlich mehrere Benutzer gleichzeitig navigieren und somit laufend Abfragen verursachen. Eine entsprechend der Benutzerzahl skalierbare Lösung ist hier von entscheidender Wichtigkeit. Eine entscheidende Entlastung dieses Moduls ist mit Hilfe des lokalen Caching des lokalen Servers zu erzielen (siehe Kapiteß.1.7).

**Archivabfrage-Modul (Archive Request Module):** Die Hauptaufgabe dieses Moduls besteht im Holen von hochauflösenden Bilddaten aus den angeschlossenen Archiven und dem Liefern derselben an den (autorisierten) lokalen Server. Diese Aufgabe wird von insgesamt drei Submodulen wahrgenommen. In einem ersten Schritt müssen die eingehenden Daten-Bestellungen an die jeweils dafür zuständigen Archive verteilt werden. Um dies effizient zu machen, könnten mehrere zugleich eintreffende Anfragen parallel abgearbeitet werden.

Die Archive antworten auf die Anfrage mit den gewünschten Daten - in der Regel einem Bildstapel - und optional mit einem Preis. Wird ein Preis mitgeschickt,

---

<sup>1</sup> Mehrere Minuten auf den Bildaufbau zu warten, wenn man eine Kachel weiterravigiert, sind nicht tragbar.

erfolgt eine Verrechnung mit dem lokalen Server. Diese Verhandlung muß entsprechenden Sicherheitsbestimmungen unterliegen. Netzwerksicherheit (vor allem Identifikation Autorisierung und Verschlüsselung) spielt hier eine wesentliche Rolle. Diese Aufgaben soll das zweite Submodul wahrnehmen, der sogenannte Verrechnungsserver (Accounting Server).

Sobald der Betrag vom Konto des lokalen Servers abgebucht wurde, wird der erhaltene Bildstapel an das letzte Submodul weitergeleitet, den sogenannten Cropping Server. Hier werden die bestellten Bilddaten, sofern das jeweilige Archiv dies nicht unterstützen sollte, exakt auf die vom Benutzer definierte Größe gebracht. Der Benutzer sollte also nur jene Daten erhalten, an denen er tatsächlich interessiert war. Der nun fertig aufbereitete Bildstapel wird an den lokalen Server und von dort an den Benutzer verschickt.

**Datenprozessierungs-Server (Data Processing Server):** Dies ist die Umsetzung der Entwurfsspezifikation aus dem Kapitel 3.1.13 (Remote Data Processing). Primäres Ziel ist es, komplexe Soft- und Hardware Ressourcen einer globalen Benutzergruppe mit vergleichsweise billiger lokaler Hard- und Software zur Verfügung zu stellen. Der Datenprozessierungs-Server sollte weiters einfach auf weitere Funktionalität skalierbar sein, folgende Grundfunktionen könnte er z.B. anbieten:

- ein Grundset von Standard Bildverarbeitungsfunktionen (Bildfilterungsfunktionen, Histogramm Manipulationen, Kontrast- und Helligkeitskorrektur, Farbmanipulationen, ...),
- automatische und manuelle Klassifizierung,
- Matching von Stereobildern und Generierung von Höhenmodellen,
- Shape from Shading für die Verfeinerung von 3D Höhenmodellen,
- Berechnung von 3D Überflugssequenzen als MPEG Video.

Das Anbieten einer Standard Fernerkundungssoftware, (wie ER-MAPPER) wäre ebenfalls denkbar, vorausgesetzt es spielt keine Rolle, von wo eine gekaufte Anzahl von Lizenzen verwendet werden. Eine Übersetzung vom GDSS Standardinterface in die von der jeweiligen Software verstandenen Befehle wäre aber notwendig, um das Entwurfskriterium des einheitlichen Zugangs zu erfüllen.

Solche Arten von Diensten sind in der Regel nicht gratis, eine entsprechende Verrechnung mit dem lokalen Server ist vorzusehen.

**Diskzugriffs-Modul (Disk Access Module):** Dieses Modul wird vom Oberflächenbrowsing-Modul (Map Browsing Modul) in Anspruch genommen. Hier sollten die Bilddaten der Planetenoberflächen gespeichert werden. Diese Karten könnte man aus einzelnen Kacheln (Map Tiles) zusammensetzen, die mit Hilfe einer

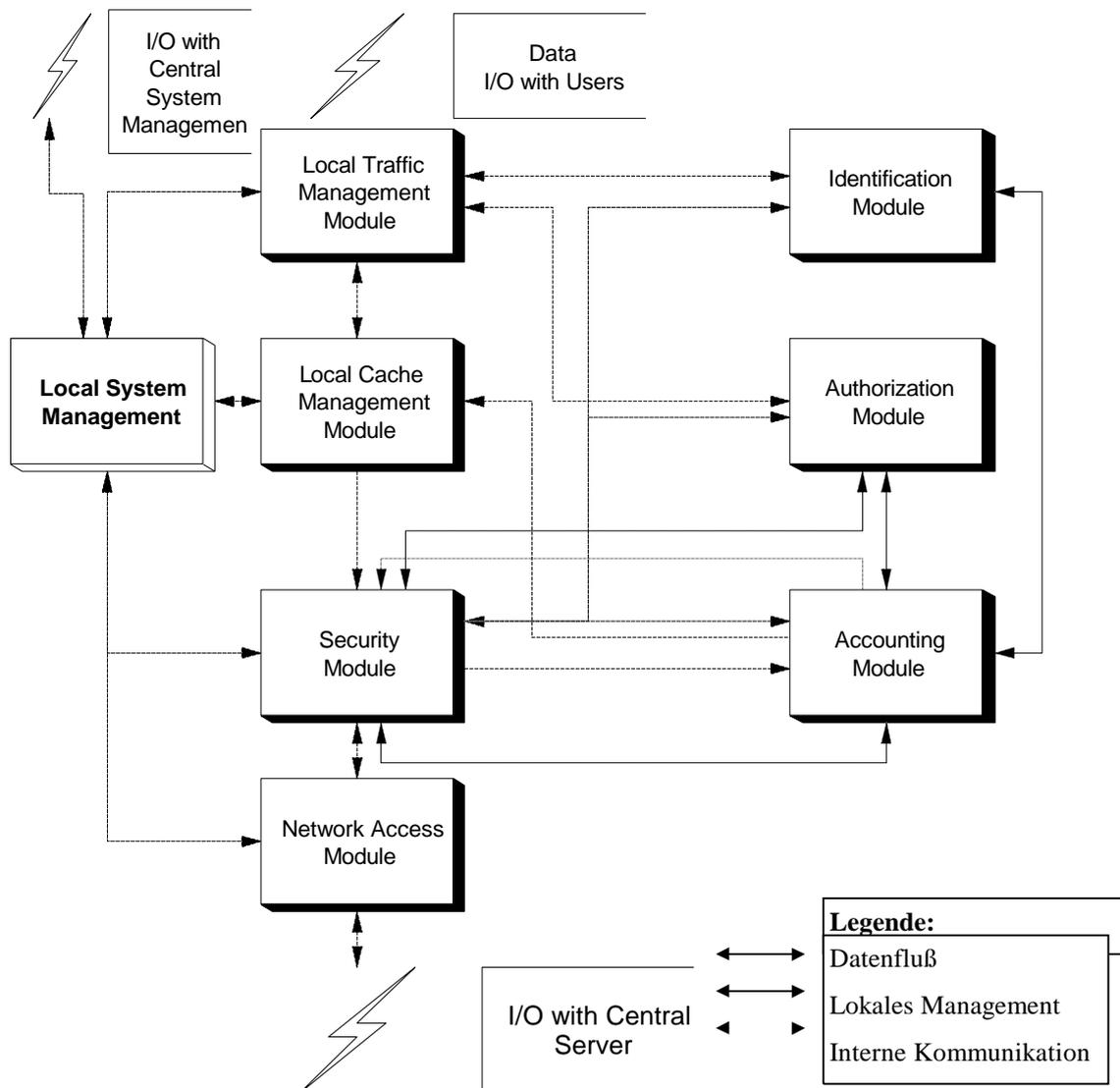
zylindrischen Projektion gewonnen werden können. Da die zylindrische Projektion in den Polgebieten einer starken Verzerrung unterworfen ist, wird für die Polgebiete jeweils eine eigene Karte gespeichert, die mit der Äquatorkarte entsprechend überlappen soll.

Die Karten sollen in verschiedenen Auflösungsstufen verfügbar sein, wobei die niedrigste Auflösung auf dem Bildschirm einen Überblick über die gesamte Planetenoberfläche geben soll. Die Speicherung könnte in einer Auflösungspyramide erfolgen. Bei den Magellan Daten hat sich gezeigt, daß wenn die beste Auflösung einer 1:4 Reduktion entspricht, das Datenmaterial der gesamten Oberfläche weniger als 1 GByte ausmacht (siehe Kapitel 4.1.1). In dieser Auflösungspyramide kann der Benutzer mit Hilfe des Oberflächenbrowsing-Moduls (Map Browsing Modul) beliebig navigieren. Das Diskzugriffs-Modul (Disk Access Modul) ist dafür verantwortlich, daß die jeweils gerade benötigten Bildkacheln rechtzeitig an den jeweiligen Benutzer geschickt werden. Dies ist eine äußerst zeitkritische Angelegenheit, die parallele Bearbeitung von Anfragen ist mit großer Sicherheit notwendig bzw. in der Architektur des Moduls zu berücksichtigen.

**Dateneinspiel-Modul (Data Input Module):** Da GDSS ein offenes System ist, sollte es auch jederzeit um weitere Bilddaten (oder Archive) ergänzt werden können. Hierzu könnte GDSS eine Standardschnittstelle anbieten, die GQP Anfragen der Kategorien II (Daten Wartung) und IV (Administrative Anfragen) versteht. Die Daten Wartung wird vor allem zur Ergänzung von bereits vorhandenen Daten gebraucht, wobei bei Erweiterung um ein komplettes Archiv zusätzlich noch die administrativen Anfragen benötigt werden. Diese umfassen u.a. Identifikation, Managementaufgaben (Art der Netzwerk Verbindung, vorhandene Ressourcen, Anzahl der lokalen Benutzer) und Schaffung einer virtuellen Verbindung.

Bei einer Datenergänzung ist zu beachten, daß nicht nur die Bilddaten des zentralen Servers, sondern vermutlich auch die räumliche Datenstruktur bzw. die Metadaten des Datenbanksuch-Moduls (Database Search Modul) geändert werden müssen.

**GDSS Systemmanagement:** Wie bereits in der Anforderungsspezifikation in Kapitel 3.1 (Funktionalität) festgelegt, wird die Managementaufgabe in eine lokale und eine globale Komponente aufgegliedert. Das lokale Systemmanagement ist für die Verwaltung aller Komponenten des zentralen Servers verantwortlich. Dies umfaßt: Performance Tuning, Konfigurationsmanagement, Reaktion auf lokale Ausfälle und Sicherheitsaspekte.



**Abbildung 3.6:** Die Hauptmodule des lokalen Servers - die Abbildung stellt den Datenfluß, die interne Kommunikation und das lokale Management dar. Das extra hervorgehobene Local System Management Modul unterliegt dem Systemadministrator.

Weiters ist im zentralen Server auch das globale Management untergebracht. Es sollte alle globalen Komponenten von GDSS überwachen. Die Aufgaben des zentralen Managements umfassen somit: Netzwerkmanagement des GDSS Backbones, Ausfallüberwachung von Systemkomponenten und entsprechende Maßnahmen hierzu, globales Performance Tuning, sowie Konfigurationsmanagement.

### 3.3.2 Design des lokalen Servers

In Abbildung 3.6 ist der prinzipielle Aufbau des lokalen Servers dargestellt. Die

verschiedenen Linienarten der Pfade repräsentieren Daten -, Management - und interne Kommunikation. Zur Verwaltung der lokalen internen Daten und Benutzer Kennwerte und zum Informationsaustausch zwischen den Modulen wird die Verwendung eines relationalen DBMS vorgeschlagen. Entsprechende Schema/Tabellen Vorschläge werden im folgenden präsentiert.

**Lokales Verkehrsmanagement Modul (Local traffic management module):** Dies ist die oberste Schicht des lokalen Servers und somit die direkte Schnittstelle zu den Benutzern. Die Aufgaben dieses Moduls umfassen:

- Anmeldung beim Identifikationsmodul,
- Autorisierung,
- Speicherung der Verkehrslastparameter jedes Benutzers und dynamische Änderung,
- Aufbau von neuen Verbindungen,
- Intelligentes<sup>1</sup> Aufteilen der Benutzerverbindungen zum lokalen Cache Modul.

Die erste Aufgabe des Moduls, sobald ein neuer Benutzer sich in das System einloggt ist die Verbindungsaufnahme mit dem Identifikations- und dem Autorisierungsmodul (siehe unten). Sodann werden die Verkehrslastparameter (ein entsprechender Vorschlag ist Tabelle 3.2 zu entnehmen) festgelegt, wobei eine Kontrolle auf Autorisierung erfolgt.

<b>User</b>	<b>Maximale Bandbreite</b>	<b>Durchschnittliche Bandbreite</b>	<b>max. Zellen Verzögerung</b>
<b>U1</b>	1 Mbit/sec	500 kbit/sec	1 sec
<b>U2</b>	40 kbit/sec	25 kbit/sec	0,1 sec

**Tabelle 3.2:** Beispiel einer Verkehrslasttabelle für den lokalen Server

**Identifikations Modul (Identification module):** Dieses Modul ist für die Erfassung der Benutzer Daten zuständig., Hier sollen alle statischen Benutzerdaten werden hier gespeichert werden. In Tabelle 3.3 ist ein Beispiel Schema angeführt.

---

<sup>1</sup> intelligentes Scheduling, also Weiterleiten je nach festgelegten Verkehrslastparametern, ist nur notwendig, wenn das zugrunde liegende Netzwerk das nicht unterstützt. Basiert das System z.B auf einem ATM Netzwerk, entfällt der Scheduler. In diesem Falle werden einfach sovielen Verbindungen aufgebaut, wie User am System liegen.

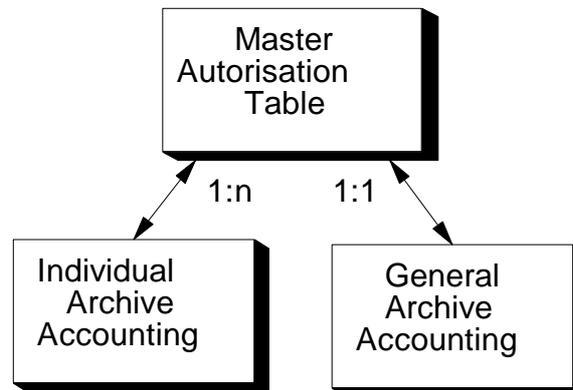
<u>User</u>	Name	Login	Paßwort	Adresse	Büro
U1	Herwig	wigl	xxxxxx	Joanneum	224
U2	Vanessa	Vani	yyyyyy	Hyperwave	12

Tabelle 3.3: Beispiel für eine Userdatentabelle

**Autorisierungs Modul (Authorization module):** Die Autorisierungsrechte legen einerseits die lokalen Schreib-/Leserechte fest, andererseits die maximalen Bestellkosten von Archiven und die max. Netzwerkkosten, die ein Benutzer in Anspruch nehmen darf, ohne tatsächlich zu bezahlen. Diese Daten können in drei Tabellen gespeichert werden, wobei ein entsprechendes Datenbank Schema in Abbildung 3.7 dargestellt ist.

In der Mastertabelle (siehe Tabelle 3.4) werden allgemeine Autorisierungsrechte gespeichert. *Lokale Schreibrechte* legen fest, wo und wieviel ein Benutzer auf seiner lokalen Platte speichern darf. Die *Priorität* regelt die Transfargeschwindigkeit von bestellten Datenmengen bzw. die Zugriffsgeschwindigkeit, wenn das Netzwerk überlastet ist. Der maximale Bandbreiten-Parameter schränkt den Benutzer in der Definition seiner Bandbreite bei einer neuen Verbindung ein. *Diemax. Netzwerkkosten* bestimmen die Zeit, die ein Benutzer das kostenpflichtige öffentliche Netz in Anspruch nehmen darf. Und die max. Bestellkosten legen fest, wieviel der Benutzer für kostenpflichtige Services ausgeben darf. Der *Archive Indiv/Gen* Eintrag sagt, ob der Benutzer individuell für jedes Archiv bestimmte Autorisierungsrechte besitzt, oder ob nur einmal Rechte vergeben wurden, die dann für alle an das GDSS angeschlossenen Archive gelten. Entsprechend dieses Eintrages ist dann entweder die individuelle Accounting Tabelle oder die generelle Accounting Tabelle gültig.

Die individuelle Tabelle unterscheidet sich von der generellen nur dadurch, daß als zusätzlicher Schlüssel auch noch die Archiv Nummer verwendet wird, um die Rechte individuell für jedes Archiv speichern zu können. Das *Daten bis max. \$* in Kombination mit den *im Zeitraum* und *letzte Bestellung* Feldern geben an, wieviel der Benutzer in einem festgelegten Zeitraum (z.B. 1 Monat) ausgeben darf. Hierzu wird einfach verglichen ob seit der letzten Bestellung und der aktuellen Bestellung schon mehr als die im Zeitraum vermerkten Zeit verstrichen ist. Wenn ja, wird der aktuelle Betrag auf Null gesetzt und von den *aktuellen Servicekosten* (siehe Tabelle 3.7) addiert. Der neue Wert wird von den *max. Bestellkosten* subtrahiert, ist das Ergebnis negativ so hat der Benutzer seine maximalen Bestellkosten bereits überschritten, die Bestellung wird verweigert. Ist der Zeitraum noch nicht verstrichen, werden die Kosten der aktuellen Bestellung zuerst zum aktuellen Betrag addiert und die neue Zahl vom *Daten bis max. \$* Feld subtrahiert. Ist der Wert negativ, hat der Benutzer sein für einen Zeitraum festgelegtes Kontingent überschritten, die Bestellung wird verweigert. Ansonsten erfolgt wiederum der



**Abbildung 3.7:** Autorisierungstabellen - Entity Relationship Diagramm. Die Autorisierung erlaubt eine Archiv individuelle oder eine generelle Berechtigung.

Vergleich mit den *max. Bestellkosten*, wie oben beschrieben. Der Betrag des *Daten bis max. \$* Feldes muß natürlich kleiner sein als *diemax. Bestellkosten*. Ist die Differenz der *akt. Servicekosten* von den *max. Bestellkosten* unter einen gewissen kleinen Schwellwert abgesunken, muß der Account vom Systemadministrator wieder neu „aufgeladen“ werden, d.h. die aktuellen Servicekosten sind auf Null zu setzen.

Autorisierungsrechte werden vom lokalen Systemadministrator festgelegt und können jederzeit verändert werden.

**Verrechnungs Modul (Accounting module):** Dieses Modul nimmt zwei wesentliche - der Verrechnung dienende - Aufgaben wahr. Die erste Aufgabe ist anfallende Netzwerkkosten pro Benutzer aufzusummieren und in der Verrechnungstabelle (siehe Tabelle 3.4) zu speichern.

<b>User</b>	<b>Lokale Schreibrechte</b>	<b>Priorität</b>	<b>Archive Indiv/Gen</b>	<b>Max. Bandbreite pro Vbdg.</b>	<b>Max. Netzwerk-kosten ATS</b>	<b>Max. Bestell-kosten</b>
<b>U1</b>	500 MB	1	Indiv	200 kbit/sec	10.000	7000
<b>U2</b>	200 MB	5	Gen	2 kbit/sec	5.000	2000

**Tabelle 3.4:** Master Tabelle für die Autorisierung. Hier werden die lokalen Schreibrechte, die Userpriorität, max. zu reservierende Bandbreite, max. verfügbare Netzwerkkosten und Bestellkosten vermerkt

<u>User</u>	<b>Daten bis max. ATS</b>	<b>Im Zeitraum</b>	<b>letzte Best. am</b>	<b>Aktueller Betrag</b>
<b>U2</b>	3000	1 Monat	02-04-96	500

**Tabelle 3.5:** *Generelle Autorisierung* - hier kann man für alle Archive die pro Benutzer verfügbare Gesamtbestellsumme und den Zeitrahmen, in dem dieser Betrag ausgegeben werden kann, festlegen.

<u>User</u>	<u>Archiv Nummer</u>	<b>Daten bis max. ATS</b>	<b>Im Zeitraum</b>	<b>letzte Best. am</b>	<b>aktueller Betrag</b>
<b>U1</b>	237	70000	5 Monate	09-05-69	3000

**Tabelle 3.6:** *Nach Archiven Individuelle Verrechnungstabelle.* Man kann für jedes Archiv extra den maximalen Betrag, und den Zeitrahmen, in dem dieser Betrag ausgegeben werden darf, festlegen.

<u>User</u>	<b>Akt. Nettwerkkosten</b>	<b>Akt. Servicekosten</b>
<b>U1</b>	1000	500
<b>U2</b>	100	700

**Tabelle 3.7:** *Tabelle mit den Gesamtkosten.* Diese ergeben sich aus der Summe von anfallenden Nettwerkkosten und Servicekosten (das sind Datenbestellungen und Inanspruchnahme von Diensten).

Die zweite Aufgabe ist die Verrechnung von in Anspruch genommenen Diensten - dies sind kostenpflichtige, hochauflösende Bilddaten als auch Dienste wie z.B. Shape-from-Shading, die Berechnung einer Durchflugssequenz als MPEG oder die Klassifizierung von Bilddaten mit Hilfe von teurer Fernerkundungssoftware.

Die Gesamtkosten ergeben sich aus der Summe dieser beiden Beträge. Die tatsächliche Bezahlung könnte durch eine digitale Signatur des Benutzers und des Systemadministrators erfolgen, welcher letzten Endes für den ordnungsgemäßen Zahlungsablauf seiner gesamten Lokalität verantwortlich ist. Die bezahlten Beträge werden von den Summen in der Tabelle subtrahiert, die Werte repräsentieren also immer die Benutzerschulden.

Zusätzlich zur Kostenverrechnung sollen noch wichtige Benutzeraktionen in der Aktionstabelle (siehe Tabelle 3.8) mitprotokolliert werden. Diese Aktionen werden vom Systemadministrator festgelegt und sollen nicht der Überwachung von Benutzern dienen. Alleiniger Zweck ist es bei Streitfragen (in der Regel Kostenfragen, unbeabsichtigtes Löschen von lokalen Daten, ...) eine Dokumentation über Benutzeraktionen zu haben.

<u>User</u>	<u>Zeit</u>	<u>Datum</u>	<u>Aktionsbeschreibung</u>
U1	10:33	09-05-69	Download von EOSDIS Bild xy
U2	22:02	02-04-96	Höhenmodell Berechnung am ICG, 2 Stunden Rechenzeit und 300 MB an Speicherplatz

**Tabelle 3.8:** *Aktionstabelle* - hier werden wichtige (vom Systemadministrator zu definierende) Benutzeraktionen mitprotokolliert. Dies dient dem Systemadministrator zur einfacheren Wartung des Servers.

**Sicherheits Modul (Security module):** Der gesamte I/O Verkehr des lokalen Servers sollte durch ein Sicherheitsmodul (Fire-Wall) fließen. Hier werden alle hereinkommenden Pakete gefiltert und analysiert. Das Sicherheitsmodul soll vom Administrator zu definierende Aktionen mitprotokollieren - diese können z.B. erfolglose Login Versuche (Hackversuche), Versuche von außen in das System zu gelangen, unerlaubtes Ändern von Systemtabellen (Überprüfung mit CRC Summe), aber auch die GDSS-systemspezifischen Aktionen (etwa eine Cache Auffrischung) umfassen. Dieses Modul zieht zwangsläufig einen Performance Verlust mit sich und ist optional. Vor allem öffentliche Forschungseinrichtungen und Universitäten werden diesen Schutzwall nicht benötigen, für kommerzielle Anwender wird er hingegen unerlässlich sein.

**Lokales Cache Management Modul (Local cache management module):** Vor allem während des Suchvorganges kann eine Unzahl von Bildausschnitten vom zentralen Server über das Netz geladen werden. Es bringt einen großen Geschwindigkeits- und Kostenvorteil, wenn man die angeforderten Bilddaten lokal speichert - vor allem zu Zeiten wo das Netzwerk zu einem hohen Prozentsatz ausgelastet ist. Erste Berechnungen für den Magellan Datensatz haben ergeben, daß die Datenmenge für Oberflächenkarten des Planeten Venus in acht Detailstufen weniger als 1 GByte ausmacht. Diese für heute verfügbare Speichermedien an und für sich kleine Datenmenge könnte man leicht lokal zwischen speichern. Es bleibt jedoch anzumerken, daß die Datenmenge sehr rasch explodieren kann, wenn man bedenkt, daß mehrere Benutzer auf mehrere Planetendaten zugreifen bzw. es natürlich auch Sinn macht, Quicklooks von geokodierten Bildern lokal zu halten.

Es ist Aufgabe des Systemadministrators, die lokale Cachestrategie und den zur Verfügung stehenden Plattenplatz einzuteilen. Dies kann auch nach Benutzern individuell erfolgen. Unter „Strategie“ versteht man die Methode, welche Daten im Cache gelöscht werden, wenn der verfügbare Speicherplatz aufgebraucht ist. Hier bietet sich z.B. an, die am wenigsten benötigten Daten oder die ältesten Daten zu überschreiben.

Eine kritische Frage jeder Cachestrategie ist das Erneuerungs (Update) Problem. Sobald im zentralen Server Änderungen an den Oberflächenkarten bzw. an den

Quicklooks vorgenommen wurden, müssen alle lokalen Server darüber informiert werden, so daß sie die eventuell gespeicherten, nun nicht mehr gültigen Daten aus dem Cache entfernen. Hierbei ist vor allem darauf zu achten, das Netzwerk nicht zu sehr zu belasten. Zusätzlich bleibt anzumerken, daß planetarische Daten nur sehr geringfügigen Änderungen unterliegen - d.h. pro Satellitenmission wird ein Datensatz erzeugt, der dann weitestgehend statisch ist. Ein sehr ähnliches Update Problem tritt bei dem in Graz entwickelten Hyper-G System [Kappe, 93] auf. Auch hier wird beim Update davon ausgegangen, daß Daten i.a. sehr viel öfter gelesen als geschrieben (verändert) werden. Die Lösung erfolgt dort mit dem sogenannten p-flood Algorithmus [Kappe, 95]. Selbiger könnte auch im GDSS Verwendung finden.

**Lokales Systemmanagement (Local system management):** Das lokale Systemmanagement soll einerseits Schaltzentrale für den lokalen Systemadministrator sein, von der alle bisher genannten Module sowie die Systemdatenbank (deren Tabellen im vorigen Abschnitt exemplarisch angedeutet wurden) verwaltet. Andererseits soll hier die Schnittstelle zum globalen Systemmanagement zu finden sein. Alle Probleme, Aktionen, Konfigurationen die globale Bedeutung haben (z.B. eine Systemabschaltung wegen lokaler Wartungsarbeiten) werden an das globale Systemmanagement weitergeleitet. Dort können dann entsprechende globale Maßnahmen getroffen werden. Im Falle des Ausfalls eines lokalen Servers z.B. könnten etwa alle noch an den lokalen Server zu schickenden Daten zwischengespeichert und zu einem späteren Zeitpunkt geschickt werden.

### 3.4 Österreichweites GDSS Szenario mit mehreren Servern

Exemplarisch ist in Abbildung 3.8 ein österreichweites GDSS Szenario dargestellt. Das Szenario wurde unter Verwendung der in Kapitel 3.2.1 vorgeschlagenen Architektur unter der speziellen Berücksichtigung der in Österreich vorhandenen Infrastruktur erstellt. In Erweiterung des vorgestellten Konzeptes wurde ein GDSS mit mehreren zentralen Servern aufgebaut (Salzburg, Wien, Graz).

Die zentralen Server sind untereinander mit dem Hochgeschwindigkeitsnetzwerk der PTA verbunden. Die Daten der zentralen Server werden jeweils auf die anderen Server gespiegelt, somit hat im Prinzip jedes Bundesland einen schnellen Zugang zum System. Archivdaten verweilen in den jeweiligen Archiven und werden nicht gespiegelt, die hochauflösenden Daten werden nur auf spezielle Anfrage (u.U. kombiniert mit einer Bezahlung), und nicht während eines Suchprozesses geliefert.

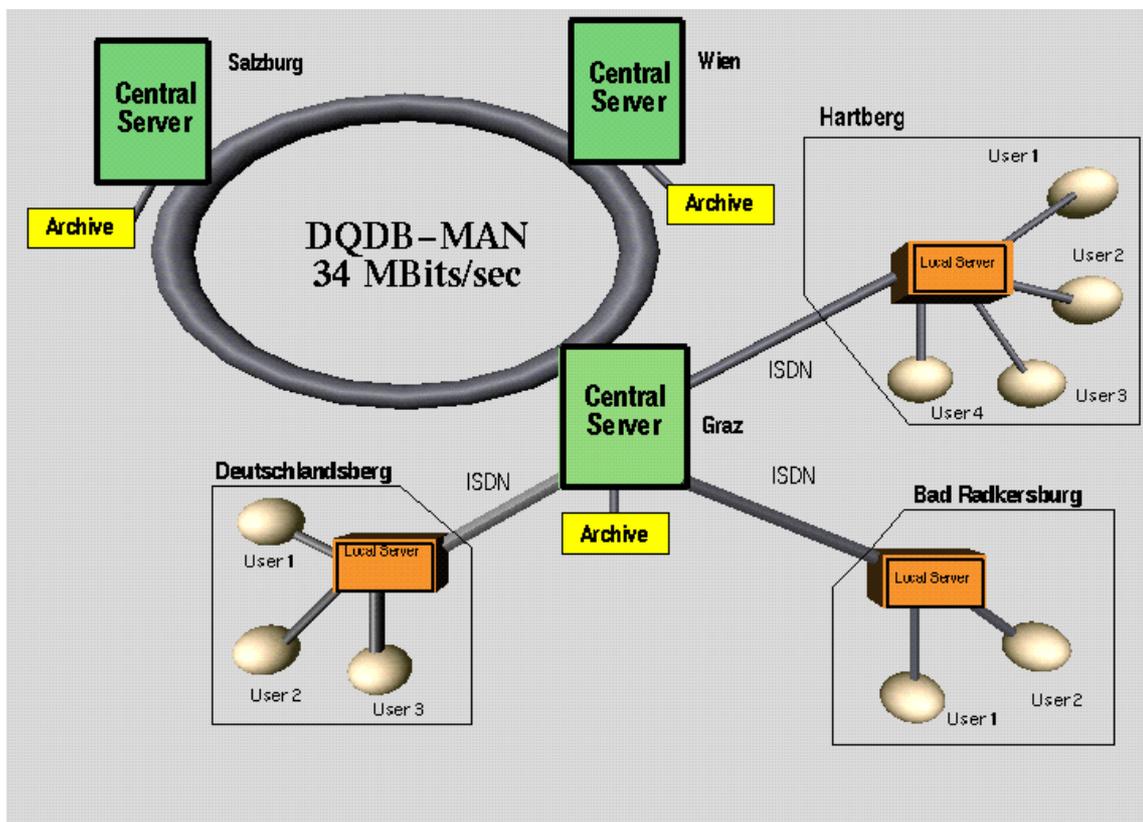
Die Aufteilung der zentralen Datenserver auf die Landeshauptstädte ergibt sich intuitiv, da sich in Österreich dort die teuersten Ressourcen und die beste Netzwerkanbindung befinden. Weiters befinden sich auch die größeren Bildarchive zumeist in den großen Städten, etwa 2D GIS Grundstückdaten, Magellan Daten,

usw.

Wie aus Abbildung 3.8 ersichtlich, sollen die einzelnen lokalen Server sternförmig am zentralen Server angeschlossen werden. Jeder Bezirk, der in GDSS integriert werden möchte, könnte etwa einen lokalen Server betreiben, der Benutzer Management, Verrechnung und Caching übernimmt. Aufgabe des lokalen Servers ist es also, eine weitere Stufe der Dezentralisierung zu schaffen und durch geeignete Cachingstrategien den Netzwerkverkehr deutlich zu entlasten.

Die Klient-Hardware sollte äußerst geringe Kosten verursachen. Idealerweise (dem GDSS Entwurfskriterium folgend: billige Hardware lokal, teure Hardware zentral) sollten lediglich eine grafikfähige Maschine (PC) und ein ISDN Anschluß benötigt werden. Der Zugang zum System ist durch die Datenorganisation des GDSS im zentralen Server für den Benutzer völlig transparent, d.h. er muß sich nicht darum kümmern, wo die von ihm benötigten Daten nun tatsächlich verweilen.

Im voll implementierten GDSS System wählt der Benutzer nun komfortabel im GUI des Klienten das Land und das Gebiet aus, an dem er interessiert ist, und formuliert eine entsprechende Abfrage. Ab da beginnt der zentrale Server zu arbeiten. Er holt sich aus den angeschlossenen Archiven die Daten, welche den Abfragekriterien genügen, schneidet exakt jene Gebiete aus, die der Benutzer definiert hat, und schickt den Bildstapel (oder auch nur ein Bild) über das Netz wieder an den



**Abbildung 3.8:** GDSS Szenario für Österreich. Die Anbindung der Länder erfolgt über ISDN Verbindungen

Benutzer retour.

### **3.5 Anwendbarkeit des GDSS Konzeptes auf MISSION**

Die MISSION (Multi-Image Synergistic Satellite Information for the Observation of Nature)-Initiative wurde in Österreich zur Erschließung von neuen Anwendungen mit hochauflösenden Satellitenbilddaten ins Leben gerufen, wobei Forscher mit potentiellen Fernerkundungsanwendern zusammengeschlossen sind [Mission, 97]. Die 9 MISSION-Projekte wurden durch die in der Arbeitsgruppe Fernerkundung (AGFE) zusammengeschlossenen österreichischen Fernerkundler und deren Anwender-Partner definiert. Zusätzlich zu den Eigenleistungen der Teilnehmer erfolgt die Finanzierung der Arbeiten an den Forschungsstellen größtenteils durch das Wissenschaftsministerium BMWFK.

In den Jahren 1996/97 wird das deutsch-russische Projekt MOMS-02/Priroda multispektrale Bilder mit Auflösungen bis zu 6 Meter/Pixel liefern. Diese Daten gilt es in angemessener Zeit in dieser Arbeitsgruppe zu verteilen. Weiters wäre es wünschenswert, Methoden zur kooperativen, interaktiven Bearbeitung dieser Daten zur Verfügung zu stellen. Im Rahmen der GDSS Pilotversuche wird im Kapitel 6 dieser Arbeit ein Werkzeug namens „X-wedge“ vorgestellt, welches basierend auf einer entsprechenden ATM Infrastruktur genau diese Funktionalität anbietet.

Neben der Erschließung von Anwendungen räumlich hochaufgelöster Satellitendaten ist die Methodenentwicklung für die Auswertung zeitlich hochaufgelöster Satellitendaten, wie sie z.B. von geostationären meteorologischen Satelliten geliefert werden, von äußerster Wichtigkeit. MSG (Meteosat Second Generation) wird Bilder in ca 6 Kanälen und in 15 Minuten Abständen liefern, was eine enorme Erhöhung des Datenflusses gegenüber dem derzeitigen Stand von 3 Kanälen und 30 Minuten bedeutet. Bereits jetzt ist es unmöglich, diese Informationsflut auszuwerten, geschweige denn unter den Partnern in angemessener Zeit zu verteilen. Viele der in Kapitel 3 vorgestellten Konzepte, die sich primär auf extraterrestrische Daten beziehen, sind jedoch auch hier anwendbar. Der im Kapitel 5 vorgestellte GDSS Prototyp müßte hierzu an die Metadatenstruktur bzw. Quicklook Darstellung von MISSION Daten angepaßt werden.

### **3.6 Zusammenfassung und Wertung**

Dieses Kapitel beschäftigt sich mit dem Konzeptentwurf eines verteilten, planetarischen Bildarchivs basierend auf einem leistungsfähigen Netzwerkbackbone. Als Testdatensatz werden die Magellandaten vom Planeten Venus der Weltraumbehörde NASA verwendet werden. Die bevorzugte Technologie, die auch aus dem Titel der Arbeit bereits hervorgeht, ist ATM.

Im ersten Abschnitt wurden auch zahlreiche Entwurfskriterien für ein globales Fernerkundungsdaten System vorgestellt und argumentiert, wie sie etwa im Projekt

MISSION auch verwendet werden könnten. Nicht alle dieser Kriterien haben dieselbe Priorität, ein entsprechender Prototyp des Systems läßt sich auch mit einer Untermenge derselben verwirklichen.

In einer „Top-Down“ Manier wurde zuerst eine allgemeine Systemarchitektur vorgeschlagen, welche dann schrittweise verfeinert wurde. Zu den meisten Systemkomponenten wurden auch entsprechende Implementierungshinweise gegeben bzw. auf bereits vorhandene Implementierung ähnlicher Systeme hingewiesen.

Die vorgestellten Konzepte sind als eine Art Skelett und Beginn für ein voll funktionstüchtiges, globales System zu sehen. Um das gesamte Projekt zu implementieren, bedarf es natürlich noch viel mehr Detailarbeit. So müssen etwa die vorgestellten Module detaillierter entworfen werden und eindeutige Schnittstellen definiert werden. Ziel dieser Arbeit war es jedoch nicht, ein voll funktionstüchtiges System zu entwerfen und zu implementieren, sondern vielmehr durch die Implementierung eines Labor Prototypen, der auf einigen wesentlichen Entwurfskriterien des Entwurfskonzeptes basiert, einen Weg für die Zukunft zu weisen.

Dieser Prototyp kann natürlich nur die wichtigsten der vorgestellten Konzepte und Entwurfskriterien berücksichtigen. Folgende Entwurfskriterien werden als unerlässlich erachtet:

- Einhaltung eines international gültigen Abfrage Standards,
- plattformunabhängiger Benutzer Klient mit GUI,
- leistungsfähiges Netzwerk (leicht skalierbar, Dienste Spezifikation),
- Lösung des Update Problems falls neue Klient Programm Version kommt,
- Anbindung einer beliebigen relationalen Datenbank an den zentralen Server,
- um globale Vernetzung zu gewährleisten, muß das System auf TCP/IP aufsetzen,
- objektorientiertes Design,
- einheitliche Abfrage, unabhängig von der im zentralen Server verwendeten Datenbank.

Die Einhaltung dieser Gesichtspunkte bestimmten in weiterer Folge den Fortgang des Projektes.

### 3.7 Literatur und Referenzen

- [ANSI, 92] ANSI/NISO Z39.50-1992, Information Retrieval Service and Protocol: American National Standard, Information Retrieval Application Service Definition and Protocol Specification for Open Systems Interconnection, 1992.  
<URL: <ftp://ftp.cni.org/pub/NISO/docs/Z39.501-992/www/Z39.50.toc.html>>, auch als Hardcopy erhältlich von NISO Press Fulfillment, P.O. Box 338, Oxon Hill, Maryland 20750-0338; phone 800-282-6476 or 301-567-9522; Fax: 301-567-9533.
- [ANSI, 95] ANSI/NISO Z39.50-1995, Information Retrieval Service and Protocol, 1995.  
<URL: <http://lcweb.loc.gov/Z3950/agency>>
- [Berson, 96] Berson S. „Classical" RSVP and IP over ATM“. INET '96, April 1996,  
<URL: <http://www.isi.edu/div7/rsvp/pub.html>>
- [Braden, 96] Braden R., Zhang L., Berson S., Herzog S., Jamin S., „Resource ReSerVation Protocol (RSVP) -- Version 1 Functional Specification“. Internet Draft, November 1996,  
<URL: <http://www.isi.edu/div7/rsvp/pub.html>>
- [CEOS, 95] CEOS, „Catalogue Interoperability Protocol (CIP) Specification - Release A“, Version 1.2, März 1996, <URL: <ftp://styx.esrin.esa.it/pub/od/CIP>>
- [CINTEX, 96] CINTEX (CEOS Catalogue Interoperability Experiment), Homepage ESRIN Server,  
<URL: [http://gds.esrin.esa.it:80/0xc06afc3d\\_0x00039ce6](http://gds.esrin.esa.it:80/0xc06afc3d_0x00039ce6)>
- [Demirtjis, 96] Demirtjis A., Berson S., Edwards B., Maher M., Braden B., Mankin A., „RSVP and ATM Signalling“. ATM Forum/96-0258, <URL: <http://www.isi.edu/div7/rsvp/pub.html>>
- [Herzog, 95] Herzog S., Shenker S., Estrin D. „Sharing the Cost of Multicast Trees: An Axiomatic Analysis“, Proceedings of ACM SIGCOMM '95,  
<URL: <http://www.isi.edu/div7/rsvp/pub.html>>
- [HyperW, 97] HyperWave R&D, Homepage., Jän. 1997, <URL: <http://www.hyperwave.com>>
- [IDN, 96] CEOS IDN (The Committee on Earth Observation Satellites International Directory Network), Homepage, <URL: <http://gcmd.gsfc.nasa.gov/ceosidn/>>
- [Kappe, 93] Kappe F. „Hyper-G: A Distributed Hypermedia System“. Proc. INET'93, San Francisco, California, pages DCC-1 - DCC-9, Internet Society, August 1993
- [Kappe, 95] Kappe F. „A Scalable Architecture for maintaining Referential Integrity in Distributed Information Systems“. JUCS, Vol.1 No.2, Feb. 28, 1995
- [Kunze, 95] Kunze J. A. „Basic Z39.50 Server Concepts and Creation“. University of California at Berkeley and U.S. National Library of Medicine, unpublished,  
<URL: <http://lcweb.loc.gov/Z3950/agency>>
- [Minoli, 94] Minoli D., Keinath R., 1994. „Distributed Multimedia Through Broadband Communications“, Artech House, ISBN 0-89006-689-2
- [Mitzel, 96] Mitzel D., Estrin D., Shenker S., Zhang L. „A Study of Reservation Dynamics in Integrated Services Packet Networks“, Proceedings of Infocom '96,  
<URL: <http://www.isi.edu/div7/rsvp/pub.html>>
- [Mission, 97] MISSION (Multi-Image Synergistic Satellite Information for the Observation of Nature) Projekt, <URL: <http://www.icg.tu-graz.ac.at/MISSION>>, Homepage, Jänner 1997, Projektverantwortliche am ICG: DI Franz Niederl ([niederl@icg.tu-graz.ac.at](mailto:niederl@icg.tu-graz.ac.at)) und DI Rainer Kalliany ([kalliany@icg.tu-graz.ac.at](mailto:kalliany@icg.tu-graz.ac.at)).
- [Mull, 94] Mullender S. „Distributed Systems“, Addison Wesley, ISBN 0-201-62427-3, 1994

- [Pani, 93] Pani G., Kappe F., Schnabel F. „The Architecture of a Massively Distributed Hypermedia System“. Internet Research: Electronic Networking Applications and Policy, 3(1):10-24, Spring 1993
- [Ragetti, 95] Ragetti D. „HyperText Markup Language Specification 3.0“, Internet Draft, Feb. 95, <URL: <http://www.w3.org/hypertext/WWW/MarkUp/html3/CoverPage.html>>
- [Vaughn, 94] Vaughn L., „Client/Server System Design and Implementation“. ISBN: 0-07-067375-6, McGraw Hill, 1994
- [Walcher, 97] Walcher W. „Design Aspects of Information Systems for Planetary Image Data“. PhD thesis, Institute for Computer Graphics, University of Technology, Graz-Austria, Jan. 1997



## 4. Kapitel

### ATM Netzwerk Backbone

*„Advance in science comes  
by laying brick upon brick  
not by sudden erection,  
of fairy palaces“*  
J.S. Huxley

Wir behandeln die Beschreibung und den Aufbau des Netzwerk Backbones im Hinblick auf die in Kapitel 3 vorgestellten Konzepte. Dies erfolgt in vier Schritten, welche den vier Unterkapiteln entsprechen.

Die Netzwerk Anforderungsanalyse beschäftigt sich mit der Netzwerkdefinition. Beginnend mit einem Anforderungsprofil, über allgemeine, finanzielle bis hin zu den technischen Auswahlkriterien werden die ersten Schritte eines Netzwerkadministrators gezeigt, bevor schließlich die Wahl auf eine bestimmte Architektur und eine Kaufentscheidung gefällt wird. Weiters wird auf den Aspekt des Netzwerkmanagements und die Bedeutung des Internets für GDSS eingegangen.

Das zweite Unterkapitel beschäftigt sich schließlich mit ATM (Asynchronous Transfer Mode), dem eigentlichen Schwerpunkt des 4. Kapitels. Es wird ein selbst erarbeitetes Kostenverrechnungsmodell für ATM Netze vorgestellt. Außerdem wird im Abschnitt 4.2.3 der Aufbau des lokalen ATM Clusters der Technischen Universität Graz beschrieben.

ATM ist nicht der einzige Hochgeschwindigkeits-Netzwerkstandard. Das dritte Unterkapitel beschäftigt sich daher mit Alternativen zu ATM. Es werden Fähigkeiten und Schwächen existierender Standards vorgestellt und anhand einer Bewertungstabelle kritisch gegenüber gestellt.

Nach der eingehenden Analyse von ATM in den ersten drei Unterkapitel, bildet eine Begründung der Wahl von ATM für GDSS den Abschluß dieses Kapitels. Es erfolgt eine Zusammenfassung der herausragenden Merkmale dieser neuen Technologie, sowie eine kritische Gegenüberstellung der Alternativen.

#### 4.1 Netzwerk Anforderungsanalyse

Bevor man sich auf eine bestimmte Netzwerktechnologie festlegt, muß man sich über den Zweck, die Anforderungen an das Netzwerk, über finanzielle und über technische Kriterien Gedanken machen. Dieser erste Schritt, die sogenannte Netzwerk Anforderungsanalyse, beinhaltet eine Definition der zu transportierenden Daten, allgemeine, finanzielle und technische Auswahlkriterien. Diese werden in den folgenden vier Unterkapitel näher behandelt.

### 4.1.1 Anforderungsprofil

Der GDSS Network Backbone muß sowohl Text- als auch Bilddaten vollduplex verschicken können. Vollduplex ist erforderlich, wenn bestellte und örtlich bereits manipulierte Bilddaten an den zentralen Server zurückgeschickt werden um z.B. eine komplexe Operation darauf anwenden zu lassen, die mit der lokalen Hardware in nicht sinnvoller Zeit berechnet werden könnte.

Dieser Bericht beschränkt sich auf ein Auswahlprofil basierend auf den Venus Magellan Daten, die dem Institut gratis zur Verfügung gestellt wurden. Dies ist aber keine große Beschränkung der Allgemeinheit, da die übertragene Datenmenge weniger vom Planeten als von der Speicherungsart abhängt. In Tabelle 4.1 ist das Gesamtvolumen an Bilddaten der Venus Magellansonde zusammengefaßt, in Tabelle 4.2 findet man die zugehörigen Metadaten. Wobei nur letztere tatsächlich in der Datenbank des zentralen Server gespeichert werden müssen, Originalbilddaten verweilen bis zu einer etwaigen Bestellung in den Archiven.

Abbildung 4.1 stellt alle Daten dar, die nun tatsächlich über das GDSS Netzwerk geschickt werden müssen. Wobei eine allgemein gültige Einteilung in Karten-, Original- und Quicklookdaten erfolgt. Die absoluten Angaben im Diagramm beziehen sich wieder auf den speziell in dieser Arbeit zu behandelnden Magellan Datensatz vom Planet Venus.

#### (A) Map Daten

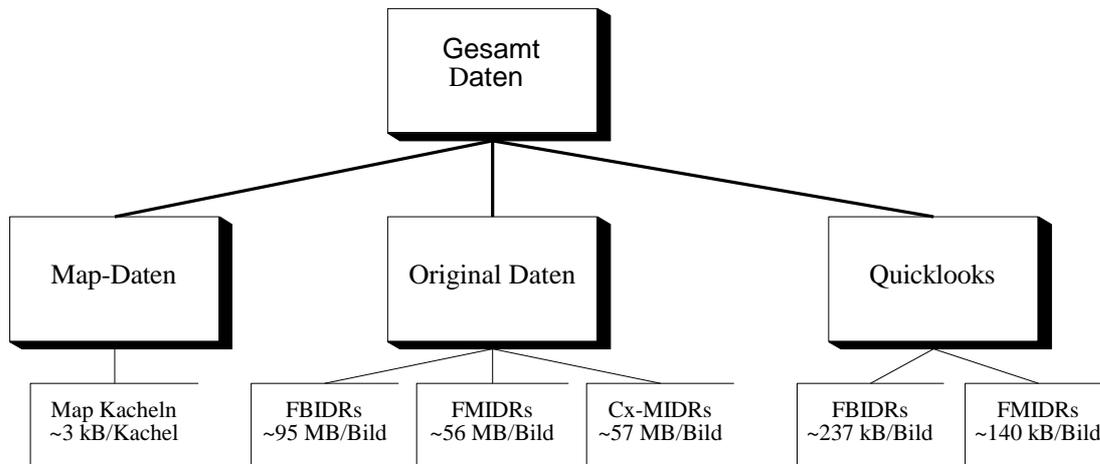
Die Bildkacheln („Map-Tiles“) sind die am häufigsten benötigten Daten. Sie machen die Einzelteile der Browsing Map aus, die von den Benutzern zur On-line Suche verwendet wird. Diese Map besteht in verschiedenen Auflösungsstufen, wobei die Ebenen eine Auflösungsrampe bilden, in die man hinein- und herauszoomen kann. In Abbildung 4.2 ist eine Bildschirmteilung dargestellt, wobei die Kacheln einen Teil einer Auflösungsstufe der Bildpyramide repräsentieren. Der Benutzer bekommt nun, je nachdem ob er hinein- oder herauszoomt bzw. in eine von acht möglichen Richtungen weiterwandert, die entsprechenden Kacheln vom zentralen Server (bzw. vom lokalen Cache) geliefert.

<b>Bezeichnung</b>	<b>Anzahl</b>	<b>Volumen</b>	<b>Beschreibung</b>
<b>FBIDR</b>	5747	400 GB	Full Resolution Basic Image Data Records
<b>FMIDR</b>	220	12.3 GB	Full Resolution Mosaicked Image Data Records
<b>C1-MIDR</b>	184	10.3 GB	Compressed 1 Mosaicked Image Data Records
<b>C2-MIDR</b>	34	2 GB	Compressed 2 Mosaicked Image Data Records
<b>C3-MIDR</b>	12	0.7 GB	Compressed 3 Mosaicked Image Data Records
<b>GxxDR</b>	6	0.2 GB	Altimeter-, Gravitationsdaten u.a.
<b>Summe:</b>	6203	426.21 GB	

**Tabelle 4.1:** Alle von der Venus verfügbaren Bilddaten im Überblick zusammen mit Anzahl und Speicherbedarf

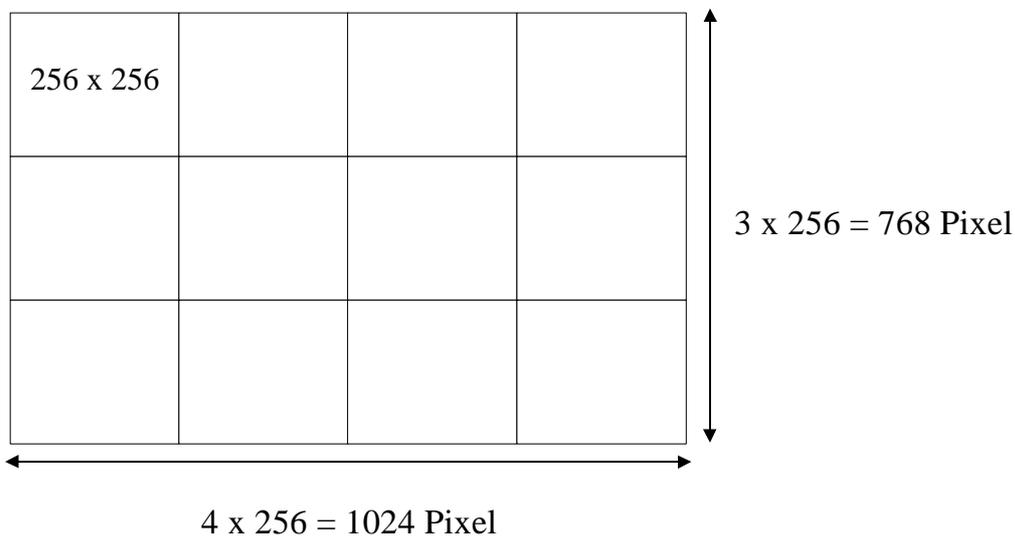
<b>Bezeichnung</b>	<b>Volumen</b>	<b>Beschreibung</b>
<b>Bild Positionsdaten</b>	2 MB	Geokodierung
<b>Engineering Daten</b>	110 MB	Magellan Engineering Workbook mit allen Steuercommandos
<b>Magellan Gazeteer</b>	0.13 MB	Katalog von benannten Venus Lokalitäten
<b>Krater Datenbank</b>	0.3 MB	Katalog von benannten Venus Kratern
<b>Gap Info</b>	1.3 MB	Informationen über Sondenaussetzer und Fehler
<b>Orbit Datenbank</b>	100 MB	Genauere Beschreibung der Flugbahn
<b>Orbit Punkt Info</b>	10 MB	Orbit Parameter: Breitengrad, Blickwinkel, etc.
<b>Summe:</b>	224.1 MB	

**Tabelle 4.2:** Übersicht der für die Bilddaten relevanten Metadaten mit Speicherbedarf. Diese Daten werden in der Datenbank des zentralen Servers gespeichert.



**Abbildung 4.1:** Übersicht der über das Netz geschickten Daten. Die absoluten Angaben in der Abbildung beziehen sich auf den Testanwendungsfall des Magellan Datensatzes vom Planeten Venus.

Ein Standard PC Graphik Bildschirm weist eine Auflösung von 1024 x 768 Pixeln auf. Betrachtet man Kacheln in der Größe von 256 x 256 Pixeln mit 1 Byte Grauwertinfo, so benötigt man 4 x 3 Kacheln, um den Graphikschirm zu füllen. Diese Kacheln hätten dann ein Datenvolumen von 256 Pixel x 256 Pixel x 1 Byte = 64 kB. Komprimiert man alle Magellan Map Kacheln mit JPEG, erhält man eine durchschnittliche Datenreduktion um den Faktor 25, was dem in Abbildung 4.1



**Abbildung 4.2:** Bildschirmeinteilung auf der Benutzer Seite. Die Gesamtauflösung beträgt 1024 x 768. Es wird aber niemals der gesamte Bildschirm zur Darstellung von Bildkacheln verwendet werde, da auch noch entsprechender Platz für Steuerungsmenüs eingeräumt werden muß. Eine typische Bildschirmeinteilung wäre daher entweder 4 x 2 (1024 x 512) oder 3 x 3 (768x768), je nachdem wie der Platz für zusätzliche Informationen verwendet wird.

dargestellten Datenvolumen von ~3 kByte entspricht. Die maximale Auflösung der Magellan Daten beträgt 75 m/Pixel, um jedoch das Datenvolumen zu reduzieren (bzw. ist für einen Suchvorgang auch nicht die volle Auflösung notwendig), wurde die maximale Auflösung auf 300 m/Pixel festgelegt. Dies entspricht einer Reduktion der Auflösung um den Faktor 4 und einer Reduktion der Daten um den Faktor 16. Bei einer insgesamt darzustellenden Fläche von 40.000 km x 20.000 km auf der Venus ergibt das eine Anzahl von 508 x 62 Bildkacheln (á 256 x 256 Pixel) in der besten Auflösung. Die niederste Auflösung setzt sich aus 4 x 3 Kacheln zusammen und beträgt 38,4 km/Pixel.

Das Gesamtdatenvolumina der Kacheln beträgt demnach für festgelegte acht Auflösungsstufen (levels of detail) 541 MB, Details sind Tabelle 4.3 zu entnehmen. Bei den absoluten Byteangaben ist zu berücksichtigen, daß alle Kacheln noch einer JPEG Kompression unterzogen wurden, welche durchschnittlich eine Reduktion von 1:25 bewirkt.

Auflösungs- Stufe	Auflösung in Kacheln	Volumen	Anzahl	Auflösung in m/Pixel
1	508 x 262	426 MB	133 096	300
2	254 x 131	85 MB	33 274	600
3	127 x 66	21 MB	8382	1200
4	64 x 33	6 MB	2112	2400
5	32 x 17	2 MB	544	4800
6	16 x 9	493 kB	154	9600
7	8 x 5	128 kB	40	19200
8	4 x 3	38 kB	12	38400
<b>Gesamt:</b>		541 MB	177 614	

**Tabelle 4.3:** Gesamtdatenvolumen der Map-Bildkacheln für 8 verschiedene Auflösungsstufen sowie einer einmaligen Auflösungsreduktion von 1:4 von ursprünglich 75 m/Pixel auf 300 m/Pixel.

Zusätzlich verlangt die gewählte Projektionsart (zylindrische Projektion) noch eine zusätzliche Speicherung der beiden Polarregionen, wobei das Projektionszentrum jeweils in den Polen gesetzt wird. Dies ist notwendig, da sonst die starke Verzerrung in den Polargebieten einen sinnvollen Suchvorgang nicht mehr zulassen würde. Die zu speichernde Gesamtdatenmenge bleibt jedoch unter 1 GB [Walcher, 95].

## **(B) Original Daten**

Die original Bilddaten des hier betrachteten Magellan Datensatzes gliedern sich in drei Kategorien, wobei die letzte nochmals in drei Subkategorien unterteilt ist. Diese Kategorien sind (siehe auch Tabelle 4.1): FBIDRs, FMIDRs und Cx-MIDRs - wobei es von der letzten Kategorie C1, C2 und C3-MIDRs gibt. Alle drei Kategorien unterscheiden sich im wesentlichen in der Auflösung, in ihrer Form und natürlich auch in ihrem Datenvolumen.

Die **FBIDRs** haben eine Größe von 350 x 220.000 Pixel mit der maximalen Auflösung von 75 m/Pixel, was bei einem Byte radiometrischer Information und einigem Overhead in der Fileinformation ein Datenvolumen von 77 - 95 MB pro FBIDR ergibt. Original Daten verweilen bis zu ihrer Bestellung in den Archiven und werden auch am seltensten im Vergleich zu den Kartendaten oder den Quicklooks über das Netz geschickt. Sie können keiner verlustbehafteten Kompression unterzogen werden, da der Benutzer die Daten in der bestellten Auflösung benötigt. Man kann sie aber sehr wohl einer verlustlosen Kompression unterziehen, was eine Datenreduktion von ca. 1:5 bewirken würde [Walcher, 94] in Abhängigkeit der Bildinformationen. Somit müßten bei einer gesamt FBIDR Bestellung an die 19 MB Daten transportiert werden.

Die **FMIDR** besitzen eine Größe von 7168 x 8192 Pixel bei der vollen Auflösung von 75m/Pixel, wobei sich jede dieser Bildkacheln aus 7 x 8 Quadraten zu je 1024 x 1024 Pixel zusammensetzt. Bei einem Byte Grauwertinformation ergibt das ein Datenvolumen von 7168 x 8192 x 1 Byte = 56 MB. Mit der oben beschriebenen Kompression erreicht man 11,2 MB an zu transportierenden Daten.

Die **Cx-MIDR** sind im Prinzip mit den FMIDRs vergleichbar, nur daß sie in ihrer Auflösung reduziert sind und in ihrer Gesamtheit daher auch ein geringeres Datenvolumen aufweisen. Cx-MIDRs haben also alle dieselbe Größe und interne Organisation - sie bestehen ebenfalls aus 7 x 8 quadratischen Bildkacheln von je 1024 x 1024 Pixel - und weisen natürlich auch dasselbe Datenvolumen wie die FMIDRs von 56 MB auf, was komprimiert wiederum 11,2 MB ergäbe. In der Auflösung unterscheiden sie sich aber signifikant. Als Reduktionsverfahren werden einfach 3x3 Pixel zu einem zusammengefaßt, wobei der neue Grauwert sich aus dem Mittelwert der neun ursprünglichen ergibt. Die Auflösung verringert sich folglich somit jeweils um den Faktor 3, das Gesamtdatenvolumen um den Faktor 9. C1-MIDR haben eine Auflösung von 225 m/Pixel (=3\*75), C2-MIDRs 675 m/Pixel (=3\*3\*75) und C3-MIDRs 2,025 km/Pixel (=3\*3\*3\*75).

## **(C) Quicklooks**

Bevor ein Benutzer eine Bestellung von Originaldaten durchführt reicht es, sogenannte Quicklooks zu präsentieren. Sie sind geokodiert und sollen einen prinzipiellen Eindruck über den Bildinhalt geben. Allerdings sind Quicklooks

hochkomprimiert und in ihrer Auflösung reduziert. Dies bedeutet natürlich einen geringeren Bedarf an Bandbreite und somit eine rasche Ladezeit während eines interaktiven Suchvorganges. An Datenvolumina fallen 237 kB für die FBIDR Quicklooks an (95 MB auf 1:16 reduziert und 1:25 JPEG komprimiert) bzw. 140 kB für die FMIDR Quicklooks (56 MB auf 1:16 reduziert und 1:25 JPEG komprimiert).

#### 4.1.2 Allgemeine Auswahlkriterien

Bevor man sich für eine Netzwerktechnologie entscheidet, sollte man die folgenden fünf Fragen beantworten:

- Wozu braucht man das Netzwerk?
- Welche funktionellen Charakteristika soll das Netzwerk haben?
- Welche Leistungsanforderungen werden an das Netzwerk gestellt?
- Life cycle goals bestimmen
  - Zuverlässigkeit und Fehlertoleranz?
  - Einfaches Austauschen von Komponenten möglich?
  - Verfügbarkeit?
  - Wartbarkeit und Tuning?
  - Skalierbarkeit?
- Welche Akzeptanz Tests sollen durchgeführt werden?

Der erste Schritt ist ziemlich offensichtlich und wurde natürlich auch in dieser Arbeit mehr als ausführlich in den Kapiteln 1 und 3 behandelt.

Die funktionellen Charakteristika, also welche Daten sollen über das Netzwerk transportiert werden, wurden im Abschnitt 4.1 festgelegt.

Die Leistungsanforderungen des GDSS an das Netzwerk sind umfangreich und im Prinzip äußerst stark von der Anzahl der Gesamtbenutzer abhängig. Für einen einzelnen Benutzer ist es primär wichtig, daß die gerade benötigten Bildkacheln so rasch geliefert werden, daß ein sinnvolles interaktives Arbeiten möglich ist. Für jeden Schritt nach rechts oder links müßten also (wenn man von 4 x 3 Kacheln am Schirm ausgeht) 3 Kacheln á 3 kByte nachgeladen werden, für jeden Schritt nach oben oder unten 4 Stück. Für einen Zoomvorgang 12 Kacheln auf einmal. Dies entspräche also einer Spitzenrate von 288 kbit/sec (= 12 Kacheln \* 3 kByte \* 8 bit), wenn man von einer Darstellungsgeschwindigkeit von 1 Sekunde ausgeht. Dies erscheint für einen interaktiven Browsingvorgang sinnvoll. Weiters sollten natürlich auch die Quicklooks in annehmbarer Geschwindigkeit geladen werden. Dies würde bei einer zumutbaren Wartedauer von 5 Sekunden ~380 kbit/sec (237 kByte \* 8 / 5)

an Bandbreite benötigen, was ca. 100 kbit/sec mehr als der Spitzenbandbreite des Browsers entspricht.

Somit kann die benötigte Bandbreite, damit ein Benutzer tatsächlich bequem arbeiten kann, mit 288 kbit/sec als Spitzenwert festgelegt werden. Überlegt man, daß ATM über den OC-3 Standard ~140 Mbit/sec an nutzbarer Bandbreite anbietet, könnten theoretisch rund 486 Benutzer gleichzeitig im System arbeiten ohne sich nennenswert zu behindern. D.h. alle 486 Benutzer könnten z.B. gleichzeitig einen Zoomvorgang ausführen und kurzfristig 288 kbit/sec an Bandbreite benötigen und trotzdem würde die Antwortzeit des GDSS nicht wesentlich über einer Sekunde sein!

Die nächsten Punkte der „Life cycle goals“ werden mit Abschnitt 4.2 ausreichend beantwortet. Eine technische Einführung in ATM erfolgt im Anhang A zu finden. Akzeptanz Tests (im speziellen Performance Messungen) werden im Kapitel 6 beschrieben.

#### **4.1.3 Finanzielle Auswahlkriterien**

Nachdem ein Administrator alle allgemeine Fragen beantwortet hat, ist es Zeit, sich Gedanken um finanzielle Kriterien zu machen. Gerade die Kosten sind eines der wichtigsten Auswahlkriterien bei der Auswahl einer Netzwerktechnologie. Es stellt sich nämlich sehr bald unweigerlich die Frage: ist die Technologie das bezahlte Geld wert? Ein wenig präziser und formaler läßt sich das wie folgt definieren.

- Kosten versus Leistung
- Kosten von:
  - Service
  - Hinzufügen von weiteren Benutzern
  - Utilisierung erhöhen
  - Weitere Applikationen hinzufügen
  - Installation
  - Anbindung von weiteren Netzwerken (lokal, remote)

Die Antworten auf diese Fragen unterliegen natürlich einem geradezu erschreckend schnellem Wechsel. Immer mehr neue Firmen werden am Markt präsent, Preise ändern sich innerhalb weniger Wochen, wenn nicht sogar in wenigen Tagen. Aus genau diesem Grunde hätte es wenig Sinn, diese Fragen hier und jetzt zu beantworten. Ziel dieser Arbeit ist es, langfristig gültige Auswahlhilfen aufzuzeigen. Netzwerkadministratoren sollten für die finanziellen Kriterien ausreichend Zeit vorsehen, eine ausführliche Markterhebung erscheint angesichts der vielen

verschiedenen Angebote unumgänglich. Eine Untersuchung der aktuellen Trends und des ATM Marktes kann man im Anhang A finden.

#### 4.1.4 Technische Auswahlkriterien

Nachdem allgemeine und finanzielle Kriterien ausführlich behandelt wurden, wollen wir uns dem letzten Schritt der Netzwerkdefinitionsphase zuwenden, den technischen Auswahlkriterien. Sie werden deswegen zuletzt behandelt, da sie im allgemeinen am einfachsten vom Netzwerkadministrator - der ja i.a. ein Techniker ist - beantwortet werden können. Technische Auswahlkriterien sind:

- Flexibilität des Protokolls? Fähigkeiten?
- Charakteristiken des Datentransfers
  - Fehlerbehandlung/Korrektur
  - Adressierung
  - Prioritäts Schemata
  - Security Merkmale
- Unterstützung von Applikationen
  - Inwieweit werden existierende Standardapplikationen unterstützt?
  - Inwieweit werden existierende Protokolle /Standards unterstützt?
- Handelt es sich bei der (dem) ausgewählten Technologie/Protokoll um einen formalen Standard?

Besonders hervorzuheben ist der letzte Punkt, ob es sich bei der ausgewählten Technologie um einen formalen Standard handelt. Unter Umständen sind die Kosten für formale Standards höher als Lösungen von Billiganbietern, die sich nur unzureichend an formale Standards halten. Auf keinen Fall sollte aus Kostengründen auf einen formalen Standard verzichtet werden. Die Abhängigkeit von einem einzigen Hersteller (und das passiert, wenn man keinen formalen Standard erwirbt) kann einem auf lange Sicht gesehen noch um ein Vielfaches teurer kommen.

Für die in dieser Arbeit favorisierte ATM Technologie werden die meisten der obigen Fragen in den Kapiteln 4.2.6, 4.3 und 4.4 positiv beantwortet.

#### 4.1.5 Netzwerkmanagement

Bei der Verwaltung eines globalen verteilten Systems wie GDSS wirft sich natürlich die Frage nach einem professionellen Netzwerkmanagement auf, wie es bereits im 3. Kapitel angedeutet wurde. Dieser Abschnitt erläutert die dort angerissenen Ideen

ein wenig genauer.

Beginnend mit der Beschreibung von geeigneten Protokollen, über die Klärung der zu verwaltenden Objekte, der Beschreibung einer typischen Funktionalität von Netzwerk Management Systemen bis hin zur Vorstellung von existierenden Netzwerkmanagement Systemen werden wichtige Facetten eines modernen Netzwerkmanagements erläutert.

Die Aufgabe von Netzwerkmanagement besteht darin, einen ordnungsgemäßen Betrieb des Netzwerkes zu gewährleisten. Dies soll mit möglichst wenig zusätzlichen Ressourcen effizient durchführbar sein. Idealerweise kann ein einzelner Administrator von einer zentralen Stelle, dem sogenannten Netzwerk Management Zentrum (Network Management Center), sein gesamtes Netzwerk observieren, konfigurieren und auf Fehler reagieren. Das Netzwerk Management Zentrum gestattet auch verschiedene Sichtweisen auf das Netzwerk, diese lassen sich wie folgt definieren [Siegl, 94]:

- **Globale Sicht.** Übersicht über das gesamte zu betreuende Netzwerk samt der Möglichkeit, in lokale Netzwerke hineinzuzoomen.
- **Physikalische Sicht.** Hiermit bekommt man ein Bild aller physikalischen Verbindungen zwischen den Netzwerkkomponenten.
- **Segment Sicht.** Mit Hilfe dieser Sicht kann man spezielle Segmente, die mittels von Bridges, Portalen oder Gateways von anderen Segmenten getrennt sind auswählen und managen.
- **Erweiterte Ansicht.** Eine genaue Repräsentation der zu wartenden Netzwerkkomponente, samt allen LEDs, Anzeigen, Tasten.

Eine wichtige Facette eines integrierenden Netzwerkmanagement Systems ist das Fehlermanagement. Es definiert das Verhalten des Gesamtsystems in einem Fehlerfalle sowie dessen Behebung. Netzwerkmanagement ist jedoch nur am Rande Gegenstand dieses Projektes, eine Einführung ins Netzwerkmanagement samt Fehlermanagement und der Vorstellung der wichtigsten Management Protokolle ist im Anhang A zu finden.

#### 4.1.6 Internet und GDSS

Das Internet ist für das GDSS als globales Verbindungsglied eine wichtige Komponente. Der Begriff „Internet“ ist Gegenstand und Hauptthema einer Unzahl von Büchern [Hahn, 94]<sup>1</sup>, [Pabrai, 93]<sup>2</sup>, [Comer, 93], [Davidson,92], [Lane, 93],

---

<sup>1</sup> eine meiner Meinung nach sehr gute Gesamteinführung, auch für den Nichttechniker geeignet

<sup>2</sup> Für UNIX Interessiert ein Standardwerk

[Laquey, 93] und [Maier, 94] und kann im Rahmen dieser Arbeit keinesfalls in seiner Gesamtheit behandelt werden, zumal es auch nicht Kernthema der Dissertation ist. Eine prinzipielle Einführung ins Internet, die prinzipiellen Protokolle und die Vernetzung Österreichs und Europa sind im Anhang A zu finden. Schlußfolgerungen und die wichtigsten Konsequenzen sind im nächsten Unterkapitel zusammengefaßt.

#### **4.1.7 Schlußfolgerungen und Konsequenzen**

Es existiert bis jetzt kein Standard für das perfekte Netzwerkmanagement System. Das OSI-FCAPS Modell ist lediglich eine Empfehlung. Die optimale Funktionalität eines Netzwerkmanagement Systems hängt allerdings sehr von den örtlichen Gegebenheiten ab. Das bedeutet im Normalfall braucht man meistens nur eine Teilmenge der OSI Empfehlung, nicht aber die gesamte Palette der Funktionalität. Zur optimalen Konfiguration eines Netzwerkmanagement Systems sind die im Anhang A vorgestellten Auswahlsschritte eine große Hilfe und sollten die Basis eines jeden professionellen Systems bilden.

Von den kommerziellen Systemen bieten IBM Netview und HP-OpenView die größte Funktionalität, besonders hervorzuheben ist das Bausteinprinzip. Man kann das Grundgerüst jederzeit um weitere Funktionen erweitern, das Grundprogramm bietet hierzu ein Standardinterface. So können herstellerspezifische Komponenten, welche meist bereits entsprechende Managementprogramme mitliefern, problemlos in ein Gesamtmanagement System eingefügt werden.

SNMP und eventuelle auch CMIP Unterstützung ist ein wesentliches Auswahlkriterium für ein brauchbares Managementsystem, jede wartbare Komponente im System sollte über ein SNMP Interface verfügen. Dies ist ein weltweit anerkannter Standard und für das Design eines globalen Managementsystems unerlässlich.

Ein geeignetes Management System für GDSS müßte als Konsequenz deshalb folgende Anforderungen, die nach Prioritäten gereiht sind, erfüllen:

- SNMP und CMIP fähig, evtl. auch Unterstützung vom zukünftigen SNMPv2 Standard
- Fehler Management („Trouble Ticket“ Generierung, umfassende Fehlerdatenbank)
- Konfiguration Management zumindest im globalen Management
- Performance Management zumindest im globalen Bereich
- Datensicherheit Management, sehr wohl im lokalen als auch im globalen Bereich wichtig - besonders im Zusammenhang mit kostenpflichtigen Daten und Bezahlung derselbigen. Verschlüsselung von Bilddaten zumeist nicht nötig.

- Kostenmanagement (Accounting) ist für den globalen Bereich wünschenswert, kann aber auch lokal Sinn machen, z.B. wenn man ein größeres Subnetzwerk mit vielen Komponenten örtlich zu verwalten hat.

Beginnend mit der historischen Entwicklung des Internets, über die globale Vernetzung bis hin zu den derzeitigen Statistiken, die alle zusammen ein enormes Wachstum verdeutlichen bedarf es keiner besonderen Worte um die Bedeutung dieses Protokolls noch extra hervorzuheben. Vielleicht nur ein paar Worte zur Entwicklung in Österreich: im Jahre 1990 begann Österreich mit 192 registrierten Internet Rechnern, bereits am 30.4.1996 waren 65555 Rechner registriert. Dies zeigt, daß auch ein so kleines Land wie Österreich keinesfalls am internationalen Trend vorbeigeht, ja ganz im Gegenteil, Österreich spielt sogar eine entscheidende Rolle in der ATM Technologie. So wurden im Frühjahr 1996 die Technischen Universitäten in Wien und in Graz auf einen ATM Backbone umgestellt (OC-3, 155 Mbit/sec), ab Februar 1997 geht ein österreichweites OC-3 ATM Netz in Betrieb.

Die Wachstumszahlen der Statistiken (siehe Anhang A) geben eindeutig den Trend zur „Global Village“, also zu einer integrierenden, plattformübergreifenden Kommunikationslösung (ähnlich dem Telefon) wieder. Eine erste Konsequenz daraus ist, daß ein globales Informationssystem wie das GDSS unter allen Umständen auf diesem Protokoll aufbauen muß, will es eine fast weltweite Deckung erreichen.

Eine zweite Konsequenz - betrachtet man die nationalen internationalen Bemühungen ATM Pilotversuche zu unternehmen bzw. bereits ganze MANs auf ATM umzustellen - ist, daß die Backbone Technologie in naher Zukunft bereits ATM heißen wird. Das nächste Kapitel beschäftigt sich eingehender mit dieser Technologie.

## 4.2 Asynchronous Transfer Mode (ATM)

*Sobald man gelernt hat,  
Neues zu verstehen,  
gibt es keinen Grund mehr,  
sich davor zu fürchten*  
M. Curie

Im Zeitalter der Schlagwörter „Multimedia“, „Video on demand“ und „Videokonferenzen“ stellt sich natürlich rasch die Frage nach einer geeigneten Netzwerklösung, die hohen Anforderungen dieser Applikationen unterstützt. Gefragt ist hier nicht nur Bandbreite, sondern auch Synchronisation, Isochronie sowie eine geeignete Definition von Diensten. Zu guter Letzt muß man auch noch über eine geeignete Verrechnungsstrategie verfügen, denn spätestens wenn man dem Benutzer erlaubt, eine durchschnittliche Bandbreite in Kombination mit einer Spitzenbandbreite zu reservieren, kann man nicht mehr - wie bisher üblich - einfach die Spitzenbandbreite verrechnen. Man muß sich etwas Neues einfallen lassen.

Dieser Abschnitt beginnt mit einem kurzen technischen Überblick. Danach werden konkret am Beispiel der TU-Graz bzw. des ICG Möglichkeiten zum Umstieg auf ATM gezeigt. Schließlich wird der Aufbau und die Inbetriebnahme des TU OC-3 Clusters zwischen zwei Instituten der TU-Graz dargestellt. Den Abschluß bilden eine Zusammenfassung sowie resultierende Schlußfolgerungen.

#### 4.2.1 Technischer Überblick

ATM ist ein verbindungsorientierter Dienst, der im Prinzip mit dem des herkömmlichen Telefonnetzes verglichen werden kann. Der entscheidende Unterschied zwischen den beiden Diensten ist jedoch, daß es sich bei ATM dennoch um einen paketorientierten Dienst handelt, allerdings nicht um einen Datagramdienst, sondern um eine virtuelle Verbindung. D.h. alle Pakete nehmen denselben Weg, nachdem einmal zu Beginn die Route aufgebaut wurde. Allerdings wird nicht die gesamte Bandbreite verschwendet, sollte es eine Sendepause geben - wie dies etwa beim leitungsvermittelnden Telefonnetz der Fall ist - denn Dank des statistischen Multiplexing von ATM und der Paketorientiertheit wird die Leitung weitaus besser genutzt, als bei herkömmlichen verbindungsorientierten Diensten.

Durch das statistische, asynchrone Multiplexing erreicht man eine weitaus bessere Kanalauslastung. Gleichzeitig schafft man sich aber den Nachteil von etwas Overhead - denn gegenüber synchronen Zellen benötigt nun jede asynchrone Zelle einen Zellenkopf. ATM Pakete werden als „Zellen“ bezeichnet, da sie eine fixe Länge von 53 Bytes (48 Byte Daten, 5 Byte Kopf) besitzen [McDysan, 94]. Die offensichtlichen Vorteile, die sich daraus ergeben, sind: garantierbare Verzögerung von Paketen sowie ein schnelles, unkompliziertes Switching an den Knoten ohne große Pufferspeicher zur Verfügung stellen zu müssen.

ATM zeichnet sich durch hohe Übertragungsgeschwindigkeiten aus, gemäß der CCITT Empfehlung I.113 [CCITT, 91] „a service or system requiring transmission channels capable of supporting rates that are greater than the primary access rate“<sup>2</sup>. ATM geht über diese Definition sogar noch hinaus. Der ITU-T (ehemalige CCITT) Standard bezeichnet ATM als Transportservice für alle existierenden Applikationen sowie als Rahmendienst (Framework) für alle zukünftigen Dienste [Prycker, 94].

Seit Mitte der 80er Jahre wurde ATM schrittweise sowohl von den Standardorganisationen ITU-T [ITU-T, 96], ANSI [ANSI, 96] und ETSI als auch vom ATM Forum [Forum, 96] standardisiert. Das ATM Forum wurde im Oktober 1991 mit vier Mitgliedern gegründet, um die Kompatibilität zwischen öffentlichen

---

<sup>1</sup> so wie das ja auch beim Telefonieren geschieht

<sup>2</sup> Die Primärrate ist in Europa mit 2 Mbit/sec standardisiert. Dies sind 30 64 kbit/sec B Kanäle und zwei 64 kbit/sec D Kanäle. In den USA ist die Primärrate auf 1.5 Mbit/sec festgelegt. Dies sind 23 B Kanäle und ein D Kanal á 64 kbit/sec.

und privaten ATM Implementierungen zu gewährleisten und um die Standardisierung der ITU-T voranzutreiben. Das ATM Forum ist keine öffentliche Standardisierungsorganisation, es arbeitet jedoch eng mit der IETF (Internet Engineering Task Force) - dem Standardisierungskörper des Internet - sowie der ITU-T zusammen. Heute umfaßt das ATM Forum mehr als 700 Mitglieder, die aus öffentlichen und privaten Netzwerkanbietern, nationalen Forschungslabors, Regierungsabteilungen, Software Firmen, Service Providern und Universitäten bestehen. Im Jahre 1991 wurde ATM von der ITU-T als Standard für B-ISDN [CCITT, 90] definiert [Acampora, 94].

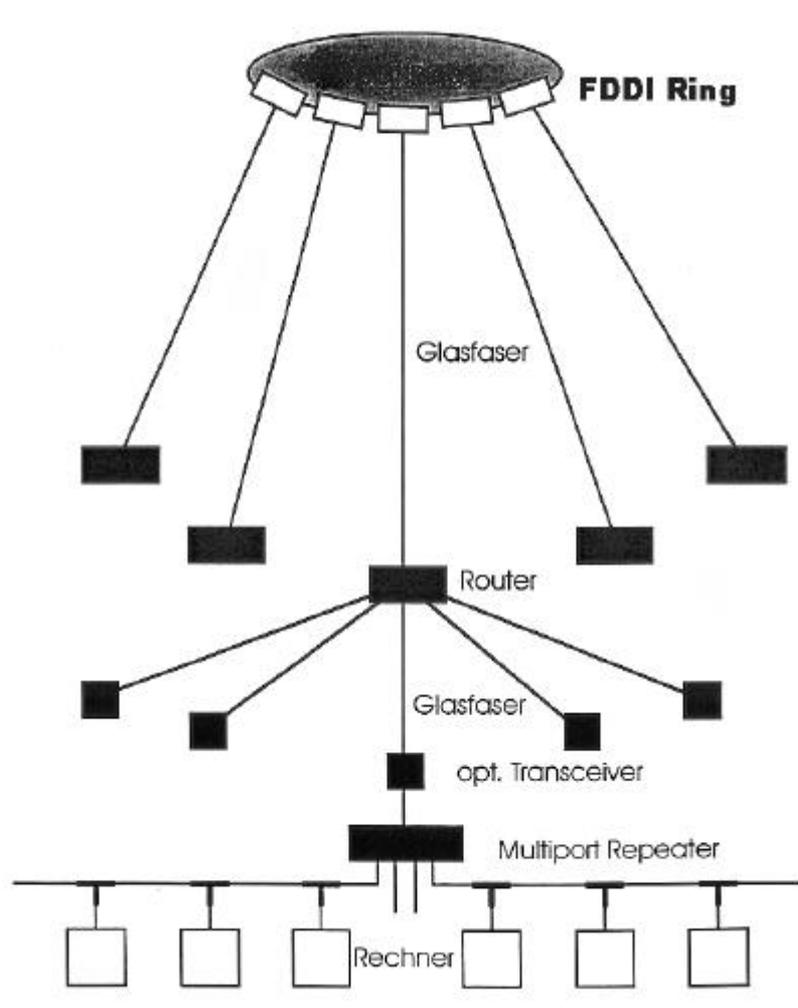
Eine detaillierte technische Einführung ist in Anhang A zu finden, wo genauer auf das ATM Referenz Protokoll Modell und die verfügbaren ATM Dienste eingegangen wird. Der nächste Abschnitt widmet sich dem Aufbau der ATM Infrastruktur an der TU-Graz und dem ICG. Diese Infrastruktur war die Basis für den GDSS Prototypen.

#### **4.2.2 Migration zu ATM am Bsp. der TUG**

In Graz hat der steigende Wunsch nach Kooperation zwischen Universitäten und außeruniversitären Forschungseinrichtungen sowie von der Industrie zur Errichtung eines Hochgeschwindigkeits FDDI Rings (100 Mbit/sec) geführt. Dies wurde 1988 initiiert und 1990 realisiert. Seit 1990 sind die Technische Universität Graz, JOANNEUM RESEARCH und die Anstalt für Verbrennungsmotoren (AVL) im Ring. Im Jahre 1991 wurde auch die Karl Franzens Universität aufgenommen.

Die physische Verkabelung ist sternförmig, die Anbindung der Teilnehmer (in der Regel die einzelnen Institute) erfolgt über Router. Die Gesamtstruktur kann man in Abbildung 4.3 erkennen. Man sieht deutlich die für FDDI typische Baumstruktur [Albert, 94].

Diese Struktur genügt auch den Forderungen eines hohen Maßes an Verkehrskontrolle (zentral möglich) und der Datensicherheit. Jedes Institut hat die Möglichkeit, quasi seine eigene Abschottung vorzunehmen, der einzige Weg nach draußen ist der Router. Im EDV Zentrum, welches sich natürlich in der Mitte des Ringes befindet, kann man sehr einfach eine Paketfilterung vornehmen, da der gesamte Verkehr dort zusammenläuft.



**Abbildung 4.3:** Struktur des FDDI Rings der TU-Graz, 1991. Die Anbindung der einzelnen Teilnehmer an den Primärring erfolgt über Router

Mitte 1995 wurde begonnen, den bestehenden FDDI Ring des MAN Graz schrittweise durch eine ATM Struktur zu ersetzen. Die sternförmige Struktur des ursprünglichen Ringes kommt dabei dem ATM Konzept, welches verbindungsorientiert mit Punkt-zu-Punkt-Verbindungen zwischen Switches arbeitet, sehr entgegen. Prinzipiell soll der bestehende Ring durch einen ATM Stern ersetzt werden, wie in Abbildung 4.4 dargestellt ist.

Die Umstellung erfolgt schrittweise in vier Phasen, wobei im November 1996 die Phase II abgeschlossen wurde. Die erste Phase umfaßt die Ausrüstung des zentralen Router der TU-Graz mit einem ATM Interface. Weiters erfolgt eine Verbindung des Routers mit dem WAN Switch des MAN Graz, welches von der Österreichischen Post und Telekom betrieben und verwaltet wird. Siehe hierzu Abbildung 4.5.

Hiermit ist es möglich, einzelne ATM Verbindungen vom Arbeitsplatz direkt über einen Glasfaserbatch an das ATM Interface des EDV-Zentrums heranzuführen. Von

dort geht es direkt zum MAN Graz und via dem ACONet ist eine nationale und internationale Anbindung möglich.

In Phase I wird der zentrale TU Router, der den Übergang ins ACONet darstellt, mit einem ATM Interface ausgestattet. Der ATM Anschluß erfolgt über eine direkte Glasfaserleitung zum jeweiligen Teilnehmer. Siehe hierzu Abbildung 4.5.

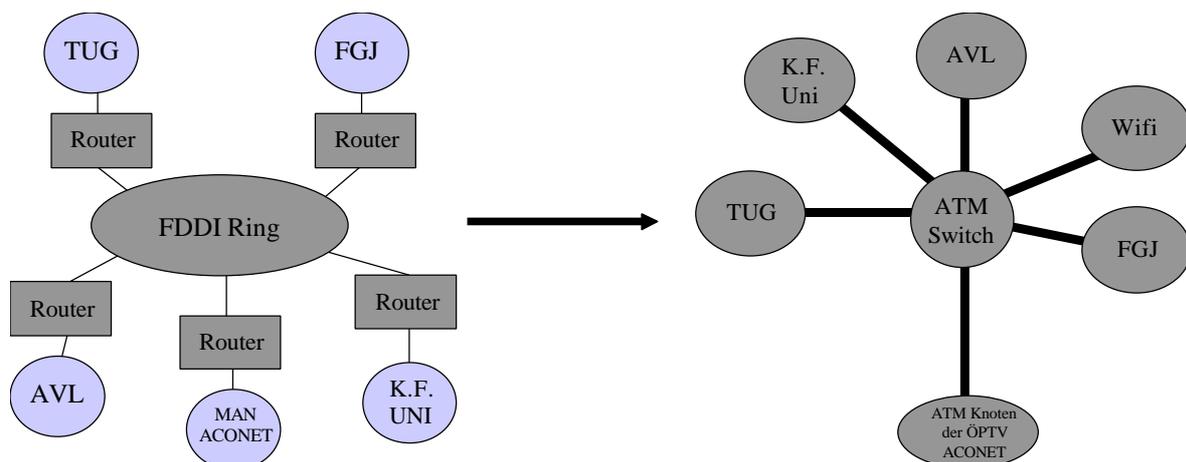
Die Phase II beinhaltet die Installation eines zentralen ATM Switches (ASX-200 von FORE Systems), die Verbindungen zu den einzelnen Institutsanbindungsroutern erfolgt mittels ATM Schnittstellen. Phase II ist in Abbildung 4.6 dargestellt.

Hierzu wurden sämtliche Router mit einem ATM Interface aufgerüstet. ATM Verbindungen können vom Arbeitsplatz über ein Glasfaserkabel an den Switch herangeführt werden. Dies ist seit November 1995 auch am TU Institut für maschinelles Sehen und Darstellen (ICG) in Verwendung, näheres dazu siehe Kapitel 4.2.3.

In Phase III wird in jedem Gebäudekomplex ein ATM Switch installiert. Diese werden mit dem zentralen Switch des EDV Zentrums verbunden. Es entsteht ein für ATM typisches sternförmiges Netz, wie in Abbildung 4.7 zu sehen ist.

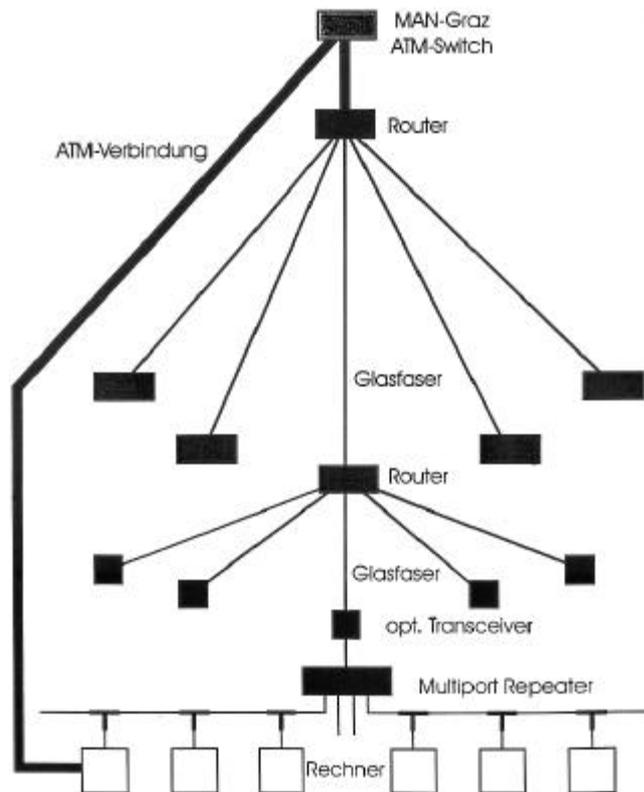
Nach Abschluß der Phase III besteht nun die Möglichkeit alle Institute in einem Gebäudekomplex mit ATM zu versorgen. ATM Verbindungen direkt zur Workstation/zum PC sind durch Direktanschluß an den Gebäudeswitch möglich. Eine umständliche Direktanbindung an den zentralen Switch des EDV Zentrums entfällt hiermit.

Die vierte und letzte Phase sieht eine Installation von ATM Switches in den



**Abbildung 4.4:** Migration von FDDI zu ATM am Beispiel der TU Graz. Die sternförmige Struktur des FDDI Ringes unterstützt den Umstieg auf ATM entscheidend. Die ATM Struktur weist einen zentralen Switch im EDV Zentrum auf, die Institute werden vorerst über Router mit ATM Interface angebunden.

einzelnen Instituten innerhalb eines Gebäudekomplexes vor. Diese Switches werden



**Abbildung 4.5:** Phase I der ATM Migration TU-Graz. Ausstattung des zentralen TU-Routers mit einem ATM Interface, einzelne Arbeitsplätze werden mit einem direkten Glasfaserbatch angebunden.

direkt über ATM an den Gebäude Switch angeschlossen, siehe hierzu Abbildung 4.8.

Jedes Institut verfügt nun über eine eigene ATM Verbindung und einen Switch, an den direkt ATM Arbeitsplätze angeschlossen werden können.

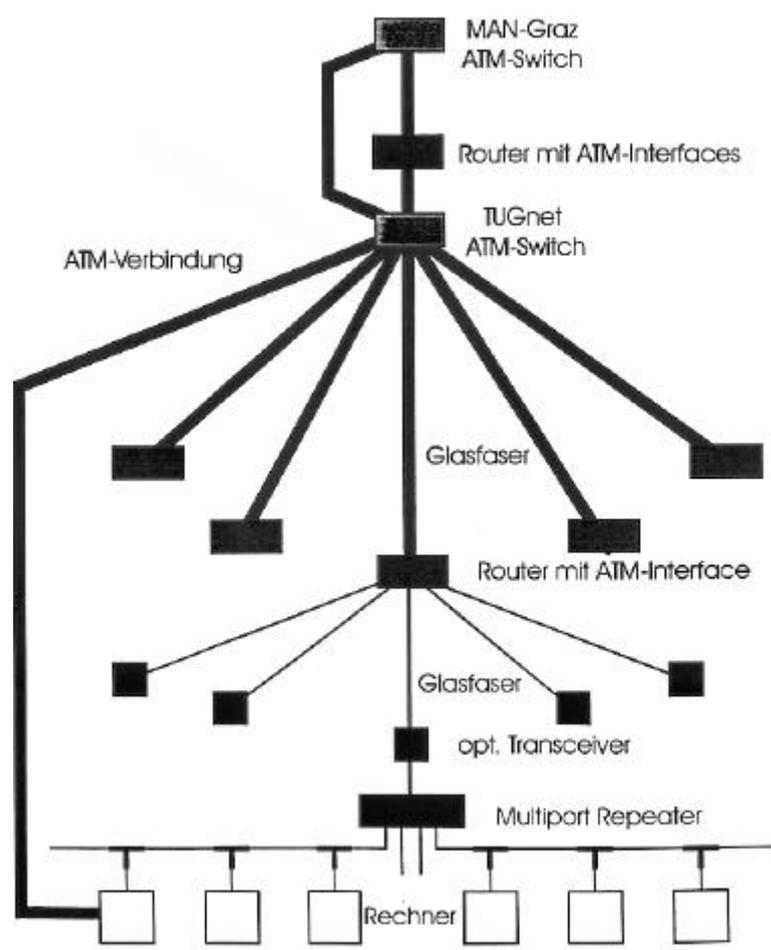
Im November 1996 war die Phase II abgeschlossen. Der Stand 1/1997 und die weiteren Ausbaupläne sind in Tabelle 4.4 zusammengefaßt.

Mit diesem 4-Phasen Modell gelingt es in überschaubarer Art und Weise den bestehenden FDDI Ring der TU Graz in ATM überzuführen. Beginnend mit einem Backbonenetz erfolgt die schrittweise Umstellung des TU Netzes bis hin zu ATM am Arbeitsplatz.

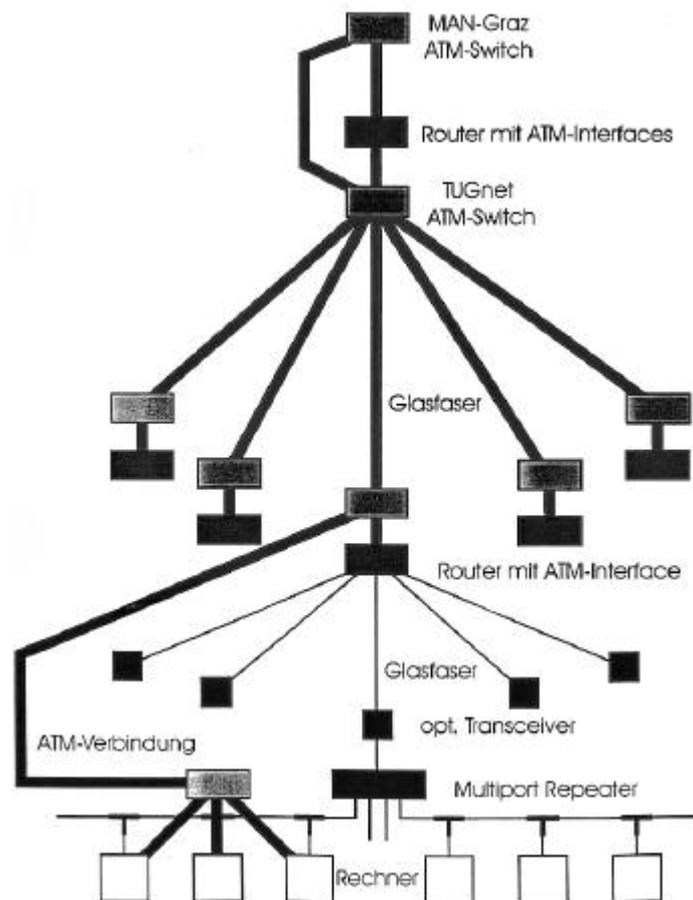
Die Phasen I-IV demonstrieren die gute Skalierbarkeit von ATM. Gerade für ein globales System wie GDSS ist ATM aus diesem Grund eine optimale Netzwerktechnologie. Bestehende Infrastrukturen müssen also nicht völlig erneuert werden sondern können schrittweise in eine ATM Hochgeschwindigkeits-Netzwerk Struktur gewandelt werden.

Teilnehmer	Stand Oktober 1996	Ausbauplan
TU Graz	ATM Phase II	ATM Phase III im Jahre 1997, ATM Phase IV nach Bedarf
FG Joanneum	FDDI	ATM ab 1. Quartal 1997
AVL	FDDI	derzeit kein weiterer Ausbau geplant
K.F. Uni	FDDI	ATM bis Ende 1996
Musikhochschule	ISDN Standleitung	ATM bis Ende 1996
Wifi Stmk.	ISDN Standleitung	ATM im Jahre 1997

**Tabelle 4.4:** Stand des TUG Netzes November 1996 und Ausbaupläne



**Abbildung 4.6:** Phase II der ATM Migration TU-Graz. Im EDV-Zentrum wird ein zentraler ATM Switch installiert. Die Router werden mit ATM Interfaces ausgestattet und mit dem Switch verbunden. Eine Anbindung von Arbeitsplätzen erfolgt direkt über ein Glasfaserkabel an den Switch.



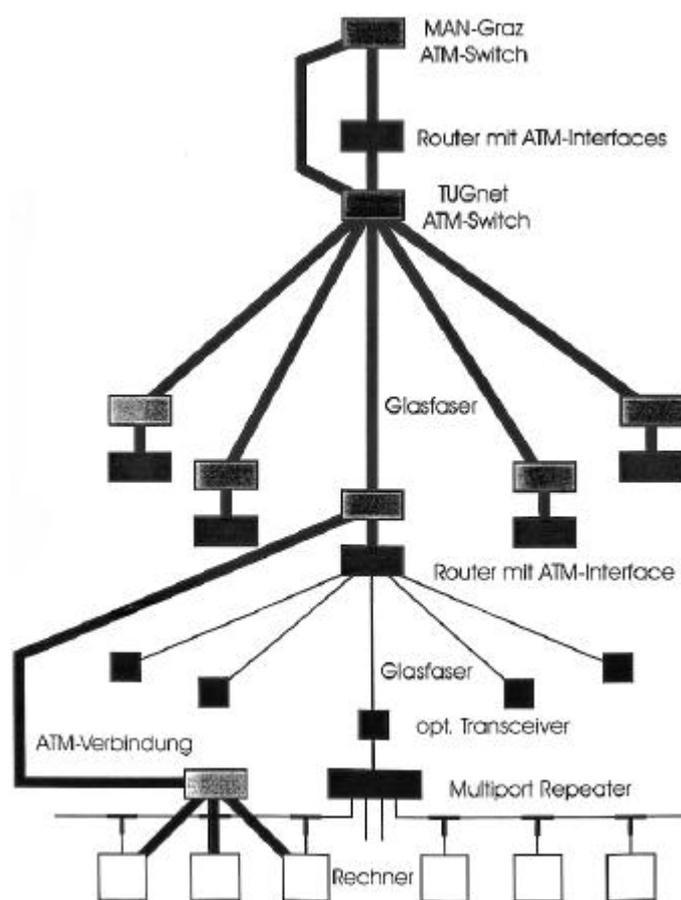
**Abbildung 4.7:** Phase III der ATM Migration TU-Graz. In den einzelnen Gebäudekomplexen werden ATM Switches installiert, welche mit dem zentralen Switch des EDV-Zentrums verbunden werden. Die Institute innerhalb des Gebäudes werden an den Switch angeschlossen.

#### 4.2.3 Aufbau des TUG OC-3 Pilotnetzwerkes

*„The trouble with doing something the first time is,  
that nobody appreciates how difficult it was.“*

Sandra Hartbach

Das erste Institut an der TUG, welches sich mit ATM beschäftigte, war das IAIK (Institut für Allgemeine Informations- und Kommunikationstechnologie). Im Jahre 1994 begannen die Forschungstätigkeiten mit der Installation eines FORE ASX-100 Switch, an dem lokal einige SUN Workstations angeschlossen wurden. Im Jahre 1995 wurde im gegenständlichen Projekt die Hochgeschwindigkeits-Netzwerk Forschung begonnen, in deren Verlaufe im Rahmen des EU-DIANE (DIstributed ANnotation Environment) [Diane, 96] Projektes die FORE VMA-200 ATM Adapter Karte für die SGI Power Challenge erworben wurde.



**Abbildung 4.8:** Phase IV der ATM Migration TU-Graz. Installation von ATM Switches in den einzelnen Instituten innerhalb eines Gebäudekomplexes. Diese Switches werden mit den Gebäude Switch verbunden. Eine ATM Verbindung direkt zum Arbeitsplatz ist somit über einen ATM Link zum Institutsswitch leicht möglich.

Zu Beginn waren die beiden Institute ICG und IAIK über den FDDI Ring der TUG mit 100 Mbit/sec verbunden, wobei der Zugang 10 Mbit/sec betrug (siehe auch Kapitel 4.2.2). Mit dem Erwerb der FORE VMA-200 Karte und der Installation eines ASX-200 ATM Switches im EDV-Zentrum war der Weg für ein erstes TUG OC/3 Versuchsnetzwerk geebnet. Der physikalische Einbau der FORE Systems Karte in die SGI Power Challenge, die damals unter IRIX 6.01<sup>1</sup>, verlief problemlos. Auch die benötigten Glasfaserkabel konnten rasch und ohne größere Verzögerungen bestellt werden. Die softwaremäßige Installation wurde weitestgehend vom Installskript von FORE Systems abgenommen. Dieses Programm setzt Root Rechte am Rechner voraus. Im wesentlichen müssen die folgenden Schritte durchgeführt werden:

- Auswahl des Treibers

---

<sup>1</sup> 64 Bit Unix Betriebssystem

- Installation des Kernels (die dev Einträge werden hier automatisch vorgenommen)
- Entscheidung ob der FORE SNMP Agent verwendet werden soll
- Entscheidung, ob man Classical IP verwenden will
  - Ja: Definition der NSAP (Network Service Access Point) Adresse des ARP (Address Resolution Protocol) Servers (siehe auch [McDysan, 94]) sowie Konfiguration der qaa (classical IP) Interfaces
  - Nein: für FORE IP erfolgt dies automatisch

Danach wird der Kernel installiert. Nach einem Booten der Maschine sind die konfigurierten Interfaces UP, sofern man noch einige Eintragungen - die im Handbuch genau erklärt werden - in die Initialisierungs Konfiguration aufnimmt.

Befehl	Erklärung
atmarp	gibt den aktuellen Status aller ein-/und ausgehenden Verbindungen an
atmstat	zeigt die aktuelle Statistik der Verbindungen an inklusive Fehler
asxmon	gibt Auskunft über den Switch Status und alle seine aktiven Verbindungen
cchan	damit lassen sich manuell virtuelle Kanäle managen
cpath	wie oben, nur für virtuelle Pfade
snmpd	SNMP Dämon für die VMA 200 Karte
looptest	testet eine Adapter Karte ob Zellen gesendet bzw. Empfangen können
topology	Zeigt die Topologie eines FORE Systems ATM Netzwerkes an

**Tabelle 4.5:** *FORE Systems Utility Programme.* Die Tabelle gibt eine Kurzübersicht über die von FORE Systems gelieferten Utility Programme und ihre Anwendung.

Die Konfiguration und Funktion kann mit den zahlreich mitgelieferten FORE Utility Programmen überprüft werden. Eine Übersicht ist in Tabelle 4.5 zusammengestellt.

Die Installation eines FORE ASX-200 ATM Switches (siehe Abbildung 4.9) gestaltet sich für einen erfahrenen Netzwerkadministrator fast ebenso einfach und schnell, vor allem, wenn man es mit dem Installationsaufwand eines Routers vergleicht. Im Prinzip müssen folgende Schritte durchgeführt werden:

- Einrichtung der IP Adressen im /etc/hosts file.
- Für SUNs muß man noch für jede angeschlossene Maschine entsprechend der konfigurierten qaa und die fa Interfaces ein „hostname.qaa0“ bzw. ein „hostname.fa0“ File anlegen, damit die Routen beim Booten erhalten bleiben.



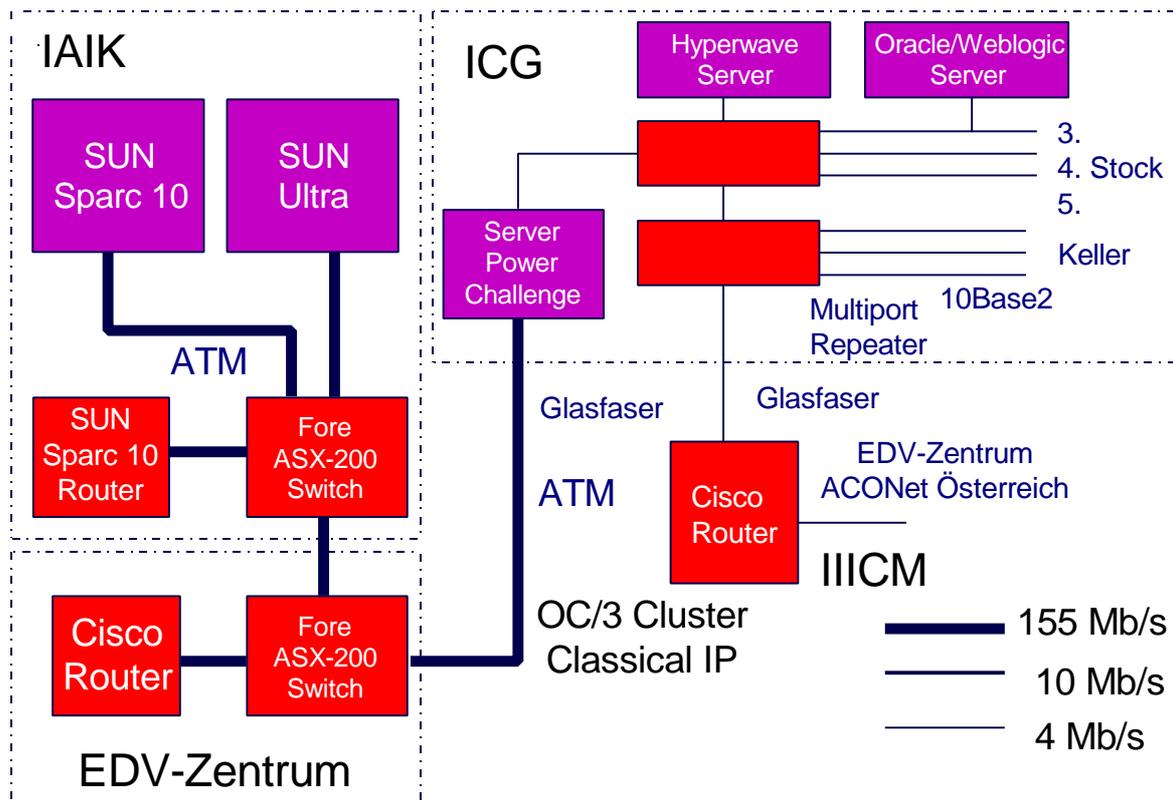
Abbildung 4.9: FORE ASX-1000 und 200 ATM Switch.

- Nach der Switch Initialisierung müssen die Routen an den lokalen Maschinen upgedated werden (route -add ...). Dies sollte bei SUNs erst erfolgen, nachdem das NIS läuft, da sonst versucht wird über die hostname.xxx files das NIS hochzufahren
- Einstellung des Signalisierungskanals (NSAP - ähnlich wie bei der Karte). Dies muß sowohl im Switch als auch an den lokalen Maschinen erfolgen, allerdings nur bei Classical IP, bei FORE IP geht alles automatisch

Danach verfügt man über SVCs, die bei Bedarf aufgebaut werden können. An Verkehrsparametern werden Peak Cell Rate, Sustained Cell Rate und Burst Tolerance unterstützt. PVCs müssen manuell kreiert werden.

Nach Abschluß der Installationsphase war die in Abbildung 4.10 dargestellte Konfiguration vorhanden. Dieses Netzwerk war Ausgangspunkt für zahlreiche Performance Messungen, die sowohl vom ICG als auch vom IAIK durchgeführt wurden (siehe auch Kapitel 7.2). Die schematische Abbildung zeigt das Institutsnetz des ICG sowie die Anbindung des IAIK über den ATM Switch des EDV Zentrums. Die Anbindung erfolgte über die FORE VMA 200 Karte über eine eigene Glasfaser ins EDV-Zentrum (siehe auch Kapitel 4.2.2, vergleiche Phase II).

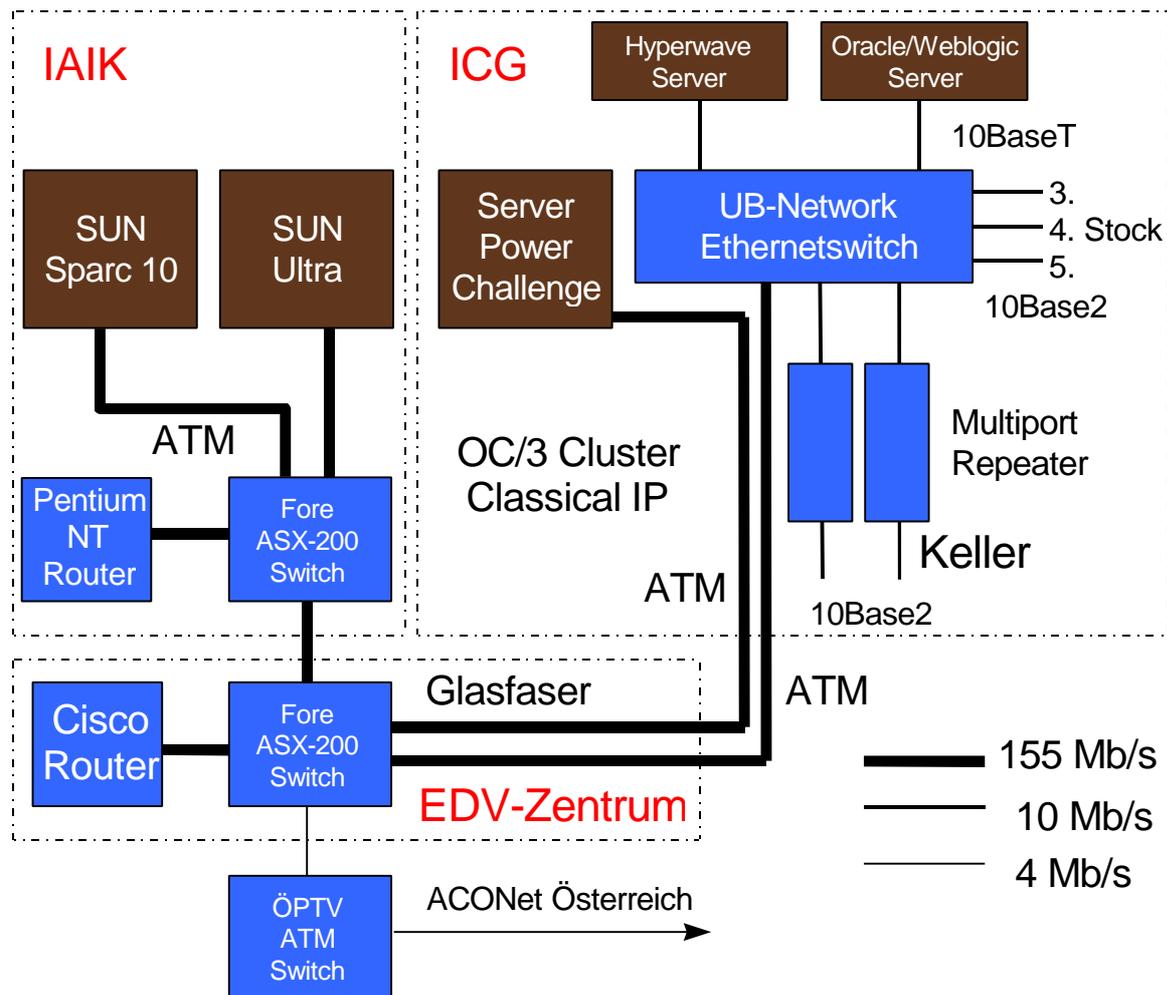
Das Netz des ICG basierte auf Ethernet, wobei alle drei Gebäudestockwerke des Institutes sowie das Erdgeschoß und die Übungsräume eine einzige Kollisions Domäne waren. Insgesamt waren über 30 SGI Unix Workstations, 4 Power Macs und zahlreiche PCs am Netz, die Leistungsgrenze des Ethernets war hiermit erreicht und es gab bereits immer häufigere Performance Engpässe. Die technische Frage war, wie kann man ohne großen Hardwareaufwand - der natürlich mit entsprechenden Kosten verbunden ist - den Gesamtdurchsatz verbessern?



**Abbildung 4.10:** TUG OC/3 Testcluster. Diese erste Konfiguration wurde im dritten Quartal 1995 (zuerst nur mit einem ATM Switch) in Betrieb genommen. Es bestand eine direkte OC/3 Verbindung zwischen den Instituten. Nur experimentelle Verbindungen wurden über die Switches aufgebaut, der restliche Datenverkehr ging noch über den Router im IICM an den FDDI Ring und von dort ins ACONet.

Die Entscheidung wurde unter Berücksichtigung der bereits bestehenden Hochgeschwindigkeitsverbindung zum EDV-Zentrum gefällt: eine Entflechtung des bestehenden Ethernets mit Hilfe eines Ethernet Switches. Die Entscheidung fiel hier auf den Switch von UB Networks (Modell Geo Rim/E) mit insgesamt 12 Ports und einem ATM Port, welches allerdings optional dazugekauft werden mußte. Die Anbindung des Power Challenge Servers erfolgte nun via OC/3 und ATM Interface an den Ethernetswitch. Die neue Netzwerkstruktur ist in Abbildung 4.11 dargestellt. Die 12 Ports sind eigene Kollisionsdomänen, die einzelnen Segmente des Ethernets sind also völlig entkoppelt. Der Switch leitet nur Pakete weiter, die für andere Segmente bestimmt sind.

Auf den ersten Blick überrascht diese Konfiguration, steht doch die Power Challenge direkt neben dem Ethernet Switch und ist trotzdem über eine ca. 1km langen „Glasfaserloop“ mit dem Switch verbunden. Das entscheidende Kriterium ist jedoch der enorme Geschwindigkeitsgewinn, der sich durch Verwendung von ATM ergibt.



**Abbildung 4.11:** Neue Netzwerkstruktur am ICG. Das bestehende Ethernet wurde mit Hilfe eines UB Network Ethernet Switches entflechtet. Der Power Challenge Server wurde über ein ATM Interface des Switches über das EDV Zentrum über LANE angeschlossen. Hiermit hat sich zwar die Distanz vergrößert, aufgrund der 155 Mbit/sec Bandbreite wurde es so allerdings ermöglicht, daß alle 12 Ports gleichzeitig auf den Server zugreifen können. Dies bewirkt eine gewaltige Performance Steigerung.

Es erfolgte eine Steigerung von 10 Mbit/sec auf 155 Mbit/sec. Somit können im Prinzip alle 12 Ports des Switches mit der vollen Bandbreite (10 Mbit/sec) mit dem Server kommunizieren. Gleichzeitig ist der Router des EDV-Zentrums bereits mit dem ACONet Router der Post und Telekom, Austria verbunden. Der gesamte externe Datenverkehr geht also ebenfalls über die schnelle OC-3 Verbindung.

Die Power Challenge ist mit dem Institutsnetz über der ATM LANE (LAN Emulation) verbunden. Dies bewirkt eine Transparenz des ATM Links - die Applikationen „sehen“ das Netzwerk bis zur Power Challenge als ein einziges Ethernet und müssen daher auch keine besonderen Maßnahmen treffen um sie zu erreichen. Dies wird von der LANE Software erledigt. Die LAN Emulation kann hier aus Platzgründen nicht im Detail beschrieben werden, nähere Details findet

man jedoch unter [Forum, 96], [McDysan, 94], [Siu, 95] und [Truong, 95].

Eine umfassende Übersicht an ATM Hardwareherstellern würde den Rahmen dieser Arbeit sprengen. Außerdem erscheinen soviele Hersteller nun am Markt, daß eine Liste sehr bald nicht mehr aktuell sein würde. An dieser Stelle sei auch auf folgende ATM WWW Seiten verwiesen, wo sich eine Seite mit Verweisen auf ATM Hersteller in der ganzen Welt befindet. Diese Seiten werden laufend auf den neuesten Stand gebracht. Man findet die Seiten unter:

<URL: <http://www.icg.tu-graz.ac.at/~herwig/Research/ATM/ATM.html>>

Am Beispiel der Migration des Institutsnetzes wurde erneut die gute Skalierbarkeit von ATM gezeigt. So würde etwa die Aufrüstung des ICG auf ein reines ATM Netz nun keine sehr großen Probleme mehr bereiten. In solch einem Fall wird der Ethernet Switch durch einen ATM Switch ersetzt. Die einfache Installierung und Inbetriebnahme der einzelnen ATM Komponenten tun ihr übriges, um einen insgesamt mehr als positiven Eindruck dieser neuen Technologie zu bekommen.

#### 4.2.4 Kostenverrechnung in ATM Netzwerken

Im Jahr 1996 ist ATM weltweit noch in einer Pilot Testphase. Dies ist auch der Grund, warum es zur Zeit noch kein zufriedenstellendes Kostenverrechnungsmodell gibt. So berechnete z.B. die Österreichische PTA 1995 die in Tabelle 4.6 zusammengefaßten Pauschal-Tarife pro Stunde, unabhängig von der tatsächlich beanspruchten Bandbreite.

Zone	2 Mbit/sec	10 Mbit/sec	34 Mbit/sec	155 Mbit/sec
Nahbereich	320	716	1320	2817
Fernbereich 1	640	1431	2639	5635
Fernbereich 2	960	2147	3959	8452
Auslandszone 1	1600	3578	6598	14087
Auslandszone 2	2560	5725	10556	22539

**Tabelle 4.6:** Tarife der Österreichischen Post und Telekom 1995. Die Angaben sind in Schillingen und beziehen sich jeweils auf eine Übertragungsdauer von einer Stunde

Es stellen sich hier sofort zwei wichtige Fragen: (1) Was passiert, wenn man weniger oder nur ein bißchen mehr als eine Stunde Daten schicken möchte? (2) Wie werden die Qualitätsparameter<sup>1</sup> von ATM berücksichtigt?

Die Antwort auf die erste Frage ist ebenso simpel wie ernüchternd: man muß

<sup>1</sup> wird im Englischen mit Quality of Service (QoS) bezeichnet - siehe dazu auch Anhang A

selbstverständlich jede Stunde voll bezahlen und jede angebrochene Stunde gilt als volle Stunde. Die zweite Frage kann man auch sehr einfach beantworten: außer der Spitzenzellenrate (Peak Cell Rate) wird kein weiterer Verkehrslastparameter berücksichtigt. Das bedeutet, selbst wenn man nur einmal einen kurzen Burst mit der spezifizierten Bitrate von z.B. 2 Mbit/sec hat und anschließend nur mehr mit z.B. 100 kBit/sec schickt, muß man die vollen 2 Mbit/sec bezahlen. Dies ist natürlich auf Dauer untragbar, da die Kunden eine solche Verrechnungsweise nicht akzeptieren würden.

Das einzige Stand 12/1996 bekannte Verrechnungsschema, welches kommerziell erhältlich und einsetzbar ist, wird von Alcatel [Alcatel, 96] angeboten. Die Switch Familie 1100 HSS verfügt über ein eigenes Network Management System (NMS), und kann sowohl Frame Relay als auch ATM Pakete weiterleiten. Die 1100 HSS Switches verfügen über die traditionellen Netzwerkmanagement Fähigkeiten (siehe auch Kapitel 4.1.5) Accounting, Performance, Sicherheit, Konfiguration und Fehlermanagement.

Die Kostenverrechnung ist im Accounting Modul realisiert und wird von einer relationalen Datenbank (Informix) unterstützt. Es wird sowohl eine Verrechnung nach Dauer als auch nach tatsächlich benutzter Bandbreite angeboten. Weiters kann man 6 Tarifklassen definieren und diese bis zu halbstündigen Zeitschlitzten gewissen Tageszeiten zuordnen. Die Zähler für Verbindungsdauer bzw. tatsächlich benötigter Bandbreite werden für jede dieser Tarifklassen getrennt geführt. Eine detaillierte Beschreibung des Modells sowie des gesamten Management Systems kann man bei [OMAS, 96] finden.

An dieser Stelle sei ein im gegeständlichen Projekt entwickeltes Kostenmanagement Modell vorgestellt. Es berücksichtigt alle bereits standardisierten Verkehrsparameter und zieht die vom Benutzer reservierte bzw. die dann tatsächlich benötigte Bandbreite in die Berechnung ein. Dies bewirkt eine faire Berechnung der anfallenden Kosten, da nicht nur eine statische Berechnung nach reservierter Bandbreite, sondern auch die dann tatsächlich benötigte Bandbreite in das Kostenmodell einbezogen wird. Außerdem ist das Modell offen für weitere neue Verkehrsparameter. Die Berechnung der Basiskosten  $K_B$ , das sind die vom Benutzer reservierten Ressourcen, erfolgt aus einer Summe der mit  $K_{i1}$  gewichteten Verkehrsparameter, wie aus Gleichung 4.1 ersichtlich.

$$K_B = K_{11} * PCR_1 + K_{21} * SCR_1 + K_{31} * MCR_1 + K_{41} * BT_1 + K_{51} * CDV_1 + K_{61} * CTD_1 + K_{71} * CLR_1$$

**Gleichung 4.1:** *Gewichtetes Kostenverrechnungsmodell (1).* Die Faktoren  $K_{i1}$  geben den jeweiligen Preis für die vom Benutzer reservierte Bandbreite an. Gewichtet werden die vom UNI standardisierten Verkehrsparameter.

Die Kosten der tatsächlich benutzten Bandbreite  $K_a$  errechnen sich ähnlich wie  $K_B$ ,

nur werden nun die neuen Kostenfaktoren  $K_2$  verwendet (siehe Gleichung 4.2), die Rücksicht auf die tatsächlich in Anspruch genommenen Ressourcen nehmen. Wobei aus den tatsächlich transportierten Daten und der reservierten Bandbreite das arithmetische Mittel berechnet wird.

$$K_U = K_{12} * PCR_2 + K_{22} * SCR_2 + K_{32} * MCR_2 + K_{42} * BT_2 + K_{52} * CDV_2 + K_{62} * CTD_2 + K_{72} * CLR_2$$

**Gleichung 4.2:** *Gewichtetes Kostenverrechnungsmodell (2).* Wie oben wichten die Faktoren  $K_2$  die UNI Verkehrsparameter. Diesmal handelt es sich jedoch um die tatsächlich in Anspruch genommenen Ressourcen.

Faire Gesamtkosten  $K$  ergeben sich aus dem arithmetischen Mittel der beiden Teilkosten  $K_B$  und  $K_U$ .

$$K = \frac{(K_B + K_U)}{2}$$

**Gleichung 4.3:** *Gewichtetes Kostenverrechnungsmodell (3).* Die fairen Gesamtkosten ergeben sich aus dem arithmetischen Mittel der reservierten und der tatsächlich in Anspruch genommenen Ressourcen.

Will man unterschiedliche Kostenbeiträge für tatsächlich verwendete und reservierte Ressourcen, so kann man einfach die beiden Teilkosten  $K_B$  und  $K_U$  nochmals wichten, indem man Gleichung 4.4 verwendet. Die Kostenbeiträge werden dann prozentmäßig gewichtet addiert, sofern man  $k_1 + k_2 = 1$  wählt.

$$K = \frac{(k_1 * K_B + k_2 * K_U)}{2}$$

mit:  $k_1 + k_2 = 1$

**Gleichung 4.4:** *Gewichtetes Kostenverrechnungsmodell (3).* Die Kostenbeiträge können mit den Faktoren  $k_1$  und  $k_2$  nochmals gewichtet werden. Hiermit kann man den tatsächlich benötigten Ressourcen und den reservierten Ressourcen unterschiedlich große Bedeutung zukommen lassen.

Für die Realisierung dieses Modells müßte man die 7 Verkehrsparameter (sofern verwendet) während der Übertragung messen sowie die ursprünglich definierten speichern. Weiters muß eine Wichtungsfaktorenmatrix  $K_j$  definiert werden. Die Verwaltung dieser Daten kann etwa mit Hilfe einer relationalen Datenbank geschehen, die pro Benutzer in einer Hand voll Tabellen diese Zahlen speichert. Eine Erweiterung dieses Modells wäre die Einführung einer weiteren Dimension in den Kostenfaktoren  $K_{ij}$  um z.B. entsprechend der Tageszeit einen entsprechenden Kostenfaktor zu erhalten.

Man sieht, das vorgestellte Modell ist äußerst flexibel in seiner Handhabung und trotz seiner Einfachheit sehr leistungsfähig. Es berücksichtigt weiters mehr als das von Alcatel verwendete Modell. Für GDSS wäre es eine optimale Strategie, da es

eine äußerst faire Verrechnung entsprechend dem vom Benutzer definierten Verkehrsparameter Vektor ermöglicht.

#### 4.2.5 Wirtschaftliche Aspekte

Wirtschaftliche Aspekte und Trendanalysen sind nicht Gegenstand dieses wissenschaftlichen Projektes. In der Praxis ist es jedoch nützlich, wenn man z.B. die Bedeutung des Produktes „ATM switch“ richtig einordnen kann. In diesem Projekt wurde darauf Rücksicht genommen und im Anhang A kann man jeweils ein Kapitel über den wirtschaftlichen Stellenwert von ATM bzw. Entwicklungen und Trends am ATM Sektor finden.

#### 4.2.6 Zusammenfassung und Bewertung

Dieses Unterkapitel durchleuchtete ATM in technischer und wirtschaftlicher Hinsicht. Zusätzlich wurde am Beispiel der TU-Graz die gute Skalierbarkeit von ATM und am Beispiel des ICG die praktische Migration zu ATM demonstriert. Folgende technische Vorteile ergeben sich für ATM:

- ✓ Hohe Bandbreite für WANs derzeit bis OC-12c (622 Mbit/sec)
- ✓ Service Qualität für jede Verbindung definierbar
- ✓ Standardisiert für eine große Anzahl von existierenden physischen Infrastrukturen (SDH, PDH, SONET, FDDI, DQDB, MM Fiber Interface)
- ✓ Einzige bekannte Technologie, die Daten, Sprache und Video global effizient transportieren kann
- ✓ LAN-Emulation ermöglicht eine ATM backbone Verbindung bestehender Ethernet/Token Ring LANs per bridges oder einen Übergang zwischen ATM und nicht-ATM LANs über eine bridge.
- ✓ Einfachere Installation von ATM Switches gegenüber Routern
- ✓ einfache Migration von bestehender Technologie (Ethernet, Token Ring, FDDI, ...)
- ✓ Hervorragende Skalierbarkeit

In wirtschaftlicher Hinsicht ist mit einem deutlichen Marktanstieg der ATM Technologie zu rechnen. Prognosen von IDC zufolge, wird sich alleine in West Europa der Anteil an ATM Technologie bis 1998 vervierfachen. In Europa wird es vor allem im LAN/MAN Sektor in den USA sowohl im LAN/MAN als auch im WAN Bereich große Zuwächse geben.

Insgesamt gesehen erscheint die im Jahre 1996 noch in der Pilotphase steckende ATM Technologie bereits sehr ausgereift. Mit ihren Adaption Schichten kann ATM, gemäß der Definition der ITU-T, alle bestehenden Dienste abdecken. Gleichzeitig

stellt es auch eine Rahmentechologie für zukünftige Dienste dar, die ohne größeren Aufwand in dieses generische Konzept eingefügt werden können. Aufgrund des ATM Forums wird die Standardisierung äußerst rasch und den Wünschen der Anwender entsprechend vorangetrieben, was eine internationale Kompatibilität gewährleistet. Betrachtet man die wirtschaftlichen Prognosen, so kann man ebenfalls einen mehr als positiven Eindruck gewinnen. Zusammenfassend erscheint also ATM die richtige Backbone Technologie für GDSS.

### 4.3 Alternativen zu ATM

ATM ist nicht die einzig verfügbare Technologie, die sowohl über eine hohe Bandbreite als auch über eine Anzahl von Diensten verfügt. Es wurden natürlich auch andere Technologien in Betracht gezogen und bewertet. Von einer eingehenden Erklärung, ähnlich wie sie bei ATM ausgeführt wurde, muß aus Platzgründen in dieser Arbeit Abstand genommen werden. Der interessierte Leser sei jedoch auf die angegebene Literatur verwiesen, die diese Technologien ausführlich behandeln. Untersucht wurden Integrated Services Digital Network (ISDN) [Kegel, 96] [Bauer, 93], Fiber Distributed Data Interface (FDDI I+II) [Sadiku, 95] [Albert, 94] [Mirchand, 93], Distributed Queued Dual Bus (DQDB) [Prycker, 94] [Albert, 94] [McDysan, 94], Frame Relay [McDysan, 94], Switched Multi-Megabit Service (SMDS) [McDysan, 94] [Sadiku, 95] und Fibre Channel [Fibre1, 96] [Fibre2, 96] [Fibre3, 94]. Eine zusammenfassende Bewertung sowie ein Vergleich mit ATM ist in Tabelle 4.7 zu finden.

Die Angaben der Geschwindigkeiten sind in Mega bzw. Giga Bit/Sekunde angegeben. Unter sehr guter Skalierbarkeit wird eine einfache Anpassung des Netzwerkes an erhöhte Anforderungen verstanden (HW oder SW). Schlecht wäre hier etwa, wenn man nur eine Transfergeschwindigkeit zur Auswahl hat (etwa bei FDDI und FDDI-II). Akzeptabel wäre, wenn man zwischen mehreren Alternativen wählen kann, „hervorragend“ wird eine benutzerseitige Dienste-Definition verstanden, wie es z.B. von Fibre Channel und vor allem von ATM unterstützt wird.

Weiters sieht man, welche Übertragungsarten von der jeweiligen Technologie unterstützt werden, sowie den prinzipiellen Verbindungsmodus (nicht die unterstützten!) und die unterstützten Verkehrsarten. Zuletzt sind noch die Protokolle aus den niedrigeren Schichten aufgeführt, über die man die jeweilige Technologie betreiben kann. Dies ist aus Kompatibilitätsgründen wichtig, man kann ja schließlich nicht für jeden Service eine neue Infrastruktur bauen.

ATM ist bis auf die maximale Transfergeschwindigkeit in allen Bereichen zumindest gleichwertig, wenn nicht besser. In der Transfergeschwindigkeit wird es zwar derzeit von Fibre Channel übertroffen, dieser Nachteil wird jedoch durch die globale Überdeckung wieder wettgemacht, die bei Fibre Channel lediglich LAN/MAN Charakter (10 km) besitzt.

	ISDN	FDDI	FDDI II	DQDB
Max. Geschwindigkeit	64 kb*	100 Mb	100 Mb	155 Mb
QoS Definition	—	—	—	—
Skalierbarkeit	☹	☹	☹	☺
unterst. Übertragungsarten	S / D	V / D	S / V / D	S / V / D
Verbindungsmodus	VO	VL	VL	VO / VL
Reichweite	global	100 km	100 km	150 km***
unterst. Verkehrsarten	S / I	A / S	A / S / I	A / S / I
Phys. Protokolle	SDH, PDH, SONET	-	-	SDH, PDH, SONET

(A)

	Frame Relay	SMDS	Fibre Channel	ATM
Max. Geschwindigkeit	1,544 Mb	45 Mb	1,062 Gb	622,08 Mb
QoS Definition	—	—	✓	✓
Skalierbarkeit	☹	☹	☺	☺
unterst. Übertragungsarten	V / D	S / V / D	V / D	S / V / D
Verbindungsmodus	VL	VL	VL / VO	VO
Reichweite	global**	150 km***	10 km	global
unterst. Verkehrsarten	A / S	A / S / I	A / S	A / S / I
Phys. Protokolle	SDH, PDH	SDH, PDH	SCSI, IPI, HIPPI, AAL-5, FC-LE, SBCCS, IEEE 802.2	SONET, SDH, PDH, FDDI, DQDB, MM Fiber Interface

(B)

Legende:

- ✓ ..... vorhanden      — .... nicht vorhanden  
☹ ..... schlecht      ☹ ..... akzeptabel      ☺ ..... hervorragend  
A ..... asynchron      I ..... isochron      S ..... synchron  
S ..... Sprache      V ..... Video      D ..... Daten

\* bei Zusammenschluß der beiden B-Kanäle bis 128 kb

\*\* ist aufgrund seiner Adressierung auf ca. 1000 Knoten begrenzt [McDysan, 94]. Unter Verwendung von Routern die mehr MAC Adressen verwalten können ist diese Beschränkung nicht gegeben (etwa CISCO Cat 5000, bis zu 31.000 MAC Adressen).

\*\*\* DQDB wurde für MAN's entworfen, die typischerweise ein Gebiet <150 km im Durchmesser decken. Prinzipiell ist aber auch mehr möglich - siehe z.B. MAN in Österreich

**Tabelle 4.7:** Vergleich von Alternativtechnologien mit ATM. Die Tabellen (A) und (B) geben übersichtlich Auskunft über die Hauptmerkmale der einzelnen Technologien. Für mehr Details siehe Text!

In bestimmten Fällen - etwa wenn man hohe Bandbreite gar nicht benötigt - kann eine Alternativtechnologie sicher Vorteile bringen, vor allem in preislicher Hinsicht. Dies liegt auch darin begründet, daß ATM eine noch relativ neue Technologie mit geringem Verbreitungsgrad ist. Daher werden die einzelnen Komponenten noch nicht in so großer Stückzahl produziert und weisen höhere Preise auf. Plant man jedoch langfristig, so ist ATM aufgrund seiner hervorragenden Skalierbarkeit und Leistungsfähigkeit in der Unterstützung aller Services bestimmt die erste Wahl.

Besondere Bestrebungen gibt es - auch von der Österreichischen PTA - hohe Bandbreiten über bestehende verdrehte Kupferleitungen zu übertragen. Derzeit bietet die österreichische Telekom die HDSL (High bitrate Digital Subscriber Line) an, die auf eine Entfernung von bis zu 6 km eine Bandbreite bis zu 2 Mbit/sec zuläßt. In Zukunft sollen auch ADSL (Asymmetric Digital Subscriber Line) und VDSL (Very high bitrate Asymmetric Digital Subscriber Line) verfügbar sein [Schenk, 96].

Mit A/VDSL hat man die zusätzliche Möglichkeit, Teilnehmer asymmetrisch anzuschließen. Die Verbindung zum Teilnehmer kann eine andere Bitrate aufweisen als die von ihm kommende. Eine Übersicht über dieser Technologien ist Tabelle 4.8 zu entnehmen.

Technologie	Reichweite	Bitrate
HDSL	bis zu 6 km	2 Mbit/sec
ADSL	2-5,4 km	1,5 - 8 Mbit/sec bzw. 16-640 kbit/sec*
VDSL	bis 1000 m	bis 55,2 Mbit/sec

\* Die Verbindung ist asymmetrisch voll duplex. Der erste Wert bezieht sich auf die Verbindung zum, der zweite auf die Verbindung vom Teilnehmer.

**Tabelle 4.8:** Verschiedene Technologien um über verdrehte Kupferdrahtleitungen Datenraten im Mbit Bereich übertragen zu können.

Da mit einer flächendeckenden „Glasfaser-zum-Haushalt“ Infrastruktur in naher Zukunft nicht zu rechnen ist, kann man diesen Technologien einen hohen Marktstellenwert zuweisen. HDSL wird von der Österreichischen Post und Telekom bereits seit Mitte 1996 angeboten, ADSL und VDSL werden je nach Verfügbarkeit der Standards bis ins Jahr 2000 folgen.

Die Kosten der PTA für die oben genannten Anschlüsse bzw. Dienste richten sich nach der örtlichen Lage der Anschlüsse, den beanspruchten Diensten, den gewünschten Bandbreiten, der Dauer der virtuellen Verbindung sowie den Vergebührungszonen (Inland oder Ausland). Die PTA erarbeitet in jedem Einzelfall entsprechend den Bedürfnissen des Teilnehmers die günstigste

Realisierungsvariante. Für Pilotprojekte werden Sonderkonditionen vereinbart. Nähere Informationen und kundenspezifische Beratung hierzu siehe [PTA, 97].

#### **4.4 Begründung der Wahl von ATM**

Nachdem in diesem Kapitel ausführlichst sowohl auf die technischen Vorteile als auch auf die wirtschaftlichen Aspekte und Trends der ATM Technologie eingegangen wurde, erfolgt hier nochmals eine Zusammenfassung der wichtigsten Gründe warum ATM gerade für GDSS wichtig ist.

- Möglichkeit der Definition von Verbindungs Qualitäten;
- hohe Bandbreite mit globaler Überdeckung;
- weitestgehend unabhängig von der physikalischen Infrastruktur - kann auf vorhandenen Transportdiensten aufsetzen;
- sehr gute Skalierbarkeit;
- internationaler Standard;
- zukünftige Trends und Entwicklungen der Wirtschaft geben eine optimistische Sichtweise, daß ATM keine Eintagsfliege ist;
- bietet die Kapazität neben der Datenübertragung auch multimediale Dienste anzubieten (Erweiterungen betreffend, z.B. Videokonferenzen, um Fernerkundungsdaten zu erklären etc.).

Diese Punkte erfüllen die im 3. Kapitel definierten Anforderungen an die Netzwerktechnologie.

## 4.5 Literatur und Referenzen

- [Acampora, 94] Acampora A. „An Introduction to Broadband Networks“. ISBN 0-306-44558-1, Pleum Press, New York 1994.
- [Albert, 94] Albert B., Jayasumana A. „FDDI and FDDI-II“, ISBN 0-89006-633-7, Artech House, 1994
- [Alcatel, 96] Alcatel Data Networks, November 1996, Homepage, <URL: <http://www.adn.alcatel.com>>
- [ANSI, 96] American National Standard Institution (ANSI), Homepage, <URL: <http://www.ansi.org>>
- [Bauer, 93] Bauer K., Geissler E., Haslinger H. „ISDN. Das Sprach-Daten-Text-Bild-Telefon“. Bohmann Verlag, 1993
- [CCITT, 90] CCITT Recommendation I.121, *Broadband Aspects of ISDN*, 1990
- [CCITT, 91] CCITT Recommendation I.113, *Vocabulary of Terms for Broadband Aspects of ISDN*, 1991
- [Comer, 93] Comer D., Stevens D. „Internetworking with TCP/IP“. Vol. 1-3, Englewood Cliffs, N.J, Prentice Hall, 1993
- [Davidson,92] Davidson R.; Muller N. „Internetworking LANs. Operation, design, and management“. Artech House, 1992
- [Diane, 96] DIstributed ANotation Environment, ACTS Eu-Projekt des Vienne Center for Parallel Computing (VCPC), November 1996, <URL: <http://www.vcpc.univie.ac.at/activities/projects/DIANE.html>>
- [Fibre1, 96] The Fibre Association, 1996. Fibre Channel: Connection to the Future. 12407 MoPAC Expressway North 100-357, P.O. Box 9700, Austin, TX 78758-9700, 1994. Voice: 1-800-272-4618 Fax: (408) 296-4457, <URL: <http://www.amdahl.com/ext/CARP/FCA/FCA.html>>
- [Fibre2, 96] Fibre Channel , 1996. CERN Fibre Channel homepage, <http://www.cern.ch/HSI/fcs/fcs.html>
- [Fibre3, 94] Fibre Channel Association, 1994. „Fibre Channel: Connection to the Future“, ISBN 1-878707- 19-1
- [Forum, 96] ATM Forum, Homepage, Dez. 1996, <URL: <http://www.atmforum.com>>
- [Hahn, 94] Hahn H., Stout R. „Internet - Complete Reference“. ISBN 0-07-881980-6, Osborne McGraw-Hill, 1994
- [ITU-T, 96] International Telecommunication Union, Homepage, <http://www.itu.ch>
- [Kegel, 96] Kegel. D., 1996. Dan Kegels ISDN Pages, <URL: <http://www.alumni.caltech.edu/~dank/isdn/>>
- [Lane, 93] Lane E., Summerhill C. „Internet Primer. For information professionals. A basic guide to internet networking technology“. Meckler Verlag, 1993
- [Laquey, 93] Laquey T.; Ryer J. „The Internet Companion. „A beginner's guide to global networking“. Addison-Wesley, 1993
- [Maier, 94] Maier G., Wildberger A. „In 8 Sekunden um die Welt, Kommunikation über das Internet“. Addison-Wesley, 1994
- [McDysan, 94] McDysan D., Spohn D., „ATM - Theory and Application“. ISBN: 0-07-060362-6, McGraw-Hill, 1994

- [Mirchand, 93] Mirchandani S., Khanna R. „FDDI: Technology and Applications“, John Wiley & Sons, 1993
- [OMAS, 96] Optional Management Application Software, 1996, Alcatel 1100 HSS Network Management System, Download über meine ATM Seite:  
<URL: <http://www.icg.tu-graz.ac.at/~herwig/Research/ATM/ATM.html>>
- [Pabrai, 93] Pabrai U „UNIX internetworking“, ISBN , Artech House, 1993
- [Prycker, 94] Prycker M. „Asynchronous Transfer Mode“. ISBN: 3-930436-03-5, Prentice Hall, 1994
- [PTA, 97] Post und Telekom, Austria - Business Center, Marburger Kai 43, 8010 Graz, Ing. Winkelbauer, Tel.: 0316 / 880 2023 oder: 0316 / 880 2020
- [Raffer, 96] Raffer P., Rehatschek H. "Die digitale Vernetzung Österreichs", Telematik Ingenieur Verband Magazin, Ausgabe 4/96, Dez. 1996
- [Sadiku, 95] Sadiku M. „Metropolitan Area Networks“. ISBN: 0-8493-2474-2, CRC-Press, 1995
- [Schenk, 96] Schenk M., Schmücking D., Wörner A., Ruge I. „VDSL (Very high bitrate Digital Subscriber Line) - die 3. Generation der digitalen Teilnehmeranschlußleitung“, Telekom Praxis 6/96, 1996, Seiten 10-19
- [Siegl, 94] Siegl M. „Was ist Netzwerkmanagement?“. EDV-Zentrum, Technische Universität Wien, White Paper, 1994
- [Siu, 95] Siu K., Jain R. „A Brief Overview of ATM: Protocol Layers, LAN Emulation, and Traffic Management“, unpublished, 1995. Download über meine ATM Seite:  
<URL: <http://www.icg.tu-graz.ac.at/~herwig/Research/ATM/ATM.html>>
- [Truong, 95] Truong H., Ellington W., Le Boudec J., Meier A., Pace J. „LAN Emulation on an ATM Network“. IEEE Communications Magazine, 33(5), pp. 70-85, May 1995
- [Walcher, 94] Walcher W. „Die Handhabung grosser digitaler Bilder am Beispiel eines Laborsystems zur Bildmessung“. Techn. Univ. Graz, Institut für Computerunterstützte Geometrie und Grafik, Diplom-Arbeit, 1994
- [Walcher, 95] Walcher W., Rehatschek H. „Design of a Distributed Database for the Magellan Data Set from Planet Venus“. IGARSS95, IEEE Proceedings catalogue number 95CH35770, Vol. II, pp. 610-1612, Florence 1995

## 5. Kapitel GDSS Implementierung

*"Ein Problem wird nicht im Computer gelöst,  
sondern in irgendeinem Kopf. Die ganze  
Apparatur dient nur dazu, diesen Kopf  
soweit zu drehen, daß er die Dinge richtig  
und vollständig sieht."*

(Charles Kettering)

Dieses Kapitel ist dem Entwurf und der Implementierung eines ersten GDSS Prototypen gewidmet. Dieser verwendet die wichtigsten der im 3. Kapitel vorgestellten Konzepte und die im Kapitel 4 beschriebene ATM Infrastruktur. Damit wird die Realisierbarkeit der wichtigsten geforderten Spezifikationen bewiesen. Diese sind nach Kapitel 3.5:

- Einhaltung eines international gültigen AbfrageStandards;
- plattformunabhängiger Klient mit GUI;
- leistungsfähiges Netzwerk (leicht skalierbar, Dienste Spezifikation);
- Lösung des Update Problems, falls neue Klienten ProgrammVersion kommt;
- Anbindung einer beliebigen relationalen Datenbank an den zentralen Server;
- um globale Vernetzung zu gewährleisten, muß das System auf TCP/IP aufsetzen;
- objektorientiertes Design;
- einheitliche Abfrage, unabhängig von der im zentralen Server verwendeten Datenbank.

Im speziellen wurden die Magellan FBIDR (Full Resolution Basic Image Data Record) Metadaten als Testdatensatz herangezogen. Der erste Abschnitt beschäftigt sich mit der Erstellung eines geeigneten Datenmodells. Hier wird grundsätzlich auf die Wahl des Datenmodells, auf die Erstellung eines Schemas, auf die Metadaten im Detail, auf Implementierungsschwierigkeiten sowie die praktische Realisierung des Modells mit dem Oracle Server eingegangen.

Der zweite Abschnitt durchleuchtet Java ein wenig näher, wobei Applets und Sicherheitsaspekte kritisch betrachtet werden.

Im dritten Abschnitt erfolgt die Beschreibung des Java JDBC (Java Data Base Connectivity) API, welches einen einheitlichen Zugriff auf ANSI SQL kompatible, relationale Datenbanken ermöglicht. Hier erfolgt auch eine Erläuterung des sich nun in vieler Munde befindlichen Begriffes der n-Tier Logik.

Der vierte und letzte Abschnitt befaßt sich mit dem Java Klienten. Struktur,

Bedienung und Funktionalität des Programmes werden erklärt.

## 5.1 Datenbankmodell

Der erste Schritt in der Implementierung war die Wahl eines geeigneten Datenmodells um in weiterer Folge den Magellan FBIDR Testdatensatz in geeigneter Weise mit der Datendefinitionssprache<sup>1</sup> des gewählten Modells zu beschreiben.

### 5.1.1 Wahl des Datenbankmodells

Grundsätzlich unterscheidet man zwischen den folgenden fünf Datenmodellen<sup>2</sup>

- Deduktives Datenmodell
- Netzwerk Datenmodell
- Hypermediales Datenmodell
- Objektorientiertes Datenmodell
- Relationales Datenmodell

Das **deduktive Modell** basiert im wesentlichen auf der Speicherung von Fakten und Regeln in der Datenbank. Unter Benutzung der Regeln kann man neue Fakten ableiten und diese zum vorhandenen Datenbestand hinzufügen [Das, 92]. Dieses Modell agiert in primitivster Form ähnlich dem menschlichen Denken und wird häufig in KI Anwendungen verwendet, die immer wieder auf Wissensdatenbanken basieren und Fakten/Regeln verwenden<sup>3</sup>.

Für GDSS ist dieses Datenmodell ungeeignet, da aufgrund der Datenbeschaffenheit weder Wissen gespeichert werden muß noch irgendwelche Schlußfolgerungen daraus abgeleitet werden sollen. Die Metadaten sind vielmehr statisch und erfahren nach der Erstellung in der Regel keinerlei Änderungen mehr.

Das **Netzwerk Datenmodell** welches auch unter dem weniger bekannten Namen CODASYL (Conference on Data Systems Language) bekannt ist, wurde 1971 von der DBTG (Data Base Task Group) des Programmier-Sprachen-Komitees definiert und standardisiert [Olle, 80]. Ähnlich zum relationalen Modell definiert es eine Schema Daten Definitions Sprache, eine Subschema Daten Definitions Sprache sowie eine Daten Manipulationsprache<sup>4</sup>.

Der verfügbare Standard ist allerdings zu alt und nicht an moderne Konzepte, wie z.B. an den objektorientierten Entwurf angepaßt. Das Modell weist starke Mängel

---

<sup>1</sup> wird im Englischen mit Data Definition Language (DDL) bezeichnet.

<sup>2</sup> Beim deduktiven Modell kann man noch ein Untermodell, das sogenannte funktionale Modell definieren

<sup>3</sup> ein gutes Beispiel hierfür ist jede Art von Diagnose

<sup>4</sup> wird im Englischen mit Data Manipulation Language (DML) bezeichnet.

bei der Datensuche und beim Löschen von Datensätzen auf. Die Suche muß manuell durch Navigation erfolgen oder entsprechend in der Klienten Software programmiert werden. Das Löschen weist zahlreiche Anomalien auf. Aus obigen genannten Gründen weisen tatsächlich realisierte DBMS des Modells keine hohe Marktpräsenz auf. Für GDSS ist das Netzwerk Datenmodell somit nicht geeignet.

Das **hypermediale Datenmodell** [Maurer1, 96] ist speziell dafür ausgelegt, binäre Informationen aller Art, also Daten, Bilder, Audio, Video, Postscript etc. in strukturierter und leicht wieder abrufbarer Form zu speichern. Prinzipiell besteht das Modell aus MM-Objekten, die jegliche Art von Information beinhalten können, sowie eine entsprechende Methode, die eine Visualisierung der Information am Bildschirm ermöglicht. So könnte etwa ein Video durch einen MPEG Player, ein Postscript File durch einen Postscript Viewer und ein Audio File durch ein Audio-Panel am Bildschirm visualisiert und dem Benutzer zugänglich gemacht werden. Neben den MM-Objekten werden in einem Hypermedialen DBMS noch Links und Anker gespeichert. Links haben nur innerhalb des DBMS Bedeutung. Sie legen die logische, vom Benutzer bestimmte Struktur der Daten fest. Anker sind die Visualisierungen derselben - also etwa eine speziell am Bildschirm gekennzeichnete Stelle am Schirm auf die man mit der Maus klicken kann.

Für das hypermediale Modell gibt es keinen internationalen Standard. Das bedeutet, daß jeder seine Daten in mehr oder weniger intuitiver Weise, so wie es für die aktuelle Anwendung am besten erscheint, speichert. Daraus resultiert z.B. die hohe Vielfalt an Erscheinungsbildern von WWW Servern im Internet. Man kann tagelang surfen und wird vermutlich nie auf ein und denselben Aufbau eines Informationsangebotes stoßen. Gerade dies sollte aber im GDSS unter allen Umständen verhindert werden. Der Benutzer soll nur ein einziges Mal und unabhängig vom Anbieter lernen, an die von ihm gewünschten Daten zu kommen. Dementsprechend gibt es auch sehr viele unterschiedliche Implementierungen und Interpretationen des hypermedialen Modells, die alle zusammen unterschiedliche Funktionalität aufweisen [Maurer1, 96], [Maurer2, 96], [Netscape, 96]. Schon aus diesen Gründen kann dieses Modell für GDSS nicht geeignet sein, weiters würde eine räumliche Suche in keiner Weise vom DBMS unterstützt werden.

Das **Objektorientierte (OO) Datenmodell** eignet sich zur Modellierung komplexer Datenstrukturen, was sich mit konventionellen Datenmodellen nur sehr schwer realisieren läßt [Kim, 90], [Delobel, 95]. Speziell in bezug auf das GDSS würde die räumliche Suche durch die Implementierung einer R-Tree Datenstruktur entscheidend beschleunigt.

Erste Versuche wurden bereits im Parallelprojekt durchgeführt [Walcher, 96]. Allerdings wurde hierbei die Datenstruktur nicht innerhalb des DBMS, sondern innerhalb des in Java programmierten Klienten Programmes implementiert. Die Daten selbst waren in einer konventionellen relationalen Datenbank gespeichert.

Dies hatte deutliche Performance Einbußen zur Folge.

Der GDSS Prototyp (siehe Kapitel 5.4), welcher in einer objektorientierten Sprache geschrieben wurde, würde natürlich ebenfalls von der Verwendung eines OO DBMS profitieren. So könnten Ergebnisse einer Datenbankanfrage direkt als Objekte übernommen werden und müßten nicht erst in Objekte umgewandelt werden, wie es z.B. bei einer relationalen Datenbank der Fall wäre. Dies bringt einen Performance Gewinn und verringert den Programmieraufwand.

Aus der bisherigen Beschreibung wäre das OO Datenmodell die erwünschte Lösung für das GDSS, derzeit gibt es jedoch zwei entscheidende Nachteile, die letztendlich darüber entschieden haben, dieses Modell nicht zu verwenden. Erstens gab es bis zum Herbst 1996 noch keinen internationalen Standard für die Abfrage bzw. den Erhalt von Daten und zweitens ist die Marktpräsenz von OO DBMS derzeit noch gering, was hohe Preise der entsprechenden Produkte gegenüber konventionellen relationalen Systemen bewirkt<sup>1</sup>.

Bestrebungen sind aber im Gange, um den ANSI SQL-92 Standard in einen SQL3 Standard überzuführen, welcher die Definition und das Management von persistenten Objekten erlauben soll. Konkret sind folgende Punkte geplant: Generalisierung und Spezialisierung von Hierarchien, mehrfach Vererbung, Benutzer definierte Datentypen, Unterstützung für wissenbasierenden Systemen, Einbindung von, rekursive Abfrage Optionen sowie zusätzliche Daten Administrations-Werkzeuge. Weiters sollen spezielle OO Merkmale wie die Spezifizierung von abstrakten Datentypen, Objekt Identifizierung, Methoden, Vererbung, Polymorphismus, Kapselung und alle weiteren mit OODBMS verknüpften Merkmale enthalten sein. Der Standard wird derzeit sowohl von der ANSI als auch von der ISO bearbeitet, der Standard soll 1999 veröffentlicht werden [SQL3, 96]. Wegen dieser Mängel wurde das OO Datenmodell für den ersten GDSS Prototypen nicht verwendet. Eine Migration bei Verfügbarkeit des Standards ist jedoch unumgänglich, da das OO Modell die Anforderungen des GDSS optimal erfüllen könnte.

Das **relationale Modell** [Codd, 70], [Mannila, 92], welches eine Teilmenge des Netzwerk Modells darstellt, ist das derzeit populärste und meistverwendete Modell auf der Welt. RDBMS (Relational Data Base Management System) sind schon seit den späten 70er Jahren in Verwendung und wurden ständig weiterentwickelt und in ihrer Performance verbessert. Aus dieser Erfahrung resultieren stabile Serversoftware und mächtige Entwicklungsumgebungen bzw. Design Werkzeuge (z.B. Oracle Developer und Designer 2000), welche in einfacher Weise den Entwurf und die Programmierung von Applikationen ermöglichen.

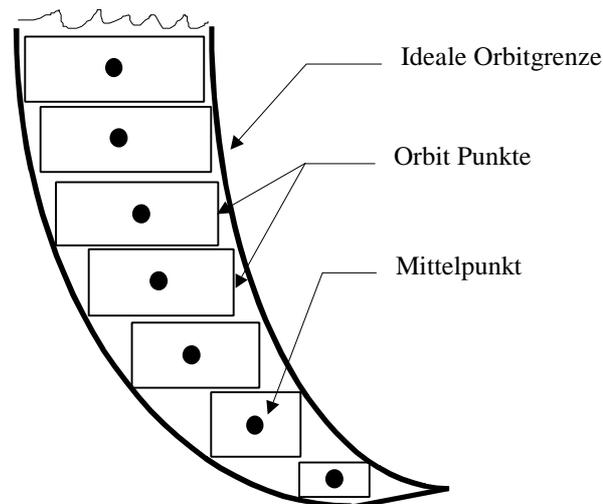
---

<sup>1</sup> so würde etwa Object Store S 17.500 pro Universitätslizenz (Stand Dez. 1996) kosten

Die Anwendung des relationalen Modells weist aber speziell im Gebiet der räumlichen Datenverwaltung einige grobe Mängel auf. So kann z.B. die für räumliche Suche benötigte R-Tree Datenstruktur nur äußerst umständlich und mit großen Performance Einbußen in das relationale Modell umgesetzt werden. Einige Hersteller bieten zwar sogenannte „Spatial Extensions“ an (z.B. Oracle, Sybase, Microsoft), jedoch sind diese Pakete nicht für einen globalen Datensatz geeignet. Sie haben bei Abfragen, die über den Null Meridian gehen, große Probleme. Entweder werden unvollständige oder gar keine Ergebnisse zurückgeliefert. Dieses Problem muß man durch eigenständige Programmierung umgehen und lösen [Walcher, 96]. Mit dem Aufkommen von OO Methoden wurde den Datenbank Entwicklern ein mächtigeres Werkzeug in die Hände gelegt, welches in viel spezifischerer Weise die Modellierung von solch Applikations-spezifischen Problemen gestattet.

Namhafte Datenbankanbieter wie z.B. Oracle, Informix/Illustra oder Sybase bieten auch OO Erweiterungen ihrer relationalen DBMS an. Diese sogenannten ORDBMS (Object Relational Data Base Management System) leiden aber unter großen Performance Problemen, da das System im Grunde auf einem relationalen Modell basiert und Abfragen/Einspielungen immer konvertiert werden müssen. Eine Optimierung auf OO Funktionen kann ebenfalls nicht erfolgen, da relationale Operatoren weiter verwendbar sind. Ein Vergleich aus der Programmierwelt wäre hier etwa C++ und Java. C++ ist in den meisten Implementierungen als OO Aufsatz realisiert, d.h. man kann auch normales C programmieren und den OO Ansatz umgehen. Java ist eine rein OO Sprache, der Programmierer muß sich dem OO Konzept anpassen.

Die Hauptvorteile des relationalen Modells sind seine Einfachheit, seine hohe Marktpräsenz und die Verfügbarkeit eines internationalen Standards (ANSI SQL-92 X3.135-1992), der von namhaften Herstellern wie Oracle, Informix, Sybase etc. eingehalten wird. Dies waren die Hauptgründe warum dieses Datenmodell für eine erste Implementierung eines GDSS Prototypen verwendet wurde. Es erfüllt eine der Hauptforderungen des GDSS - die Verwendung von internationalen Standards. Somit wird es den Benutzern des GDSS ermöglicht, Archive auf der ganzen Welt mit ein und demselben Benutzerinterface und derselben Abfragesprache abzusuchen.



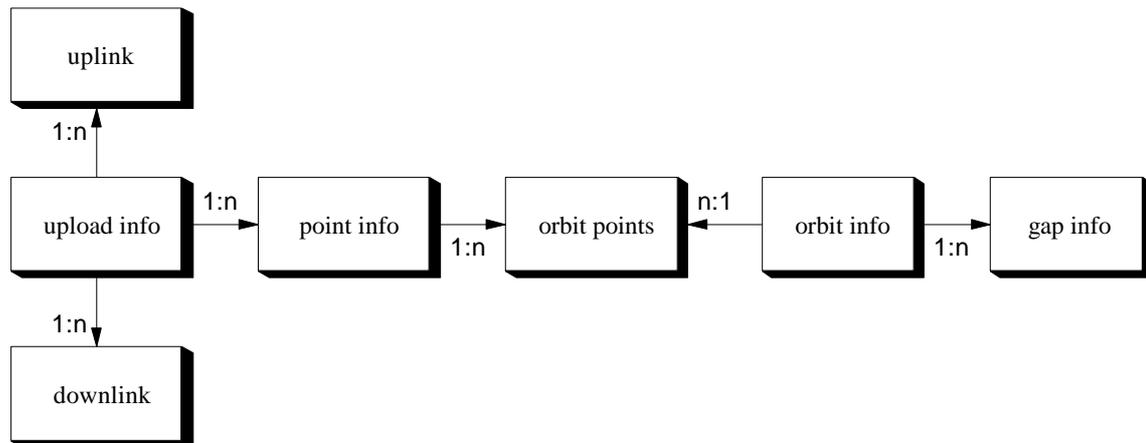
**Abbildung 5.1:** Aufzeichnung der Orbits (FBIDR). Die Abbildung zeigt einen Ausschnitt eines Orbits samt den Orbit Punkten, aus denen der Orbit eigentlich besteht. Daraus folgt auch die leichte Ausfranzung der Orbitgrenzen.

### 5.1.2 Details zur Magellan FBIDR Aufzeichnung

Im Sommer 1996 erhielt das ICG den Magellan FBIDR (Full Resolution Basic Image Data Record) Datensatz von der Washington University, St. Louis/Missouri, welche einen „Geosciences-Subnode“ des NASA PDS betreibt. Die Aufzeichnung der FBIDRS (auch als Orbits bezeichnet), welche gleichzeitig die Bild-Rohdaten darstellen, erfolgte in 3 Zyklen mit jeweils vom Südpol zum Nordpol reichenden Streifen<sup>1</sup>. Beim 1. Und 3. Zyklus wurde der Sensor auf die linken Seite aber unter verschiedenem Winkel ausgerichtet, im 2. auf die rechte Seite. Das daraus resultierende Bildmaterial ist mit Streifen („noodles“) vergleichbar, die eine Länge von rund 220.000 Pixel und eine Breite von nur ca. 350 Pixel aufweisen. Dies entspricht bei einer Auflösung des Radar Sensors von 75m/Pixel ungefähr 16.500 km x 26,25 km. Die Orbits des ersten und des dritten Zyklus kann man für Stereoprozessierung verwenden, da sie von der selben Seite aber unter verschiedenem Winkel aufgenommen wurden<sup>2</sup>.

<sup>1</sup> Daher die Bezeichnung Orbits

<sup>2</sup> Da es sich um Radarbilder handelt, kann man zur Stereoauswertung nur Bilder verwenden, die von derselben Seite aufgenommen wurden - Bilder von der anderen Seite sehen völlig anders aus und sind praktisch kaum paßbar. Bei optischen Bildern ist dieses Problem nicht gegeben.



**Abbildung 5.2:** Datenbankschema der Magellan FBIDR (Orbit) Metadaten. Die Metadaten beinhalten alle Informationen vom Senden des Kommandos bis zur Auswertung, die einzelnen Punkte der Orbits, sowie die Information über Störungen und Ausfälle des Satelliten.

Jeder Orbit wurde aus einer Anzahl kleiner Rechtecke, sogenannten Orbit Punkten, zusammengesetzt, wie in Abbildung 5.1 illustriert. Da alle Rechtecke in etwa dieselbe Breite haben, wurde nur der Mittelpunkt derselben gespeichert. Dies ist in Abbildung 5.1 durch die dicken Punkte innerhalb der Rechtecke angedeutet.

Jeder Orbit setzt sich aus ca. 700 Punkten zusammen, deren Positionen und Daten in den Magellan Metadaten Tabellen gespeichert sind. Eine nähere Erklärung der Metadaten erfolgt im nächsten Abschnitt.

### 5.1.3 Datenbankschema für die Magellan FBIDR Metadaten

Die Magellan Orbit Daten weisen ein Gesamtvolumen von 110,897 Mbyte auf und wurden im ASCII Format geliefert, wobei die Zeilen durch „Carriage Return“ und die Einträge innerhalb einer Zeile durch ein „|“ Zeichen getrennt waren.

Entsprechend des in Kapitel 5.1.1 gewählten Datenmodells wurde zu der Datenstruktur ein Datenbankschema mit Hilfe eines Entity-Relationship Diagrammes erstellt. Das Datenbankschema ist in Abbildung 5.2 zu sehen.

Die **Uplink** Tabelle enthält zu jedem gesendeten Kommando (einem sogenannten „upload“) Kommentare. Die 1:n Relation bedeutet: zu einem Kommando können n Zeilen Kommentare vorhanden sein. Somit setzt sich der Primärschlüssel für die Uplink Tabelle aus der „upload number“ und der „text line number“ zusammen. Die Attribute und die Datentypen der Tabelle sind Abbildung 5.3 zu entnehmen, Primärschlüsselattribute sind kursiv geschrieben.

<b>Uplink</b>	
<i>upload number</i>	char(10)
<i>text line number</i>	integer
one line of text	char(60)

**Abbildung 5.3:** Struktur der Uplink Tabelle. Kursiv sind die Primärschlüssel dargestellt

<b>Downlink</b>	
<i>upload number</i>	char(10)
<i>text line number</i>	integer
one line of text	char(60)

**Abbildung 5.4:** Struktur der Downlink Tabelle. Kursiv sind die Primärschlüssel dargestellt

Die **Downlink** Tabelle enthält ähnlich der Uplink Tabelle Kommentare zu den einzelnen Kommandos, allerdings nach deren Ausführung. Hier findet man Bemerkungen zu Erfolg/Mißerfolg oder andere allgemeine Informationen zu den Kommandos. Datentypen und Attribute sind Abbildung 5.4 zu entnehmen. Die 1:n Relation ist wie oben zu interpretieren.

<b>Upload Info</b>	
<i>upload number</i>	<i>char(10)</i>
first orbit number	integer
last orbit number	integer
number of command sequences (points)	Integer

**Abbildung 5.5:** Struktur der Upload Info Tabelle. Kursiv ist der Primärschlüssel dargestellt.

In der **Upload** Tabelle ist gespeichert, welche Orbits durch das Kommando aufgezeichnet wurden. Das bedeutet, mit jedem Kommando wurde eine ganze Gruppe von Orbits aufgezeichnet. Weiters ist hier die Anzahl der Punkte aus der sich die einzelnen Orbits zusammensetzen gespeichert (siehe auch Kapitel 5.1.2). Die Struktur der Tabelle ist Abbildung 5.5 zu entnehmen. Die 1:n Relation zur „Point Info“ Tabelle besagt, daß zu einem Kommando n Orbits gehören (ein Kommando galt für eine Orbit Gruppe).

<b>Point info</b>	
<i>upload number</i>	char(10)
<i>command sequence number (point number)</i>	integer
planetary radius	real
pulse repetition frequency	real
SAR antenna boresight incidence angle	real
SAR antenna boresight latitude	real
SAR azimuth resolution	real
SAR effective looks	real
SAR ground distance (range) resolution	real
SAR ground swath width	real
SAR look angle	real
SAR predict signal to ambiguity ratio	real
SAR predict signal to thermal noise ratio	real
spacecraft altitude	real
sub-spacecraft latitude	real
true anomaly angle	real

**Abbildung 5.6:** *Struktur der Point Info Tabelle.* Kursiv sind die Primärschlüssel der Tabelle dargestellt

Die **Point-Info** Tabelle enthält die Daten der vorausberechneten Orbit Parameter für jeden Punkt einer Orbit Gruppe, wobei eine Orbit Gruppe einem „Upload Kommando“ entspricht. Nimmt man z.B. an, ein Kommando betraf die Orbits 400 - 450 und die Punkte innerhalb der Orbits wurden mit 1 - 700 vom Norden in den Süden numeriert, so wäre etwa die Breite des ersten Punktes 89 Grad, die des zweiten 88,75 usw. - dies ist für alle Punkte der Orbits 400 - 450 gleich. Deswegen wird diese Information für alle Punkte einer Orbit Gruppe gespeichert. Die genauen enthaltenen Daten sowie die jeweiligen Datentypen sind aus Abbildung 5.6 ersichtlich. Zu jedem Orbit gehören nun n Orbit Punkte, daher die 1:n Relation auf die Orbit Points Tabelle.

<b>Orbit Points</b>	
<i>upload number</i>	char(10)
<i>orbit number</i>	integer
<i>command sequence number (point number)</i>	integer
spacecraft event time	char(23)
SAR antenna boresight longitude	real
sub-spacecraft longitude	real

**Abbildung 5.7:** Struktur der Orbit Points Tabelle. Kursiv sind die Primärschlüssel der Tabelle dargestellt.

<b>Orbit Info</b>	
<i>orbit number</i>	integer
ascending node longitude	real
look direction	char(10)
start time of mapping	char(23)
stop time of mapping	char(23)
navigation solution ID	char(40)
orbital eccentricity	real
orbital inclination	real
orbital semi-major axis	real
start time of orbit	char(23)
stop time of orbit	char(23)
periapsis argument angle	real
time of periapsis	char(23)
start time of polar mapping	char(23)
stop time of polar mapping	char(23)

**Abbildung 5.8:** Struktur der Orbit Info Tabelle. Kursiv ist der Primärschlüssel dargestellt.

In der **Orbit Points** Tabelle sind die wenigen Daten gespeichert, die für jeden einzelnen Orbit Punkt einer Kommando Gruppe unterschiedlich sind. Somit sind hier zu jedem Orbit alle enthaltenen Punkte gespeichert (~700/Orbit), was einer Gesamtzahl von 1.796.466 Einträgen bzw. 98,1 MByte Daten entspricht. Die Struktur der Tabelle ist Abbildung 5.7 zu entnehmen. Dies ist die größte Tabelle

innerhalb der Magellan Orbit Metadaten.

<b>Gap Info</b>	
<i>orbit number</i>	integer
start latitude	real
start longitude	real
stop latitude	real
stop longitude	real
software version ID	integer
data gap length	real

**Abbildung 5.9:** Struktur der Gap Info Tabelle. Kursiv ist der Primärschlüssel dargestellt.

Die **Orbit Info** Tabelle beinhaltet alle spezifischen Informationen zu den einzelnen Orbits. Die 9455 Einträge entsprechen den abgesetzten Kommandos um Orbits in drei Zyklen aufzuzeichnen. Allerdings enthalten nur ca. 5200 tatsächlich brauchbares Bildmaterial. Die Struktur der Tabelle ist in Abbildung 5.8 zu sehen. Jeder Orbit besteht aus n Punkten - dies bewirkt die 1:n Relation auf die Orbit Points Tabelle.

Die letzte Tabelle - **Gap Info** - enthält Informationen über Aussetzer des Aufnahmesensors. Die jeweiligen Stellen zeichnen sich als schwarze Flächen im Bildmaterial aus und sind z.B. bei Bildbestellungen nutzloser Ballast. Eine Auswertung dieser Tabelle ermöglicht die fast vollständige Rekonstruktion der Planetenoberfläche, da normalerweise zumindest aus einem der drei Zyklen Bildmaterial vorhanden ist [Walcher, 96]. Der Aufbau der Tabelle ist in Abbildung 5.9 dargestellt. Jeder Orbit kann n Löcher aufweisen, daher die n:1 Relation auf die Orbit Info Tabelle.

Insgesamt sind in den Magellan Metadaten 7 Tabellen enthalten, die zusammen 110,879 MB an Daten beinhalten. Eine Gesamtübersicht ist in Tabelle 5.9 zu finden.

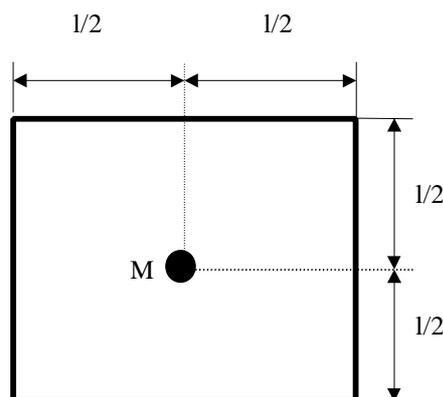
Tabellen Name	Anzahl der Einträge	Größe in MB
Uplink	1391	0,076
Upload Info	196	0,004
Downlink	2334	0,117
Point Info	84.926	9,9
Orbit Points	1.796.466	98,1
Orbit Info	9.455	1,5
Gap Info	19.077	1,2

**Tabelle 5.9:** Übersicht über die Magellan Metadaten. Insgesamt sind 7 Tabellen vorhanden, die ein Gesamtvolumen von 110,897 MB aufweisen.

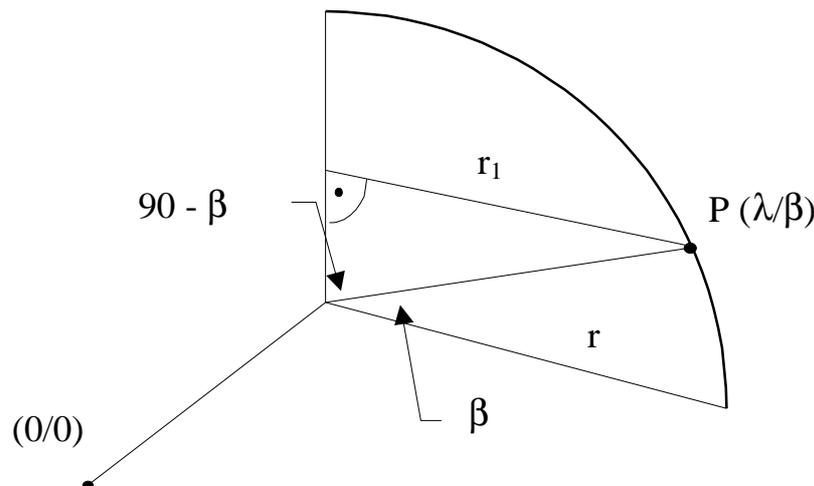
#### 5.1.4 Implementationsdetails

Die Suche nach Orbits für eine Region basiert auf der Definition eines Suchquadrates auf der Oberfläche des Planeten. Dieses Quadrat wird durch den Mittelpunkt M in Längen- und Breitengraden sowie der Ausdehnung  $l$  des Suchquadrates wie in Abbildung 5.10 dargestellt angegeben.

Die Positionen der Punkte sind in den Metatabellen in Grad gespeichert. Da für den Benutzer eine Angabe der Länge in km intuitiver als eine in Grad ist, muß für eine Abfrage die Größe der Abfrageregion in Grad umgerechnet werden. Dies kann am Äquator mit Gleichung 5.1 berechnet werden.



**Abbildung 5.10:** Definition eines Suchquadrates. Das Quadrat wird durch den Mittelpunkt M und die Ausdehnung  $l$  bestimmt.



**Abbildung 5.11:** Berechnung des Verkürzungsfaktors für den Umfang in Abhängigkeit vom Breitengrad. Der dicke Bogen kennzeichnet die Planetenoberfläche,  $r$  ist der Radius des Planeten,  $\beta$  ist der Breitengrad des Punktes  $P (\lambda/\beta)$ .  $\lambda$  ist die Länge,  $\beta$  ist die Breite in Grad.

$$L_{grad} = \frac{U}{360}$$

**Gleichung 5.1:** Umrechnung einer gegebenen Strecke entlang des Äquators in Grad

$U$  ist der Umfang des Planeten, dieser ist im Falle der Venus 38.024,581 km. Um eine Umrechnung ober- bzw. unterhalb des Äquators korrekt durchzuführen, muß man die Berechnung mit dem jeweils um einen Faktor verkürzten Umfang vornehmen. Dieser Faktor berechnet sich aus dem Verhältnis des Planetenradius  $r$  zu dem verkürzten Wert  $r_1$ , der sich an der Stelle des Punktes  $P$  ergibt - siehe hierzu Abbildung 5.11.

Das Verhältnis  $r : r_1$  ergibt sich gemäß Gleichung 5.2 als  $\cos(\beta)$ .

$$r = r_1 * \sin(90 - \beta) = r_1 * \cos(\beta) \Rightarrow \frac{r}{r_1} = \cos(\beta)$$

**Gleichung 5.2:** Berechnung des Umrechnungsfaktors - Länge in Grad - oberhalb des Äquators

Somit kann man die Länge  $L_g$  (Länge in Grad) des Suchquadrates mit bekanntem  $l/2$  und gegebenem Breitengrad  $\beta$  (siehe Abbildung) gemäß Gleichung 5.3 berechnen.

$$L_{grad} = \frac{\frac{l}{2} * 360}{U} * \cos(\beta)$$

**Gleichung 5.3:** Gesamtformel zur Längenberechnung einer Strecke in Grad auf einer beliebigen Stelle auf dem Planeten.

Um eine Abfrage mit einem Suchquadrat vorzunehmen, bei der man alle Orbits

erhalten soll, von denen sich zumindest ein Orbit Punkt innerhalb des Quadrates befindet, wurde wie folgt eine SQL [Date, 93] Abfrage in drei Schritten formuliert:

1. Finde alle Punkte, die innerhalb des Suchquadrates liegen
2. Finde einen Punkt pro Orbit, verwirfe die übrigen
3. Suche die zu den Punkten aus (2) jeweils gehörige Orbit Information

Der Schritt (1) läßt sich mit dem folgenden SQL Statement durchführen, wenn das Quadrat durch  $((L_1, B_1), (L_2, B_2))$  festgelegt ist:

```
SELECT ORBITPT.ORBIT_NUMBER
  FROM ORBITPT, PTINFO
 WHERE (ORBITPT.SAR_ANTENNA_BORESIGHT_LONG>=L1
 AND ORBITPT.SAR_ANTENNA_BORESIGHT_LONG<=L2
 AND PTINFO.SAR_ANTENNA_BORESIGHT_LATITUDE>=B1
 AND PTINFO.SAR_ANTENNA_BORESIGHT_LATITUDE<=B2
 AND PTINFO.UPLOAD_NUMBER = GEO_ORBITPT.UPLOAD_NUMBER
 AND
          PTINFO.COMMAND_SEQUENCE_NUMBER
ORBITPT.COMMAND_SEQUENCE_NUMBER) =
```

Möchte man die Duplikate<sup>1</sup> entfernen (Schritt (2)), so ergänzt man Zeile 1 einfach durch das Schlüsselwort DISTINCT, welches automatisch alle Doppel eliminiert, also:

```
SELECT DISTINCT ORBITPT.ORBIT_NUMBER
[ ... wie oben ...]
```

Für den Schritt (3) muß man obige SQL Abfrage als Subquery formulieren und die eindeutigen Ergebnisse der Abfrage in eine neue Abfrage füttern. Das geschieht so<sup>2</sup>:

```
SELECT *
FROM ORBITI
WHERE ORBITI.ORBIT_NUMBER IN
  (SELECT DISTINCT ORBITPT.ORBIT_NUMBER
   FROM ORBITPT, PTINFO
   WHERE (ORBITPT.SAR_ANTENNA_BORESIGHT_LONG>= L1
 AND ORBITPT.SAR_ANTENNA_BORESIGHT_LONG<= L2
 AND PTINFO.SAR_ANTENNA_BORESIGHT_LATITUDE>=B1
 AND PTINFO.SAR_ANTENNA_BORESIGHT_LATITUDE<=B2
 AND PTINFO.UPLOAD_NUMBER = ORBITPT.UPLOAD_NUMBER
 AND
          PTINFO.COMMAND_SEQUENCE_NUMBER
ORBITPT.COMMAND_SEQUENCE_NUMBER)) =
```

---

<sup>1</sup> Man bekommt pro Suchquadrat z.B. 10 Punkte vom Orbit 3, 46 Punkte vom Orbit 4 und 23 Punkte vom Orbit 5. Tatsächlich benötigt man aber nur jeweils einen Punkt von den Orbits 3,4 und 5 um an die Orbitinformationen zu kommen. Deswegen muß man die Duplikate verwirfen.

<sup>2</sup> die vorherige Abfrage ist kursiv dargestellt

Dieses Statement liefert die Informationen aller Orbits, die mindestens einen Punkt innerhalb des Suchquadrates haben. Möchte man die Orbits noch zusätzlich nach Zyklen aufteilen, so muß man obige Abfrage noch um die folgende Zeile erweitern:

```
AND ORBIT.LOOK_DIRECTION = '<Zyklus Kennung>'
```

Mögliche Werte der Zyklus Kennung sind: LT-NOM1, LT\_MAXSTO3, LT\_NOM2, LT\_NOM3, LT\_NOMVV3, RIGHT\_C252, LT\_STE. LT-NOM1, LT\_MAXSTO3, LT\_NOM2, LT\_NOM3 und LT\_NOMVV3 liefern Orbits des 1. Zyklus, RIGHT\_C252 liefert Orbits des 2. Zyklus und LT\_STE bzw. LT\_STEREO3 liefern Orbits des 3. Zyklus.

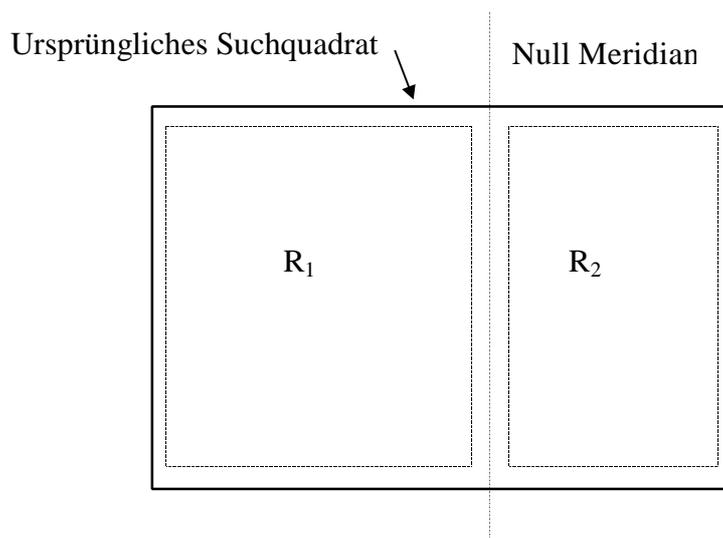
Die eben formulierte Abfrage funktioniert auf der gesamten Planetenoberfläche mit einer Ausnahme: Abfragen, die den Null Meridian überdecken, liefern kein vollständiges Ergebnis. Dies ist sofort einsichtig, wenn man folgendes Beispiel betrachtet. Sei das Suchquadrat  $((L_1, B_1), (L_2/B_2))$  durch  $((355/10), (5/20))$  definiert so würden folgender Teil der Abfrage kein Ergebnis zurückliefern:

```
[...]
WHERE (ORBITPT.SAR_ANTENNA_BORESIGHT_LONG >= 355 (L1)
      AND ORBITPT.SAR_ANTENNA_BORESIGHT_LONG <= 5 (L2))
[...]
```

Es gibt keine reelle Zahl die gleichzeitig  $>$  als 355 und  $<$  als 5 ist. Obige Abfrage würde also insgesamt keine Ergebnisse zurückliefern, obwohl es sehr wahrscheinlich Orbits in besagtem Rechteck geben würde.

Die Lösung des Problems ist die Aufteilung des Suchquadrates in 2 Rechtecke, die durch den Null Meridian begrenzt werden (gemäß Abbildung 5.12). Um nun alle Punkte innerhalb des ursprünglichen Suchquadrates zu erhalten, muß man die Abfrage sowohl auf  $R_1$  als auch auf  $R_2$  anwenden. Die Definition der Rechtecke erfolgt unter Verwendung des obigen Beispiels wie folgt:  $R_1((355/10), (360/20))$ ,  $R_2((0/10), (5,20))$ . Die Ergebnisse werden per UNION addiert.

Abfragen dieser Art bedeuten natürlich einen Performance Verlust, da für ein Suchquadrat praktisch zwei Abfragen getätigt werden müssen. Da dies allerdings mit geringerer Suchgröße gemacht wird, dauert die Abfrage insgesamt kaum länger als eine nicht den Null Meridian überdeckende. Lediglich der Overhead, der aus 2 Übertragungen resultiert, verlangsamt die Abfrage ein wenig.



**Abbildung 5.12:** Problem bei Abfragen, die den Null Meridian überdecken. Das ursprüngliche Suchquadrat muß nun in 2 Rechtecke aufgeteilt werden, die Abfrage auf jedes der beiden Rechtecke angewandt werden und die Ergebnisse werden addiert.

### 5.1.5 Inbetriebnahme des Oracle Enterprise Servers

Im Juli 1996 erfolgte die Installation des Oracle Enterprise Servers Version 7.2.3 von einer CD auf eine SGI Indy. Diese verfügte über eine 200 MHz MIPS RS 4400 CPU und lief unter dem Betriebssystem IRIX 5.3. Die Installation des neuen 64 Bit Betriebssystems IRIX 6.2 ist geplant, hierfür wurde im Dezember 1996 der Oracle Server auf die Version 7.3.2.2 gebracht. Diese Version ist für IRIX 6.2 freigegeben.

Die Installation erfordert prinzipiell zweimal Root Rechte, das erste Mal um eine „dba“ (Data base administrator) Gruppe sowie einen Account anzulegen<sup>1</sup>, das zweite Mal um Post-Installations Schritte auszuführen bzw. einen Port für das SQL NET V2 einzurichten. Die gesamte Installation erforderte 468,944 MB an Plattenplatz, die Magellan Metadaten benötigten dann rund weitere 320 MB, was ein Gesamtvolumen von ca. 800 MB ergibt. Dieses Volumen ist für einen zentralen Server als äußerst gering anzusehen, zieht man die laufend fallenden Preise für Speichermedien in Betracht. Weiters bietet Oracle eine große Funktionsvielfalt sowie eine Skalierung der Datenbanksoftware, so daß der Server auch bei starker Inanspruchnahme nicht in die Knie geht. Folgende Produkte wurden installiert:

- Multimedia APIs
- ORACLE Common Libraries and Utilities
- Oracle Help
- Oracle Office Automation Core

<sup>1</sup> unter dem die Serversoftware installiert werden kann

- Oracle Server Manager
- Oracle Server Manager (Motif)
- Oracle Trace
- Oracle Unix Installer and Documentation Viewer
- Oracle WebServer
- ORACLE7 Parallel Query option
- ORACLE7 Server (RDBMS)
- PL/SQL V2
- Precomp
- Pro\*C
- Remote Operations
- SLAX: parser
- SQL\*Net (V2)
- SQL\*Plus
- TCP/IP Protocol Adapter (V2)
- ToolKit 2.1 Base
- ToolKit 2.1 Extension

Von der Developer 2000 CD wurde noch der Oracle Browser installiert. Dieses Produkt ermöglicht eine graphische Visualisierung der erstellten Tabellen sowie das Erstellen von Schemata, das komfortable Abfragen von Daten sowie das Editieren selbiger. Das Pro\*C Tool sowie der Precompiler wurden für das Einbringen von Quicklooks benötigt, welches im Kapitel0 näher erläutert wird.

Für den GDSS Prototypen wurde aus Mangel an Plattenplatz auf die wichtigen Optionen „Parallel Server“ sowie „Distributed Data“ und „Advanced Replication“ verzichtet. Diese Optionen ermöglichen eine Skalierung des zentralen Servers gemäß der Benutzerkapazität und der Menge der zu verwaltenden Daten. Da der erste GDSS Prototyp als Testdatensatz allerdings nur den Magellan Datensatz verwalten mußte, waren diese Erweiterungen vorläufig nicht nötig, zumal sich die Anzahl der Benutzer Abfragen ausschließlich auf wissenschaftliche Testpersonen beschränkte. Für ein kommerziellen GDSS sind diese Optionen jedoch sehr wichtig.

Sobald die Grundinstallation abgeschlossen war, wurden folgende weitere Schritte getätigt:

- Einrichtung von „Tablespace“ mit dem Server Manager für den Magellan Datensatz. Insgesamt wurden rund 340 MB an Platz aufgeteilt auf 5 Datenfiles reserviert.
- Erzeugung von Benutzer Accounts (vorläufig meinen eigenen und den eines Diplomanden)
- Kreation des Datenbankschemas (siehe Kapitel5.1.3) sowie der Relationen und

der Primärschlüssel mit Hilfe des Oracle Browsers von der Developer CD.

- Einspielen der Magellan Metadaten unter Verwendung des „SQL-Load“<sup>d</sup> Programmes. Hierfür mußte man pro Datensatz ein entsprechendes Script File schreiben, welches die Datentypen sowie den Einspielvorgang beschrieb.
- Installation des SQL Net V2. Ein umfangreicher Schritt, der für die Anbindung des Oracle Servers an das TCP/IP Protokoll notwendig war. Über diese Schnittstelle wurde dann in Folge der Weblogic Server angebunden (siehe Kapitel 5.3)
- Inbetriebnahme des Weblogic Servers. Diese Software ermöglicht die Anbindung von Java Applets über JDBC an Sybase, Informix, MS SQL Server sowie Oracle.
- Der letzte Schritt war die Implementierung des Java Client Programmes, eine nähere Beschreibung erfolgt in den Kapiteln 5.2 und 5.4.

Der erste Prototyp des GDSS sollte auch die Speicherung und die Abfrage von Bildern ermöglichen. Oracle bietet hierzu die Möglichkeit, Bilder als sogenannte BLOBs (Binary Large Objects) in einer Tabelle zu speichern - dies wird jedoch nicht direkt unterstützt, weswegen ein kleines C Programm geschrieben werden mußte welches eine entsprechende Tabelle erzeugt und Bilddaten einspielen kann. Dieses Programm wird im nächsten Abschnitt übersichtlich beschrieben.

### 5.1.6 Einbringung von BLOBs in die Oracle Datenbank

Prinzipiell ist das relationale Datenbankmodell nicht auf die Speicherung von räumlichen Daten (Bilddaten) ausgerichtet. Viele Datenbankhersteller - u.a. auch Oracle - haben jedoch erkannt, daß die Speicherung von Bilddaten im multimedialen Zeitalter ein immer wichtiger werdendes Funktionsmerkmal ist. Sie griffen daher zu einem Behelf, der es dem Benutzer letztendlich gestattete, sogenannte BLOBs (binary large objects) in eine Datenbank einzufügen. Ein BLOB ist ein abstrakter Datentyp, der beliebige binäre Information (Sprache, Daten, Bilder, ...) aufnehmen kann. Beim Oracle RDBMS ist der Datentyp derzeit auf 2 GB pro Objekt beschränkt, was jedoch für GDSS mehr als ausreichend ist, da hier ja nur JPEG komprimierte Quicklooks in der Größenordnung von einigen kB zur Verfügung gestellt werden sollen.

Obwohl der Datentyp „long raw“ in Oracle für die Aufnahme von binären Daten vorhanden ist, gibt es doch keine direkte Unterstützung, binäre Daten einzubringen. Um dies zu bewerkstelligen wurde ein C Programm (storeimage) entwickelt, welches folgende Funktionen erfüllt:

- Erstellung einer „Image“ Tabelle,

---

<sup>1</sup> Für eine genaue Beschreibung siehe „Oracle7 Server Utility Users's Guide - Release 7.1“, Kapitel 4

- Einfügen von JPEG Bildern in die Datenbank,
- Herausholen von JPEG Bildern,
- Auflisten des vorhandenen Bestandes an Bildern.

Die „Image“ Tabelle hat die in Abbildung 5.13 dargestellte Struktur.

<b>ImageTable</b>	
<i>BLOBSKEY</i>	VARCHAR2(30)
BLOBTYPE	VARCHAR2(5)
HEIGHT	NUMBER
WIDTH	NUMBER
IMG_SIZE	NUMBER
CREATOR	VARCHAR2(20)
CREATED	DATE
MODIFIED	DATE
MODIFIED_BY	VARCHAR2(20)
BLOB	LONG RAW

**Abbildung 5.13:** *Struktur der ImageTable Tabelle.* Die Bedeutung der einzelnen Attribute ist intuitiv klar bis auf den Blobtype. Hier wird das Format des Bildes gespeichert - also etwa: JPEG, GIF, TIF, etc.

Das „imagestore“ Programm wurde mit dem Oracle Pro\*C Compiler in ein C Programm übersetzt und dann mit den entsprechenden Oracle Libraries verlinkt. Das lauffähige Programm sowie der Quellcode befinden sich am Oracle Account des ICG. Probleme bei der Installation des Oracle Enterprise Servers unter IRIX 5.2 bzw. 6.2 sind im Anhang B zusammengefaßt.

## 5.2 Java Programme und Applets

Seit der Ankündigung von Java bei SunWorld im Jahre 1995 ist die Popularität des Produktes in knapp zwei Jahren geradezu unglaublich gewachsen. Noch bevor der erste Java Compiler im Jänner 1996 herauskam, wurde Java bereits zum Industrie Standard für Internet Applikationen deklariert [Vanhel, 96]. Bereits in der ersten Hälfte von 1996 wurde Java von einer Reihe führender Software Unternehmen von Sun Microsystems lizenziert. Dazu zählten: Adobe, Asymetrix, Borland, IBM, Macromedia, Metrowerks, Microsoft, Novell, Oracle, Spyglass und Symantec. Diese und andere Hersteller planen, Java in ihre Betriebssysteme und Desktop-Produkte zu integrieren.

Diese Verbreitung war natürlich ein entscheidender Grund, warum für den GDSS

Prototypen Java als Programmiersprache ausgewählt wurde. Java bietet aber noch mehr Vorteile, die da wären:

- Plattformunabhängigkeit,
- Flexibilität,
- Netzwerkorientiertheit,
- Sicherheit,
- Offenheit.

Vor allem die Plattformunabhängigkeit und die Offenheit, die Java durch seine zahlreichen APIs (Application Programmer Interface) erhält, sind zwei weitere Gründe, welche die Wahl von Java für das GDSS rechtfertigen. Unter APIs versteht man spezielle Klassen und Methoden, die es dem Java Programm ermöglichen, mit der realen Welt zu kommunizieren. Eine genauere Beschreibung der APIs würde den Rahmen dieser Arbeit sprengen, eine Übersicht kann man in [Walcher, 96] finden - eine genaue Beschreibung in [Vanhel, 96] bzw. auf den Java Webseiten [Java, 97]. Weitere zu empfehlende Literatur wäre [Arnold, 96] und [Back, 96]. Ein Überblick über die historische Entwicklung von Java ist im Anhang C zu finden.

### 5.2.1 Java Applets

Wie bereits erwähnt, ist Java eine eigene Programmiersprache, mit der man eigene alleinstehenden Applikationen programmieren und compilieren kann. Gerade in Zusammenhang mit dem WWW gibt es ein weiteres wichtiges Merkmal von Java zu besprechen, das „Applet“.

Im Gegensatz zu herkömmlichen Compilern, die Programme für eine spezifische Plattform bzw. für ein spezifisches Betriebssystem compilieren, übersetzt ein Java Compiler den Quellcode in ein von der Hardware Architektur unabhängigen Bytecode. Dieser Byte Code kann nur von der sogenannten Java „Virtual Machine“ (VM) ausgeführt werden, einem normalerweise in Software programmierten Java Prozessor. Es gibt von Sun bereits auch schon die Ankündigung für einen in Hardware programmierten Java Chip der in der Sun Java Station eingebaut ist (siehe Abbildung 5.14). Doch obwohl es bereits einen Preis für die Java Station gibt (unter \$1000) war der Chip Anfang Jänner 1997 noch nicht verfügbar.

Ein „Applet“ besteht nun aus einem solchen Bytecode, der bei Bedarf vom jeweiligen WWW Server mit Hilfe eines „Class-Loaders“ heruntergeladen wird. In der Regel geschieht dies über den Aufruf auf einer HTML Seite mit dem <applet> Tag, welches ein Java fähiger Browser (z.B. Netscape, MS Internet Explorer, Hot Java) interpretieren kann. Selbiger Browser verfügt auch über eine Java VM, die den erhaltenen Bytecode sodann plattformunabhängig exekutiert.



**Abbildung 5.14:** *Java Station.* Die von Sun angekündigte Java Station, die über einen in Hardware realisierten Java Chip verfügt. Bis Anfang Jänner 97 war der Chip allerdings noch nicht verfügbar. (Quelle: <URL: <http://www.sun.com>>)

Zuvor wird der erhaltene Code jedoch noch einer Verifizierung unterworfen, um sicherzugehen, daß das geladene Applet keine unerlaubten Speicherzugriffe macht bzw. korrekten Code enthält. Dies verlängert die Ladedauer, beeinträchtigt jedoch nicht die Exekutionsgeschwindigkeit des Programmes. Die VM des Compilers kann prinzipiell auf zwei Arten implementiert sein: mit einem Interpreter oder mit einem „Just in Time“ (JIT) Compiler. Ersterer ist die langsamere aber einfachere Lösung, jeder Befehl wird analysiert, die damit assoziierte Funktion wird aufgerufen. Dies ist mit einem Interpreter vergleichbar. Mit einem JIT Compiler werden die Befehle kurz vor der Exekution in die plattformeigene Sprache übersetzt, was einen deutlichen Performance Gewinn mit sich bringt. Da die JIT Compiler auf der Benutzer Plattform laufen, bleibt die Portabilität des Codes erhalten [vanhel, 96].

Ein weiterer Vorteil von Java Applets ist der Wegfall von zeitintensiven Updates. Bei einem konventionellem Client-Server System mußte der System Administrator die Klienten-Software auf jeder einzelnen Plattform austauschen. Ein Applet wird automatisch bei jedem Start des Browsers vom Server geladen und dann in Folge lokal exekutiert. Man erhält also stets die neueste Version. Auf Grund der eben genannten Vorteile wurde für den GDSS Prototypen ein Java Applet als Klient Anwendung verwendet.

### 5.2.2 Sicherheitsaspekte

Eines der wichtigsten Entwurfsmerkmale von Java war der Sicherheitsaspekt. Gerade wenn man seinem System Zugang zu einem weltweiten Netz verschafft, steigt die Gefahr von Viren und Hackversuchen unkontrollierbar an. Dies wirkt sich besonders bei Betriebssystemen aus, die kaum über eingebaute Schutzmechanismen verfügen, so etwa MS-DOS in Kombination mit Windows 3.11 oder Windows 95. Windows NT verfügt über einige Sicherheitsmechanismen, während UNIX schon über durchaus ausreichende Sicherheitsmerkmale verfügt.

Dennoch erscheint das Herunterladen und Exekutieren von fremden Programmen auf dem eigenen Computer auf den ersten Blick als bedenklich, könnte es sich doch

hierbei auch um ein sogenanntes „trojanisches Pferd“ handeln. So könnte etwa ein harmlos erscheinender Bildformat Konvertierer einen Prozeß starten, der im Hintergrund läuft, und die Festplatte formatiert oder Daten löscht. Diese Ängste sind jedoch zum größten Teil unbegründet, da Java über sehr gute Sicherheitsmechanismen verfügt.

Insgesamt besteht das Java Sicherheitssystem aus drei Komponenten: dem „Class-Loader“, dem Bytecode Verifizierer und dem Sicherheitsmanager. Die zweite Komponente, der Bytecode Verifizierer, wurde bereits im Kapitel 5.2.1 behandelt. Sie stellt sicher, daß das Java Programm korrekt kompiliert wurde und die Restriktionen der VM befolgt. Der „Class Loader“ separiert Applets, die von verschiedenen Servern - und auch vom eigenen - geladen wurden voneinander. Dies verhindert, daß eine Klasse sich als eine im Browser eingebaute Klasse ausgeben kann und dementsprechend mehr Rechte bekommen kann. Der Sicherheitsmanager schließlich implementiert einen Sicherheitsalgorithmus, der die lokale VM anweist, welche Aktionen unter welchen Umständen getätigt werden dürfen. Ein Schlüssel Aspekt wäre zum Beispiel File I/O. Solche Anfragen werden immer zuerst zum Sicherheitsmanager geschickt, wo entschieden wird, ob dem Java Programm vertraut werden kann oder nicht. Der Sicherheitsmanager ist eine voll in Java geschriebene Klasse und kann bei Bedarf ersetzt werden. Selbstverständlich nicht von einem fremden Applet. So erlaubt z.B. der Sicherheitsmanager von Netscape auf der Klient Seite überhaupt keinen File I/O. Dies erfüllt natürlich einen hohen Sicherheitsstandard, ist aber eine grobe Restriktion von Java Applets. Doch wie gesagt ist der Manager austauschbar und den jeweiligen Anforderungen anpaßbar.

### 5.2.3 Kritische Bewertung

Trotz der umfangreichen Sicherheitsvorkehrungen gab es in der Version 1.01 des JDK (Java Development Kit) noch deutliche Sicherheitslecks [McGraw, 96]. So gibt es etwa Applets, die im Hintergrund weiterlaufen<sup>2</sup>, ein Fenster ohne die Warnung „Untrusted Java Window“ aufmachen und nach einem Login/Passwort verlangen. Alle eingegebenen Paßwörter werden sodann als E-mail an eine beliebige Adresse verschickt.

Weiters gibt es auch noch jede Menge Fehler in der derzeitigen Implementierung des Sicherheitsmanagers vom Netscape Navigator. Entsprechende Applets, die mehr oder weniger geringen Schaden anrichten, sind unter [Applet, 96] zu finden, diese Seite ist allerdings mit Vorsicht zu genießen! Man sollte diese Seite wirklich nur mit einem einigermaßen sicheren Betriebssystem wie UNIX ansehen<sup>3</sup>, will man nicht

---

<sup>1</sup> Das ist ein Programm, welches etwas anderes tut, als es vorgibt zu tun

<sup>2</sup> Applets werden nicht beendet, wenn man eine neue WWW Seite ladet!

<sup>3</sup> Selbst unter UNIX kann es zu Problemen kommen - ein Backup von wichtigen Daten am Account ist unbedingt anzuraten!

Datenverluste oder fortan abnormales Benehmen des PC riskieren. Es ist zu erwarten, daß zwar die meisten Sicherheitslecks in der Version 1.1 des JDK gestopft sind, es jedoch mit jedem neuen Feature von Java neue Möglichkeiten einer Attacke geben wird.

Auch Java Viren existieren bereits und sind in ihrer Gefährlichkeit nicht zu unterschätzen. Bei sorgfältiger Programmierung und unter Beachtung aller von Java angebotenen Sicherheitsmechanismen ist das Risiko jedoch in akzeptierbarem Rahmen zu halten [LaDue, 96]. Alles in allem ist Java eine aufstrebende Programmiersprache, nicht zuletzt wegen ihrer Plattformunabhängigkeit und Offenheit.

Derzeit ist die Exekution von Java Programmen noch recht langsam, da meistens Interpreter und in Software implementierte VM's verwendet werden. Jedoch hat Sun bereits die Verfügbarkeit der Java Station mit dem ersten Quartal 1997 angekündigt. Diese Maschine verfügt über eine in Hardware implementierte VM, die eine Programm Exekution entscheidend verschnellern kann. Bis jetzt war Java eher zu Steuer Zwecken verwendet worden, rechen-intensive Applikationen wurden nur äußerst langsam exekutiert. Mit dem Java Chip dürfte sich das ändern.

Insgesamt ist Java mit den sogenannten Java Applets aufgrund folgender Vorteile eine gute Wahl: Plattformunabhängigkeit, Flexibilität, Netzwerkorientiertheit, Sicherheit und Offenheit - alles zusammen wichtige Design Aspekte die völlig konform mit jenen des GDSS sind.

### **5.3 Client-Server Kommunikation unter Verwendung von JDBC**

Nach der Wahl des DBMS und der Festlegung der Klienten Software auf Java erfolgte nun der dritte Schritt in der Implementierung des GDSS Prototypen: die Schaffung einer Verbindung zwischen dem Java Klient Programm und dem Oracle DBMS. Um das in Kapitel 3 definierte Anforderungsprofil des GDSS nicht zu verletzen, mußte eine standardisierte und gleichzeitig offene Lösung gefunden werden, die keinesfalls starr an die Oracle Datenbank gebunden sein durfte.

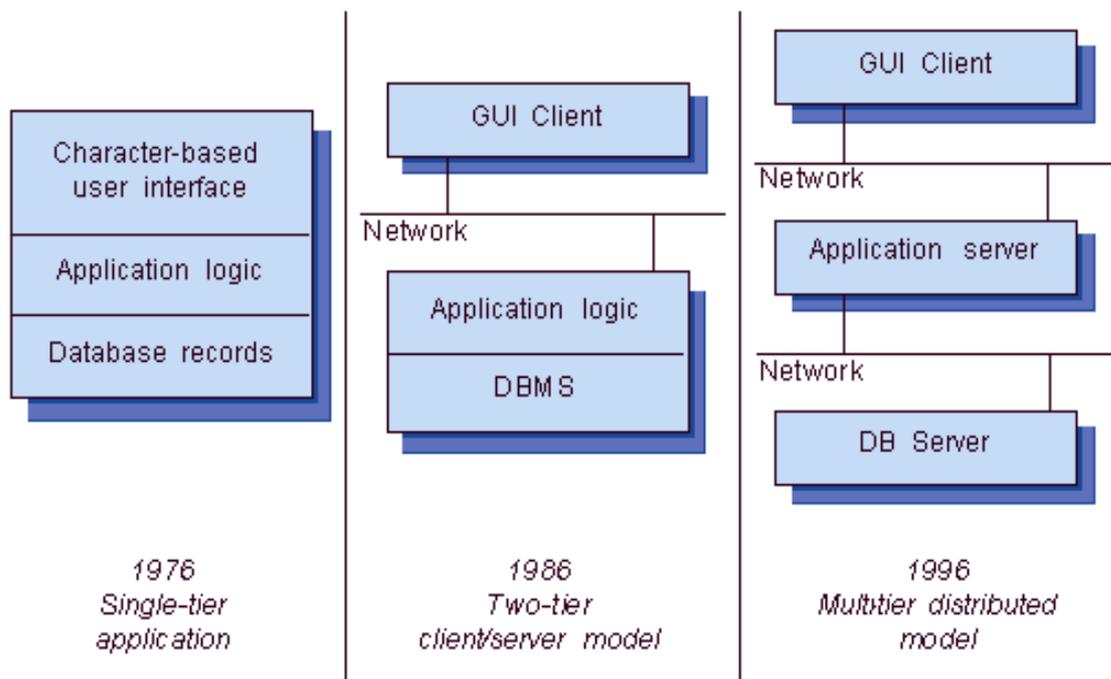
Die folgenden Unterkapitel stellen eine genau solche Lösung unter Verwendung des JDBC (Java Data Base Connectivity) API von Java in Verbindung mit dem Weblogic Application Server vor.

#### **5.3.1 Datenbankkommunikation eines offenen Systems**

Die ersten funktionierenden RDBMS kamen in den frühen 80er Jahren auf den Markt und basierten auf einem einfachen, aber auch sehr ineffizientem Prinzip: jede Applikation, die auf ein RDBMS zugreifen wollte, mußte dies über den Aufruf von eigenen Datenbank Bibliotheksroutinen oder gar über den Start einer eigenen Instanz des RDBMS bewerkstelligen. Diese Lösung, auch 1-tier Logik genannt, verfügte über eine ganze Reihe von signifikanten Nachteilen: äußerst ineffizient im

Umgang mit Rechner Ressourcen, nicht geeignet für einen Einsatz in einem Netzwerk und somit konnte kaum Unterstützung für plattformübergreifende Transaktionen zwischen mehreren Benutzern geboten werden. Diese Lösung ist in Abbildung 5.15 skizziert.

Bereits in den späten 80er Jahren wurde die 1-tier Logik modifiziert - es erfolgte eine Auftrennung in ein bis in die 90er Jahre typisches Client/Server System (siehe auch Abbildung 5.15). Ein solches System ist durch einen (multithreaded) Server charakterisiert, der vor den Klient Programme gestartet wird. Der Klient sendet in der Folge Abfragen, Daten und Updates in Form von SQL Statements an den Server, welcher dieselbigen ausführt und die Resultate an denselbigen zurückschickt. Die Visualisierung erfolgt auf der Klient Seite meist mittels komfortabler GUIs. Dieses Konzept war weitaus sparsamer mit der Ausnutzung von lokalen Ressourcen und außerdem viel leichter zu koordinieren bzw. in vorhandene



**Abbildung 5.15:** Von der 1-tier Logik zur n-tier (oder multitier) Logik. Das Diagramm zeigt die Entwicklung von der 1-tier Logik, bei der sich das GUI, die Applikations Logik sowie das DBMS auf einer Maschine befinden, hin zur multitier-Logik, bei der das Klient Programm, der Applikations Server und das DBMS sich auf verschiedenen Maschinen befinden und über ein Netzwerk verbunden sind. (Quelle: <URL: <http://www.weblogic.com>>)

LAN Netzwerk Umgebungen einzubinden.

Das Client/Server Konzept wurde in den letzten 10 Jahren äußerst oft und auch effizient eingesetzt, allerdings ergaben sich mit der Weiterentwicklung der

Computer Software folgende Probleme:

- In Client/Server Umgebungen versorgt jeder Hersteller sein Produkt mit eigenen Bibliotheken (Libraries) und APIs - es gibt somit keine genormte Client/Server Kommunikation sondern nur eine herstellerspezifische. ODBC (Open Data Base Connectivity) von Microsoft ist ein erster Schritt die Kommunikation zu standardisieren - jedoch ist diese Lösung derzeit weder vollständig implementiert noch standardisiert.
- Klient Bibliotheken und ODBC haben eine sehr niedrige Abstraktionsstufe. Der Programmierer muß sich auf die Ebene einzelner Records hinunter begeben, anstatt high-level Befehle verwenden zu können.
- In großen Firmen ist Klient Software auf vielen hundert PCs installiert. Jedesmal wenn eine neue Version des DBMS herauskommt muß man jeden einzelnen PC updaten. Dies ist mit erheblichem Aufwand und Kosten verbunden.
- Große Firmen haben üblicherweise eine Vernetzung zu ihren Filialen, die über ein langsames WAN angebunden sind. Werden nun nur einzelne low-level SQL Statements geschickt, bedeutet das eine deutliche Leistungseinbuße im Gegensatz zum Schicken von high-level Statements.
- Klient PCs sind in ihrer Speicherkapazität (Memory und Platte) beschränkt. Immer aufwendigere Client Software erfordert ein oftmaliges Erneuern der Hardware, was natürlich mit erheblichen Kosten verbunden ist.

Alle oben genannten Nachteile lassen sich mit einer entsprechenden 3-tier (oder multitier) Logik beheben. Die multitier-Logik trennt, wie in Abbildung dargestellt, den Benutzer Klient vom Applikations Server und dem DBMS. Dies hat nun den entscheidenden Vorteil, daß applikationsspezifische Bibliotheken beim sogenannten „Application Server“ verweilen, der Klient kann herstellerunabhängige high-level Kommandos an den Applications Server schicken. Dieser bildet dann die Abfragen mit Hilfe der herstellerabhängigen Bibliotheksroutinen auf die entsprechenden DBMS ab.

Dies ermöglicht letzten Endes den Anschluß von beliebigen relationalen Datenbanken an den Application Server, auf die alle mitsamt durch ein und dieselbe Klient Software zugegriffen werden kann. Somit fällt natürlich auch der Speicherplatzbedarf auf der Klient Seite für die Bibliotheken weg. Den WAN Netzwerkzugriff kann man mit Hilfe eines entsprechenden Protokolls und einem entsprechend mit Cache ausgestattetem Application Server ebenfalls entscheidend beschleunigen (etwa die einzelnen Abfragen zu Paketen zusammenfassen). Das Update Problem bekommt man unter Verwendung von Java Applets (siehe Kapitel 5.2) in den Griff - hierbei wird stets die aktuellste Applet Version vom Server geladen, lokale Wartung entfällt völlig.

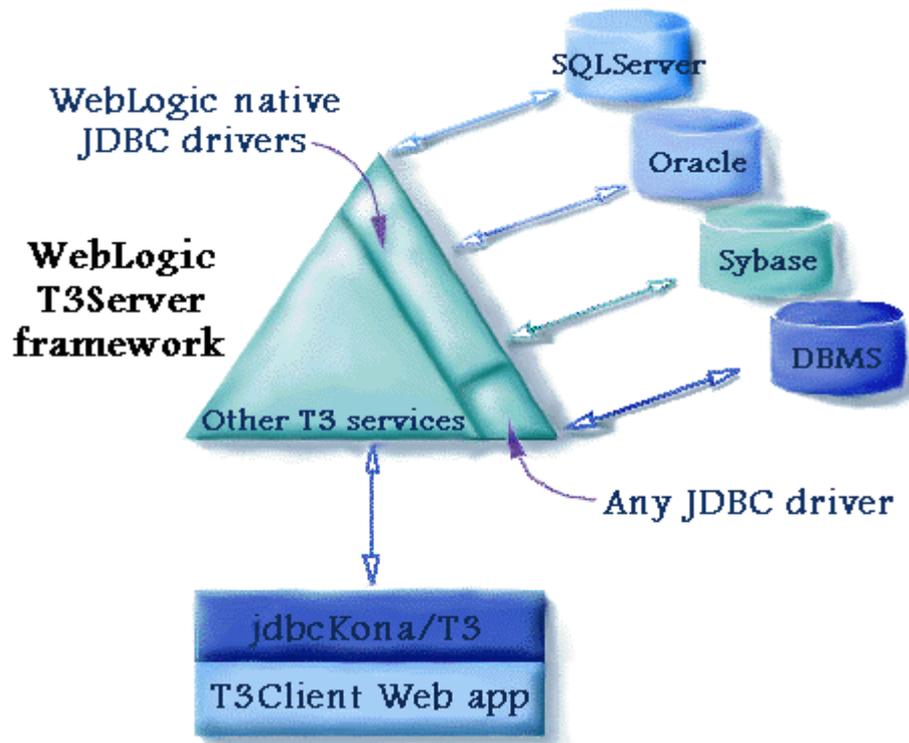
Um schließlich noch das wichtige Entwurfskriterium des internationalen Standards zu erfüllen, wurde das von Sun im Juli 1996 vorgestellte JDBC (Java Data Base Connectivity) verwendet [JDBC, 97]. JDBC ist ein Standard Java API, welches mit einem einheitlichen Interface Zugriffe auf ANSI SQL kompatible Datenbanken aller großen Hersteller ermöglicht. Im Prinzip besteht JDBC aus einem allgemeinen Interface und einem herstellerspezifischen Treiber, der die Standardabfragen unter Verwendung von Bibliotheksfunktionen des herstellerspezifischen Treibers an das jeweilige DBMS weiterleitet.

Folgende Hersteller haben bereits Treiber entwickelt (oder sind dabei einen zu programmieren): Agave Software Design, Asgard Software, Borland International Inc., Bulletproof, Connect, Inc., Cyber SQL Corporation, DataRamp, Dharma Systems Inc., Gupta Corporation, IBM's Database 2 (DB2), Imaginary (mSQL), Informix Software Inc., Net Dynamics, O2 Technology, Object Design Inc., Open Horizon, OpenLink re, Oracle Corporation, Persistence Software, Presence Information Design, PRO-C Inc., Recital Corporation, RogueWave Software Inc., Sanga, SAS Institute Inc. (tm), SCO, StormCloud Development, Sybase Inc., Symantec, Thunderstone, Visigenic Software, WebLogic Inc. und XDB Systems, Inc. Eine genaue Übersicht der bereits verfügbaren Treiber findet man unter [JDBCDr, 97].

In Kombination mit dem eben vorgestellten multitier Konzept bieten Java und das JDBC API mächtige Werkzeuge um plattformunabhängig auf ANSI SQL kompatible Datenbanken beliebiger Hersteller über ein Netzwerk zuzugreifen. Der Quellcode auf der Klient Seite bleibt hierbei weitestgehend unabhängig vom verwendeten relationalen DBMS<sup>1</sup>. Genau diese Lösung wurde mit der jdbcKona/T3 Software von Weblogic Inc. [Weblogic, 96] realisiert. Eine nähere Beschreibung erfolgt im nächsten Abschnitt.

---

<sup>1</sup> Man muß nur das Laden der gewünschten Treiber veranlassen.



**Abbildung 5.16:** Die Weblogic jdbcKona/T3 multitier Lösung. Von einem Java Klienten kann man herstellerunabhängige, high-level Kommandos an den Application Server schicken, der diese dann unter Zuhilfenahme von herstellerabhängigen Bibliotheken an das jeweilige DBMS weiterleitet. Die Antworten werden wieder einheitlich an den Klienten zurückgeliefert. (Graphik von Weblogic. Inc. <URL: <http://www.weblogic.com>>)

### 5.3.2 Implementierung einer multitier Lösung

Nach einer im gegenständlichen Projekt durchgeführten Marktuntersuchung bezüglich multitier Lösung in Kombination mit der Anbindung einer Oracle Datenbank an ein Java JDBC Applet wurde die in San Francisco ansässige Firma Weblogic Inc. ausfindig gemacht [Weblogic, 96]. Weblogic bietet u.a. zwei Software Produkte an, die eine multitier Lösung mit den obigen Vorgaben ermöglichen: jdbcKona/T3 - ein multitier Application Server und jdbcKona/Oracle, eine JDBC Treiber Implementierung für Oracle Datenbanken. Derzeit werden auch noch Treiber für Sybase und MS SQL Server angeboten. Mit dieser Software kann man die in Abbildung 5.16 skizzierte multitier Lösung realisieren.

Die T3-Client Java Applikation schickt unter Verwendung des JDBC APIs einheitliche, high-level Kommandos an den T3 Server. Dieser setzt die Kommandos nun unter Zuhilfenahme von herstellerepezifischen Bibliotheken, die auf dem Application Server verweilen, auf das jeweilige DBMS um. Somit braucht man für

alle Klient Programme nur einmal die herstellerepezifischen Bibliotheken zur Verfügung haben und nicht wie bisher üblich bei jedem Klienten.

Der Weblogic T3 Server ist vollständig in Java geschrieben und somit ebenfalls weitestgehend plattformunabhängig. Ab der Version 2.2 gibt es ein komfortables Server Management Tool, welches in einfacher Art und Weise die Konfiguration und die Wartung des Servers ermöglicht. Das Produkt wird durch eine umfangreiche Beispielsammlung sowie ausgezeichneten On-Line Support via E-mail abgerundet. Eine genauere Beschreibung der Server Software sowie der zahlreichen Fähigkeiten würde den Rahmen dieser Arbeit sprengen, kann jedoch unter [Weblogic2, 96] gefunden werden.

Für den GDSS Prototypen wurde ein jdbcKona/T3 Server in Verbindung mit dem jdbcKona/Oracle Treiber aufgesetzt. Um eine IP Verbindung zum Oracle DBMS zu bekommen, mußte noch zusätzlich der TNS (Transparent Network Service) von Oracle installiert und in Betrieb genommen werden (siehe auch Kapite0). Damit war die Basis für den GDSS Prototypen geschaffen und die weitere Arbeit auf die Programmierung einer geeigneten Klient Software reduziert, welche die aufgebaute Infrastruktur ausnützen konnte. Bevor wir uns aber dem vierten und letzten Schritt (Kapitel 5.4) widmen, werden noch Alternativen/Erweiterungen zu JDBC bzw. in bekannter Manier die kritische Bewertung behandelt.

### 5.3.3 Alternativen und zukünftige Lösungen

Bis Anfang Jänner 1997 hatte Oracle noch keinen fertigen „native“ JDBC Treiber. Zahlreiche Anfragen bei der Hotline ergaben, daß ein Treiber in Entwicklung sei, es aber noch mindestens bis Ende 1996 dauern werde. Am 14. November 1996 hat sich Oracle allerdings mit Weblogic Inc. geeinigt, die Weblogic jdbcKona Produkte auf der Web Server CD ab Version 7.x zu vertreiben. Oracle unterstützt hierbei Weblogic in der Entwicklung des Treibers maßgebend. Man kann vermuten, daß Oracle mit dieser Lösung Auslangen finden wird.

Für zukünftige Erweiterungen wäre natürlich ein JDBC Treiber interessant, der den gerade im Standardisierungsprozeß befindlichen SQL/3 ANSI Standard [SQL3, 96] unterstützt. Auf die resultierenden Vorteile eines OO Datenmodells wurde bereits in Kapitel 5.1.1 ausführlich eingegangen, hier bleibt nur zu sagen daß ein OO Modell sich optimal in das objektorientierte Java einfügen würde, entsprechend umständliche Konvertierungen zwischen dem relationalen und dem OO Modell könnten entfallen. Dies würde sich in einer Leistungssteigerung sowie einer Verschmälerung des Quellcodes auswirken.

Bis Jänner 1997 hat sich keine Entwicklung abgezeichnet, den voraussichtlich 1999 fertig werdenden SQL/3 Standard per JDBC zu unterstützen. Vermutlich ist es aber für solch eine Entwicklung noch zu früh, da bis jetzt nicht einmal noch der relationale JDBC Standard vollständig implementiert ist. Seit November 1996 ist die

JDBC Version 1.10 verfügbar, die auch mit dem JDK 1.1 ausgeliefert wird.

### 5.3.4 Kritische Bewertung und Erfahrungen

Im Juli 1996 wurde der GDSS Prototyp implementiert. Die Weblogic Software war vorerst nur in einer Beta Version verfügbar. Dies wirkte sich zum Teil in fehlerhafter Software aber auch in Installationsschwierigkeiten und schlechter bzw. fehlerhafter Dokumentation aus. Vor allem die Installation einer neuen Version war immer wieder mit großen Änderungen und somit mit viel manueller Arbeit verbunden. Dies hat sich allerdings ab Version 2.2 stabilisiert. Die fehlerhaften Stellen der Dokumentation wurden ausgebessert bzw. durch ausführlichere Beschreibungen ersetzt. Seit August 1996 gibt es auch eine Mailing Liste, mit der registrierte Weblogic Benutzer stets über Neuigkeiten und Patches informiert werden.

Der Preis für den jdbcKona/T3 Server beträgt 1799 Dollar, der des jdbcKona/Oracle Treibers 399 Dollar (Stand Jänner 1997)<sup>1</sup>.

## 5.4 FBIDR Query Tool - Ein GDSS Klienten Prototyp

Basierend auf der bisher aufgebauten Architektur, bestehend aus einem Hochgeschwindigkeits ATM Netz, einem Oracle Datenbank Server sowie einem Weblogic Server und einem zugehörigen JDBC Treiber wurde ein Java Klienten Programm geschrieben, welches einen ersten Zugang zu den Magellan FBIDR Metadaten (siehe auch Kapitel 5.1.2) über das WWW ermöglichen sollte. Das Anforderungsprofil des Programmes lautete wie folgt:

- Eingabe eines Punktes auf der Oberfläche samt einem Gebiet (Point of Interest), welches als Quadrat mit dem Punkt als Zentrum angenommen wird.
- Als Ergebnis sollen alle FBIDR Metadaten geliefert werden, die wenigstens einen FBIDR Punkt im definierten Interessensgebiet (Region of Interest) haben.
- Das Ergebnis muß durch eine Auswahlmöglichkeit der gewünschten Zyklen gefiltert werden können.
- Die Speicherung und das Laden von BLOBS soll mit Hilfe eines Demo-Abfrageknopfes demonstriert werden.

Das Anforderungsprofil wurde bewußt einfach gehalten, da die wissenschaftliche Arbeit nur die Funktionstüchtigkeit demonstrieren soll, und keineswegs ein kommerziell vertreibbares Programm entstehen soll. Es wurde bei der

---

<sup>1</sup> Öffentliche Forschungseinrichtungen und Universitäten bekommen allerdings spezielle Rabatte, die sich zwischen 75% und 50% bewegen. Universitäten können für Lehrzwecke auch eine 6 Monate Gratis Lizenz bekommen, eine solche wurde dem ICG für den GDSS Prototypen verwendet.

Programmierung jedoch besonderer Wert in eine exakte Unterteilung von Modulen sowie eine ausführliche Dokumentation des Quellcodes gelegt. Eine spätere Erweiterung ist somit ohne große Schwierigkeiten durchführbar. Die folgenden 2 Unterkapitel geben einen Überblick über den Programmaufbau und die Funktionalität des Prototypen.

#### 5.4.1 Programmstruktur

Eine bis ins Detail gehende Beschreibung des Programmes kann hier aus Platzgründen nicht erfolgen. Dies ist jedoch gar nicht notwendig, da der Quellcode ohnehin ausgezeichnet dokumentiert ist und einem einigermaßen in Java erfahrenen Programmierer bestimmt keine Verständnis Schwierigkeiten bereiten wird. Hier wird nur ein grober Überblick in die Struktur und Funktionsweise des Programmes gegeben bzw. einige Schlüsselstellen etwas ausführlicher behandelt. Die Kenntnis von Java wird vorausgesetzt und kann aus zahlreichen Büchern (z.B. [Arnold, 96], [Vanhel, 96], [Back, 96]) oder dem WWW [Java, 97] erworben werden.

Das Java Programm ist aus mehreren Modulen aufgebaut, deren logische Struktur in Abbildung 5.17 dargestellt ist. Das GUI Modul regelt Dinge wie das Layout der Buttons, Eingabe Elemente und das Aussehen/die Erzeugung der einzelnen Fenster.

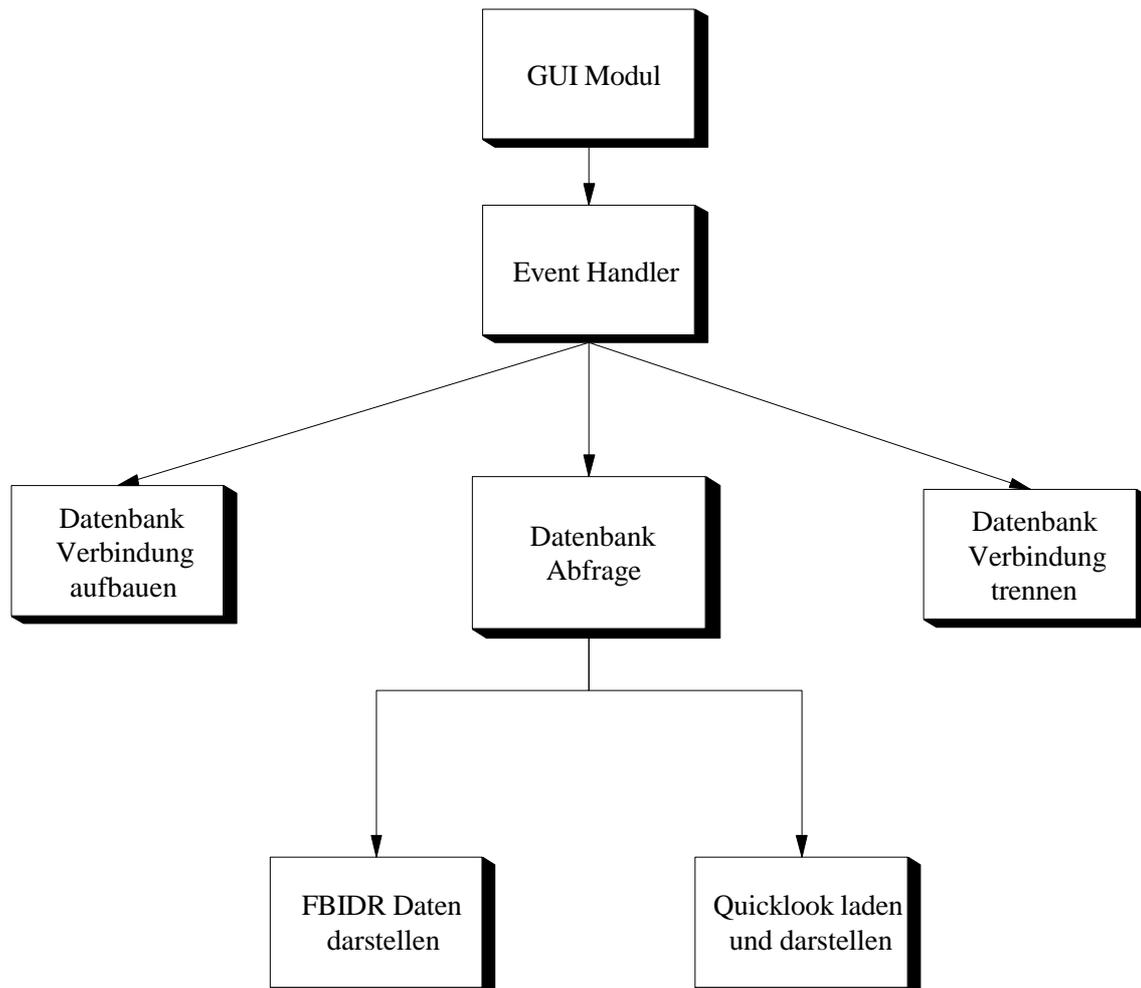
Alle Benutzer Aktionen im GUI werden nun direkt von einem „Event Handler“ (Ereignis Zentrale) ausgewertet. Dieser ist nochmals in einen Keyboard und einen GUI Event Handler unterteilt. Im Prinzip können 3 mögliche Benutzer Aktionen geschehen: die Datenbankverbindung aufbauen, die Datenbank abfragen und die Verbindung trennen.

Das Datenbank Verbindungs Modul ruft zwei weitere Submodule auf, die schrittweise zuerst die Verbindung zum T3 Applikations Server und dann die Verbindung zum DBMS aufbauen.

Die Verbindung zum T3 Server erfolgt im Submodul *public boolean ConnectT3Server (String Server)* und kann mit folgenden Zeilen realisiert werden:

```
// Create a new T3 client
t3 = new T3Client(Server); // Server must be of the format: 't3://<full server name>:port'
// Connect to the T3 client to the T3 server
t3.connect();
```

Die Verbindung zum DBMS erfordert mehr Parameter, vor allem benötigt man auch einen Account auf dem Datenbank Server samt Login und Paßwort. Die Verbindung des Applikations Servers mit dem TNS (Transparent Network Service) von Oracle geschieht in 3 Schritten, die im folgenden kurz erläutert werden.



**Abbildung 5.17:** Logische Module des GDSS Prototypen. Das GUI gibt Ereignisse direkt an einen Event Handler. Dieser verteilt die Benutzer Aktionen an die einzelnen Programm Teile. Siehe Text für nähere Erläuterungen.

Zuerst muß man die Parameter für einen Login in ein sogenanntes „Properties“ Objekt schreiben und den entsprechenden Treiber für den T3 Klient und die Datenbank definieren. Dies geschieht wie folgt:

```
// Set the properties for the DBMS connection
Properties props = new Properties();
props.put("user", login);
props.put("password", passwd);
props.put("server", Dbname);
props.put("weblogic.t3", t3);
```

```
// Classname and URL of the JDBC driver for the T3Server-to-DBMS connection
props.put("weblogic.t3.driver", "weblogic.jdbc.oci.Driver");
props.put("weblogic.t3.url", "jdbc:weblogic:oracle");
props.put("weblogic.t3.cacheRows", cacherows);
```

Anschließend muß man den JDBC Treiber laden ...

```
// Load the JDBC driver
Class.forName("weblogic.jdbc.t3client.Driver");
```

... und beim Treiber Manager registrieren, ...

```
// Make the connection to the T3 server
conn = DriverManager.getConnection("jdbc:weblogic:t3client", props);
```

... der sodann die Verbindung zum DBMS herstellt. Das rückgelieferte Handle Objekt (conn) beinhaltet bei positivem Abschluß der Prozedur alle wichtigen Verbindungsdaten. Dieses Handle muß in Zukunft bei allen JDBC Aufrufen angegeben werden, die sich auf diese Datenbank beziehen.

Das Trennen der Verbindung erfolgt mit den folgenden Zeilen (Modul *public boolean DisconnectDatabase ()*):

```
// close connections to database and the T3 server
conn.close();
t3.disconnect();
conn = null;
```

Es werden die Verbindungen zum DB Server und zum Applikations Server aufgetrennt.

Für die Datenbankabfrage gibt JDBC ein genormtes Interface vor, welches man den jeweils aktuellen Spezifikationen entnehmen kann. Die letzte Spezifikation (Jänner 1997) ist JDBC 1.10 und ist als Postscript File bei [JDBC, 97] elektronisch beziehbar. Das Java Applet für den GDSS Prototypen wurde unter Verwendung von JDBC 1.0 und der Spezifikationsversion 0.70 geschrieben, leichte Änderungen der gezeigten Quellcodes sind also nicht auszuschließen.

Die Abfrage nach den FBIDRs erfolgt mit der Funktion *public boolean QueryDataBase (float fLat, float fLong, float fLength, boolean Cycle1, boolean Cycle2, boolean Cycle3)*, wobei als Parameter die Koordinaten (Breite, Länge) des POI (point of interest), die Länge des Suchquadrates und die gewünschten Zyklen übergeben werden. Nach der Umrechnung der Länge in Grad, die in Kapitel 5.1.4 näher erläutert wurde, erfolgt der Aufbau der SQL Abfrage. Diese Abfrage wurde

ebenfalls bereits in Kapitel 5.1.4 ausführlich beschrieben samt den Problemen, die bei einer Überdeckung des Null Meridians auftreten. Hier wird das Augenmerk ausschließlich auf die Umsetzung der Abfrage auf das JDBC API gelegt.

Um Abfragen mit variablen Parametern durchführen zu können, unterstützt JDBC sogenannte „Prepared Statements“. Man konstruiert die SQL Abfrage normal als String, setzt allerdings an die Stellen, wo später die aktuellen vom Benutzer eingegebenen Abfrage Parameter eingefügt werden sollen, ein „?“<sup>1</sup>. Ein Ausschnitt aus der in Kapitel 5.1.4 behandelten Abfrage könnte etwa so aussehen:

```
"WHERE (GEO_ORBITPT.SAR_ANTENNA_BORESIGHT_LONG>= ? "+  
"AND GEO_ORBITPT.SAR_ANTENNA_BORESIGHT_LONG<= ? "+
```

Das Einsetzen der Parameter in das vorbereitete Statement erfolgt dann mit folgendem Code:

```
PreparedStatement pstmt1 = conn.prepareStatement(sQueryFBIDRs);  
pstmt1.setFloat (1,fWest);  
pstmt1.setFloat (2,fEast);
```

Wobei *fWest* das erste Fragezeichen ersetzen würde, *fEast* das zweite. Man beachte auch die Verwendung des Objektes *conn*, welches die Methode für das Vorbereiten des Statements anbietet und gleichzeitig das verbindende Element zu einer ganz bestimmten Datenbank ist. Trotzdem kann man aber mit herstellerunabhängigen Statements arbeiten - genau das ist einer der großen Vorteile von JDBC.

Nachdem man die Abfrage fertiggestellt hat, kann man sie nun mit folgendem Code ausführen:

```
ResultSet rs; // define a result set for database queries  
rs = pstmt1.executeQuery();
```

*Rs* ist das Ergebnis Objekt der Abfrage. Dieses Objekt enthält eine große Anzahl von Methoden, um die erhaltenen Ergebnisse zu lesen bzw. um Metainformationen des DBMS zu erhalten. Genauer ist der JDBC Spezifikation bzw. der JDBC Klassenbeschreibung zu entnehmen. Für das FBIDR Java Applet wird z.B. die folgende Methode verwendet, um die Daten als Strings zu bekommen<sup>1</sup>:

```
rs.getString("NAVIGATION_SOLUTION_ID")
```

---

<sup>1</sup> der Datentyp String wurde gewählt, um eine einfache Darstellung der Ergebnisse in einem Textfenster zu gestatten

Die Abfrage des Quicklooks wurde nur exemplarisch implementiert, d.h. es wurden zu Testzwecken nur 3 Quicklooks in die Datenbank eingebracht. Das Java Applet ladet statisch einen Quicklook aus der Datenbank. Hier sollte lediglich die Möglichkeit der Speicherung von Bildern in einer relationalen Datenbank illustriert sowie die Abfrage via eines WWW Klienten demonstriert werden.

Die Abfrage verwendet eine Weblogic spezifische Klasse, die sogenannte *TableDataSet* Klasse. Diese Klasse stellt die Ergebnisse einer Abfrage direkt als Tabelle zur Verfügung. Man hat die Möglichkeit mit WHERE oder ORDER BY die Daten zu sortieren bzw. die Änderungen als neue Tabelle zu speichern. Dies bringt u.a. einen Geschwindigkeitsvorteil, da nicht ständig Daten über das Netzwerk zum Klienten gebracht werden müssen. Der entsprechende Code Abschnitt sieht wie folgt aus:

```
// Create a new table data set
td = new TableDataSet(conn, "imagetable");

// td.where ("blobskey="+sBlobsKey); // refine query
td.selectStmt().setQuery("blobskey", sBlobsKey);

// Retrieve the images from the database
td.fetchRecords();

// get the first record of the result set
Record rec = td.getRecord(0);

// visualize image in a result frame
if (!DisplayImage (rec))
    return false;
```

Die letzten Zeilen rufen die Funktion auf, die das Bild schließlich in einem Fenster am Schirm darstellt (siehe auch Kapitel 5.4.2). Die Darstellung ist auch einigermaßen trickreich, da die vorliegende Version 1.01 von Java die Übergabe von Bildern als JPEG Bytearray an den internen Decoder nicht direkt unterstützte - lediglich RGB Arrays wurden akzeptiert. Dieses Problem wurde mit der Klasse *ArrayImageSource* umschifft, die im Prinzip der Java VM ein Laden des JPEG Bildes von einer nicht existenten URL vorgaukelt und dann dieses Bild direkt an den JPEG Decoder von Java übergibt.

Dies waren die wichtigsten Komponenten des auf 1244 Zeilen angewachsenen Java Hauptprogrammes. Der Rest beschäftigt sich mit dem GUI. Dies ist jedoch ein eigenes großes Gebiet und nicht Thema dieses Projektes. Interessierte Leser können sich in der Literatur [Arnold, 96], [Back, 96], [Vanhel, 96] bzw. direkt am WWW [Java, 97] ausführlich darüber informieren. Das nächste Unterkapitel behandelt die

Bedienung und die Funktionalität des Java GDSS Prototyp Klienten.

#### 5.4.2 Bedienung und Funktionalität

Das Java Applet ist über jeden Java fähigen Browser (z.B. Netscape, MS Explorer, Hot Java) unter der URL:

<[http://pds.icg.tu-graz.ac.at/~oracle/ora\\_java/oratest.html](http://pds.icg.tu-graz.ac.at/~oracle/ora_java/oratest.html)>

aufzurufen, und erscheint nach der Initialisierung wie es in Abbildung zu sehen ist.

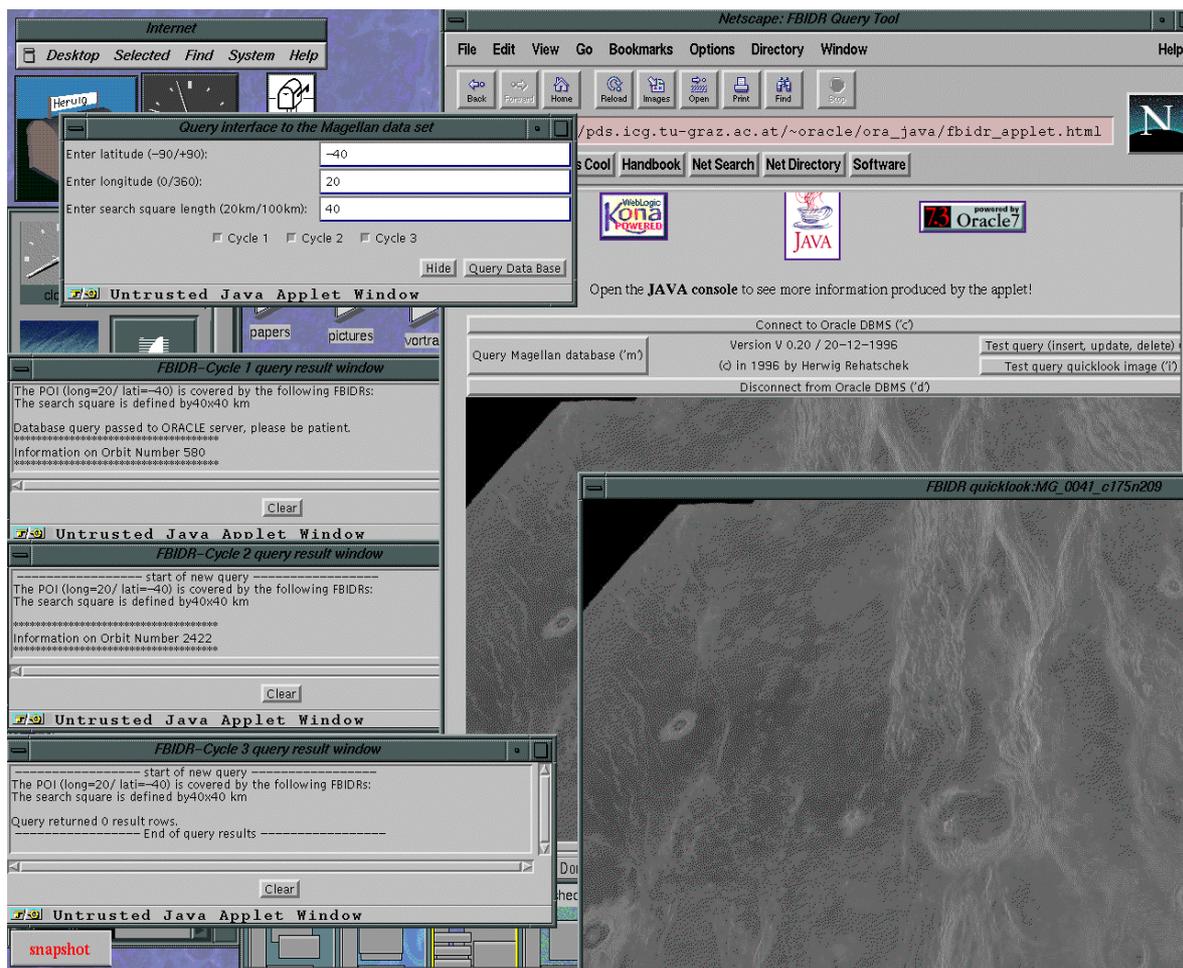
Im rechten oberen Teil sieht man das Hauptfenster des Applets, welches die Bedienelemente für die Datenbankverbindung/-trennung, die Abfrage des Magellan Datensatzes sowie die Buttons für eine Testabfrage und die Abfrage eines Quicklooks birgt. Die Testabfrage demonstriert anhand einer Testtabelle das einfache Einfügen, Updaten und Löschen von Datensätzen mit JDBC und hat nur demonstrativen Charakter.

Das Laden von Quicklooks erfolgt sowohl in einem getrennten Fenster als auch im Hauptfenster des Applets, wie man im rechten unteren Teil von Abbildung 5.18 erkennen kann. Es wurden beide Lösungen implementiert, da letzten Endes nur der kommerzielle Kunde entscheiden kann, welche Darstellung er bevorzugt.

Die Darstellung im Applet Hauptfenster ist zwar kompakter (man erstickt nicht in einer Fensterflut), jedoch ist die Darstellungsgröße begrenzt. Die Größe des Fensters kann man entsprechend dem geladenen Bild anpassen, allerdings kann durch eine Vielzahl an Fenstern schnell die Übersicht verlorengehen, vor allem wenn sie sehr viel Platz beanspruchen.

Links oben sieht man nun das Abfrage Formular für die Magellan FBIDR Daten. Nach Eingabe der Koordinaten des POI (point of interest) und der Größe des Suchquadrates in km erfolgt als letztes noch die Auswahl der gewünschten Zyklen per Checkboxes im unteren Teil des Fensters.

Die Ergebnisse der Abfrage werden - entsprechend den drei Zyklen - in drei verschiedenen Fenstern ausgegeben, welche in Abbildung links unten zu finden sind. Das aktuelle Abfrage Beispiel (POI: -40 Grad Breite, 20 Grad Länge, 40 km Suchquadrat Länge, alle 3 Zyklen) hat etwa je einen Orbit vom 1. (nämlich Orbit 580) und 2. Zyklus (Orbit 2422) und keinen vom 3. Zyklus zurück geliefert.



**Abbildung 5.18:** GUI des GDSS Java Prototypen nach der Initialisierung unter Verwendung von Netscape 2.2. Rechts das Hauptfenster des Applets, links das Abfrage Fenster und die Ergebnis Fenster aufgeteilt nach den 3 Zyklen. Siehe Text für nähere Erläuterungen.

### 5.4.3 Versuchsszenario

Das FBIDR-Abfrage-Werkzeug ist im Prinzip an keine bestimmte Architektur bzw. an die Verwendung einer bestimmten Netzwerktechnologie gebunden. D.h. es funktioniert auch über eine Standard Internet (TCP/IP) Verbindung. Will man jedoch Interaktivität garantieren, und dies speziell bei einer hohen Benutzeranzahl die womöglich alle Bilddaten laden, so muß man auch für eine entsprechende Netzwerkarchitektur sorgen. Somit wurde das FBIDR Abfrage Tool über das in Abbildung 5.19 dargestellte ATM Versuchsnetzwerk betrieben.

Da die derzeitige Ausbaustufe des FBIDR Abfrage Tools noch kein Laden von Originaldaten unterstützt, konnte die zur Verfügung stehende Bandbreite bei weitem nicht vollständig ausgenutzt werden. Dieser Versuch stellte lediglich die Funktionalität des FBIDR Abfrage Tools über ein ATM Netzwerk sicher. Richtig

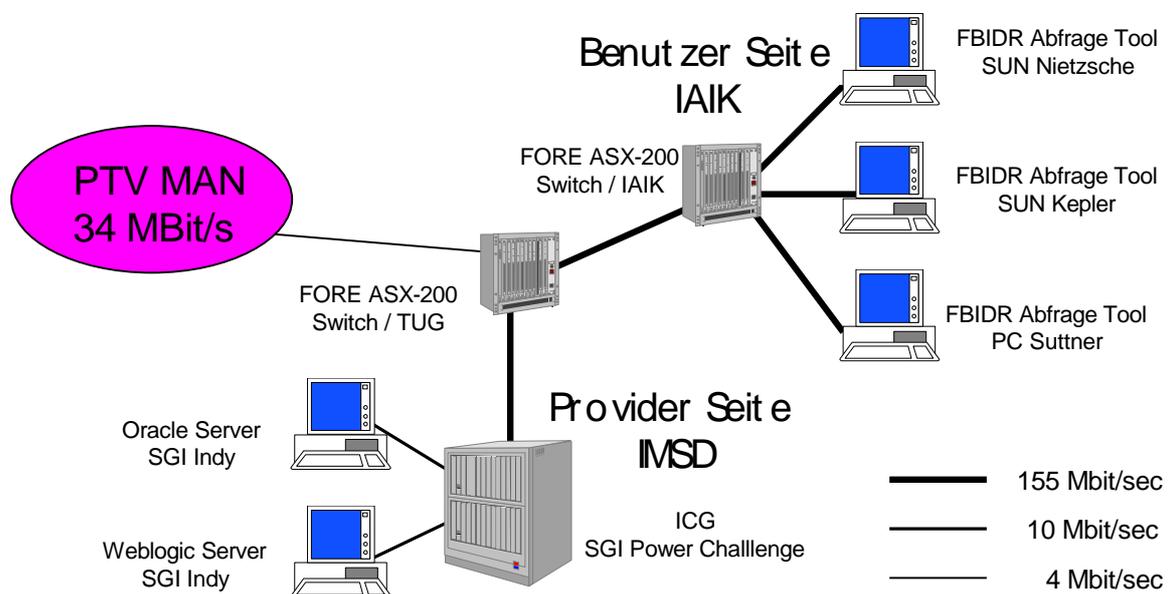
gefordert wird das ATM Netzwerk aber erst, wenn das Laden von Original Daten<sup>1</sup> unterstützt wird.

#### 5.4.4 Probleme und Erfahrungen

Die gesamte Entwicklung des Java Applets ermöglichte uns einen guten Einblick in die Java Programmierung sowie in die Erstellung von komfortablen GUIs. Als besonders nützlich erwies sich das COSMO Produkt von SGI [SGI, 97], das eine integrierte Java Entwicklungsumgebung samt symbolischen Debugger zur Verfügung stellt. Vor allem der Debugger ist äußerst hilfreich bei der Fehlersuche, die ja sonst alles andere als trivial wäre.

Probleme ergaben sich hauptsächlich bei der Programmierung des GUIs sowie der Übergabe von einem JPEG Bytestream an den Java JPEG Decoder. Ersteres Problem wurde durch Studieren von zahlreicher Literatur sowie des Online Java Tutorials am Sun Java Server [Java, 97] behoben.

Einer besonderen Erwähnung bedürfen auch die zahlreichen Java Newsgroups, die in Tabelle 5.10 aufgeschlüsselt sind. Dort kann man zu vielen Fragestellungen und Themen Antworten finden, da Java Programmierer auf der ganzen Welt dort ihre Beiträge liefern.



**Abbildung 5.19:** GDSS Versuchsszenario. Der FBIDR Prototyp wurde auf dem lokalen ATM Pilotnetzwerk getestet. Das ICG fungierte hierbei als zentraler Server, das IAIK als Benutzer Seite. Der FORE ASX-200 Switch kann als lokaler Server im einfachsten Sinne gesehen werden.

<sup>1</sup> die sich bei Magellan z.B. im Bereich von 56 - 95 MB bewegen

Name	Newsgroup	Zweck
Advocacy	comp.lang.java.advocacy	Diskussionsforum der Java Befürworter
Announce	comp.lang.java.announce	Neueste Entwicklungen auf dem Java Sektor
API	comp.lang.java.api	Fragen zu den Java APIs
Miscellaneous	comp.lang.java.misc	Allgemeine Fragen zu Java
Programmer	comp.lang.java.programmer	Diskussionsforum der Java Programmierer
Security	comp.lang.java.security	Fragen zu Java Sicherheitsaspekten
Setup	comp.lang.java.setup	Fragen zur JDK Installation
Technical	comp.lang.java.tech	Technische Fragen zu Java

**Tabelle 5.10:** *Java Newsgroups im Usenet.* Die Tabelle gibt eine Übersicht über alle derzeit existierenden sich mit direkt mit Java beschäftigenden Newsgroups sowie die dort prinzipiell behandelten Themen .

Das Laden eines JPEGs als Bytestream aus einer Datenbank heraus funktionierte tadellos, jedoch die Übergabe an den Java JPEG Decoder bereitete Probleme. Dies wurde mit einer eigenen Klasse bewerkstelligt (ArrayImageSource), deren Grundgerüst von Ron Daniels Jr<sup>1</sup> stammt. Dies ist eines der vielen Beispiele über den Erhalt von Hilfe aus der ganzen Welt unter Verwendung der Erfindung unseres Jahrhunderts, des Internets.

Die Suchquadrate müssen auf eine maximale Größe von 100 km beschränkt bleiben, da sonst eine Abfrage anstatt weniger Minuten plötzlich etlichen Stunden und mehr in Anspruch nehmen könnte. Dies ist auf die bereits in Kapitel 5.1.1 erwähnte schlechte Speichermöglichkeit von räumlichen Daten zurückzuführen. Die Suchzeit steigt mit dem linearen Anstieg des Suchrechteckes leider exponentiell. Dies wirkt sich besonders durch die Größe der *orbit points* Tabelle (siehe Kapitel 5.1.2) dramatisch aus.

Die Beschränkung auf 100 km beeinträchtigt die Funktionsweise des GDSS Prototypen jedoch nicht, da FBIDR ohnehin im Schnitt nur 350 Pixel breit sind, was bei der Auflösung von 75m/Pixel 26,25 km entspricht. Mit dem Suchgebiet von 100 km kann man also bis zu 4 nebeneinander liegende FBIDR Streifen mit einer Abfrage bekommen. Dies entspricht vollkommen dem Anforderungsprofil, wo ja eigentlich nur der FBIDR gesucht ist, der wenigstens einen Punkt innerhalb der

<sup>1</sup> beschäftigt bei der US Regierung (Los Alamos National Laboratory, Los Alamos, NM, USA 87545); E-mail: rdaniel@lanl.gov, WWW: <http://www.acl.lanl.gov/~rdaniel/>

Suchregion hat.

## 5.5 Zusammenfassung und Schlußfolgerungen

Das Kapitel 5 beschrieb die Implementierung eines GDSS Laborversuchssystems. Beginnend mit der Wahl des geeigneten Datenbankmodells, über die Beschreibung des Magellan FBIDR Metadatensatzes, die Inbetriebnahme des Oracle Servers, den Aufbau einer multitier Umgebung bis hin zur Beschreibung des Java Klient Programmes und seiner Funktionalität.

Die Wahl des relationalen Modells ist mit der Verfügbarkeit eines international gültigen ANSI Standards und der großen Verfügbarkeit von RDBMS zu begründen. Das relationale Modell ist jedoch nicht optimal zur Verwaltung von räumlichen Daten geeignet, was deutliche Performance Probleme aufwirft. Eine Migration zum OO Datenmodell als zukünftige Erweiterung ist dringend nötig, ist aber derzeit wegen Nichtvorhandenseins eines internationalen Standards zum Scheitern verurteilt. Ein OO Modell würde den großen Vorteil einer direkten Implementierbarkeit von räumlichen Datenstrukturen innerhalb des DBMS - derzeit müssen entsprechenden Algorithmen im langsamen Java implementiert werden [Walcher, 96] - mit sich bringen.

Java mit seinen zahlreichen APIs und der Möglichkeit Applets zu programmieren, hat sich aufgrund von Plattformunabhängigkeit, Flexibilität, Netzwerkorientiertheit, Sicherheit und Offenheit zu einem Industriestandard für Internet Applikationen der 90er Jahre etabliert. Die verwendete Version 1.01 weist aber noch bedenkliche Sicherheitslecks auf. Mit einer Behebung ist aber in der neuen Version 1.1 zu rechnen. Die Hauptvorteile von Java, aufgrund derer es für GDSS gewählt wurde, sind Plattformunabhängigkeit, Offenheit und automatisches Update.

JDBC (Java Data Base Connectivity) ist ein Java API, welches die Anbindung beliebiger relationaler DBMS erlaubt, sofern der jeweilige Anbieter einen „native Driver“ zur Verfügung stellt. Für GDSS wurde allerdings eine multitier Lösung unter Verwendung der Weblogic Inc. Software implementiert. Diese bietet den zusätzlichen Vorteil, daß mehrere Datenbanken an einem Application Server angebunden werden können und das Klienten Programm über keine herstellerspezifischen Bibliotheken verfügen muß. JDBC ist seit Dezember 1996 in Version 1.10 bei Sun verfügbar [JDBC, 97].

Das vollständig in Java implementierte Klienten Programm ist plattformunabhängig auf jeder Architektur lauffähig, die einen Java kompatiblen Browser anbietet. Das Programm gestattet die Abfrage von FBIDR innerhalb eines Suchgebietes, sowie die Testabfrage eines Quicklooks. Das Suchgebiet des GDSS Prototypen ist auf 100 km begrenzt, was allerdings das Anforderungsprofil des Prototypen auf jeden Fall erfüllt. Für eine Erweiterung des Suchgebietes >100 km ist entweder die Migration zu einem OO Modell oder die Anschaffung zusätzlicher Hardware für den Oracle

Server von Nöten, um die DBMS Aktionen zu beschleunigen. Letztere Lösung wird aber auch sehr schnell am Plafond anstehen, da die Suchzeit exponentiell mit der linearen Vergrößerung des Suchgebietes steigt. Solange jedoch kein Standard für die OO Abfrage von Datenbanken vorliegt, muß man mit der vorliegenden Lösung wohl oder übel Vorliebe nehmen, will man nicht das Dasein einer Insellösung riskieren.

## 5.6 Literatur und Referenzen

- [Applet, 96] A Collection of Increasingly Hostile Applets, Homepage, <URL: <http://www.math.gatech.edu/~mladue/HostileApplets.html>>
- [Arnold, 96] Arnold K., Gasling J. „Java die Programmiersprache“. Addison Wesley, ISBN: 3-8273-1034-2., 1996
- [Back, 96] Back S., Beier S., Bergius K., Majorczyk P. „Professionelle Java Programmierung - Leitfaden für Entwickler“. Thomson Verlag, ISBN: 3-8266-0249-8, 1996
- [Codd, 70] Codd E.F. „A relational model of data for large shared data bases“. Comm. ACM, 13, 377-387, 1970
- [Das, 92] Das S. „Deductive databases and logic programming“. ISBN 0-201-56897-7, Addison-Wesley, 1992
- [Date, 93] Date Ch. „A guide to the SQL standard : a user's guide to the standard relational language SQL / C“. ISBN 0-201-55822-X, Addison-Wesley, 1993
- [Delobel, 95] Delobel C. „Databases : from relational to object-oriented systems“. ISBN 1-850-32124-8, International Thomson Publ., 1995
- [Java, 97] Java Soft, Homepage, Jän. 1997, Homepage, <URL: <http://java.sun.com/>>
- [JDBC, 97] Java Data Base Connectivity (JDBC), Homepage, Jän. 97, <URL: <http://splash.javasoft.com/jdbc/>>
- [JDBCDr, 97] Java Data Base Connectivity (JDBC) implementierte Treiber, Jän. 97, <URL: <http://splash.javasoft.com/jdbc/jdbc.drivers.html>>
- [Kim, 90] Kim W. „Introduction to object-oriented databases“. ISBN 0-262-11124-1, MIT Press, 1990
- [LaDue, 96] LaDue M. D. „Java Insecurity“, white paper, 1996 <URL: [http://www.math.gatech.edu/~mladue/Java\\_insecurity.html](http://www.math.gatech.edu/~mladue/Java_insecurity.html)>
- [Mannila, 92] Mannila H., The design of relational databases. ISBN 0-201-56523-4, Addison-Wesley, 1992
- [Maurer1, 96] Maurer H. „Hyperwave - The Next Generation Web Solution“. ISBN 0-201-40346-3, Addison Wesley, 1996
- [Maurer2, 96] Maurer H., Scherbakov N. „Multimedia Authoring for Presentation and Education- The Official Guide to HM-Card“. ISBN 3-89319-928-4, Addison-Wesley, Bonn 1996.
- [McGraw, 96] McGraw G., Felten E. „Java Security: Hostile Applets, Holes, & Antidotes“. ISBN: 0-471-17842-X, John Wiley and Sons, 1996
- [Netscape, 96] Netscape Corporation, WWW Server Anbieter, Homepage, Dez. 1996, <URL: <http://www.netscape.com/>>
- [Olle, 80] Olle T. „The CODASYL approach to Data Base Management“. John Wiley Sons Publ. Company, 1980
- [SGI, 97] Silicon Graphics Inc. (SGI), Homepage, Jän. 97, <URL: <http://www.sgi.com>>
- [Vanhel, 96] Vanhelsuwé L., Phillips I., Hsu G., Sankar K., Ries E., Rohaly T., Zukowski J., Mastering Java. Sybex, ISBN: 0-7821-1935-2, 1996
- [SQL3, 96] SQL Standards, homepage, Dez. 1996, <URL: [http://www.jcc.com/sql\\_std.html](http://www.jcc.com/sql_std.html)>
- [Walcher, 96] Walcher W. „Design Aspects of Information Systems for Planetary Image Data“. Dissertation am Institut für Maschinelles Sehen und Darstellen, Technische Universität Graz, Graz 1996
- [Weblogic, 96] Weblogic Inc., JDBC driver provider for Oracle, Sybase and MS SQL Server, Dez. 96, Homepage, <URL: <http://www.weblogic.com>>
- [Weblogic2, 96] Weblogic jdbcKona/T3 Architektur und Client Implementierungsbeschreibung, Dz. 96, <URL: [http://www.weblogic.com/classdocs20/API\\_jdbct3.html](http://www.weblogic.com/classdocs20/API_jdbct3.html)>



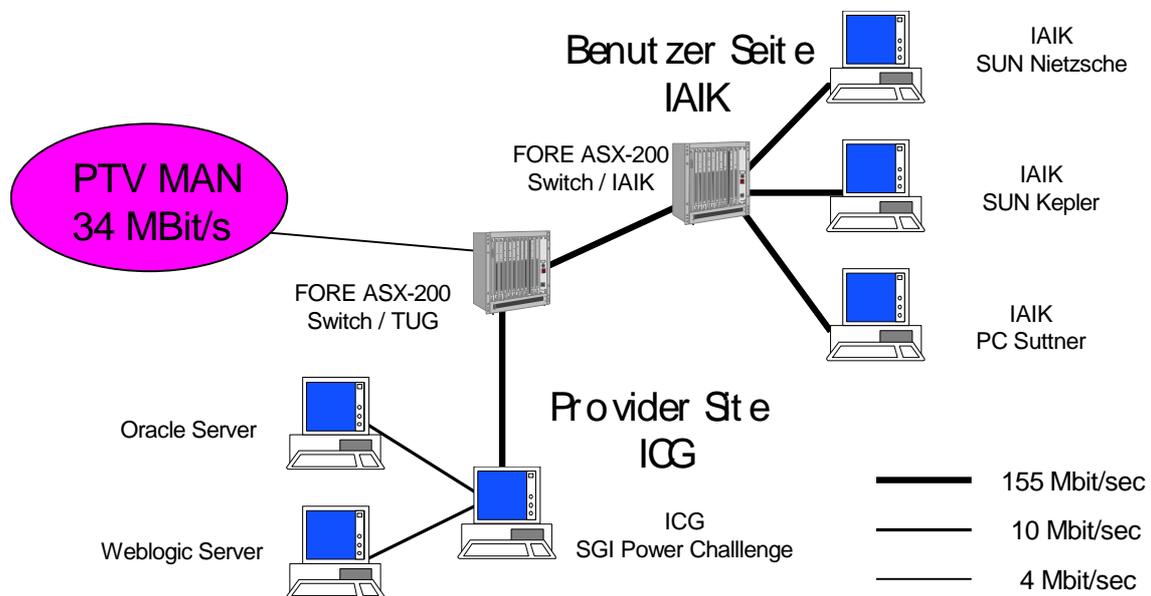
## 6. Kapitel: Ergebnisse

„Wissenschaftler ist jemand,  
dessen Einsichten größer sind  
als seine Wirkungsmöglichkeiten.  
Gegenteil: Politiker“  
(Helmar Nahr)

Dieser Abschnitt berichtet über Ergebnisse des Projektes, deren Grundlagen in Kapitel 4 und 5 ausführlicher beschrieben wurden. Weiters werden Performance Messungen und deren Resultate am ATM Netzwerk sowie zwei ATM Pilotversuche und die daraus resultierenden Schlußfolgerungen beschrieben, die im Rahmen eines EU-Projektes durchgeführt wurden.

### 6.1 Aufbau des ATM Netzwerk Backbones

Ein ATM-Pilot-Netzwerk zwischen dem Institut für Allgemeine Informations- und Kommunikationstechnologie (IAIK) wurde aufgebaut. Am ICG wurde eine SGI Power Challenge (ICG) mit einer VMA-200 ATM Karte von FORE Systems ausgestattet, um eine ATM Punkt-zu-Punkt-Verbindung zum IAIK schalten zu können. Per Glasfaserkabel ist diese direkt an den bereits installierten ASX-200 ATM Switch von FORE Systems im EDV-Zentrum der TU-Graz und damit an das IAIK angeschlossen. Somit war ein erstes Pilot-Netzwerk auf Basis von OC-3c mit



**Abbildung 6.1:** ATM Pilotversuchsnetz zwischen zwei Instituten der TU Graz. Die SGI Power Challenge ist direkt mit dem ATM Switch des EDV Zentrums verbunden, von wo es eine Verbindung zum ATM Switch des IAIK gibt. Dort sind zahlreiche Workstations und PCs angeschlossen.

155 Mbit/sec geschaffen (Abbildung 6.1).

Am IAIK waren eine Reihe von Workstations und PCs (siehe auch Kapitel 6.2) angeschlossen. Mit der SGI Power Challenge waren der Oracle Web Server und der Weblogic Server verbunden, die einen Prototypen des in Kapitel 3 beschriebenen zentralen Servers realisieren.

Dieses Basisnetz war Ausgangspunkt für Pilotversuche, die in den folgenden Abschnitten beschrieben werden.

## 6.2 Performance Messungen am ATM Netzwerk

Es wurden am lokalen ATM Pilot Netzwerk - wie in Abbildung 6.1 dargestellt bzw. in Kapitel 5 technisch veranschaulicht - Performance Messungen durchgeführt. Hierzu wurde das Netperf Tool Version 2.1 [Netperf, 96] von HP verwendet. Es wurde der Durchsatz von 2 Protokollen, nämlich FORE IP (FIP) und Classical IP (CIP) getestet. Als Kontrollprotokoll wurde einerseits TCP, andererseits UDP getestet. Die verwendete Hardware Ausstattung ist Tabelle 6.11 zu entnehmen.

Plattform	Betriebs System	ATM Karte	Unterstützte Protokolle	Max. Übertragungsgeschw.
SUN Sparc-10/512	Sun OS 4.1.3	SBA-200	CIP, FIP, LANE 1.0	155 Mbit/sec (OC-3)
SUN ULTRA-1	Solaris 2.5	SBA-200	CIP, FIP, LANE 1.0	155 Mbit/sec (OC-3)
PC Pentium-100	Windows NT 3.51	PCA-200PC	FIP, LANE 1.0	155 Mbit/sec (OC-3)
SGI Power Challenge L	IRIX 6.2	VMA-200	CIP, FIP, LANE 1.0	155 Mbit/sec (OC-3)

**Tabelle 6.11:** Übersicht über die für die Performance Messungen verwendete Hardware. Sämtliche ATM Karten sind von FORE Systems Inc.

Um die erhaltenen TCP/IP Ergebnisse geeignet zu interpretieren, wurde der theoretisch maximal erreichbare Durchsatz berechnet. Hierbei wurde nur Rücksicht auf den vom Protokoll verursachten Overhead gelegt, nicht aber auf Geschwindigkeitseinbußen, welche etwa durch Betriebssystem-Overhead oder Interprozeßkommunikation verursacht sind.

Das Netzwerk wurde unter optimalen Bedingungen - also im nicht geladenen Zustand - getestet. An den Tests waren jeweils nur zwei Maschinen gleichzeitig beteiligt, an den beiden Switches trat somit keine „Congestion“ auf.

Der Protokoll-Overhead setzt sich aus drei Stufen zusammen, aus dem von der physischen Schicht, von der ATM Schicht und von der AAL (ATM Adaptation Layer) Schicht verursachten. Diese werden in den folgenden Abschnitten beschrieben.

### 6.2.1 Overhead der physischen Schicht

Für die physikalische Übertragung wurden SONET STS-3c/OC-3c Übertragungsrahmen verwendet. Generell ist der Overhead bei SONET für OC-Nc<sup>1</sup> festgelegt durch Gleichung 6.1 [Sadiku, 95]. Wobei für SONET der „SectionOverhead“ und der „PathOverhead“ = 9 Byte sind, der LineOverhead = 18 Byte.

$$\text{Overhead}_{OC-Nc} = N * \text{SectionOverhead} + 1 * \text{PathOverhead} + N * \text{LineOverhead}$$

**Gleichung 6.1:** SONET Overhead

Dies ergibt für OC-3c (N=3) 90 Byte Overhead für SONET OC-3c<sup>2</sup>. Somit läßt sich die theoretisch nutzbare maximale Bandbreite nach der physikalischen Schicht gemäß Gleichung 6.2 berechnen, zieht man den FramePayload von SONET gleich 2430 Byte hinzu.

$$BW_{phy} = \frac{\text{FramePayload}}{\text{FrameLength}} * \text{Bitrate} = \frac{2430 - 90}{2430} * 155,520 = 149,760 \text{ Mbit / sec}$$

**Gleichung 6.2:** maximale Bandbreite nach der physikalischen Schicht

Die Bandbreite  $BW_{phy}$  nach Abzug des Overheads der physikalischen Schicht ist demnach 149.760 Mbit/sec [Prycker, 94].

### 6.2.2 Overhead der ATM Schicht

Der Overhead der ATM Schicht errechnet sich durch das bei ATM verwendete Zellenformat von 5 Byte Kopf und 48 Byte Nutzlast, was die für ATM typische Gesamtzellenlänge von 53 Byte ergibt. Der durch die ATM Schicht verursachte Overhead berechnet sich demnach gemäß Gleichung 6.3.

$$BW_{atm} = \frac{\text{CellPayload}}{\text{CellLength}} * BW_{phy} = \frac{48}{53} * 149,760 = 135,632 \text{ Mbit / sec}$$

**Gleichung 6.3:** Durch die ATM Schicht verursachte Overhead

---

<sup>1</sup> wobei N für den Testversuch gleich 3 war

<sup>2</sup> Es sei übrigens darauf hingewiesen, daß ATM eigene Übertragungsrahmen verwendet, die sich vom „normalen“ OC-3 dadurch unterscheiden, daß der PathOverhead nur *einmal* für N-faches Multiplexen gespeichert wird! Bei OC-3 würde der PathOverhead demnach N \* PathOverhead sein.

Die Bandbreite  $B_{w_{atm}}$  nach Abzug des Overheads der ATM Schicht beträgt 135,632 Mbit/sec.

### 6.2.3 Overhead der AAL Schicht

Der letzte betrachtete Overhead wird von der AAL Schicht verursacht. Hier wirken sich die Overheads der verwendeten Protokolle (CIP und FIP in Kombination mit TCP) direkt aus. Da es einen Unterschied in bezug auf die MTU (Maximum Transfer Unit) zwischen FIP und CIP gibt, werden die Overheads getrennt berechnet. Für das Verständnis dieser Berechnung sind einigermaßen fundierte Kenntnisse im TCP/IP Protokoll notwendig, die in dieser Arbeit vorausgesetzt werden und worauf aus Platzgründen nicht näher eingegangen werden kann. Für genauere Informationen sei auf die Literatur verwiesen [Arick, 96], [Gleissner, 91].

Sowohl die VMA-200 Karte (passend für SGI Power Challenge) als auch die SBA-200 Karte (passend für SUNs) unterstützen zwei Implementierungen des IP über dasselbe physikalische Interface - nämlich FIP und CIP - als auch LANE 1.0 (ATM LAN Emulation), vergleiche auch Tabelle 6.11. Die PCA-200PC Karte (passend für Windows NT PCs) unterstützt hingegen nur FIP und LANE 1.0.

CIP benutzt Q.2931 zur Signalisierung [Laubach, 94], während FIP das von FORE Systems selbst entwickelte SPAN (Simple Protocol for ATM Network signaling) verwendet. Ferner benutzt CIP IEEE 802.LLC/SNAP (Logical Link Control/SubNetwork Attach Point) für die Paket Kapselung und die Default IP MTU von 9180 Byte mit einer maximalen Nutzlast von 9140 Byte. FIP hingegen verwendet ein Broadcast ARP (Address Resolution Protocol) und keine Kapselung, was in einer MTU von 9188 Byte und einer maximalen Nutzlast von 9148 Byte resultiert [Fore, 95].

Gemäß [Andrikop, 96] und einem TCP/IP Kopf von 20 Byte kann die durch den AAL verursachte Bandbreitenreduktion entsprechend den Gleichungen 6.4, 6.5, 6.6 und 6.7 berechnet werden.

$$BW_{ip}(CIP) = \frac{CIP\_MTU}{CPCS\_PDU} * BW_{atm} = \frac{9180}{9216} * 135,632 = 135,102 \text{ Mbit / sec}$$

**Gleichung 6.4:** Bandbreitenreduktion durch IP Overhead (CIP)

$$BW_{tcp}(CIP) = \frac{CIP\_MTU - TCPHdr - IPHdr}{CIP\_MTU} * BW_{ip} = \frac{9180 - 20 - 20}{9180} * 135,102 = 134,513 \text{ Mbit / sec}$$

**Gleichung 6.5:** Bandbreitenreduktion durch TCP Overhead (CIP)

$$BW_{ip}(FIP) = \frac{FIP\_MTU}{CPCS\_PDU} * BW_{atm} = \frac{9188}{9216} * 135,632 = 135,220 \text{ Mbit / sec}$$

**Gleichung 6.6:** Bandbreitenreduktion durch IP Overhead (FIP)

$$BW_{tcp}(FIP) = \frac{FIP\_MTU - TChdr - IPHdr}{FIP\_MTU} * BW_{ip} = \frac{9188 - 20 - 20}{9188} * 135,220 = 134,631 \text{ Mbit / sec}$$

**Gleichung 6.7:** Bandbreitenreduktion durch TCP Overhead (FIP)

Der CPCS (Common Part Convergence Sublayer) wird im AAL5 benutzt um die vom BSD Sockets API Interface kommenden Daten zu kapseln. Der CPCS weist eine variable Länge zwischen 1 und 65535 Byte auf. Die Implementierung des AAL5 fügt 36 Byte Overhead bei CIP und 28 Byte bei FIP hinzu, was für beide Protokolle eine Gesamtpaketlänge von 9216 Byte (MTU plus Overhead) ergibt [Andrikop, 96].

Die Bandbreite  $BW_{tcp}(CIP)$  nach Abzug des Overheads ist demnach mit 134,513 Mbit/sec zu beziffern, diejenige von FIP,  $BW_{tcp}(FIP)$  mit 134,631 Mbit/sec. Durch die um 8 Byte größere MTU beim FIP Protokoll erhält man einen etwas größeren Durchsatz.

Basierend auf dieser theoretisch verfügbaren Bandbreite wurden nun die Meßergebnisse entsprechend bewertet. Dies wird im nächsten Abschnitt beschrieben.

#### 6.2.4 Meßergebnisse

Die präsentierten Ergebnisse beruhen auf Messungen, die auf einem unbelasteten ATM Netzwerk zwischen der SGI Power Challenge (SPC) im ICG, SUNs und einem PC im IAIK (siehe Abbildung 6.1) durchgeführt wurden. Für TCP/IP wurden fünf Experimente gemäß Tabelle 6.12 durchgeführt.

Das erste Experiment fand zwischen der SPC und einer SUN Ultra (siehe Tabelle 6.11) unter Verwendung von FIP statt. Als Testparameter wurden folgende Werte verwendet: die „Send/Receive Socket Size“ betrug 53248 Byte<sup>1</sup>, die Paketgröße wurde zwischen 520 und 65535 Byte variiert, wobei das Inkrement 12 Byte ausmachte. Die Versuchsdauer betrug jeweils 10 Sekunden. Der höchste damit erzielte Durchsatz war 54,43 Mbit/sec, was rund 40,43% des theoretisch maximal erzielbaren Durchsatzes sind. Das Ergebnis des Versuches ist in Abbildung 6.2 veranschaulicht.

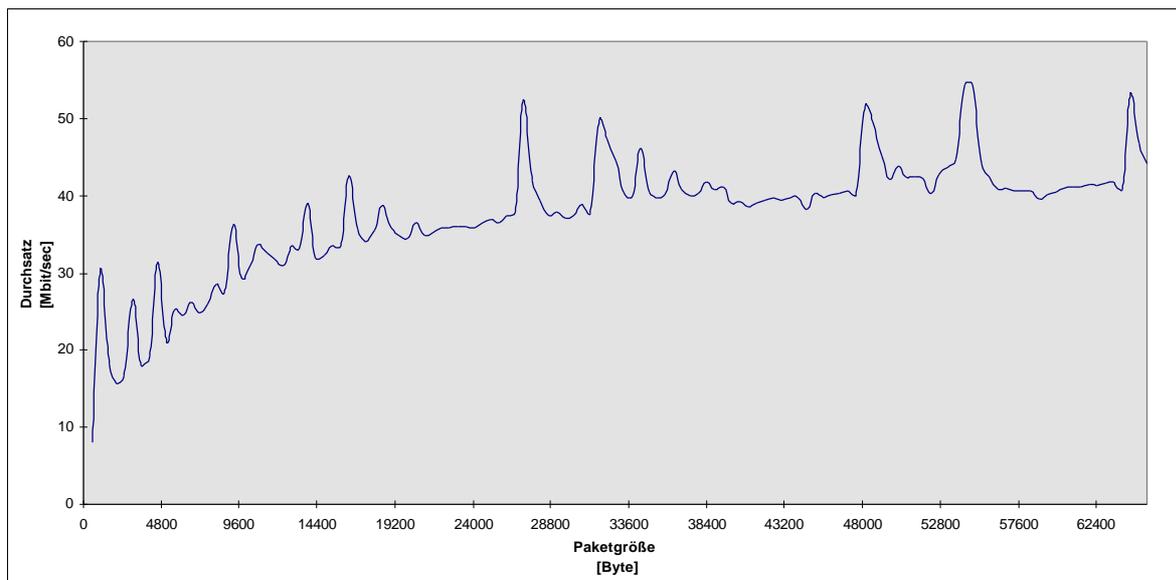
---

<sup>1</sup> Dieser Wert entspricht der maximal erlaubten Send/Receive Socket Puffergröße bei SUNs

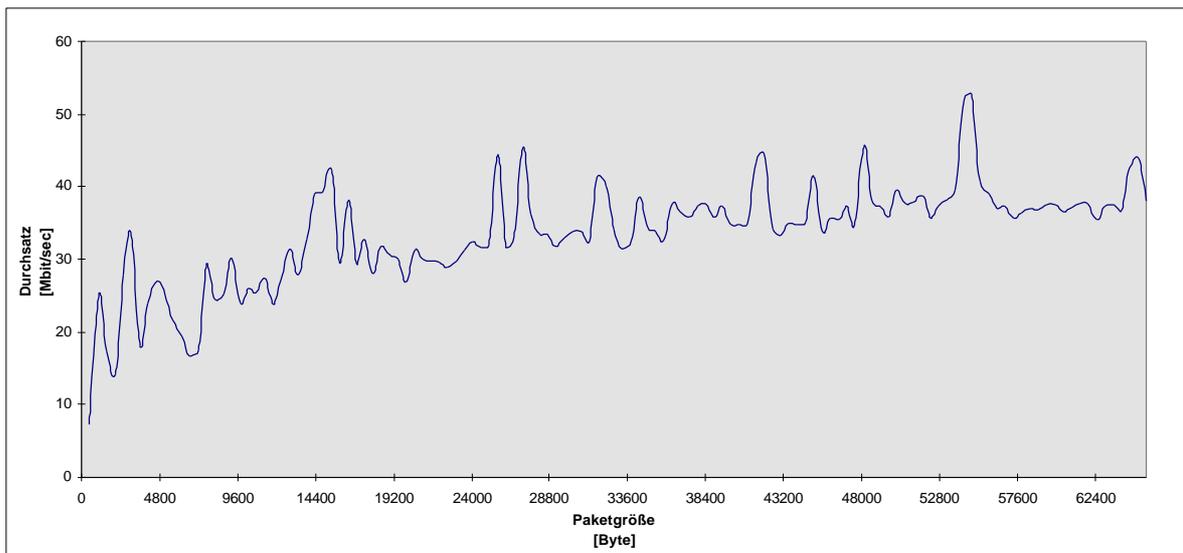
Experiment Nr.	Von / auf	Protokoll	Ergebnis in Diagr.
1	SPC $\Rightarrow$ SUN Ultra	TCP/FIP	6.2
2	SPC $\Rightarrow$ SUN Sparc10	TCP/FIP	6.3
3	SPC $\Rightarrow$ PC	TCP/FIP	6.4
4	SPC $\Rightarrow$ SUN Ultra	TCP/CIP	6.5
5	SPC $\Rightarrow$ SUN Sparc10	TCP/CIP	6.6

**Tabelle 6.12:** Durchgeführte Performance Experimente am ATM Pilot Netzwerk. Insgesamt wurden in 5 Tests der Durchsatz von FIP und CIP über ATM getestet. Mit dem PC konnte mangels Verfügbarkeit eines Treibers CIP nicht getestet werden.

Das zweite Experiment wurde zwischen der SPC und einer SUN Sparc10 durchgeführt, ebenfalls unter der Verwendung von FIP. Die Testparameter sind dieselben wie im ersten Experiment, das Ergebnis ist in Abbildung 6.4 dargestellt. Der höchste Durchsatz betrug 52,75 Mbit/sec. Dies entspricht ca. 39,18% des theoretisch erreichbaren maximalen Durchsatzes.

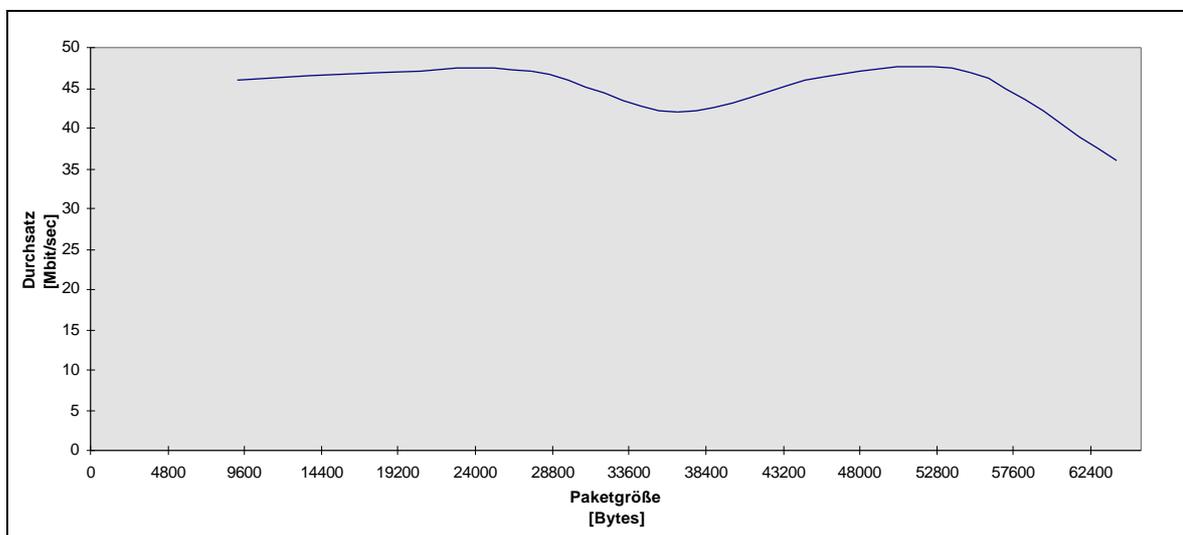


**Abbildung 6.2:** Performance Ergebnis zwischen der SGI PC und einer SUN Ultra unter Verwendung von TCP/FIP. Die Paketgröße wurde in 12 Byte Schritten beginnend mit 520 bis 65535 Byte erhöht, der jeweils damit erzielte Durchsatz ist auf der vertikalen Achse in Mbit/sec eingezeichnet.



**Abbildung 6.3:** Performance Ergebnis zwischen der SGI PC und einer SUN Sparc10 unter Verwendung von TCP/FIP. Die Paketgröße wurde in 12 Byte Schritten beginnend mit 520 bis 65535 Byte erhöht, der jeweils damit erzielte Durchsatz ist auf der vertikalen Achse in Mbit/sec eingezeichnet.

Das dritte Experiment fand zwischen der SPC und einem PC Pentium100 statt. Dieses Experiment wurde zu einem Zeitpunkt durchgeführt, wo noch ein Treiber für FIP am PC existierte. Leider wurden damals andere Parameter verwendet.



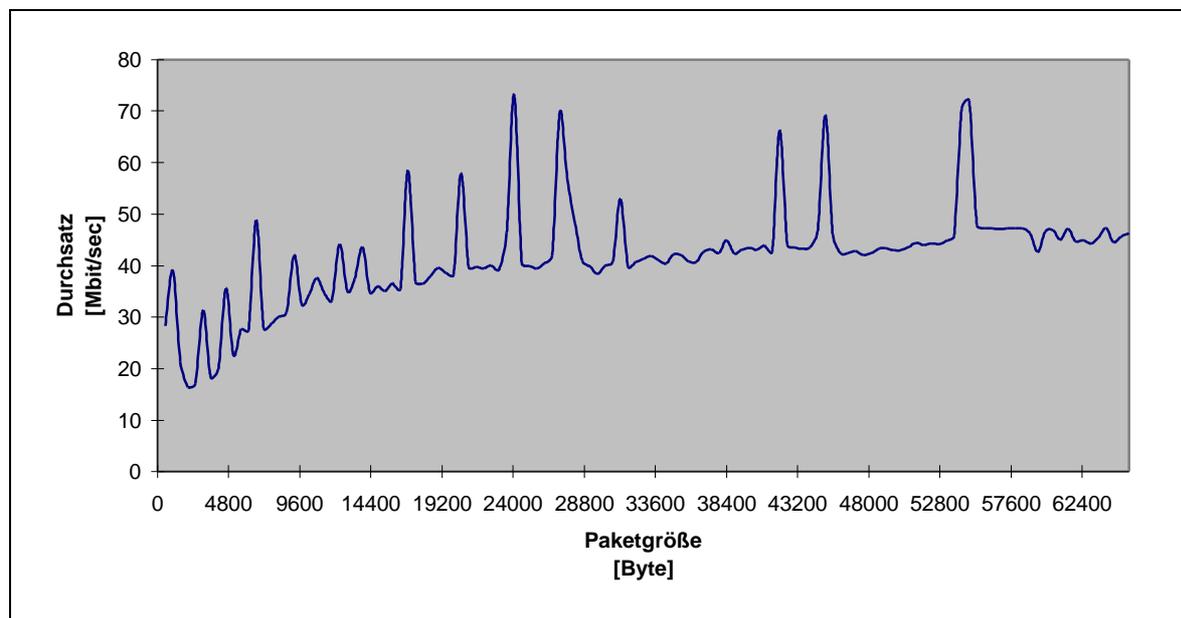
**Abbildung 6.4:** Performance Ergebnis zwischen der SGI PC und einem PC Pentium100 unter Verwendung von TCP/FIP. Die Kurve basiert auf nur 7 aufgenommenen Werten, deren Parameter denen des 1. und 2. Experimentes gleichen. Ab der Version 4 unterstützt FORE Systems jedoch FIP am PC nicht mehr, wodurch das Experiment nicht nochmals durchgeführt werden konnte.

Den Parametern aus Experiment 1 und 2 standen nur insgesamt 7 Ergebnisse zur Verfügung, die dieselben Parameter aufwiesen. Das Experiment konnte auch nicht

nochmals durchgeführt werden, da der neue FORE Systems Treiber Version 4.0 für PC Karten nur mehr LANE 1.0 und weder CIP noch FIP unterstützt. Somit ist die Ergebniskurve (siehe Abbildung 6.4) nicht ganz so aussagekräftig wie die aus den Experimenten 1 und 2. Eine prinzipielle Aussage über den Durchsatz ist aber dennoch möglich. Der maximal erzielte Durchsatz lag bei 48,24 Mbit/sec, was rund 35,68% des theoretisch maximal erzielbaren Wertes sind.

Das vierte Experiment testete den Durchsatz von CIP zwischen der SPC und einer SUN Ultra. Die Testparameter entsprechen 1:1 denen aus den Experimenten 1 und 2, der maximal erzielte Durchsatz war 73,26 Mbit/sec. Das sind rund 54,53% des theoretisch erzielbaren Gesamtdurchsatzes. Das Ergebnis ist in Abbildung 6.5 dargestellt.

Das fünfte und letzte Experiment testet schließlich den Durchsatz von CIP zwischen der SPC und einer SUN Sparc10. Aufgrund von Problemen mit dem FORE Systems Treiber Version 4.0 unter Solaris 2.5 betreffend CIP und LANE 1.0 konnte für das letzte Experiment ebenfalls nur auf einen Datensatz mit insgesamt 7 Werten, die den bereits aus den Experimenten 1-5 bekannten Parametern entsprechen, zurückgegriffen werden. Der maximal erzielte Durchsatz lag bei 46,83 Mbit/sec und entsprach somit 34,9% des theoretisch maximal erzielbaren Wertes (siehe Abbildung 6.6).



**Abbildung 6.5:** Performance Ergebnis zwischen der SGI PC und einer SUN Ultra unter Verwendung von CIP. Die Paketgröße wurde in 12 Byte Schritten beginnend mit 520 bis 65535 Byte erhöht, der jeweils damit erzielte Durchsatz ist auf der vertikalen Achse in Mbit/sec eingezeichnet.

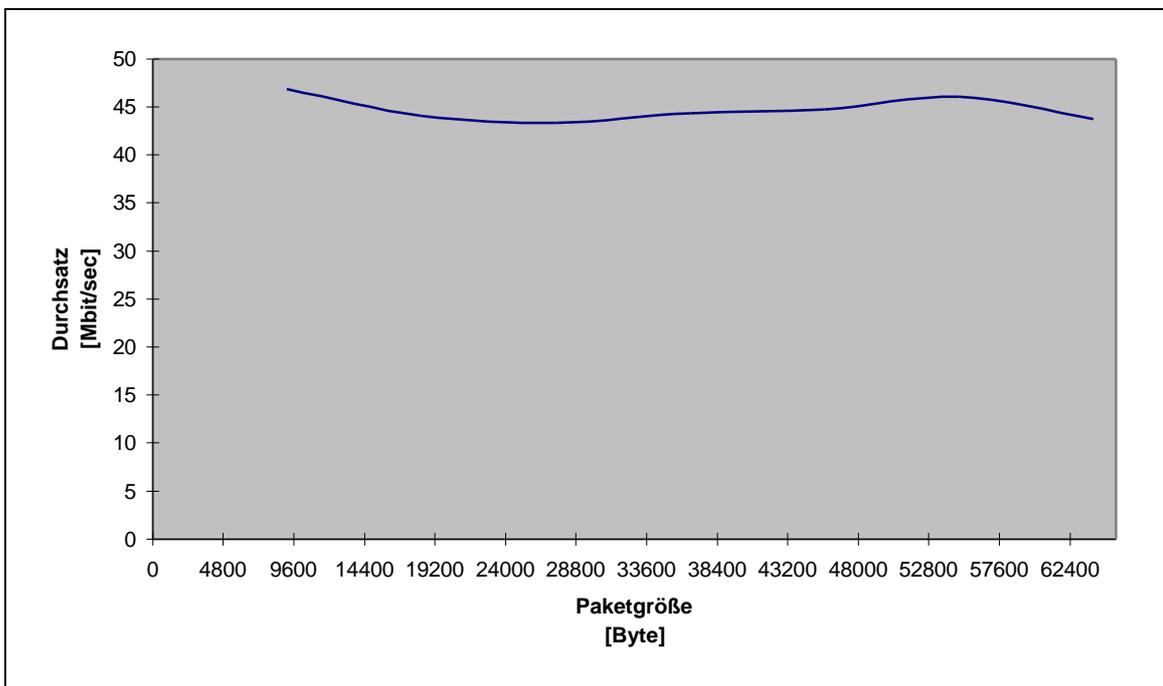
<sup>1</sup> Die Werte stammen aus einem früher durchgeführten Versuch, der allerdings auch die Send/Receive Socket Größe variierte. Somit wurden nur jene Werte herausgegriffen, wo die Socket Größe der des 1. und 2. Experimentes entsprach.

Nicht getestet werden konnte der Durchsatz von CIP zwischen der SPC und dem PC Pentium, da FORE Systems CIP über ATM am PC in keiner Weise unterstützt.

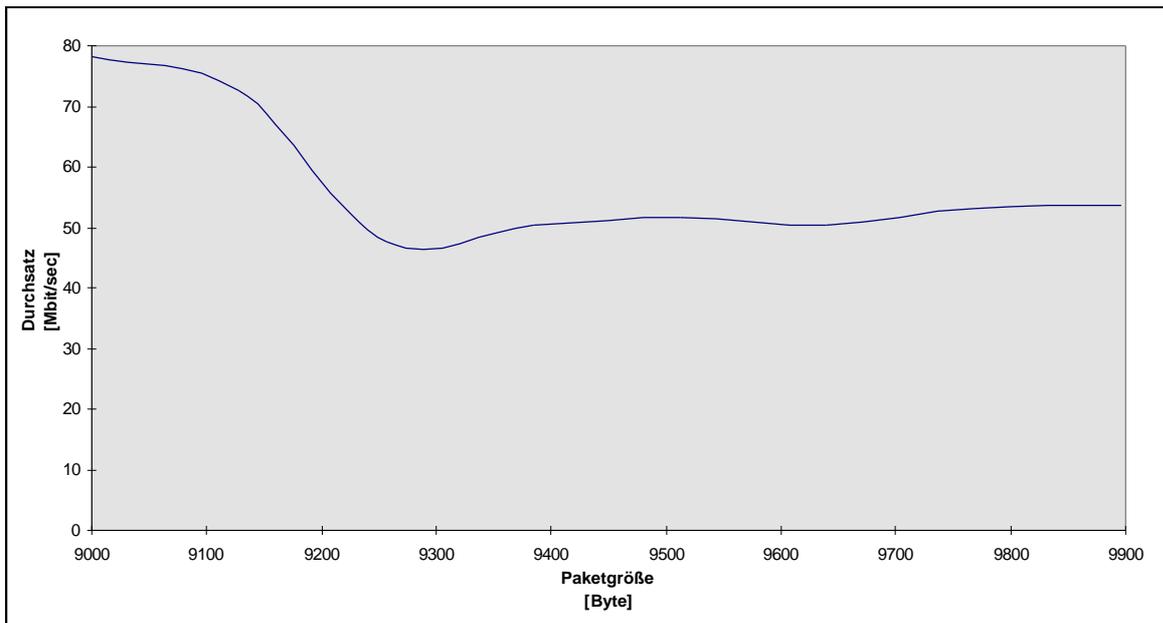
Weiters wurde der Durchsatz zwischen der SPC und den SUNs bzw. dem PCs unter Verwendung von UDP untersucht. Die Experimente 1-5 samt Verweis auf die erhaltenen Ergebnisse sind in Tabelle 6.13 beschrieben.

Experiment Nr.	Von / auf	Protokoll	Ergebnis in Diagramm
1	SPC ⇒ SUN Ultra	UDP/FIP	6.7
2	SPC ⇒ SUN Sparc10	UDP/FIP	6.8
3	SPC ⇒ PC	UDP/FIP	6.9
4	SPC ⇒ SUN Ultra	UDP/CIP	6.10
5	SPC ⇒ SUN Sparc10	UDP/CIP	6.11

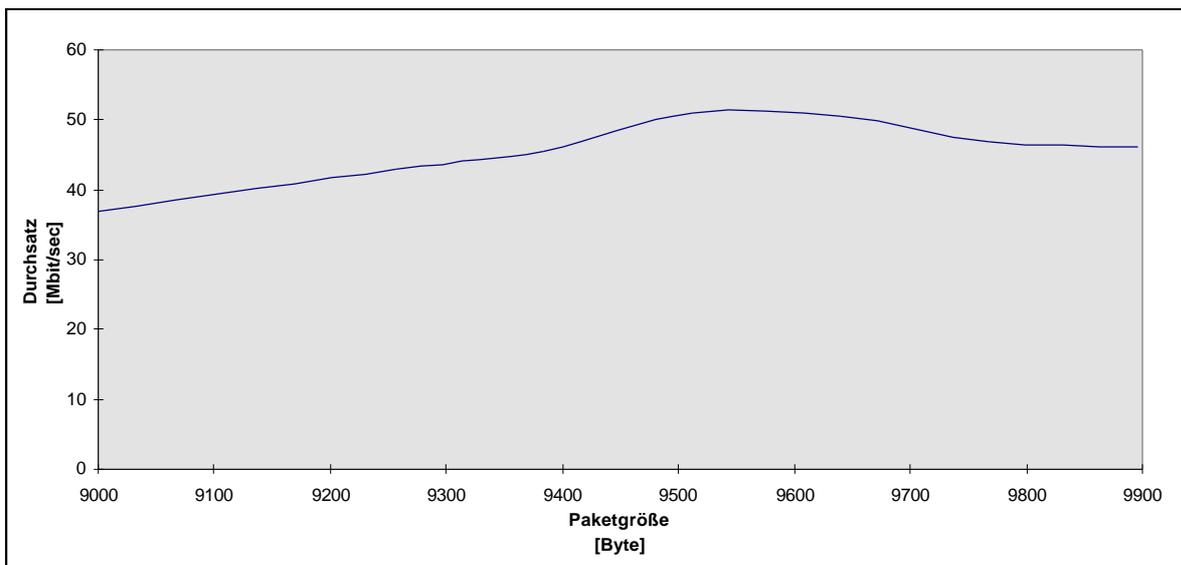
**Tabelle 6.13:** Durchgeführte Performance Experimente am ATM Pilot Netzwerk. Insgesamt wurden in 5 Tests der Durchsatz von FIP und CIP über ATM unter Verwendung von UDP getestet. Mit dem PC konnte mangels Verfügbarkeit eines Treibers CIP nicht getestet werden.



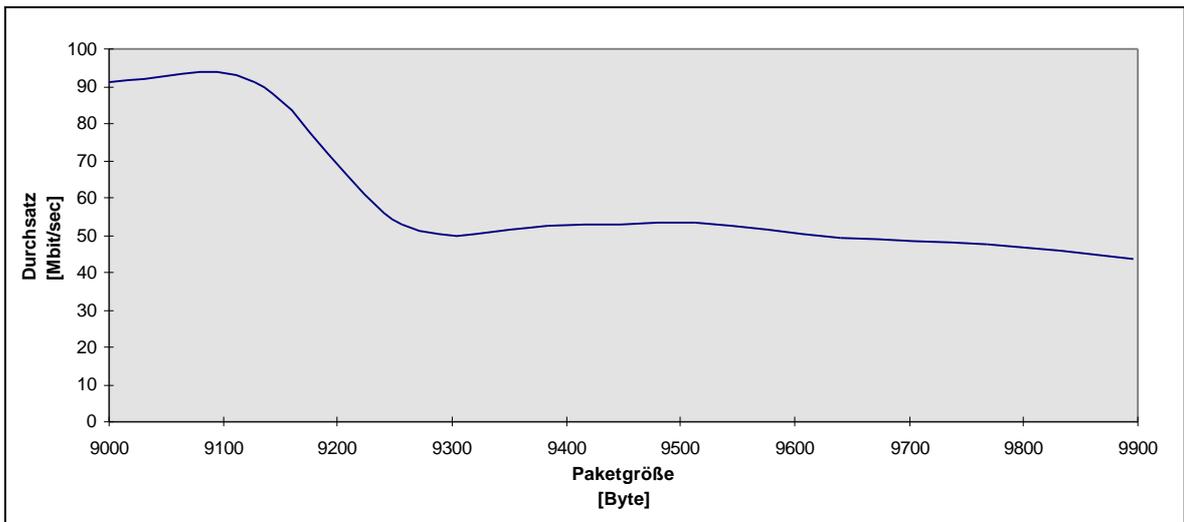
**Abbildung 6.6:** Performance Ergebnis zwischen der SGI PC und einer SUN Sparc10 unter Verwendung von TCP/CIP. Aufgrund von Treiber Schwierigkeiten mußte auf den Datensatz eines alten Versuches zurückgegriffen werden, und es konnten nur 7 Werte berücksichtigt werden. Die Parameter sind gleich wie in den vorangegangenen Versuchen.



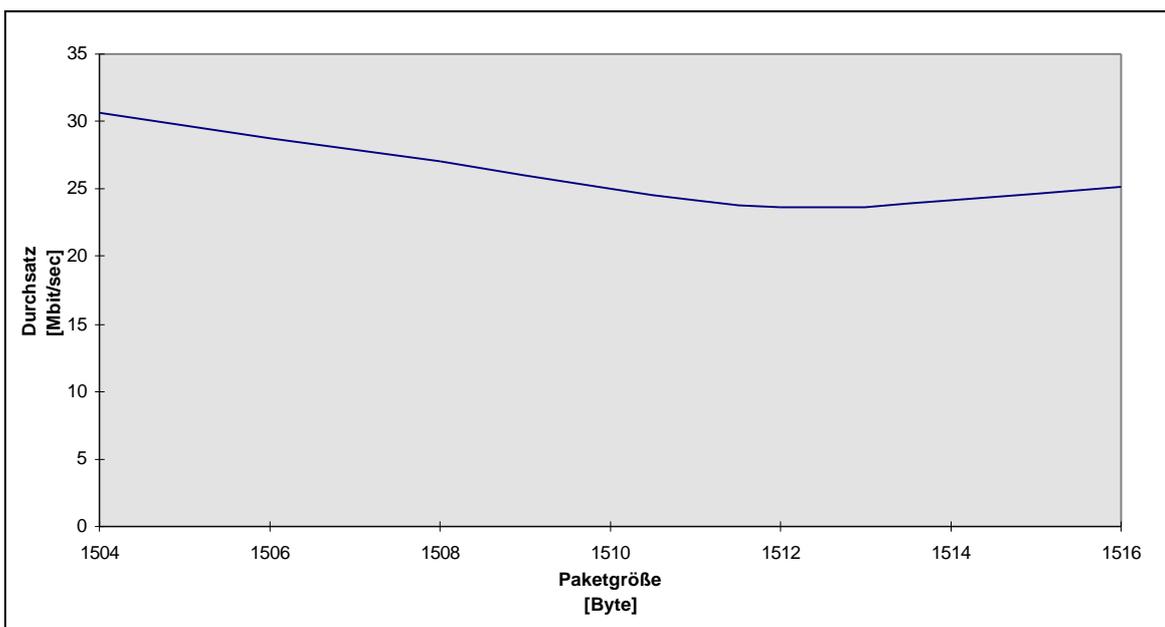
**Abbildung 6.7:** Performance Ergebnis zwischen der SGI PC und einer SUN Ultra unter Verwendung von UDP/FIP. Die Paketgröße wurde zwischen 9128 und 9896 Byte mit einem Inkrement von 128 Byte variiert, die Versuchsdauer betrug 30 Sekunden.



**Abbildung 6.8:** Performance Ergebnis zwischen der SGI PC und einer SUN Sparc10 unter Verwendung von UDP/FIP. Die Paketgröße wurde zwischen 9128 und 9896 Byte mit einem Inkrement von 128 Byte variiert, die Versuchsdauer betrug 30 Sekunden.



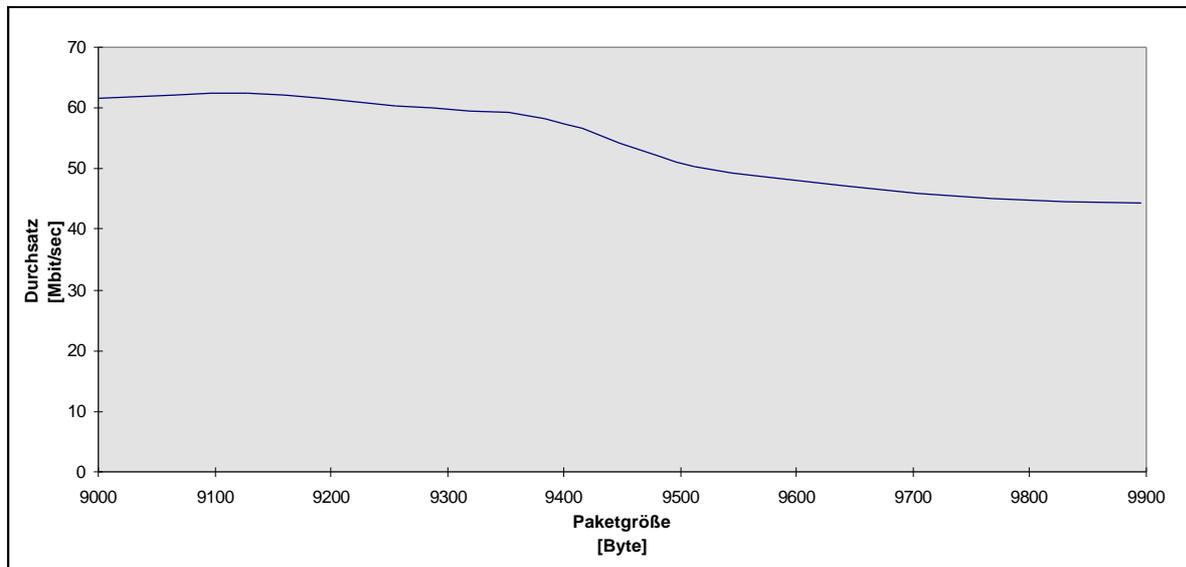
**Abbildung 6.9:** Performance Ergebnis zwischen der SGI PC und einer SUN Ultra unter Verwendung von UDP/CIP. Die Paketgröße wurde zwischen 9128 und 9896 Byte mit einem Inkrement von 128 Byte variiert, die Versuchsdauer betrug 30 Sekunden.



**Abbildung 6.10:** Performance Ergebnis zwischen der SGI PC und eines PC Pentium100 unter Verwendung von UDP/FIP. Die Paketgröße wurde zwischen 1504 und 1516 Byte variiert, wobei das Inkrement 4 Byte betrug. Die Versuchsdauer betrug 30 Sekunden.

Bei den Experimenten 1,2 sowie 4 und 5 wurden die Paketgröße beginnend von 9128 Byte mit einem Inkrement von 128 Byte bis 9896 Byte erhöht. Die Versuchsdauer betrug jeweils 30 Sekunden. Für das Experiment 3 stand nur ein alter Datensatz zur Verfügung, UDP/IP wird ab Treiber Version 4.0 von FORE

Systems für PCs nicht mehr unterstützt.



**Abbildung 6.11:** Performance Ergebnis zwischen der SGI PC und einer SUN Sparc10 unter Verwendung von UDP/CIP. Die Paketgröße wurde zwischen 9128 und 9896 Byte mit einem Inkrement von 128 Byte variiert, die Versuchsdauer betrug 30 Sekunden.

Deswegen konnte kein neuer Versuch mit denselben Parametern erfolgen. Die in Abbildung 6.10 dargestellten Ergebnisse beruhen auf 4 Messungen, die bei Paketgrößen von 1504, 1508, 1512 und 1516 Byte durchgeführt wurden. Die Versuchsdauer betrug 30 Sekunden. Die Ergebnisse sind nicht direkt mit jenen aus den Experimenten 1,2 4 und 5 vergleichbar, sondern sind als Anhaltspunkt zu verstehen.

### 6.2.5 Bewertung der Ergebnisse

Die erzielten Ergebnisse sind samt der erreichten Prozentmarke der theoretisch maximalen Bandbreite in Tabelle 6.14 zusammengefaßt.

Plattform	Max. Durchsatz in Mbit/sec*			
	TCP/FIP	TCP/CIP	UDP/FIP	UDP/CIP
SGI ⇔ SUN Sparc 10	52,75 (39,22%)	46,83 (34,78%)	55,37	62,34
SGI ⇔ SUN Ultra-1	54,43 (40,46%)	73,26 (54,42%)	78,18	91,25
SGI ⇔ PC Pentium 100	48,24 (35,86%)	N/A	30,65	N/A

\* bei den TCP Experimenten wurde zusätzlich noch der Prozentsatz vom max. theoretisch erreichbaren Durchsatz angegeben

**Tabelle 6.14:** Ergebnisse der TCP und UDP Performance Experimente unter Verwendung des HP netperf Tools.

Generell läßt sich bei TCP/(C/F)IP die Tendenz erkennen, daß der Durchsatz mit steigender Paketgröße zunimmt. Spikes sind mehrmals an Vielfachen der MTU zu erkennen. Das ist mit der Tatsache zu erklären, daß bei Paketgrößen gleich der MTU die Aufteilung ohne Overhead erfolgen kann. Bei Paketgrößen gleich  $MTU + 1$  Byte ist der Overhead im letzten Paket maximal, da bei nur 1 Byte Nutzlast der gesamte TCP/IP Kopf sowie  $(MTU-1)$  Bytes leerer Nutzlast geschickt werden.

Für die Realität kann man die Spikes daher ignorieren und den Mittelwert annehmen. Im realen Datenverkehr wird unter normalen Umständen weder eine große Paketzahl mit der Paketgröße gleich  $MTU + 1$  Byte (schlechtester Fall) noch mit der Paketgröße gleich  $MTU$  (optimaler Fall) geschickt. Die Paketgröße des letzten Paketes wird statistisch irgendwo dazwischen variieren. Im Laborversuch wurden allerdings während der gesamten Versuchszeit solche Pakete mit hohem bzw. fast keinem Overhead geschickt, wodurch sich die Spikes ergaben.

Die Magellan Daten, die gemäß dem Systemkonzept aus Kapitel 3 hauptsächlich über das Netz geschickt werden, sind Oberflächen Kacheln („Map-Tiles“) und Quicklooks. Ihr ungefähres Datenvolumen beträgt laut der Netzwerkanforderungsanalyse Kapitel in 4.1 rund 3 kByte für eine Kachel, sowie zwischen 140 und 237 kByte für Quicklooks, abhängig vom erzielten Kompressionsgrad. Das erzielte Ergebnis zeigt einen höheren Datendurchsatz bei größeren Paketgrößen. Somit könnte man also Quicklooks am besten mit der maximalen Paketgröße von  $\sim 50$ kByte übertragen<sup>1</sup>, was pro Quicklook also zwischen 3 und 5 Paketen entspräche. Bei einem maximalen Durchsatz von gemessenen 73,26 Mbit/sec bei TCP/CIP (siehe Tabelle 6.14) käme das einer Zeit von 25,9 msec gleich.

Für die rund 3kByte Kachel empfiehlt sich eine gebündelte Übertragung, da laut Versuchsergebnis bei kleinen Paketgrößen der Durchsatz geringer ist. Diese Tatsache ist konform mit dem im Kapitel 3.1.9 vorgeschlagenen Datenvorhersage Algorithmus. Geht man gemäß Kapitel 4.1 von einer  $4 \times 3$  Kachel Aufteilung am Schirm aus, so kann man diese 12 Kacheln sowie die 12 der nächst besseren und der nächst niedrigeren Auflösung auf einmal schicken, was einem Datenvolumen von  $36 \times 3$ kByte = 108 kByte entspricht. Bei einem gemessenen maximalen Durchsatz von 73,26 Mbit/sec würde die Ladezeit für die 36 Kacheln 11,8 msec betragen.

Negativ ist anzumerken, daß bis Ende 1996 keine Treiber von FORE Systems für CIP bzw. FIP Unterstützung am PC angeboten wurden. CIP wurde generell nie angeboten, FIP verschwand ab Version 4.0. Außerdem gibt es Probleme unter Solaris 2.5 mit der Treiber-Version 4.0 in Zusammenhang mit LANE 1.0 und CIP. Dieses Problem bestand noch Ende 1996 und war Grund, daß etliche Performance Messungen nicht durchgeführt werden konnten. Es bleibt unklar, ob die Ursache

---

<sup>1</sup> weil man die maximale Sende/Empfangs Socketgröße von ca. 53 kByte (SUN) nicht überschreiten kann

hierfür bei der Rechenplattform oder der FORE ATM Hard- und Software zu suchen ist.

Die teilweise schlechten Ergebnisse im Hinblick auf den theoretisch erreichbaren Wert, werden hauptsächlich durch die geringe Verarbeitungsgeschwindigkeit der Maschinen verursacht - hier besteht noch Optimierungsbedarf. Die Hardware wird jedoch laufend verbessert. So hat etwa SGI angekündigt, im Laufe des Jahres 1997 eine eigene ATM Karte auf den Markt zu bringen, welche anstatt des langsamen I/O Busses den schnellen 1.2 GByte/s „Sustained Bus“ verwenden soll. Auf die Performance dieser Karte kann man schon jetzt gespannt sein. Dadurch kann man sich eine weitere Annäherung der derzeit erzielten Ergebnisse an die theoretisch maximal erzielbaren Werte erwarten.

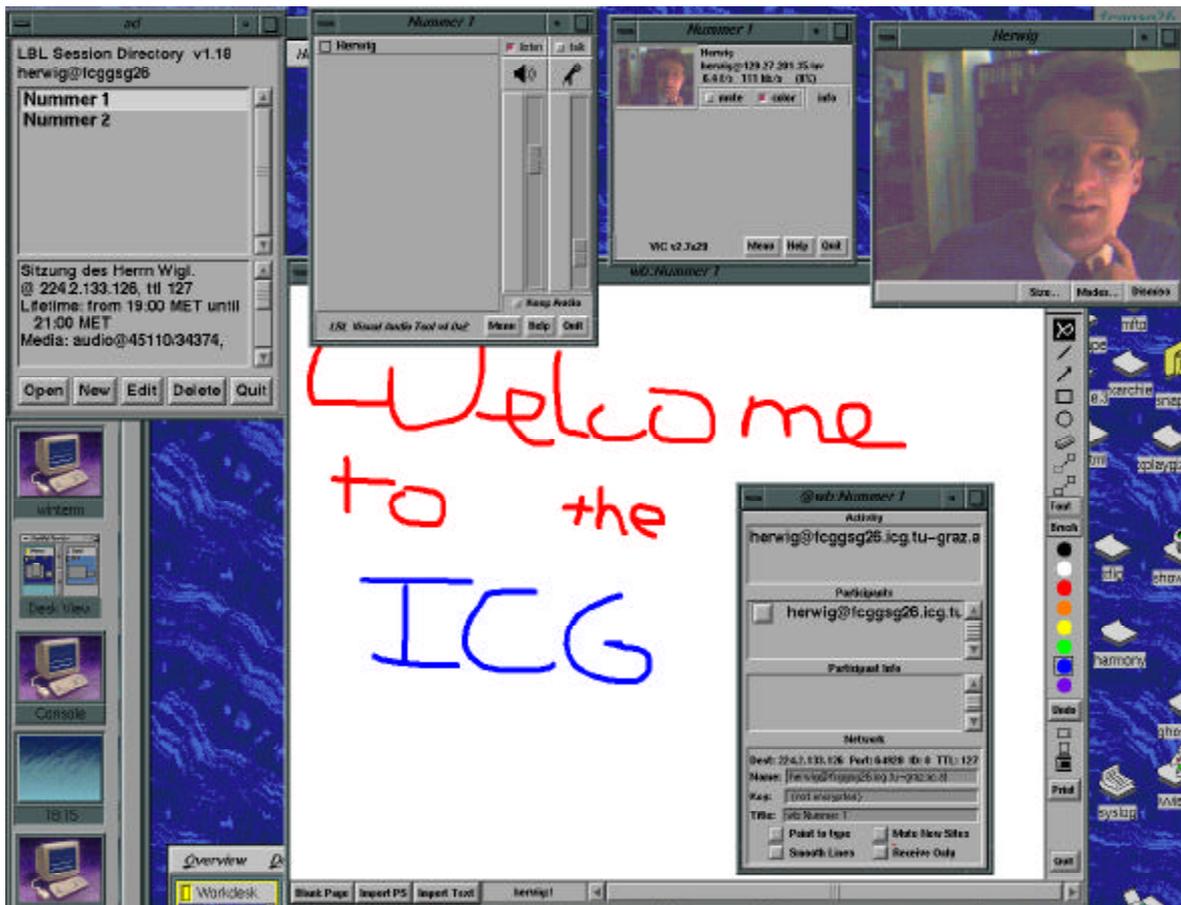
Der PC liegt weit hinter der Leistung der Workstations zurück. Dies ist teils darauf zurückzuführen, daß der PCI Bus nicht für Hochgeschwindigkeitsadapter optimiert ist, teils darauf, daß Windows NT in keiner Weise für ATM eingerichtet ist. Auch hier ist eine deutliche Verbesserungen im Durchsatz erst mit der Verfügbarkeit besserer Hardware bzw. Treiber zu erhoffen.

### **6.3 Video Konferenz ATM Pilotversuch mit dem VCPC**

Um die Fähigkeiten von ATM bezüglich Transport von Sprache und Video zu testen, wurde ein Videokonferenz-Pilotversuch zwischen dem ICG in Graz und dem VCPC (Vienna Center for Parallel Computing) durchgeführt. Der Versuchsablauf sowie das Ergebnis werden in diesem Unterkapitel näher erläutert.

#### **6.3.1 Versuchsablauf**

Für den Versuch fanden ausschließlich Public Domain Programme Verwendung. Die Videoübertragung wurde mit dem VIC (VIdeo Conferencing Tool) [VIC, 96], die Audioübertragung mit dem VAT (VIdeo Audio Tool) [VAT, 96] bewerkstelligt. Beide Programme wurden von der Network Research Group am Lawrence Berkeley National Laboratory (LBL) entwickelt. Für die verteilte Visualisierung von Bildern, Texten und Diagrammen wurde die ebenfalls von LBL entwickelte WB (White Board) Applikation [WB, 96] installiert. Alle Programme sind eigentlich für den Einsatz im Internet M-Bone (Multicasting Backbone) [Macedonia, 96] vorgesehen und funktionieren selbst bei geringer Bandbreite noch mit ausreichender Qualität.



**Abbildung 6.12:** *Bildschirmaufbau einer Videokonferenz.* Links oben das Sitzung-Koordinierungs Programm, rechts daneben die Audio Applikation. In der rechten Ecke die Videoübertragungs Einheit. In der unteren Hälfte ist das White Board mit seinem Kontrollfenster zu sehen.

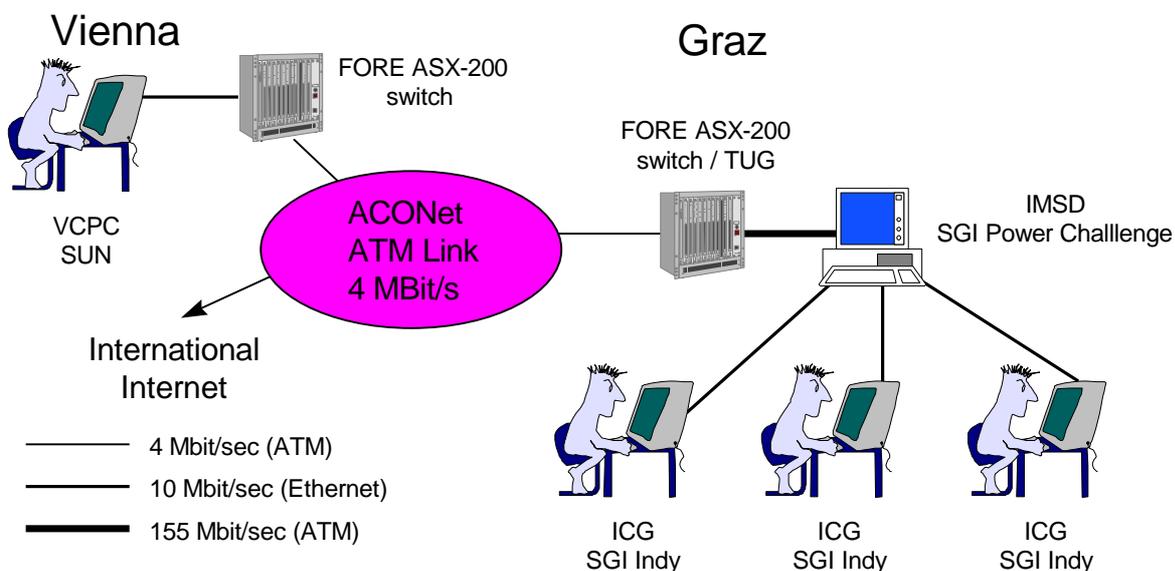
Um schließlich die einzelnen Sitzungen zu koordinieren, wurde das von der UCL (University College London) entwickelte Session Directory [SD, 96] Programm verwendet. Mit diesem Programm kann man eine Videokonferenz Sitzung definieren - also z.B. ob man das White Board verwendet, ob man Audio will etc. - und bei allen anderen Teilnehmern, die das Programm ebenfalls laufen haben, anmelden. Um an der jeweiligen Sitzung teilnehmen zu können, klickt man die gewünschte Sitzung einfach an.

Wie sich die einzelnen Programme am Bildschirm darstellen, ist Abbildung 6.12 zu entnehmen. Man sieht in der linken oberen Ecke die SD, die derzeit 2 Sitzungen (Nummer 1 und Nummer 2) anbietet, an denen man durch einfaches Selektieren teilnehmen kann. In der rechten oberen Ecke sieht man die beiden Fenster des Video Übertragungsprogrammes, wobei rechts das vergrößerte Teilnehmerfenster, links das Übersichtsfenster mit allen an der aktuellen Sitzung teilnehmenden Personen zu sehen ist. Gleich links daneben befindet sich das Fenster zur Steuerung des Sprachverkehrs.

Die untere Bildschirmhälfte wird von den beiden Fenstern des White Boards genutzt. Das kleine Fenster dient wiederum - ähnlich wie beim Video Tool - der Übersicht über alle an der aktuellen Sitzung teilnehmenden Personen, das große weiße Fenster ist die unter allen Benutzern geteilte Fläche, wo jeder nach Belieben zeichnen, schreiben bzw. Postscript Graphiken einfügen kann.

Das White Board gleicht einem Block mit beliebig vielen Seiten, die man entsprechend umblättern kann. Das Kontrollfenster ermöglicht es weiter, die Eingaben einzelner Teilnehmer aus- bzw. einzublenden. Dies ist besonders hilfreich, wenn eine große Teilnehmerzahl an der Konferenz beteiligt ist und die Seite unübersichtlich zu werden droht. Ein Anwender kann selbstverständlich - im Gegensatz zu dem im Kapitel 6.4 beschriebenen Versuch - nur die von ihm selbst eingebrachten Texte, Graphiken, Bilder manipulieren bzw. löschen.

Der ATM Pilotversuch verwendete das in Abbildung 6.13 dargestellte ATM Backbone Netzwerk. Der gesamte Versuch wurde im Rahmen des DIANE (Distributed ANnotation Environment) [DIANE, 96] EU Projektes durchgeführt, welches die verteilte Annotation von wissenschaftlichen Arbeiten sowie multimediale Dienste zwischen heterogenen Plattformen, die über einen ATM



**Abbildung 6.13:** Videokonferenz Szenario, wie es im GDSS Testversuch verwendet wurde. Das ICG in Graz war über eine SGI Power Challenge direkt an den ATM Netzwerk Backbone der TU Graz angeschlossen. Von dort gab es über das ACONet eine ATM Verbindung nach Wien, wo am dortigen EDV Zentrum der UNI Wien das VCPC angebunden war. Dieses Szenario diente zahlreichen Versuchen, die im Text ausgeführt sind.

Backbone verbunden sind, evaluiert.

Der vorliegende ATM Pilotversuch demonstriert die Funktionalität der obigen

Applikationen zwischen SUNs und SGI Indys über einen ATM Netzwerk Backbone entsprechend Abbildung 6.13. Neben der Videokonferenz wurde das White Board dazu benutzt, um Pascal Quellcode des VCPC zu kommentieren und zu erklären. Hierzu wurden auch Postscript Flußdiagramme visualisiert und vom Teilnehmer am ICG hinterfragt.

### 6.3.2 Bewertung des Ergebnisses

Insgesamt gesehen eigneten sich alle verwendeten Applikationen für das Aufsetzen einer geographisch verteilten Videokonferenz zwischen heterogenen Plattformen. Da die Programme für den M-Bone entworfen wurden, liefern sie auch noch bei geringer Bandbreite - die Übertragungsrate läßt sich beim Video Tool manuell einstellen - akzeptierbare Ergebnisse. Das Whiteboard wird von der geringen Bandbreite am wenigsten beeinflusst, kritischer sind hier die EchtzeitTools Audio und Video. Das Audio Tool senkt bei Bandbreitenengpässen automatisch die Übertragungsrate, wodurch jedoch in der Praxis keine Nennenswerten Verständigungsschwierigkeiten entstanden. Das Video Tool senkt die Frame Rate. Dadurch werden schnelle Bewegungsabläufe nur ruckartig übertragen. Dies hat in der Praxis auch zu keiner nennenswerten Beeinträchtigung geführt, da man sich während einer Videokonferenz zumeist ohnehin nicht ständig in Bewegung befindet.

Da das Internet in Normalfall jedoch nur über eine begrenzte Bandbreite verfügt, die zwischen allen Teilnehmern gerecht geteilt werden muß, war die maximale Bandbreite der Video Applikation mit 1 Mbit/sec begrenzt. Um jedoch eine ruckfreie Übertragung zu gewährleisten, müßte man z.B. unter der Verwendung von MPEG-II mindestens 1,5 Mbit/sec rein zur Verfügung haben [Fogg, 95]. Dazu muß man allerdings einen eigenen, ungeladenen Kanal verwenden - dieser stand für den Versuch nicht zur Verfügung. Ein weiterer Nachteil war die Begrenzung des White Board Postscript Import auf maximal 32 kByte. Komplexere Graphiken konnten somit nicht verteilt werden, sehr wohl aber einfacher Programm Text, der sich über mehrere Seiten des White Boards hin zog und von allen Teilnehmern annotiert werden konnte.

Sobald diese Beschränkung aufgehoben wird, eignen sich die getesteten Applikationen für einen Einsatz im GDSS zur Annotierung von Fernerkundungsdaten bzw. zum Austausch von Wissen. Geeignet wäre das Tool auch für das im Kapitel 3.5 vorgestellte Anwendungsszenario „MISSION“, wo ein primäres Ziel der Austausch und die Annotierung von Fernerkundungsdaten zwischen den Partnern ist.

## 6.4 ATM Pilotversuch mit dem VCPC: Application Sharing

Zweck dieses Versuches war es, die Fähigkeiten eines ATM Netzwerk Backbones

bezüglich der Verwendung von verteilten X-Window Applikationen zu testen. Eine solche Applikation könnte eine Software zur Bearbeitung von Fernerkundungsdaten - wie etwa der ER-Mapper - sein. Diese Software sollte mehreren, geographisch entfernten Personen die Möglichkeit bieten, Fernerkundungsdaten zu bearbeiten und zu annotieren. X-Window Applikationen haben in der Regel wegen ihrer GUIs einen hohen Bedarf an Bandbreite, insbesondere wenn es sich um komplexe Fernerkundungs-Applikationen handelt. Das Unterkapitel stellt den Versuchsablauf, die verwendeten Applikationen sowie das Ergebnis dar.

#### 6.4.1 Versuchsablauf

Dieser Versuch wurde, wie der im Kapitel 6.3 beschriebene, mit einem Public Domain Programm namens X-wedge durchgeführt, das allerdings speziell für die SGI Plattform konvertiert werden mußte<sup>1</sup>. X-wedge wurde für das CIO (Coordination, Implementation and Operation of multimedia services) Projekt [Gutekunst, 95] von der ETH Zürich entwickelt, erste Verwendung fand es allerdings erst im sogenannten BETEUS (Broadband Exchange for Trans-European USage) Projekt [Beteus, 96], [Walter, 95], [Gutekunst, 95]. Ziel dieses Projektes ist es, Initiativen in der Fernlehre sowie für kooperative Netzwerke zu setzen. Die X-wedge Applikation ging als ein wichtiges Werkzeug in dieses Projekt ein. Grundsätzlich ermöglicht X-wedge die verteilte Gruppenarbeit zwischen mehreren X-Window Workstations.

X-wedge erlaubt das Teilen („sharing“) von Applikationen zwischen mehreren heterogenen UNIX Plattformen, wobei im Moment SUNs und SGIs unterstützt werden (bald auch Windows NT<sup>2</sup>). X-wedge stellt nicht nur das GUI der lokal laufenden Applikation auf den Schirmen der anderen Teilnehmer dar, sondern erlaubt auch die Vergabe von Benutzungsrechten. So kann etwa der Besitzer C der verteilten Applikation den Benutzern A und B erlauben, die Applikation ebenfalls zu bedienen. Dies geschieht natürlich mit allen damit verbundenen Konsequenzen. Teilt man z.B. ein X-terminal, und der Benutzer A nutzt seine Rechte, um „rm -rf<sup>3</sup>“ einzugeben, so passiert das auch genauso, als hätte der Besitzer C diesen Befehl ausgeführt. Bezüglich der Sicherheit muß man sich also vorher Gedanken machen bzw. darf man kritische Applikationen eben nur Benutzern freigeben, welche zu vereinbarten Verhaltensweisen verpflichtet sind. Mit diesem Merkmal wird jedoch Remote Prozessierung sehr einfach möglich, eine rechenintensive Applikation kann lokal auf einem Großrechner verweilen, das Benutzerinterface und die Ergebnisse werden beim Benutzer visualisiert.

---

<sup>1</sup> Hierbei sei dem Entwickler Markus Brunner, von der ETH Zürich, für die Unterstützung gedankt.

<sup>2</sup> Dies hängt von der Verfügbarkeit eines X-Servers unter NT ab, da X-wedge nur über das X-Window Protokoll läuft.

<sup>3</sup> das UNIX Kommando um Files rekursiv zu löschen

Dieser Versuch wurde ebenfalls im Rahmen des DIANE Projektes (siehe Kapitel 6.3) durchgeführt und benutzte ebenfalls das in Abbildung 6.13 dargestellte Szenario. Im Versuch selbst wurden 2 Applikationen - eine am VCPC, eine am ICG - gestartet und in einem ersten Schritt die Funktionalität dem jeweils anderen Partner erklärt. Dies erfolgte durch einfaches Demonstrieren der GUI Funktionalität - der andere Partner konnte mitschauen. Sodann wurden die Benutzerrechte freigegeben und der jeweilige Partner konnte das soeben Gelernte nun praktisch anwenden.

#### **6.4.2 Bewertung des Ergebnisses**

Die erste Version des X-wedge Programmes war noch reichlich instabil, da die Portierung auf die SGI noch nicht hundertprozentig funktionierte. Dies wirkte sich in zahlreichen unkontrollierten Abstürzen aus, weiters funktionierte das Verteilen des Netscape Browsers nicht. Dieser wurde verwendet, um die Funktionalität eines Java Applets zu demonstrieren. Außerdem wurden die Bildschirmfarben nicht korrekt dargestellt.

X-wedge-Version 5.1, die letzte Revision des Programmes, erwies sich dann aber als weitaus stabiler. Die oben beschriebenen Experimente konnten ohne Zwischenfälle zügig durchgeführt werden, da Netscape mittlerweile ordnungsgemäß verteilt werden konnte. Lediglich die Darstellung der Farben war nach wie vor nicht korrekt, dies liegt allerdings an den unterschiedlichen Farbtabellen, die SUN und SGI verwenden. Verwendet man X-wedge auf ein und demselben System, ergibt sich dieses Problem nicht mehr. Die Durchführung der Versuche wurde aber durch die Falschfarben-Darstellung nicht beeinflusst.

Insgesamt war der Versuch ein Erfolg. Es war mit X-wedge problemlos möglich, X-Windows Applikationen in Arbeitsgruppen zu teilen. Der ATM Backbone stellte trotz der 4 Mbit/sec Beschränkung zwischen Graz und Wien genug Kapazität zur Verfügung, um ein ungestörtes Arbeiten mit der verteilten Applikation - d.h. die Benutzer konnten aufgrund der Reaktionszeiten des Systems nicht unterscheiden, ob sich die Anwendung lokal oder an einem anderen Ort befand - zu gewährleisten.

Eine Integration in das GDSS System ist anzustreben, X-wedge könnte Aufgaben wie: Fernzugriff auf Großrechner und Fernlehre von Fernerkundungssoftware wahrnehmen.

### **6.5 GDSS Prototyp zum Abfragen von FBIDR**

Im Rahmen des Projektes wurde ein GDSS Prototyp entwickelt, welcher auf Java, einem Applikations-Server von Weblogic Inc., JDBC, sowie einer Oracle Datenbank beruht. Dieser Prototypen ist im 5. Kapitel beschrieben. Hier seien nur kurz die Funktionalität des GDSS Prototypen sowie die wichtigsten Ergebnisse zusammengefaßt.

Nach dem Start präsentiert sich das Programm wie in Abbildung 6.14. Im oberen Teil sieht man das Benutzer Interface mit den Buttons zum Verbinden mit der Magellan Datenbank, der Abfrage von FBIDR, dem Trennen der Datenbankverbindung sowie zwei Buttons zum Testen von Abfragen bzw. zum Herunterladen und Darstellen von Magellan Quicklooks.

Das Programm ermöglicht die Abfrage von Orbits des Magellan Venus Datensatzes innerhalb eines Suchquadrates. Nach Eingabe der Koordinaten des POI (point of interest) und der Größe des Suchquadrates in km erfolgt als letztes noch die Auswahl der gewünschten Zyklen per Checkboxen im unteren Teil des Fensters. Das Abfragefenster mit einer Koordinateneingabe ist in Abbildung 6.15 dargestellt.

Als Ergebnis werden die gespeicherten Daten jener Orbits zurückgeliefert, die wenigstens einen Aufnahmepunkt (siehe Kapitel 5.1.2) innerhalb des Suchquadrates haben. Zusätzlich werden die Ergebnisse nach Zyklen sortiert und in drei verschiedenen Ergebnisfenstern angezeigt. Das Ergebnis einer Abfrage, welche jeweils zwei Orbits im ersten (Nr. 1553 und Nr. 1554) und im zweiten Zyklus (Nr. 3374 und Nr. 3375) und keinen Orbit des dritten Zyklus zurückgeliefert hat, ist Abbildung 6.17, Abbildung 6.18 und Abbildung 6.19 zu entnehmen.

Der Klient ist als Java-Applet implementiert worden und vollständig in Java geschrieben. Er kann mit jedem Standard-javafähigem Internet-Browser aufgerufen und exekutiert werden. Updates entfallen, da beim Start die jeweils letzte Version des Applets vom Server geladen wird.

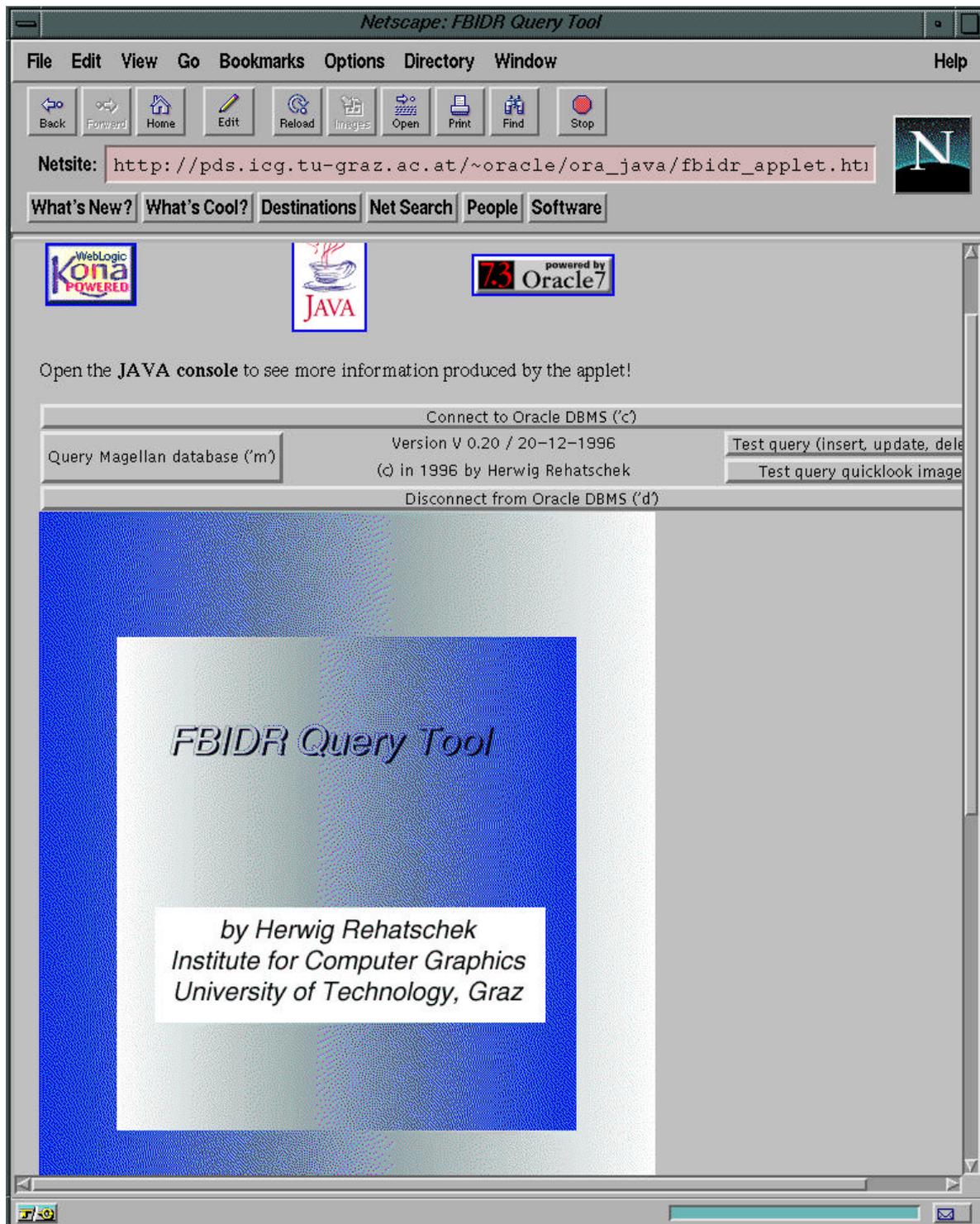
Der Prototyp verwendet eine multitier Logik, das bedeutet, daß sich beim Benutzer keine herstellereinspezifischen Bibliotheken (z.B. für Oracle, Sybase) befinden müssen. Diese verweilen bei einem sogenannten Applikations-Server, der von Weblogic Inc. geliefert wurde. Dieser Server setzt die Abfragen des jeweils verbundenen Applets auf die spezifische Datenbank um, die händlerspezifischen Bibliotheken verweilen dort und sind unabhängig von der Zahl der angeschlossenen Benutzer nur einmal gespeichert.

Hinter dem Weblogic Server steht ein relationales DBMS, wobei hier die Wahl auf Oracle gefallen ist. Der Oracle Server speichert einerseits alle Magellan Metadaten, sowie exemplarisch auch einen Quicklook, der als BLOB im JPEG Format mit einem eigens dafür entworfenen Programm eingebracht wurde. Dieser Quicklook kann zur Demonstration über das Java Applet geladen und - wie in Abbildung 6.20 dargestellt - visualisiert werden. Diese Funktion kann später dahingehend erweitert werden, daß man zu den Abfrage Ergebnissen automatisch auch die zugehörigen Quicklooks anzeigen kann.

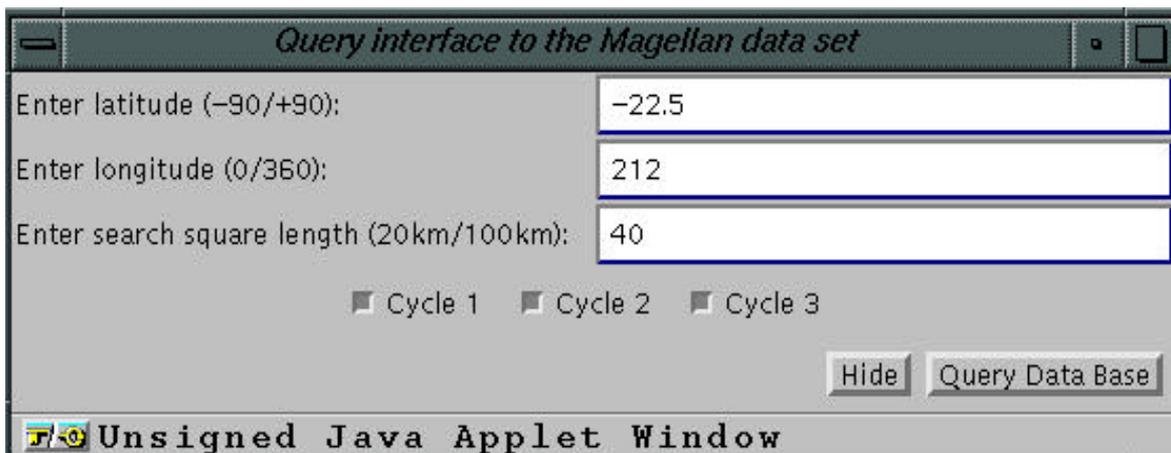
Die Versuchskonfiguration des GDSS Prototypen, welche auch vom im 4. Kapitel beschriebenen ATM Backbone Gebrauch macht, ist in Abbildung 6.16 dargestellt. Das ICG fungiert als Datenprovider, indem es einen zentralen Server für die NASA Magellan Daten mit einer international genormten Schnittstelle (ANSI SQL) zur

Verfügung stellt. Am ICG laufen zwei Server, wobei Anfragen von Klienten direkt an den - gemäß der im 5. Kapitel beschriebenen multi-tier Logik - Applikations-Server weitergeleitet werden. Dieser greift dann auf die Oracle Datenbank zu. Vorteil des multi-tier Logik ist, daß man am Applikations-Server noch weitere Datenbanken anschließen kann. So könnte z.B. eine Datenbank zur Verwaltung der MISSION Metadaten der bestehenden Magellan Datenbank hinzugefügt werden.

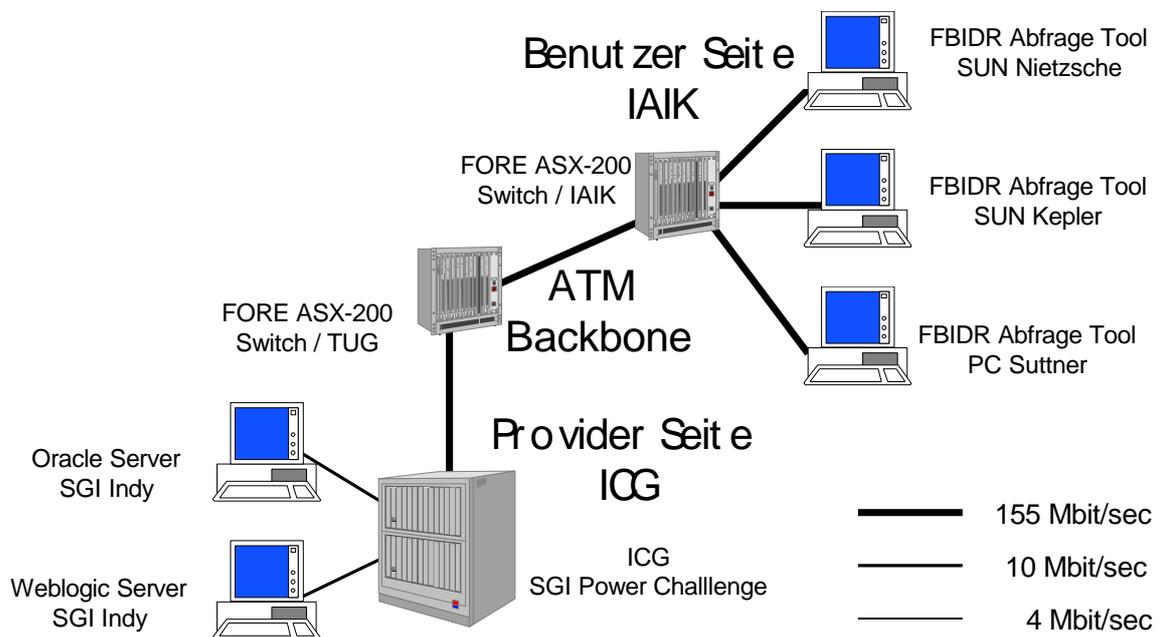
Das IAIK stellt den Benutzern eine Reihe von GDSS Klienten zur Verfügung, welche über den ATM Backbone mit dem zentralen Server am ICG verbunden sind. Die GDSS Klienten ermöglichen die Suche nach FBIDRn des NASA Magellan Datensatzes, sowie das Herunterladen eines exemplarisch gespeicherten FBIDR Quicklooks.



**Abbildung 6.14:** *Benutzer Interface des GDSS Prototypen.* In der oberen Hälfte des Applets befinden sich die Benutzer Buttons. Mit den beiden langen Buttons kann man sich zur Datenbank verbinden bzw. diese Verbindung wieder trennen. Der Button auf der linken Seite ermöglicht die Suche nach FBIDRn. Die Buttons auf der rechten Seite dienen zu Testzwecken. Der obere ermöglicht eine Testquery (einfügen, auffrischen und löschen von Datensätzen in einer Testtabelle), der untere ermöglicht das Herunterladen eines Magellan Quicklooks.



**Abbildung 6.15:** Abfragefenster des GDSS Prototypen. In die ersten beiden Felder erfolgt die Eingabe der Suchkoordinaten in Breiten- ( $-90^{\circ}$  -  $+90^{\circ}$ ) und Längengraden ( $0^{\circ}$  -  $360^{\circ}$ ). Danach definiert man die Seitenlänge des gewünschten Suchquadrates in km, wobei Angaben zwischen 20 und 100 km erlaubt sind. Zuletzt kann man anhand der Checkboxes auswählen, von welchen der drei Zyklen man Ergebnisse erhalten möchte.



**Abbildung 6.16:** Laborversuchs Konfiguration des GDSS Prototypen. Auf der rechten Seite der Abbildung ist das IAIK, welches das FBIDR Abfrage Tool einer Reihe von Benutzern zur Verfügung stellt. Das Abfrage Tool ist über den zwischen dem IAIK und dem ICG aufgebauten ATM Backbone direkt mit der SGI Power Challenge am ICG verbunden, welche die Abfragen gemäß der im Kapitel 5 beschriebenen multi-tier Logik an den Weblogic Applikations-Server weiterleitet. Dieser wiederum kann auf die Oracle Datenbank Server zugreifen, welcher die Magellan-Metadaten verwaltet.



**Abbildung 6.17:** Abfrage Ergebnisse für den ersten Zyklus. Es wurden die Orbits mit den Nummern 1553 und 1554 gefunden, die mindestens einen Aufnahmepunkt innerhalb des Suchrechteckes hatten. Das Suchrechteck war durch einen Suchpunkt bei  $-22,5^\circ$  Breite, 212 Grad Länge und einer Suchquadrat-Seitenlänge von 40 km definiert.

```

----- start of new query -----
The POI (long=212/ lati=-22.5) is covered by the following FBIDRs:
The search square is defined by 40x40 km

*****
Information on Orbit Number 3374
*****

ASCENDING_NODE_LONGITUDE | LOOK_DIRECTION | START_TIME_OF_MAPPING
-----
300.8452186105 | RIGHT_C252 | 91/300-20:29:37.400

STOP_TIME_OF_MAPPING | NAVIGATION_SOLUTION_ID | ORBITAL_ECCENTRICITY
-----
91/300-20:56:39.133 | ID = M1299-16 | .3941052202247

ORBITAL_INCLINATION | ORBITAL_SEMIMAJOR_AXIS | START_TIME_OF_ORBIT
-----
96.36792729013 | 10424.66419024 | 91/300-19:55:31.800

STOP_TIME_OF_ORBIT | PERIAPSIS_ARGUMENT_ANGLE | TIME_OF_PERIAPSIS
-----
91/300-23:11:05.000 | 149.8383652798 | 91/300-20:27:34.029

START_TIME_OF_POLAR_MAPPING | STOP_TIME_OF_POLAR_MAPPING
-----
91/300-20:52:34.768 | 91/300-20:57:36.471

*****
Information on Orbit Number 3375
*****

ASCENDING_NODE_LONGITUDE | LOOK_DIRECTION | START_TIME_OF_MAPPING
-----
300.8449265456 | RIGHT_C252 | 91/300-23:45:10.600

STOP_TIME_OF_MAPPING | NAVIGATION_SOLUTION_ID | ORBITAL_ECCENTRICITY
-----
91/301-00:12:12.333 | ID = M1299-16 | .394106082982

ORBITAL_INCLINATION | ORBITAL_SEMIMAJOR_AXIS | START_TIME_OF_ORBIT
-----
96.36881626486 | 10424.66025111 | 91/300-23:11:05.000

STOP_TIME_OF_ORBIT | PERIAPSIS_ARGUMENT_ANGLE | TIME_OF_PERIAPSIS
-----
91/301-02:26:38.133 | 149.8366109833 | 91/300-23:43:07.178

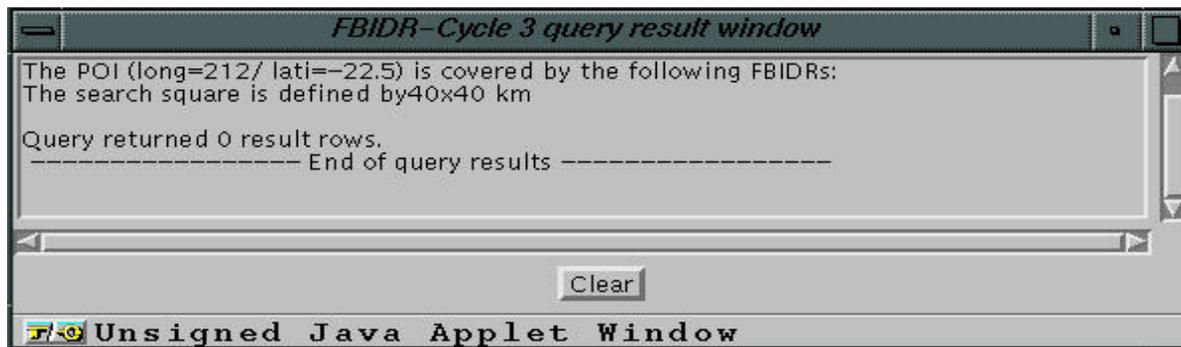
START_TIME_OF_POLAR_MAPPING | STOP_TIME_OF_POLAR_MAPPING
-----
91/301-00:08:07.962 | 91/301-00:13:09.664

Query returned 2 result rows.
----- End of query results -----

Clear
-----

```

**Abbildung 6.18:** Abfrage Ergebnisse für den zweiten Zyklus. Es wurden die Orbits mit den Nummern 3374 und 3375 gefunden, die mindestens einen Aufnahmepunkt innerhalb des Suchrechteckes hatten. Das Suchrechteck war durch einen Suchpunkt bei  $-22,5^\circ$  Breite, 212 Grad Länge und einer Suchquadrat-Seitenlänge von 40 km definiert.



**Abbildung 6.19:** Abfrage Ergebnisse für den dritten Zyklus. Es wurden keine Orbits des dritten Zyklus gefunden, die mindestens einen Aufnahmezeitpunkt innerhalb des Suchrechteckes hatten. Das Suchrechteck war durch einen Suchpunkt bei  $-22,5^\circ$  Breite, 212 Grad Länge und einer Suchquadrat-Seitenlänge von 40 km definiert.



**Abbildung 6.20:** Visualisierung eines FBIDR Quicklooks. Exemplarisch wurde ein Quicklook in der Magellan Datenbank als BLOB (Binary Large Object) gespeichert. Diesen kann man über das Benutzer

Interface herunterladen und in einem eigenen Fenster anzeigen.

## 6.6 Literatur und Referenzen

- [Andrikop, 96] Andrikopolous I., Örs T., Matijasevic M., Leitold H., Jones S., Posch R. „TCP/IP Throughput Performance Evaluation for ATM Local Area Networks“. Proc. of the IFIP TC6 „Fourth workshop on Performance Modeling and Evaluation of ATM Networks“, Illkey/Great Britain, July 1996, pp. 72/1 - 72/11, Download über meine ATM Seite: <URL: <http://www.icg.tu-graz.ac.at/herwig/Research/ATM/ATM.html>>
- [Arick, 96] Arick M. „The essential guide to TCP/IP commands“. ISBN 0-471-12569-5, Wiley, New York, 1996
- [Beteus, 96] Broadband Exchange for Trans-European Usage (BETEUS), ETH-Zürich, project description, Dec. 96, <URL: <http://www.tik.ee.ethz.ch/Projects/projects.html> >
- [DIANE, 96] DIANE (DIstributed ANnotation Environment), EU project description VCPC, Dec. 96, <URL: <http://www.vcpc.univie.ac.at:80/activities/projects/DIANE.html> >
- [Fogg, 95] C. Fogg. MPEG-2, *Frequently Asked Questions*, 11. Mai 1995, Antworten auf die 68 meistgestellten Fragen betreffend MPEG 1,2,4. <URL: <http://www-plateau.cs.Berkeley.EDU/mpegfaq/MPEG-2-FAQ.htm>>
- [Fore, 95] Fore Systems Inc. „ForeRunner SBA-100/200 ATM Sbus Adapter User’s Manual“. May 1995
- [Gleissner, 91] Gleissner M. „Netzwerkstandard TCP/IP“, ISBN 3-927921-11-4, Watter Verlag, Regensburg, 1991
- [Gutekunst, 95] Gutekunst T., Bauer D., Caronni G., Plattner B. „A Distributed and Policy-Free General-Purpose Shared Window System“. IEEE/ACM Transactions on Networking, Vol. 3, No. 1, Februar 1995
- [Gutekunst2,95] Gutekunst T., Bauer D., Caronni G., Plattner B. „A Distributed and Policy-Free General-Purpose Shared Window System“. IEEE/ACM Transactions on Networking, Vol. 3, No. 1, Februar 1995
- [Laubach, 94] Laubach M. „Classical IP and ARP over ATM“. Hewlett-Packard Laboratories, IETF RFC 1577, Jänner 1994, <URL: <http://www.es.net/pub/rfcs/rfc1577.txt>>
- [Macedonia, 96] M. Macedonia, D. Brutzman, „M-Bone - An Overview“. Unpublished, 1996 <URL: [http://www-mice.cs.ucl.ac.uk/mice/mbone\\_review.html](http://www-mice.cs.ucl.ac.uk/mice/mbone_review.html)>
- [Netperf, 96] Hewlett-Packard Company, 1996 „Netperf: A Network Performance Benchmark“. Revision 2.1, <URL: <http://ww.cup.hp.com/netperf/NetperfPage.html>>
- [Prycker, 94] Prycker M. „Asynchronous Transfer Mode“. ISBN: 3-930436-03-5 Prentice Hall, 1994
- [Rehat2, 97] Rehatschek H., Homepage: Basis Informationen, Publikationen und Forschung, Jän. 97, <URL: <http://www.icg.tu-graz.ac.at/wigl>>
- [Sadiku, 95] Sadiku M. „Metropolitan Area Networks“. ISBN: 0-8493-2474-2, CRC-Press, 1995
- [SD, 96] Session Directory (SD) Version 1.18, Public domain white board tool developed by University College London (UCL), <URL: <ftp://ftp.ee.lbl.gov/conferencing/sd/>>
- [VAT, 96] Video Audio Tool (VAT) Version 4.0a2, Public domain tool implemented by the Network Research Group at the Lawrence Berkeley National Laboratory, <URL: <http://www-nrg.ee.lbl.gov/vat/>>
- [VIC, 96] Video Conferencing Tool (VIC) Version 2.7a29, Public domain tool implemented by the Network Research Group at the Lawrence Berkeley National Laboratory, 1996 <URL: <http://www-nrg.ee.lbl.gov/vic/>>

- [Walter, 95] Walter T., Brunner M, Plattner B. „BETEUS - Broadband Exchange for Trans-European Usage“. Proceedings of the IEEE Symposium on Data Highways, Bern, Schweiz, ,Oktober 1995
- [WB, 96] White Board Tool (WB) Version 1.60, Public domain tool implemented by the Network Research Group at the Lawrence Berkeley National Laboratory,  
<URL: <http://www-nrg.ee.lbl.gov/wb/>>

## 7. Kapitel

### Zusammenfassung und zukünftige Arbeit

*„Das letzte Ziel aller wissenschaftlichen Erkenntnis besteht darin, das größtmögliche Tatsachengebiet aus der kleinstmöglichen Anzahl von Axiomen und Hypothesen zu erhellen“  
(Albert Einstein)*

Im folgenden werden die wichtigsten Schritte und Ergebnisse des Projektes zusammengefaßt und die sich daraus ergebenden Konsequenzen diskutiert. Dies führt zu Empfehlungen über die Weiterführung der Arbeit zu einem voll implementierten GDSS System, welches die in Kapitel 3 beschriebene Funktionalität aufweist.

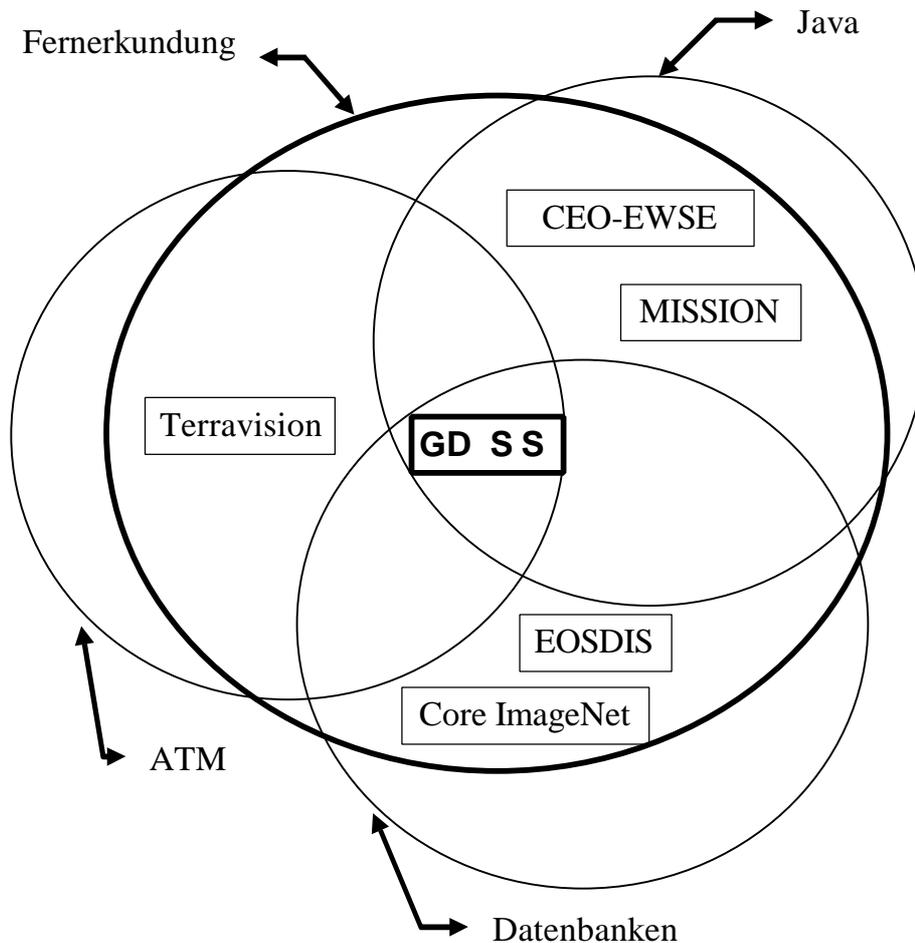
#### 7.1 Zusammenfassung

Ziel der Projektes war es, eine Systemarchitektur, Speicherkonzepte sowie ein Abfragesystem für den NASA Magellan Datensatz vom Planet Venus im speziellen, sowie Managementkonzepte für Fernerkundungsdaten im allgemeinen zu erstellen. Dies ist als Pionierarbeit zu sehen, die einen ersten funktionierenden Prototyp als Ergebnis aufweisen kann. Die insgesamt erzielten Ergebnisse reichen teilweise sogar über diese Ziele hinaus. Die eigentliche Innovation des Projektes liegt nicht in einem der Teilgebiete ATM, Java, Datenbank Systeme oder Fernerkundung sondern in der Vereinigung dieser Gebiete, wie in Abbildung 7.1 dargestellt.

In Abbildung 7.1 sind weiters einige andere Systeme eingeordnet, die im 3. Kapitel dieser Arbeit beschrieben wurden und den Stellenwert des GDSS Prototypen klarer hervorheben. Core ImageNet, welches die höchste Wertung im 3. Kapitel erhalten hatte, verfügt zwar über ein Datenbank System, nützt jedoch weder eine Hochgeschwindigkeits-Netzwerktechnologie noch Java. Gleich verhält es sich mit EOSDIS Version 0 von der NASA. Der EWSE Browser des CEO Projektes und das MISSION Projekt benutzen zwar Java-Applets, verfügten jedoch zum Stichtag 5/1995 weder über ein Datenbank System noch benutzen sie ATM. Terravision hingegen, die Applikation des MAGIC Netzes, macht Gebrauch von der ATM Technologie, benutzt jedoch weder ein Datenbanksystem noch Java.

Die prinzipiellen Entwurfskriterien für den Prototypen eines GDSS Systems waren eine Untermenge derjenigen, welche ausführlich im 3. Kapitel besprochen wurden. Dies sind:

- (1) offene Systemarchitektur,
- (2) Verwendung von internationalen Standards,
- (3) Hochgeschwindigkeits-Netzwerkbackbone,



**Abbildung 7.1:** *Innovation des GDSS Prototypen.* Das GDSS System und der GDSS Prototyp vereinen die vier Gebiete Java, Datenbank, ATM und Fernerkundung. Die Darstellung gibt weiters über die Positionierung von GDSS zu einigen anderen in dieser Arbeit vorgestellten Systemen Auskunft.

- (4) Graphisches Benutzerinterface,
- (5) einheitliches GUI, unabhängig von den verwendeten Archiven.

Die Darstellung der Arbeit gliederte sich in drei große Abschnitte, die auf die Entwicklung eines GDSS Laborversuchssystems hinzielen und die obigen 5 Entwurfskriterien berücksichtigen.

Der erste Abschnitt war der Entwurf eines prinzipiellen Konzeptes zur globalen Verwaltung von planetarischen Bild- und Metadaten. Dies wurde unter der speziellen Rücksichtnahme auf bereits vorhandene Archive getätigt. Hauptprobleme der bisherigen Archive waren, daß der Benutzer die exakte Position/Adresse des Archivs in bezug auf einen bestimmten Datensatz kennen mußte, daß jedes Archiv ein anderes, neu zu erlernendes Benutzerinterface aufwies, daß keinerlei Kostenverrechnungsmechanismen berücksichtigt wurden, daß der Zugriff über sehr

langsame Netzwerkverbindungen stattfand und daß man nicht Teile von Bildern bestellen konnte.

Das vorgestellte GDSS Konzept nimmt auf alle obigen Nachteile Rücksicht. Das 3. Kapitel führt eine Grundarchitektur ein, die dann noch in zwei Schritten verfeinert wird. Das Konzept ist ein Rahmen, welcher jedoch durch entsprechende weitergehende Projekte gefüllt werden kann.

Der zweite Abschnitt beschäftigt sich mit der Auswahl einer geeigneten Netzwerktechnologie sowie mit dem Aufbau eines Versuchsnetzwerkes. Der äußerst wichtige Aspekt für globale Systeme, das Netzwerkmanagement, wird detailliert beschrieben (siehe Anhang A). Außerdem wird noch auf die Wichtigkeit des IP Protokolls eingegangen, welches aus der derzeitigen Sicht wegen seiner weltweiten Verbreitung das einzige Protokoll ist, welches für ein globales System Verwendung finden kann. Die Wahl der Netzwerktechnologie fiel auf den 1991 von der damaligen CCITT (jetzt ITU-T) vorgestellten ATM Standard. ATM ist ein international genormter Standard und weist fünf große Vorteile auf:

- Unterstützung von hohen Bandbreiten (derzeit bis 622 Mbit/sec OC-12c) für WANs,
- einzig bekannte Technologie, die global Daten, Sprache und Video transportieren kann,
- Qualität der Dienste festlegbar,
- weitestgehend unabhängig von der physikalischen Schicht,
- gute Skalierbarkeit,
- LAN-Emulation.

Nach dem Vergleich mit Alternativtechnologien stellte sich heraus, daß ATM das hohe Anforderungsprofil des GDSS erfüllen kann und auch eine gute Wahl für die Zukunft ist.

Es wurde in der Folge ein ATM Versuchsnetzwerk zwischen dem ICG (Leitung: Prof. Leberl) und dem IAIK (Leitung: Prof. Posch) aufgebaut. Ursprünglich wurde die am ICG befindliche SGI Power Challenge mit einem FORE ASX-200 Switch im EDV-Zentrum verbunden. Von dort ging es zu einem weiteren ASX-200 Switch im IAIK, wo SUNs und PCs angeschlossen waren. Dieses Szenario (und eine Erweiterung desselben nach Wien) war Ausgangspunkt von Performance Messungen und Experimenten, die in Kapitel 6 beschrieben wurden.

Die ATM Karte der Power Challenge wird seit Herbst 1996 auch professionell vom ICG genutzt, indem die Power Challenge nun über den ATM Backbone und dem EDV-Zentrum (wo auch der Router steht) an einen Ethernet Switch angeschlossen ist, der sich ebenfalls am ICG befindet. Vorteil ist die Aufteilung des bisher aus einer großen Domäne bestehenden Institutsnetzes in insgesamt 7 Kollisions Domänen bzw. daß alle Kollisions-Domänen nun gleichzeitig auf die Power

Challenge zugreifen können.

Der dritte Abschnitt des GDSS Projektes beschäftigte sich mit der Implementierung von Klienten Software, um ein exemplarisches System auf Basis der ersten beiden Schritte zu entwerfen. Dieses System verwendete als Testdaten die von der NASA erhaltenen Magellan Metadaten.

Das System sollte vor allem die Entwurfskriterien (1), (2), (4) und (5) berücksichtigen. Somit fiel die Wahl auf das von SUN vorgestellte Java - eine objektorientierte (OO) Sprache, die die Entwicklung von sogenannten Applets gestattet. Applets sind vorcompilierte binäre Programme, die plattformunabhängig mit jedem Java-fähigen Web-Browser exekutiert werden können. Das Update-Problem von typischen Client/Server Szenarien wird dadurch gelöst, denn jedesmal wenn der Benutzer das Programm startet, wird die neueste Version vom Server heruntergeladen und lokal ausgeführt.

Um eine einheitliche Abfrage-Schnittstelle zu gewährleisten, wurde auf einen internationalen ANSI Standard, nämlich SQL, zurückgegriffen. Obwohl ein OO Datenmodell für die Speicherung von räumlichen Daten besser geeignet wäre, konnte dieses Modell mangels Fehlens eines internationalen Standards nicht ausgewählt werden. Erst ab 1999 ist ein ANSI Standard für SQL/3 in Sicht, der OO Konzepte unterstützen wird.

Um die Herstellerunabhängigkeit zu garantieren, wurde eine multitier-Logik implementiert. Dies wurde durch Aufsetzen eines eigenen Applikations-Servers erreicht, der ankommende herstellerunabhängige Abfragen unter Verwendung lokal vorhandener Bibliotheksroutinen an die jeweiligen Datenbanken weiterreicht.

Auf der Klienten Seite findet das von SUN vorgestellte JDBC (Java Data Base Connectivity) Verwendung, welches ein API zu allen relationalen Datenbanken darstellt, sofern der jeweilige Datenbankhersteller einen entsprechenden Treiber anbietet. Die Treiber werden beim Applikations Server gespeichert, der Benutzer des Klienten braucht sich daher um keinerlei Datenbanken und zu lernende Befehle Sorgen machen.

Auf der Server Seite wurde ein relationales DBMS in Betrieb genommen. Hierbei fiel die Wahl auf den weltweit am Markt führenden Datenbankhersteller Oracle. Im Sommer 1996 wurde der Oracle Enterprise Server Version 7.2.3 in Betrieb genommen und mit den Magellan Metadaten gespeist. Im Dezember 1996 wurde der Server auf die Version 7.3.2.2 aufgerüstet.

Ein Java Applet wurde implementiert, welches die Abfrage von FBIDRs (Full Resolution Basic Image Data Records) zu einem definierten Suchgebiet gestattet. Zusätzlich konnte das Ergebnis nach den 3 Zyklen von Magellan gefiltert werden. Exemplarisch wurde die Speicherung von Quicklooks als BLOBs (Binary Large Objects) in der Oracle Datenbank sowie das Laden des Bildes in das Java Applet

aufgezeigt. Das Java Applet verwendet ein GUI und läuft plattformunabhängig mit jedem Java fähigen Web Browser.

Insgesamt hat das Projekt also drei Hauptergebnisse geliefert: ein Basis-Konzept für die Verwaltung und kooperative Benutzung von planetarischen Bilddaten, eine Netzwerktechnologie, um diese Daten zu verteilen und Remote Prozessierung zu erlauben, sowie einen funktionierenden Prototypen, der die wichtigsten Entwurfskriterien berücksichtigt.

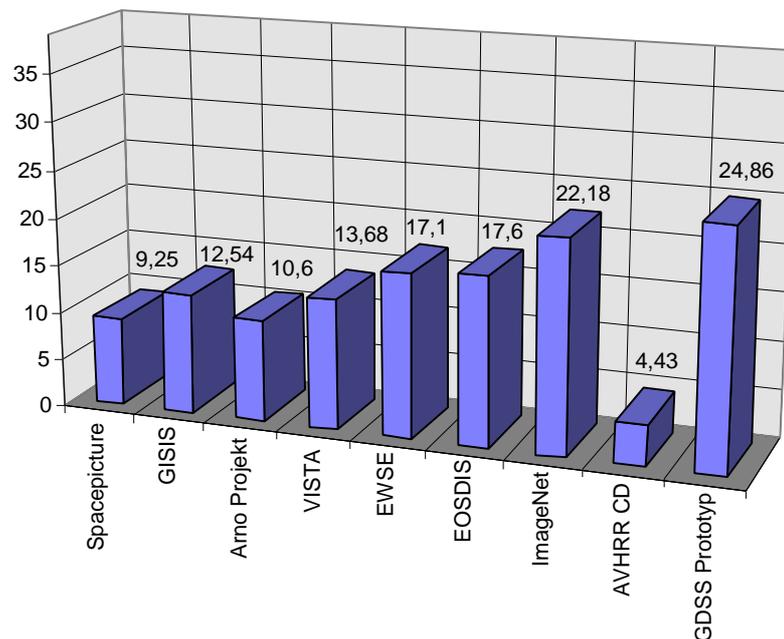
## 7.2 Bewertung des Systems

Im Kapitel 2 dieser Arbeit wurden bestehende Systeme nach 13 Kriterien bewertet und so miteinander verglichen. Das System mit der größten Punktwertung war das kommerziell erhältliche ImageNet von Core Systems. Es erreichte insgesamt eine Wertung von 22,18 Punkten von 39 möglichen. Um das Ergebnis dieser Arbeit in die Reihe der untersuchten Systeme einzuordnen, wurde in Tabelle 2.11 eine Gesamtwertung aller Ergebnisse des GDSS Projektes versucht. Primär fließt hier das FBIDR Abfrage Tool mit seiner multi-tier Logik ein, als ergänzende Komponenten sind das Video Konferenz System und das Applikations verteilende Werkzeug X-wedge zu sehen, welche ebenso über den ATM Backbone eingesetzt wurden. Mit einer Wertung von 24,86 Punkten liegt das GDSS Laborversuchssystem 2,68 Punkte vor dem Core ImageNet. Die hohe Wertung ist vor allem durch die Verwendung von Standards (TCP/IP und SQL), die Verwendung eines Hochgeschwindigkeits-ATM-Backbones, die Plattformunabhängigkeit, das verteilte Datenkonzept (multi-tier Logik) und die Modularität (OO Programmierung) zu begründen.

Die Wertungen aller im Kapitel 2 untersuchten Systeme sowie die Wertung des GDSS Prototyp Systems sind in Abbildung 7.2 veranschaulicht.

Merkmalsbeschreibung	Detailbeschreibung	Detailwertung	Gesamtwertung
Graphical User Interface	Übersichtlichkeit	3	2.5
	Funktionalität	1	
	Interaktivität	3	
	Bedienungskomfort	3	
Suchmechanismen	point of interest	3	0.5
	region of interest	3	
	Überdeckung	0	
	Meta Informationen	0	
HS Netzwerk Backbone	> 1.5 Mbit/sec (T1)	3	3
	> 44.736 Mbit/sec (T3)	3	
	> 155 Mbit/sec (OC-3)	3	
Plattformunabhängigkeit	SUN	3	3
	SGI	3	
	HP	3	
	IBM	3	
	PC	3	
	Apple	3	
Client-Server Konzept		3	3
Kostenmanagement		0	0
Verteiltes Datenkonzept		3	3
Offenes System	CIP	0	0.86
	IDN mit DIF	0	
	CINTEX Interface	0	
	Z39.50	0	
	SQL	3	
	TCP/IP	3	
	SNMP	0	
Kommerziell erhältlich		0	0
Globales Management		0	0
Lokales Management		3	3
Keine spezielle Client Software notwendig		3	3
Modulares System		3	3
Systemwertung:			24,86

Tabelle 7.15: Gesamtwertung für das GDSS Laborversuchssystem



**Abbildung 7.2:** Vergleichspunktwertung aller untersuchten Geo-Datensysteme inklusive dem GDSS Prototypen. Die höchste zu erreichende Punktezahl war 39 - das GDSS Laborversuchssystem erreicht mit 24,86 Punkten die höchste Wertung. An zweiter Stelle befindet sich Core's ImageNet, Platz drei belegt die Version 0 des NASA EOSDIS.

### 7.3 Schlußfolgerungen

Die Entwicklung von netzwerkorientierten Rechnertechnologien ist sehr dynamisch, neue Systeme erscheinen laufend. Es gilt daher, die rasche Abfolge von technischen Neuerungen für die Anwendung „Fernerkundung“ nutzbar zu machen. Hierbei ist zu beachten, daß die Möglichkeiten neuer Technologien den Ansprüchen dieser Anwendung gerecht werden.

Erste Ansätze von graphischen Oberflächenkarten in mehreren Auflösungsstufen gibt es auch bei den planetarischen Systemen, wenngleich derzeit noch kein Bildkatalog dahintergestellt ist. So verfügt etwa die Mars Multi-Scale Map (MMSM) [MMSM, 97] über einen Oberflächen-Browser, der es per GUI gestattet, die Oberfläche in mehreren Auflösungsstufen darzustellen. Trends zu kommerziellen Hochgeschwindigkeits-Netzwerken bei Fernerkundungs-Systemen konnten bisher nur national, jedoch nicht international festgestellt werden. Die Einführung globaler Hochgeschwindigkeits-Netzwerke ist derzeit noch auf Pilotversuche beschränkt.

Reine OO Datenbanken haben nach wie vor eine geringe Bedeutung. Das EOSDIS

der NASA wird in der Version 1 auf dem ORDBMS von Illustra/Informix aufsetzen, also bereits einen entscheidenden Schritt in Richtung Objektorientiertheit tätigen. Andere Systeme werden folgen, auch auf dem ICG wird derzeit ein entsprechendes Forschungsprojekt gestartet: der im Parallelprojekt [Walcher, 97] entstandene Venus Atlas Prototyp soll von einer derzeit relationalen auf eine OO Datenbank (Object Store) umgesetzt werden.

Mit Einführung eines internationalen Standards (SQL/3), was aus derzeitiger Sicht im Jahre 1999 geschehen wird, ist ein großer Umbruch in der Verwaltung räumlicher Bilddaten zu erwarten.

Das im Kapitel 3.5 vorgestellte MISSION (Multi-Image Synergistic Satellite Information for the Observation of Nature) Projekt kann durch die im GDSS entwickelten Konzepte direkt profitieren. Einerseits können die theoretischen Konzepte für die Verteilung von Daten eingearbeitet werden, andererseits kann man das im GDSS verwendete „Application Sharing“ Tool X-wedge für eine verteilte Annotierung und Bearbeitung der Daten verwenden. Der GDSS Prototyp könnte durch entsprechende Erweiterung des GUI und der Datenbank auch zur Suche von Bilddaten herangezogen werden.

### 7.3.1 Schlußfolgerungen aus dem Systementwurf

Der Entwurf der Systemarchitektur wurde bewußt offen und flexibel gehalten. Viele Systeme verwenden bereits eine große Anzahl von Entwurfskriterien, die auch in der vorliegenden Arbeit als wichtig erkannt worden sind. So bieten etwa CORENet und EURIMAGE einfache Verrechnungsprozeduren in der Fernerkundung an, indem eine E-mail mit den Bestelldaten an den Anbieter geschickt wird. Aufgrund der geringen und nicht vom Benutzer spezifizierbaren Bandbreite des Internets ist zu erwarten, daß die Bestellung von großen Fernerkundungs-Bilddaten weiter auf postalischem Wege und nicht elektronisch durchgeführt wird. Erst die Einführung einer Netzwerktechnologie, die einerseits die Definition einer Dienstqualität zuläßt und andererseits eine hohe Bandbreite zur Verfügung stellt (wie etwa ATM), könnte hier vermutlich eine Änderung herbeiführen.

Im Systementwurf wurden Entwurfskriterien für ein globales Fernerkundungsdaten System vorgestellt und argumentiert, wie sie etwa im Projekt MISSION auch verwendet werden könnten. Nicht alle dieser Kriterien haben dieselbe Priorität, ein entsprechender Prototyp des Systems läßt sich auch mit einer Untermenge derselben verwirklichen. Folgende Entwurfskriterien können für ein verteiltes Fernerkundungsdaten-System als unerläßlich erachtet werden:

- Einhaltung eines international gültigen Abfrage Standards,
- plattformunabhängiger Benutzer Klient mit GUI,
- leistungsfähiges Netzwerk (leicht skalierbar, Dienste Spezifikation),

- Lösung des Update Problems falls neue Klient Programm Version kommt,
- Anbindung einer beliebigen relationalen Datenbank an den zentralen Server,
- um globale Vernetzung zu gewährleisten, muß das System auf TCP/IP aufsetzen,
- objektorientiertes Design,
- einheitliche Abfrage, unabhängig von der im zentralen Server verwendeten Datenbank.

### 7.3.2 Schlußfolgerungen aus dem Netzwerk Backbone

Es existiert bis jetzt kein Standard für das perfekte Netzwerkmanagement System. Das OSI-FCAPS Modell ist lediglich eine Empfehlung. Die optimale Funktionalität eines Netzwerkmanagement Systems hängt allerdings sehr von den örtlichen Gegebenheiten ab. Das bedeutet im Normalfall braucht man meistens nur eine Teilmenge der OSI Empfehlung, nicht aber die gesamte Palette der Funktionalität. Zur optimalen Konfiguration eines Netzwerkmanagement Systems sind die im Anhang A vorgestellten Auswahlsschritte eine große Hilfe und sollten die Basis eines jeden professionellen Systems bilden.

Von den kommerziellen Systemen bieten IBM Netview und HP-OpenView die größte Funktionalität, besonders hervorzuheben ist das Bausteinprinzip. Man kann das Grundgerüst jederzeit um weitere Funktionen erweitern, das Grundprogramm bietet hierzu ein Standardinterface. So können herstellerspezifische Komponenten, welche meist bereits entsprechende Managementprogramme mitliefern, problemlos in ein Gesamtmanagement System eingefügt werden.

SNMP und eventuelle auch CMIP Unterstützung ist ein wesentliches Auswahlkriterium für ein brauchbares Managementsystem, jede wartbare Komponente im System sollte über ein SNMP Interface verfügen. Dies ist ein weltweit anerkannter Standard und für das Design eines globalen Managementsystems unerlässlich.

Ein geeignetes Management System für GDSS müßte als Konsequenz deshalb folgende Anforderungen, die nach Prioritäten gereiht sind, erfüllen:

- SNMP und CMIP fähig, evtl. auch Unterstützung vom zukünftigen SNMPv2 Standard
- Fehler Management („Trouble Ticket“ Generierung, umfassende Fehlerdatenbank)
- Konfiguration Management zumindest im globalen Management
- Performance Management zumindest im globalen Bereich
- Datensicherheit Management, sehr wohl im lokalen als auch im globalen Bereich

wichtig - besonders im Zusammenhang mit kostenpflichtigen Daten und Bezahlung derselbigen. Verschlüsselung von Bilddaten zumeist nicht nötig.

- Kostenmanagement (Accounting) ist für den globalen Bereich wünschenswert, kann aber auch lokal Sinn machen, z.B. wenn man ein größeres Subnetzwerk mit vielen Komponenten örtlich zu verwalten hat.

Die zunehmende Bedeutung des Internets im kommerziellen Bereich wird durch die derzeitigen Statistiken, die alle zusammen ein exponentielles Wachstum aufweisen, verdeutlicht (siehe Anhang A). Österreich begann im Jahre 1990 mit 192 registrierten Internet Rechnern, bereits am 30.4.1996 waren es 65555 Rechner. Dies zeigt, daß auch ein so kleines Land wie Österreich keinesfalls am internationalen Trend vorbeigeht, ja ganz im Gegenteil, Österreich spielt sogar eine entscheidende Rolle in der ATM Technologie. So wurden im Frühjahr 1996 die Technischen Universitäten in Wien und in Graz auf einen ATM Backbone umgestellt (OC-3, 155 Mbit/sec), derzeit geht ein österreichweites OC-3 ATM Netz in Betrieb.

Die Wachstumszahlen der Statistiken in Anhang A geben eindeutig den Trend zur „Global Village“, also zu einer integrierenden, plattformübergreifenden Kommunikationslösung (ähnlich dem Telefon) wieder. Eine erste Konsequenz daraus ist, daß ein globales Informationssystem wie das GDSS unter allen Umständen auf diesem Protokoll aufbauen muß, will es eine fast weltweite Deckung erreichen.

Eine zweite Konsequenz - betrachtet man die nationalen internationalen Bemühungen ATM Pilotversuche zu unternehmen bzw. bereits ganze MANs auf ATM umzustellen - ist, daß die Backbone Technologie in naher Zukunft bereits ATM heißen wird.

ATM erweist sich laut der Wertung aus Kapitel 4 als „hervorragend“ skalierbar. Eine Migration von bestehenden Technologien auf ATM ist ebenfalls ohne hohen technischen Aufwand möglich. Im 4. Kapitel wird am Beispiel der TU-Graz die gute Skalierbarkeit von ATM und am Beispiel des ICG die praktische Migration zu ATM demonstriert. Der Umstieg von bestehenden Netzwerkinfrastrukturen auf ATM ist für kommerzielle und staatliche Institutionen demnach schrittweise mit jeweils klar definierten Kosten möglich.

Betrachtet man die wirtschaftlichen Prognosen, ist mit einem deutlichen Marktanstieg der ATM Technologie zu rechnen. Prognosen von IDC zufolge, wird sich alleine in West Europa der Anteil an ATM Technologie bis 1998 vervierfachen. In Europa wird es vor allem im LAN/MAN Sektor in den USA sowohl im LAN/MAN als auch im WAN Bereich große Zuwächse geben.

Technisch gesehen erscheint die im Jahre 1996 noch in der Pilotphase steckende ATM Technologie bereits sehr ausgereift. Mit ihren Adaption Schichten kann ATM, gemäß der Definition der ITU-T, alle bestehenden Dienste abdecken. Gleichzeitig

stellt es auch eine Rahmentechologie für zukünftige Dienste dar, die ohne größeren Aufwand in dieses generische Konzept eingefügt werden können. Aufgrund des ATM Forums wird die Standardisierung äußerst rasch und den Wünschen der Anwender entsprechend vorangetrieben, was eine internationale Kompatibilität gewährleistet.

Die ATM Technologie ist zukunftsweisend, wie sich aus nationalen und internationalen Pilotversuchen bestätigen läßt. In Österreich wird derzeit ein kommerziell verfügbares OC-3 ATM Netzwerk in Betrieb genommen [Raffer, 96]. In den USA ist ATM bereits kommerziell erhältlich und als Backbone Technologie im Einsatz. Auch lokal setzt sich ATM durch, so etwa im medizinischen Bereich, wo hohe Bandbreiten für die Übertragung hochauflösender Bilddaten benötigt werden [Schnepf, 95].

### 7.3.3 Schlußfolgerungen aus dem GDSS Prototypen, Java, JDBC und DBMS

Trotz der umfangreichen Sicherheitsvorkehrungen gab es in der Version 1.01 des JDK (Java Development Kit) noch deutliche Sicherheitslecks [McGraw, 96]. Auch weist die derzeitige Implementierung des Sicherheitsmanagers vom Netscape Navigator (Version 2.02) etliche Fehler auf. Applets, die mehr oder weniger geringen Schaden anrichten, sind unter [Applet, 96] zu finden. Es bleibt zu hoffen, daß die meisten Sicherheitslecks in der Version 1.1 des JDK gestopft sind, allerdings bleibt gleichzeitig zu erwarten, daß sich durch neue Features andere Sicherheitslecks ergeben. Die zahlreichen Sicherheitslücken von Java sind jedenfalls noch bedenklich, was eine professionelle kommerzielle Einsetzung der Applets am Markt derzeit hemmt.

Zum Stichtag 1/1996 ist die Exekution von Java Programmen noch recht langsam, da in Software implementierte „Virtual Machines“ verwendet werden. Jedoch hat Sun bereits die Verfügbarkeit der Java Station mit dem ersten Quartal 1997 angekündigt. Diese Maschine verfügt über eine in Hardware implementierte VM, die eine Programm Exekution entscheidend verschnellern kann. Bis jetzt war Java eher zu Steuer Zwecken verwendet worden, rechen-intensive Applikationen wurden nur äußerst langsam exekutiert. Mit der Einführung des Java Chips ist eine deutliche Leistungssteigerung zu erwarten.

Als besonders nützlich für die Erstellung von Java Applets erwies sich das COSMO Produkt von SGI, das eine integrierte Java Entwicklungsumgebung samt symbolischem Debugger zur Verfügung stellt. Vor allem der Debugger ist äußerst hilfreich bei der Fehlersuche, die ja sonst alles andere als trivial wäre. Probleme ergaben sich hauptsächlich bei der Programmierung des GUIs sowie der Übergabe von einem JPEG Bytestream an den Java JPEG Decoder. Hieraus folgt, daß die Entwicklung von Java Programmen und Applets zum Stichtag 12/1996 keinen unbedeutenden Entwicklungsaufwand darstellt.

Insgesamt ist Java mit den sogenannten Java Applets aufgrund folgender Vorteile eine gute Wahl: Plattformunabhängigkeit, Flexibilität, Netzwerkorientiertheit, Sicherheit und Offenheit - alles zusammen wichtige Design Aspekte für globale Systeme in der Fernerkundung die völlig konform mit jenen des GDSS sind.

Bereits im Kapitel 5.1.1 wurde auf die schlechte Speichermöglichkeit von räumlichen Daten im relationalen Datenmodell hingewiesen. Die Suchzeit steigt mit dem linearen Anstieg des Suchrechteckes exponentiell. Dies wirkt sich im bestehenden GDSS Prototyp besonders durch die Größe der *orbit points* Tabelle (siehe Kapitel 5.1.2) aus. Hieraus folgt, daß ein relationales DBMS - welches den Vorteil einer international standardisierten Abfragesprache bietet - nicht ohne weiteres für die Verwaltung von räumlichen Daten geeignet ist. Hier ist man zur Verwendung von sogenannten „Spatial Options“ gezwungen, die eine räumliche Speicherung im relationale Modell ermöglichen. Diese Erweiterungen werden z.B. von Oracle und Informix angeboten. Eine andere Möglichkeit wäre, die Datenmenge intelligent zu reduzieren, sodaß die Abfragen wieder schneller werden. Dies ist jedoch stark von der Art des Datensatzes abhängig und daher nur individuell anzuwenden. Für den NASA Magellan Datensatz wird dies derzeit am ICG durchgeführt (siehe auch Kapitel 7.4).

JDBC (Java Data Base Connectivity) ist ein Java API, welches die Anbindung beliebiger relationaler DBMS erlaubt, sofern der jeweilige Anbieter einen „native Driver“ zur Verfügung stellt. Für GDSS wurde allerdings eine multitier Lösung unter Verwendung der Weblogic Inc. Software implementiert. Diese bietet den zusätzlichen Vorteil, daß mehrere Datenbanken an einem Applikations-Server angebunden werden können und das Klienten Programm über keine herstellereispezifischen Bibliotheken verfügen muß. JDBC ist seit Dezember 1996 in Version 1.10 bei Sun verfügbar [JDBC, 97]. Im Gegensatz zu ODBC (Open Database Connectivity) von Microsoft, ist JDBC tatsächlich plattformunabhängig. ODBC weist eine starke Abhängigkeit zum Betriebssystem Windows auf, wenngleich auf Anfrage und Bezahlung bei der Firma Visigenic [Visigenic, 97] auch ODBC Treiber für jede UNIX Plattform erhältlich sind. Möchte man jedoch unter MS Windows eine Datenbankverbindung herstellen und gleichzeitig wenig Programmieraufwand betreiben, so ist man mit ODBC besser beraten. ODBC ist bereits länger am Markt etabliert und wird von so gut wie jeder Datenbank unterstützt. JDBC ist zum Stichtag 1/1997 noch mit erhöhtem Programmieraufwand (Java Programmierung) und mangelnder Treiberunterstützung verbunden. Aufgrund der überaus vielen Ankündigungen (auch der Oracle Webserver wird in der Version 3.0 JDBC unterstützen) ist jedoch ein Anstieg der Popularität von JDBC zu erwarten.

Die Wahl von Java in Kombination mit JDBC, multitier Logik und Oracle hat sich als stabile Lösung erwiesen. Die Forderung nach einem internationalen Standard wurde mit Verwendung des ANSI SQL Standards erfüllt. Betrachtet man die

derzeitigen Trends, wird die Zukunft im OO Bereich liegen. Hier wiederum werden vermutlich die ORDBMS (Object Relational Data Base Management Systems) die Vorrangstellung übernehmen, da die Umstellung vom RDBMS zu einem ORDBMS mit keinem großen Aufwand verbunden ist und alte Applikationen weiterhin verwendet werden können.

#### 7.4 Mögliche Verbesserungen und zukünftige Arbeit

Der derzeitige GDSS Prototyp berücksichtigt noch nicht alle vorhandenen Magellan Metadaten. So werden derzeit die „Gap-Info“, der „Feature Table“ bzw. die FMIDR Informationen noch nicht in Zusammenhang mit den Abfragen verwendet. Eine erste Erweiterung könnte die Einbindung der „Gap Info“ sein, um die vorhandene Datenmenge entsprechend zu reduzieren und somit die Abfragen zu beschleunigen. Derzeit sind alle FBIDR Punkte, ohne Rücksicht darauf, ob überhaupt ein sinnvolles Ergebnis zurückgeliefert wurde, in einer Tabelle gespeichert. Die Informationen der Gap-Info Tabelle können auch für die Prozessierung einer globalen Überdeckungskarte verwendet werden.

Ein zweiter Schritt könnte die Einbeziehung der Merkmale und der FMIDR Tabelle sein, um einerseits nach Merkmalen suchen zu können und um andererseits zu einer Suchregion neben den FBIDRs auch die entsprechenden FMIDRs samt Quicklooks zu bekommen. Die FBIDRs wurden in einer parallel zu diesem Projekt verlaufenden Arbeit bereits prozessiert, Quicklooks sind in zwei verschiedenen Größen am ICG verfügbar [Walcher, 97].

Die bestehenden Abfrage Mechanismen sind entscheidend beschleunigbar, wenn man die Metadaten auf die jeweilige Abfrage hin optimiert. So könnte zum Beispiel die Datenmenge der Orbit Punkte mit Hilfe der Gap-Informationen reduziert werden, was sich in einer deutlichen Reduktion der Punkte und einer damit verbundenen deutlichen Beschleunigung der Suchabfragen auswirken würde. Weiters bietet sich an, jeweils Gruppen von gespeicherten Orbitpunkten mit dem fast selben Längengrad als eine Gerade mit Start-/Endpunkt zu speichern. Dies dürfte nach ersten Schätzungen die Datenmenge in der Orbit-Punkte-Tabelle um ein Fünftel verringern und die Suche entscheidend beschleunigen. Ein entsprechendes Projekt wird derzeit am ICG durchgeführt.

Weiters bietet sich auch die Verwendung der Oracle „Spatial Data Option“ an, die spezielle Algorithmen und Methoden für die Verwaltung von räumlichen Daten innerhalb von relationalen Datenbanken zur Verfügung stellt. Diese Option stand während des Projektes noch nicht zur Verfügung, wird aber derzeit am ICG in

---

<sup>1</sup> Dies ist dadurch begründet, daß alle Kommandos/Ergebnisse an Magellan vorausberechnet wurden - erst als der Sensor wirklich die Daten schickte, war bekannt, welche Teile aufgrund von Störungen nicht aufgenommen wurden.

einem weiterführenden Projekt untersucht.

Ein entscheidender Schritt für das gesamte GDSS Konzept wäre die Migration vom relationalen Datenbank Modell hin zu einem OO. Das OO Modell ermöglicht in einfacher Weise die Implementierung von entsprechenden räumlichen Datenstrukturen<sup>1</sup>, die eine Suche entscheidend beschleunigen würden. Außerdem ist Java eine OO Sprache, die derzeitige Konvertierung vom relationalen ins OO Modell würde ebenfalls entfallen. Weiters würde dies eine deutliche Verschmälerung des Quell-Kodes mit sich bringen. Die Verwendung eines OO Modells wurde wegen Fehlens eines entsprechenden Standards verworfen. Jedoch sollte ab 1999 ein Standard zur Verfügung stehen.

Obwohl das gesamte Projekt stark von dem NASA Magellan Datensatz der Venus beeinflusst worden ist, sind die meisten Konzepte, Ideen und die entwickelte Software auch auf andere Datensätze (Geodaten inkludiert) projizierbar. So können die erzielten Ergebnisse und die prinzipiellen Konzepte wertvollen Input für nationale Projekte, wie etwa MISSION (Multi-Image Synergistic Satellite Information for the Observation of Nature) [Mission, 97] oder internationale Projekte, wie etwa das von der EU initiierte CEO (Center for Earth Observation) [CEO, 97] oder NASA's MTPE (Mission to Planet Earth) [MTPE, 97] liefern.

---

<sup>1</sup> etwa R-Trees [Samet, 90], [Walcher, 97]

## 7.5 Literatur und Referenzen

- [Applet, 96] A Collection of Increasingly Hostile Applets, Homepage,  
<URL: <http://www.math.gatech.edu/~mladue/HostileApplets.html>>
- [CEO, 97] CEO (Center of Earth Observation), Homepage, Jän. 1997, <URL: <http://ceo-www.jrc.it>>
- [JDBC, 97] Java Data Base Connectivity (JDBC), Homepage, Jän. 97,  
<URL: <http://splash.javasoft.com/jdbc/>>
- [McGraw, 96] McGraw G., Felten E. „Java Security: Hostile Applets, Holes, & Antidotes“. ISBN: 0-471-17842-X, John Wiley and Sons, 1996
- [Mission, 97] MISSION (Multi-Image Synergistic Satellite Information for the Observation of Nature), Jan. 97, Homepage, <URL: <http://www.icg.tu-graz.ac.at/mission>>
- [MMSM, 97] Mars Multi-Scale Map (MMSM), University of California, Homepage, Jän. 1997,  
<URL: <http://www.c3.lanl.gov/~cjhamil/browse/mars.html> >
- [MTPE, 97] Nasa, Mission To Planet Earth (MTPE), Homepage (MTPE), Jän. 1997,  
<URL: <http://www.hq.nasa.gov/office/mtpe/>>
- [Raffer, 96] Raffer P., Rehatschek H. „Die digitale Vernetzung Österreichs“, Telematik Ingenieur Verband Magazin Ausgabe 4/96, Dez. 1996
- [Samet, 90] Samet H. „Applications of Spatial Data Structures“, ISBN: 0-201-50300-X, Addison and Wesley, 1990
- [Schnepf, 95] Schnepf J., Du D., E. Russel-Ritenour, Fahrman A. „Building Future Medical Education Environments Over ATM Networks“, Comm. of the ACM, Vol. 38, No. 2, Feb. 1995, Seite 55-69
- [Visigenic, 97] Visigenic Inc., Homepage, Jänner 1997, <URL: <http://www.visigenic.com>>
- [Walcher, 97] Walcher W., Design Aspects of Information Systems for Planetary Image Data. PhD thesis, Institute for Computer Graphics, University of Technology, Graz-Austria, Jän. 1997



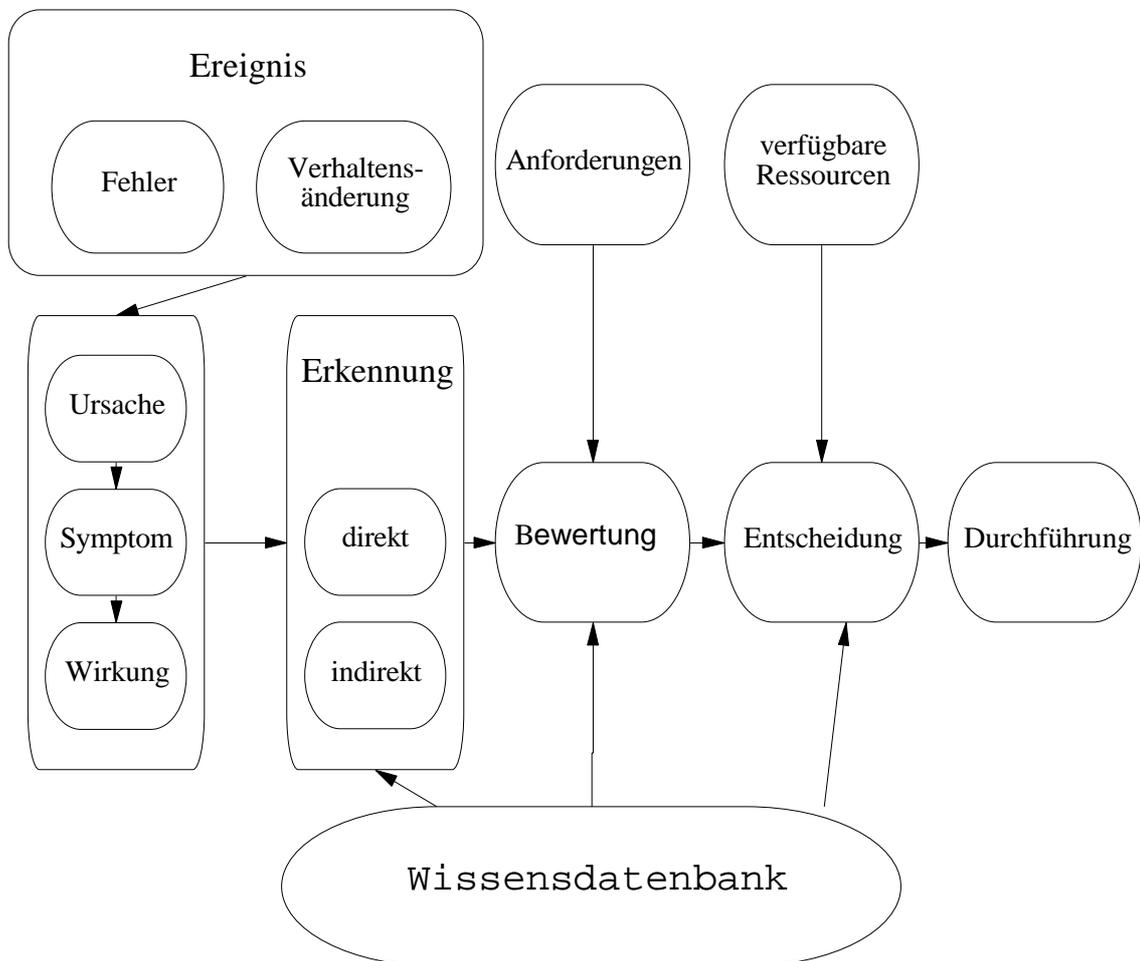
## Anhang

### Anhang A: Netzwerk

#### Einführung ins Netzwerkmanagement

Netzwerkmanagement besteht aus vielen unterschiedlichen Komponenten, welche im Anhang Abschnitt „Management Funktionalität“ genauer definiert werden. Eine wichtige Komponente, das Fehlermanagement, wird zu Beginn behandelt. Abstrakt kann man Fehlermanagement wie in Abbildung Anhang A.3 formulieren [Siegl, 94].

Der gesamte Ablauf beginnt mit einem Ereignis, wobei ein Ereignis entweder durch



**Abbildung Anhang A.3:** Schematische Darstellung von Netzwerkmanagement. Alles beginnt mit einem Ereignis, dessen Wirkung zu erkennen ist und dessen Ursache es in der Folge zu erkennen gilt. Danach erfolgt eine Bewertung sowie eine Entscheidung und in letzter Konsequenz eine Maßnahme.

einen Fehler (also z.B. ein Server fällt aus) oder eine Verhaltensänderung eines oder mehrerer Benutzer erfolgen kann. Eine solche Veränderung könnte in etwa darin bestehen, daß ein Benutzer der bis dato immer einen A4 Nadeldrucker benutzt hat plötzlich wegen neuer Projektanforderungen eine A0 Tintenstrahl Farbdruker verwendet und dadurch anstatt weniger kByte auf einmal MB an Daten schickt, was natürlich in einem Engpaß an Bandbreite in der lokalen Domäne resultiert.

Jedes Ereignis erzeugt eine Wirkung. Jede Wirkung äußert sich wiederum durch Symptome und jedes Symptom hat eine (oder mehrere) Ursachen. Der Schluß von der Ursache zur Wirkung ist eindeutig, der Umkehrschluß mitunter nicht. Ein und dieselbe Wirkung kann also mehrere Ursachen haben. So könnte z.B. die Ursache „Server antwortet nicht auf ein Ping“ die Ursache „Server ist ausgefallen“ oder aber auch „Leitung zum Server ist defekt“ haben.

Es gilt nun in einem nächsten Schritt aufgrund der Wirkung eine Erkennung der Ursache zu versuchen. Diese kann entweder direkt (z.B. ein Benutzer ruft den Administrator an und sagt, die Leitung ist beschädigt worden) oder indirekt (z.B. durch entsprechende Überwachung des Systems, Statistiken etc.) erfolgen. Für die letztere Methode, die in der Praxis auch die häufigere ist, ist es besonders wichtig, charakteristische Kenngrößen des Netzes zu ermitteln und ständig zu messen. Eine Veränderung dieser Werte läßt dann einen Schluß auf einen Fehler und wenn die Kennwerte gut genug gewählt wurden, auch auf dessen Ursache zu. Es empfiehlt sich, hier eine Wissensdatenbank anzulegen, um häufig wiederkehrende Wirkungen schneller und präziser einordnen zu können.

Nach der Erkennung des Problems erfolgt eine Bewertung desselben. Es muß nun aufgrund der Anforderungen<sup>1</sup> festgestellt werden, ob das aufgetretene Ereignis den regulären Betrieb des Netzwerkes stört. Auch hierfür empfiehlt es sich, eine Wissensdatenbank über alle bereits aufgetretenen Ereignisse und deren Bewertung zu Hilfe zu nehmen bzw. anzulegen.

Sobald eine Bewertung erfolgt ist, muß eine Entscheidung getroffen werden, wie nun auf das Problem reagiert wird. Dies wird unter Zuhilfenahme von verfügbaren Ressourcen und der nun schon bekannten Wissensdatenbank bewerkstelligt. Der letzte Schritt ist die Durchführung der aus dem Entscheidungsprozeß gewonnenen Maßnahme(n).

Der erste Unterabschnitt widmet sich nun zunächst einmal der niedrigsten Ebene im Netzwerkmanagement, der Protokollebene.

### **Internet Management Protokolle SNMP und CMIP**

Um die Kommunikation zwischen zu managenden Netzwerkkomponenten und dem

---

<sup>1</sup> darunter versteht man Bandbreite und Verfügbarkeit

Netzwerk Management System aufrechtzuerhalten, benötigt man ein Standard Management Protokoll. Deren gibt es gleich zwei, das Simple Network Management Protocol (SNMP) von der Internet Engineering Task Force (IETF) Untergruppe der Internet Society und das Common Management Information Protocol (CMIP) von der ISO.

SNMP gilt seit seiner Standardisierung durch die IETF im Jahre 1990 [SNMP, 90] als der de facto Standard für Netzwerkmanagement Kommunikation. SNMP setzt auf das TCP/IP Protokoll UDP (User Datagram Protocol) des Internets auf und ging ursprünglich aus dem Simple Gateway Monitoring Protocol hervor, das zur Performance Überwachung von IP Gateways erstellt wurde. Das Interesse bzw. die Applikationen die CMIP verwenden nehmen kontinuierlich ab, auf lange Sicht gesehen hat SNMP (von dem mittlerweile schon in einer 2. Version standardisiert wird) eindeutig die besseren Chancen [Muller, 93], deswegen wird hier das SNMP ausführlicher behandelt.

Das SNMP ist ein sehr flexibles Protokoll. Es erlaubt virtuell jede Art von Objekt zu managen, wobei ein Objekt Hardware, Software oder eine logische Verbindung sein kann. Objekte werden von den Herstellern beschrieben und in jeder zu managenden Komponente wird dies in der sogenannten MIB (Management Information Base) gespeichert. Die MIB kann nun mit Hilfe des SNMP abgefragt und sogar verändert werden. Das Netzwerk Management System kommuniziert mit sogenannten Agenten in den einzelnen Komponenten, welche wiederum auf die MIB zugreifen. Agenten sind keineswegs passiv, sie können also nicht nur auf Anfragen vom Netzwerk Management System antworten sondern auch selbst Nachrichten generieren. Man spricht in solch einem Falle von einer sogenannten „Trap.“ Sie wird auf Grund eines zu definierenden Ereignisses ausgelöst, z.B. könnte man, wenn die Gerätetemperatur einen gewissen Schwellwert überschreitet, eine Nachricht an den System Administrator schicken und eine automatische Abschaltung der Komponente vornehmen. Traps sind allerdings insofern gefährlich, als daß bei einem Ausfall des Elements keinerlei Nachrichten mehr generiert werden können und somit die Netzwerkmanagementzentrale nicht informiert werden kann. Ein Polling von der Zentrale ausgehend empfiehlt sich als das weitaus bessere Vorgehen, da hier in jedem Falle eine Aussage getroffen werden kann. Selbst wenn das Element ausgefallen ist kann man dies aufgrund eines „Time-outs“ noch feststellen.

Der erste MIB Standard definierte 110 Objekte, primär für Routing Variablen um heterogene Netzwerke zu verbinden. Der letzte MIB Standard, auch bekannt unter MIB-II, definiert 165 Hauptobjekte, wobei allerdings Typen und Anzahl der Objekte ständig im Wachstum begriffen sind [Muller, 93]. Eine große Anzahl von Händlern erweitert die Standard MIBs auch um eigene Funktionen, um ihre Produkte noch besser verwaltbar zu machen. Dies wirft allerdings ein Inkompatibilitätsproblem auf bzw. erschwert es implizit andere Werkzeuge für das

Netzwerkmanagement zu verwenden, als die vom Hersteller gelieferten.

Das 1988 vorgestellte und im Mai 1990 von der IETF standardisierte SNMP Protokoll weist einige entscheidende Schwächen auf. So fehlten z.B. Sicherheitsmechanismen und ein ereignisgesteuertes Polling zur Erkennung von Systemfehlern. Bereits 1992 wurde von der IETF ein Update auf SNMPv2 vorgeschlagen, das Dinge wie einen Sicherheitsstandard, Remote Konfiguration und auch administrative Basisfunktionen zur Verfügung stellen sollte. An Sicherheitsaspekten wurden Datenverschlüsselung, Autorisierung und die Integritätssicherung der Nachrichten berücksichtigt. Leider konnten sich die Arbeitsgruppen der IETF nicht auf einen formalen Standard reichen, und so wurde die Standardisierung im September 1995 abgebrochen und bis zum Zeitpunkt der Erstellung dieses Textes (April 1996) nicht wieder aufgenommen.

Kurz nachdem die offizielle Standardisierung gescheitert war, haben sich allerdings die größten Netzwerkhersteller (Bay Networks, BGS Systems, Cisco Systems, HP and SNMP Research) zusammengeschlossen und sich auf einen de facto Industriestandard geeinigt der SNMPv2\* genannt wurde. Das Ergebnis dieser Bemühungen wurde am 2. April 1996 in Las Vegas, Nevada demonstriert [SNMPv2\*, 96]- gleichzeitig erging auch ein offizieller Appell an die IETF ihre Arbeit wieder aufzunehmen. Ziel der Gruppe ist es, SNMPv2\* sowohl als einen vom Markt akzeptierten Industriestandard als aber auch als formalen IETF Standard einzuführen. SNMPv2\* ist eine Erweiterung der von der IETF zuletzt als RFC 1901 vorgeschlagenen SNMPv2c [SNMPv2c, 96] Spezifikation, die zwar administrative Funktionen beinhaltet, Sicherheitsaspekte aber gänzlich vernachlässigt. SNMPv2\* unterstützt sowohl administrative Funktionen als auch Sicherheitsaspekte und Remote Konfiguration. Weiters wird eine Abwärtskompatibilität gegenüber bestehenden SNMP basierenden Systemen gewährleistet, um eine Einführung bzw. Umstellung so kostengünstig als möglich zu gestalten.

Das von der ISO standardisierte **CMIP Protokoll** verwendet ebenfalls einen objektorientierten Ansatz, allerdings ist die Anzahl der Objekte genau festgelegt und auch nicht erweiterbar. Die Objekte repräsentieren Netzwerkgeräte, also Router, Gateways, Workstations, etc. und sind auf OSI Netzwerke zugeschnitten. Die CMIP Datenbank definiert weiters auch Relationen zwischen den einzelnen Geräten, was über die Fähigkeiten des SNMP Protokoll hinausgeht.

Prinzipiell unterscheidet sich CMIP von SNMP in zwei Dingen: erstens verwendet SNMP nur vier Schichten des OSI Modells, da es auf dem TCP/IP Protokoll aufsetzt - CMIP hingegen braucht in jedem Falle einen sieben Schichten Protokoll Stapel. Dies resultiert natürlich in einem erhöhtem Speicherbedarf in den

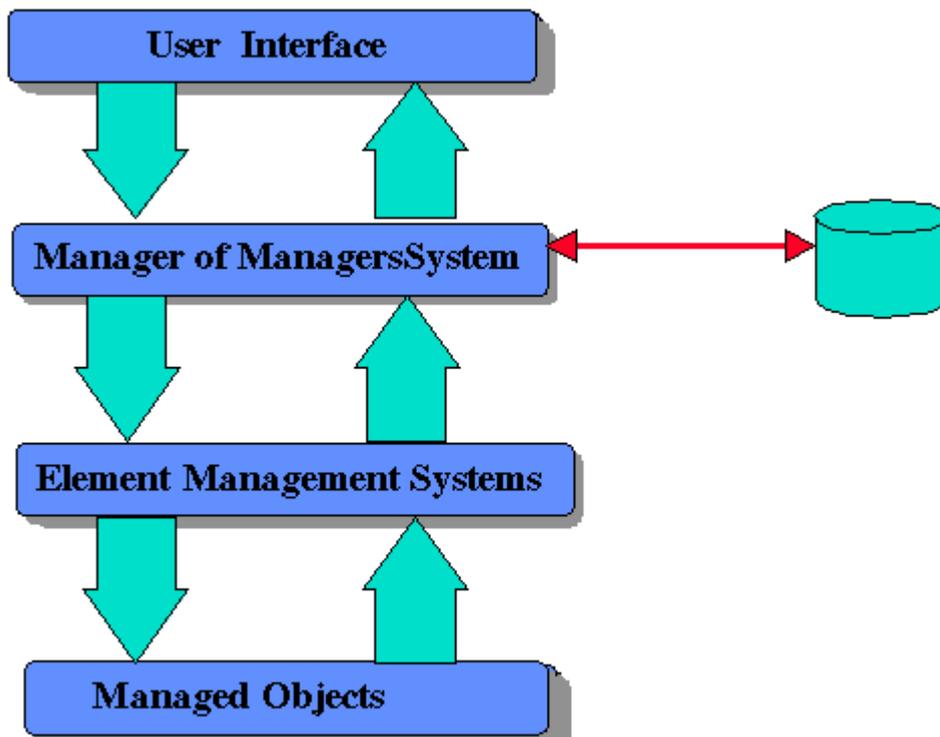
---

<sup>1</sup> ein typischer SNMP Klient benötigt ca. 20kB an Speicher, ein typischer CMIP Klient hingegen an die 400kB

Netzwerkkomponenten, der jedoch in den meisten Fällen nicht verfügbar ist. Zweitens verwendet SNMP das UDP Protokoll, was einen verbindungslosen Datenverkehr zuläßt. CMIP hingegen basiert auf einem verbindungsorientierten Konzept. Das bedeutet, es muß zu jedem zu verwaltenden Gerät im Netzwerk eine Verbindung aufgebaut werden, die dann auch fortwährend bestehen bleiben muß. Dies stellt offensichtlich höhere Anforderungen an das zugrundeliegende Netzwerk, da die zusätzlich benötigte Bandbreite ja erst einmal zur Verfügung gestellt werden muß bzw. müssen auch die einzelnen Agenten in den Geräten intelligenter reagieren und eine höhere Funktionalität aufweisen.

Um den CMIP Standard jedoch dennoch mit vernünftigen Speicheranforderungen zu implementieren, haben IBM und die 3Com Corporation eine abgespeckte Version von CMIP mit dem Namen HLM (Heterogeneous LAN Management) entwickelt. HLM ist zwar kompatibel mit den 7 Schichten OSI Anforderungen, bei der Implementierung wurde allerdings der Hauptaugenmerk auf die untersten beiden Schichten (physischer und Daten Link Schicht) gelegt, sodaß man mit Hilfe von HLM Software Module erzeugen kann, die weniger als 30kB an Speicher benötigen. Weiters sei erwähnt, daß einige Hersteller ihre Management Systeme sowohl auf SNMP als auch auf CMIP ausgelegt haben. Als Beispiel sei HP-OpenView genannt, welches fähig ist, SNMP Nachrichten in CMIP Protokoll kompatibel zu wandeln und diese an andere OSI kompatible Netzwerke weiterzuschicken.

Zusammenfassend kann man feststellen, daß CMIP für den OSI Netzwerkstandard entwickelt wurde und daher in seiner Flexibilität (da auf OSI Objekte beschränkt) sehr eingeschränkt ist. Auch der Speicherbedarf bei den Netzwerkkomponenten ist sehr hoch, da alle 7 OSI Schichten unterstützt werden. Der SNMP Standard der IETF ist hingegen offen bezüglich der zu definierenden Objekte, wobei ein Objekt Software, Hardware, eine logische Verbindung oder auch sogar OSI Objekte repräsentieren können. SNMP setzt auf dem verbindungslosen UDP Protokoll auf und benötigt auf der Geräteseite nur einen sehr geringen Speicherbedarf, da nur 4 OSI Schichten benötigt werden. Mit SNMP kann der CMIP Standard quasi inkludiert werden, deswegen ist dem SNMP Protokoll Standard, der in der Version 2 auch Sicherheitsaspekte berücksichtigten wird, auf lange Sicht gesehen den Vorzug zu erteilen.



**Abbildung Anhang A.4:** Die vier Schichten eines Netzwerkmanagement Systems. In der untersten Schicht befinden sich die zu managenden Objekte, darauf aufsetzend findet man die Element Management Systeme, die gewisse Teile eines Netzwerkes verwalten. Die sogenannten Manager von Manage Systemen vereinigen nun die Informationen von den Element Management Systemen und stellen sie mittels eines Benutzer Interfaces dem Benutzer zur Verfügung. (Quelle: <URL: <http://netman.cit.buffalo.edu/Doc/DStevenson/>>)

### Management Architektur

Nachdem nun die Kommunikation zwischen Netzwerkmanagement Zentrum und den zu managenden Komponenten ausführlich behandelt wurde, widmet sich der folgende Abschnitt der eigentlichen Architektur eines Netzwerkmanagement Systems. Es wird ein vier Schichten-Modell vorgestellt, dessen Funktionalität und Aufgaben beschrieben werden.

Prinzipiell kann man ein Netzwerkmanagement System aus vier Schichten bestehend beschreiben [Stevenson2, 95]. Jede dieser Schichten stellt Funktionen zur Verfügung, die es letztendlich ermöglichen, die Geräte in einem Netzwerk zu verwalten. Diese Architektur ist in Abbildung Anhang A.4 dargestellt.

In der untersten Ebene befinden sich die zu verwaltenden Objekte (Managed Objects). Unter einem Objekt versteht man Workstations, Peripheriegeräte, Router ... bzw. alles was in irgendeiner Form Überwachung bzw. Management benötigt. Ein Objekt muß keineswegs Hardware sein, sondern kann auch Software (also eine

Applikation), eine logische Verbindung oder allgemein gesehen irgendeine Funktion innerhalb des Netzwerkes repräsentieren.

Die nächste Schicht, Element Management Systeme, verwaltet spezielle Teile eines großen Netzwerkes, so kann man z.B. den Sun Net Manager dazu verwenden, um SNMP Elemente im Netzwerk zu managen, HP-Open View um CMIP Geräte ansprechen zu können. Weiters können Element Manager asynchrone Verbindungen, Multiplexer aber auch einzelne Applikationen managen.

Manager von Management-Systemen integrieren die Element Management Systeme zu einer nach außen hin scheinenden heterogenen Gesamtheit. Dies inkludiert auch eine Korrelation zwischen den Alarmen der einzelnen Element Management Systeme. Die meisten Daten kommen jedoch von den zu managenden Objekten, das Manager Programm legt diese sodann in einer Datenbank ab, wo sie für weitere Konsolidierung bzw. Verarbeitung zur Verfügung stehen.

Die oberste und gleichzeitig wichtigste Schicht stellt das Benutzer Interface dar. Dessen Aufgabe besteht darin, die gesamte gesammelte Information<sup>1</sup> der Management Information System (MIS) Gruppe in geeigneter Form zur Verfügung zu stellen. Die geeignete Visualisierung der Daten ist ein entscheidendes Kriterium für ein gutes Netzwerkmanagement System, denn wenn es nicht gelingt, die Systemadministratoren über alle wichtigen Ereignisse im System ausreichend informiert zu halten, kann man in der Folge auch keine wichtigen Entscheidungen zur Fehlerbehebung, für das Tuning etc. treffen. Das Netzwerkmanagement System würde seine Wirkung vollkommen verfehlen.

In diesem Abschnitt wurde die prinzipielle Architektur eines Netzwerkmanagement Systems und deren Funktion beschrieben. Offen blieb die tatsächliche Funktionalität eines Management Programmes, um die gerade vorgestellten Terminologie zu verwenden: die Funktionalität des Managers der Management Systeme. Dies wird im nächsten Unterkapitel behandelt.

### **Management Funktionalität**

Das am weitesten verbreitete Modell für das Grundgerüst eines Netzwerk Management Systems ist das Open System Interconnection (OSI) „FCAPS“ Modell für MFAs (Management Functional Areas) der ISO (International Standardization Organisation). Das OSI Modell ist sehr allgemein und breitgefächert, nur wenige Netzwerkmanagement Systeme decken tatsächlich alle Bereiche ab - die meisten Implementationen beschränken sich auf Teilmengen (siehe dazu auch Anhang A, Abschnitt „Kommerzielle Netzwerkmanagement Systeme“). Der Gesamtkomplex des Managements läßt sich laut OSI Modell in folgende fünf Teilbereiche aufteilen:

---

<sup>1</sup> also Real Time Alarme, Trend Analysen, Statistiken usw.

- Fehler Management (**F**ault Management)
- Konfigurations Management (**C**onfiguration Management)
- Kostenmanagement (**A**ccounting)
- Performance Management
- Sicherheits Management

Dieses Modell deckt die wichtigsten Bereiche einer modernen Systemadministration ab, dennoch kann es natürlich nicht alle Bereiche berücksichtigen. Mögliche weitere Management Gebiete wären z.B.:

- Individuelle Kostenverrechnung
- Service (Systems) Management
- Wartungskosten Management

Die folgenden Abschnitte geben eine detailliertere Beschreibung der einzelnen Bereiche des OSI Modells.

### **(A) Fehler Management**

Unter Fehler Management versteht man im wesentlichen drei wichtige Schritte die im auch dem eingangs beschriebenen allgemeinen Netzwerkmanagement entsprechen. Es sind dies: Detektion des Fehlers, Isolation sowie Behebung desselbigen um wieder normale Betriebsbedingungen zu erhalten. Diese Funktionalität sollte von jeder Software, die sich selbst als Netzwerkmanagement System deklariert, erfüllt werden. Probleme sollten wenn möglich graphisch dargestellt werden, textorientierte Anzeigen werden rasch unübersichtlich. Die meisten Systeme bewerkstelligen das Fehlermanagement durch kontinuierliches Polling der zu verwaltenden Objekte, es sind jedoch auch sogenannte „Trap Mechanismen“ oder Logfile Scanning denkbar. Letzterer Mechanismus untersucht in regelmäßigen Abständen von Systemkomponenten angelegte Logfiles (z.B. den syslog File in UNIX Systemen) und generiert je nach eingestellten Filterbedingungen Alarme.

Alarme müssen nicht nur in fast Echtzeit (real-time) generiert sondern auch verarbeitet werden. Es ist daher sofort einsichtig, daß eine Meldung der Art „Disk cache connection broken because of Signal 12“ nicht sehr aussagekräftig ist sofern man nicht ein eingefleischter Profi für die Komponente ist, die diesen Fehler generiert hat. Ein Systemadministrator sollte zwar sehr wohl über ein entsprechendes Fachwissen bzw. Erfahrung verfügen, jedes Detail kann und soll er aber nicht wissen. Deswegen ist es entscheidend in welcher Form Alarme an den Systemadministrator gelangen. Da auf die Nachricht, wie eingangs erwähnt, auch

sehr rasch reagiert werden muß, sind folgende Informationen auf jeden Fall zu jeder Alarmnachricht (zusätzlich zum Namen und der Auftrittszeit) hinzuzufügen:

- Wen soll man verständigen?
- Handlungen die man sofort setzen muß
- Welches Fachpersonal muß man holen?
- Welche Firma muß man anrufen für den Service?

Hat der Administrator diese Informationen nicht sofort zur Hand, muß er wertvolle Zeit damit verbringen herauszufinden, womit er anfangen soll. Das ist nicht produktive Zeit, in der u.U. eine ganze Firma stillstehen kann und die Kosten können sich schnell in erschwindelnde Höhen begeben - und dies bei jedem Auftreten eines Fehlers. Selbstverständlich empfiehlt es sich auch, bereits bekannte Probleme in einer Wissensdatenbank zu speichern und bei auftretenden Problemen erst einmal einen Diagnoseprozeß zu starten. Dies kann den Techniker mit weiteren hilfreichen Zusatzinformationen versorgen, was wiederum eine raschere Behebung der Störung ermöglicht.

Jede Störungsbehebung sollte auch eine sogenannte „trouble-ticket“ Generierung inkludieren. Ein „trouble-ticket“ beinhaltet Informationen über Wartungsmaßnahmen an den zu managenden Komponenten. Jede getätigte Maßnahme und auch von wem sie ausgeführt wurde und die dadurch entstandenen Kosten werden festgehalten. Diese Daten sind für ein effizientes Kostenmanagement eine unentbehrliche Hilfe - denn nur wenn man die tatsächlichen Wartungskosten kennt kann man daran arbeiten, sie zu erniedrigen. So kann man nun z.B. die Anzahl der reparierten Videokarten über gewisse Zeiträume beobachten und schließlich zum Schluß kommen, daß ein Austausch billiger kommt als jegliche weitere Reparatur. Ebenso kann es auf lange Sicht billiger sein, einen gesamten PC zu tauschen als nur die Platte.

### **(B) Konfigurations Management**

Dies ist neben dem Fehlermanagement einer der wichtigsten Teile eines guten Netzwerk Management Systems. Es gibt dem Administrator darüber Auskunft, welche Hardware Komponenten im Netz installiert sind und wo sie sich befinden. Weiters wird natürlich Veränderung, Wegnahme und Hinzunahme von Komponenten ermöglicht. Ein dynamisches Update der aktuellen Darstellung in vernünftigen Zeitintervallen, welche in der Datenbank des Netzwerk Management Systems gespeichert ist, ist ebenfalls notwendig.

### **(C) Kostenmanagement (Accounting)**

Nur sehr wenige Netzwerkmanagement Systeme unterstützen Accounting

Funktionen. Hierbei handelt es sich um ein umfangreiches Kostenmanagement aller Netzwerk Komponenten. Es werden Netzwerkutilisierung, Zugriffe von wem/wann/wielange sowie Auslastung der einzelnen Komponenten beobachtet. Die daraus entstehenden Statistiken und Reports sind für jeden System Manager eine unabdingbare Hilfe in Bezug auf Lokalisation von gegenwärtigen und zukünftigen Engpässen sowie Einblick in die Ressourcenauslastung.

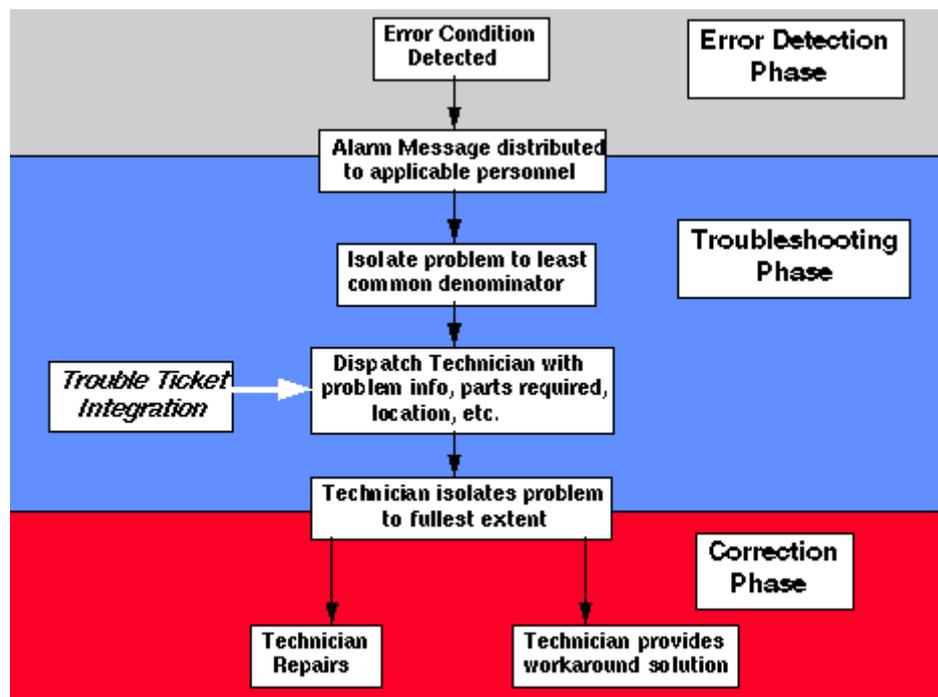
#### **(D) Performance Management**

Dies ist eine sehr oft gewünschte Eigenschaft eines Netzwerk Management Systems. Besonders im LAN Bereich ist es jedoch sehr schwer die Performance zu überwachen, da dies ja nur über eine Paket Analyse (Packet Analyzer) erfolgen kann. Hierzu ist wiederum die Installation von Remote Monitoring (RMON) MIBs erforderlich, was aber auch einen zusätzlichen Overhead bzw. natürlich eine Leistungsverminderung des Netzwerkes mit sich bringt. Es ist an dieser Stelle sorgfältig zu prüfen welchen Nutzen man zu welchen Kosten bekommt.

Das Performance Management sollte auch Auskunft über Verkehrslast, Antwortzeitverhalten und Netzwerkverfügbarkeit geben. Antwortzeitverhalten sollten natürlich auch für einzelne Netzsegmente eruiert sein, die Generation von aussagekräftigen Statistiken und Diagramme ist unabdingbar. Aus der Verfügbarkeit lassen sich die überaus wichtigen Werte MTBF (Mean Time Between Failure), durchschnittliche Funktionszeit in einem definierten Zeitintervall sowie Funktionszeit in Stunden berechnen [Stevenson2, 95].

#### **(E) Sicherheits Management**

Die meisten Netzwerkmanagement Systeme unterstützen nur Sicherheitsaspekte soweit sie typische Netzwerkkomponenten (Bridges, Router, Gateways, ...) angeht. Dies wird üblicherweise mit Login und Paßwort erreicht. Einige Netzwerkmanagement Systeme unterstützen auch die Einbringung von externen Alarmen, die z.B. die Gebäudesicherheit (Feueralarm) betreffen. Systemsicherheit ist Angelegenheit der Systemadministration und nicht des Netzwerkmanagements, und sollte auch dort abgedeckt werden. So stellt z.B. UNIX eine ganze Reihe von Sicherheitsmechanismen standardmäßig bereit.



**Abbildung Anhang A.5:** Ein ideales Netzwerkmanagement System, entsprechend dem ISO FCAPS Modell. Von der Fehlererkennungsphase, über die Fehlereingrenzung bis hin zur Behebung sind alle wichtigen Schritte dargestellt. (Quelle: <URL: <http://netman.cit.buffalo.edu/Doc/DStevenson/>>)

Datensicherheit Management spielt im Zusammenhang mit Fernerkundungsdaten nur eine untergeordnete Rolle, vor allem wenn es sich um planetarische Daten handelt. Fernerkundungsdaten sind zumeist nur für besonders spezifische Gruppen interessant und somit für die meisten Leute unwichtig, von einer Verschlüsselungen von Bilddaten kann also in den meisten Fällen abgesehen werden. Eine Ausnahme bilden Daten von Spionagesatelliten. Die Sicherheitsaspekte können sich also in den meisten Fällen rein auf Transaktionsvorgänge zur Bezahlung von Bilddaten beschränken, was wiederum keinen Performance Verlust beim Laden und Darstellen der Bilder bereitet.

Innerhalb des Netzwerkmanagement Systems wäre eine Paßworthierarchie wünschenswert, falls es mehrere Systemadministratoren mit unterschiedlichen Rechten geben soll. So könnte man z.B. gewisse Datenbanken für alle lesbar machen, aber nur für den Super User beschreibbar.

### Schritte zur Anschaffung eines Netzwerkmanagement Systems

Die folgenden zwei Abschnitte geben eine schrittweise Anleitung, wie man ein Netzwerkmanagement System, das alle bisher besprochenen Funktionen unterstützt, schrittweise aufbauen kann. Das Gesamtsystem ist in Abbildung Anhang A.5 gezeigt.

### Schritt A: Aufbau einer Anforderungsliste

Der erste Schritt besteht in der Aufstellung einer Anforderungsliste, hierzu kann wie folgt systematisch vorgegangen werden:

- Erstellung einer Liste von Informationen, die man von jedem zu verwaltenden Element erhalten will. Jedes Informationsstück muß bis ins Detail beschrieben werden - dies betrifft nicht nur die Art der Information (also z.B. aktueller Wert oder Durchschnittswert) sondern auch den Datentyp der Information.
- Mit dieser Liste werden nun jene Organisationen aufgesucht, die für die Verwaltung des jeweiligen Elementes verantwortlich sind. Die jeweiligen Organisationen sollen nun entscheiden, welche Informationen für sie wichtig sind um eine effiziente Wartung durchführen zu können. Es sollten nur jene Informationen übrig bleiben, die auch wirklich eine Vereinfachung der Arbeit ermöglichen<sup>1</sup>
- Formulierung der Report Generierung für die jeweiligen Geräte.
  - ◆ Welche Elemente müssen Alarme (in Echtzeit) generieren?
    - Für diese Geräte müssen Auslösezeitpunkte definiert werden (z.B. pro Stunde wird dreimal ein Alarm generiert)
    - Definition von Prioritäten für die Alarme
    - Definition von Diagnoseprozessen, die automatisch bei bestimmten Alarmen ausgelöst werden, um dem Management Personal die Arbeit zu erleichtern
    - Definition von geeigneten Polling Intervallen - z.B. alle fünf Minuten
  - ◆ Welche Elemente sollen über längere Zeiträume (z.B. monatlich) einen Report liefern?
    - Überwachung der Verfügbarkeit dieser Geräte
    - Überwachung der Häufigkeit der Benutzung
  - ◆ Welche Komponenten dienen der Trendanalyse bzw. dem Performance Tuning?
    - Überlegung von Methoden um die von diesen Komponenten gelieferten Daten sinnvoll zu kombinieren und zweckmäßig zu verwenden

---

<sup>1</sup> z.B. keine verwirrenden Nachrichten

- Gespräch mit dem Firmenmanagement führen um sicherzustellen, daß alle zu managenden Gebiete berücksichtigt worden sind.
  - ◆ Hierzu sollte die Rolle und die Funktionalität des Netzwerkmanagement Systems erklärt werden
    - Erhöhung der Produktivität in den Filialen
    - Minimierung der mittleren Fehlerbehebungszeit
    - Das System stellt einen pro-aktiven Ansatz zur Lokalisierung und Behebung von auftretenden Fehlern zur Verfügung
    - Förderung des Informationsflusses in der gesamten Organisation
  - ◆ Sammlung aller Anforderungen an das Management System die für den wirtschaftlichen Teil der Firma von Relevanz sind
    - Diese Sammlung darf keineswegs auf SNMP verwaltbare Geräte reduziert werden - gibt es Geräte die nicht über die geforderte Intelligenz verfügen, so sollte darauf später zurückgekommen und eine entsprechende Aufrüstung beantragt werden.
- Implementierung der Anforderungen, wobei für jedes Objekt die gesammelten Anforderungen genauestens eingehalten werden müssen. Die einzelnen Implementierungen müssen ein gesamt integrierendes System ergeben.
- Nach der Implementierung müssen die einzelnen Organisationen, die für die Wartung der jeweiligen Komponenten verantwortlich sind, über die Inbetriebnahme verständigt werden.
- Nach der ersten Report Periode muß ein Vergleich der Ergebnisse mit den Anforderungen stattfinden
  - ◆ Unter Umständen einige Anforderungen verändern
  - ◆ Reports werden sich verändern, sobald die einzelnen Filialen mehr und mehr an Professionalität gewinnen (weniger ausführliche und zielführendere Meldungen sind nun ausreichend)

Mit dem Vorgehen nach obiger Liste kann in systematischer Art und Weise ein Netzwerkmanagement System aufgebaut werden, das dem OSI FCAPS Modell entspricht.

### **Schritt B: Checkliste für den MIS Manager**

Dieser Abschnitt zeigt typische Fragen auf, die sich der Hauptverantwortliche für den Aufbau eines funktionierenden Management Systems stellen muß.

**Wieviel wird das System kosten?**

Hier sollte man stets beachten, daß Geld meistens von Leuten kommt, die nicht genug technisches Verständnis haben um das gesamte System zu verstehen. Es liegt nun am Geschick und dem Verständnis des MIS Managers, den technisch nicht so versierten Geldgebern die Anschaffung von teurer Soft- und Hardware verständlich zu machen. Der MIS Manager muß also ein Mensch sein, der sowohl von der wirtschaftlichen Seite als auch von der technischen Seite etwas versteht.

Außerdem sollte man stets auch die sogenannten versteckten Kosten berücksichtigen, die sich durch die individuelle Einrichtung des Systems an den einzelnen Lokalisationen ergeben können. Jedes Netzwerkmanagement System muß speziell auf die Bedürfnisse zugeschnitten werden, dies verursacht im allgemeinen nicht zu vernachlässigende Zusatzkosten, die im Gesamtbudget entsprechend zu berücksichtigen sind.

**Ist das neue Management System in das bestehende System integrierbar und wird es dieses entscheidend verbessern?**

Man muß als Techniker stets darauf bedacht sein, ein komplett neues System nur wegen der neuen Technologie wegen anzuschaffen. Das neue System muß eng mit den wirtschaftlichen Komponenten der Firma kooperieren und die Leistungsfähigkeit dieser entscheidend steigern. Das beste Managementsystem hilft nichts, wenn es die Funktionalität eines bestehenden Systems völlig umwirft bzw. es so kompliziert zu handhaben ist, daß keiner damit sinnvoll arbeiten kann. Ein solches System ist von Anfang an zum Scheitern verurteilt.

**Ist das vorgeschlagene System ein Modul basierendes?**

Die Nachteile eines nichtmodularen Systems werden sofort offensichtlich, wenn man z.B. das System erweitern will. Hier ist man gezwungen eine neue Version des Gesamtsystems zu kaufen, was sich natürlich auch in den Kosten niederschlägt. Bei einem modularen System hingegen ist nur ein Modul auszutauschen. Weiters ist man bei einem Gesamtsystem auch an einen Hersteller gebunden, und diesem auf Gedeih und Verderb ausgeliefert. Geht der Hersteller etwa in Konkurs, ist man gezwungen früher oder später das gesamte System auszutauschen. Ein modernes, modulares Netzwerkmanagement System gestattet die Verwendung von Modulen die von beliebigen Herstellern stammen können. Beim Ausschauen von Netzwerkmanagement Systemen ist daher auf ein entsprechendes Standardinterface (z.B. HPOpenView bietet dies an) bzw. eine Erweiterungsmöglichkeit zu achten!

**Ist das entstehende System nur ein Element Managementsystem oder ein Integrierendes Element Managementsystem?**

Beim Kauf eines Systems muß man stets aufpassen, was man für sein Geld bekommt. Die meisten Anbieter verkaufen ihre Systeme als Manager von Element

Management Systemen, obwohl nur ein Teil des OSI FCAPS Modells unterstützt wird.

### **Was wird durch das System überwacht?**

Es empfiehlt sich einen strengen Vergleich zwischen den Fähigkeiten des vorgeschlagenen Systems und den informationsliefernden Komponenten der Firma anzustellen. Können jetzt nicht schon alle Komponenten integriert und überwacht werden, so wird es später auch kaum möglich sein.

### **Verbessert das vorgeschlagene System die Arbeit des bestehenden Support Personals oder fügt es im Gegenzug noch weiteres Personal hinzu?**

Bei der Anschaffung eines neuen Systems sollte man sorgsam darauf bedacht sein, nicht noch weiteres Personal zu benötigen und die Arbeit des bestehenden Personals in keiner Weise zu vereinfachen. Solche Leute sind meist hochspezialisierte Techniker, deren Gehälter sich entscheidend auf das Firmenbudget auswirken können. Eine geeignete Vorgehensweise wäre etwa: zuerst für die gesamte Firma zu beantworten was soll das neue System können und was funktioniert jetzt schon. Dies sollte dann im Detail für jede einzelne Abteilung/Filiale erarbeitet werden.

### **Kommerzielle Netzwerkmanagement Systeme**

Es wäre angesichts der raschen Entwicklung auf dem Soft- und Hardware Sektor sinnlos, an dieser Stelle existierende Netzwerkmanagement Systeme im Detail vorzustellen, da ständig neue Systeme entstehen bzw. neue Funktionalität hinzugefügt wird. Eine Beschreibung hätte also nur kurze Gültigkeit. Um einen generellen Überblick zu bekommen sei auf die Literatur [Muller, 93] verwiesen. Hier werden die wichtigsten kommerziellen Netzwerkmanagement Systeme allgemein beschrieben.

Für aktuellste Informationen sei auf das für unsere Zeit bezeichnende Medium World Wide Web (WWW) hingewiesen, wo Anbieter aus aller Welt stets die neuesten Informationen über ihre Produkte bekanntgeben. Die derzeit größten Systeme sind (die Referenzen enthalten die jeweilige URL für das WWW): Netview (IBM) [Netview, 96] und die NetView Association (DEC und IBM) [NetAssoc, 96], HP-OpenView [OpenView, 96], Solstice Enterprise Manager (Sun Microsystems Inc.) [Solstice, 96], Tivoli [Tivoli, 96] und Cabletron Spectrum Network [Cabletron, 96].

## **Internet Grundlagen**

Das Internet ist für das GDSS als globales Verbindungsglied eine wichtige Komponente. Dieser Abschnitt fasst wichtiges Grundlagenwissen sowie Hintergrundinformationen zu diesem Thema zusammen.

## Historisches

Wie so viele wichtige Technologien hat auch das Internet einen militärischen Ursprung. Im Jahre 1967 kam ein „Request for Proposal“ von der amerikanischen Defense Advanced Research Projects Agency (DARPA) an die University of California Los Angeles (UCLA) mit dem Inhalt, ein geeignetes Protokoll für ein paketorientiertes Netzwerk zu spezifizieren - ein solches Netzwerk wurde auch parallel von der DARPA entwickelt (Projektname ARPANET [Adoba, 93]).

Etliche Institute am UCLA waren interessiert, gemeinsam wurde ein Proposal an das DARPA erstellt, mit dem Thema ein „Network Measurement Center“ für das ARPANET Projekt zu entwickeln. Somit wurde also ein Set von Werkzeugen programmiert, um das entstehende ARPANET zu observieren<sup>1</sup>.

Im Jahre 1969 wurde von BBN [BBN, 96] ein Interface Message Processor (IMP) entwickelt und an das UCLA geliefert. Einem ersten Feldversuch des ARPANETS stand nichts mehr im Wege. Das UCLA wurde mit zwei weiteren Universitäten verbunden, dem Stanford Research Institute (SRI) und der University of California Santa Barbara (UCSB) - als vierter Knoten war noch die University of Utah in Salt Lake City beteiligt. Die Verbindungen waren 50kbit/sec Links. Obwohl die Demonstration einwandfrei ablief, gab es dennoch große Probleme. Vor allem wenn das Netzwerk stark belastet wurde, gab es immer wieder Schwierigkeiten, den Speicher für das Zusammensetzen der Nachrichten zu organisieren.

1972 waren bereits 19 Knoten im ARPANET. Die Entwicklungen gingen aber nur mehr schleppend voran, da die heterogene Struktur des Netzwerkes einen großen Softwareentwicklungsaufwand hervorrief. Für unzählige Plattformen galt es Interface Software zu schreiben. Im Oktober 1972 fand eine Konferenz über Computer Kommunikation in Washington - BBN plante eine Demonstration des ARPANET über die gesamte USA. Die Demonstration war ein durchschlagender Erfolg, eine große Anzahl von wichtigen Persönlichkeiten und Firmen aus aller Welt bekundeten ihr Interesse und stimmten einer Entwicklungsunterstützung zu [Adoba, 93].

Die Geburt des Internet begann mit den ersten Arbeiten am Transmission Control Protocol (TCP). Eine erste Draft Version wurde 1973 an der University of Sussex vorgestellt. 1974 programmierte Richard Karp<sup>2</sup> den ersten TCP Code. Schon in den ersten Entwicklungsphasen wurden nicht nur das ARPANET sondern auch „packet

---

<sup>1</sup> Das Entwicklerteam bestand aus Stephen Cocker (machte zwischen 1967 und 1972 seine graduate work am UCLA), Jon Postel (war verantwortlich für die Zusammenstellung der Request for Comments (RFC)), Robert Braden (war ein Mitarbeiter vom Computer Zentrum des UCLA), Michael Wingfield (baute das erste Interface vom Internet zu einem Xerox Data Systems Sigma 7 Computer) und David Crocker (wurde später eine der zentralen Figuren für die Entwicklung des E-mail Standards im ARPANET).

<sup>2</sup> Richard Karp wurde später Präsident von ISDN technologies in Paolo Alto

radio“ und Satellitenkommunikation integriert. 1977 gab es die erste große Demonstration des neuen Internets. Es handelte sich um eine Art Schlachtfeldsimulation, die für das amerikanische DoD inszeniert wurde. Insgesamt wurden 94.000 Meilen an Distanz überwunden, wobei auch Satellitenlinks darunter waren. Es ging nicht ein Bit an Information verloren.

1981 wurde das TCP/IP Protokoll das erste Mal Bestandteil von Unix: 3Com [3Com, 96] implementierte UNET. Ab 1985 begann das Internet dann exponentiell zu wachsen. Zu diesem Zeitpunkt hatte nämlich das DARPA finanzielle Mittel in BBN investiert, um eine Implementierung des TCP/IP im UNIX BSD 4.2 zu verwirklichen. Sobald dies geschehen war, begannen große Computerhändler wie z.B. SUN Micro Systems [SUN, 96] UNIX BSD als Basis ihrer kommerziellen Produkte zu vertreiben. Dem Siegeszug des Internets stand nun nichts mehr im Wege.

### **Internetprotokolle**

Das Internet verfügt über eine sehr große Anzahl von Protokollen, die in dieser Arbeit nicht im Detail behandelt werden können. Hierzu sei auf die Literatur [Hahn, 94] verwiesen. Hier werden nur die wichtigsten Protokolle des Internet, das Internet Protokoll (IP), das Transmission Control Protocol (TCP) sowie das User Datagram Protocol (UDP) kurz vorgestellt.

Ursprünglich gab es nur ein einziges Protokoll, das TCP. Es sorgte für eine fehlerfreie End-zu-End Verbindung. Schon bald bemerkte man aber, daß dies nicht immer von Vorteil war - da der Overhead kein geringer war. Das nochmalige Senden der Pakete machte sich etwa auch beim Senden von Sprache unangenehm bemerkbar. Gerade bei Audio Anwendungen macht es Sinn, wenn man ein fehlerhaftes Paket einfach ignoriert und dafür ein kurzes Rauschen in Kauf nimmt, als eine zeitaufwendige Neusendung des Paktes zu veranlassen und dadurch die Stimme zu verzerren.

Dies war der Grund, warum das TCP Protokoll in zwei getrennte Protokolle geteilt wurde: das IP auf OSI Schicht 3 und das TCP auf OSI Schicht 4. Die Aufgabe des IP besteht einzig und alleine darin, eine geeignete Route für das Paket durch das Netzwerk zum Endpunkt zu finden, eine fehlerfreie End-zu-End Verbindung wird nicht garantiert. Das TCP sorgt für die Aufteilung in entsprechende Datenpakete, dem Ordnen und Zusammensetzen selbiger am Ende der Leitung, sowie für das Wiedersenden von verlorengegangenen oder fehlerhaften Paketen. Zusätzlich wurde für Dienste, die eine schnelle Verbindung aber nur geringe Datensicherheit benötigen (z.B. Audio Anwendungen) das UDP geschaffen, welches eine abgespeckte Version des TCP verkörpert. Das UDP garantiert keine fehlerfreie End-zu-End Verbindung. Fehlerhafte oder verlorengegangene Datenpakete werden nicht nochmals gesendet.

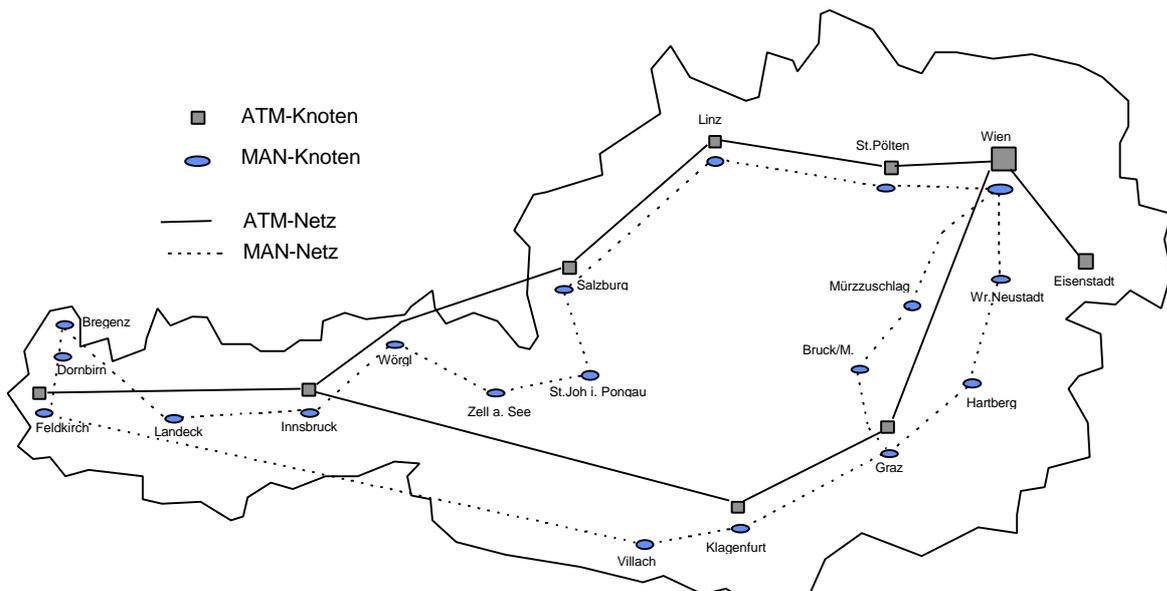
Hauptziel des IP Protokolls war die Portierbarkeit auf alle existierenden Plattformen und Netzwerktechnologien [Adoba, 93], sei es nun Frame Relay [McDysan, 94], ISDN [Kegel, 96], X.25 [McDysan, 94], FDDI [Albert, 94], Ethernet [Tanenbaum, 90], IBM-Token Ring [Tanenbaum, 90], etc. oder auch ATM. Dieses Ziel wurde auf jeden Fall erreicht, deswegen ist das TCP/IP Protokoll auch noch in absehbarer Zukunft das wichtigste Verbindungsglied für heterogene Plattformen auf der ganzen Welt und natürlich auch für ein globales System wie das GDSS von entscheidender Bedeutung. Unabhängig von der zugrundegelegten Netzwerktechnologie, wobei die Konzentration in dieser Arbeit auf ATM gelegt wird, kann das darauf aufgesetzte Protokoll immer TCP/IP heißen. Für eine detailliertere Behandlung der Protokolle sei auf eine äußerst gute Beschreibung der Internet Protokolle in [Hedrick, 87] verwiesen bzw. auf den RFC 793 (Request for Comment) der Internet Engineering Task Force (IETF) [RFC793, 81], der die Transmission Control Protocol Spezifikation in allen technischen Details beschreibt.

### **Die Vernetzung von Österreich und Europa**

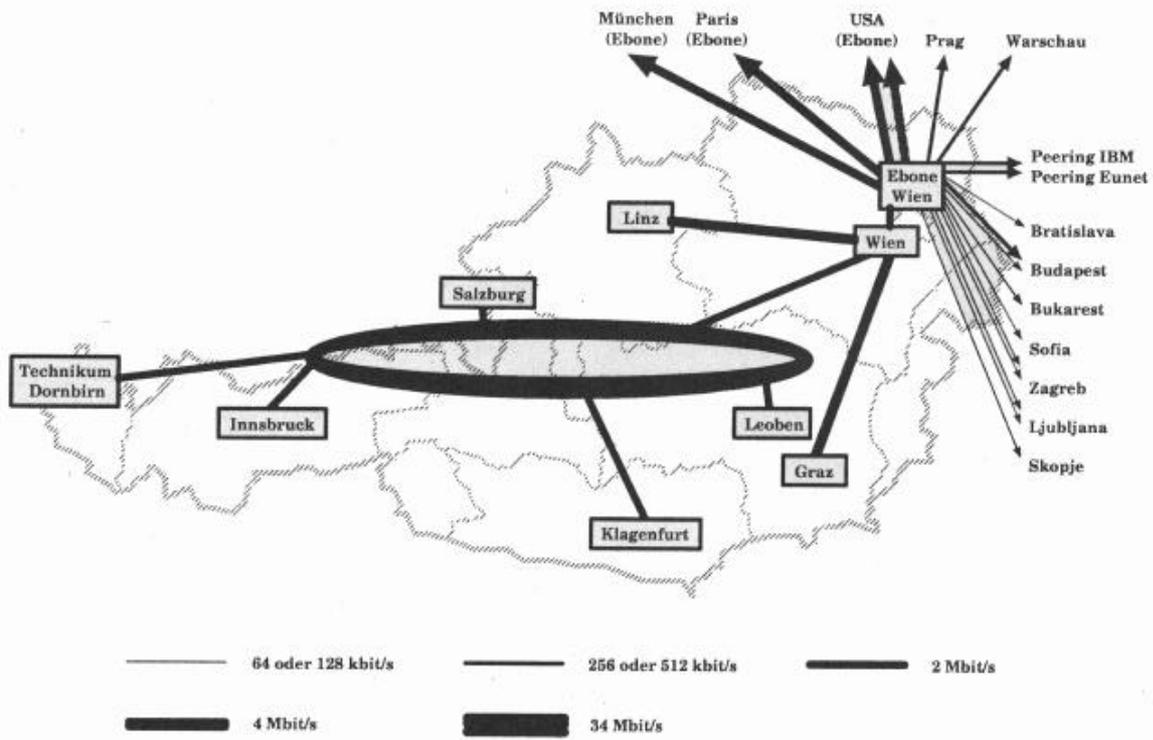
Die folgenden zwei Abschnitte geben Aufschluß über die Netzwerkinfrastruktur von Österreich und Europa. In Österreich ist das ACONET (Academical Computer NETwork) dominierend, für Europa und den Anschluß an Amerika ist es der EBONE. Zieht man ein österreichweites oder ein europaweites Szenario von GDSS in Betracht, so ist es natürlich von Bedeutung, die prinzipielle Infrastruktur und die wichtigsten Verbindungen zu kennen.

#### **(A) ACONET (Academical Computer NETwork)**

Die Entwicklung einer geeigneten Kommunikationsinfrastruktur für den österreichweiten Wissenschafts- und Forschungsbereich wurde bereits 1981 begonnen. 1986 wurde dann der ACONET Verein gegründet, dem die EDV-Zentren aller österreichischen Universitäten beitraten. In der zweiten Hälfte des Jahres 1990 wurde das ACONet schließlich errichtet. In einer ersten Stufe handelte es sich hierbei um ein privates X.25 Netz, welches die Universitäten in Graz, Linz, Leoben, Wien, Klagenfurt, Salzburg und Innsbruck mit einer Ringtopologie verband. Zunächst handelte es sich um 9 kbit/sec Links, später wurden die Links mit 64 kbit/sec betrieben. Im Jahre 1990 erhielt die UNI Wien einen Anschluß an das Internet durch die Firma IBM. Dies erfolgte mit einer 64 kbit/sec Standleitung nach



**Abbildung Anhang A.6:** MAN und ATM Netz der PTA in Österreich. Alle Landeshauptstädte und andere wichtige Städte sind am MAN, der mit 34 Mbit/sec (E3) betrieben wird. Das ATM Netz, welches ab Februar 1997 in Betrieb genommen wird, hat 155 Mbit/sec (OC-3) als Basisbandbreite.



**Abbildung Anhang A.7:** Das österreichische ACONet. Die Abbildung zeigt den E3 (34 Mbit/sec) Distributed Queue Dual Bus der ÖPTV. Die Universitäten haben einen 2 Mbit/sec Schlitz, wobei Graz und Linz eine Ausnahme darstellen. Beide Städte hängen seit März 1996 direkt mit einem 4 Mbit/sec ATM Link an Wien. Von Wien gibt es zahlreiche Verbindungen in die Nachbarländer über den E-Bone Anschluß.

Genf, von wo aus ein 1.5 Mbit/sec Link in die USA existierte. Im Jahre 1992 wurde das X.25 Netz auf das IP Protokoll umgestellt. Außerdem wurde die Übertragungsgeschwindigkeit von 64 auf 128 kbit/sec erhöht.

Basis des heutigen ACONets ist der Hochgeschwindigkeits Ring (MAN) der PTA (Post und Telekom, Austria), der mit Distributed Queue Dual Bus (DQDB) betrieben wird und mit E3 (34 Mbit/sec) fährt. Ab Februar 1997 wird parallel zum DQDB auch ein ATM Netzwerk in Betrieb genommen, welches einen österreichweiten Ring mit OC-3 (155 Mbit/sec) bilden wird [Raffer, 96]. Das gesamte Netz der PTA mit MAN und ATM Knoten ist in Abbildung Anhang A.6 dargestellt.

Die Universitäten in Wien, Salzburg, Innsbruck, Leoben und Klagenfurt sowie das Technikum in Dornbirn sind über 2 Mbit/sec Links und dem SMDS (Switched Multimegabit Data Service) an den Ring angebunden [Kunft, 95], die Universitäten Graz und Linz sind jeweils über einen 4 Mbit/sec ATM Link direkt an Wien angeschlossen. Private Kunden können jederzeit Bandbreite von der PTA kaufen. Der heutige Stand<sup>1</sup> ist in Abbildung Anhang A.7 zu sehen.

## **(B) E-BONE**

Der E-Bone (European Backbone) wurde bei einem Treffen der RARE in Amsterdam (Réseaux Associés pour la Recherche Européenne oder Association of European Research Networks) am 2. September 1991 ins Leben gerufen. In diesem Monat trafen sich Betreiber von europäischen akademischen Forschungsnetzen, um die längst ausstehenden Probleme einer gesamteuropäischen Verbindung der bisher abgekapselten Netze zu verwirklichen.

Im Jänner 1992 wurde dieser Versuch formell gestartet, folgende Spezifikationen wurden gemacht (nähere Details sind [EBONE, 92] zu entnehmen):

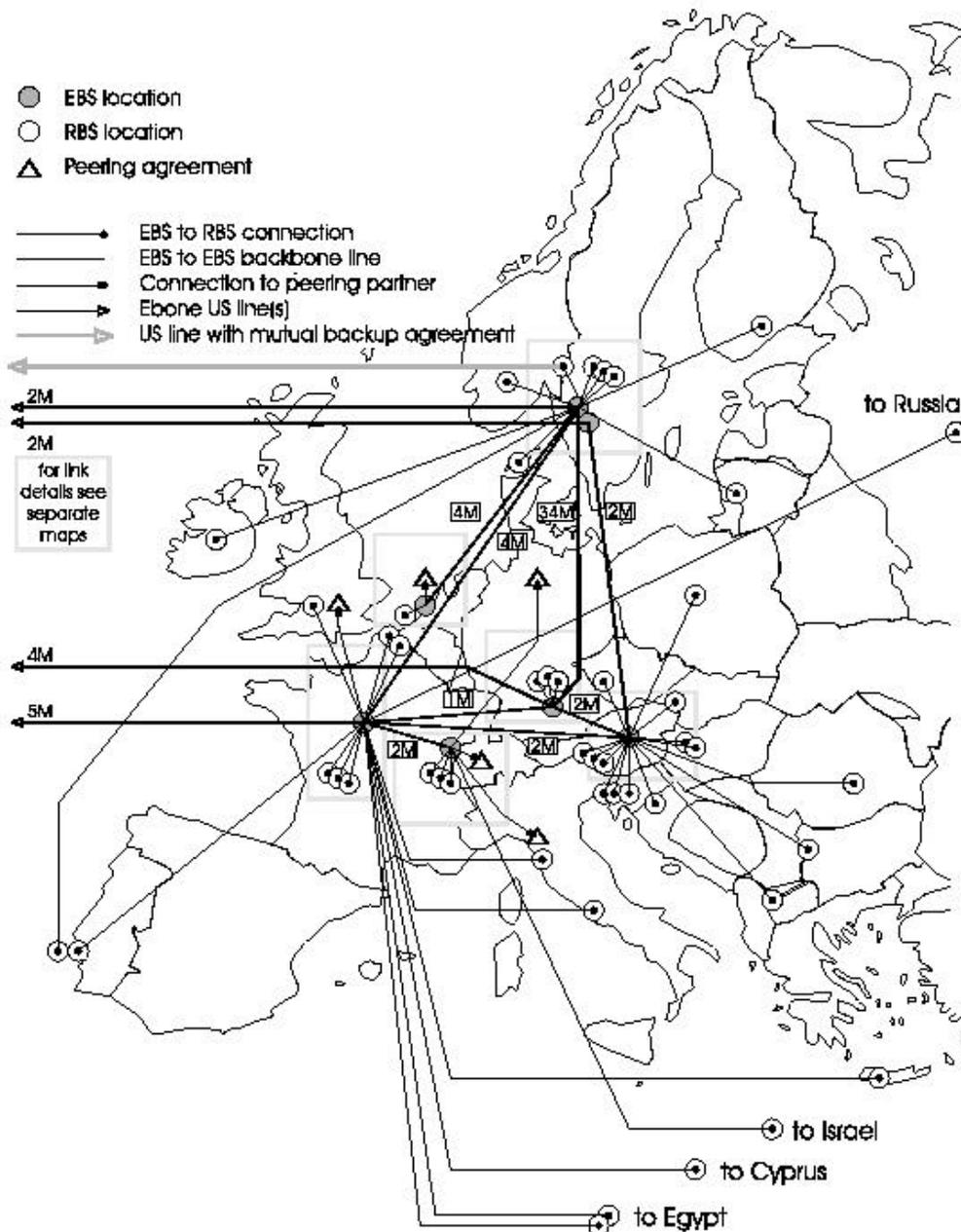
- OSI Layer 1 + 2:
  - Ethernet
  - FDDI (Dual Attachment/Doppelring [Albert, 94])
  - V.35 oder HSSI mit PPP (RFC 1171 [RFC1171, 90]) oder Frame Relay (CCITT I.122)
- Routing:
  - IP mit BGP (RFC 1163 [RFC1163, 90]) oder static routes
  - CLNS (Connection Less mode Network Service) ISO CD 10747
- Management:
  - SNMP (RFC 1157 [SNMP, 90])

---

<sup>1</sup> Juni 1996

## Ebone Configuration

per May 31, 1996



Map maintained by Woerber@CC.UniVienna.ac.at

**Abbildung Anhang A.8:** Die Topologie des E-Bones mit Stand Mai 1996. Man sieht die interkontinentalen Verbindungen in die USA mit derzeit insgesamt 13 Mbit/sec sowie die internationalen Links zwischen allen wichtigen europäischen Städten, welche mit wenigstens 256 kbit/sec ausgeführt sind (Quelle: ftp://ftp.univie.ac.at).

Weiters wurde ein E-Bone Operation Team (EOT) gebildet, das die Verfügbarkeit des Backbones gewährleistete sowie eine E-Bone Administration Uni (EAU), die Kosten und Administrierung übernahm. Das Operationszentrum ist in Stockholm am Royal Institute of Technology, das Netzwerk Management wird vom E-Bone Management Komitee übernommen, das sich aus Vertretern einiger ausgewählter Organisationen zusammensetzt. Derzeit verfügt der E-Bone über 20 Mitglieder, wobei ACONet Austria natürlich auch vertreten ist.

Die Basis Topologie des E-Bone war eine 4 Knoten Diamantstruktur, wobei 2 Links in die USA geplant waren. Die Links sollten wenigstens 512 kbit/sec haben, einige hatten zu Beginn allerdings nur 64 kbit/sec bzw. 256 kbit/sec. Seit 1992 hat sich diese Topologie entscheidend verändert bzw. sind viele neue Links mit höheren Kapazitäten angefügt worden. Eine aktuelle Übersichtskarte (Stand Mai 1996) des E-Bones ist Abbildung Anhang A.8 zu entnehmen.

### **Internetstatistiken**

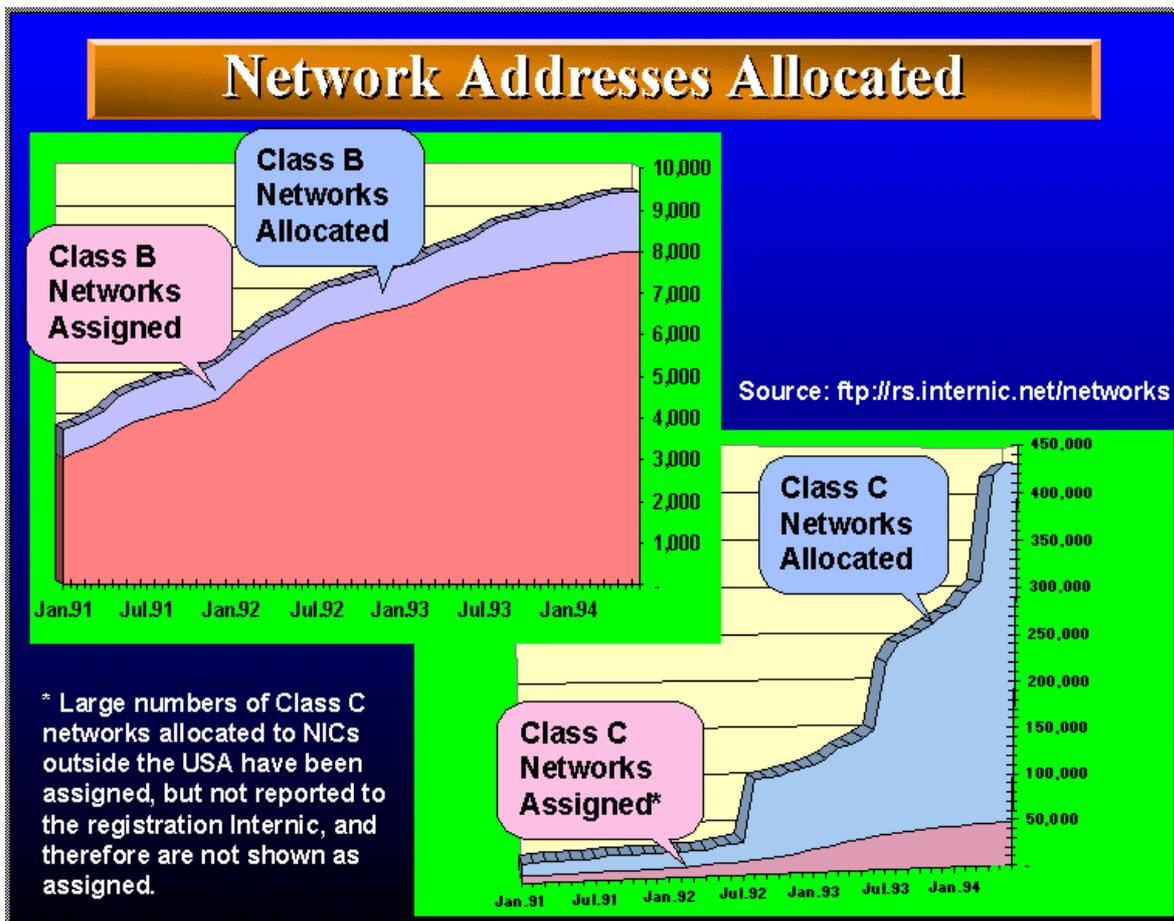
Wir berichten über einige statistische Untersuchungen die über das „Netz der Netze“ gemacht wurden. Es soll hier vor allem das exponentielle Wachstum des Netzes in den letzten Jahren gezeigt werden. Letzten Endes sollte mit Hilfe der Statistiken<sup>1</sup> verdeutlicht werden, daß am Design eines globalen Datenaustausch und Serversystems wie dem GDSS kein Weg am TCP/IP vorbeiführen kann.

Die erste Statistik gibt Auskunft darüber, wieviele Netzwerke der Klassen B (2 Byte Netzwerk, 2 Byte Host) und C (3 Byte Netzwerk, 1 Byte Host) seit Jänner 1994 registriert bzw. tatsächlich alloziert worden sind. Registrierte Domänen sind in der Datenbank von InterNIC [Internic, 96] zu finden, so natürlich auch die Domäne des ACONets. Betrachtet man Abbildung Anhang A.9 so ist die Kurve von Klasse B Netzwerken noch einigermaßen linear, die Kurve von Klasse C Netzwerken hingegen ist eindeutig exponentiell.

In Abbildung Anhang A.10 kann man das Netzwerkwachstum in Europa sehen. Die teils hohen Zuwachsraten lassen sich meist mit den sehr guten Infrastrukturen der Länder erklären. Die Statistik zeigt z.B., daß sich in Frankreich die Zahl der Netzwerke vom April 93 bis zum Juli 94 fast verfünffacht hat.

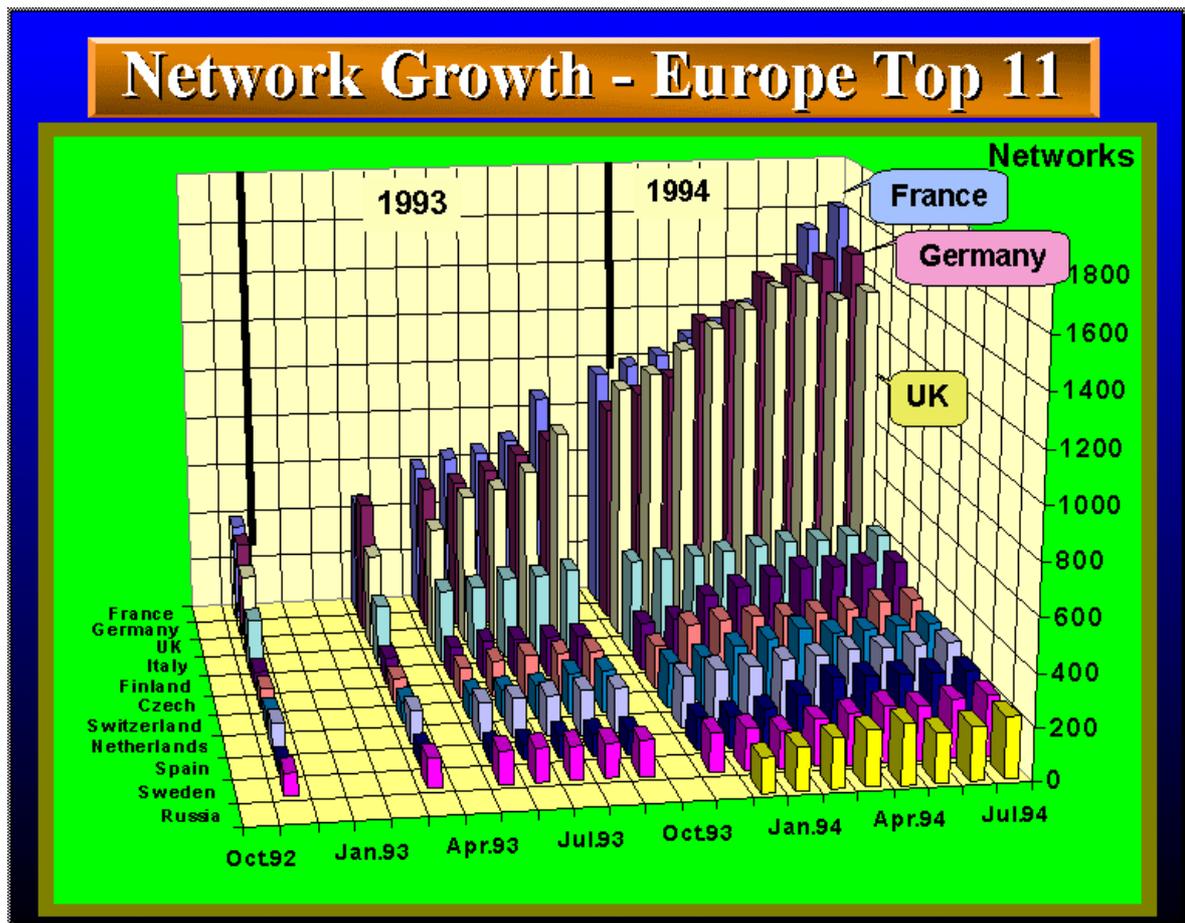
---

<sup>1</sup> Alle Graphiken wurden vom ftp Server der Internet Society entnommen. Updates kann man jederzeit dort herunterladen. Die ftp Adresse ist: <ftp://ftp.isoc.org/charts/networks-gifs/>. Alle Graphiken können frei verwendet werden, solange die © Anmerkung nicht beseitigt wird.

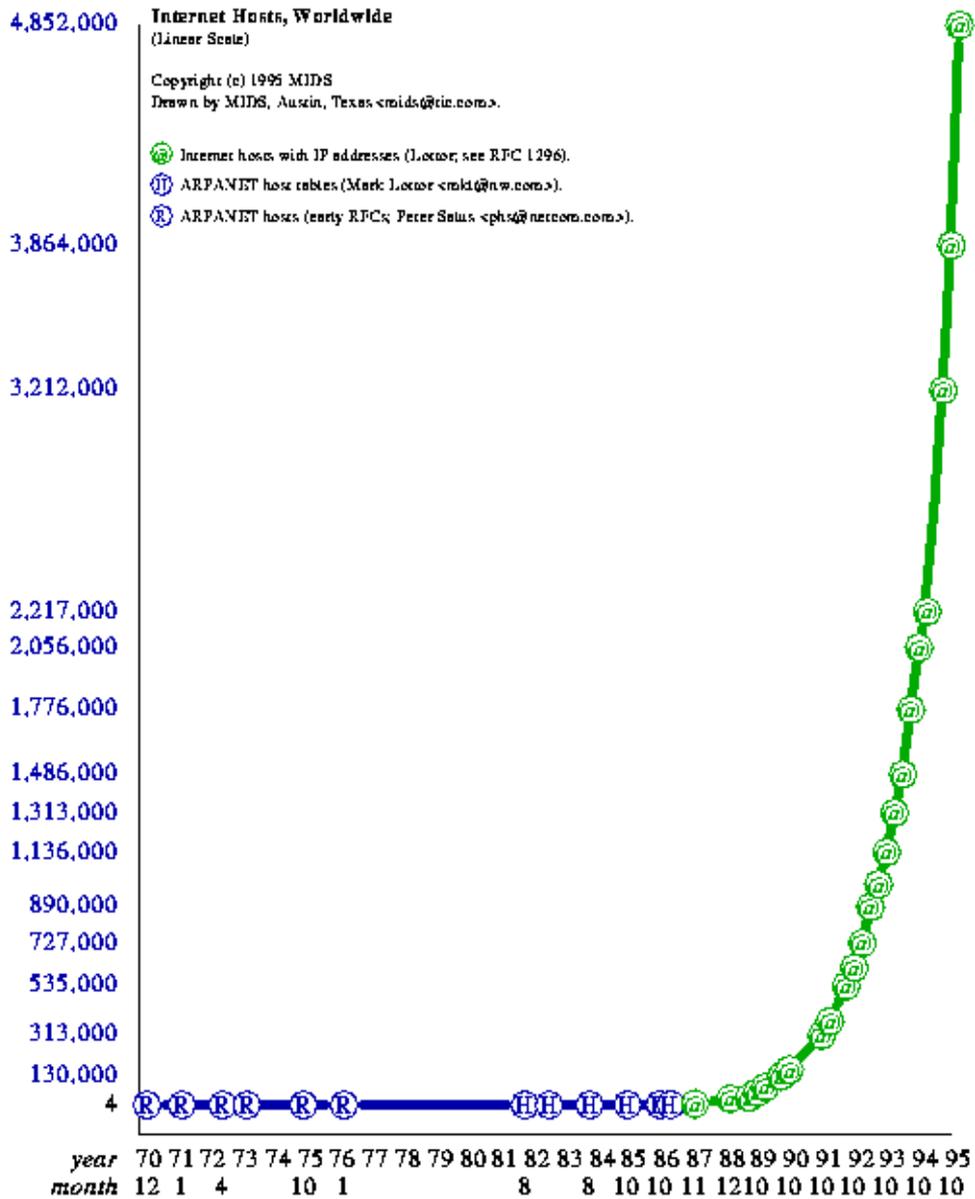


**Abbildung Anhang A.9:** Wachstum der Klasse B und C Netzwerke. Die beiden Graphiken geben Auskunft darüber, wieviele Netzwerke der B und C Klasse von Jänner 1991 bis Jänner 1994 zugewiesen (assigned) bzw. alloziert (allocated) wurde. Zugewiesene Netzwerke sind auch tatsächlich bei Internic registriert worden. (Quelle: <ftp://ftp.isoc.org/charts/networks-gifs/>)

Derzeit verfügt das Internet weltweit über mehr als 35.000 Netzwerke mit insgesamt fast 5.000.000 Hosts. Das Wachstum der Hosts weltweit ist in Abbildung Anhang A.11 dargestellt. Das weltweite Wachstum des Internets ist eine exponentielle Kurve - dargestellt in Abbildung Anhang A.12. Besonders hervorzuheben wäre, daß das Wachstum in den USA - dem Ursprungsland des Internet - bereits hinter dem Rest der Welt zurücksteht. Und die beeindruckendste Zahl der Graphik ist wohl, daß derzeit alle 30 Minuten weltweit ein weiteres Netzwerk dem Internet hinzugefügt wird.



**Abbildung Anhang A.10:** Die Abbildung gibt Aufschluß über den Zuwachs an Netzwerken in Europa. Voran ist Frankreich, direkt gefolgt von Deutschland und Großbritannien. Der hohe Zuwachs ist zum größten Teil den in diesen Ländern guten Infrastrukturen zu verdanken - aber auch in den restlichen Ländern ist das Wachstum beachtlich. (Quelle: <ftp://ftp.isoc.org/charts/networks-gifs/>)



**Abbildung Anhang A.11: Wachstum des Internets.** Die Abbildung zeigt das Wachstum des frühen ARPANETS und späteren ARPANETS sowie das Wachstum der Hosts mit IP. (Quelle: <ftp://ftp.isoc.org/charts/networks-gifs/>)

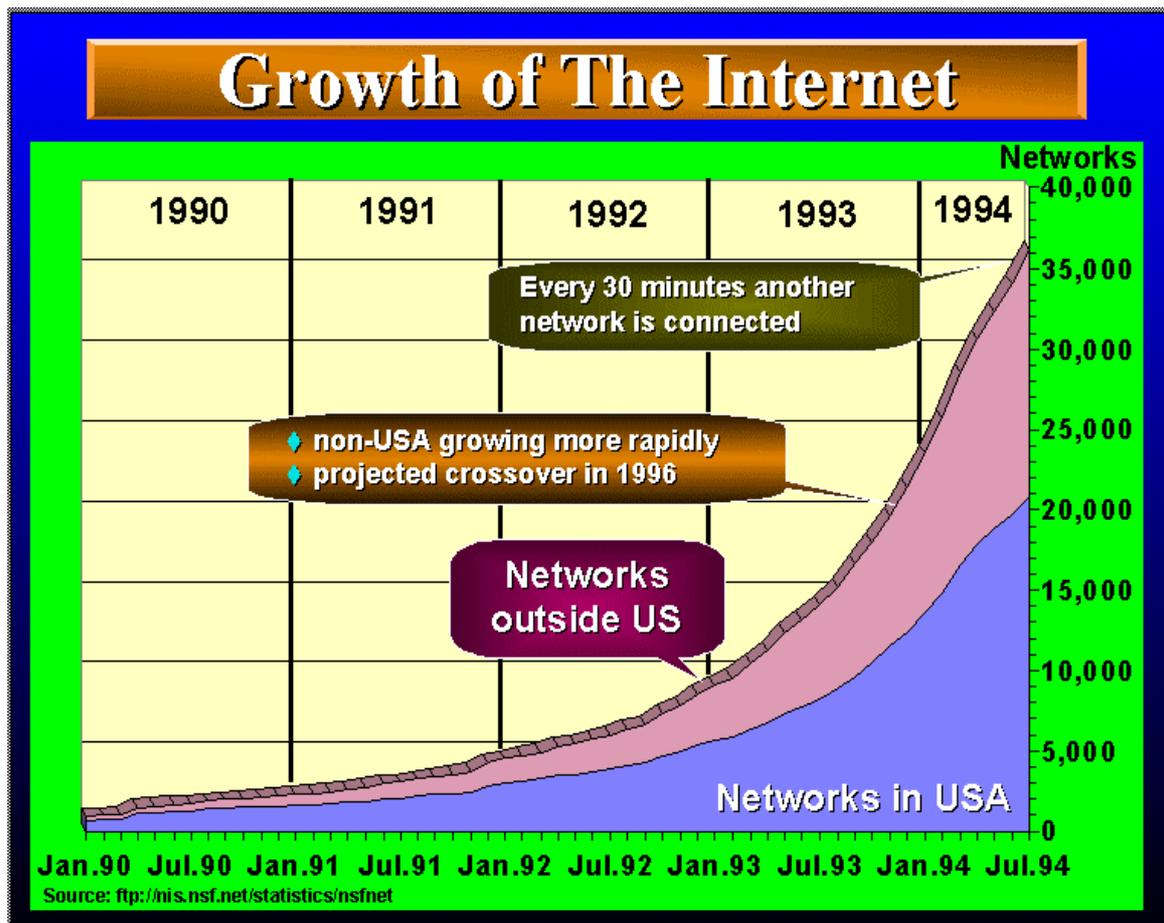


Abbildung Anhang A.12: Der Wachstum von US Netzwerken im Vergleich zum Rest der Welt Alle 30 Minuten wird ein weiteres Netzwerk an das Internet angeschlossen (Quelle: <ftp://ftp.isoc.org/charts/networks-gifs/>)

## Technische Grundlagen von ATM

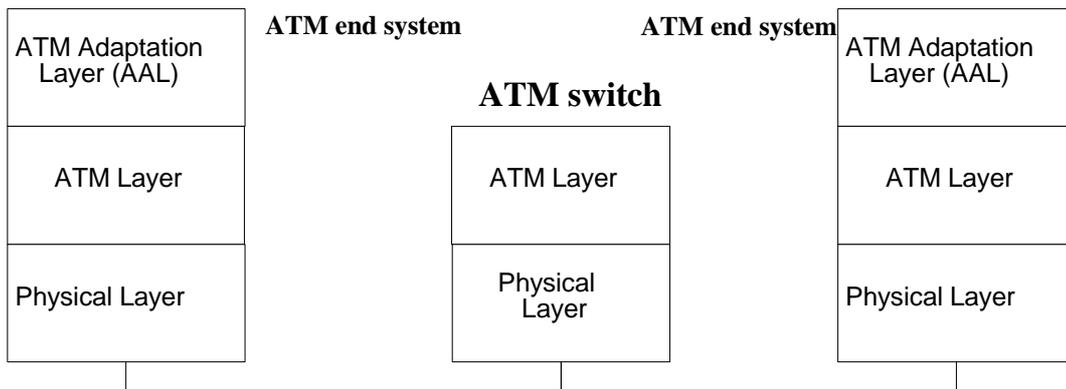
Nach dem technischen Überblick in Kapitel 4 werden hier einige Details von ATM näher durchleuchtet.

### Das Protokoll Referenz Modell

Das ATM Protokoll besteht aus drei wichtigen Schichten, deren Basisfunktionen hier kurz erläutern werden. Eine typische End-zu-End Verbindung ist in Abbildung Anhang A.13 dargestellt. Das ATM Protokoll Referenz Modell ist Abbildung Anhang A.14 zu entnehmen.

### ATM Adaptions Schicht (ATM Adaptation Layer ... AAL):

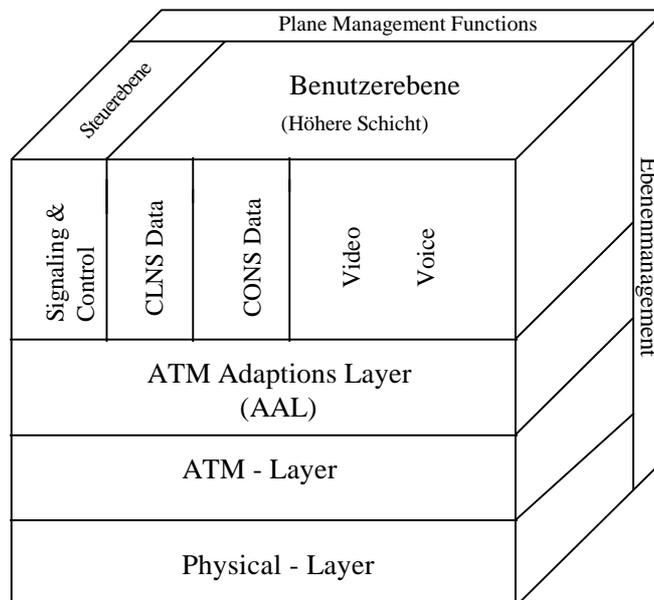
Die ATM AALs sind dafür verantwortlich, alle existierenden Datendienste (sowohl verbindungsorientiert/paketorientiert als auch synchron/asynchron/isochron) zu unterstützen und transparent für den Benutzer zu übertragen.



**Abbildung Anhang A.13:** ATM End-zu-End Verbindung. Die Abbildung zeigt eine typische End-zu-End Verbindung via einem Netzwerk Switch. Innerhalb des Netzwerkes wird ausschließlich das reine ATM Protokoll verwendet. Dies ist in der ATM Schicht implementiert. An den Endverbindungen agieren die AALs, um z.B. TCP/IP für den User transparent zu übertragen.

Ursprünglich wurden vier AALs von der ITU-T definiert, die direkt mit Service Klassen von A-D bezeichnet wurden. Später wurden AAL3 und AAL4 zu einem Layer zusammengefaßt und mit AAL3/4 bezeichnet. Dieser AAL unterstützt nun sowohl Klasse C als auch D

AAL 5 wurde vom ATM Forum spezifiziert und es wird von der ITU-T erwogen, diese Schicht ebenfalls für Dienste der Klasse C zu erwägen Prycker, 94]. Die Entwicklung wurde von zahlreichen Herstellern und Anwendern vorangetrieben, da



**Abbildung Anhang A.14:** Das ATM Protokoll Referenzmodell. Nur die unteren 3 Schichten werden von ATM wahrgenommen, die oberen Schichten stellen den Zugang zu höheren Schichten (Anwendungen) dar.

der von der ITU-T spezifizierte AAL3/4 einen zu hohen Overhead besitzt und gleichzeitig eine zu geringe Anzahl von Bit Fehlern per CRC erkennt.

AAL2 ist noch nicht fertig spezifiziert. Der Fertigstellung dieses Standards wird sowohl von der ITU-T als auch vom ATM Forum keine besonders hohe Priorität eingeräumt: bis zum Jahr 2000 wird der AAL2 in keiner Arbeitsgruppe behandelt werden. Es gibt jedoch Bemühungen, den AAL5 für Video Übertragung zu benutzen - siehe hierzu [Sisalem2, 95]. Ende 1996 wurde eine Draft Version für einen neuen ATM Adaptions Layer, den sogenannten AAL-CU (CU ... Composite User) eingereicht. Er ist für den Transport von zeitabhängigen Diensten mit geringer Bitrate ausgelegt, also für komprimiertes Video in Kombination mit komprimierten Ton oder für nicht komprimiertes Audio Datenmaterial. Ein möglicher daraus resultierender Standard soll nach letzten Informationen bei der ITU-T AAL6 heißen.

Abbildung Anhang A.15 gibt darüber Auskunft, welche AALs man für bestimmte Diensteanforderungen verwenden kann. So benötigt man für eine IP über ATM Übertragung z.B. AAL 3/4, da TCP/IP keine zeitliche Relation zwischen Quelle und Ziel verlangt, eine variable Bitrate akzeptiert und ein verbindungsloser (Datagram) Dienst ist.

Mit den fünf AALs ist es ATM möglich, alle existierenden Datendienste zu übertragen, was auch eines der Hauptziele der ATM Technologie war. Auf Grund der äußerst allgemeinen Merkmale bietet ATM auch einen Rahmen für zukünftige, noch zu definierende Dienste.

	<b>Service Klasse</b>			
<b>Anforderungen</b>	Klasse A	Klasse B	Klasse C	Klasse D
Zeitbeziehung zwischen Quelle und Ziel	benötigt		nicht benötigt	
Bit Rate	konstant	variabel		
Verbindungs Modus	verbindungsorientiert			verbindungslos
AAL(s)	AAL1	AAL2	AAL3/4 oder AAL5	AAL3/4 oder AAL5
Beispiele	DS1, E1, px64, unkompr. Video	VOD, Audio	Frame Relay	IP, SMDS, X.25

**Abbildung Anhang A.15:** *ATM Service Klassen.* Definition der einzelnen AALs samt Beispielen für welche bereits existierende Services sie angewendet werden können.

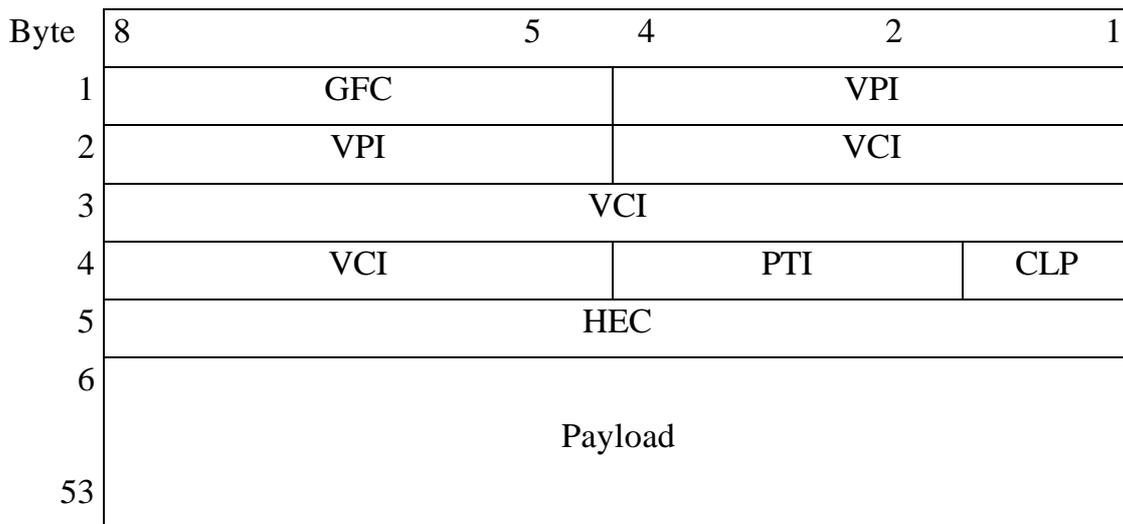
### Die ATM Schicht:

Das Erscheinungsbild der ATM Schicht variiert in Abhängigkeit davon, ob sich die Schicht in einem Endsystem oder in einem Switch befindet. Im ersteren ist die ATM Schicht dafür verantwortlich, die von der physikalischen Schicht kommenden Zellen an den AAL weiterzugeben und umgekehrt. Dies funktioniert über das sogenannte UNI (User Network Interface), welches auch die Definition eines Dienstes (Quality of Service - siehe auch nächsten Abschnitt) zulässt. Nur wenn der definierte Dienst über die gesamte Kommunikationsstrecke garantiert werden kann, wird die Verbindung aufgebaut - ansonsten verweigert.

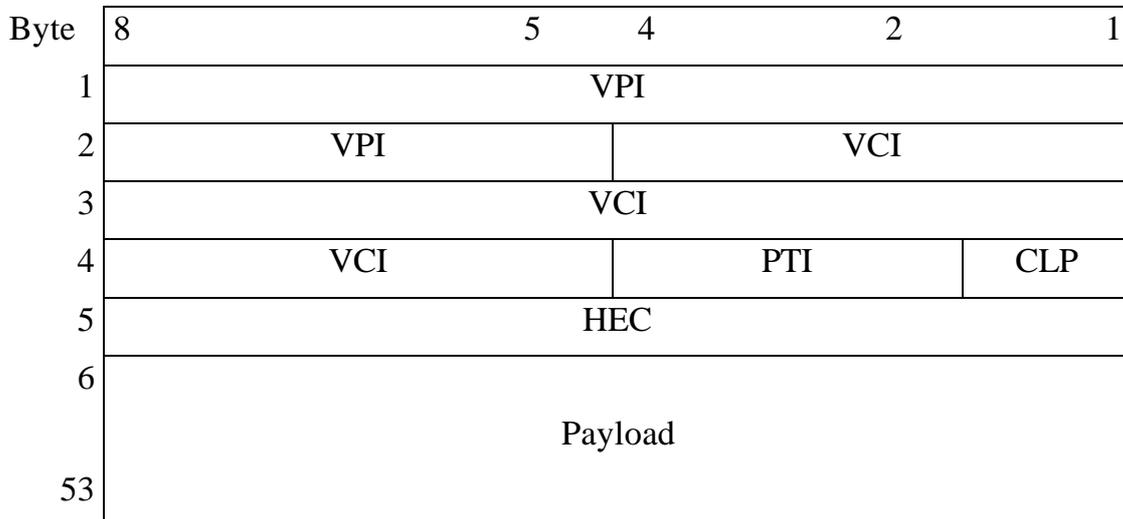
Weiters obliegen der ATM Schicht Verkehrsmanagement Aufgaben wie etwa Reaktion bei Überlastung (congestion control, generic flow control) als auch die Verwaltung der CAC (Connection Admission Control) und der UPC (Usage Parameter Control). Die CAC dient zum Aufbau der Verbindung mit einem bestimmten Dienste Qualität (Quality of Service ... QoS), die UPC ist eine Überwachungsfunktion und prüft, ob der Benutzer die definierten Parameter auch einhält [Saito, 94]. Im Netzwerk ist die ATM Schicht hauptsächlich für eine korrekte Weiterleitung der Zellen zwischen den einzelnen Knoten zuständig. Dies geschieht über das sogenannte NNI (Network Network Interface). Entsprechend ob es sich um einen Endknoten bzw. um eine Switch zu Switch Kommunikation handelt gibt es zwei verschieden definierte ATM Zellenköpfe. Diese sind in Abbildung Anhang A.16 dargestellt.

Verbindungen werden in ATM mit sogenannten „virtual paths“ (virtuelle Pfade) erzeugt, wobei jeder virtuelle Pfad aus mehreren „virtual channels“ (virtueller Kanal) bestehen kann. Um ein geeignetes Routing zu gewährleisten, verfügen sowohl die UNI als auch die NNI Zellen jeweils über ein Virtual Path Identifier (VPI) und ein Virtual Channel Identifier (VCI) Feld. Diese beiden Felder formen zusammen das Routingfeld, welches aber immer nur lokale Bedeutung hat. D.h. jeder Switch kann Inhalte dieser Felder ändern, entsprechend wird die Zelle an ihr Ziel gesteuert.

Der UNI Header beinhaltet 24 Bits (VPI+VCI) für User Sessions, was 16 Millionen verschiedenen Verbindungen pro ATM Endinterface entspricht. Der NNI beinhaltet 28 Bits (VPI+VCI) für Session Identifikationen, was einer Anzahl von mehr als 268 Millionen entspricht. Die restlichen Felder sind Generic Flow Control (GFC), Header Error Correction (HEC), Payload Type Indicator (PTI) und Cell Loss Priority (CLP). Die genaue Bedeutung würde den Rahmen dieser Einführung bei weitem sprengen, ausführliche Erklärungen sind in [McDysan, 94], [Prycker, 94], [Händel, 94] und [Goralski, 94] nachzulesen.



(a)



(b)

**Abbildung Anhang A.16: ATM Zellenformat.** (a) stellt eine ATM Zelle für das UNI dar, spezifisch dafür das Generic Flow Control (GFC) Feld. In (b) ist eine Zelle des NNI dargestellt, spezifisch hierfür die 12 Bit für den VPI (Virtual Path Identifier) - siehe Text für nähere Erläuterungen.

### Die physische Schicht:

Die physische Schicht ist in zwei Sublayer gegliedert. Die Physical Medium Dependent (PMD) Schicht nimmt die Aufgabe des Timings der einzelnen Bits über das physische Medium wahr. ANSI, CCITT/ITU-T und das ATM Forum haben diese Schicht definiert. Folgende Schnittstellen werden angeboten [McDysan, 94], [Onvural, 94]:

- STM-1/SONET OC-3 (STS-3c) mit 155.52 Mbit/sec

- STM-4/SONET OC-12 (STS 12c) mit 622.08 Mbit/sec
- DS1 mit 1.544 Mbit/sec
- DS2 mit 6.312 Mbit/sec
- DS3 mit 44.736 Mbit/sec
- FDDI mit 100 Mbit/sec
- DQDB mit 155.52 Mbit/sec
- Multimode Fiber Interface mit 155.52 Mbit/sec
- E3 mit 34 Mbit/sec
- E4 mit 139.264 Mbit/sec

Der zweite Sublayer, der sogenannte Transmission Convergence Sublayer (TCS) ist für die Konversion des Bitstromes in ATM Zellen bzw. umgekehrt zuständig. Diese Schicht interpretiert auch das HEC (Header Error Correction) Feld, welches einen 1 Bit fehlerkorrigierenden Kode aufweist.

### **ATM Dienste**

Vor dem Verbindungsaufbau hat der Benutzer die Möglichkeit, einen sogenannten „Quality of Service“ (QoS) Vektor zu definieren. Dieser Vektor umfaßte in der UNI Version 3.0 sechs Parameter, ab Version 3.1 [UNI31, 95] sieben. Die letzte Version des UNI ist Version 4.0 [UNI40, 96]. Sie ist in zwei Subdokumente aufgesplittet: Die Traffic Management Spezifikation [UNI40a, 96] und die ATM UNI Signalling Spezifikation [UNI40b, 96]. Die einzelnen Parameter sind:

#### **Spitzenzellenrate (Peak Cell Rate):**

Die maximale Bandbreite, die ein Benutzer verwendet. Diese darf nur kurzzeitig (peak) verwendet werden.

#### **Durchschnittsrate (Sustained Cell Rate):**

Die durchschnittliche Bandbreite - gemessen über eine lange Zeit, mit der ein Benutzer Daten verschickt.

#### **Zellenverlustwahrscheinlichkeit (Cell Loss Ratio):**

Beschreibt den prozentmäßigen Anteil der Zellen, die während des Transports verloren gehen dürfen (etwa während Netzwerkfehlern oder Überlastung).

#### **Zellentransport Verzögerung (Cell Transfer Delay):**

Dieser Wert beschreibt die verstrichene Zeit zwischen Netzeintritt und Austritt.

Diese Zeit schließt die Leitungsverzögerung, die Service Zeiten an den Warteschlangen und die Wartezeiten an denselbigen sowie die Verzögerungen an den zahlreichen Zwischenknoten ein.

**Zellenverzögerungs Variation (Cell Delay Variation):**

Hiermit wird die Varianz der Zellentransport Verzögerung spezifiziert. Eine hohe Zellverzögerungs Variation zieht große Puffer mit sich.

**Burst Toloranz (Burst Tolerance):**

Die Burst Toleranz gibt an, wie lange der Benutzer maximal mit der Spitzenzellenrate senden darf.

**Minimale Zell Rate (Minimum Cell Rate):**

Dieser Wert beschreibt die minimale Datenrate, mit der ein Benutzer Daten verschickt. Diese Rate wird vom Netzwerk mit einer gewissen hohen Wahrscheinlichkeit garantiert ( $10^8$  bis  $10^{10}$  sind typische Werte [Prycker, 94]). Erhöht der Benutzer allerdings die Rate, so werden die Zellen dieses Kanals als erstes weggeworfen, falls eine Überlastung eintritt.

Die Definition eines QoS mit Hilfe der oben beschriebenen sieben Parameter nennt man einen sogenannten spezifizierten QoS Kontrakt. Neben diesem existiert auch ein unspezifizierter QoS Vertrag. Dieser spezifiziert nur einen Teil der obig beschriebenen Parameter.

Der Sinn liegt darin, daß spezifizierte QoS Verträge alleine u.U. ziemlich viel Bandbreite verschwenden können, vor allem wenn Benutzer Bandbreiten reservieren, die sie dann niemals effektiv benutzen bzw. nur burstartig. Die unspezifizierten QoS Kontrakte nutzen diese Restbandbreiten aus bzw. teilen sich die vorhandene Bandbreite untereinander fair auf, indem jeder gleich viel bekommt (best effort Basis). Folgende unspezifizierte QoS Kontrakte gibt es:

**Konstante Bit Rate (Constant Bit Rate):**

Dieser Service wird dazu verwendet, um Leitungsvermittlung (Telephonnetz) zu emulieren. Bei Überlastung wird das sogenannte CLP (Cell Loss Priority) Bit gesetzt, welches der Quelle via den sogenannten Resource Management Zellen zurückgeschickt wird. Es obliegt nun dem Sender, seine Datenrate zu reduzieren, unterläßt er dies, gehen die Zellen bei weiterer Überlastung des Knotens verloren (Warteschlangenüberlauf).

**Variable Bit Rate (Variable Bit Rate):**

Die Variable Bit Rate wird vor allem für das Senden von komprimierten Daten mit Echtzeitanforderung benötigt, wie etwa für komprimiertes Video im MPEG-I oder II

Format. Für diesen Service wird die Zellenverzögerungszeit und die Zellenverzögerungs Variation definiert. Für nicht Echtzeitanforderungen benötigt man die Zellenverzögerungs Variation nicht, ein Beispiel hierfür wäre: Multi-Media E-mail.

#### **Verfügbare Bit Rate (Available Bit Rate):**

Dieser Service wurde für einen Datenverkehr spezifiziert, der zwar nicht empfindlich gegen Verzögerung aber sehr wohl gegen Zellenverlust ist. Ein typisches Beispiel dafür sind binäre Files, wo schon ein simpler Bitfehler ausreichen kann, um das Programm funktionsuntüchtig zu machen, oder noch schlimmer die Funktionsweise zu verändern. Der Dienst nutzt die vorhandene Bandbreite in fairer Weise aus, d.h. im Falle von Überlastung ist es Aufgabe des Senders, seine Datenrate zu reduzieren. Teilen sich mehrere ABR (Available Bit Rate) Verbindungen eine Leitung, so resultiert das im Idealfall in einer gleichmäßigen Aufteilung der vorhandenen Bandbreite zwischen den verschiedenen ABR Diensten. ABR ist auch ein idealer Service um TCP/IP über ATM zu übertragen. Dies ist derzeit ein großes Forschungsgebiet, genauere Informationen sind [Li, 95] [Sisalem, 95] [Perloff, 95] zu entnehmen.

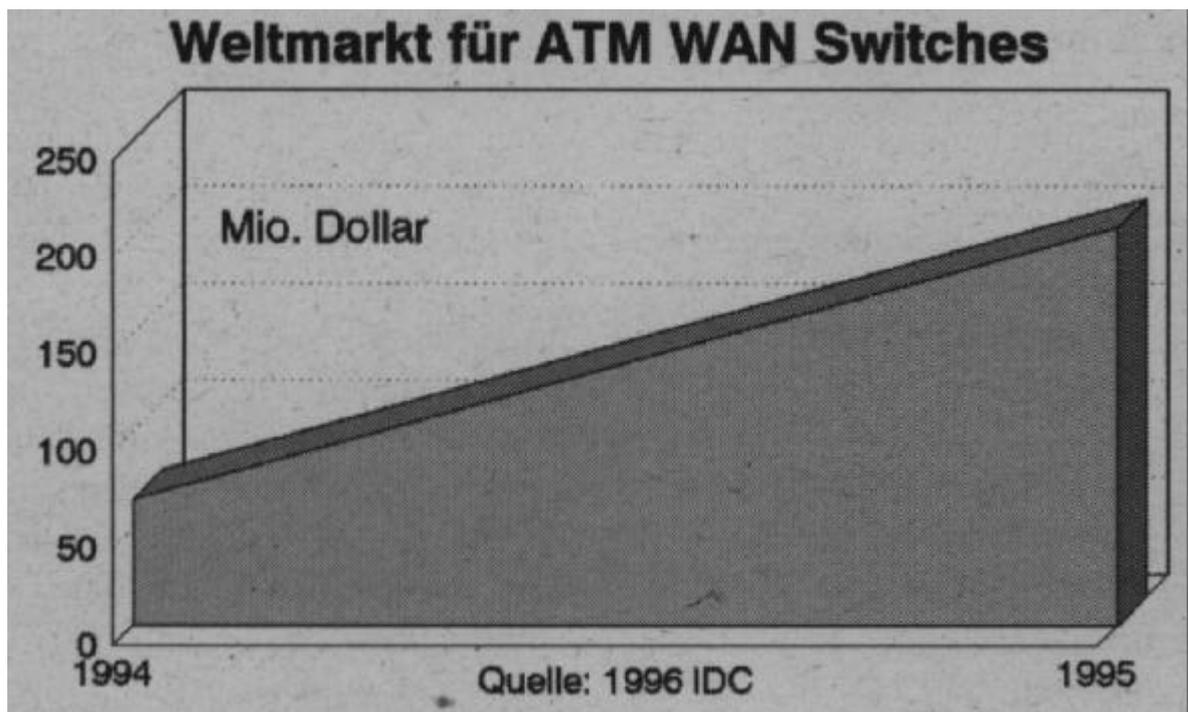
#### **Nicht spezifizierte Bit Rate (Unspecified Bit Rate):**

Der Dienst wendet sich an Applikationen, die weder sehr sensitiv gegenüber Zellenverlust noch gegen Zellenverzögerung sind. Es wird jegliche noch irgendwie übriggebliebene Bandbreite benutzt, Zellen gehen bei Überlastung ohne Warnung verloren. Der Sender wird also nicht informiert, seine Datenrate zu senken. Der Dienst benötigt daher keine Überwachungsfunktion, ob definierte Parameter eingehalten werden. Eine generische Flußkontrolle muß daher an den Endsystemen in einer höheren Schicht implementiert sein.

Abschließend sei noch bemerkt, daß es eine rege Forschungstätigkeit im Bereich IP over ATM gibt. Hier sei auf eine Reihe von Publikationen verwiesen, die dieses Thema in großer Detailtiefe behandeln: [Cole, 95], [Perloff, 95], [Sisalem, 95].

### **Wirtschaftliche Aspekte von ATM**

Bevor man zu einer neuen Technologie umsteigt, muß man natürlich zuerst eine generelle Idee über den Marktstellenwert, Entwicklungen und Trends des Produktes haben. Selbiger wird in diesem Abschnitt näher beschrieben.

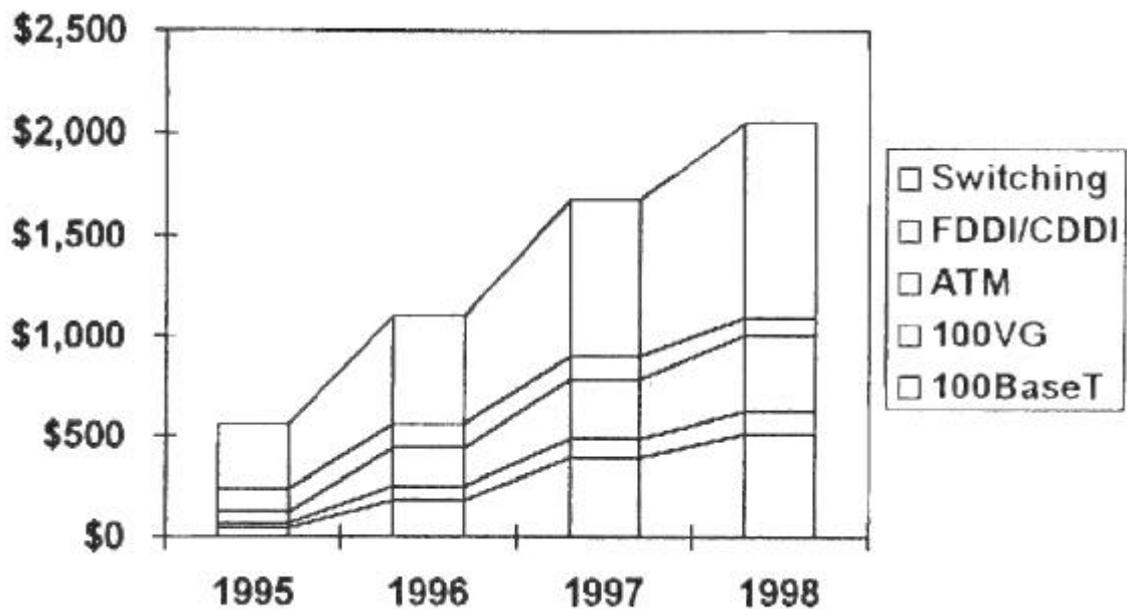


**Abbildung Anhang A.17:** ATM WAN Markt für switches. Das Jahr 1995 war für den Datenkommunikationsmarkt von einem Boom geprägt. Alleine der ATM WAN switching Markt hat sich verdreifacht, aber auch der Modem Markt und der Frame Relay Markt konnten zulegen.

„Es geht also nicht nur darum, möglichst viele Bits in möglichst kurzer Zeit in eine Leitung zu stopfen, es gilt auch den Transfer entsprechend zu organisieren. [...] Die nach heutigem Stand der Technik zukunftsicherste Methode der Breitbandkommunikation, die sowohl Geschwindigkeit als auch Organisation beinhaltet, heißt ATM.“ [Gober, 96] Dies stammt von Franz Gober, Marketing Manager für den Netzbereich bei der DEC Österreich AG. Doch sieht die Zukunft von ATM wirklich so rosig aus?

Zumindest der Weltmarkt für ATM Switches boomt. In Abbildung Anhang A.17 ist eine Statistik von IDC (International Data Corporation) dargestellt. Konkret hat sich der Weltmarkt für ATM WAN Switches 1995 verdreifacht.

Laut den Marktforschern der IDC [Gober, 96], hat es auch auf den anderen Segmenten des Datenkommunikationsmarktes viele Neuerungen gegeben. So hat etwa der Modemmarkt um 82% zugenommen, der Frame Relay [McDysan, 94] Markt hat sich immerhin verdoppelt. Doch läßt dies eine weltweite Anerkennung von ATM schließen, oder ist diese Statistik zu stark von den USA beeinflusst, wo es am meisten ATM Installationen bzw. Pilotversuchsnetze zu verzeichnen gibt?



**Abbildung Anhang A.18:** Ausgaben für Netzwerk Ausrüstung in West Europa. Die Angaben sind in Millionen Dollar zu verstehen - die einzelnen 5 Linien sind von oben nach unten den jeweiligen Kategorien Switching, FDDI/CDDI, ATM, 100VGanyLAN und 100BaseT zugeordnet. (Quelle IDC, 1996)

Eine weitere Studie von IDC ist in Abbildung Anhang A.18 zu finden. Diese Graphik gibt Auskunft über die Ausgaben in West Europa, die für High Speed LAN Ausrüstung (Interface Karten, Hubs und Switches) getätigt wurden bzw. eine Prognose für die Zukunft. Die Angaben sind in Millionen Dollar.

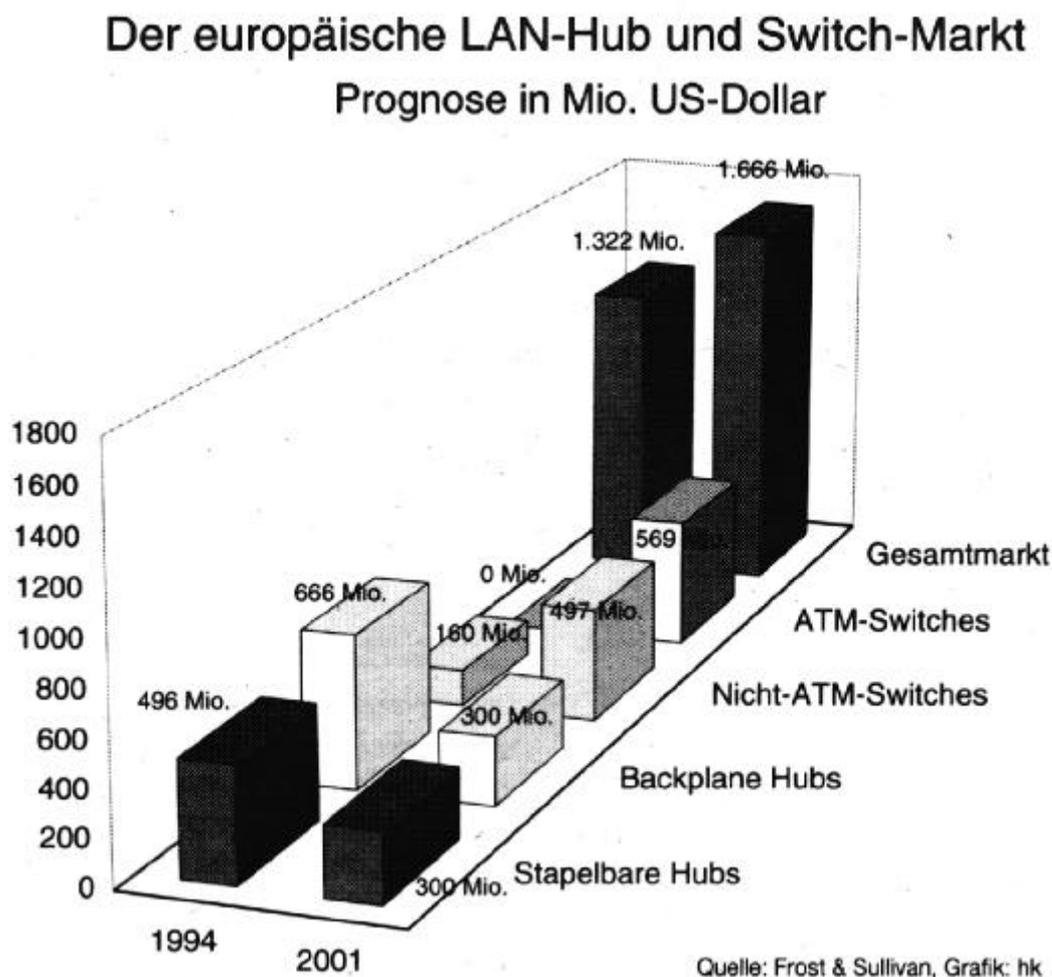
Die Prognose von 1995 bis 1998 stellt hier eine gute Vervielfachung des ATM Anteils dar, den größten Zuwachs im LAN Bereich erfahren jedoch die Switched-Technologien wie etwa Switched-Ethernet und Switched-Token Ring. Eine Migration von dieser Technologie zu ATM stellt jedoch keine besonderen Schwierigkeiten dar. So bieten die meisten Switch Hersteller bereits ATM Interfaces an, man hat also die Möglichkeit, das Netzwerk entsprechend den Anforderungen zu skalieren.

Dies wurde auch am institutseigenen Netz so gemacht (siehe auch Kapitel 4.4.5). Einfache Skalierbarkeit ist eines der wichtigsten Merkmale von ATM. Entgegen den herkömmlichen LAN Technologien (Ethernet, IBM Token Ring, Peer-to-Peer Netzwerke, Fast Ethernet, VGanyLAN) ist es keine broadcast Technologie sondern verbindungsorientiert, basierend auf Switching-Technologie. Skalierbarkeit ist daher äußerst einfach zu verwirklichen.

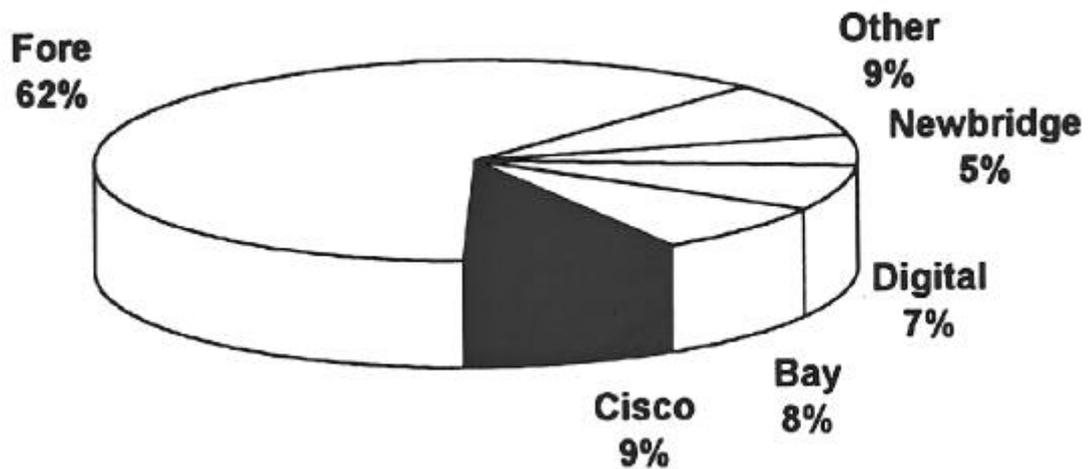
Vom Bekanntheitsgrad in Europa sieht es mit ATM laut einer Umfrage vom ATM Forum allerdings noch nicht so gut aus [Distler, 96]. Befragt wurden insgesamt 350 Telekommunikations Strategen in Groß- und Mittelbetrieben in insgesamt 12 verschiedenen europäischen Staaten. Das Ergebnis spiegelt einen erstaunlich niedrigen

Bekanntheitsgrad von ATM wieder. So haben 32% aller befragten Strategen den Begriff ATM noch nicht einmal gehört. In Unternehmen mit mehr als 200 Millionen ECU Einnahmen war ATM nur bei 30% bekannt. In den Mittelbetrieben war der Bekanntheitsgrad von ATM schon immerhin bei 50%. Dennoch prognostiziert das ATM Forum einen Anstieg der ATM Anschlüsse in Europa von derzeit knapp 1000 auf 22.000 im Jahre 2000, wenn die folgenden Bedingungen erfüllt sind:

- aggressive Ausbreitung der öffentlichen ATM Anbieter in den größten Städten Europas in den nächsten fünf Jahren
- aggressive Preis Politik
- aggressives Marketing von ATM als eine Gesamtlösung



**Abbildung Anhang A.19:** Umsatzprognose des europäischen Netzwerkmarktes laut einer Studie von Frost&Sullivan. Bis ins Jahr 2001 soll demnach der ATM Switchanteil den höchsten Teil des Marktes innehalten. Andere Switches folgen schon mit einigem Abstand. Backplane Hubs und stapelbare Hubs haben zusammen schon fast weniger Umsatzvolumen als die ATM Switches alleine.



**Abbildung Anhang A.20:** Marktanteile der einzelnen Hersteller am ATM Segment. Eindeutiger Führer ist FORE Systems, der erste Verfolger, Cisco, ist mit 53% zurück und derzeit nicht wirklich dominierend.

Dies löst laut der ATM Forum Studie einen sogenannten „Lawineneffekt“ aus, der den exponentiellen Anstieg der Anschlüsse plausibel macht. Eine Studie von Frost & Sullivan [Kapper, 96], die bis ins Jahr 2001 reicht, prognostiziert ein ähnlich optimistisches Wachstum des ATM Marktes. Demnach soll der ATM Switch Markt im Jahr 2001 den Gesamtmarkt anführen - siehe hierzu Abbildung Anhang A.19.

Haben im Jahre 1994 die Backplane Hubs unangefochten mit 666 Millionen US Dollar die Spitze gehalten, so wird dies im Jahr 2001 gerade umgekehrt sein. Laut der Graphik werden dann nämlich ATM Switches die Führung übernehmen, Backplane Hubs und stapelbare Hubs zusammen werden kaum das Umsatzvolumen von ATM Switches mehr erreichen.

Von der Herstellerseite gibt es einen eindeutigen Marktführer in der ATM Technologie, wie man in Abbildung Anhang A.20 erkennen kann. FORE Systems [FORE, 96] führt den weltweiten Markt mit 62% Anteil an.

Der erste Verfolger, der renommierte Router und Netzwerkkomponenten Hersteller Cisco, liegt mit 9% weit dahinter. FORE Systems wurde 1990 von vier Professoren<sup>1</sup> der Carnegie Mellon Universität gegründet. FORE Systems hat sich von Beginn an auf ATM Hardware spezialisiert und in seinem mehrjährigen Bestehen eine Menge Know-how aufgebaut [Bierck, 96]. Darin scheint auch die große Marktpräsenz letztendlich begründet. Alle anderen Hersteller haben ATM Produkte mehr als Nebenerwerb in ihre Produktpalette aufgenommen und bieten neben ATM auch noch zahlreiche andere Netzwerkprodukte an. Somit konnten sie nicht alle

<sup>1</sup> Es waren: Francois Bitz, Onat Menzilcioglu, Robert Sansom und Eric Cooper. Die ersten Buchstaben der Vornamen bilden den Firmennamen FORE.

Investitionen im ATM Segment tätigen, was wiederum den hohen Marktanteil von FORE plausibel macht.

Insgesamt spiegeln die zahlreichen Studien doch eine deutlich positive Einstellung gegenüber ATM wider. Vom wirtschaftlichen Standpunkt her ist eine ATM Backbone Lösung für GDSS auf jeden Fall vertretbar.

### **Zukünftige Entwicklungen und Trends**

Prognosen über zukünftige Marktentwicklungen und Trends sind bei Anschaffung einer neuen Technologie von entscheidender Bedeutung, will man doch seine Investitionen als einigermaßen gesichert sehen. Prinzipiell lassen sich im ATM Segment zwei Hauptmärkte unterscheiden. Es sind dies der europäische und der nordamerikanische Markt. Beide Märkte weisen unterschiedliche Charakteristiken auf, die im folgenden Abschnitt dargestellt werden.

Der Markt für ATM basierende Produkte und Services am Kommunikationssektor kann in drei Segmente unterteilt werden [Roberts, 96]:

1. Öffentliche ATM Infrastruktur
2. LANs und MANs
3. WANs

Unter der **öffentlichen ATM Infrastruktur** versteht man die Gesamtheit von physischen Medien, die von einem „Provider“ angeboten wird, um eine Vielfalt an Diensten zur Verfügung zu stellen. Es handelt sich um ein Segment mit relativ wenigen Anbietern. Hauptsächlich findet man hier öffentliche Telefon Gesellschaften, Kabel TV Anbieter und internationale Provider. In Europa werden öffentliche Telekommunikationsnetzwerke entweder von der EU oder von den einzelnen Staaten betrieben und gewartet. Gemäß dem Bangemann Report [Bangem, 94] wurde aber ein Beschluß verfaßt, den Infrastrukturmarkt für den kommerziellen Wettbewerb zu öffnen. Dies soll für alle Mitgliedsstaaten der EU im Jahre 1998 in Kraft treten. Mit diesem Jahr fällt auch das Monopol der Österreichischen Post und Telekom, Austria.

Prinzipiell kontrolliert die EU die Interoperabilität der internationalen Netzwerke durch die Vorschrift von ETSI Standards. Öffentliche Anbieter werden gezwungen, sich diese Standards zu kaufen und sich daran zu halten. Diese umfassen derzeit Telefonnetz und Privatnetze bis zu 2 Mbit/sec. Es wurden aber bereits 34 Mbit/sec (E3), Frame Relay und 155 Mbit/sec (STM1 bzw. OC/3) auf die Liste aufgenommen. Diese Dienste können öffentliche Provider freiwillig anbieten.

Die derzeit sehr hohen Tarife in Europa verhindern jedoch eine rasche kommerzielle Nutzung. Das große Ziel in Europa besteht in der Errichtung einer öffentlichen ATM Infrastruktur, die sowohl alle momentanen Dienste als auch zukünftige

Dienste ohne große Erweiterungen transportieren kann. Die Wartungskosten werden damit auf ein Minimum schrumpfen, da es anstatt vieler hoch spezialisierter Netze nur mehr ein einziges zu verwalten gilt.

Gibt es in Europa noch eine recht strikte Trennung zwischen Infrastrukturanbieter und Diensteanbieter, so ist in den USA dies bereits oft unter einem Hut. Der Markt ist also weit mehr liberalisiert als in Europa. Außerdem besteht nicht der Zwang für öffentliche Anbieter, ETSI Standards (oder vergleichbare ANSI Standards) zu kaufen. Politisch wird in den USA von der Regierung eine Summe von 3 Billionen Dollar auf eine Periode von 5-6 Jahren für Forschungszwecke zur Verfügung gestellt. In Europa gibt es lediglich nationale Förderungen, die allerdings 1994 mit dem Zusammenschluß von 18 europäischen Telefon Kompanien ein internationales Versuchspilotnetzwerk initialisiert haben. Der Trend in Europa geht aber ebenfalls zur Fragmentierung des Marktes, die mit der zunehmenden Liberalisierung zweifellos eintreten wird. Somit werden der europäische und amerikanische Markt in Zukunft im öffentlichen Infrastruktur Sektor kaum noch nennenswerte Unterschiede aufweisen.

Unter dem LAN/MAN Sektor versteht man die gesamten Komponenten, die Workstations, PCs und andere Ressourcen über relativ kurze Distanzen<sup>1</sup> verbinden. An Komponenten versteht man hier alle Arten von Rechnern, vom Laptop bis hin zum Großrechner (Mainframe). Weiters zählt man auch Adapter Karten, Switches, Hubs und Router hinzu. Dieses Marktsegment ist sehr groß und zeichnet sich vor allem durch viele verschiedene Technologien und äußerst kurze Innovationszyklen aus. Dadurch ist auch eine hohe Fragmentierung gegeben, die sich aus dem Wettbewerb der einzelnen Händler und Anbieter ergibt. Es gibt jedoch gegenüber dem Infrastruktur Segment keine Einschränkungen durch Monopole oder die EU. Somit sind die Unterschiede dieses Marktsegments zwischen Europa und Amerika verschwindend gering. So ergeben sich starke Ähnlichkeiten in [Roberts, 96]:

- existierenden LAN Technologien
- Applikationen
- LAN/WAN Integration
- Komponenten und Händler
- Benutzeranforderungen und -erwartungen

Einer der wichtigsten Unterschiede ist der Zeitvershub zwischen dem europäischen und dem amerikanischen Markt. Dieser Unterschied beträgt zwischen 6 und 18 Monaten. Damit in engem Zusammenhang steht auch die etwas geringere

---

<sup>1</sup> etwa in einem Gebäude (LAN) oder einem Kampus (MAN)

Marktfragmentierung in Europa. Da die meisten neuen LAN Technologien in den USA entwickelt werden, kann es passieren, daß eine Technologie nicht international anerkannt und ein völliger Flop wird. Aufgrund des Zeitverschubs kommt diese Technologie jedoch nie nach Europa. Ein weiterer Unterschied liegt in den verschiedenen WAN Interfaces (E0 und T0, E1 und T2, E3 und T3 usw.).

Unter einem WAN Segment versteht man Netzwerke, die eine oder mehrere örtlich getrennte Niederlassungen verbinden<sup>1</sup>. Es kann sich hierbei um ein privates (also gemietete Leitungen), um ein öffentliches Netz oder um eine Mischung aus beiden Varianten handeln. Ein besonders wichtiger Faktor sind die Kommunikationskosten. ATM soll WAN Verbindungen erschwinglich machen und die Schlüsseltechnologie für multimediale Desktop-zu-Desktop WAN Kommunikationen sein.

Das Marktsegment ist derzeit relativ klein, ist aber im Wachsen begriffen. Die Märkte in den USA und in Europa weisen signifikante Unterschiede auf. So ist etwa in Europa der Markt von öffentlichen Anbietern und VAS (Value Added Service providers) bestimmt. In Amerika hingegen ist es der Wettbewerb zwischen den IXC (Inter-Exchange Carriers) Anbietern, wobei die Verbindungen meist auf T1 (1.5 Mbit/sec) oder T3 (45 Mbit/sec) basieren.

Ein weiterer großer Unterschied besteht auch in den Tarifen. So ist etwa in Europa eine E1 (2 Mbit/sec) Hochgeschwindigkeitsverbindung 4 mal so teuer wie in Amerika eine T1 (1.5 Mbit/sec) Verbindung<sup>2</sup>. Daraus folgt auch eine vermehrte Verwendung von langsameren Diensten in Europa, wie das äußerst populäre X.25, welches in den USA fast gar nicht verwendet wird.

Somit ist in Europa ATM derzeit im WAN Sektor kaum in Verwendung, der MAN Sektor ist aber bereits expandierend. Und mit der zunehmenden Liberalisierung des Marktes und dem Fall der Monopole wird auch in Europa ATM bald in diesem Marktsegment an Popularität gewinnen. In den USA hat der ATM WAN Markt bereits begonnen, erste große private WANs werden in den Jahren 1997 und 1998 erwartet.

Abschließend kann festgestellt werden, daß es teilweise signifikante Unterschiede zwischen dem US Markt und dem europäischen Markt gibt. Diese sind jedoch mit der zunehmenden Liberalisierung des europäischen Marktes zum größten Teil am Verschwinden. Das ATM Forum wird das seinige beitragen, um die regionalen Unterschiede weiter geringer werden zu lassen bzw. das Bewußtsein der Benutzer für nahtlose internationale Kommunikation zu erhöhen.

---

<sup>1</sup> typischer weise wird ein Firmenhauptsitz mit den Filialen verbunden

<sup>2</sup> Stand 1996

## Literatur und Referenzen zum Anhang A

- [3Com, 96] 3Com Network Corporation, Mai 1996. Homepage. <URL: <http://www.3com.com>>
- [Adoba, 93] Adoba B. „The Online User’s Encydopedia“. ISBN 0-201-62214-9, Addison Wesley, 1993
- [Albert, 94] Albert B., Jayasuamana A. „FDDI and FDDI-II“, ISBN 0-89006-633-7, Artech House, 1994
- [Alcatel, 96] Alcatel Data Networks, November 1996, Homepage, <URL: <http://www.adn.alcatel.com>>
- [Bangem, 94] Bangemann M. et. „Europe and the global information society Recommendations to the European Council“, Report der High-Level Group on the Information Society für ein Treffen des Europarates im Juni 1994 in Korfu, <URL: <http://www.earn.net/EC/report.html>>
- [BBN, 96] BBN Internet Research Corporation. Homepage. <URL: <http://www.bbn.com>>
- [Bierck, 96] R. Bierck „From the Kitchen Table to the Corner Office“. Pittsburgh’s business magazine, Ausgabe März 1996, Seiten 26-30
- [Cabletron, 96] Cabletron Corporation Homepage. Cabletron Spectrum Network. <http://www.ctron.com/Catalog/Net-Management/>, Mai 1996
- [Cole, 95] Cole R., Shur D., Villamizar C. „IP over ATM: A Framework Document“, IETF Internet draft Dokument, 1995. draft-ietf-ipatm-framework-doc-02, Download über meine ATM Seite:  
<URL: <http://www.icg.tu-graz.ac.at/~herwig/Research/ATM/ATM.html>>
- [Distler, 96] Distler K., Bosser T., Grant M. „Survey Reveals Low Awareness of ATM in Europe“. White paper, 1996. Download über meine ATM Seite:  
<URL: <http://www.icg.tu-graz.ac.at/herwig/Research/ATM/ATM.html>>
- [DOS 1, 95] DOS International, „Das Einmaleins des Internet“, Markt & Technik Verlag, Ausgabe 1/95, Seite 248-250
- [EBONE, 92] E-BONE, 1992. The European Backbone 1992 - Specifications, Workplans, Organisations.  
<URL: <ftp://nic.nordu.net/ebone/docs/ebone-92.txt>>
- [FORE, 96] Fore systems, Homepage, Dez. 1996, <URL: <http://www.fore.com>>
- [Gober, 96] Gober F., „Der Asynchrone Transfer Modus: Geschwindigkeitsrausch auf dem Datenhighway“. Computerwelt 16/96, April 1996, S. 28,29
- [Goralski, 94] Goralski W. „The Future of High-Speed Networking“, ISBN 1-56607-037-6, Computer Technology Research Corp., 1994
- [Hahn, 94] Hahn H., Stout R. „Internet - Complete Reference“. ISBN 0-07-881980-6, Osborne McGraw-Hill, 1994
- [Händel, 94] Händel R., Huber M., Schröder S. „ATM Networks“, ISBN 0-201-42274-3, Addison-Wesley Verlag, 1994
- [Hedrick, 87] Hedrick C. „Introduction to the Internet Protocols“. Rutgers State University of New Jersey, 1987. <URL: <http://NIC.MERIT.EDU/introducing.the.internet/intro.to.ip>>
- [Internic, 96] InterNIC, Homepage, <http://rs.internic.net>, Oktober 1996
- [Kapper, 96] Kapper H., „Innovation als treibende Kraft: Europas Netzwerkmarkt“. Computerwelt 32-33/96, August 1996, Seite 9

- [Kegel, 96] Kegel. D., 1996. Dan Kegels ISDN Pages, <URL: <http://www.alumni.caltech.edu/~dank/isdn/>>
- [Kunft, 95] Kunft W., Rastl P., Steinringr H. „Das Österreichische Akademische Computernetz“, EDV-Zentrum Wien, ftp-server, 1995.  
<URL: <ftp://ftp.univie.ac.at/netinfo/aconet/aconet-doc.grman.ps>>
- [Li, 95] Li H., Siu K., Tzeng H., Ikede C., Suzuki H. „TCP Performance over ABR and UBR Services in ATM“. Proceedings of International Phoenix Conference on Computers and Communications, Sept. 1995, Download über meine ATM Seite:  
<URL: <http://www.icg.tu-graz.ac.at/~herwig/Research/ATM/ATM.html>>
- [McDysan, 94] McDysan D., Spohn D., „ATM - Theory and Application“. ISBN: 0-07-060362-6, McGraw-Hill, 1994
- [Muller, 93] Muller N. „Intelligent Hubs“. ISBN 0-89006-698-1, Artech House, 1993
- [NetAssoc, 96] IBM, DEC Netview Association Homepage für Netzwerkmanagement.  
<URL: <http://www.raleigh.ibm.com/nva/nvaover>, Mai 1996>
- [Netview, 96] IBM Homepage für Netview Netzwerk Management.  
<URL: <http://www.raleigh.ibm.com/nv6/nv6prodhtml>, Mai 1996>
- [Onvural, 94] Onvural R. „Asynchronous Transfer Networks: Performance Issues“. ISBN 0-89006-662-0, Artech House Verlag, 1994
- [OpenView, 96] HP OpenView Netzwerkmanagementsystem, Mai 1996,  
<URL: <http://www.hp.com/nsmd/ov/main.html>>
- [Perloff, 95] Perloff M., Reiss K. „Improvements to TCP performance in High Speed ATM Networks“. Communications of the ACM, Vol. 38 No. 2., 1995, pp. 91-100.
- [Prycker, 94] Prycker M. „Asynchronous Transfer Mode“. ISBN: 3-930436-03-5, Prentice Hall, 1994
- [Raffer, 96] Raffer P., Rehatschek H. "Die digitale Vernetzung Österreichs", Telematik Ingenieur Verband Magazin, Ausgabe 4/96, Dez. 1996
- [Rastl, 96] Rastl P. „Wo geht's hier ins Internet? ACONet und seine internationalen Netzverbindungen“. Pipeline Magazin, Nr 19/Juni 1996, S. 5,6
- [RFC793, 81] Internet Engineering Task Force (IETF) RFC 793. Transmission Control Protocol - DARPA Internet Program Protocol Specification, Sept. 1981, für das DARPA aufbereitet vom UCLA, <URL: <http://ds.internic.net/rfc/rfc793.txt>>
- [RFC1163, 90] Internet Engineering Task Force (IETF) RFC 1163, 1990. „A Border Gateway Protocol (BGP)“. <URL: <http://ds.internic.net/rfc/rfc1163.txt>>
- [RFC1171, 90] Internet Engineering Task Force (IETF) RFC 1171, 1990. „The Point-to-Point Protocol for the Transmission of Multi-Protocol Over Point-to-Point Links.  
<http://ds.internic.net/rfc/rfc1171.txt>
- [Roberts, 96] Roberts A., Litchfield C., Horvath S. „Comparison of European and US Market Trends in ATM Implementation“, ATM Forum European Market Awareness Committee, unpublished, 1996. Download über meine ATM Seite:  
<URL: <http://www.icg.tu-graz.ac.at/herwig/Research/ATM/ATM.html>>
- [Saito, 94] Saito H. „Teletraffic Technologies in ATM Networks“. ISBN: 0-89006-622-1, Artech House, 1994.
- [Siegl, 94] Siegl M. „Was ist Netzwerkmanagement?“. EDV-Zentrum, Technische Universität Wien, White Paper, 1994
- [Sisalem, 95] Sisalem D., Schulzrinne H., „Congestion Control in TCP: Performance of Binary

- Congestion Notification Enhanced TCP Compared to Reno and Tahoe TCP“, Proc. of International Conference on Networks and Protocols (ICNP), Columbia Ohio 1996, Download über meine ATM Seite: <URL: <http://www.icg.tu-graz.ac.at/~herwig/Research/ATM/ATM.html>>
- [Sisalem2, 95] Sisalem D. „Video Transmission over ATM and IP“, unpublished, 1995. Download über meine ATM Seite: <URL: <http://www.icg.tu-graz.ac.at/herwig/Research/ATM/ATM.html>>
- [SNMP, 90] Internet Engineering Task Force (IETF) RFC 1157. „A Simple Network Management Protocol (SNMP)“. <URL: <http://www.es.net/pub/rfc/rfc1157.txt>, May 1990>
- [SNMPv2\*, 96] Presse Aussendung. „Major Network Equipment and Management Vendors Demonstrate SNMPv2\* and Call for the IETF to Reconvene the Open Standardization Process“. Webserver von SNMP Research, <http://www.snmp.com/pressrel/vegas96-v2demo.html>, April 1996
- [SNMPv2c, 96] Internet Engineering Task Force (IETF) RFC 1901. „Introduction to Community-based SNMPv2“. <http://www.es.net/pub/rfc/rfc1901.txt>, Jänner 1996
- [Solstice, 96] Sun Microsystems Solstice Enterprise Network Manager, Homepage, , Mai 1996, <URL: <http://www.sun.com/products-n-solutions/sw/solstice/em-products/network/ent.man.html>>
- [Stevenson, 95] Stevenson D., Hillery N., Byrd G. „Secure Communications in ATM Networks“. Communications of the ACM, Vol. 38 No 2, pp. 46-53, 1995
- [Stevenson2, 95] Stevenson D., „Network Management: What it is and what it isn't“, unpublished, April 1995, <URL: <http://netman.cit.buffalo.edu/Doc/DStevenson/>>
- [SUN, 96] Sun Micro Systems, Mai 1996. Homepage. <URL: <http://www.sun.com>>
- [Tanenbaum, 90] Tanenbaum A. „Computer Netzwerke“. ISBN 3-925328-79-3, Wolfram, 1990
- [Tivoli, 96] Tivoli Inc. Homepage für Netzwerkmanagement. <http://www.tivoli.com>, Mai 1996
- [UNI31, 95] ATM Forum, 1995. „User Network Specification Version 3.1“, ISBN 0-13-393828-X, Prentice Hall Verlag
- [UNI40, 96] UNI Version 4.0, ATM-Forum: Traffic Management Specification, Version 4.0. AF-TM95-0013R2
- [UNI40a, 96] UNI Version 4.0, The ATM Forum Technical Committee, April 1996. af-tm-0056.000
- [UNI40b, 96] UNI Version 4.0, The ATM Forum, Technical Committee, July 1996. af-sig-0061.000



## Anhang B: Datenbank

### Erfahrungen und Probleme bei der Installation vom Oracle Enterprise Server

Im Prinzip ging die Oracle Installation relativ rasch vonstatten, einige aufgetretene Probleme sollten jedoch nicht verschwiegen werden. Sie sollten zukünftigen Oracle Administratoren eine kleine Hilfe geben.

- ? Beim ersten Installationsversuch wurde mehr Software als nötig installiert - eine Deinstallation scheiterte vorerst, da angeblich Oracle (trotz Shutdown!) schon lief
- ! Das Problem kann man manuell lösen, indem man alle laufenden ora\_\* Prozesse tötet sowie das File „sgadef<Datenbankname>.dbf“ im dbs Verzeichnis löscht. Dieses File sagt dem Server, daß bereits eine Datenbankinstanz läuft.
  
- ? Installation eines native JDBC (Java Data Base Connectivity) Treibers
- ! Oracle bat bis Dezember 1996 keinen Treiber an - ich werde von der Konkurrenz (Informix) auf einen Drittanbieter - Weblogic Inc.<sup>1</sup> - verwiesen. Diese Software jedoch auf Grund einer Übereinkunft zwischen Oracle und Weblogic Inc. seit Dezember 1996 auch auf der Oracle Developer CD enthalten.
  
- ? Der Löschen von Relationen im Oracle Browser funktioniert nicht mit Edit/Cut wie im Handbuch bzw. in der On-Line beschrieben.
- ! Man muß das Kommando: Schema/Remove verwenden.
  
- ? Beim Upgrade auf die Version 7.3.2 läßt sich „orainst“ auf einmal nicht mehr starten. Das Programm bricht mit der Fehlermeldung „dba Group not defined“ ab, dies trotzdem die Variable eindeutig definiert ist.
- ! Das Problem liegt in einem Bug des Oracle Servers bezüglich Yellow Pages/NIS. Scheinbar funktioniert die ordnungsgemäße Abfrage des Oracle Administrator Paßwortes sowie der Gruppe nicht über die NIS Datenbank. Die Behebung des Problems erfolgt durch eine Eintragung der dba Gruppe sowie des Oracle Accounts in das lokale Gruppen/Paßwort File. Sodann funktioniert alles wie gehabt.
  
- ? Alle Motif Programme haben Probleme mit dem NIS, dies sind vor allem der Browser und der Server Manager. Das Problem äußert sich, wenn ein normaler Datenbank Benutzer (also kein Administrator am Oracle Account) versucht sich einzuloggen. In diesem Falle wird der Einlogvorgang mit einer Fehlermeldung quittiert, eine Verbindung zur Datenbank bleibt erfolglos.
- ! Dieses Problem ist ähnlich dem vorherigen - nicht nur die Server Software sondern auch einige Tools haben das Problem, daß NIS Paßwörter nicht ordnungsgemäß abgefragt werden können. Die Lösung ist ähnlich wie oben: jeder

---

<sup>1</sup> <URL: <http://www.weblogic.com>>

der die Oracle Tools benutzen will, muß seinen Account Eintrag im lokalen Passwortfile auf der Maschine des Servers haben. Außerdem bekam ich von der Oracle Hotline noch den Tip die Browser Software nicht im selben Verzeichnis wie die des Servers zu installieren, da dies zu Problemen führen kann. Ich konnte jedoch trotzdem ich dasselbe Verzeichnis verwendet hatte keinerlei Probleme erten.

Insgesamt muß man noch die äußerst kompetente Hotline der Firma Oracle erwähnen. Die dort beschäftigten Mitarbeiter haben in sehr professioneller Art und Weise bei einer Vielzahl von Problemen geholfen.

## Anhang C: java

### Historisches

Im Jahre 1990 begann bei Sun Microsystems ein Projekt namens „Green“, dessen Leiter Patrick Naughton, James Goslin und Mike Sheridan waren. Ziel des Projektes war es, eine neue Programmiersprache zu entwickeln, die Haushaltsgeräte wie z.B. Videorecorder, Telefone, etc. von derzeit eher passiven zu prozessorgesteuerten Begleitern des Menschen machen soll. Diese Sprache muß natürlich völlig unabhängig von der jeweiligen Hardware Plattform arbeiten können, egal ob Kaffeemaschine, Videorecorder oder PC [Röder, 96] [Vanhel, 96].

Schon bald wurde eine neue objektorientierte Programmiersprache namens „Oak“ (Eiche) von James Gosling geschaffen. James Gosling hatte die Schwächen bisheriger OO Sprachen - nämlich: mangelnde Stabilität und Zuverlässigkeit - erkannt und in der neuen Sprache zu eliminieren versucht. Oak sollte auch unerfahrenen Programmieren die Möglichkeit geben, stabile Applikationen zu entwickeln. Selbst krasse Bedien- und Programmierfehler durften nicht zu einem unkontrollierten Fehlverhalten der Applikation führen. Im Prinzip war Oak ein abgespecktes C++, ohne jedoch Befehle für den direkten Zugriff auf die umgebende Hardware zu beinhalten. Hier vielleicht noch die Antwort auf die berechtigte Frage, warum gerade „Oak“ als Name ausgewählt wurde: während Gosling einen entsprechenden Namen seiner neuen Sprache nachdachte fiel sein Blick gedankenverloren aus dem Fenster, und dabei sah er einen Eichenbaum.

Nachdem Oak im Jahre 1992 erstmals dem damaligen Sun Chef Scott McNealy vorgestellt wurde, war dieser so begeistert, daß er das Projekt Team sofort in ein eigenständiges Unternehmen namens „First Person“ verwandelte. Die neue Sun Tochter sollte sofort Verhandlungen mit den wichtigsten Herstellern von Haushaltsgeräten und Konsumgütern herstellen. Weiters erfolgte eine Teilnahme an einer Ausschreibung für Settop Boxen von Time Warner. Jedoch sowohl die Gespräche mit den Elektrokonzernen als auch die mit Time Warner verliefen im Sand. Im Jahre 1994 wurde First Person aufgelöst, Oak drohte in Vergessenheit zu geraten.

Bill Joy, Mitbegründer von Sun Microsystems erkannte jedoch noch rechtzeitig das ungeheure Potential des plattformunabhängigen Oak und schlug vor, es für das WWW zu verwenden. Oak könnte somit dem bis dato fast ausschließlich passiven Web neues Leben einhauchen. Und so geschah es dann auch. Zusammen mit Gosling und Naughton wurde das Oak Projekt gerettet und ein neuer Name, Java wurde gefunden. Oak konnte aus politischen Gründen nicht mehr weiter verwendet werden, da mit diesem Namen zuviel Negatives verbunden wurde. In den USA stand Java als das Synonym für das Grundnahrungsmittel eines jeden Programmierers: Kaffee.

Seit diesem Zeitpunkt ist der Einzug von Java auf dem Internet nicht mehr zu stoppen, kaum mehr ein Webanbieter will auf Java Programme verzichten und auch in firmeneigenen Netzen (sogenannten Intranets) spielen Java Applets und Java Programme eine immer entscheidendere Rolle. Vor allem die

Plattformunabhängigkeit und die immer zahlreicheren APIs trugen entscheidend zum Siegeszug von Java bei.

### **Literatur und Referenzen zum Anhang B**

- [Röder, 96] Röder T. „Die Geschichte von Java: Von der Eiche zum Kaffee“. PC Professional Sonderausgabe 1996, Faszination Java, Ziff-Davis Verlag, Seite 4-7
- [Vanhel, 96] Vanhelsuwé L., Phillips I., Hsu G., Sankar K., Ries E., Rohaly T., Zukowski J., Mastering Java. Sybex, ISBN: 0-7821-1935-2, 1996

## Anhang D: Akronyme

Der folgende Anhang stellt eine Übersicht über häufig auftretende Akronyme in der Telekommunikation (mit spezieller Rücksicht auf ATM Technologie) dar. Ein Großteil davon wird auch im Text verwendet, andere findet man in diverser Fachliteratur bzw. Publikationen. Diese Liste, die allerdings keinen Anspruch auf Vollständigkeit erhebt, soll dem Leser helfen, sich erfolgreich im Dschungel von Abkürzungen in der Telekommunikation zu orientieren. An dieser Stelle sei auch noch auf den äußerst guten Akronym-Server am Internet hingewiesen, wo man On-line fast zu jeder Abkürzung eine Erläuterung erhält:

<URL: <http://www.ucc.ie/info/net/acronyms/acro.html>>

**AAL**

ATM Adaptation Layer

**ABR**

Available Bit Rate

**ANSI**

American National Standards Institute

**API**

Application Programming Interface

**APPN**

Advanced Peer to Peer Network

**ARP**

Address Resolution Protocol

**ARQ**

Automated Repeat reQuest

**ATM**

Asynchronous Transfer Mode

**ATMARP**

ATM Address Resolution Protocol

**ATS**

Abstract Test Suite

**B-ISDN**

Broadband Integrated Services Digital Network

**BER**

Bit Error Rate

**BPS**

Bits per second

**BT**

Burst Tolerance

**BUS**

Broadcast and Unknown Server

**CAC**

Connection Admission Control

**CBR**

Constant Bit Rate

**CCITT**

Consultative Committee on International Telephone &amp; Telegraph

**CCR**

Current Cell Rate

**CDT**

Cell Delay Tolerance

**CDV**

Cell Delay Variation

**CDVT**

Cell Delay Variation Tolerance

**CER**

Cell Error Ratio

**CI**

Congestion Indicator

**CL**

Connectionless

**CLP**

Cell Loss Priority

**CLR**

Cell Loss Ratio

**CMIP**

Common Management Interface Protocol

**CO**

Connection Oriented

**COS**

Class of Service

**CPCS**

Common Part Convergence Sublayer

**CRC**

Cyclic Redundance Check

**CS**

Convergence Sublayer

**CSI**

Convergence Sublayer Indication

**CTD**

Cell Transfer Delay

**CTV**

Cell Tolerance Variation

**DCE**

Data Communication Equipment

**DLC**

Data Link Control

**DLCI**

Data Link Connection Identifier

**DQDB**

Distributed Queue Dual Bus

**DS-0**

Digital Signal, Level 0

**DS-1**

Digital Signal, Level 1

**DS-2**

Digital Signal, Level 2

**DS-3**

Digital Signal, Level 3

**DSU**

Data Service Unit

**DTE**

Data Terminal Equipment

**DXI**

Data Exchange Interface

**ELAN**

Emulated Local Area Network

**FCS**

Frame Check Sequence

**FDDI**

Fiber Distributed Data Interface

**FEC**

Forward Error Correction

**FRS**

Frame Relay Service

**FUNI**

Frame User Network Interface

**GCRA**

Generic Cell Rate Algorithm

**GFC**

Generic Flow Control

**GRC**

Generic Reference Configuration

**HDLC**

High Level Data Link Control

**HEC**

Header Error Control

**ICMP**

Internet Control Message Protocol

**IDU**

Interface Data Unit

**IEEE**

Institute of Electrical and Electronics Engineers

**IETF**

Internet Engineering Task Force

**ILMI**

Interim Local Management Interface

**IP**

Internet Protocol

**IPX**

Novell Internetwork Packet Exchange

**ISO**

International Organization for Standardization

**ITU**

International Telecommunications Union

**JPEG**

Joint Photographic Experts Group

**LAN**

Local Area Network

**LANE**

Local Area Network Emulation

**LAPD**

Link Access Procedure D

**LEC**

LAN Emulation Client

**LECS**

LAN Emulation Configuration Server

**LES**

LAN Emulation Server

**LLATMI**

Lower Layer ATM Interface

**LLC**

Logical Link Control

**LLC/SNAP**

Logical Link Control/Subnetwork Access Protocol

**LMI**

Layer Management Interface

**LSB**

Least Significant Bit

**MAC**

Medium Access Control

**MAN**

Metropolitan Area Network

**MBS**

Maximum Burst Size

**MCR**

Minimum Cell Rate

**MCTD**

Mean Cell Transfer Delay

**MIB**

Management Information Base

**MID**

Message Identifier

**MPEG**

Motion Picture Experts Group

**MSAP**

Management Service Access Point

**MSB**

Most Significant Bit

**MSVC**

Meta-signaling Virtual Channel

**MTP**

Message Transfer Part

**MTU**

Message Transfer Unit

**N-ISDN**

Narrowband Integrated Services Digital Network

**NE**

Network Element

**NEBIOS**

Network Basic Input/Output System

**NHRP**

Next Hop Resolution Protocol

**NMS**

Network Management System

**NNI**

Network to Network Interface

**NP**

Network Performance

**NRM**

Network Resource Management

**NSAP**

Network Service Access Point

**NSP**

Network Service Provider

**NT**

Network Termination

**OAM**

Operations, Administration and Maintenance

**ODI**

Open Data-Link Interface

**OSI**

Open systems Interconnection

**P-NNI**

Private Network to Network Interface

**PAD**

Packet Assembler and Disassembler

**PBX**

Private Branch eXchange

**PCM**

Pulse Code Modulation

**PCR**

Peak Cell Rate

**PCVS**

Point to Point Switched Virtual Connections

**PDH**

Plesiochronous Digital Hierarchy

**PDU**

Protocol Data Unit

**PHY**

Physical Layer of the OSI model

**PMD**

Physical Layer Dependent sub-layer

**POH**

Path Overhead

**POI**

Path Overhead Indicator

**PT**

Payload Type

**PTI**

Payload Type Identifier

**PVC**

Permanent Virtual Circuit

**PVCC**

Permanent Virtual Channel Connection

**QD**

Queuing Delay

**QOS/QoS**

Quality of Service

**QPSX**

Queue Packet and Synchronous Circuit Exchange

**RIP**

Routing Information Protocol

**RISC**

Reduced Instruction Set Computing

**RM**

Resource Management

**RSVP**

Resource Reservation Protocol (signalling protocol for IP to reserve resources)

**SAAL**

Signaling ATM Adaptation Layer

**SAP**

Service Access Point

**SAR**

Segmentation and Reassembly

**SCR**

Sustainable Cell Rate

**SDH**

Synchronous Digital Hierarchy

**SDU**

Service Data Unit

**SMDS**

Switched Multi-megabit Data Services

**SNA**

Systems Network Architecture

**SNAP**

Sub Network Access Protocol

**SNMP**

Simple Network Management Protocol

**SOH**

Section Overhead

**SONET**

Synchronous Optical Network

**STM**

Synchronous Transfer Mode

**STM1**

Synchronous Transport Mode 1 -- 155mbits/sec

**STP**

Shielded Twisted Pair cable

**STS-3c**

Synchronous Transport System-Level 3 concatenated

**SVC**

Switched Virtual Circuit

**SVCI**

Switched Virtual Circuit Identifier

**SVP**

Switched Virtual Path

**T1S1**

ANSI T1 Subcommittee

**TC**

Transmission Convergence

**TCP**

Transmission Control Protocol

**TCP-IP**

Transmission Control Program/Internet Protocol

**TCS**

Transmission Convergence Sublayer

**TDM**

Time Division Multiplexing

**TE**

Terminal Equipment

**TM**

Traffic Management

**UBR**

Unspecified Bit Rate

**UDP**

User Datagram Protocol

**UNI**

User Network Interface

**UPC**

Usage Parameter Control

**UTP**

Unshielded Twisted Pair cable

**VBR**

Variable Bit Rate

**VC**

Virtual Channel ( Virtual Circuit)

**VCC**

Virtual Channel Connections

**VCI**

Virtual Channel Identifier

**VLAN**

Virtual Local Area Network

**VP**

Virtual Path

**VP/VC**

Virtual Path, Virtual Circuit

**VPC**

Virtual Path Connection

**VPCI/VCI**

Virtual Path Connection Identifier/Virtual Channel Identifier

**VPI**

Virtual Path Identifier

**WAN**

Wide Area Network