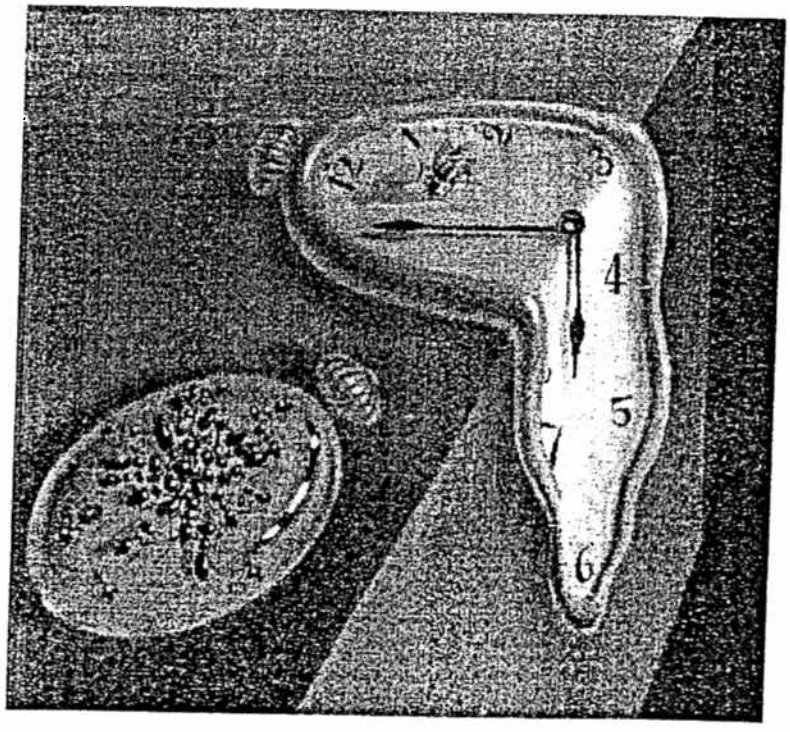
Institut für Automation
Abt. für Automatisierungssysteme

Technische
Universität
Wien

Projektbericht Nr. 183/1-25
Oktober 1991

Das Workstation-LAN der Abteilung Automatisierungssysteme

Ulrich Schmid



Ausschnitt aus: Salvador Dali, "Die Beständigkeit der Erinnerung"

Das Workstation–LAN der Abteilung Automatisierungssysteme

(Institut für Automation, TU–Wien)

ULRICH SCHMID
TU–Wien, Institut für Automation 183/1
Treitlstraße 3, A–1040 Wien

Oktober 1991

Der vorliegende Bericht beschreibt die Struktur des heterogenen Local Area Networks der Abteilung Automatisierungssysteme (Institut für Automation, TU–Wien), welches mittlerweile etwa dreißig – zum Teil sehr unterschiedliche – Rechner verbindet. Er bietet eine einführende Übersicht über die Möglichkeiten und Limitationen des Netzwerks und dokumentiert so das (vorläufige) Endergebnis der langjährigen Tätigkeit des Autors auf dem Gebiet der Geräteplanung für die Abteilung.

1. Rahmenbedingungen und Entwicklungsgeschichte

Die Abteilung Automatisierungssysteme im Institut für Automation (vormals Institut für Technische Informatik) an der TU–Wien wurde 1988 mit dem Ziel geschaffen, das Gebiet des Computereinsatzes in Automatisierungssystemen in Forschung und Lehre abzudecken. Verbunden mit der Gründung des Ordinariats durch die Berufung von o.Prof. Dr. G.–H. Schildt war, neben der Bereitstellung der Abteilungsräumlichkeiten und der Übungsräume im 4. bzw. im 1. Stock der Treitlstraße 3, auch die Zusage von Berufungsmitteln in der Höhe von insgesamt etwa 3.9 Mio.S. Derartige Berufungsmittel werden vom BMWF grundsätzlich nur zeitlich gestaffelt, also auf mehrere Jahre verteilt, zur Schaffung der notwendigen Geräte–Infrastruktur zur Verfügung gestellt. An Personal wurden dem Ordinariat zunächst zwei Universitätsassistenten, ein Techniker und eine Organisationsassistentin (Sekretärin) zugeordnet.

Es galt daher, ein stufenweise finanzierbares Konzept zu entwickeln bzw. zu realisieren, welches den momentanen, aber auch den sich abzeichnenden mittelfristigen Anforderungen in bezug auf Verwaltung, Forschung und Lehre genügen und trotzdem einige Jahre dem State–of–the–Art entsprechen würde. Diese Aufgabe wurde dem Autor im Jahre 1988 übertragen.

Von ganz entscheidendem Einfluß auf das Gerätekonzept waren, abgesehen von mehr oder weniger selbstverständlichen Grundvoraussetzungen wie dem Einsatz vernetzter UNIX–Workstations als Arbeitsplatzrechner, vor allem folgende Aufgaben der Abteilung in der Lehre:

- *Laborübung Prozeßautomatisierung*
  Diese 1.5–stündige Pflichtlehrveranstaltung konkretisiert einige Stoffinhalte der Vorlesung *Prozeßautomatisierung* (Grundlagen paralleler Systeme, Regelungstechnik und Programmierung von Automatisierungssystemen) und ist für

Informatiker aller Wahlfächer im 4. Semester bestimmt; die Anzahl der Übungsteilnehmer liegt demzufolge bei etwa 300 Studenten.

- *Informatikpraktika*
  Derartige Projektarbeiten zielen auf eine umfassende persönliche Betreuung der einzelnen Studenten ab; für ihre Durchführung ist eine gewisse Anzahl dedizierter Arbeitsplätze unerläßlich.

- *Diplomarbeiten & Dissertationen*
  Auch hier gilt im Prinzip das bereits bei den Praktika Gesagte. Diplomarbeiten und Dissertationen werden aber oftmals im Rahmen konkreter Forschungsprojekte durchgeführt, sodaß in der Regel auch projekteigene Maschinen dafür herangezogen werden können.

Neben diesen umfangreichen und kostspieligen Übungserfordernissen waren natürlich auch

- *Arbeitsplatzrechner sowie allfällige Server–Maschinen*

für die Verwaltung und die Forschung von zentraler Bedeutung. Ebenfalls – wenn auch nur mittelfristig – mußten schließlich auch Möglichkeiten zur Einbindung konkreter Forschungsprojekte werden. Bei letzteren sind zwei Varianten zu unterscheiden:

- *Interne Abteilungsprojekte*
  Derartige Forschungsprojekte sind in Punkto Ressourcen (unter anderem auch Räumlichkeiten!) und Mitarbeiter eng an die Abteilung gekoppelt; Forschungsförderungsfonds–Projekte (FWF) fallen üblicherweise in diese Kategorie.

- *Externe Abteilungsprojekte*
  Forschungsprojekte dieser Art sind im Prinzip autonom; die gelegentliche Nutzung von Abteilungs–Ressourcen bzw. die Mitarbeit von Abteilungsmitgliedern ist aber selbstverständlich nicht ausgeschlossen.

Dies also waren (und sind!) die wichtigsten Anforderungen, denen das zu entwickelnde Gerätekonzept genügen mußte – und zwar möglichst optimal und trotzdem finanzierbar.

Angesichts der im Jahre 1988 von Seiten des renommierten Workstation–Herstellers Apollo betriebenen, absolut konkurrenzlosen Universitäts–Preispolitik (vor allem im Vergleich mit Sun oder DEC) und unter Berücksichtigung der guten Erfahrungen mit Apollo–Workstations, die Prof. Schildt im Zuge seiner Tätigkeit bei Siemens gemacht hatte, fiel die prinzipielle Entscheidung schließlich zugunsten des 12 MBit/sec Apollo Domain Token–Ring LANs aus. Eingebunden in dieses Netzwerk wurden zum einen Apollo Workstation unter dem Betriebssystem *Domain/OS* (AEGIS, UNIX BSD4.3 und UNIX SysV) als Arbeitsplatzrechner/Server, und zum anderen auch MS–DOS PC–ATs (mit Apollo Ring–Controllern und

MS-NET) für den Übungsbetrieb: Mit ca. 2.7 Mio.S. aus Berufungsmitteln und einem kleineren Betrag aus der ordentlichen Dotation* wurden im Jahre 1988 insgesamt 14 (zum Teil diskless) Apollo-Workstations und 9 PC-ATs sowie diverse Peripheriegeräte, Spezialhardware und Software angeschafft.

Mit dieser Erstausstattung konnte der Abteilungsbetrieb und, im Sommersemester 1989, auch erstmals der Übungsbetrieb aufgenommen werden. Die prinzipielle Idee des Einsatzes von diskless Apollo-Workstations und PCs mit schreibgeschützter Harddisk (wegen allfälliger Probleme mit Computer-Viren) in Verbindung mit einigen Server-Maschinen erwies sich im Übungsbetrieb als sehr praktikabel. Allerdings zeigte dieser erste "Probedurchgang", daß die (Mit-)Verwendung der primären Abteilungsrechner als Server für die Übungsmaschinen nicht auf Dauer möglich sein würde: Sowohl aus Gründen der Performance als auch des Datenschutzes (die unvermeidlichen "Hacker" unter den Übungsteilnehmern) war es unumgänglich, die Übungsmaschinen in ein separates Netzwerk auszulagern.

Im Jahre 1989 wurden demzufolge 4 weitere Apollo-Workstations inklusive der notwendigen Peripherie und Software um insgesamt etwa 1.2 Mio.S. aus Berufungsmitteln (die damit aufgebraucht waren) und einer kleineren Summe aus der o.Dot. angeschafft. Damit konnten sowohl die Arbeitsplätze für die inzwischen neu hinzugekommenen Mitarbeiter der Abteilung als auch drei Server-Maschinen für die Übungen bereitgestellt werden. Im Sommersemester 1990 wurde der Übungsbetrieb auf dem nun vom Abteilungs-Ring total getrennten Übungs-Ring aufgenommen.

Der heutige Gerätebestand der Abteilung wurde schließlich durch die Anschaffung zweier HP9000/425t-Workstations** aus dem Fonds der sogenannten Diebold-Mittel für die Lehre (1990: ca. 0.62 Mio.S.) und eines PC-486 sowie diverser Peripheriegeräte, Disk-Upgrades, ... aus Mitteln der (erstmals zugewiesenen) außerordentlichen Dotation (1991: ca. 0.3 Mio.S.) erreicht. Ein Teil dieser Maschinen ist übrigens, einer glücklichen Konstellation zufolge, unter Hardware-Wartung.

Ebenfalls in diesem Bericht berücksichtigt ist die momentan vorhandene (Fremd-)Geräteausstattung der aktuellen Forschungsprojekte. Konkret handelt es sich dabei um einige DEC-Maschinen eines externen Projektes CIM, welche vom IUCCIM (Interuniversitären CIM-Zentrum) bereitgestellt wurden, und um eine Sun Sparcstation sowie diverse 68030 VME-Systeme, die zu einem internen (FWF-)Projekt VTA des Autors gehören.

Die eben skizzierte Entwicklungsgeschichte der Hardware war es auch, die die Software-Struktur des Gesamtsystems primär bestimmte. Aufgrund der Tatsache, daß zu Beginn (abgesehen von den PCs) ausschließlich Apollo-Maschinen vorhanden waren, konnte das gesamte System-Management auf den komfortablen Möglichkeiten von Domain/OS (DDS, Domain Distributed Service) aufgebaut werden. Bedingt durch die inzwischen vorhandenen Fremdmaschinen, die Entwicklungen des UNIX-Marktes und, nicht zuletzt, durch den HP/Apollo-Deal erschien es aller-

---

* Die o.Dot. durfte damals noch für die Anschaffung von Software verwendet werden; einer damaligen Konvention der Fachgruppe Informatik zufolge bestand jedoch kein Anspruch auf außerordentliche Dotation, solange noch Berufungsmittel vorhanden waren.

** Die Firma Apollo war inzwischen von HP aufgekauft worden.

dings empfehlenswert, die "reinrassige" Domain/OS-Umgebung konsequent auf Standard-UNIX - Kompatibilität (basierend auf TCP/IP) auszurichten. Auf diese Weise ist es möglich, sowohl die äußerst komfortablen Möglichkeiten von Domain/OS auf den HP/Apollo-Maschinen zu nutzen, als auch andere UNIX-Maschinen ihrer limitierten Funktionalität gemäß zu integrieren. Im übrigen wird im folgenden die Terminologie *Node* verwendet, um eine (Domain/OS-)Maschine in ihrer Domain/OS-Funktionalität anzusprechen; im Gegensatz dazu kennzeichnet der Terminus *Host* eine Maschine in ihrer UNIX-Funktionalität.

Die ohnedies schon sehr unangenehme allgemeine Problematik der Systemadministration eines derartigen Netzwerks wird in unserem speziellen Falle noch dadurch (unerträglich) verschärft, daß an der Abteilung bis jetzt noch keine Stelle für einen Systemadministrator existiert. Weder die an der Abteilung beschäftigten fünf Universitätsassistenten noch der (Hardware-)Techniker sind in der Lage, diese aufwendige Tätigkeit neben den übrigen Aufgaben vernünftig wahrzunehmen. So wurde zum Beispiel die Erstinstallation der Maschinen im Jahre 1988 größtenteils von Univ. Ass. Dr. Johann Blieberger durchgeführt; diese ließ jedoch, aufgrund der fehlenden Dokumentation und Protokollierung, keine ausreichende (Fremd-)Reproduktion der Installation im Falle von Ausfällen zu.

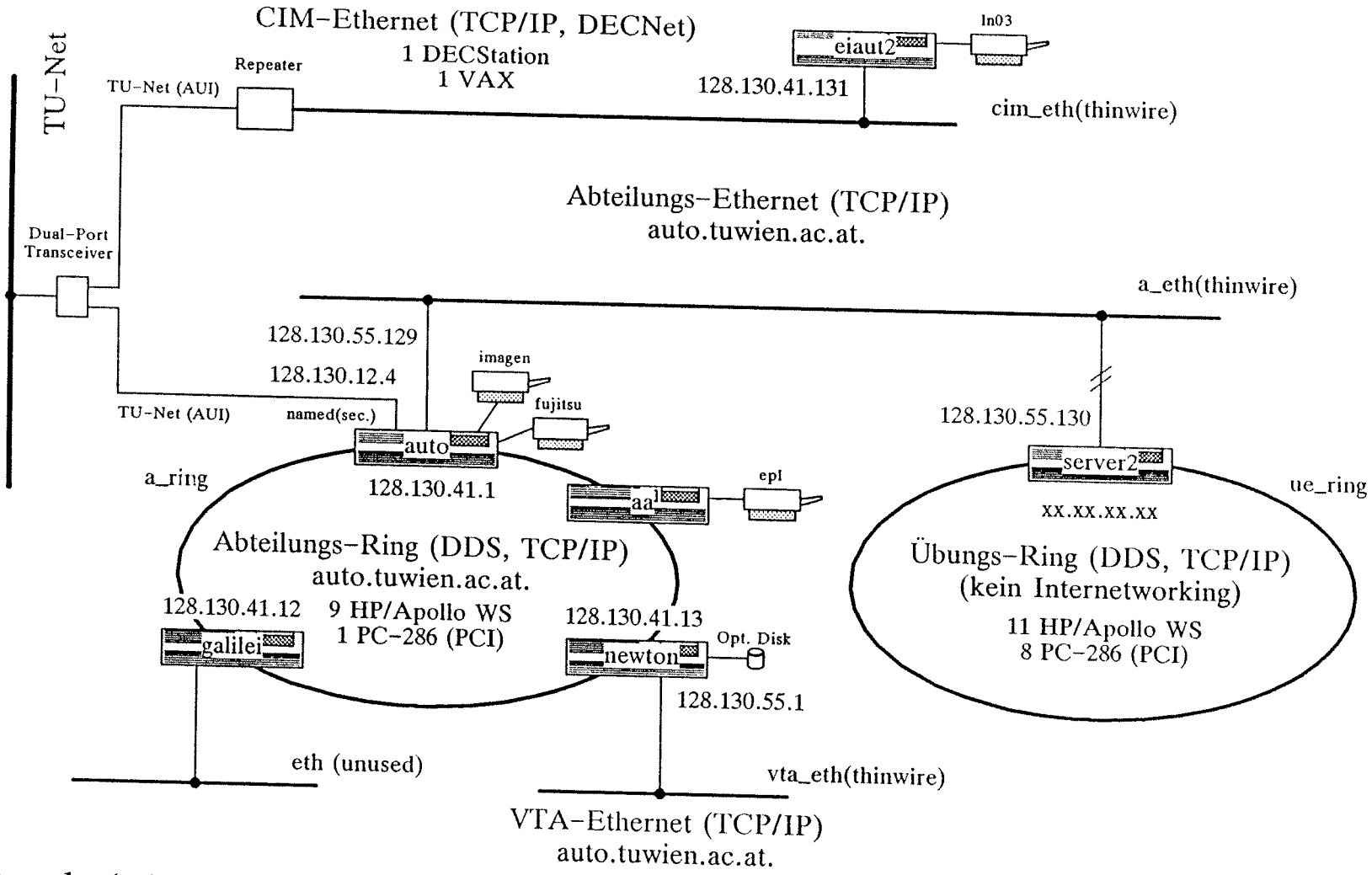
Die Entwicklung eines Systemadministrations-Konzeptes inklusive der notwendigen Installations-Dokumentation und -Protokollierung erfolgte erstmals bei der von Univ. Ass. DI Stefan Stöckler und dem Autor durchgeführten Neuinstallation des (separaten) Übungs-Ringes für die Übungen im Sommersemester 1990. Dieses Konzept wurde von den Studienassistenten Günter Glaser und Klaus Schossmaier anläßlich der Installation des Übungs-Ringes für das Sommersemester 1991 adaptiert und wesentlich weiterentwickelt. Die entsprechenden Protokolle erlauben es mittlerweile, die Aufgabe der jährliche Neuinstallation des Übungs-Ringes einem Nicht-Fachmann zu übertragen. Im übrigen hat sich die Grundsatzentscheidung, nur Workstations (also mit Graphik-Monitor) anzuschaffen (und nicht die nur geringfügig billigeren, monitorlosen Server-Ausführungen) in Punkto Flexibilität und Installationsfreundlichkeit sehr bewährt.

Basierend auf den Erfahrungen mit der Administration des Übungs-Ringes konnte schließlich auch ein geeignetes Konzept für das gesamte Abteilungsnetzwerk entwickelt werden. Die dafür erforderlichen Protokolle wurden im Zuge der hauptsächlich von Günter Glaser und Klaus Schossmaier durchgeführten Neuinstallation der Abteilungsmaschinen erstellt. Das Systemadministrations-Konzept bzw. die zugehörigen Unterlagen sind unabdingbare Voraussetzungen für die gegenwärtig noch erforderliche "verteilte" Systemadministration - und darüberhinaus auch jene Basis, auf der ein (hoffentlich einmal der Abteilung zugeordneter) Systemadministrator bequem aufsetzen kann.

2. Übersicht

Die an der Abteilung befindlichen Geräte sind in mehreren, voneinander relativ unabhängigen Subnetzen untergebracht, deren prinzipielle Struktur folgendermaßen aussieht:

Übersicht über die gesamte Netzwerk-Struktur



Stand: 1.11.1991

Ulrich Schmid

(1) *Abteilungs–Ethernet a_eth*
Dieses (momentan noch spärlich besetzte) Subnetz dient der Einbindung allfälliger nicht–Domain/OS–Maschinen. Wegen der unklaren Situation von HP/Apollo ist derzeit ja noch nicht abzuschätzen, wie der mittelfristige Support von Domain/OS aussehen wird: Im schlimmsten Falle müßten alle im Laufe der Zeit noch hinzukommenden Maschinen* mit Hilfe der gewöhnlichen UNIX–Mechanismen (basierend auf TCP/IP) in diesem Subnetz integriert werden.

(2) *Abteilungs–Ring a_ring*
In diesem Subnetz befinden sich alle Domain/OS–Maschinen der Abteilung (mit Ausnahme der Maschinen für den Lehrbetrieb). Wie schon erwähnt, bildet der Abteilungs–Ring (gegenwärtig noch) den Kern der gesamten Netzwerkstruktur, wobei allerdings in Hinblick auf die UNIX–Kompatibilität zusätzlich zu den Domain/OS–Services auch solche auf der Basis von TCP/IP voll unterstützt werden.

(3) *VTA–Ethernet vta_eth*
In diesem Subnetz befinden sich die vom FWF für das interne Projekt VTA bereitgestellten Maschinen.

(4) *CIM–Ethernet cim_eth*
In diesem völlig selbständigen Netzwerk befinden sich die Maschinen des externen CIM–Projektes. Eine Kopplung zwischen dem bisher beschriebenen Abteilungs–LAN und dem CIM–Ethernet besteht nur über das TU–Net.

(5) *Übungs–Ring ue_ring*
In diesem Subnetz befinden sich alle für den Lehrbetrieb der Abteilung bestimmten Maschinen. Neben drei Apollo–Workstations, die sowohl als Arbeitsplätze für Praktikanten als auch als Server fungieren, finden sich hier 8 diskless Workstations und 8 PCs sowie einige Drucker, die in der Laborübung *Prozeßautomatisierung* verwendet werden. Die PCs sind mit Apollo Ring–Controllern ausgerüstet und über Domain PCI (ein MS–NET Derivat) an die Server gekoppelt. Die lokale Harddisk der PCs enthält nur die für die Übungen notwendigen MS–DOS Programme (Editor, Compiler, Debugger, ...) und ist hardwaremäßig write–protected, sodaß Probleme mit Computer–Viren a priori ausgeschaltet sind. Alle Daten der Übungsteilnehmer werden im Filesystem der Server abgelegt. Nähere Erläuterungen dazu sind in [SS] zu finden.

Der Übungs–Ring ist an sich für den vollständig autarken Betrieb auf der Basis von DDS ausgelegt, obwohl parallel dazu auch die elementaren TCP/IP–Services verfügbar sind. Allerdings sind die jeweiligen IP–Adressen nicht offiziell registriert, die Maschinen daher auch "von außen" nicht (direkt) erreichbar. Eine Ausnahme bildet hier nur der Host *server2*, der, wie im vorherigen Bild ersichtlich, auch am Abteilungs–Ethernet angeschlossen ist (bzw. werden kann). Obwohl die *server2* keinerlei Internet Routing durchführt, ist dennoch ein File Transfer oder ein remote Login auf

* Aus integrationstechnischen Überlegungen wären SunOS–Maschinen vorzuziehen.

eine (bzw. sogar von einer) Übungs-Maschine möglich; zuvor ist lediglich ein remote Login auf der *server2* erforderlich. Da jedoch derartige Dinge höchstens dazu dienen, gelegentlich die Files eines Praktikanten auf eine Abteilungs-Maschine zu holen, bleibt die Verbindung zwischen der *server2* und dem Abteilungs-Ethernet aus Sicherheitsgründen (Hacker!) normalerweise getrennt.

Das generelle LAN-Konzept sieht nun einen alle Möglichkeiten von Domain/OS ausnützenden Abteilungs-Ring vor, der parallel dazu auch die Verwendung seiner Ressourcen über UNIX-Mechanismen gestattet. Tatsächlich bilden die Subnetze (1)-(3) (a_eth, a_ring und vta_eth) ein homogenes TCP/IP-Netzwerk, welches transparent in bezug auf folgende Services ist:

- DARPA TCP/IP-Services (*telnet, rexec*)
- Account-Management
- Printer-Service (*lpd*)
- X-Windows
- Filesystem (NFS)
- Mail

Völlig ausgeklammert sind hier jedoch der Übungs-Ring und das CIM-Ethernet. In diesem Zusammenhang ist festzuhalten, daß für die Integration neu hinzukommender Maschinen insgesamt 3 Varianten (mit progressiv abnehmenden Möglichkeiten) existieren:

(1) *Primäre Abteilungsmaschinen und interne Forschungsprojekte*
Hier gibt es unterschiedliche Alternativen für

(a) *Domain/OS-Maschinen*
Für derartige Maschinen bietet sich eine Integration im Abteilungs-Ring an, die bei weitem die komfortabelsten Möglichkeiten eröffnet.

(b) *UNIX-Maschinen*
Bei der Integration im Abteilungs-Ethernet, oder, falls bei einem internen Forschungsprojekt aus Performance-Gründen ein eigenes Segment erforderlich sein sollte, im jeweiligen Projekt-Subnetz, können alle oben erwähnten TCP/IP-Services in Anspruch genommen werden.

Es ist aber klar, daß die Administratoren solcher Maschinen (vor allem der UNIX-Hosts) alle in den folgenden Abschnitten vorgestellten Restriktionen und Konventionen strikt einhalten müssen, um die Integrität des gesamten Netzwerkes nicht zu gefährden!

(2) *Externe Forschungsprojekte*
Die zugehörigen Maschinen müssen grundsätzlich in einem separaten

(Ethernet-)Netzwerk betrieben werden; ein TCP/IP-Anschluß an das Abteilungs-LAN ist nur über eine Bridge bzw. ein Gateway oder aber über das TU-Net zulässig. Von den oben erwähnten TCP/IP-Services sind in diesem Falle lediglich folgende verfügbar:

- DARPA TCP/IP-Services (*telnet, rexec, ftp*)
- Printer-Service (*lpd*)
- Mail

Die auf diese Übersicht folgenden Abschnitte beziehen sich natürlich nurmehr auf das durch a_eth, a_ring und vta_eth gebildete *Abteilungs-LAN*, auch wenn dies nicht mehr extra erwähnt wird.

3. Software-Struktur

Dieser Abschnitt beschäftigt sich mit prinzipiellen Aspekten der System-Software des Abteilungs-LANs; siehe dazu auch Abschnitt 4 (*Systemadministration*) und Anhang A (*Gesamtkonfiguration*).

3.1 Domain/OS

Domain/OS Distributed Services (DDS) werden gegenwärtig nur im Abteilungs-Ring bereitgestellt; insbesondere findet keinerlei Domain-Routing auf das Abteilungs-Ethernet statt.

Bei der Verteilung der diversen Funktionen bzw. Server auf die einzelnen Maschinen hat sich die von einem "zentralen" Node ausgehende Struktur sehr gut bewährt: Da der Maschine *aa* die Authorized Area sowie die wichtigsten Server (*glbd, rgyd, ns_helper, prmgr* ...) zugeordnet sind, kann sie (stand-alone!) als "Keimzelle" eines eigenständigen Domain/OS-Netzwerkes verwendet werden. Dadurch war es etwa anläßlich des Domain/OS-Upgrades von SR10.1 auf SR10.3 möglich, eine Maschine nach der anderen in das neue SR10.3-Netzwerk "herüberzuziehen", ohne dabei in der Zwischenzeit die Funktionsfähigkeit des verbleibenden SR10.1-Netzwerkes preiszugeben; als letztes wurde der "zentrale" SR10.1-Node übernommen.

Der Maschine *aa* obliegt auch das PCI-Service der im Abteilungs-Ring befindlichen PCs, die dadurch dessen Ressourcen (vor allem das Filesystem und die Drucker) verwenden können. Im Zusammenhang mit TCP/IP stellt die *aa* aber lediglich den *tcp_admin*-Node für die anderen TCP/IP-Hosts sowie den Mailbox-Server bereit; siehe dazu auch die Abschnitte 3.3 (*TCP/IP*) und 3.7 (*Mail*).

Um nämlich eine zu starke Konzentration der Server-Funktionen auf der *aa* zu vermeiden, wurden die (aufwendigen) Funktionen des TCP/IP-Gateways und des TCP/IP-Naming-Servers sowie die eines alternativen *tcp_admin*-Nodes der Maschine *auto* übertragen. Daneben stellt letztere auch noch Replicas für die allerwichtigsten Domain/OS-Server, *glbd* und *rgyd*, bereit. Nur auf diese Weise ist

garantiert, daß die im Falle der Nichtverfügbarkeit der *aa* bzw. der *auto* notwendigen Uminstallations-Arbeiten überhaupt bzw. mit vertretbarem Aufwand durchgeführt werden können.

Angesichts der vielen Aufgaben der Nodes *aa* und *auto* ist aber klar, daß diese als Arbeitsplätze nur beschränkt verwendbar sind. So befindet sich etwa die *aa* (inklusive des Laserdruckers *epl*) in der Bibliothek, wo sie hauptsächlich für den Zugang zur Bibliotheksverwaltung und zur Unterstützung von Besprechungen eingesetzt wird.

3.2 Account-Management

Konzeptuell basiert das gesamte Account-Management auf einer projektorientierten Group-Struktur und (BSD-)Project-Lists. Für jeden speziellen Aufgabenbereich (z.B. die Abteilungsagenden oder ein internes Projekt wie VTA) ist daher eine eigene Group vorgesehen; eine bestimmte Person kann aber klarerweise Mitglied mehrerer solcher Groups sein.

Das praktische Account-Management geht grundsätzlich vom Domain/OS Registry aus, da die für UNIX-Maschinen notwendigen Files */etc/passwd* usw. mittels geeigneter Tools aus dem Registry gewonnen und auf alle UNIX-Hosts* übernommen werden können. In diesem Zusammenhang ist zunächst einmal festzuhalten, daß die Inkompatibilität einiger UNIX-IDs auf Systemen verschiedener Hersteller (z.B. die Group *bin* auf Ultrix- bzw. HP/Apollo- oder SunOS-Systemen) ein total konsistentes Account-Management in einem heterogenen Netzwerk unmöglich macht. In der Praxis sind aber ohnedies nur netzwerkweit einheitliche UNIX-IDs für User-Accounts (also Accounts für die primären Systembenutzer) vonnöten, soferne nämlich systemnahe Manipulationen auf UNIX-Hosts grundsätzlich nur direkt (lokal) an der jeweiligen Maschine erfolgen.

Das bedeutet, daß bei der Integration einer UNIX-Maschine in das Abteilungs-LAN die maschinenspezifischen, system-internen Einträge im originalen */etc/passwd*- und */etc/group*-File beibehalten und lediglich um die (aus dem Domain/OS-Registry gewonnenen) User-Entities erweitert werden müssen; in der Praxis erfolgt dies durch ein kleines (setuid-*root*) Programm. Im Sinne der Konsistenz des im Abschnitt 3.5 (*Filesystem und Protection*) vorgestellten, netzwerkglobalen Filesystems sind allerdings nur lokal bekannte User prinzipiell nicht erlaubt, außerdem sind, wie schon erwähnt, alle systemnahen Manipulationen direkt an der jeweiligen (UNIX-)Maschine durchzuführen!

Selbstverständlich erfordert diese Strategie jedoch die Einschränkung der umfangreichen Möglichkeiten des Domain/OS-Registrys zugunsten der UNIX-Konventionen. So können unter anderem nur Accounts mit der Abbreviation *p* (Person only) angelegt werden; außerdem verlieren die in Domain/OS verfügbaren Organizations ihre Bedeutung fast völlig.

Abgesehen von den diversen system-internen Persons, Groups, Organizations und Accounts existieren nun folgende "Standard-Entities":

---

* Bedingt durch die geringe Anzahl von nicht-Domain/OS-Maschinen ist momentan noch kein NIS-Server (Yellow Pages) vorgesehen.

Org	UNIX ID	(Def–) Group	UNIX ID	Person	UNIX ID	Bemerkungen
none	12	*inst*	1000	*inst*	1000	Abteilungsagenden/Muster–Abteilungsmitglied
				schi	1001	Prof. Schildt
				.	1002	.
				.		.
		dipl	2000	*dipl*	2000	Diplomanden/Muster–Diplomand
				glaser	2001	G. Glaser
				.	2002	.
				.		.
		prak	3000	*prak*	3000	Praktikanten/Muster–Praktikant
				huelble	3001	M. Huelble–Koenigsberger
				.	3002	.
				.		.
		staff	10	*staff*	1500	Staff/Muster–Staffmember
				s	1501	U. Schmid
				.	1502	.
				.		.

Persons in der (Default–)Group *inst* sind normale Mitglieder der Abteilung und demzufolge auch Members in *dipl* und *prak*; Mitglieder der Group *staff* haben darüberhinaus auch Systemverwalter–Funktionen und gehören *inst*, *dipl*, *prak* und *wheel* (su *root*!) an.

Die in der Tabelle oben angeführten "Muster–Persons" bzw. die zugehörigen Accounts (etwa *dipl[.dipl.none]*) sind übrigens ein recht brauchbarer Weg, um Templates für User–spezifische Files (etwa *.cshrc*) bereitzustellen. Derartige Templates erleichtern die Erstellung gewisser Installations–Scripts (z.B. für das Anlegen neuer Diplomanden) ganz wesentlich.

Für jeden (weiteren) Aufgabenbereich, etwa das Projekt VTA, muß nun eine eigene Organization und Group eingerichtet werden. Dabei ist es ist günstig, für beide dieselbe UNIX–ID zu vergeben; bei sehr umfangreichen Projekten mit abgrenzbaren Teilaufgaben sind allerdings auch mehrere verschiedene Groups unter ein und derselben Organization denkbar:

Org	UNIX ID	(Def–) Group	UNIX ID	Person	UNIX ID	Bemerkungen
vta	10000	*vta*	10000	*vta*	10000	Mitarbeiter VTA/Muster–Mitarbeiter
				k	10001	W. Kastner
				.	10002	.
				.		.
		staff	10	*jk*	10040	J. Klasek

Org	UNIX ID	(Def–) Group	UNIX ID	Person	UNIX ID	Bemerkungen
				.	10041	.
				.		.
.	10050	.	10050	.	10050	.
.		.		.		.

Da in der Regel zumindest die Person des Projektleiters einer anderen Default–Organization und –Group angehören wird (etwa *s.staff.none*), müssen die zugehörigen Membership–Lists natürlich um derartige Persons erweitert werden.

Die unbeschadet der erzwungenen UNIX–Konventionen noch nutzbaren Möglichkeiten des Domain/OS–Registrys erlauben es nun, die Account–Verwaltung eines konkreten Projektes nach Bedarf zu delegieren. Aufgrund einer geeigneten Vergabe der Rechte innerhalb des Registrys, nämlich

Registry Owner:	*root.staff.%*
Organization Owner:	*%.staff.%*
Group Owner:	*%.staff.%*
Person Owner:	*%.%.%*

können zunächst einmal die allgemeinen Registry–Policies/Properties nur vom Systemadministrator (*root.staff.none*) geändert werden. Das Anlegen neuer Groups und Organizations ist hingegen jedem Mitglied einer *staff*–Group erlaubt; neue Persons dürfen sogar (im Prinzip) von jedermann angelegt werden.

Die erwähnten Varianten der Account–Verwaltung eines konkreten Projektes ergeben sich nun durch die Möglichkeit der Weitergabe der Ownership der zugehörigen Projekt–Organization und –Group(s):

- *Account–Verwaltung exklusiv durch den Projektleiter*
  Die Ownership der Projekt–Organization und –Group(s) wird auf die SID des Projektleiters (z.B. *s.staff.%* oder *schi.inst.none*) gesetzt; Persons und Accounts können dann nur vom Projektleiter selbst verwaltet werden.

- *Account–Verwaltung durch (alle) Mitglieder der staff–Group*
  Die Ownership der Projekt–Organization und –Group(s) wird auf *%.staff.%* gesetzt; dem für die Account–Verwaltung zuständigen Mitarbeiter muß natürlich ein Account mit der Default–Group *staff* eingerichtet werden. Auf diese Weise kann das Person– und Account–Management eines Projektes vollständig delegiert werden, ohne dem Projektleiter (bzw. gar dem Systemadministrator) die Möglichkeit des Eingriffes zu nehmen.

Natürlich muß auch die Ownership der im Zuge des Account–Managements eines Projektes angelegten Persons entsprechend (also etwa gleich der jeweiligen Projekt–Organization) gesetzt werden. Übrigens ist die Verwaltung der Organizations nur wegen der erwähnten Registry–Protection erforderlich, die notwendige UNIX–Kompatibilität wird dadurch in keinster Weise berührt.

Im Zusammenhang mit den zuvor vorgestellten "Standard-Entities" ist zu bemerken, daß deren Ownership auf *%.staff.none* (statt auf *%.staff.%*) gesetzt wird, wodurch Manipulationen durch ein Mitglied der *staff*-Group eines Projektes (z.B. *jk.staff.vta*) ausgeschlossen sind.

Ein an dieser Stelle noch nicht behandelbares Problem stellt das Management systemweit gültiger Homedirectories für die einzelnen Accounts dar. Hierbei treten Probleme mit den Namenskonventionen von Domain/OS bzw. NFS auf, und zwar unter anderem dann, wenn das Login eines Users mit einem Homedirectory auf einer nicht-Domain/OS-Maschine auf einem Domain/OS-Node erfolgt (oder vice versa). Der entsprechende Lösungsansatz ist im Abschnitt 3.5 (*Filesystem und Protection*) zu finden.

3.3 TCP/IP

Wie schon mehrfach erwähnt, basieren die für die UNIX-Kompatibilität vorgesehenen Mechanismen auf TCP/IP, weshalb alle Maschinen im Abteilungs-LAN (ausgenommen die im Instituts-Ring befindlichen PCs) reguläre TCP/IP-Hosts sind. Alle Domain/OS-Maschinen benötigen dazu einen dedizierten *tcp_admin*-Node (*aa*, siehe dazu auch Abschnitt 3.1 (*Domain/OS*)), der zentralisiert bestimmte Konfigurationsfiles bereitstellt und dadurch die Verwaltung ganz wesentlich vereinfacht. Aus Redundanzgründen steht übrigens auch die Maschine *auto* als alternativer *tcp_admin*-Node zur Verfügung.

Natürlich ist das Abteilungs-LAN über das TU-Net am (weltweiten) Internet angeschlossen. Das TU-Net an sich ist ein offiziell registriertes (Class-B) Netz mit der IP-Adresse 128.130.xx.xx, welches als Domain *tuwien.ac.at.* zentralisiert vom Rechenzentrum der TU-Wien betreut wird. Es unterstützt Subnetze auf der Basis der (etwas ungewöhnlichen) Subnet-Mask 0xffffff80.

Den Subnetzen des Abteilungs-LANs wurden nun folgende IP-Adressen (Subnetz-Nummern) zugeordnet:

(1) *a_eth*: 128.130.55.129 - 128.130.55.254

(2) *a_ring*: 128.130.41.1 - 128.130.41.126

(3) *vta_eth*: 128.130.55.1 - 128.130.55.126

In Punkto TCP/IP-Naming-Service bildet das Abteilungs-LAN eine eigene Subdomain *auto.tuwien.ac.at.*. Leider werden derzeit alle (Sub-)Domains an der TU-Wien zentralisiert* von den TU-Net Naming-Servern betreut; auf der Maschine *auto* befindet sich lediglich ein Secondary für die lokale Domain. Alle anderen Hosts nehmen das remote Naming-Service der *auto* (und der TU-Net Naming-Server) in Anspruch. Im Falle der Nichtverfügbarkeit derselben erfolgt die Hostname/IP-Address-Resolution mit Hilfe der */etc/hosts*-Files, welche aber nur die im Abteilungs-LAN befindlichen Hosts enthalten.

---

* Dadurch ist es unter anderem notwendig, jeden einzelnen Host am TU-Rechenzentrum registrieren (also eintragen) zu lassen.

Tatsächlich sind nun folgende (potentielle) Gateways/Naming-Server vorhanden:

- *auto*
  Diese Maschine fungiert als Gateway zwischen dem TU-Netz, dem Abteilungs-Ethernet und dem Abteilungs-Ring und darüberhinaus als secondary Naming-Server für die Domain *auto.tuwien.ac.at.*.

- *newton*
  Diese Maschine bildet das Gateway zwischen dem VTA-Ethernet und dem Abteilungs-Ring.

- *galilei*
  Da auch diese Workstation mit einem Ring- und einem Ethernet-Controller ausgerüstet ist, kann sie im Bedarfsfall analog zum Node *newton* konfiguriert werden.

- *server2*
  Über diese Maschine ist eine TCP/IP-Verbindung zum (nicht offiziell registrierten) Übungs-Ring möglich; allerdings findet hier keinerlei Routing statt.

Von den auf TCP/IP aufbauenden, elementaren DARPA-Services wird aus Sicherheitsgründen nur das (für X-Windows notwendigen) *telnet* und das *rexec* zugelassen, wobei grundsätzlich alle Login-Versuche mittels *syslogd* aufgezeichnet werden. Über die Maschine *auto* sind darüberhinaus auch File-Transfers mittels *ftp* möglich. Dadurch ist natürlich zu Beginn jeder remote Session ein vollständiges Login erforderlich; die lediglich auf */etc/hosts.equiv* aufbauende Authentisierung für das "bequemere" *rlogin* oder *rsh* würde aber ein untragbares Sicherheitsrisiko darstellen.

Auf diese Weise sind also alle Voraussetzungen geschaffen, die die Bereitstellung TCP/IP-basierender höherer Services (etwa NFS und X-Windows) erlauben.

3.4 Printer-Service

Das Printer-Service innerhalb des Abteilungs-Ringes basiert auf dem komfortablen *prmgr/prsvr*-Konzept von Domain/OS. Darüberhinaus wird im gesamten Abteilungs-LAN aber auch das auf *lpr/lpd* basierende UNIX (BSD) Service voll (daß heißt, für alle Drucker) unterstützt.

Da aufgrund eines Software-Fehlers in Domain/OS SR10.3 das (dafür eigentlich vorgesehene) Konzept eines *lpd*-Spoolnodes nicht funktioniert, muß auch auf jeder Domain/OS-Maschine ein eigener *lpd* laufen. Ähnlich wie im Zusammenhang mit den *tcp_admin*-Nodes kommt aber einem Node (*aa*) insoferne eine besondere Bedeutung zu, als dieser den *lpd's* auf den anderen Maschinen im Abteilungs-Ring das benötigte */etc/printcap*-File zur Verfügung stellen muß. Durch die

Angabe entsprechender *prf*-Kommandos in diesem File können nun alle *prmgr/prsvr*-unterstützten Drucker quasi für *lpr/lpd* "exportiert" werden. Konkret handelt es sich dabei um folgende *lpd*-(Pseudo-)Printer:

epl	EPL 7500
epl_ps	EPL 7500 Postscript (Transparent Mode)
imagen	IMAGESTATION/S
imagen_imp	IMAGESTATION/S Impress (Transparent Mode)
fujitsu	Fujitsu DL2400

Alle diese Drucker können im */etc/printcap*-File eines z.B. im Abteilungs-Ethernet befindlichen UNIX-Hosts als remote Printer an irgend einer Domain/OS-Maschine (etwa der *aa*) deklariert und daher mittels *lpr* angesrochen werden. Voraussetzung dafür ist natürlich, daß */etc/hosts.lpd* des jeweiligen Printerserver-Hosts den anfordernden Host enthält; siehe dazu auch die Bemerkungen im Abschnitt 3.3 (*TCP/IP*).

3.5 Filesystem und Protection

Die Brauchbarkeit eines Netzwerkes hängt natürlich in ganz besonderer Weise von der Flexibilität des Filesystems ab. In diesem Zusammenhang sind unter anderem folgende Ziele zu berücksichtigen:

(1) *Netzwerkweit einheitliche Struktur*
Basierend auf den guten Erfahrungen mit dem in Domain/OS verwendeten Konzept eines netzwerkweiten Rootdirectories //, in dem alle Nodes als top-level Directories eingetragen sind, wurde dieser Ansatz auch in Hinblick auf NFS adaptiert. Um also auch im Rahmen der UNIX-Funktionalität netzwerkweit identische File-Namen bereitstellen zu können, existiert auf jeder Maschine im Abteilungs-LAN folgende Directory-Struktur:

Dir	**Subdir**	**Sub[2] dir**	**Domain/OS-Nodes**	**UNIX-Hosts**
/NET	*a_eth*	*auto*	Link auf *//auto*	Link auf */NET/a_ring/auto*
		xyz	*auto*: Mount-Point *xyz*:/[2]) sonst: Link auf *//auto/NET/a_eth/xyz*	*xyz*: Link auf / sonst: Mount-Point *xyz*:/[1])
		.		
		.		
	a_ring		Link auf //	Mount-Point //[3])
		auto		
		aa		
		newton		

Dir	Subdir	Sub^2 dir	Domain/OS–Nodes	UNIX–Hosts
		.		
		.		
	vta_eth	*newton*	Link auf *//newton*	Link auf */NET/a_ring/newton*
		vta_host	*newton*: Mount–Point *vta_host:/*[2] sonst: Link auf *//newton/NET/a_eth/vta_host*	*vta_host*: Link auf / sonst: Mount–Point *vta_host:/*[1]
		.		
		.		

Hierzu ist folgendes zu bemerken:

1) Ein äußerst unangenehmer grundsätzlicher Nachteil von NFS besteht in der Tatsache, daß alle an einem NFS–Server gemounteten Filesysteme (egal ob lokal* oder remote) für einen NFS–Client unsichtbar bleiben. Manche Implementierungen von NFS (z.B. Sun–NFS) erlauben es deshalb, die "fehlenden" Filesysteme des Servers an den "leeren Ästen" des bereits gemounteten Filesystems (lokal am Client!) erneut zu mounten, wodurch die gesamte Directory–Struktur des Servers verfügbar gemacht wird.

2) Einige NFS–Implementierungen (unter ihnen leider auch Domain NFS) stellen die im vorigen Punkt erläuterte Möglichkeit nicht zur Verfügung. Die einzig mögliche Abhilfe besteht darin, das jeweilige Directory (z.B. */NET/vta_eth/vta_host*) als solches zu belassen und darin weitere Subdirectories anzulegen, die dann als Mount–Points für alle benötigten Filesysteme (und Directories) des remote Hosts (*vta_host*) dienen. Um aber auf diese Weise wirklich dessen gesamtes Rootdirectory / verfügbar zu machen, sind in der Regel sehr viele einzelne Mounts notwendig.

   Im Zusammenhang mit den Domain/OS–Maschinen im Abteilungs–Ring genügt es hier natürlich, die gerade besprochene Struktur nur auf der jeweiligen Gateway–Maschine (*newton*) einzurichten; für die anderen Nodes genügt ein Link darauf.

3) Eine ganz wesentliche organisatorische Vereinfachung ergibt sich hier durch die Tatsache, daß alle UNIX–Hosts das Network–Rootdirectory // des Abteilungs–Ringes mounten können; // kann dabei von jedem Domain/OS–Node mit NFS exportiert werden. Sinnvollerweise stellen daher auch nur die Gateways *auto* und *newton* NFS–Server bereit, die jeweils nur alle "direkt" angeschlossenen UNIX–Hosts bedienen.

---

* Domain/OS Server–Maschinen (glücklicherweise) ausgenommen!

Das File */etc/exports* eines Hosts muß also nur das jeweilige Rootdirectory (Domain/OS: //, UNIX: /) und, bei UNIX-Servern, eventuelle zusätzliche* Filesysteme exportieren, und zwar für den (gewöhnlichen) Zugriff von allen Hosts im Abteilungs-LAN aus. Die Möglichkeit eines *root*-Zugriffes wird dabei aber prinzipiell nicht eingeräumt.

Im Zusammanhang mit der praktischen Verwendbarkeit dieses Konzeptes ist es natürlich wichtig, in Links oder anderen (netzwerkweiten) Referenzen die netzwerkglobalen File-Namen (etwa */NET/a_ring/auto/...*) zu verwenden. Daß hierfür aber die Mount-Struktur auf allen Hosts konsistent sein muß, versteht sich von selbst.

Im übrigen ist zu beachten, daß NFS lediglich Bytestream-Files kennt, wodurch Kompatibilitätsprobleme mit den in Domain/OS exisiterenden typed Files auftreten können. Infolgedessen ist vor allem die Verwendung von Domain/OS-Software in Kombination mit remote NFS-Files mit Vorsicht zu genießen.

(2) *Effiziente Organisations-Struktur*
Von zentraler Bedeutung ist natürlich die Entwicklung einer brauchbaren "logischen" Directory-Struktur. Die wichtigsten Anforderungen sind hier

- *Working-Directories an der üblicherweise benutzten Maschine*
  Um das Netzwerk zu entlasten, ist es - netzwerkweites Filesystem hin oder her - sinnvoll, wenn ein Benutzer möglichst viel auf der lokalen Platte seiner Maschine arbeiten kann.

- *Bereitstellung eines "globalen Blickes"*
  Trotz der (durch die Beachtung des vorigen Punktes nicht zu verhindernden) Dezentralisierung der Daten muß eine zusammenfassende Sicht auf alle Abteilungs-Angelegenheiten (Projekte, User, Praktika+Diplomarbeiten, ...) gewährleistet sein.

- *Minimal Toolset*
  Die von den einzelnen Usern verwendeten Tools sollen an sich nicht ungebührlich** eingeschränkt werden. Allerdings ist dadurch für globale Datenbestände (z.B. das Inhaltsverzeichnis der Projektberichte) eine Art "kleinster gemeinsamer Nenner" erforderlich, etwa ASCII- oder TeX-Files.

Zu Lösung dieser widersprüchlichen Forderungen wurde eine Organisations-Struktur realisiert, deren Kern eine zentrale, auf der Maschine *aa* befindliche Directory-Struktur ist:

---

* Daher ist es ratsam, zusätzliche Filesysteme (vor allem mehrere Partitions auf einer Disk) wo es geht zu vermeiden!

** Hiermit soll aber keinesfalls einer willkürlichen Verwendung aller möglichen Tools das Wort geredet werden. In Sinne der Integration der gesamten Abteilung ist ganz im Gegenteil die Beschränkung auf einige wenige (Standard-)Software-Systeme sicherlich notwendig.

Dir	Subdir	Sub^2 dir	Sub^3 dir	Bemerkungen
/ABT	*TEMPLATES*			Global verwendbare Templates (TU-Table, ...)
	PA			*Prozeßautomatisierung*
		vorlesung		Enthält (vor allem) Links auf lokale Directories
		uebung		Enthält (vor allem) Links auf lokale Directories
	INFO1			*Einführung in die Informatik I*
		vorlesung		Enthält (vor allem) Links auf lokale Directories
		uebung		Enthält (vor allem) Links auf lokale Directories
	PRAK+DIPL	*templates*		Allgemeine Praktika+Diplomarbeiten/Templates
		offene		offene Themen
		aktuelle		in Bearbeitung befindliche Themen
		fertige		fertiggestellte Praktika/Diplomarbeiten
	PROJEKTE			Abteilungs-Projekte
		VTA		(Link auf) Projektdirectory VTA
		.		.
		.		.
	REPORTS			Projektberichte (Schriftenreihe)
		net_e		(Link auf) Projektbericht *net*
		.		.
		.		.
	USER			(Nicht-projektspezifische) User
		inst	*inst*	Abteilungsmitglieder/Muster-Abteilungsmitglied
			schi	(Link auf) Homedirectory Prof. Schildt
			.	.
			.	.
		dipl	*dipl*	Allgemeine Diplomanden/Muster-Diplomand
			glaser	(Link auf) Homedirectory G. Glaser
			.	.
			.	.
		prak	*prak*	Allgemeine Praktikanten/Muster-Praktikant
			huelble	(Link auf) Homedir. M. Huelble-Koenigsberger
			.	.
			.	.
		staff	*staff*	Staffmembers/Muster-Staffmember
			s	(Link auf) Homedirectory U. Schmid
			.	.
			.	.
/LOCAL				Lokale Directories der Maschine *aa*

Auf jedem anderen Host befindet sich eine völlig gleichartig aufgebaute Directory-Struktur:

Dir	Subdir	Sub^2 dir	Bemerkungen
/ABT	*TEMPLATES*		Link auf /NET/a_ring/aa/ABT/TEMPLATES
	PA		Link auf /NET/a_ring/aa/ABT/PA
	INFO1		Link auf /NET/a_ring/aa/ABT/INFO1
	PRAK+DIPL		Link auf /NET/a_ring/aa/ABT/PRAK+DIPL
	PROJEKTE		Lokale Abteilungs-Projekte
		VTA	(Link auf) Projektdirectory VTA
		.	.
		.	.
	REPORTS		Link auf /NET/a_ring/aa/ABT/REPORTS
	USER		Lokale (und nicht-projektspezifische) User
		inst	Enthält Homedirectories lokaler Abteilungsmitglieder
		dipl	Enthält Homedirectories lokaler Diplomanden
		prak	Enthält Homedirectories lokaler Praktikanten
		staff	Enthält Homedirectories lokaler Staffmembers
/LOCAL			Lokale Directories der jeweiligen Maschine

Zu beachten ist, daß im Unterschied zur zentralen Struktur auf der *aa* in den Subdirectories */ABT/PROJEKTE* und */ABT/USER* nicht alle, sondern nur die auf der jeweiligen Maschine lokal zu findenden Projekte bzw. User enthalten sind.

Während also der Aufbau von */ABT* vorgegeben (und darüberhinaus auch auf allen Maschinen gleich) ist, liegt die Organisation von */LOCAL* völlig im Ermessen des Owners/Users der jeweiligen Maschine. Natürlich versteht sich das Ganze aber lediglich als "konzeptuelles Gerüst"; die vorgestellte Struktur kann also je nach Bedarf beliebig erweitert bzw. modifiziert werden. Ihr weitergehender Aufbau ist darüberhinaus auch wesentlich komplexer, vor allem im Zusammenhang mit dem Management von Projekten; auf eine detaillierte Erläuterung muß jedoch aus Platzgünden verzichtet werden.

(3) *Netzwerkweit eindeutige Homedirectories*
Basierend auf der bisher vorgestellten Struktur ist es natürlich einfach möglich, netzwerkweit gültige Homedirectories im Registry einzutragen (die ja dann auch in */etc/passwd* übernommen werden); siehe dazu auch Abschnitt 3.2 (*Account-Management*). Dem auf dem Host *zuse* im Abteilungs-Ring befindlichen Abteilungsmitglied *schi.inst.none* könnte etwa das Homedirectory */NET/a_ring/zuse/ABT/USER/inst/schi* zugeordnet werden.

(4) *Dichte Protection*
Die bei weitem unangenehmste Konsequenz der netzwerkweiten UNIX-Kompatibilität liegt in der Preisgabe der mächtigen ACL-Protections von Domain/OS begründet. Die generelle Protection muß daher auf die schwachen BSD-Mechanismen zurückgreifen, wobei im Normalfall folgende Konventionen eingehalten werden sollten:

Owner: -rwx- (Inheritance by Process)
Group: -r-x- (Inheritance by Parent Initial File/Directory)
Others: -----

Diese Rechte (und natürlich der jeweilige Group-Owner) sind in den Initial File- und Initial Directory-ACLs des jeweiligen Parent-Directorys zu setzen. Selbstverständlich können dabei gewissen Groups, etwa *inst*, auch modifizierte Group-Rechte (z.B. -rwx-) zugeordnet werden.

(5) *Backup*
Durch das Vorhandensein einer optischen Platte im Netzwerk erfolgt die Datensicherung natürlich hierauf. Automatisch (und periodisch) gesichert wird prinzipiell der gesamte Tree */NET/a_ring/aa/ABT*. Deshalb ist es wichtig, diese in Punkt (2) vorgestellte globale Struktur auf dem laufenden zu halten: Wird etwa auf irgend einer Maschine ein Homedirectory für ein neues Abteilungsmitglied angelegt, so wird dieses automatisch ab dem Moment mitgesichert, ab dem der Link in */NET/a_ring/aa/ABT/USER/inst* eingetragen wird.

3.6 X-Windows

Alle im Abteilungs-LAN integrierten Hosts sind grundsätzlich vollwertige Server (und natürlich auch Clients) für X-Windows. Da jeder solche Host im */etc/X0.hosts*-File eines jeden Hosts eingetragen ist, kann X-Windows (in Verbindung mit *telnet*) über das gesamte Netzwerk hinweg transparent verwendet werden.
Domain/OS-Nodes stellen übrigens einen speziellen shared-mode X-server (Release X11.3) bereit, der die gleichzeitige Verwaltung von DM- und X-Windows erlaubt. Dieser wird im DM-owned Root Mode betrieben, sodaß im Normalfall (also wenn keine X-Applikation ein X-Window auf der jeweiligen Maschine erzeugt hat) von X-Windows nichts zu bemerken ist.

3.7 Mail

Das Mail-Konzept im Abteilungs-LAN basiert im Prinzip auf einem zentralen Mailbox-Server (Mail-Spoolnode) in Verbindung mit *sendmail*-Daemons auf allen Hosts. Das bedeutet zunächst einmal, daß grundsätzlich alle Hosts SMTP-Connections für hereinkommende Mail zulassen. Da nun das Mailbox-Directory */usr/spool/mail* eines (normalen) Hosts nur ein Link auf das "globale" Mailbox-Directory */NET/a_ring/aa/usr/spool/mail* am Mail-Spoolnode *aa* ist, kann die Mail an jede beliebige Maschine gesendet werden: sie landet trotzdem in der (eindeutigen) Mailbox des jeweiligen Users und ist darüberhinaus auch von jeder beliebigen Maschine aus zu lesen.

Für lokale (also Abteilungs-interne) Mail sind daher folgende Mail-Adressen, etwa für den User *schi*, zulässig:

- *schi@auto*
- *schi@zuse*
- *schi*

Vom Internet aus (also von "außen") sind hingegen Adressen wie

- *schi@auto.tuwien.ac.at.*
- *schi@zuse.auto.tuwien.ac.at.*

erforderlich. Die erstere Variante rechnet übrigens damit, daß das Gateway *auto* auch vom TU-Net aus (also von "außen") unter *auto.tuwien.ac.at.* zu erreichen ist. Statt der Maschine *zuse* in der zweiteren Mail-Adresse wäre natürlich auch die Angabe jeder beliebigen anderen Maschine im Abteilungs-LAN zulässig.

Falls im Zusammenhang mit dem Management eines (internen) Projektes eine "projektorientierte" Adressierung gewünscht sein sollte, so kann einer (Projekt-)Maschine der (Alias-)Hostname des Projektes gegeben werden. So hat etwa die Gateway-Maschine zum VTA-Ethernet (*newton*) auch den Alias *vta*, wodurch die Mail an den Projektmitarbeiter *jk* unter anderem an folgende Adressen geschickt werden kann:

- *jk@vta.auto.tuwien.ac.at.*
- *jk@vta_host.auto.tuwien.ac.at.*
- *jk@auto.tuwien.ac.at.*

4. Systemadministration

Die Aufgabe dieses Abschnittes ist es, kurz einige grundlegende Aspekte der Systemadministration vorzustellen. Bei der Entwicklung des zugrundeliegenden Konzeptes wurde besonders auf die Reproduzierbarkeit der Installation Wert gelegt: Falls eine Maschine ausfallen sollte, so muß sie (bzw. eine allfällige Ersatzmaschine) ohne allzu großen Aufwand wieder re-installiert werden können. Nun ist aber das Backup der gesamten Disk(s) einer Maschine in der Regel problematisch (und darüberhinaus auch sehr zeitaufwendig), weshalb hier ein anderer Weg beschritten wurde.

Der gesamte Prozeß der Installation einer Maschine ist zunächst einmal in zwei Phasen gegeliedert, und zwar

- *Installation der System-Software*
  Im Zuge dieser Phase wird eine Maschine soweit installiert, daß sie alle für das Netzwerk notwendigen Funktionen wahrnehmen kann. Für jede einzelne Maschine existiert hier ein vollständiges Protokoll aller jener Tätigkeiten, die zur vollständigen Installation der jeweiligen System-Funktionalität (siehe

Anhang A (*Gesamtkonfiguration*)) notwendig sind. Für dessen Erstellung und Verwaltung ist der Systemadministrator zuständig.

- *Installation der User-Software*
  Während dieser Phase, für die bereits der Owner der jeweiligen Maschine zuständig ist, wird auf einer Maschine die (vom ihm) gewünschte User-Software installiert. Hier gibt es - falls überhaupt erforderlich - für jedes Produkt eine eigene Anleitung, die von demjenigen erstellt wird, der die Erstinstallation durchführt.

Von den erwähnten Installations-Protokollen bzw. -Anleitungen müssen übrigens zwei verschiedene Arten bereitgestellt werden, nämlich

- Erstinstallations-Protokolle/Anleitungen (*virgin installation*)
- Folgeinstallations-Protokolle/Anleitungen (*nonvirgin installation*)

Für das prinzipielle Verständnis der diversen Protokolle erscheint es notwendig, kurz auf einige interne Details der Systemadministration einzugehen. Auf jeder Maschine existiert ein Directory */install/site_config*, welches zunächst einmal die für die Basis-Installation der System-Software notwendigen Konfigurations-Files enthält. Darüberhinaus bildet ein darin enthaltener Directory-Tree das Root-Directorys / der jeweiligen Maschine (allerdings meist unvollständig) nach, wodurch er Kopien aller jener Files aufnehmen kann, die von den im Zuge der Standard-Installation im Directory-Tree / der jeweiligen Maschine angelegten abweichen (also geändert oder zusätzlich erzeugt wurden). Diese Kopien werden mit Hilfe spezieller Sicherungs-Tools erzeugt, die auch die originalen ACLs* mit übernehmen. Dadurch ist es möglich, anschließend an eine Standard-Installation mit Hilfe eines einzelnen Kommandos alle modifizierten Files nach / zu kopieren und somit "in Kraft" zu setzen.

Zusätzlich zu den lokalen, nur die Konfiguration des jeweiligen Nodes enthaltenden site_config-Directories existiert auf den "zentralen" Maschinen *aa* und *auto* je ein vollständiges */install/site_config* (also eines, das für jede Maschine im Abteilungs-Ring eine Kopie des jeweiligen site_config-Trees enthält); die vorhin erwähnten Tools verwalten diese Struktur selbständig, sie kopieren also zu sichernde Files sowohl in das lokale site_config-Directory als auch in das der *aa* und der *auto*. Dadurch ist es einfach, das lokale Directory jedes beliebigen Nodes bei Bedarf zu restaurieren.

In Anbetracht der geringen Anzahl der momentan im Abteilungs-LAN vorhandenen UNIX-Hosts ist die Verwaltung einer entsprechenden (vollständigen) site_config-Struktur (noch) nicht sinnvoll. Geeignete Kandidaten für deren spätere Bereitstellung wären natürlich allfällige NIS-Server, die allerdings auch einen remote *root*-Access über NFS erlauben müßten.

In Wirklichkeit ist es nun sogar notwendig, mehrere verschiedene Sicherungs-Trees parallel zu verwalten. Im Zuge der Installation einer User-Software

---

* Während dies bei Domain/OS-Nodes problemlos möglich ist, ist bei UNIX-Hosts etwas Vorsicht geboten.

ist es nämlich manchmal notwendig, irgendwelche System-Files (z.B. */etc/rc.user*) zu modifizieren. Würden diese nun mit den vorhin erwähnten Tools gesichert, so wären sie für eine spätere System-Installation wertlos, da sie ja bereits die "Existenz" der entsprechenden User-Software voraussetzen!

Aus diesem Grunde sind drei verschiedene Sicherungs-Trees vorgesehen: *system*, *low* und *high*. Diese Aufspaltung trägt einer zugegebenermaßen unter Umständen etwas willkürlichen, aber organisatorisch sinnvollen Gliederung der User-Software in low-level (allfällige Compiler, gmr2d, OSF-Motif, ...) und high-level (TeX, Context, Maple, ...) Rechnung. Zusätzlich zu den Trees existieren natürlich auch drei verschiedene Sicherungs-Toolsets, die jeweils in ihre zugehörige Directory-Struktur (*system*, *low* oder *high*) sichern.

Hier ergibt sich übrigens ein organisatorisches Problem, und zwar bedingt durch die Tatsache, daß jegliche Software-Installation für gewöhnlich als *root* durchgeführt werden muß. Dadurch steigt die Gefahr, aus Unachtsamkeit einmal ein falsches Sicherungs-Tool zu erwischen. Ein sehr einfacher - leider aber auch sehr schwacher - Schutzmechanismus besteht darin, im *.login* der Homedirectories* von *root* eine Abfrage (System-, low- oder high-level User-Installation) vorzusehen und abhängig davon in verschiedene "Pseudo-Homedirectories" zu gehen.

5. Erweiterungsmöglichkeiten

Natürlich eröffnet das vorgestellte Konzept eine ganze Reihe von Erweiterungsmöglichkeiten; einige davon wären:

- *Einbindung von PCs über PC-NFS*
  Momentan ist eine Integration von MS-DOS PCs nur im Abteilungs-Ring möglich, siehe auch Abschnitt 3.1 (*Domain/OS*). Für PCs, die mit einem Ethernet-Controller ausgerüstet sind, bietet sich hingegen die Integration im Abteilungs-Ethernet (unter Verwendung von PC-TCP/NFS und der Maschine *auto* als Server) an.

- *Domain/OS Routing*
  Im Zusammenhang mit weiteren Domain/OS-Maschinen ist es unter Umständen günstiger, diese nicht mit Ring-, sondern mit Ethernet-Controllern auszurüsten und in das Abteilungs-Ethernet einzubinden. Auch kann es aus Performance-Gründen notwendig sein, die Maschinen eines allfälligen weiteren (internen) Forschungsprojektes in ein eigenes Segment (Ring oder Ethernet) auszulagern. In jedem Falle müßten dann Domain/OS Netzwerk-Nummern vergeben und das jeweilige Gateway als DDS-Router konfiguriert werden.

- *NIS-Service (Yellow Pages)*
  Sollte im Laufe der Zeit die Anzahl der UNIX-Maschinen (vorzugsweise Sun) größer werden, so empfiehlt sich aus administrativen Gründen die Einführung von NIS-Services. Dabei sollte im Endeffekt eine Struktur ähn-

* Zu beachten ist ja, daß *root* auf jedem UNIX-Host ein eigenes Homedirectory hat; lediglich im Abteilungs-Ring gibt es ein netzwerkweit eindeutiges Homedirectory.

lich der des Abteilungs-Ringes angepeilt werden, die für derartige Dienste zwei "zentrale" Hosts vorsieht.

- *UUCP-Connections*
  Für externe Forschungsprojekte, bei denen aus irgendwelchen Gründen TCP/IP-Verbindungen ausscheiden, ist UUCP natürlich eine brauchbare Alternative. Allerdings ist zu beachten, daß "öffentlich" zugängliche UUCP-Connections in das Abteilungs-LAN ein beträchtliches Sicherheitsrisiko darstellen, vor allem bei unzureichender Administration!

6. Danksagungen

Es ist an sich beinahe müßig, zu erwähnen, daß die Entwicklung, vor allem aber die Realisierung des vorgestellten Gerätekonzeptes, ohne die Unterstützung bzw. die tätige Mitarbeit so vieler Kollegen nicht möglich gewesen wäre.

An erster Stelle danken möchte ich Prof. Schildt für das in mich gesetzte Vertrauen - und die Erlaubnis, seine Berufungsmittel mit vollen Händen auszugeben; ohne ihn wäre ich wohl nie in die Lage gekommen, ein derartiges Konzept ausarbeiten und realisieren zu können.

Besonders hervorzuheben ist in diesem Zusammenhang auch die ausgezeichnete Zusammenarbeit mit dem BMWF, und zwar vor allem die wohlwollende - und unbürokratische - Abwicklung der Gerätefinanzierung durch Herrn Dr. Kolarsky; seinem Entgegenkommen ist es zu danken, daß die normalerweise nur längerfristig bereitstehenden Berufungsmittel im Laufe von nur zwei Jahren zur Verfügung gestellt werden konnten.

Schließlich bleibt noch, allen jenen zu danken, die aktiv an der Realisierung des Abteilungs-LANs beteiligt waren: Meinem Kollegen Stefan Stöckler für seine Mithilfe bei der Entwicklung des Konzeptes und, nicht zuletzt, für die - oft nächtelangen - gemeinsamen Systemadministrations-Tätigkeiten; unserem Techniker Christian Kral für die Installation und Betreuung der Hardware; vor allem aber Günter Glaser und Klaus Schossmaier für die ungezählten Stunden, die sie der Software-Installation und deren Dokumentation geopfert haben.

7. Literatur

[M1] **HP/Apollo Domain/OS Manual,** *"Managing BDS System Software",* Order Number 010853-A00.

[M2] **HP/Apollo Domain/OS Manual,** *"Configuring and Managing TCP/IP",* Order Number 008543-A02.

[M3] **HP/Apollo Domain/OS Manual,** *"Administering the Domain/OS Registry",* Order Number 015363-A00.

[M4] **HP/Apollo Domain/OS Manual,** *"Using NFS on the Domain Network",* Order Number 010414-A00.

[M5] **HP/Apollo Domain/OS Manual,** *"Using the X Window System on Apollo Workstations",* Order Number 015213-A02.

[M6] **HP/Apollo Domain/OS Manual,** *"Installing Domain Software with Apollo's Release and Installation Tools",* Order Number 008860-A02.

[S1] **Sun SunOS Manual**, *"System and Network Administration"*, Rev. A, Part Number 800-3805-10

[SS] **U. Schmid, S. Stöckler**, *"Konzept der Laborübung Prozeßautomatisierung"*, Projektbericht des Instituts für Automation, TU-Wien, Nr. 183/1-21, 1991.

A. Gesamtkonfiguration

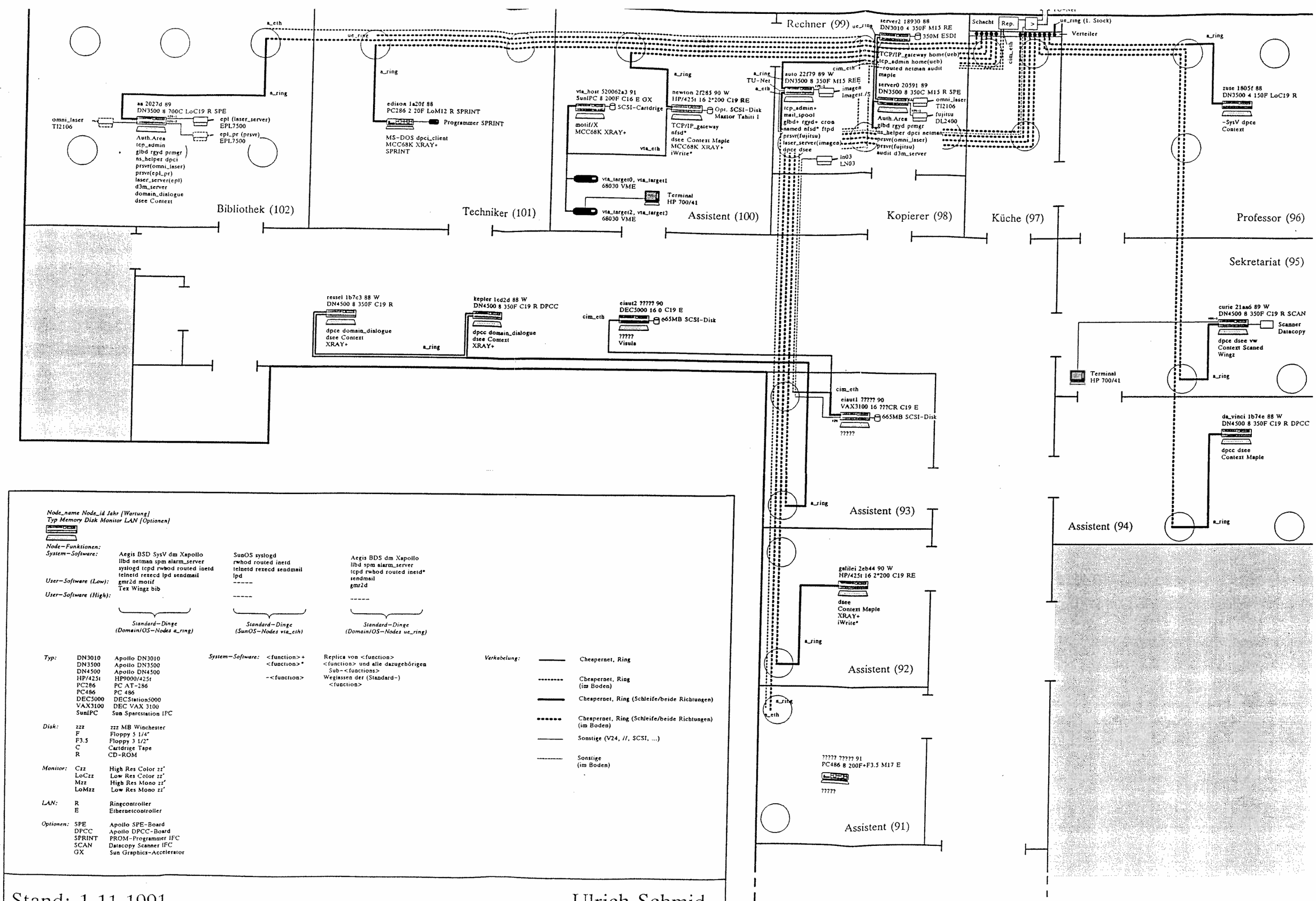
Im Anhang finden sich noch einige Plots der Gesamtkonfiguration des Abteilungs-LANs, und zwar

- *Gesamtdarstellung*

- *Maschinenkonfiguration*

Basis derselben ist ein in Multilayer-Technik erstellter "Grundriß" der Abteilungs-Räumlichkeiten, in dem alle Maschinen, Peripheriegeräte und Verkabelungen eingezeichnet wurden. Ebenfalls eingetragen ist natürlich die genaue Hard- und Software-Konfiguration jeder einzelnen Maschine.

Daß dieser Plan, ebenso wie die zur Systemadministration gehörende sonstige Dokumentation einer konsequenten Aktualisierung durch den Systemadministrator bedarf, liegt auf der Hand - mindestens ebenso wie die Tatsache, daß die Abteilung 183/1 überhaupt eines Systemadministrators bedarf!



Node_name Node_id Jahr [Wartung]
Typ Memory Disk Monitor LAN [Optionen]

Node-Funktionen:

	Aegis-Dinge (Domain/OS-Nodes a_ring)	SunOS-Dinge (SunOS-Nodes vta_eth)	Aegis-Dinge (Domain/OS-Nodes ue_ring)
System-Software:	Aegis BSD SysV dm Xapollo llbd netman spm alarm_server syslogd tcpd rwhod routed inetd telnetd rexecd lpd sendmail	SunOS syslogd rwhod routed inetd telnetd rexecd sendmail lpd -----	Aegis BDS dm Xapollo llbd spm alarm_server tcpd rwhod routed inetd* sendmail gmr2d
User-Software (Low):	gmr2d motif Tex Wingz bib	-----	-----
User-Software (High):			

Typ:

DN3010	Apollo DN3010
DN3500	Apollo DN3500
DN4500	Apollo DN4500
HP/425t	HP9000/425t
PC286	PC AT-286
PC486	PC 486
DEC5000	DECStation5000
VAX3100	DEC VAX 3100
SunIPC	Sun Sparcstation IPC

Disk:

zzz	zzz MB Winchester
F	Floppy 5 1/4"
F3.5	Floppy 3 1/2"
C	Cartdrige Tape
R	CD-ROM

Monitor:

Czz	High Res Color zz'
LoCzz	Low Res Color zz'
Mzz	High Res Mono zz'
LoMzz	Low Res Mono zz'

LAN:

R	Ringcontroller
E	Ethernetcontroller

Optionen:

SPE	Apollo SPE-Board
DPCC	Apollo DPCC-Board
SPRINT	PROM-Programmer IFC
SCAN	Datacopy Scanner IFC
GX	Sun Graphics-Accelerator

System-Software:

<function>+	Replica von <function>
<function>*	<function> und alle dazugehörigen Sub-<functions>
-<function>	Weglassen der (Standard-) <function>

Verkabelung:

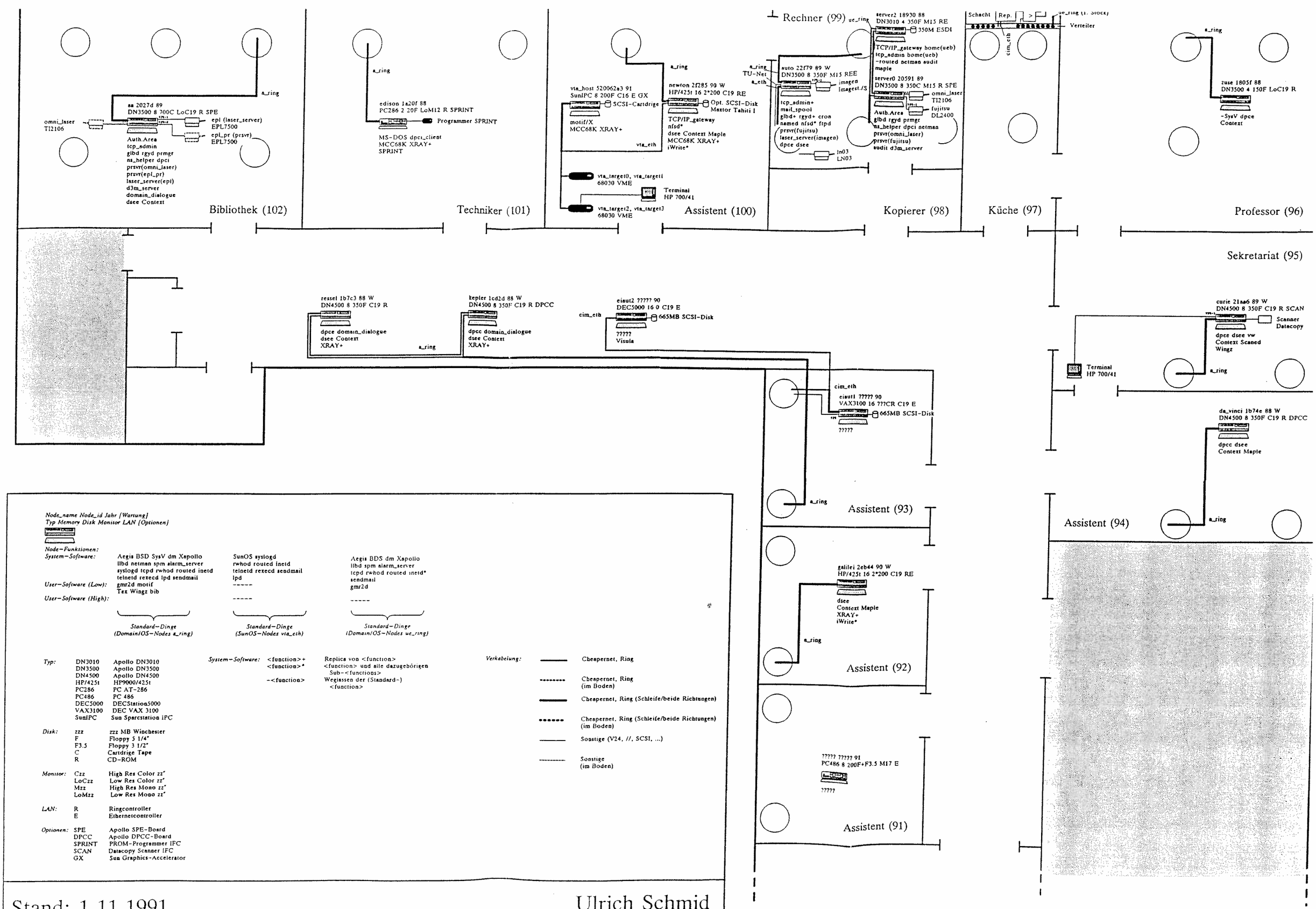
———	Cheapernet, Ring
--------	Cheapernet, Ring (im Boden)
▬▬▬	Cheapernet, Ring (Schleife/beide Richtungen)
▪▪▪▪▪▪	Cheapernet, Ring (Schleife/beide Richtungen) (im Boden)
———	Sonstige (V24, //, SCSI, ...)
--------	Sonstige (im Boden)

Stand: 1.11.1991

Ulrich Schmid

Layer 100: Beschriftung

Layer 119: a_eth(M)	Layer 120: a_ring(M)	Layer 121: vta_eth(M)	Layer 123: ue_ring(M)	Layer 124: cim_eth(M)	
Layer 129: a_eth	Layer 130: a_ring	Layer 131: vta_eth	Layer 133: ue_ring	Layer 134: cim_eth	



Node_name Node_id Jahr [Wartung]
Typ Memory Disk Monitor LAN [Optionen]

Node-Funktionen:

System-Software:	Aegis BSD SysV dm Xapollo libd netman spm alarm_server syslogd tcpd rwhod routed inetd telnetd rexecd lpd sendmail	SunOS syslogd rwhod routed inetd telnetd rexecd sendmail lpd -----	Aegis BDS dm Xapollo libd spm alarm_server tcpd rwhod routed inetd* sendmail gmr2d
User-Software (Low):	gmr2d motif Tex Wingz bib	-----	-----
User-Software (High):		-----	-----
	Standard-Dinge (Domain/OS-Nodes a_ring)	*Standard-Dinge (SunOS-Nodes vta_eth)*	*Standard-Dinge (Domain/OS-Nodes ue_ring)*

Typ:

DN3010	Apollo DN3010
DN3500	Apollo DN3500
DN4500	Apollo DN4500
HP/425t	HP9000/425t
PC286	PC AT-286
PC486	PC 486
DEC5000	DECStation5000
VAX3100	DEC VAX 3100
SunIPC	Sun Sparcstation iPC

Disk:

zzz	zzz MB Winchester
F	Floppy 5 1/4"
F3.5	Floppy 3 1/2"
C	Cartdrige Tape
R	CD-ROM

Monitor:

Czz	High Res Color zz"
LoCzz	Low Res Color zz"
Mzz	High Res Mono zz"
LoMzz	Low Res Mono zz"

LAN:

R	Ringcontroller
E	Ethernetcontroller

Optionen:

SPE	Apollo SPE-Board
DPCC	Apollo DPCC-Board
SPRINT	PROM-Programmer IFC
SCAN	Datacopy Scanner IFC
GX	Sun Graphics-Accelerator

System-Software:

<function>+	Replica von <function>
<function>*	<function> und alle dazugehörigen Sub-<functions>
-<function>	Weglassen der (Standard-) <function>

Verkabelung:

———	Cheapernet, Ring
--------	Cheapernet, Ring (im Boden)
═══	Cheapernet, Ring (Schleife/beide Richtungen)
••••••	Cheapernet, Ring (Schleife/beide Richtungen) (im Boden)
———	Sonstige (V24, //, SCSI, ...)
---------	Sonstige (im Boden)

Stand: 1.11.1991

Ulrich Schmid

Layer 100: Beschriftung
Layer 119: a_eth(M) Layer 120: a_ring(M) Layer 121: vta_eth(M) Layer 123: ue_ring(M) Layer 124: cim_eth(M)
Layer 140: Deckel Layer 150: Umrisse