

WISSENSCHAFTLICH-TECHNISCHE BERICHTE

FZR-274

Juli 1999

ISSN 1437-322X

Archiv-Ex.:

Jahresbericht 1997/98 und 1. Halbjahr 1999

Abteilung
Kommunikation und Datenverarbeitung

Editor: Ruprecht Fülle

Herausgeber:
FORSCHUNGSZENTRUM ROSSENDORF
Postfach 51 01 19
D-01314 Dresden
Telefon +49 351 26 00
Telefax +49 351 2 69 04 61
<http://www.fz-rossendorf.de/>

Als Manuskript gedruckt
Alle Rechte beim Herausgeber

FORSCHUNGSZENTRUM ROSSENDORF

WISSENSCHAFTLICH-TECHNISCHE BERICHTE



FZR-274

Juli 1999

**Jahresbericht 1997/98
und 1. Halbjahr 1999**

Abteilung
Kommunikation und Datenverarbeitung

Editor: Ruprecht Fülle

Inhaltsverzeichnis

1	Einordnung	4
2	Personelle Besetzung	5
3	Hauptergebnisse im Berichtszeitraum	6
4	Zentrale Server	8
4.1	Erneuerung und Ausbau der Zentralen Server	8
4.2	Hard- und Software-Ausstattung der Zentralen Server	9
4.3	Betrieb der Zentralen Server	13
4.4	Nutzung der Zentralen Server	14
5	Datennetze	18
5.1	Erweiterung des Verkabelungssystems	18
5.2	Erhöhung der Performance	18
5.3	Einführung von Breitband-Technologien	19
5.4	Übergang zum Virtuellen LAN (VLAN)	19
5.5	Zugriff zu externen Netzen	19
5.6	Telefoneinwahl in das Lokale Datennetz	20
5.7	Erreichter Stand des Datennetzes	20
6	Benutzerservice	24
6.1	Unterstützung dezentraler PCs	24
6.2	Numerik, Programmbibliotheken, Computeralgebra	26
6.3	Programmierung	27
6.4	Parallelisierung	27
6.5	Visualisierung	28
6.6	Textverarbeitung	29
6.7	Informationsdienste	30
6.8	Benutzerseminare	31
7	Wissenschaftliche Kooperation	32
8	Konferenzbeiträge und Vorträge	34
9	Ausführliche Berichte zu ausgewählten Themen	35

Vorwort

Galten unsere Arbeiten in den zurückliegenden Jahren dem Auf- und Ausbau der informationstechnischen Infrastruktur am Forschungsstandort Rossendorf (FSR)*, so lag in diesem Berichtszeitraum der Schwerpunkt unserer Tätigkeit bei der Innovation der Rechen- und Kommunikationstechnik sowie unserer Dienste für das Forschungszentrum.

Besonderes Gewicht hatte die Strukturierung und Unterstützung der PC-Welt im Forschungszentrum.

Die Generation der CONVEX-Rechner wurde im Bereich des Computing durch moderne Parallelrechner mit hoher Prozessorleistung abgelöst. Den Fileservice und interaktive Dienste übernahmen IBM-Server RS/6000 der obersten Leistungsklassen.

Auf dem Gebiet der Datennetze begann die Migration zu einer neuen Netzgeneration, u.a. durch den Übergang zum Fast Ethernet und den Aufbau von virtuellen Netzstrukturen. Gleichzeitig erhielt der Forschungsstandort eine leistungsstärkere Anbindung (2 MBit/s) an das Wissenschaftsnetz.

Gemeinsam mit der Kommission für Datenverarbeitung des FZR wurden Lösungen zur Vereinheitlichung und Strukturierung der PC-Technik im Forschungszentrum sowie zu deren effektiveren, weitgehend zentralen Administration erarbeitet und in Angriff genommen. Unser Ziel ist, hiermit die PC-Nutzer spürbar zu entlasten und gleichzeitig die PC-Welt zu ordnen und zu harmonisieren.

Der FSR verfügte am Ende des Berichtszeitraums über ca. 900 PCs, 160 Workstations und 95 XTerminals, die über ein strukturiertes Ethernet miteinander verbunden waren, Zugang zum Wissenschaftsnetz (WIN) und damit zu internationalen Netzen hatten. Im lokalen Netz des FSR werden gegenwärtig mehr als 1400 IP-Adressen verwaltet. Als Zentrale Server stehen 14 Rechner unterschiedlicher Architektur und Leistungsprofile zur Verfügung.

Die Berichte der Abteilung Kommunikation und Datenverarbeitung erschienen bisher alle zwei Jahre. Die Evaluierung des Forschungszentrums durch den Wissenschaftsrat, aber auch wichtige Arbeiten um den Jahreswechsel waren Anlaß, den Berichtszeitraum diesmal bis 30.06.1999 zu erweitern. Informationen über die Abteilung sind auch unter

<http://www.fz-rossendorf.de/FVTK/>

zu finden.

Unser Dank gilt den Zuwendungsgebern und dem Vorstand des FZR für die Förderung unserer Arbeiten. Der Leitung der Hauptabteilung Technische Infrastruktur danken wir für vielfältige Unterstützung.

Ruprecht Fülle

* Jahresbericht 1993/94. Hrg. R.Fülle, FZR 100, Juli 1995; Jahresbericht 1995/96. Hrg. R.Fülle, FZR-187, Juli 1997;

U.Harms,: The IT-development at the Research Centre Rossendorf near Dresden; PRIMEUR, March 1997; <http://www.hoise.com/primeur/>

1 Einordnung

Die Abteilung Kommunikation und Datenverarbeitung (FVTK) versteht sich als Koordinierungs- und Kompetenzzentrum für die Prozesse der Datenverarbeitung und Datenkommunikation am Forschungsstandort Rossendorf. Die Abteilung ist Teil der wissenschaftlich-technischen Infrastruktur. Sie ist der Hauptabteilung Technische Infrastruktur des FZR zugeordnet.

Entsprechend ihren Aufgaben gliedert sich die Abteilung FVTK in die Arbeitsgruppen

- Zentrale Server
- Datennetze
- Benutzerservice.

Zwischen diesen Gruppen bestehen sehr enge fachliche Verflechtungen.

Die Abteilung verfügt über 14 Planstellen, darunter 5 Wissenschaftlerstellen. Mehrere Stellen sind geteilt und mit Teilzeitkräften besetzt.

Die Abteilung ist für Weiterentwicklung und Koordinierung des Gesamtsystems der Datenkommunikation und Datenverarbeitung im FZR, für die bedarfsgerechte Gestaltung und das Management der institutsübergreifenden Datennetztechnik (Backbone-Netz, aktive Netzkomponenten und Zugang zu externen Netzen) zuständig. Die Abteilung betreibt gegenwärtig 14 Zentrale Server für ressourcenintensive Rechnungen im High-end-Bereich sowie für infrastrukturelle Dienste.

Gleichzeitig unterstützt die Abteilung die dezentrale Rechentechnik u. a. durch File-Service, wie Backup und Bereitstellung von Netzlaufwerken auf den Zentralen Servern für PCs.

Der Benutzerservice der Abteilung berät und unterstützt die Nutzer auf den Gebieten Numerik, Fortran- und C-Programmierung, Optimierung / Parallelisierung, Visualisierung, Informationssysteme (WWW), Textsatzsysteme sowie System- und Netzwerknutzung. Die Nutzer dezentraler Rechentechnik sind in diese Dienste einbezogen.

Zum Benutzerservice gehören außerdem die Koordinierung und Beschaffung von PC-Software. Seit 1997 ist das FZR in den Microsoft SELECT- Vertrag, den der Freistaat Sachsen abgeschlossen hat, involviert. Die Abteilung FVTK verwaltet Campuslizenzen der Firmen IBM , HP, SUN, DEC und Research Systems (IDL) für das FZR.

Die Abteilung FVTK wird künftig die zentrale Administration der PCs im gesamten FZR in Zusammenarbeit mit lokalen Administratoren übernehmen.

Die Abteilung arbeitet eng mit den Verantwortlichen für Datenverarbeitung der Organisationseinheiten am Forschungsstandort zusammen.

Seit 1997 ist eine Kommission für Datenverarbeitung (DV-Kommission) im Auftrage des Vorstandes und des Wissenschaftlich-Technischen Rates des FZR tätig. Ihr gehören Vertreter aller Institute und Zentralabteilungen des Forschungszentrums an. Sie vertritt die Interessen der Nutzer bei der Gestaltung der informationstechnischen Infrastruktur und der Dienste der Abteilung FVTK. Vorsitzender der Kommission ist Herr Prof. Dr. F.-P. Weiß, Direktor des Instituts für Sicherheitsforschung des FZR.

2 Personelle Besetzung

Abteilungsleitung

- Leiter der Abteilung: Doz. Dr. habil. Ruprecht Fülle
- Sekretariat: Siegrid Michel *

Gruppe Zentrale Server

- Leitung der Gruppe, Systemmanagement: Dipl.-Ing. Ursula Fromm
- Dateipflege, Backup, Archiv: Dipl.-Math. Helga Kirsch
- Technik der Zentralen Server: Dipl.-Ing. (FH) Klaus Kirsch
- Betriebsorganisation: Dipl.-Ing. (FH) Gerhard Nickel * *
- Operating der Zentralen Server: Andreas Geyer
Katrín Paul (bis 01.06.1998)
Kerstin Ries (seit 01.11.1998)

Gruppe Datennetze

- Leitung der Gruppe, Netzmanagement: Dr.-Ing. Wolf-Jochen Linnemann
- Netztechniker: Dipl.-Ing. (FH) Uwe Eisold
- Netztechniker: Ing. Rolf Rüdrieh *
- Netztechniker: Ing. Dieter Köhler *

Gruppe Benutzerservice

- Leitung der Gruppe, Paralleles Rechnen: Dr. Manfred Kunicke
- Numerik und Fortran: Dipl.-Math. Erhard Müller
- Visualisierung: Dipl.-Math. Horst Böttger *
Dipl.-Math. Nils Schmeißer *
- Textsysteme: Dr. Galina Vinel * *
- Arbeitsplatz-Software: Ing. Therese Jäger *
- PC-Technik: Uwe Hofmann

* Teilzeitstelle, * * geringfügige Beschäftigung

3 Hauptergebnisse im Berichtszeitraum

Die Abteilung Kommunikation und Datenverarbeitung nahm im Berichtszeitraum zwei weitere Parallelrechner der Fa. HP mit je 8 Prozessoren als Compute-Server in Betrieb. Mit der Installation von 3 IBM-Systemen, RS/6000 S70 und 2 * F50, wurde die Voraussetzung für Modernisierung und Ausbau des File-Services geschaffen. Der Server RS/6000 S70 steht auch für interaktive Dienste zur Verfügung. Die genannten neuen Server lösen die inzwischen moralisch verschlissenen CONVEX C3-Maschinen und das SPP1000 ab, die im Laufe des Jahres 1999 außer Betrieb genommen werden. Die Dienste wurden auf 14 Zentrale Server neu verteilt und die Software-Produkte auf den neuesten Stand gebracht.

An den Zentralen Servern waren im Juni 1999 mehr als 800 Nutzer eingetragen, von denen 520 ständig aktiv waren. 11 Abteilungen des FZR verwendeten Speicherplatz der Zentralen Server über die bereitgestellten File-Dienste als abteilungseigene Netzlaufwerke für ihre PCs. Standortweit greifen insgesamt 550 PCs auf Dienste der Zentralen Server (Nutzung Netzlaufwerke, Druckdienste u.a.) zu.

Die Graphikdienste haben hinsichtlich ihres wissenschaftlich-technischen Niveaus und ihres Nutzungsumfanges einen weiteren Aufschwung genommen: Die Technologie der Videoerstellung wurde weiterentwickelt und bei der Erzeugung von Videos zur Simulation von Prozessen eingesetzt. Neben der Bearbeitung und Herstellung von Druckvorlagen und Vortragsfolien am Farbdrucker Canon CLC700 (40.000 Farbdruke pro Jahr) hat sich auch der Druck von farbigen A0-Postern (über das Netz) zu einem zentrumsweit genutzten Dienst (ca. 350 Poster pro Jahr) entwickelt.

Neben dem systematischen Ausbau des lokalen Datennetzes auf gegenwärtig über 1400 aktive IP-Adressen mit standortweit mehr als 900 PCs, 160 Workstations und 95 XTerminals galten die Arbeiten der Einführung innovativer Netztechnologien, u.a. durch Anfertigung einer Studie über die Migration zu ATM. In stark belasteten Netzabschnitten fand der Übergang zum switched Ethernet (10 MBit/s und 100 MBit/s) statt. Zur Verbesserung der Transparenz bei der Netzverwaltung wurden VLAN-Strukturen eingeführt. Der Anschluß des FZR an das Wissenschaftsnetz (WIN) konnte im April 1998 von 128 KBit/s auf 2 MBit/s aufgerüstet werden. Der Standort nutzt nunmehr eine (2-MBit/s-) Teilkapazität des Breitband-WIN (B-WIN) am Gemeinschaftsanschluß in der TU Dresden.

Die informationstechnische Infrastruktur am Forschungsstandort hat sich im Berichtszeitraum intensiv weiterentwickelt. Die zunehmende Belastung der Nutzer durch die Administration der Rechentechnik am Arbeitsplatz führte zu der Forderung, diese künftig weitgehend zu zentralisieren und der Abteilung FVTK zu übertragen. Der Benutzerservice der Abteilung baute inzwischen eine Domain-Struktur mit Werkzeugen zur Administration von PCs unter MS-NT auf und installierte eine zentrale Ressource-Domain.

Gemeinsam mit der Kommission für Datenverarbeitung wurden Maßnahmen zur Vereinheitlichung und Harmonisierung vor allem der PC-Welt am FZR eingeleitet, z.B. durch die Definition von Standard-PCs und die Organisation eines zentralen Einkaufs der PCs nach den festgelegten FZR-Standards.

Der Benutzerservice beschaffte für die dezentrale Rechentechnik am Forschungsstandort im Jahr 1997 insgesamt 553 Software-Produkte im Wert von insgesamt 730 TDM. 1998 waren es 648 Software-Produkte im Gesamtwert von 740 TDM.

Die Abteilung betreibt den WWW-Server des Forschungszentrums. Die Gestaltung der WWW-Dokumente obliegt einer Redaktionsgruppe, der Vertreter der Institute und Zentralabteilungen des FZR angehören.

Mitarbeiter der Abteilung waren in Projekte von Instituten einbezogen oder lösten spezielle Programmieraufgaben. Sie unterstützten Wissenschaftler bei der Installation und beim Einfahren von Programmen, z. B. für das Beschleunigerprojekt ELBE.

Die Abteilung beteiligte sich aktiv an der Ausbildung von Studenten und Doktoranden sowie von Umschülern und Berufsschülern (Laboranten und Bürokaufleute).

4 Zentrale Server

4.1 Erneuerung und Ausbau der Zentralen Server

Im Berichtszeitraum fand eine grundlegende Erneuerung der zentralen Rechentechnik statt. Sie wurde vor allem durch den hohen moralischen Verschleiß der CONVEX-Server C3220, C3820 und SPP1000 notwendig. Diese Anlagen haben inzwischen ein Alter von 5 bis 8 Jahren erreicht, und die Firma CONVEX hat nach ihrem Kauf durch Hewlett Packard Hard- und Software dieser Systeme nicht mehr weiterentwickelt.

Als Ersatz für die CONVEX-Server im Compute-Bereich installierte die Abteilung zunächst zwei Parallelrechner der Fa. HP: 1997 ein S-Class-System SPP2000 und 1998 einen V-Class-Rechner V2250 mit jeweils 8 Prozessoren PA 8000 bzw. PA 8200.

	Zubau	Anzahl Prozessoren	Kapazitäts- Zuwachs/MFlops	Gesamt- Kapazität/MFlops
1993	CONVEX 3820	2	80	80
1995	CONVEX SPP1000	4	200	280
1996	IBM RS/6000-591	1	40	320
1997	HP SPP2000	8	1248	1568
1998	HP V2250	8	1624	3192

Tabelle 1: Kapazitätsentwicklung an den Zentralen Compute-Servern

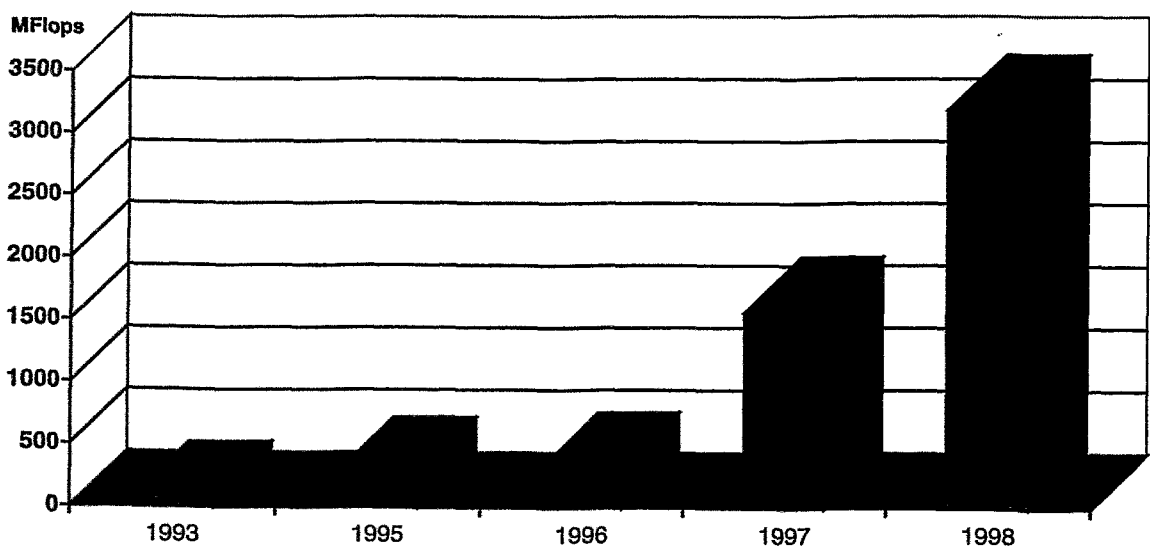


Abbildung 1: Installierte Compute-Kapazität an den Zentralen Servern

1998 wurden drei neue Systeme IBM RS/6000 (2 Server F50 mit je 2 Prozessoren und ein Enterprise-Server S70 mit 4 Prozessoren) als File-Server bzw. als Server für interaktive Dienste installiert. Ausschlaggebend für die Wahl dieser Rechner waren die Stabilität der Entwicklungslinie RS/6000, die Leistungsfähigkeit des Backup- und Archivierungssystems ADSM, das breite Spektrum an Third-Party-Software und unsere guten Erfahrungen mit den Modellen 550 und 591. Die neuen Systeme nahmen Anfang 1999 den Betrieb auf. Auf ihnen sind neueste Software-Versionen installiert.

Ende März 1999 wurde die CONVEX C3220 außer Betrieb genommen. Die Stilllegung der CONVEX C3820 und des SPP1000 wird im September 1999 folgen.

Vorbereitet wird die Anschaffung eines leistungsfähigen Compute-Servers. Er soll die Performance der Institutsserver, z.B. Workstations mit vier Prozessoren, erheblich übersteigen.

Mit der Erweiterung infrastruktureller Dienste (Oracle 8, PC-Administration) kamen auch leistungsfähige PCs mit Intel-Architektur als Zentrale Server hinzu.

Zu den Zentralen Servern zählt neben einem A3-Color-Drucker Canon CLC 700 seit 1997 auch ein Drucker Design Jet 2500 CP der Fa. HP. Er dient der Herstellung von A0-Farbdrukken, vor allem von Postern.

4.2 Hard- und Software-Ausstattung der Zentralen Server

(Stand: Juni 1999)

CONVEX C3820 (rhea)

Anzahl Prozessoren:	2
Hauptspeicher:	512 MByte
Plattenspeicher:	17 GByte + 63 GByte (SCSI)
Peripherie:	ACL6/176 (6 DLT), DAT
Betriebssystem:	ConvexOS V11.5.1
Archivierungssysteme:	UniTree2.0, CVDM 1.1
Compiler:	C V5.0, Fortran77 V9.1
Grafik:	AVS 5.0
Einsatz:	Archiv-, File- und Compute-Server

HP SPP2000 (juno)

Anzahl Prozessoren:	8 (PA-RISC 8000)
Hauptspeicher:	2 GByte
Plattenspeicher:	40 GByte
Betriebssystem:	SPP-UX V5.2.1
Compiler:	C,C++ V11.0.1, VAST f90/V3.4.3, f77 (Exemplar Fortran V.1.1)
Bibliotheken:	MLIB V4.1
weitere Software:	CXpa 4.2, CXdb V4.1, HP-MPI V1.2, HP-PVM V3.3.11, NQS+ V2.3
Einsatz:	Compute-Server

HP V2250 (pales)

Anzahl Prozessoren: 8 (PA-RISC 8200)
Hauptspeicher: 2 GByte
Plattenspeicher: 36 GByte
Betriebssystem: HP-UX/V11.00
Compiler: HP C ANSI C, C++ V11.0.1, HP Fortran f90, java
Bibliotheken: MLIB V4.1
weitere Software: HP DDE, xdb (Debugger), HP-MPI V1.3, LSF Batch
Einsatz: Compute-Server

CONVEX SPP1000/CD-4N (vesta)

Anzahl Prozessoren: 4
Hauptspeicher: 512 MByte
Plattenspeicher: 8 GByte
Betriebssystem: SPP-UX V3.2
Compiler: C V6.2.1.0, Fortran V9.3
Bibliotheken: CERNLIB, LAPACK, NAGLIB, SCILIB, VECLIB, VECLIB8
weitere Software: Convex-PVM, APC, CXdb, CXpa
Einsatz: Compute-Server

IBM RISC RS/6000 S70 (hera)

Anzahl Prozessoren: 4 (PowerPC RS 64)
Hauptspeicher: 2 GByte
Peripherie: SCSI- und SSA-Plattensystem, Bandlaufwerke 8 mm und 4 mm
Betriebssystem: AIX V4.3.2
Compiler: Fortran V5.1, NAG77, g77 V0.5.23, CSet++ V3.1.2, gcc V2.8.1
Bibliotheken: CERNLIB, NAGLIB, Numerical Recipes für C und Fortran, Textware System für Fortran77 Library Mark 18
weitere Software: Maple V, L^AT_EX GNU-Software
Einsatz: Zentraler Login-Server, interaktive Dienste, News-, File- und Pop-Server

IBM RISC RS/6000 F50 (nike)

Anzahl Prozessoren: 2 (604e3)
Hauptspeicher: 512 MByte
Peripherie: SCSI- und SSA-Plattensystem, Bandlaufwerk 8 mm
Betriebssystem: AIX V4.3.2
Einsatz: WWW-, FTP- und File-Server

IBM RISC RS/6000 F50 (selene)

Anzahl Prozessoren: 2 (604e3)
Hauptspeicher: 512 MByte
Peripherie: SCSI- und SSA-Plattensystem, Bandlaufwerk 8 mm
Betriebssystem: AIX V4.3.2
Backup-System: ADSM
Einsatz: Backup- und File-Server

IBM RISC RS/6000 Mod. 591 (rike)

Anzahl Prozessoren: 1
Hauptspeicher: 1024 MByte
Plattenspeicher: 24 GByte/SCSI, 18 GByte/SSA
Peripherie: Bandlaufwerk 8 mm, CD-Tower (6 Laufwerke)
Betriebssystem: AIX 4.2.1
Compiler: Fortran V3.2, CSet++ V3.1.2
Bibliotheken: OSL, NAGLIB, CERNLIB, FZRLIB
Computeralgebra: MapleV Rel.4, Mathematika V3.0
DBMS: ORACLE 7.3.2
Einsatz: Datenbank- und Compute-Server

IBM RISC RS/6000 Mod. 550 (rita)

Anzahl Prozessoren: 1
Hauptspeicher: 128 MByte
Plattenspeicher: 12 GByte
Peripherie: Laufwerke für CD-ROM, Magnetband 8mm und 1/4 Inch
Betriebssystem: AIX 3.2.5
Compiler: Fortran V3.1, CSet++ V3.1.1
Bibliotheken: CERNLIB, FZRLIB, ESSL, OSL, NAGLIB
Computeralgebra: AXIOM V2.1
Textsystem: \LaTeX
Einsatz: File-Server, interaktive Dienste

IBM RISC RS/6000 Mod. 43P 140 (ns1)

Anzahl Prozessoren: 1
Hauptspeicher: 128 MByte
Plattenspeicher: 12 GByte
Peripherie: CD-ROM, Exabyte, 8mm-MB
Betriebssystem: AIX 4.2.1
Einsatz: Mail- und Name-Server (DNS)

HP9000/735 (apollo)

Taktfrequenz: 99 MHz
Hauptspeicher: 208 MByte
Plattenspeicher: 7 GByte
Betriebssystem: HP-UX 9.05
Compiler: Fortran 9000 V9.16, C Ansi 9.61
Grafik: AVS 5.01
Bibliotheken: BLAS Library, ConvexMLIB
Einsatz: Grafik- und File-Server

HP9000/735 (minos)

Taktfrequenz: 99 MHz
Hauptspeicher: 208 MByte
Plattenspeicher: 7 GByte
Betriebssystem: HP-UX 9.05
Compiler: Fortran 9000/V9.16, C Ansi 9.61
Grafik: AVS 5.01
Bibliotheken: BLAS Library, ConvexMLIB
Einsatz: Grafik-Server

HP9000/715 (ikon)

Taktfrequenz: 75 MHz
Hauptspeicher: 64 MByte
Plattenspeicher: 7 GByte
Videosystem: Parallax PowerVideo700 Plus
Betriebssystem: HP-UX 9.05
Compiler: Fortran 9000 V9.16, C Ansi 9.61
Grafik: AVS 5.01, Parallax Video Development Environment
Bibliotheken: BLAS Library, ConvexMLIB
Einsatz: Grafik-Server, CD-Brennen, Video-Aufzeichnungen

HP9000/715 (hermes)

Taktfrequenz: 75 MHz
Hauptspeicher: 32 MByte
Plattenspeicher: 6 GByte
Betriebssystem: HP-UX 10.20
Einsatz: HP-UX SW-Depot-Server

Pentium II (atlas)

Anzahl Prozessoren: 2 (Pentium II)
Taktfrequenz: 400 MHz
Hauptspeicher: 1 GByte
Plattenspeicher: 27 GByte, Hot Swap Disks, Raid 5
Betriebssystem: Red Hat Linux 5.2, Kernel 2.2.9
Einsatz: Datenbank-Server mit DBMS Oracle 8

HP LH 4 Pentium II (NTS4)

Anzahl Prozessoren: 1 (Pentium II Xeon)
Taktfrequenz: 400 MHz
Hauptspeicher: 256 MByte
Plattenspeicher: 36 GByte, Hot Swap Disks, Raid 5
Betriebssystem: MS Windows NT 4.0
Einsatz: SMS-Datenbankserver mit SQL 6.5

Darüber hinaus betreibt die Abteilung FVTK weitere PCs für infrastrukturelle Dienste.

4.3 Betrieb der Zentralen Server

Die Zentralen Server werden durchgehend betrieben. Operatoren sind werktags von 6.00 bis 16.30 Uhr anwesend. Außerhalb dieser Zeit kontrollieren die Systemverantwortlichen den Betrieb der Anlagen über Telefoneinwahl in das lokale Datennetz des FSR.

Abschaltungen wurden kurzzeitig und nur bei Erweiterungen oder Umbauten der Hardware vorgenommen. Störungen des Elektronetzes blieben, soweit sie innerhalb der Autonomiezeit der Unterbrechungsfreien Stromversorgung (30 min) lagen, ohne Auswirkungen auf den Betrieb der Server.

4.4 Nutzung der Zentralen Server

Genutzte Rechenleistungen

Rechenleistungen an den Zentralen Compute-Servern wurden wie bisher vor allem von den physikalischen und physikalisch-technischen Arbeitsrichtungen des Forschungsstandortes Rossendorf in Anspruch genommen (Tabelle 2). Da gegenwärtig nur wenige Anwendungen parallelisiert sind, werden die Pararallelrechner mittels Subcomplex-Verwalter bzw. Process Resource Manager vorwiegend als n Single-Prozessor-Maschinen genutzt.

		FWI ¹	FWK ²	FWS ³	Sonstige
1997	SPP 2000 (juno)	90 %	2 %	6 %	2 %
	SPP 1000 (vesta)	21 %	18 %	59 %	2 %
	RS/6000 Mod. 591 (rike)	74 %	6 %	7 %	13 %
1998	V 2250 (pales)	81 %	17 %	< 1 %	2 %
	SPP 2000 (juno)	60 %	17 %	19 %	4 %
	SPP 1000 (vesta)	< 1 %	2 %	98 %	< 1 %
	RS/6000 Mod. 591 (rike)	25 %	5 %	3 %	67 %
1999 (bis 30.6.)	V 2250 (pales)	48 %	21 %	28 %	3 %
	SPP 2000 (juno)	49 %	7 %	40 %	4 %
	SPP 1000 (vesta)	< 1 %	< 1 %	100 %	< 1 %
	RS/6000 Mod. 591 (rike)	< 1 %	9 %	< 1 %	91 %

Tabelle 2: Anteil an der genutzten Prozessorkapazität Zentraler Compute-Server

Unter "Sonstige" sind auch Rechnungen für das ELBE-Projekt berücksichtigt.

¹Institut für Ionenstrahlphysik und Materialforschung

²Institut für Kern- und Hadronenphysik

³Institut für Sicherheitsforschung

Der Hauptanteil der Prozessorzeit an den Zentralen Servern wurde für folgende Einsatzgebiete genutzt:

- **Materialforschung**
 - Ab-initio-Rechnungen zu Struktur und Eigenschaften von Bornitrit
 - Atomistische Simulation der Ionenimplantation und ihrer Anwendung in der Silizium-Technologie
 - Monte-Carlo-Simulation der Evolution von Nano-Clustern
 - Computersimulation von Implantations- und Depositionsprozessen
 - * Computersimulation der Abscheidung diamantartiger Kohlenstoffschichten
 - Computersimulation von Prozessen bei der Ionenbestrahlung (“Blocking to Channeling“, “Random to Channeling“)
- **Kern- und Hadronenphysik**
 - Diagonalisierung großer Matrizen zur Eigenwertberechnung für das Schalenmodell von Atomkernen in der Nähe von $N=50$ (Matrizen bis zum Rang 19.000!)
 - Lösung der Transportgleichung für relativistische Schwerionen-Stöße
 - Berechnung von Dileptone-Signalen bei ultrarelativistischen Schwerionen-Stößen
- **Sicherheitsforschung**
 - Berechnung von Neutronenfeldern in Druckwasserreaktoren WWER 1000
 - Deterministische Rechnungen zum Neutronentransport in Reaktoren
 - Abschirmungsrechnungen für Forschungsreaktoren
 - Druckschlag-Simulation in Rohrleitungssystemen unter Berücksichtigung der Fluid-Struktur-Wechselwirkung
 - Rechnungen zur Schmelze-Struktur-Wechselwirkung mittels ANSYS
 - Berechnung der Kritikalität u. a. bei der Umlagerung von Spaltmaterial
 - Transportrechnungen für Gamma- und Neutronenfelder zur Dosisbestimmung
 - Rechnungen zum Ionentransport
 - Monte-Carlo-Rechnungen unter Anwendung von PVM zur Modellierung einer Plasma-Neutronenquelle
- **Strahlenschutz**
 - Ausbreitungs- und Expositionsrechnungen für die Strahlenschutz-Umgebungsüberwachung
- **Beschleunigertechnik**
 - Rechnungen zum Strahlführungssystem von ELBE
 - Tracking-Rechnungen für das ELBE-Projekt
 - Rechnungen zur Abschirmung am Projekt ELBE

Infrastrukturelle Dienste

Neben der Nutzung als Compute-Server erfüllen die Zentralen Server infrastrukturelle Aufgaben:

- als File-Archiv (C3820 mit einem Bandroboter ALT Odetics), wobei dessen Volumen von 700 GBytes zu 30 % durch die Nutzer belegt worden ist,
- als File-Server (C3220, ab 1999 F50) für das File-Backup, u. a. für dezentrale Workstations;
Ende 1998 waren 23 Workstations in diesen Dienst, der sowohl die zyklische Sicherung aller als auch das tägliche Backup veränderter Dateien umfaßt, einbezogen. Das bisher weitgehend manuell organisierte Backup wird ab 1999 automatisiert und mittels ADSM auf der IBM F50 (selene) durchgeführt.
- Bereitstellung von Netzlaufwerken ("virtuellen Laufwerken"): Auf diese Netzlaufwerke greifen ca. 500 PCs zu. Für 11 Abteilungen sind "Abteilungs"-Netzlaufwerke eingerichtet.
- als Datenbankserver (RS/6000 Mod. 591, Datenbank Oracle 7 mit WWW-Server) für Intranet-Informationsdienste, z. Zt. für das Projekt ROBIS (Rossendorfer Bibliotheks-Informationssystem),
zusätzlich seit 1999 Oracle 8 auf Linux-Server ("atlas")
- als Distribution-Server (HP9000/735, "hermes") für HP-Software
- als Fontserver für XTerminals (RS/6000 Mod. 591)
- als Mail-Server (RS/6000 43P140, "ns1")
- als Pop-Server (S70, "hera")
- als WWW-Server (HP9000, "apoll", ab 1999 F50, "nike")
- NetNews-Server (C3220, "hera", ab 1999 S70, "hera")
- Ftp-Server (C3220, "hera", ab 1999 F50, "nike") und
- als Name-Server (C3220, "hera", ab 1999 RS/6000 43P140, "ns1") der Internet-Domain fz-rossendorf.de.

Spezielle Dienste

Neben den genannten Leistungen hat die Abteilung ständig das Spektrum ihrer Dienste erweitert:

- Druck- und Scandienste
 - Farbdruck A3 und A0 (Poster)
 - Doppelseitiger Farbdruck (A4)
 - Foliendruck (A4)
 - Scannen von Papiervorlagen und Dias

Standortweit können Dateien über das Netz zum Drucker übertragen werden. Der Operator kontrolliert die Graphikdateien und nimmt nach Rücksprache mit dem Nutzer gegebenenfalls Korrekturen vor.

Das Drucken von graphischen Darstellungen auf Papier und Folie hat sich zu einem intensiv genutzten Dienst entwickelt:

1998 wurden insgesamt 41.000 Seiten (A4) und 350 Poster (A0) gedruckt.

- Zur Erzeugung von Video-Sequenzen auf professionellem Niveau stehen seit 1995 eine entsprechende Geräteketten und leistungsfähige Tools zur Verfügung (vgl. Abschnitte 6.4 und 9).
- Ein Windows-NT-Server (Pentium II, "NTS00") wurde eingerichtet, auf dem die Tektronix-Software WinDD verfügbar ist. Er gestattet, Microsoft-PC-Applikationen von Tektronix-XTerminals aus zu nutzen.
- Ein weiterer PC unter Windows NT dient als zweiter CD-ROM-Server. Er ist mit zwei CD-ROM-Tower ausgerüstet und kann z. Zt. im Netz insgesamt 14 CD-ROMs bereitstellen.
- Das Brennen von CD-ROMs wird seit 1996 angeboten. Der CD-ROM-Brenner läßt sich je nach gewünschtem Dateiformat an UNIX- oder MS-DOS-Rechner anschließen. Hauptnutzer sind gegenwärtig die Zentralabteilung Forschungs- und Informationstechnik zur Archivierung von CAD-Unterlagen und die Zentralbibliothek zum Sichern des FZR-Fotoarchivs.
- Ferner betreut die Gruppe Zentrale Server die Telefondatenerfassung an der TK-Anlage des Standorts.

5 Datennetze

5.1 Erweiterung des Verkabelungssystems

Das lokale Netz des Forschungsstandortes wurde im Berichtszeitraum weiter ausgebaut.

Auf Anforderung der Nutzer wurden vorhandene Inhouse-Verkabelungen ergänzt bzw. modifiziert. Geringfügige Erweiterungen übernahmen dabei Mitarbeiter der Gruppe Datennetze, während umfangreichere Arbeiten, wie der Aufbau ganzer Netzsegmente, Fremdfirmen übertragen wurden.

Mit Sanierung und Neubau erhielten folgende Gebäude neue Verkabelungssysteme (ausschließlich S/STP-Kabel, Kategorie 5):

- Gebäude 01 (Wachgebäude, Eingang)
- Gebäude 20.2 (Alarmzentrale)
- Gebäude 74 (Gebäude FVTD)
- Gebäude 86.1 (VKTA)
- Gebäude 97a (Tandemgebäude, Anbau)
- Gebäude 101.2 ("Schmiede", MHD-Anbau)
- Gebäude 120 (Gebäude Kernphysik).

Lichtwellenleiterkabel (62,5 μ Gradienten-Faser und 9 μ Monomode-Faser) wurden zwischen folgenden Gebäuden neu verlegt:

- | | | |
|------------------------------------|-----|----------------------------------|
| • Gebäude 01 (Wachgebäude) | und | Gebäude 4a (Laborgebäude) |
| • Gebäude 20.2 (Alarmzentrale) | und | Gebäude 8a (Laborgebäude) |
| • Gebäude 92 (PET-Labor) | und | Gebäude 07 (Zyklotrongebäude) |
| • Gebäude 07 (Zyklotron) | und | Gebäude 97b (Tandemgebäude) |
| • Gebäude 97b (Tandemgebäude) | und | Gebäude 13c (Rechenzentrum) |
| • Gebäude 13c (Rechenzentrum) | und | Gebäude 120 (Gebäude Kernphysik) |
| • Gebäude 120 (Gebäude Kernphysik) | und | Gebäude 40 (Gebäude ELBE) |
| • Gebäude 13 (Vorstandsgebäude) | und | Gebäude 4a (Flachbau) |
| • Gebäude 4a (Flachbau) | und | Gebäude 74 (Gebäude FVTD) |
| • Gebäude 74 (Gebäude FVTD) | und | Gebäude 101.2 (MHD-Anbau) |
| • Gebäude 94 (Gebäude FWR) | und | Gebäude 96 (VKTA) |
| • Gebäude 94 (Gebäude FWR) | und | Gebäude 86.1 (VKTA). |

Für alle diese Projekte erarbeitete die Abteilung FVTK die Aufgabenstellung und legte die technologische Konzeption fest.

5.2 Erhöhung der Performance

Die Erhöhung der Performance des Lokalen Netzes am Forschungsstandort vollzog sich in drei Etappen:

- Trennung der Last durch Einbau von Brücken in die damaligen Subnetze 4, 8 und 12 im Ergebnis von Lastanalysen (1996/97)
- Migration vom shared zum switched (10-MBit-)Ethernet in Abschnitten mit hohem Datenverkehr (seit 1997)

- Einführung des Fast Ethernet (seit 1998) parallel zum 10-MBit/s-Ethernet.

Mit dieser Entwicklung ging (seit 1998/99) die Einführung von VLAN-Strukturen (virtuellen Netzstrukturen) einher.

5.3 Einführung von Breitband-Technologien

Sowohl der wachsende Datenverkehr im LAN als auch die Ankündigung neuartiger Applikationen waren Anlaß zur Einarbeitung in Breitband-Technologien und in die Möglichkeiten der Migration zu diesen.

Dem damaligen Trend folgend, war zunächst der Einsatz von ATM (Asynchronous Transfer Mode) im Backbone-Bereich und in bilderzeugenden Abschnitten des Forschungsstandorts in Betracht gezogen worden. Im Rahmen einer Diplomarbeit wurde die Verwendung von ATM für die Übertragung von Bilddaten aus PET- und Detektorlabor als Modellfall untersucht.

Der Lösungsvorschlag sah den Aufbau einer ATM-Substruktur mit ATM-Eingangsswitch zum WIN, einem ATM-Backbone-Bereich (155 MBit/s) mit einem weiteren ATM-Switch im PET-Labor und ATM-Anbindungen (155 MBit/s) an die bildgebenden und -verarbeitenden Endstellen vor. Die übrigen Bereiche sollten über Workgroup-Switches mit dem zentralen ATM-Eingangsswitch verbunden werden.

Da die Übertragung von Video- und Audiodaten noch nicht aktuell und wegen des relativ langsamen B-WIN-Anschlusses nicht attraktiv war, wurde das ATM-Projekt zunächst zurückgestellt und der schrittweise Ersatz des 10-MBit-Ethernet durch Fast Ethernet (100 MBit/s) begonnen.

In Gebäuden mit entsprechender Inhouse-Verkabelung werden 10- und 100-MBit-Ethernet nebeneinander betrieben, so in den Gebäuden 13c, 97a, 101.2(MHD) und 120.

Als aktive Komponenten kommen anstelle der MMAC von Cabletron nunmehr SmartSwitches 6000 und 2000 desselben Herstellers zum Einsatz. Für diese stehen auch entsprechende Module im Managementsystem SPECTRUM zur Verfügung.

5.4 Übergang zum Virtuellen LAN (VLAN)

Um die Domains am Standort unabhängig von der physischen Struktur des Netzwerkes verwalten zu können, wurden VLAN-Strukturen gemäß IEEE 802.1d und IEEE 802.1q eingerichtet. Sie basieren z.Z. auf dem SecureFast-VLAN-Manager Vers.1.7, Rev.1 der Fa. Cabletron. Dieser ist inzwischen in das Netzwerk-Management-System (Cabletron) integriert.

In der Migrationphase müssen die VLAN-Abschnitte und geroutete Subnetze parallel betrieben werden und miteinander kommunizieren. Die Umstellung auf VLAN verläuft schrittweise und bisher ohne größere Probleme.

5.5 Zugriff zu externen Netzen

Im Jahr 1997 war die Last am WIN-Zugang des Forschungsstandorts so weit gestiegen, daß der geringe Durchsatz von 128 KBit/s den Datenverkehr mit externen Partnern und den

Zugriff zum Internet spürbar behinderte, vor allem während der regulären Arbeitszeit.

Im April 1998 konnte die Transferleistung des WIN-Anschlusses von 128 KBit/s auf 2 MBit/s erhöht werden. Der Forschungsstandort nutzt nunmehr eine 2-MBit/s-Teilkapazität des 34-MBit/s-Anschlusses an der TU Dresden. Mit diesem Anschluß an das Breitband-WIN (B-WIN) steht als WIN-Protokoll nicht mehr X.25, sondern IP (over ATM) zur Verfügung. Damit entfällt auch der direkte Zugriff auf Datex-P. Die betroffenen Dienste, wie Electronic Banking, wurden rechtzeitig auf ISDN umgestellt.

5.6 Telefoneinwahl in das Lokale Datennetz

Bis 1998 erfolgte die Einwahl in das lokale Netz des Forschungsstandorts über einen Terminalserver und damit auf dem Nutzungsniveau von VT-100-Terminals. Seit 1998 steht ein leistungsfähigerer Zugang über den Access-Server MAX 1800 (Fa. Ascend) mit 8 ISDN-Kanälen (so) zur TK-Anlage des Standorts (HICOM 390, Fa. Siemens) zur Verfügung. Er ermöglicht die Verbindung über das Protokoll ppp auf dem Niveau eines Hosts.

5.7 Erreichter Stand des Datennetzes

Das lokale Netz (LAN) des Forschungsstandortes verfügte am Ende dieses Berichtszeitraums über mehr als 1400 verwaltete Netzadressen, u.a. von ca. 900 PCs, 160 Workstations und 95 XTerminals. Das Netz verbindet flächendeckend die Gebäude des Kerngeländes am Forschungsstandort.

Das LAN stellt ein strukturiertes Ethernet dar. Es ist mittels zweier Router (CISCO 4500 und AGS 3+) in Subnetze untergliedert (Abbildung 2).

Das Backbone-Netz hat die Struktur eines Sternes. Die Gebäude sind über Lichtwellenleiter-Kabel (LWL) (62,5 μ Multimode, 12 bis 24 Gradientenfasern, bzw. 9 μ Monomode, 24 Fasern) verbunden.

Der technologischen Entwicklung entsprechend, bestehen am Standort unterschiedliche In-house-Verkabelungssysteme:

- Thinwire-Systeme:
Gebäude 4, 4a, 4b, 6, 7, 8a, 9, 13;
Installierung: 1992/93.
- Twisted-Pair-Systeme:
 - IBM-Typ-1-Kabel: Gebäude 94, z. T. Gebäude 13;
Installierung: 1991/92;
 - S/STP-Kabel, Kategorie 5:
Gebäude 8b, 8d, 13c, 74, 86.1, 87, 90, 91, 92, 93, 97a, 101;
Verlegung ab 1993 als verbindliche Norm des FSR.

Als aktive Komponenten werden überwiegend, seit 1992 durchgängig Cabletron- und CISCO-Produkte eingesetzt. Lediglich bei einigen älteren Verkabelungen ist noch Hirschmann-Technik im Einsatz.

Besondere Lösungen bestehen im Subnetz 4 (Rechenzentrum): Die Zentralen Server sind über einen Cabletron-Switch, z. T. auch über FDDI, untereinander und mit dem übrigen LAN verbunden.

Inzwischen wird in kommunikationsintensiven Abschnitten neben dem 10-MBit/s- Ethernet auch Fast Ethernet betrieben, soweit es die Inhouse-Verkabelung gestattet. Als aktive Komponenten kommen SmartSwitches 6000 und 2000 der Fa. Cabletron zum Einsatz. Sie unterstützen nicht nur das 100-MBit/s-Ethernet, sondern auch switched und shared Ethernet mit 10 MBit/s.

In folgenden Gebäuden werden gegenwärtig: 100-MBit/s- und 10MBit/s-Ethernet parallel betrieben:

- Gebäude 13c (Rechenzentrum)
- Gebäude 97a (Tandemgeb.-Anbau)
- Gebäude 101.2 (MHD-Anbau) und
- Gebäude 120 (Kernphysik).

Über das Ethernet werden hauptsächlich das Protokoll TCP/IP, und in geringem Umfang DECnet und IXP betrieben. Hinzu kommen die Protokolle NetBIOS und NetBEUI für die Einbindung von PCs. Mit NetBT (NetBIOS over TCP/IP) wird die Verbindung zur UNIX-Welt auch über Subnetzgrenzen hergestellt. Netzwerkweite Verzeichnis-, Druck- und Dateidienste werden mit NFS und dem SMB-Protokoll (Server Message Block, SAMBA) realisiert.

Abbildung 3 zeigt den Stand des Einsatzes von ISDN Ende 1998 im LAN des Forschungsstandortes. Es diente als Übergangslösung zur Anbindung peripherer Gebäude am Standort. Mit dem Ausbau des LWL-Backbones im Jahre 1998 wurde die Anwendung von ISDN im LAN wegen seiner geringen Übertragungsgeschwindigkeit stark reduziert: Lediglich zum Labor im Felsenkeller in Dresden-Plauen besteht noch eine ISDN-Verbindung. Das FunkLan fand in besonderen Situationen, z.B. bei Umbauten am Netz, Anwendung.

Der Anschluß des LAN zum Wissenschaftsnetz (Protokoll: IP over ATM) hat gegenwärtig eine Übertragungsrate von 2 MBit/s.

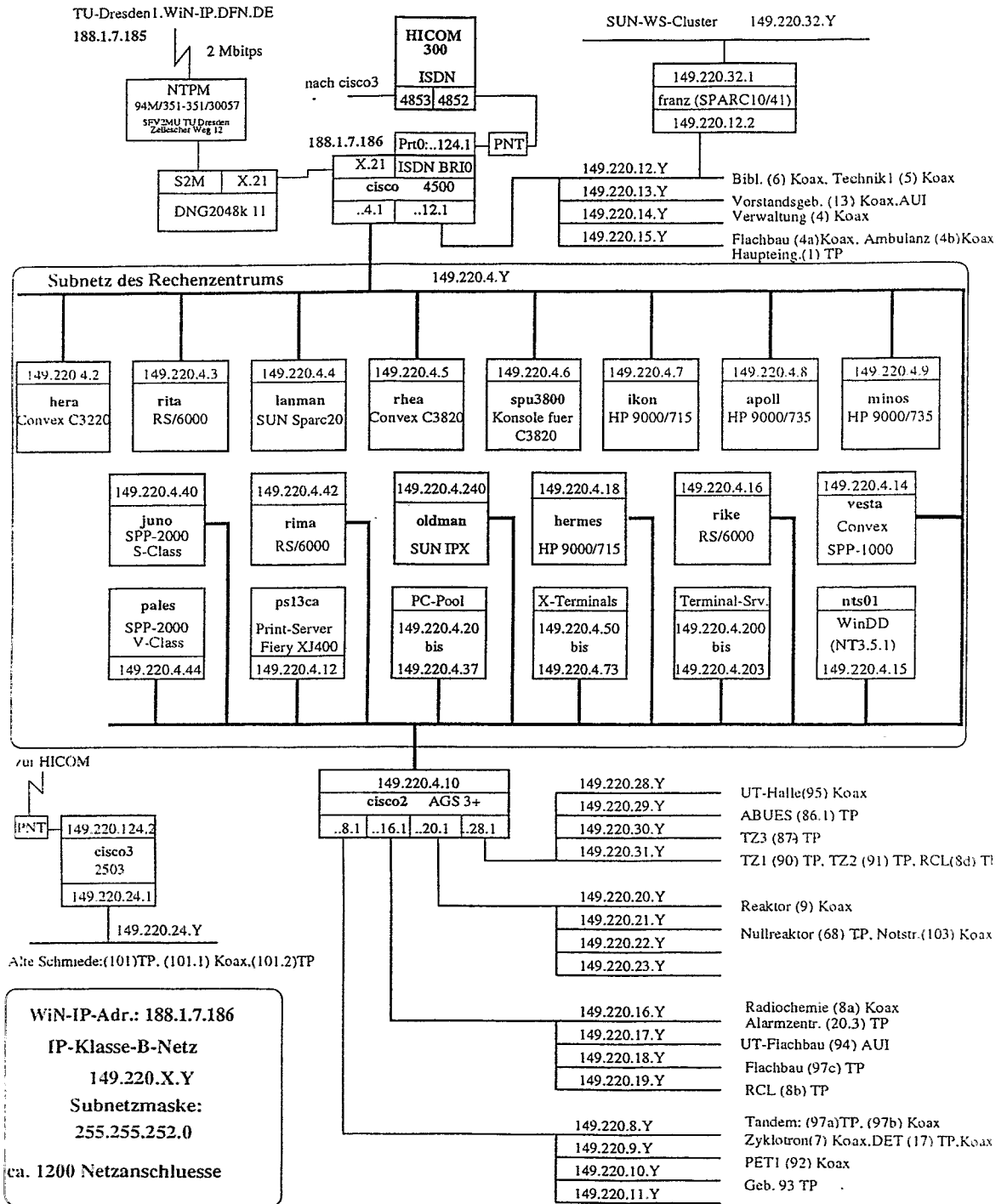
Das gesamte LAN wird mit Hilfe des Systems SPECTRUM (Fa. Cabletron) überwacht und administriert.

Am Forschungsstandort sind unter der IP-Domain-Adresse 149.220. . inzwischen drei IP-Domains installiert:

- fz-rossendorf.de (Forschungszentrum Rossendorf e.V.)
- vkta-rossendorf.de (Verein für Kernverfahrenstechnik und Analytik e.V.)
- nuklearmedizin.de (Deutsche Gesellschaft für Nuklearmedizin e.V.)

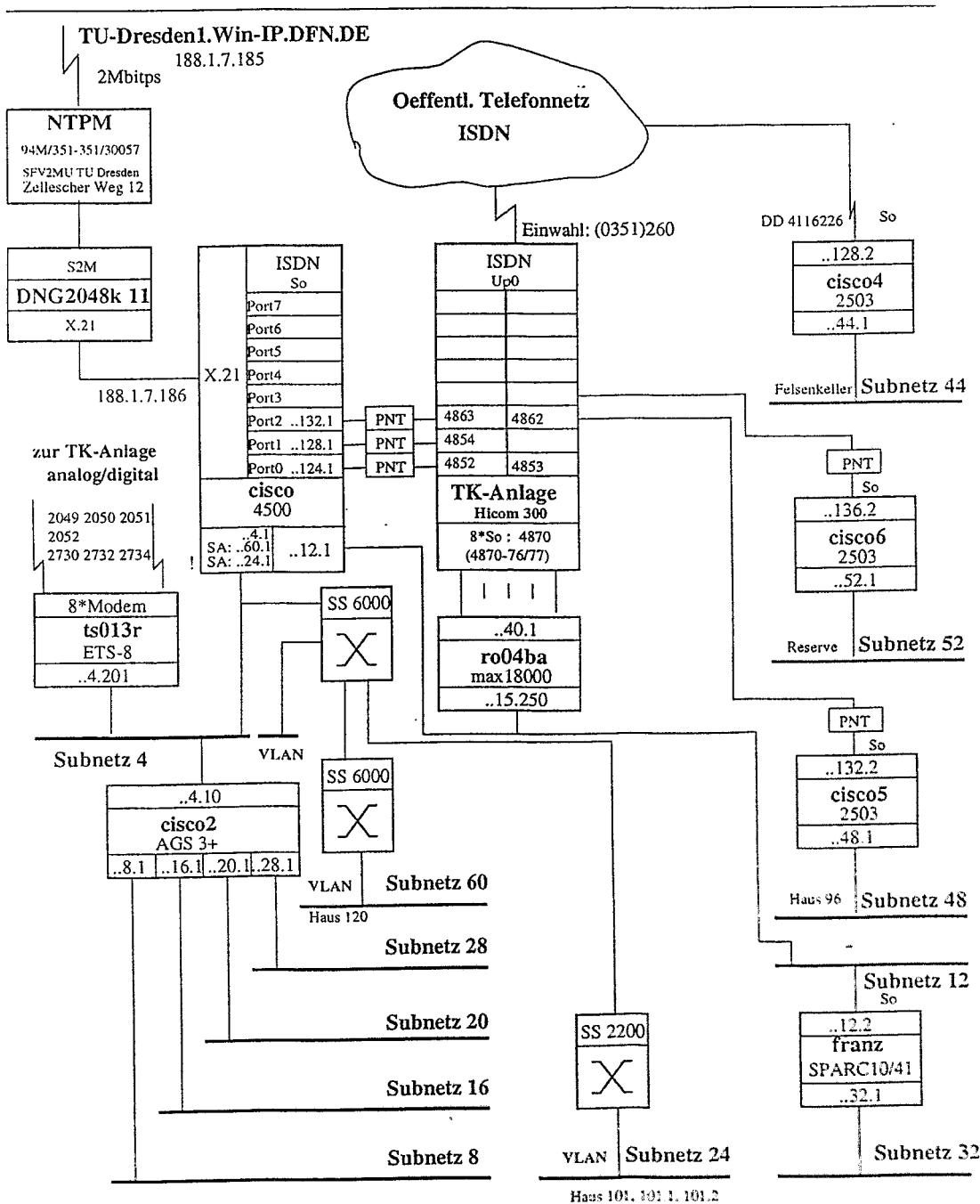
Als Name-Server dienen:

- ns1 (149.220.4.2) Master-Name-Server
- ns2 (149.220.8.11) Slave-Name-Server
- noc.rrz.uni-koeln.de (134.95.100.209) Slave-Name-Server
- oldman (149.220.4.240) Silent Name-Server



FZR Rechenzentrum Gruppe Datennetze **FZR-LAN : Subnetz-Struktur, Okt. 98** fz-rossendorf.de vkta-rossendorf.de nuklearmedizin.de Datei: tzrlan8 fig

Abbildung 2: Subnetzstruktur des lokalen Netzes am Forschungsstandort Rossendorf



FZR e.V. Rechenzentrum

FZR-LAN : Netz-Struktur /VLAN/ ISDN
 Dater vlan2 VLAN: Sub 4, Sub 60, Sub 24 Nov. 1998

Netzgruppe **FZR-LAN**

Abbildung 3: Struktur des lokalen Netzes, einschließlich ISDN-Anbindungen

6 Benutzerservice

Die Aufgaben der Gruppe Benutzerservice umfassen die

- Unterstützung zentraler und dezentraler PCs, einschließlich der Koordinierung und Beschaffung von Software, die
- Beratung der Nutzer zentraler und dezentraler Rechentechnik auf den Gebieten
 - Numerik, Bibliotheken, Computeralgebra,
 - Programmierung (Fortran, C, OOP)
 - Parallelisierung
 - Visualisierung
 - Textverarbeitung sowie
- Mitarbeit (Numerik, Programmierung, Parallelisierung) an ausgewählten wissenschaftlich-technischen Projekten.

Ein Teil der inzwischen routinemäßig durch die Gruppe Zentrale Server angebotenen Dienste wurde von der Gruppe Benutzerservice erschlossen und installiert.

Alle an den Zentralen Servern neu bereitgestellten Software-Produkte wurden in Benutzerseminaren vorgestellt. Zu jedem Seminar erhielten die Nutzer Unterlagen mit Arbeitsanleitungen. In letzter Zeit wurden stattdessen entsprechende WWW-Seiten bereitgestellt.

Im Berichtszeitraum fanden insgesamt 12 Benutzerseminare statt. Ihre Themen sind im Abschnitt 6.7 zusammengestellt.

Eine neue Dimension erhält der Benutzerservice durch die Einrichtung einer zentralen Administration der PCs im Forschungszentrum.

6.1 Unterstützung dezentraler PCs

Die Unterstützung der dezentralen PCs beschränkte sich in der Vergangenheit entsprechend den damaligen personellen Kapazitäten auf die Koordinierung und Beschaffung von PC-Software sowie auf die Bereitstellung von Netzlaufwerken an den Zentralen Servern.

Entsprechend den Anforderungen der Nutzer und den Möglichkeiten von Windows NT begann der Benutzerservice mit dem Aufbau eines zentrumsweiten Supports zur Installation und Wartung von Microsoft-Software. Er schließt die Unterstützung am FZR benötigter Anwendungsprogramme ein.

Mit dem Ziel, eine zentralisierte Wartung der PC-Software einzuführen, wurde eine Testinstallation des Systems NICE aus dem CERN durchgeführt. In die Arbeiten waren Gastwissenschaftler aus dem VIK Dubna einbezogen, die über entsprechende Erfahrungen mit NICE verfügen. Inzwischen wurde die automatische Installation der Betriebssysteme Windows 95 und Windows NT 4.0 mit 6 verschiedenen Netzwerkkarten realisiert. Dabei hat der Nutzer nur eine Startdiskette einzulegen. Die Installation läuft dann automatisch ab.

Begonnen wurde ferner der Aufbau einer zentrumsweiten NT-Domainstruktur. Erster Schritt

hierbei waren die Einrichtung und Konfiguration der Primär- und Backup-Domain-Controller. In allen Subnetzen sind WIN-Server eingerichtet.

Im Aufbau befinden sich die servergestützte Installation und Konfigurierung von Software-Paketen.

Um das Systemmanagement von Microsoft- und UNIX-Welt zusammenzuführen, wurde ein gemeinsames Verwaltungsschema erarbeitet und der Aufbau einer zentralen Nutzerdatenbank begonnen.

Linux wird neben Windows NT im wissenschaftlich-technischen Bereich immer mehr Bedeutung erlangen. Die Abteilung FVTK bietet deshalb neben der Unterstützung für UNIX auch die Beratung von Linux-Anwendern an. FVTK selbst nutzt Linux als Betriebssystem des neuen Datenbankservers für Oracle 8.

Da auf Linux vorrangig GNU-Software eingesetzt wird, wartet der Benutzerservice GNU-Produkte auch für PCs (C-Compiler, Ghostview, Ghostscript, Emacs und weitere Werkzeuge). Das Spektrum der unterstützten GNU-Compiler wurde im Berichtszeitraum um die Sprachen C++, Fortran 77, Ada und Eiffel erweitert.

6.2 Numerik, Programmbibliotheken, Computeralgebra

Neben der operativen Beratung und Unterstützung der Nutzer auf dem Gebiet der numerischen Mathematik (einschließlich Programmierung) wurden folgende Leistungen erbracht:

- Entwicklung des Programms PICO für radiochemische Untersuchungen
- Verbesserung und Erweiterungen am Programmsystem RAPI zur Berechnung und graphischen Darstellung von Strahlenexpositionen
- Bereitstellung von Algorithmen zur Entfaltung von Tiefenprofilen, die mittels Positronenstrahlen verschiedener Energien gewonnen wurden,
- Tests und Bereitstellung von numerischen Algorithmen
 - für die Bestimmung von Eigenwerten komplexer hermitescher Matrizen und
 - für kubische Spline-Interpolation
- Hilfe bei gnuplot-Anwendungen.

Einen Schwerpunkt der Nutzerunterstützungen bildeten Abschirmungsrechnungen für das Projekt ELBE. Hierzu wurde der EGS4-Code für den Elektron-Positron-Transport zur Bestimmung der Dosisleistung installiert und erprobt. Vergleichsrechnungen wurden mit den Programmsystemen MCNP (Monte Carlo N Particle Transport Code) und GEANT ausgeführt. Zur Bewertung der Ergebnisse wurden Kontakte zu den Autoren von EGS4 und zu erfahrenen Anwendern von MCNP hergestellt.

Folgende numerische Bibliotheken sind auf neuen Release-Ständen an den Zentralen Servern verfügbar:

- CERN-Library, Vers. 98 (an pales, rita, rike, rhea, vesta)
- NAG-Fortran-77-Library
- NAG-Fortran-90-Library, Rel 2.a
- NAG-Text-Ware-System für Fortran-77-Library Dokumentationssystem (an hera und rita)
- HP-MLIB
- NAG-C-Library, Mark 4 (an junjo)
- Textware-C-Library Dokumentationssystem (an junjo)

Für Ad-hoc-Rechnungen sind auf Wunsch von Nutzern die folgenden Computeralgebra-Systeme installiert und erschlossen:

- AXIOM, Vers. 2.1 (an rita)
- Maple, Rel. 4 (an rike und Rel. 5 (an hera)
- Mathematica V.3.0 (an rike)

Die Bereitstellung mehrerer Computeralgebra-Systeme am Forschungsstandort entspricht den Wünschen der Nutzer, da die Systeme unterschiedliche Stärken und Schwächen haben (siehe auch Jahresbericht 1995/96).

6.3 Programmierung

Die Abteilung FVTK unterstützt neben Fortran und C auch C++ und andere objektorientierte Sprachen. Sie bietet entsprechende Anwenderunterstützung an.

Fortran ist nach wie vor die am häufigsten verwendete Programmiersprache am Forschungsstandort. Neben Fortran 77 stehen Fortran 90 und in einer Implementierung der Numerical Algorithm Group (NAG) auch Fortran 95 zur Nutzung zur Verfügung.

Compiler wurden auf neue Release-Stände gebracht. Damit waren im Juni 1999 folgende Compiler an den Zentralen Servern installiert:

- Fortran-90-Compiler (an pales)
- xlf90-Compiler (an hera)
- NAG-Fortran-95-Compiler (an rike)

Die Nutzer erhielten Unterstützung beim Einsatz der neuen Software-Produkte.

Die verfügbaren Rechnersysteme und ihre Fortran-Compiler wurden hinsichtlich ihrer Leistungsfähigkeit mittels der Programme

- LINPACK-Benchmark und
- Matrizeninversion nach Gauß-Jordan

getestet.

Für mehrere Fortran-Programme wurde der Convex Performance Analyzer (CXpa) eingesetzt, um die Programmeinheiten zu ermitteln, die die meiste Zeit beanspruchen und optimiert werden sollten. Durch die Optimierung dieser Einheiten konnte die Effizienz der ausgewählten Fortran-Programme spürbar verbessert werden.

6.4 Parallelisierung

Im Auftrage und in Zusammenarbeit mit dem Institut für Ionenstrahlphysik und Materialforschung des FZR entstand ein parallelisierter Code zur Simulation von Entstehung, Wachstum und Ostwald-Reifung von Nanokristallen. Um maximale Effizienz zu erreichen, wurde der Code auf dem HP-Parallelrechner SPP2000 mit shared Memory implementiert. Die Untersuchung verschiedener Optimierungs- und Parallelisierungsstrategien zeigte, daß bereits auf der ersten Stufe der Parallelisierung, der sequentiellen Optimierung, erhebliche Geschwindigkeitsgewinne erreichbar sind. Insbesondere spielt die Optimierung von Variablenstrukturen hinsichtlich der effizienten Nutzung von Cache-Systemen eine entscheidende Rolle. Wegen der ccNUMA-Architektur (Cache Coherent Non Uniform Memory Access) der benutzten Parallelmaschine gewinnt die Cache-Optimierung sehr an Stellenwert.

Wie sich bei der Erarbeitung des o.g. Codes wiederholt zeigte, ist das Hinzuziehen informatischer und mathematischer Fachkompetenz des Rechenzentrums bei der Codierung parallelierter Algorithmen unumgänglich. Insbesondere sind Detailkenntnisse von Maschinen- und Systemarchitektur eminent wichtig, um effiziente Programmcodes zu erhalten.

Das Problem erhält eine weitere Dimension, wenn Server unterschiedlicher Architektur und Leistung zu einem virtuellen Cluster vereinigt werden. Entsprechende Untersuchungen fanden im Rahmen einer Studie statt, über die im Abschnitt 9 berichtet wird.

Konkrete Unterstützung erhielt das Institut für Sicherheitsforschung bei der Parallelisierung eines Nutzerprogramms.

6.5 Visualisierung

Die Visualisierung von Daten hat sich zu einem wichtigen Dienst der Abteilung entwickelt. Er reicht von der Unterstützung einfacher 2D-Graphik (gnuplot) bis zur Entwicklung anspruchsvoller Animationswerkzeuge. Letztere wurde im Berichtszeitraum auf hohem Niveau fortgeführt. Sie erbrachte wichtige Neuerungen.

Über das Drucken von Dokumenten und Folien am A3-Color-Drucker Canon CLC700 in nahezu photorealistischer Qualität sowie über das Drucken von A0-Postern war bereits im Abschnitt 4.4 berichtet worden.

Die Möglichkeiten zur Erzeugung von Visualisierungen und nunmehr auch von Animationen wissenschaftlicher Daten wurden neben den bisherigen Zugängen über AVS und IDL um ein Linux-basiertes Werkzeug erweitert.

Neben der Generierung von Motion-JPEG-Datenströmen zur Ausgabe auf der Parallax-Karte können nunmehr auch MPEG-1- und MPEG-2-Datenströme erzeugt werden. Letztere sind besonders zum Abspielen per Software und von CDs geeignet. Sie besitzen gegenüber analogen Aufzeichnungsverfahren eine bedeutend höhere Qualität. Erste Anwendung war die Digitalisierung eines Videos über das ROBL-Strahlrohr für die Hannover-Messe 1999.

Die Herstellung weiterer AVS-Module für die flexible Generierung von JPEG-Daten aus AVS wird eine Hauptkomponente für den Aufbau einer leistungsfähigen Video-Generierung bilden.

Im Auftrage des Instituts für Sicherheitsforschung wurde ein Programm zur Visualisierung und Animation von Dichtedaten aus dem "Gasdynamic Trop-"Experiment am Budker-Institut Novosibirsk geschrieben.

Eine völlig neue Art der Präsentation entstand mit der Darstellung der ELBE-Quelle: Unter Nutzung vorhandener Software wurde ein interaktives WWW-Informationssystem auf der Grundlage von VRML (Virtual Reality Modelling Language) geschaffen. Während bisherige Präsentationen und Visualisierungen wenig Raum für Präsenz und Interaktion ließen, bietet die Beschreibung mittels VRML bedeutend mehr Freiheiten. VRML ist insbesondere geeignet, Interaktionen mit graphischen Elementen zu organisieren: Nach Generierung eines Ereignisses durch den Betrachter ist es möglich, auf Objektparameter einzuwirken und somit eine Reaktion des visualisierten Systems auszulösen.

An den Zentralen Servern werden die High-end-Visualisierungssysteme AVS (Advanced Visualization System, Fa. UNIRAS) und IDL (Interactive Data Language) zur Nutzung bereitgestellt und gepflegt. Das äußerst leistungsfähige, aber ressourcenintensive AVS (Version V5.02) bildet die Grundlage für die Erzeugung von Videosequenzen. Seine sonstige Anwendung ist zugunsten von IDL am Standort spürbar zurückgegangen. Das FZR verfügt inzwischen über 10 IDL-Lizenzen.

Seit Herbst 1996 existiert in der Abteilung eine komplette, professionell arbeitende Linie zur Herstellung von Videosequenzen. Sie enthält eine Workstation HP 9000/715 (ikon), die mit einer PARALLAX-Karte bestückt ist.

Die Weiterentwicklung dieser Technologie galt im Berichtszeitraum der Verfeinerung und der Erweiterung der Werkzeuge zur automatisierten Herstellung von Bildwechselfolgen. Sie betreffen vor allem die Problematik Szene-Kamera-Rotation-Positionsveränderungen. Die Arbeiten sind im Abschnitt 9 ausführlicher beschrieben.

Im Berichtszeitraum wurden drei Nutzer- und vier Demonstrationsvideos hergestellt.

Gepflegt, an weiteren Servern installiert bzw. um neue Komponenten ergänzt wurden die folgenden Graphikwerkzeuge:

- `xv` – X11-basierter Previewer zur Darstellung und Konvertierung von Abbildungen in verschiedenen Graphikformaten
- `xfig` – Programm zur interaktiven Erzeugung graphischer Abbildungen unter X11
- `xmgr` – Graphikprogramm zum Plotten von Datenfiles und math. Funktionen
- `mpeg_play` – X11-basierter Previewer für Videos
- `ImageMagick` – Softwarepaket zur Darstellung und Konvertierung von Abbildungen in verschiedenen Graphikformaten.
- `gnuplot` (UNIX V3.7)

6.6 Textverarbeitung

Die Aktivitäten auf dem Gebiet des Textsystems \LaTeX (\TeX) waren 1997/98 einerseits auf die Betreuung der Nutzer ausgerichtet. Andererseits galt es, trotz personeller Einschränkungen einen Überblick über das sich stürmisch entwickelnde Fachgebiet zu wahren und eine FZR-spezifische und nutzergerechte Auswahl von bereitzustellender Software zu treffen.

Folgende \LaTeX (\TeX)Versionen und \LaTeX (\TeX)Werkzeuge sind installiert:

- $\LaTeX_{2\epsilon}$ <1998/06/01> (an hera), <1997/12/01> (an rita, rike und junio)
- sowie Macropakete und Styles für das Publizieren bei
 - American Physical Society (Physical Review A, B C, D, E or Letters)
 - American Mathematical Society $\mathcal{A}\mathcal{M}\mathcal{S}$
 - Optical Society of America
 - American Institute of Physics (18 Zeitschriften)

- Elsevier Science Publishers B.V.
- Springer-Verlag

Auch die kyrillische $\LaTeX_2\epsilon$ -Version ist als zusätzlich installiertes Formatfile verfügbar. Im Berichtszeitraum wurden folgende \LaTeX -Werkzeuge gepflegt, an weiteren Zentralen Servern installiert bzw. um neue Komponenten ergänzt:

- latex2html – Konverter, der aus einem \LaTeX - ein analoges HTML-File mit Ausnahme einiger \LaTeX -Konstruktionen erzeugt
- Ghostscript – X11-basierter PostScript-Previewer V4.0 mit Funktionserweiterungen
- Ghostview – erweiterte Bedienoberfläche für Ghostscript
- xtem – eine komfortable Benutzeroberfläche für die $\LaTeX(\TeX)$ -Umgebung
- LyX- Benutzeroberfläche, die das WYSIWYG-Prinzip bei der Erstellung von \LaTeX -Dokumenten realisiert,

6.7 Informationsdienste

Das FZR ist seit 1994 im Internet präsent. Es betrieb bis 1997 den WWW-Server CERN, Version 3.0 an der C3220 (hera), danach den WWW-Server Apache, der seit 1999 auf einer IBM RS/6000 F50 (nike) installiert ist. Bei der Betreuung des Servers und der Pflege der Dokumente besteht folgende Arbeitsteilung:

- Für den Betrieb und die Pflege des WWW-Servers ist die Abteilung FVTK verantwortlich.
- Die Gestaltung der WWW-Dokumente obliegt einer Redaktionsgruppe aus Vertretern der Organisationseinheiten im FZR.

Die Statistik des WWW-Servers registrierte im zweiten Halbjahr 1998 monatlich im Mittel 60.000 Zugriffe durch 3.100 externe Anforderer.

Als Browser hat sich am Standort der Netscape Navigator durchgesetzt. Er steht in der aktuellen Version 4.5 am Zentralen Server IBM RS/6000 S70 (hera) zur Verfügung und wird auch PC-Nutzern empfohlen. Mosaic wird kaum noch eingesetzt.

Für das Rossendorfer Bibliotheksinformationssystem (ROBIS) wurde das Datenbanksystem Oracle in der Version 8.0.5 auf dem Linux-Server atlas installiert. Die Oracle-Version 7.3 ist nach wie vor auf dem Server IBM RS/6000 Mod. 591 (rike) für andere Anwendungen verfügbar.

Zur Unterstützung der Institute und Abteilungen des FZR bei der Organisation von Konferenzen wurde ein WWW-basiertes Werkzeug zur Anmeldung entwickelt. Es besteht aus einer Schablone für das Anmeldeformular und einem Programm zur Erfassung der Anmelde-daten und zur Benachrichtigung der Organisationen. Das Institut für Radiochemie und das Institut für Ionenstrahlphysik und Materialforschung des FZR konnten das Werkzeug bereits mit Erfolg einsetzen.

6.8 Benutzerseminare

Datum	Nr.	Referent	Thema
20.05.97	74	E.Müller	NAG-Bibliotheken
27.05.97	75	H.Böttger	Animation durch Kamerabewegungen (Vorstellung neuer Werkzeuge und Ergebnisse)
17.06.97	76	U.Fromm E.Müller M.Kunicke	Start des Nutzerbetriebs am Rechner junio - Technische Daten und Betriebssystem - Informationen zur Nutzung - Programmierung mit Fortran und C - Paralleles Rechnen
20.01.98	77	N.Schmeißer	C-Grundkurs I Einführung Typkonzept
27.01.98	78	N.Schmeißer	C-Grundkurs II - Kontrollstrukturen - Funktionen - Blockkonzept
03.02.98	79	N.Schmeißer	C-Grundkurs III - Modulkonzept - Preprozessor - Bibliotheken - Dynamische Speicherverwaltung
22.09.98	80	N.Schmeißer	C++-Grundkurs I (Konzepte des objektorientierten Programmierens)
29.09.98	81	N.Schmeißer	C++-Grundkurs II - Crash Course C - Ad-hoc-Polyphormismus - Strukturen und Klassen
06.10.98	82	N.Schmeißer	C++-Grundkurs III - höhere Konzepte - Templates - weitere OO-Sprachen
09.03.99	83	U.Fromm E.Müller	Neue Server (IBM S70, IBM F50) - Vorstellung der neuen Server - Programmierung mit Fortran und C - Computeralgebrasysteme
16.03.99	84	E.Müller U.Fromm	Nutzung von HP V-Class (pales) und IBM S70 (neue hera) - Fortran und C am Rechner pales - CPU-Zeitvergleiche - NAG-Bibliotheken und Informationssysteme - AIXwindows-Umgebung am Rechner IBM S70

- 14.04.99 85 N.Schmeisser Linux
- Linux-Kernel
 - Installation und Konfiguration
 - Anwendungen
 - X11, Desktops (GNOME, KDE), KOffice

7 Wissenschaftliche Kooperation

Die Abteilung ist um fachliche Kontakte zu gleichen Einrichtungen im In- und Ausland bemüht. Mitarbeiter der Abteilung sind aktive Mitglieder folgender Gremien und Arbeitsgruppen:

- DFN-Verein:
Dr. R. Fülle (im Auftrage des FZR)
- Arbeitskreis EDV der WGL:
Dr. R. Fülle (stellv. Vorsitzender)
- Koordinierungsausschuß für Datenverarbeitung (KODA) der HGF:
Dr. R. Fülle (als Gast)
- Arbeitsgruppe Graphische Datenverarbeitung der HGF:
H. Böttger (als Gast)
- DANTE (Deutsche \LaTeX -Nutzergruppe):
Dr. G. Vinel
- DIGI (Deutsche Interessengemeinschaft Internet):
Dr. W.-J. Linnemann
- Deutsche CONVEX-Nutzergruppe (bis 1998):
Dr. M. Kunicke (Vorsitzender)
- NAG User Association:
E. Müller

Die Zusammenarbeit mit dem Vereinigten Institut für Kernforschung (VIK) Dubna (Rußland), Laboratorium für Rechentechnik, wurde intensiv fortgesetzt. 1997 bis Mitte 1999 arbeiteten drei Mitarbeiter des VIK als Gastwissenschaftler des FZR an Aufgaben der Abteilung mit:

- V. Krylov:
Gemeinsame Erarbeitung einer Pilotanwendung für die Video-Animation von wissenschaftlichen Daten; Abschluß der Arbeiten;
Dezember 1997
- V. Pervushov:
Installation und Anpassung einer NICE-Testversion;
Dezember 1998
- V. Galaktionov, V. Pervushov
Programmierung von Werkzeugen zur Nutzerverwaltung;
Mai 1999

Die Abteilung beteiligte sich auch im Berichtszeitraum an der Ausbildung von Studenten und Schülern:

- Vorlesungen und Übungen an der TU Dresden, Fakultät Mathematik und Naturwissenschaften: "Rechentechnische Hilfsmittel des Physikers" (Doz. Dr. R. Fülle)
- Vorlesungen und Übungen an der Universität Dubna: "Paralleles Rechnen" (Dr. M. Kunicke)
- Seminarvorträge an der TU Dresden, Fakultät Mathematik und Naturwissenschaften sowie Fakultät Elektrotechnik:
 - "Paralleles Rechnen – Eine Einführung" (Dipl.-Math. N. Schmeißer)
 - "Objektorientiertes Programmieren – Theorie, Methodik, Anwendung" (Dipl.-Math. N. Schmeißer)
- Betreuung von Diplom-, Beleg- und Praktikumsarbeiten
 - "Installation einer Testversion des PC-Management-Systems WinWatch in das System SPECTRUM"; Praktikumsarbeit von Roland Maaz, Betreuer im FZR: Dr. W.-J. Linnemann
 - "Implementierung eines Basisschemas zur Berechnung der Geschwindigkeitspolaren von Segelflugmodellen" Praktikumsarbeit von Ulrich Friedemann, Modellflugclub Rossendorf e.V. Betreuer im FZR: Dipl.-Math. N. Schmeißer, Dipl.-Ing. H. Langenhagen
 - "Berechnung der Geschwindigkeitspolaren eines Segelflugmodells unter Einbeziehung im Windkanal vermessener Flügelprofile"; Praktikumsarbeit von Thomas Brenner, Betreuer im FZR: Dipl.-Math. N. Schmeißer, Dipl.-Ing. H. Langenhagen. Vortrag bei der Nachwuchstagung der Deutschen Gesellschaft für Luft- und Raumfahrt

8 Konferenzbeiträge und Vorträge

Schmeißer, N.:

“A Parallel Code for Kinetic 3D Lattice MC Simulations of Nucleation, Growth and Ostwald Ripening of Nanocrystals“;
HP/CONVEX High Performance User Group, Dallas, Februar 1997

Fülle, R.:

“The IT-Infrastructure of the Forschungszentrum Rossendorf“;
VIK Dubna, September 1997

Böttger, H.:

“Animation durch Kamerabewegungen“ (mit Vorführungen);
Advanced Visual Systems, Anwender-Konferenz, November 1997

Böttger, H.:

“Gestaltung von Video-Vor- und Abspann“
Vorstellung neuer Tools,
Vorführung von 5 Demo-Videos
“Animation durch Kamerabewegung“;
Universität Ulm, Workshop “Nutzung von AVS zur Herstellung von Videos“, Februar 1998

Schmeißer, N.:

“Modelling the State of Distributed Objects and Redistribution Based on the Object Dependency Graph“;
VIK Dubna, November 1998

Kunicke, M.:

“Bringt EPIC eine neue Qualität in der DV?“;
HTW Dresden, Seminar des FB Informatik/Mathematik, Juni 1999

9 Ausführliche Berichte zu ausgewählten Themen

Nils Schmeißer:	Entwurf von Werkzeugen zur zentralen Nutzerverwaltung
Erhard Müller, Hartmut Guratzsch, Bärbel Naumann:	Dosisberechnungen für Beamdumps des ELBE-Beschleunigers
Horst Böttger:	Neue Hilfsmittel für die Herstellung von Videos

Diese Berichte sind Beispiele aus unseren Tätigkeitsfeldern

- Zentrale Nutzerverwaltung
- Mitarbeit an Projekten und
- Visualisierung

Entwurf von Werkzeugen zur zentralen Nutzerverwaltung

Nils Schmeißer

1 Motivation

Die Verwaltung von Nutzern einer heterogenen Rechnerwelt wie am Forschungsstandort Rosendorf ist im allgemeinen ein sehr aufwendiges Unterfangen, da selbst minimale Änderungen von Daten des Nutzers, zum Beispiel die Zuweisung einer neuen Telefonnummer, die Änderung von Systemdateien/ bzw. Einträgen auf allen Rechnern, auf denen der Nutzer registriert ist, nach sich ziehen. Zwar kann die Verwaltung durch die Nutzung von NIS bei UNIX bzw. einer Domainstruktur bei WindowsNT vereinfacht werden, aber die zentrale Verwaltung von Nutzern beider Welten ist damit noch nicht gegeben.

Die entwickelten und hier vorgestellten Werkzeuge ermöglichen eine zentrale Nutzerverwaltung für beliebige Betriebssystemfamilien. Einigt man sich innerhalb der vorhandenen Systeme auf wenige Regeln, so ist es ohne weiteres möglich, eine zentrale, von den Betriebssystemwelten unabhängige Nutzerverwaltung einzuführen. Die Unabhängigkeit der Nutzerverwaltung ist außerdem auch deshalb anzustreben, weil

1. damit eine Schnittstelle zu den Personaldatenbeständen der Verwaltung (z.B. Telefonbuch) gegeben ist und
2. Betriebssysteme erfahrungsgemäß häufiger wechseln als die Hauptgruppe der Nutzer.

Die für eine zentrale Nutzerverwaltung einzuhaltenden Regeln sind sehr einfach:

- Die Administratoren müssen sich der gemeinsamen Verwaltungsstruktur unterordnen, d.h. alle Aktionen müssen mit den Werkzeugen des Nutzerverwaltungssystems ausgelöst werden
- Gleichartige Verwaltungskonzepte sollten auf allen Betriebssystemen gleichartig interpretiert und eingesetzt werden, z.B. UNIX und WindowsNT:
 - NIS-verwaltete Einheiten entsprechen Domänen
 - UNIX-Gruppen entsprechen WindowsNT-Gruppen
 - Windows-Arbeitsgruppen werden nicht genutzt
 - Nutzernamen für ein und dieselbe Person sind gleich.

Das nachfolgend beschriebene System wurde im ersten Halbjahr 1999 unter Mitwirkung zweier Gastwissenschaftler aus dem VIK Dubna konzipiert und teilweise implementiert. Hauptaugenmerk lag dabei auf

- der Interoperabilität, im Falle des FZR auf der Unterstützung sowohl von WindowsNT als auch UNIX
- der klaren Trennung zwischen systemspezifischen Aufgaben und Verwaltungsaufgaben, sowie

- einer streng modularen Implementation.

Wesentlich für die Nutzbarkeit des Systems sind neben der guten Handhabbarkeit ein hohes Maß an Fehlertoleranz und die Fähigkeit zum Wiederanlauf im Fehlerfall.*

Im Zusammenhang mit der Implementation kamen moderne Technologien der Programmierung zum Einsatz (Fachsprachenkonzepte, Konzepte des verteilten objektorientierten Programmierens), auf die in diesem Beitrag ebenfalls verwiesen wird.

2 Konzept

Planung und Konzeption

Für die Erstellung des zu implementierenden Konzeptes waren folgende Forderungen ausschlaggebend:

1. Haltung aller relevanten Daten in genau einem zentralen Datenbestand
2. die Wartung bzw. der Zugriff auf diesen Datenbestand muß von beliebiger Stelle im Standort aus durch autorisierte Personen möglich sein
3. Ableitung aller zur Einrichtung und Pflege von Nutzerkonten nötigen Daten direkt aus dem Datenbestand
4. automatisierte Übernahme von Änderungen bzw. Neueintragungen im zentralen Datenbestand in die Systemeinstellungen
5. Unterstützung verschiedener Betriebssystem-Umfelder (z.B. UNIX und WindowsNT)

Aus diesen Forderungen leitet sich unmittelbar die Struktur des Management-Systems ab

- Datenbankgestützte Lösung (SQL-fähiges Datenbanksystem)
- Zugriff auf die Datenbank via WWW-Browser (da plattformunabhängig und de facto überall verfügbar)
- Dienststruktur in Form von Daemons

Die Systemarchitektur ist in Abbildung 1 grafisch dargestellt.

Verfahren

Ein Nutzer/Antragsteller wählt ein WWW-Formular aus, um

- einen Antrag zur Erteilung einer Zugangsberechtigung (Nutzerkonto) zu stellen,
- einen Änderungswunsch anzumelden oder
- die Löschung seiner Zugangsdaten zu beantragen.

*Die Modularisierung (Dienstaufteilung) erlaubt im Fehlerfall die Analyse und den kontrollierten Neustart von Diensten.

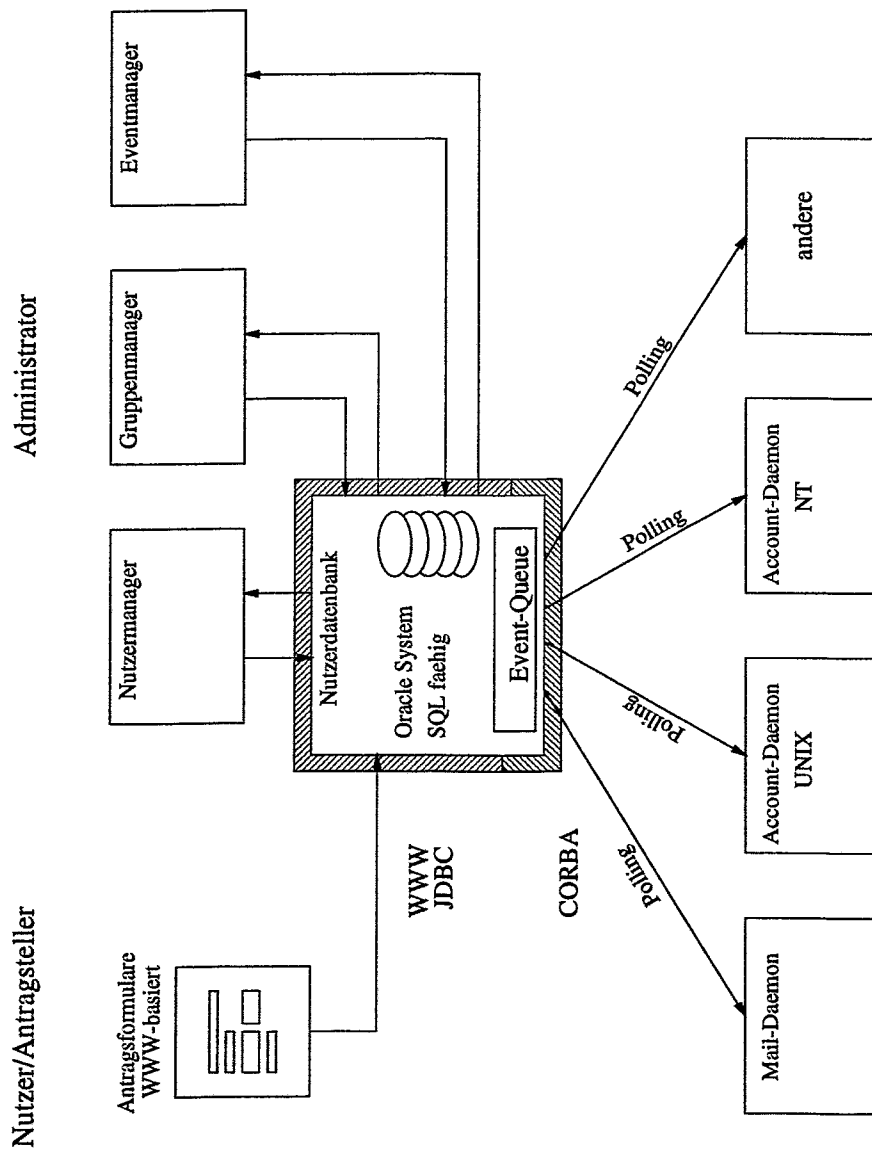


Abbildung 1: Dienststruktur und Einordnung der Nutzerdatenbank

Die Daten des Antragsformulars werden in einer temporären Tabelle der Datenbank abgelegt, der entsprechende Administrator wird automatisch per E-mail über das Eintreffen eines neuen Auftrages informiert.

Der Administrator startet seinerseits einen passenden Klienten (Manager), mit dessen Hilfe die eingegebenen Daten bearbeitet und die notwendigen Aktionen abgefordert werden können. Für jede, über einen Klienten angeforderte Aktion wird ein Eintrag in eine Warteschlange generiert.

Die auf verschiedenen Betriebssystemen installierten Dienste fragen diese Warteschlange zyklisch ab und führen die angeforderten Aktionen aus.

3 Implementation

Datenbankentwurf

Die Struktur der Datenbank ergibt sich aus

- den zu registrierenden "Hauptdaten" für einen Nutzer (Name, Vorname, Büro),
- den "Kommunikationsdaten" (Telefonnummer, E-mail-Adresse),
- den zum Anlegen einer Zugangsberechtigung nötigen Informationen.

Daten über den Aufbau der zu verwaltenden Rechnerstruktur mußten in dieser Entwurfsphase ebenfalls integriert werden. Der Aufbau einer separaten Rechner-/ Hardwaredatenbank folgt aber als nächster logischer Schritt. Auf das Design der Nutzerdatenbank hat die spätere Auskopplung der entsprechenden Tabellen keinen Einfluß.

Auf eine detaillierte Beschreibung der Datenbank, insbesondere der Relationen und Integritätsregeln, muß an dieser Stelle verzichtet werden, da sie den Rahmen dieses Beitrages sprengen würde. Bezüglich der exakten Beschreibung sei auf [Sch99] verwiesen.

Datenbankschnittstelle

Um die Unabhängigkeit vom Datenbanksystem zu gewährleisten, wird zum Zugriff auf den Datenbestand die Verwendung von SQL nach ANSI X3.135-1992/ISO/IEC 9075:1992 ([ANS89], [ISO92]) festgelegt.

Zwei Gründe zwingen an dieser Stelle zur Festlegung einer streng einzuhaltenden Übereinkunft über die Schnittstelle(n) zur Datenbank:

1. Die Implementation kann nicht durch eine einzige Person erfolgen.
2. Um den Aufwand bei der Implementation so gering wie möglich zu gestalten, sollte für alle unterstützten Plattformen nur eine minimale plattformabhängige Komponente vorhanden sein.

Aus der Forderung nach standortunabhängiger Verfügbarkeit folgt zwingend die Nutzung der *Java Database Connectivity Option* (JDBC) in der reinen Java-Form[†] (JDBCThin).

Da die Dienste systemnahe Aufgaben erfüllen, werden sie vorteilhaft in C/C++ implementiert.

[†]Oracle bietet z.B. neben einem reinen Java Treiber auch eine JDBC Variante, die das Oracle Call Interface (OCI) nutzt.

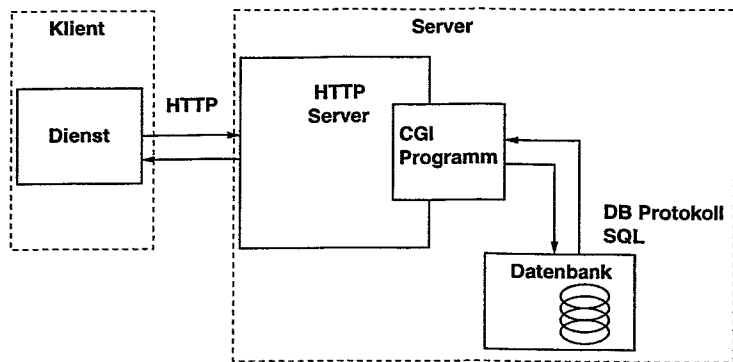


Abbildung 2: Datenbankzugang über das Common Gateway Interface

Der Datenaustausch zwischen Dienst und Datenbank kann z.B. unter Nutzung einer Bibliothek erfolgen, die alle zum Verbindungsaufbau und Versenden von SQL-Anweisungen nötigen Funktionen enthält (z.B. die OCI-Bibliothek). Diese Bibliothek muß allerdings auf allen Plattformen zur Verfügung stehen und sollte natürlich nicht zu teuer sein. Letzteres ist bei der OCI-Bibliothek **nicht** der Fall.

Eine Möglichkeit, hier Kosten zu sparen und dazu noch ein plattformunabhängiges Konzept einsetzen zu können, bietet das *Common Gateway Interface* (CGI). Server-seitig wird hierfür ein CGI-Programm eingerichtet, daß via SQL auf den Datenbankserver zugreift und die Ergebnisse der Anfrage als HTML-Text an den Server zurückliefert (Abbildung 2). Die Einbettung dieser Technologie in das *Request-Broker-Konzept* wird im Abschnitt 9 genauer beschrieben.

Ein HTTP-*Stub* † (Komponente des Dienstes, die zum Zugriff auf den Datenbank-Server verwendet wird) kann in der Regel relativ einfach implementiert werden. Insbesondere auf POSIX-konformen Systemen (dazu zählen fast alle UNIX-Derivate **und** WindowsNT) sind dabei nur minimale Änderungen bei der Portierung des Dienstes nötig.

4 Technologie

Datenbankzugriff via JDBC

JDBC stellt die zum Zugriff auf eine SQL-fähige Datenbank notwendige Grundfunktionalität in Form von Klassen bereit. Unmittelbar JDBC-basierte Anwendungen bestehen aus einer Folge von SQL-Anweisungen, die mittels spezieller Nachrichten (im Sinne des objektorientierten Programmierens) an die Datenbank gesendet werden. Zum Auswerten von Ergebnissen steht eine spezielle Containerklasse zur Verfügung. Die Nutzung von JDBC erlaubt einen anweisungsorientierten Zugang zur Datenbank.

In einem relationalen Datenbankmodell stellen *Tabellen* und *Views* die "höchsten" Elemente dar. Die Eigenschaften von Tabellen und Views lassen sich sehr allgemein in Form abstrakter Basisklassen beschreiben.

† Als *Stub* wird eine Subroutine bezeichnet, die die Verbindungsschicht eines Kommunikationsprotokolls transparent auf einen Funktionsaufruf abbildet.

Tabelle :

- besteht aus benannten und typisierten Spalten
- erlaubt Einfügen, Suchen, Verändern und Löschen von Datensätzen (Records)
- implementiert ein Cursor-Konzept (ein Cursor ist ein Zeiger, der innerhalb einer Gruppe von Datensätzen einen bestimmten adressieren kann)
- kann Datensätze exportieren oder importieren

View :

- hat alle Eigenschaften einer Tabelle
- seine Spalten sind eindeutige Abbildungen auf Spalten von Tabellen oder Views, sie erben damit die entsprechenden Typ-Eigenschaften und Constraints, können aber andere Namen tragen
- modelliert zusätzlich Relationen zwischen Tabellen

Diese Eigenschaften können mit den sprachlichen Möglichkeiten von Java auf Attribute einer Klasse abgebildet werden:

Attribut	Tabelleneigenschaft
String tablename	Tabellenname
Vector columnnames	Spaltennamen
Vector columntypes	Spaltentypen
Hashtable record	Datensatz
NewRecord()	Anlegen eines neuen Datensatzes
set(cname, value)	Setzen des Wertes einer Spalte cname im aktuellen Datensatz;
Insert()	Einfügen des aktuellen Datensatzes;
Sync()	Update/Synchronisation des aktuellen Datensatzes mit der Datenbank;
Find()	Suchen ("exact match") eines Datensatzes; die Suchkriterien werden extra spezifiziert; bei erfolgreicher Suche verweist der aktuelle Datensatz auf den ersten gefundenen Eintrag;
get(cname)	Auslesen der Spalte cname aus dem aktuellen Datensatz;
Delete()	Löschen des aktuellen Datensatzes

Eine Schnittstelle zu einer Tabellenklasse hat folgende Gestalt:

```
interface ITable {
    public void NewRecord(); // neuen Datensatzpuffer anlegen
    public void setString(String cname,String value);
    public void setInt(String cname,int value);
    public String getString(String cname); // liest Spalte Datensatz
    public String getInt(String cname); // setzt Spalte im Datensatz
    public boolean Insert(); // fuegt Datensatz in die Datenbak ein
    public boolean Sync(); // synchronisiert Cache mit Datenbank
    public boolean Find(); // sucht ersten Datensatz
    public boolean Next(); // sucht naechsten Datensatz
}
```

Beispiel: Implementation von Table.Find

```
class Table implements ITable {
    boolean Find() {
        boolean status=false;
        String sqlstmt=new String("select * from "+tablename);
        // record enthaelt die Suchspezifikation in der Form
        // ("Spaltenname",Wert); werden mit AND verknuepft
        if (record!=null) {
            if (record.size(>0) {
                sqlstmt=sqlstmt+"where ";
                nexpr=record.size();
                for (int i=0;i<record.size();i++) {
                    String cname=(String)(columns.elementAt(i));
                    String value=(String)(record.get(cname));
                    sqlstmt=sqlstmt+cname+"='"+value+"'";
                    nexpr--;
                    if (nexpr>0)
                        req=req+" and ";
                }
            }
        }
        // SQL Request absetzen
        sqlrequest=new SQLRequest(sqlstmt,connection);
        // Test, ob ResultSet der Anfrage nicht leer ist
        if (sqlrequest.NotNull()) {
            // Fuellen von record mit dem ersten Datensatz
            try {
                record=new Hashtable();
                for (int i=1;i<=columns.size();i++) {
                    String s=sql.Result().getString((String)(columns.elementAt(i-1)));
                    if (s==null)
                        record.put((String)(columns.elementAt(i-1)),"");
                    else
                        record.put((String)(columns.elementAt(i-1)),s);
                }
                status=true;
            } catch (SQLException ex);
        }
        return status;
    }
}
```

Die Eigenschaften eines Views können ähnlich auf eine Klasse abgebildet werden.

Die Nutzung dieser Basisklassen reduziert den Kodierungsaufwand nicht, wenn sie, wie sie sind, genutzt werden. Sie stellen eine zu allgemeine Implementation dar. Erst ihre Spezialisierungen bringen einen tatsächlichen Vorteil.

Beispiel:

eine Tabelle "Personen"

ID:Int	Name:String
1	Schmeißer
2	Pervushov

führt zur Spezialisierung

```
class Personen_Table extends Table {
    public Person_Table(Connection con) { /* T. Definition*/ }
    public int ID() { return getInt("ID"); }
    public void ID(int value) { setInt("ID",value); }
    public String Name() { return getString("Name"); }
    public void Name(String value) { setString("Name",value); }
}
```

Code, der bisher explizit Spaltennamen als Parameter an die Methoden der Tabellenklasse übergeben mußte, wird dadurch einfacher und robuster. Das nachfolgende Code-Fragment verdeutlicht dies.

```
Table p;
p=new Table(con);
p.addColumn(''ID'',T_INT);
p.addColumn(''Name'',T_STR);
p.NewRecord();
p.setString(''Name'',''Schmeisser'');
if (p.Find())
int id=p.getInt(''ID'');

Personen_Table p;
p=new Personen_Table(con);

p.NewRecord();
p.Name(''Schmeisser'');
if (p.Find())
int id=p.ID();
```

Die Spezialisierung bietet außerdem einen weiteren Vorteil: Sie läßt sich automatisieren. Damit ist es möglich, eine *Fachsprache* zu generieren, die die Definition der Datenbank in einer lesbaren Art und Weise gestattet. Durch einen entsprechenden Compiler kann dieser *Fachsprachentext* z.B. auf Java- Quelltext, wie oben beschrieben, abgebildet werden. Zusätzlich können verschiedenste andere Zieltexte, z.B. OCI-basierter Code oder SQL-Anweisungen, generiert werden.

Innerhalb des hier beschriebenen Projektes wurde eine solche Fachsprache (DBDL - Database Definition Language) entwickelt.

Der folgende Text beschreibt eine Datenbank aus drei Tabellen, zwischen denen *one-to-many-Relationen* bestehen.

```

Database Telefondatenbank {
  Table Personen {
    primarykey ID:Integer;
    unique Name:String;
  }

  Table Anschluss {
    primarykey ID:Integer;
    unique Telefonnummer:String;
  }

  Table Eintrag {
    PersonID:Integer -> Personen.ID;
    AnschlussID:Integer -> Anschluss.ID;
  }

  View PersonNummer {
    Name=Personen.Name;
    Nummer=Anschluss.Telefonnummer;
    Where:"Anschluss.ID in (select AnschlussID from
          Eintrag where PersonID=Personen.ID)"
  }
}

```

Die entworfene Sprache enthält neben Typbezeichnern für Standardtypen und Konstrukten zur Tabellen- und View-Definition auch Elemente zur Beschreibung von *Constraints* (Eigenschaften und Relationen). Für die genaue Beschreibung sei auf [Sch99] verwiesen.

Weiterhin wurde ein Java Applet entwickelt, mit dessen Hilfe eine visuelle Entwicklung einer Datenbank möglich ist. Das Applet liest und generiert ebenfalls Text in DBDL.

Die Arbeitsschritte vom Entwurf der Datenbank bis zur Implementation von Klienten sind in Abbildung 3 dargestellt.

Datenbankzugriff via Request-Broker-Technologie

Prinzipiell greifen Klienten einer Datenbank über SQL-Anweisungen auf selbige zu.

Zur Vereinfachung können Abfragen in SQL-Prozeduren zusammengefaßt werden, die in genau der selben Art und Weise, wie auch normale SQL-Anweisungen aufgerufen werden. Für ein Programm ist dieser Aufruf **nicht transparent**, d.h. der Klient muß sich zuzüglich zum Aufruf einer Prozedur ebenfalls um die Parameterübergabe und die Entgegenname der Ergebnisse kümmern.

Günstiger ist in diesem Fall die Abbildung des Modells auf eine Interface-definierte Client-Server-Architektur. Diese Technologie ist relativ neu und beinhaltet im wesentlichen folgende Schritte:

1. Festlegung der Schnittstelle zwischen Klient und Server
2. Auswahl einer geeigneten Fachsprache zur Beschreibung der Schnittstelle[§]

[§]Diese Sprache wird gewöhnlich als *IDL - Interface Definition Language* bezeichnet.

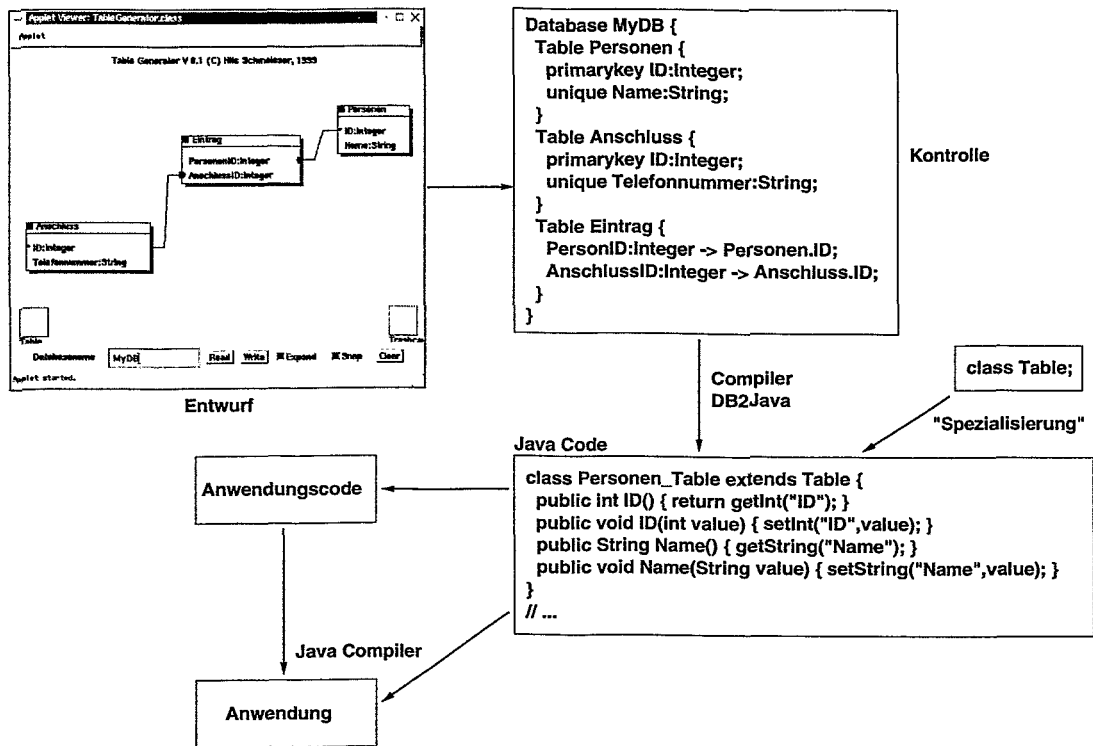


Abbildung 3: Arbeitsschritte und Werkzeuge zum Datenbankdesign

3. Auswahl einer geeigneten Kommunikationsbibliothek
4. Auswahl geeigneter Sprachen zur Definition der Klienten und Server sowie Abbildung der Schnittstellendefinition auf diese Sprachen
5. (automatische) Generierung von *Stub* und *Skeleton*
6. Implementation von Klient bzw. Server ausgehend von Stub bzw. Skeleton

Als *Stub* bzw. *Skeleton* werden die Komponenten bezeichnet, die die Verbindung zum *Request Broker* auf der Seite des Klienten bzw. des Servers herstellen.

Die Übergabe von Parametern an eine Prozedur/Methode durch den Klienten und die Weiterleitung des Aufrufes an den Server bzw. die Entgegennahme von Argument durch die Implementation der Prozedur/Methode und die Rückgabe von Ergebnissen vollzieht sich nunmehr vollkommen **transparent** für beide Seiten.

Diese Technologie wird *Request-Broker-Technologie* genannt und findet zur Zeit sowohl in *CORBA* (*Common Object Request Broker Architecture*, [Cor], [Red96], [Say97]) als auch im *DCOM* (*Distributed Component Object Model* ([Box98]) ihre Verwirklichung. Die prinzipielle Funktionsweise ist in Abbildung 4 dargestellt.

Eine Implementation dieser Technologie unter Nutzung des HTTP-Protokolls als Kommunikationsschicht bietet der *Oracle Application Server (OAS)* (Abbildung 5).

Das Verfahren zur Nutzung des OAS sieht wie folgt aus:

1. Der Klient ruft eine Prozedur des Interfaces auf, die implizit einen Aufruf des zugehörigen Stubs generiert.

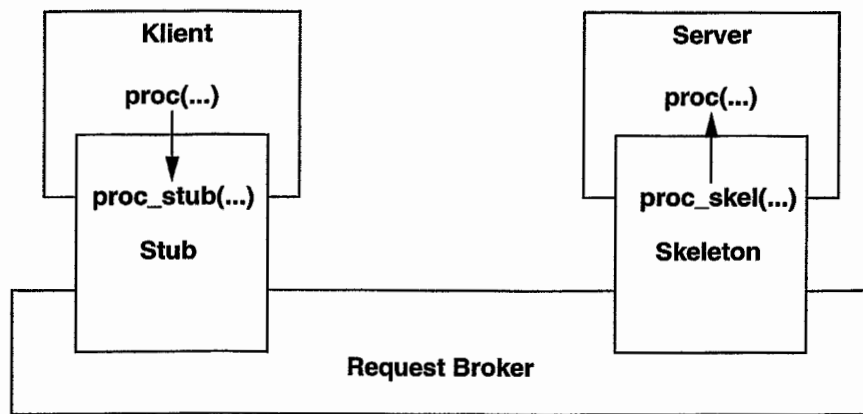


Abbildung 4: Schema einer Request Broker Architektur



Abbildung 5: Aufbau des Oracle Application Servers 4.0

2. Der Stub der Prozedur schickt eine Anfrage in Form eines URI mit Abfragetext (query string)


```
http://oas_host:port/ database.procedure[?parameter_name=wert]
```

 an den OAS.
3. Der Application-Server empfängt den URL-Request, ruft das Skeleton der zugehörige PL/SQL-Prozedur auf und übergibt dieser die im Abfragetext enthaltenen Parameter.
4. Das Skeleton ruft die Implementation der Prozedur und generiert aus den Ergebnissen des Aufrufes eine HTML-strukturierte Ausgabe, die vom OAS als Antwort auf den URL-Request an den Klienten-Stub zurückgegeben wird.
5. Im Stub erfolgt eine Rückwandlung des empfangenen HTML-Textes in Werte für die aktuellen Variablenparameter des Prozeduraufrufes auf der Klientenseite.

Für die in diesem Beitrag vorgestellte Anwendung braucht der Request Broker nur mit einem geringen Funktionsumfang ausgestattet zu sein. Da der OAS seinerseits bereits alle nötigen Request-Broker-Komponenten enthält, ist der Request Broker durch das Hinzufügen einer Basisklasse, die den Datenaustausch via HTTP organisiert, auf der Klientenseite bereits vollständig implementiert.

```

class ORAWEBRequest {
public:
    ORAWEBRequest() { }
    ORAWEBRequest(char *oashost,unsigned int port,char *base);
    int Call(char *method,unsigned int argc,char* argv[],int _out[]);
};

```

Ein Stub wird folgendermaßen generiert:

```

class UserDB {
public:
    // ...
    virtual int GetCID(char* Name,int &CID) {
        char argv[2][256];
        int _out[2];
        sprintf(argv[0],"Name=%s",name); _out[0]=0;
        _out[1]=0;
        ORAWEBRequest::Call("UserDB.GetCID",2,argv,_out);
        sscanf(argv[1],"CID=%d",&CID);
    }
    // ...
};

```

Da der Stub in diesem Fall keine weitere Funktionalität beinhaltet[¶], ist er gleichbedeutend mit dem Interface auf der Klientenseite.

Die Implementation von `ORAWEBRequest::Call` erzeugt aus den übergebenen Parametern einen URL-Request

```
http://oas_host:port/base/UserDB_Skeleton.GetCID?Name=name&CID=0
```

Der OAS ruft daraufhin die Prozedur `GetCID` aus dem Paket `UserDB_Skeleton` und die ihrerseits `UserDB.GetCID` auf.

```

package body UserDB_Skeleton is
    --- ...
    procedure GetCID(Name in varchar2) is
        CID number;
    begin
        UserDB.GetCID(ClassName,CID);
        Htp.Prn('CID=');
        Htp.Prn(CID);
        Htp.nl;
    end GetCID;
    --- ...
end UserDB_Skeleton;

```

[¶]Microsoft verwendet im Component Object Model z.B. ein Proxy Konzept zur lokalen Pufferung von Objektzuständen.

```

package body UserDB is
    --- ...
    procedure GetCID(Name in varchar2,CID out number) is
    begin
        --- Implementation von GetCID
    end GetCID;
    --- ...
end UserDB;

```

Ein Klient kann nun eine neue Instanz der Klasse UserDB erzeugen (und damit eine Verbindung zum Server-Objekt aufbauen) und die Methode GetCID rufen.

```

// ...
UserDB *userdb=new UserDB("selene.fz-rossendorf.de",7070,"UDBC");
// ...
int cid;
char hostname[1025];
gethostname(hostname,1024);
userdb->GetCID(hostname,cid);
// ...

```

Die Schnittstelle zwischen Klient und Server wird in einer speziellen IDL beschrieben, für die es einen Compiler gibt, der sowohl den Stub- (C++) als auch den Skeleton-Code und den Rahmen für die Implementation (PL/SQL) generiert.

Beispiel :

```

interface UserDB:ORAWEBRequest {
    UserDB(String host in,Integer port in,String base in)
        :ORAWEBRequest(host,port,base);
    // ...
    GetCID(String Name in,Integer CID out);
    // ...
}

```

Die vollständige Definition der verwendeten Interface Definition Language findet sich in [Sch99].

Literatur

- [ANS89] ANSI. Information Systems - Database Language - SQL (includes ansi x3.168-1989). Technical report, ANSI X3.135-1992 (R1998), 1989.
- [Box98] D. Box. *Essential COM*. Addison-Wesley, 1998.
- [Cor] CORBA and OMG information resources. <http://www.omg.org/>
- [ISO92] ISO. Information Technology - Database Languages - SQL. Technical report, ISO/IEC 9075:1992, 1992.
- [Red96] J.-P. Redlich. CORBA 2.0 - Esperanto für objekte? *unix/mail*, 5:319, 1996.
- [Say97] A. Sayegh. *CORBA - Standard, Spezifikation, Entwicklung*. O'Reilly, 1997.
- [Sch99] Nils Schmeißer. Zentrale Nutzerdatenbank. *nicht veröffentlicht*, 7 1999.

Dosisberechnungen für Beamdumps des ELBE-Beschleunigers

E. Müller¹, H. Guratzsch², B. Naumann³

1 Einführung

Die Strahlungsquelle ELBE ist für eine Elektronenenergie von 40 MeV und einen Strahlstrom von 1 mA konzipiert. Die Beschleunigung erfolgt mit supraleitenden 1,3-GHz-Resonatoren in zwei Stufen zu je 20 MeV. Zunächst wird der Beschleuniger für 20 MeV mit Injektor, Strahlführung, Diagnoseeinheiten und Beamdump aufgebaut. Nach Installation der zweiten Beschleunigerstufe muß der Beamdump für mindestens 40 MeV ausgelegt werden. In diesem Fall, wird dem Anliegen der konservativen Herangehensweise im Strahlenschutz folgend, für die Berechnungen zum Beamdump ein um 10 MeV höherer Wert verwendet. Die im folgenden beschriebenen Rechnungen wurden daher bei Elektronenenergien von 20 MeV und 50 MeV durchgeführt.

Aufgabe der Berechnungen ist es, richtungsabhängige Werte der Dosis pro eingeschossenes Elektron als „Quellterme“ für weitere Abschirmungsberechnungen zu erzeugen. Die Beamdumps müssen so abgeschirmt werden, daß die gesetzlich [6] zulässigen Werte der Dosisleistung in personenzugänglichen Bereichen eingehalten werden. Formal läßt sich mit Monte-Carlo-Programmen auch die Dosis pro Elektron hinter den Abschirmungen berechnen, jedoch ist dann die benötigte Rechenzeit sehr lang. Für ELBE werden die Abschirmungsberechnungen mit den energieabhängigen Transmissionswerten der NCRP-51 [8] ausgeführt. Die Verbindung zwischen Dosisleistung (Dosis/h) und Dosis pro Elektron wird über folgende Beziehung hergestellt:

$$\frac{1}{e} = 2.25 \cdot 10^{19} \frac{1}{mA \cdot h} . \quad (1)$$

Das Material für den Strahlabsorber muß zur Wärmeabführung an das Kühlwasser ein guter Wärmeleiter sein. Die Neutronenproduktions-Rate soll aus Strahlenschutzgründen gering sein, um unnötige Aktivierungen zu vermeiden. Mögliche Materialien sind Graphit, Aluminium und Kupfer. Keines der Materialien erscheint aber auf den ersten Blick optimal zu sein. Graphit hat zwar auf Grund der niedrigen Ordnungszahl eine geringe Neutronenausbeute und verfügt auch über gute Wärmeeigenschaften, birgt aber die Gefahr der Bildung von Graphitniederschlägen auf der gekühlten Oberfläche der supraleitenden Kavität infolge von Kathodenzerstäubung am Beamdump in sich. Niederschläge auf der Kavität gefährden die Supraleitung. Aluminium zeigt hohe Löslichkeit im deionisierten Kühlwasser und wird bereits bei ca. 600° C weich. Die Relevanz der Bildung von ²⁶Al (Halbwertszeit: $T_{1/2} = 7 \cdot 10^5 a$) aus Aluminium wird in [5] abgeschätzt. Kupfer hat sehr gute konstruktive Eigenschaften, aber die Neutronenausbeute ist infolge der hohen Ordnungszahl ziemlich groß. Als langlebiges Nuklid wird ⁶⁰Co ($T_{1/2} = 5,3 a$) produziert. Die endgültige Auswahl des Beamdump-Materials ist noch nicht abgeschlossen. Deshalb werden - auch zur Unterstützung dieser Auswahl - Berechnungen zum Dosisverhalten am Beamdump für alle drei Materialien ausgeführt.

¹FZR, Abt. Kommunikation und Datenverarbeitung

²FZR, Zentralabteilung Neue Beschleuniger

³FZR, Institut für Kern- und Hadronenphysik

Für die Dosisberechnungen können die in Rossendorf bereits implementierten Monte-Carlo-Programme EGS4 [1], MCNP [2], GEANT [3] und FLUKA [4] eingesetzt werden. Die aktuellen Versionen dieser Programme behandeln Transportprozesse mit Teilchen im Energiebereich von einigen keV bis zu mehreren TeV. Sie sind somit sowohl für die Niederenergie- als auch für die Hochenergiephysik einsetzbar. Anwendung finden sie weiterhin in der Medizin, z. B. für die Strahlungsplanung in der Strahlentherapie. Die Programme berechnen physikalische Prozesse mit Elektronen, Positronen, Photonen - und bei MCNP und FLUKA auch mit Neutronen - entsprechend der Wechselwirkungswahrscheinlichkeiten in den Materialien. Eine Berechnung von Photoneutronenquellen mit der gegenwärtig aktuellen MCNP-Version ist allerdings nur in zwei Stufen möglich, was eine gewisse Vertrautheit mit MCNP voraussetzt [10].

In dieser Arbeit werden erste Ergebnisse vorgestellt, die mit EGS4, MCNP und GEANT gewonnen wurden.

2 Bemerkungen zu den benutzten Rechenprogrammen EGS4, MCNP und GEANT

EGS4 wurde von SLAC (Stanford Linear Accelerator Center) entwickelt. EGS ist die Abkürzung für Electron-Gamma-Shower. Für Dosisberechnungen in Zylindergeometrien kann das spezielle EGS4-Anwenderprogramm DOSRZ benutzt werden. Mit DOSRZ wurde die gemittelte Energiedosis in $Gy \cdot cm^2$ je Elektron ($1 Gray[Gy] = 1 J/kg$) in ausgewählten Regionen an den unter 3.1 und 3.2 beschriebenen Graphit- und Kupfer-Beamtdumps für parallelen Strahl in z-Richtung (auf den folgenden Abbildungen von links nach rechts) berechnet. Die Dosis in Gy je Elektron ergibt sich durch Division mit der Strahlfläche. In den untersuchten Beamtdumps sind in Strahlrichtung Kegel und Kegelmünder ausgespart (s. Abbildung 1 und Abbildung 2). Diese Aussparungen wurden im Geometriebeschreibungsteil der DOSRZ-Eingabedatei durch Zylinderringe mit abnehmenden Innenradien angenähert.

Um in einer speziellen Region außerhalb des Beamtdumps die Äquivalentdosis zu bestimmen, wurde jeweils in Strahlrichtung eine 10 cm breite Wasser- bzw. Gewebeschicht (anstelle von Luft) als Probekörper in die Eingabedatei eingefügt. Im Bereich dieser Schichten sind die Energiedosiswerte und die Äquivalentdosiswerte gleich, da für Elektronen und Photonen der Qualitätsfaktor Q den Wert 1 hat [7]. Q berücksichtigt das unterschiedliche Strahlenrisiko der verschiedenen Strahlenarten. Bei Verwendung von Gewebe statt Wasserschichten unterscheiden sich die Dosiswerte nur unwesentlich. Die in Abbildung 1 und Abbildung 3 in runden Klammern angegebenen Äquivalentdosiswerte wurden mit Wasserschichten ermittelt.

Zur Kontrolle, ob EGS4 und das spezielle Anwenderprogramm DOSRZ auf dem Zentralen Server rike (IBM RISC RS/6000 Mod. 591) des FZR richtig installiert sind, wurden Vergleichsrechnungen mit dem DOSRZ-Programm auf einem PentiumII-PC an der Universität von Gent durchgeführt. Dabei zeigte sich eine Übereinstimmung in den ermittelten Dosiswerten. Auch mit dem EGS4-Anwenderprogramm XYZDOS wurden auf dem Server rike für vergleichbare quaderförmige Anordnungen und quadratischem Elektronenstrahlquerschnitt Dosiswerte berechnet, die nur geringfügig ($\leq 10\%$) von den mit DOSRZ ermittelten Werten abweichen.

MCNP (Monte Carlo N-Particle Transport Code) wurde im Los Alamos National Laboratory entwickelt. Im FZR ist auf dem Zentralen Server pales (HP V-Class-Rechner) die MCNP-Version 4B installiert. Die interessierenden mittleren Äquivalentdosiswerte (in Sievert je Elektron) können bei der Nutzung des MCNP-Programms u.a. mit Hilfe von ICRP-21-Konvertierungsfaktoren ([2], [9]) bestimmt werden. Das Einfügen zusätzlicher Wasserschichten in die Beschreibung der Geometrie ist daher nicht erforderlich. Ferner können beim Einsatz von MCNP und GEANT (s.u.) die kegelförmigen Aussparungen auf einfache Weise exakt in den Eingabedateien erfaßt werden. MCNP wurde für alle Beamtdumps eingesetzt, und die berechneten Dosiswerte sind in die Abbildungen 1, 3 und 5 eingefügt worden.

GEANT (Generation of Events and Tracks), im CERN entwickelt, wurde nur für Dosisberechnungen um den Graphit-Beamtdump eingesetzt. Die mit GEANT berechneten Äquivalentdosiswerte sind in Abbildung 1 mit aufgenommen worden (Werte in eckigen Klammern). Für die beiden Abschnitte am rechten Bildrand ($160 \text{ cm} \leq z \leq 170 \text{ cm}$) reichte die verwendete Teilchenanzahl noch nicht aus, um statistisch gesicherte Werte zu ermitteln. Deshalb wurden sie in Abbildung 1 nicht übernommen.

Die Dosiswerte vor und in unmittelbarer Nähe des Graphitzylinders stimmen recht gut mit den Ergebnissen von EGS4 und MCNP überein.

Alle Berechnungen mit EGS4, MCNP und GEANT wurden mit den globalen Cut-off-Werten (untere Energieschwellen für den Teilchentransport) von 0.7 MeV für Elektronen und 0.01 MeV für Photonen ausgeführt. Niedrigere Cut-off-Werte bewirken eine wesentliche Verlängerung der Rechenzeit ohne bemerkenswerte Veränderung der ermittelten Dosiswerte. Für 5 Millionen Ereignisse waren beim Graphitdump bei der EGS4-Nutzung an der rike ca. 16 Stunden CPU-Zeit erforderlich, beim MCNP-Einsatz an der pales 6,5 Stunden. Dabei war der ausgewiesene relative Fehler der ermittelten Dosiswerte bei EGS4 ca. doppelt so hoch als bei MCNP, aber bei beiden Codes kleiner als 1%. Mit GEANT wurden am Zentralen Server hera (IBM-S70) für 40 Millionen Ereignisse ca. 25 Stunden Rechenzeit benötigt. Die angegebenen, beträchtlich voneinander abweichenden Rechenzeiten sind kein Maß für die jeweilige Leistungsfähigkeit der drei Programme, da EGS4, MCNP und GEANT auf verschiedenen Zentralen Servern des FZR installiert sind, deren Rechenleistung unterschiedlich ist.

3 Ermittelte Äquivalentdosiswerte für verschiedene Beamdumps

3.1 Graphit-Beamdump

In Abb. 1 ist der zylindrische Graphit-Beamdump (Dichte: $\rho=1.7 \text{ g/cm}^3$) mit einem ausgesparten Kegel dargestellt (blaues Gebilde). Es wird angenommen, daß auf diesen Beamdump zentral in Achsrichtung ein paralleler Elektronenstrahl (50 MeV) trifft, dessen Quelle sich 2 m links vor dem Beamdump befindet. Abbildung 1 enthält auch die mit MCNP, EGS4 und GEANT berechneten mittleren Äquivalentdosiswerte.

In der Abbildung sind oben die Werte für $15 \text{ cm} \leq r \leq 50 \text{ cm}$ und

$$-200 \text{ cm} \leq z \leq -190 \text{ cm}$$

$$25 \text{ cm} \leq z \leq 35 \text{ cm}$$

$$60 \text{ cm} \leq z \leq 70 \text{ cm}$$

$$160 \text{ cm} \leq z \leq 170 \text{ cm}$$

eingetragen,

links, über dem Strahlrohr die Werte für $10.5 \text{ cm} \leq r \leq 15 \text{ cm}$ und

$$-200 \text{ cm} \leq z \leq -190 \text{ cm} \quad \text{und}$$

rechts vom Beamdump die Werte für $0.0 \text{ cm} \leq r \leq 0.1784 \text{ cm}$ und

$$60 \text{ cm} \leq z \leq 70 \text{ cm}$$

$$160 \text{ cm} \leq z \leq 170 \text{ cm}.$$

Graphit-Beamdump

grau: Eisen blau: Graphit
gelb: Luft

Beamdumpplaenge: 60.0 cm

Beamdumpradius: 15.0 cm

Strahlflaeche: 0.1 cm**2

Strahlradius : 0.1784 cm

Elektronenenergie: 50 MeV

Dosiswerte in Sv/electron

MCNP: 5 Mio. Ereignisse

DOSRZ: 3.5 Mio. Ereignisse
(Werte in runden Klammern)

GEANT: 50 Mio. Ereignisse
(Werte in eckigen Klammern)

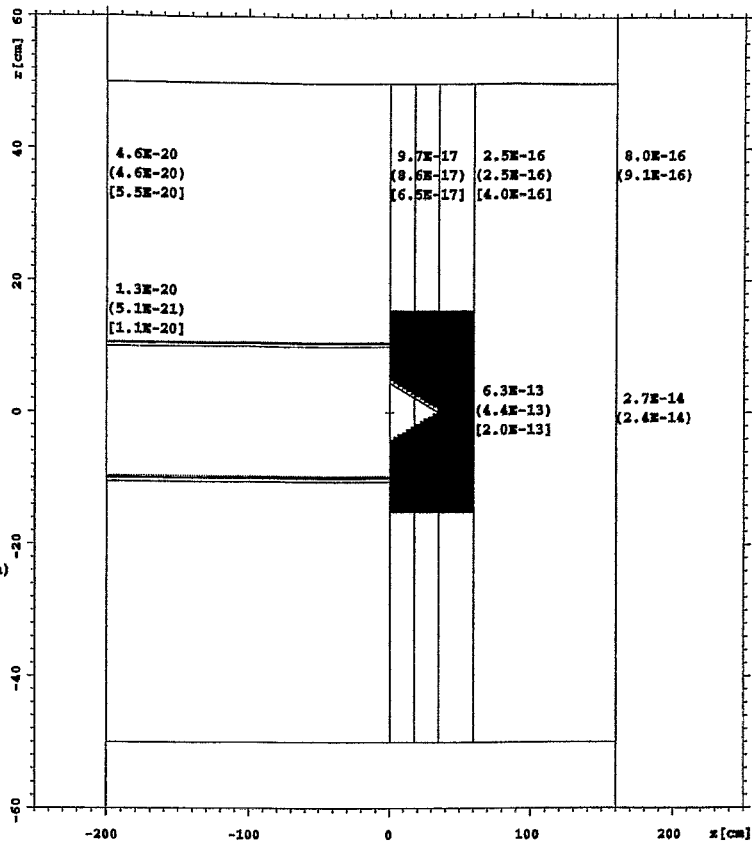


Abb. 1

3.2 Kupfer-Beamdump

Abbildung 2 veranschaulicht die Form des zylindrischen Kupfer-Beamdumps (Dichte: $\rho=8.933 \text{ g/cm}^3$) mit ausgespartem Kegelstumpf und Kegel. Das grün dargestellte Kupfergebilde (Radius: 6.65 cm) ist mit einem 0.7 cm dicken Wassermantel und einem 0.5 cm dicken Eisenmantel umhüllt. Es wird davon ausgegangen, daß auf den Beamdump zentral in z-Richtung ein paralleler Elektronenstrahl (20 MeV) mit einem Strahlradius von 4.9 cm trifft, dessen Quelle sich 2 m links vor dem Beamdump befindet.

Cu-Beamdumpe

grau: Eisen gruen: Kupfer
gelb: Luft blau: Wasser

Beamdumplaenge: 34.15 cm

Beamdumradius: 7.85 cm

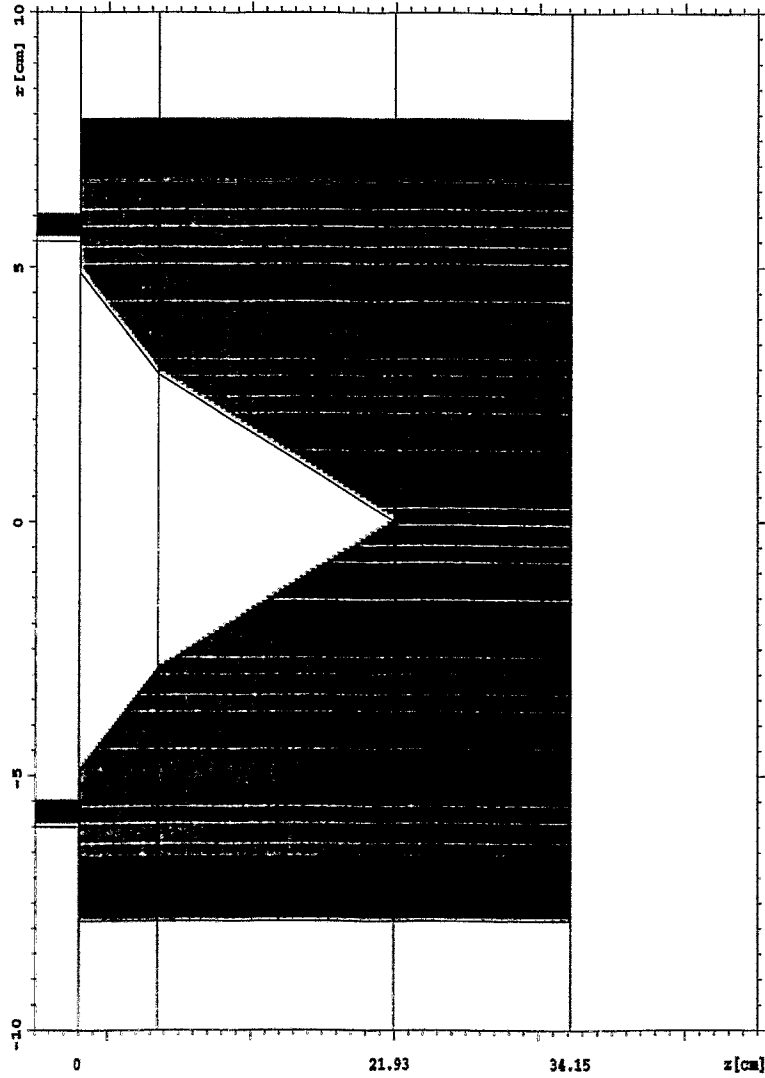


Abb. 2

Abbildung 3 enthält die für den Kupfer-Beamdump mit MCNP und dem EGS4-Anwenderprogramm DOSRZ berechneten mittleren Äquivalentdosiswerte. Die DOSRZ-Werte sind in runden Klammern eingeschlossen. Sie wurden nur für die z-Bereiche

$$\begin{aligned} -200 \text{ cm} &\leq z \leq -190 \text{ cm} \\ 129.15 \text{ cm} &\leq z \leq 139.15 \text{ cm} \\ 229.15 \text{ cm} &\leq z \leq 239.15 \text{ cm} \end{aligned}$$

ermittelt.

Mit MCNP wurden für die z-Werte -200, -100, 0, 34.15, 134.15 und 234.15 Äquivalentdosen berechnet. Die Dosiswerte bei $z = 21.93$ (Kegelspitze) sind unter dem Beamdump eingetragen.

Für die genannten z-Bereiche bzw. -Werte sind in r-Richtung für die Abschnitte

$$\begin{aligned} 0 \text{ cm} &\leq r \leq 4.9 \text{ cm} \\ 45 \text{ cm} &\leq r \leq 50 \text{ cm} \\ 90 \text{ cm} &\leq r \leq 100 \text{ cm} \\ 180 \text{ cm} &\leq r \leq 200 \text{ cm} \end{aligned}$$

die Dosiswerte in Abbildung 3 eingefügt worden.

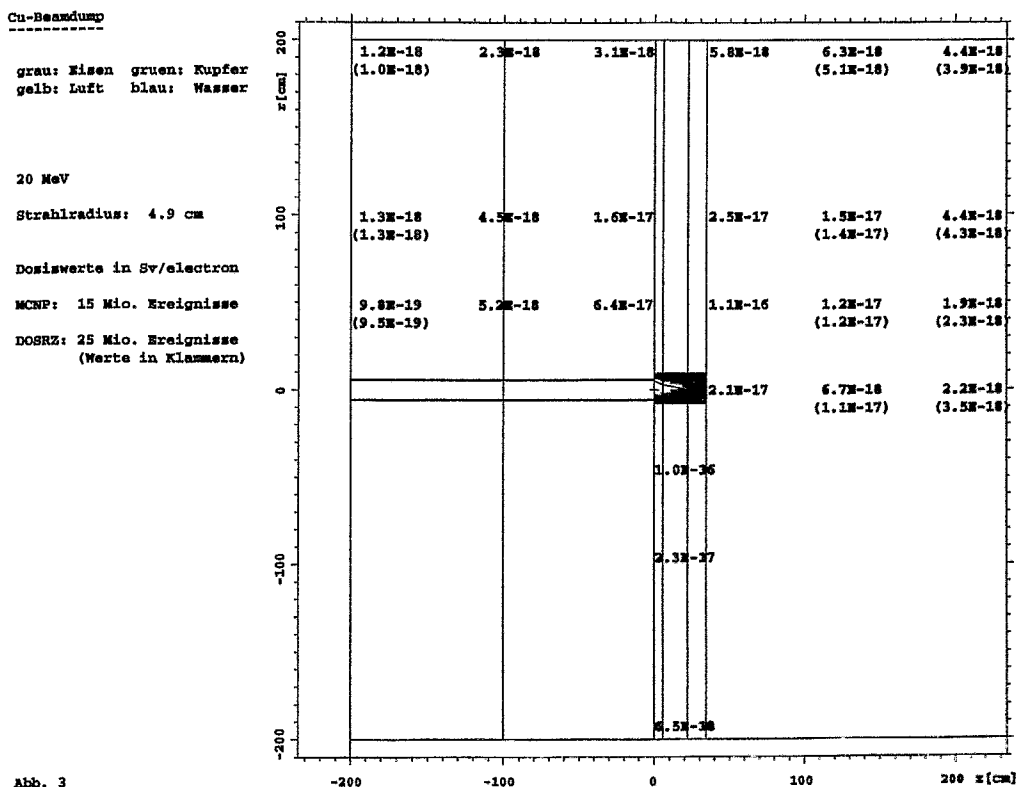


Abb. 3

3.3 Aluminium-Beamdump

In Abbildung 4 ist der untersuchte zylindrische Aluminium-Beamdump (Dichte: $\rho=2.702 \text{ g/cm}^3$) dargestellt. Man erkennt von links nach rechts einen Kegelstumpf, einen schmalen Zylinder und einen Kegel, die aus dem Aluminiumkörper (pinkfarben) ausgespart wurden. Zwecks Kühlung sind in den Körper Kühlrippen (in blau) eingearbeitet, die mit Wasser durchströmt werden sollen. In den durchgeführten Berechnungen wurde davon ausgegangen, daß auf den Beamdump zentral in z-Richtung ein paralleler Elektronenstrahl (20 MeV) mit einem Strahlradius von 5.1 cm trifft, dessen Quelle sich 2 m links vor dem Beamdump befindet.

Al-Beamdump

grau: Eisen pink: Aluminium
gelb: Luft blau: Wasser

Beamdumplänge: 41.0 cm

Beamdumpradius: 13.8 cm

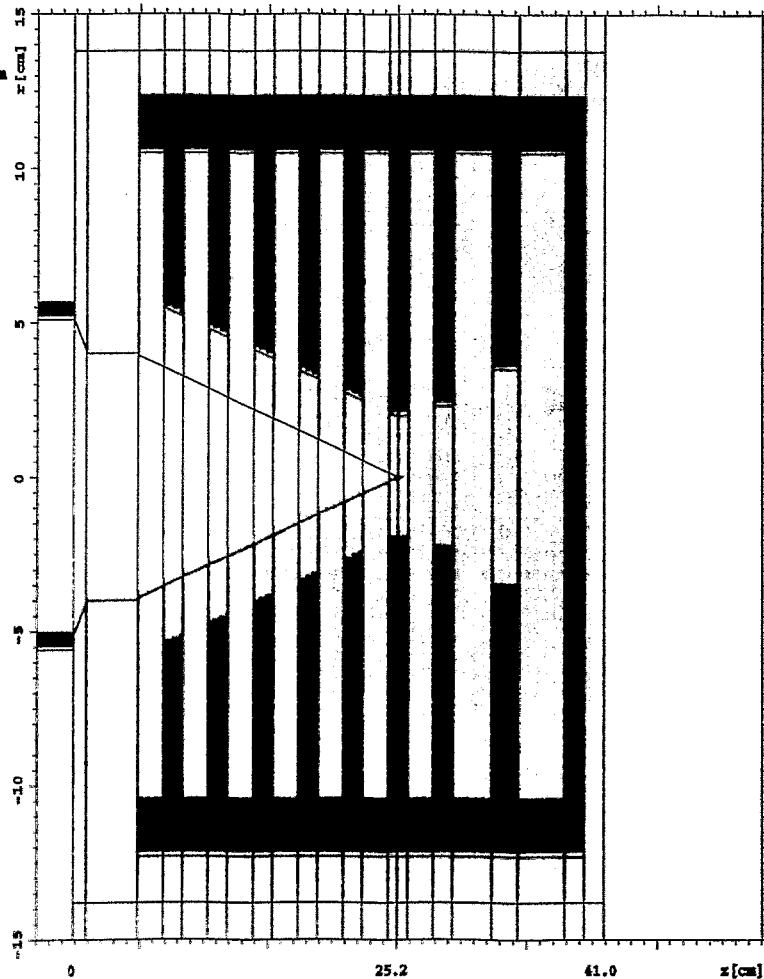


Abb. 4

Abb. 5 enthält die mit MCNP für den Aluminium-Beamdump berechneten mittleren Äquivalentdosiswerte.

Sie wurden für die z-Werte -200, -100, 0, 25.2, 41.0, 141.0 und 241.0

und in r-Richtung für die Abschnitte

12.3 cm \leq r \leq 13.8 cm (außer für z = 25.2)

45 cm \leq r \leq 50 cm

90 cm \leq r \leq 100 cm

180 cm \leq r \leq 200 cm

ermittelt und in die Abbildung eingefügt. Die Dosiswerte bei z=25.2 sind in den unteren Teil des Bildes eingetragen worden (in eckigen Klammern).

Al-Beamdump

grau: Eisen pink: Aluminium
gelb: Luft blau: Wasser

20 MeV

Strahlradius: 5.1cm

Dosiswerte in Sv/electron;

Werte bei z=25.2 (Kegelspitze)
in eckigen Klammern

MCNP: 5 Mio. Ereignisse

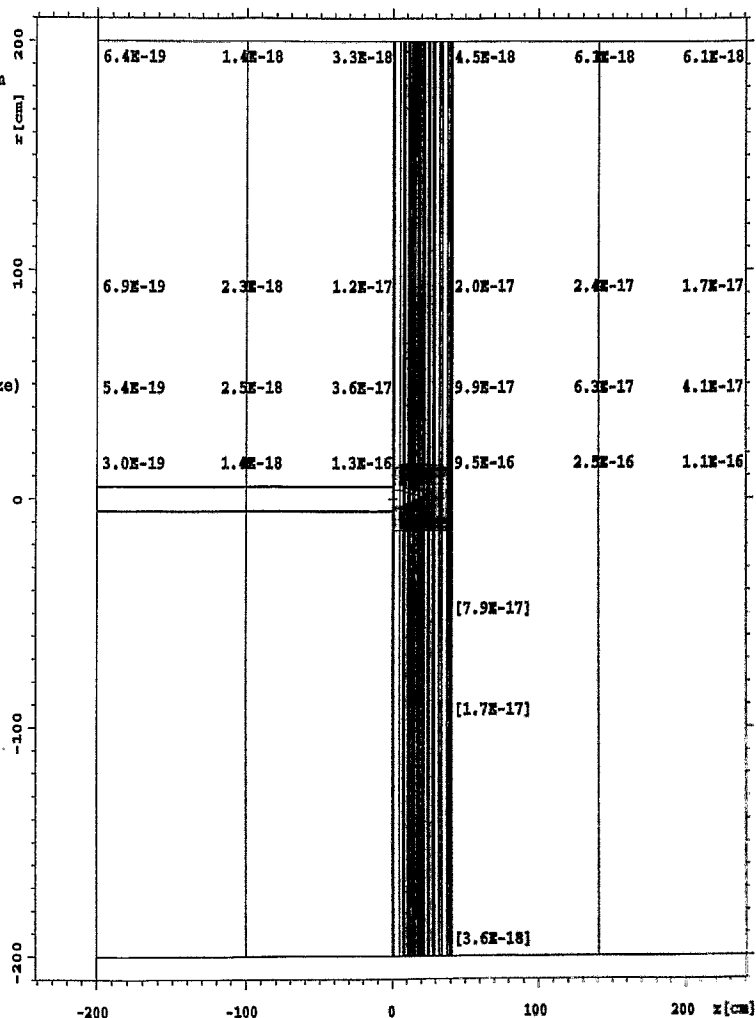


Abb.5

4 Resümee

Für den elektromagnetischen Schauer stimmen die mit den Programmen EGS4, MCNP und GEANT berechneten Dosiswerte gut überein (Abbildung 1 und Abbildung 3). Bei den hier beschriebenen Anwendungen hat sich MCNP als das nutzerfreundlichste Programm erwiesen.

Inzwischen wurde festgelegt, für die erste Ausbaustufe (20 MeV) des ELBE-Beschleunigers die Strahltests mit einem Kupfer-Beamdump durchzuführen. Aus Strahlenschutzgründen muß der Beamdump lokal abgeschirmt werden. Grundlage für die Berechnung der Abschirmungsdicke sind die in Abbildung 3 angegebenen Äquivalentdosiswerte. Da während der Bestrahlung der Beamdump aktiviert wird, muß die örtliche Abschirmung auch den Strahlenschutz für den Personenzutritt nach Bestrahlungsende gewährleisten. Detaillierte Berechnungen zur Aktivierung verschiedener Beamdump-Materialien bei 50 MeV und einem Strahlstrom von 1 mA wurden bereits durchgeführt und in [5] dokumentiert. Die meßtechnische Erfassung der Aktivitätsentwicklung wird dann in den Testexperimenten erfolgen.

Literatur

- [1] W. R. Nelson, H. Hirayama, D. W. O. Rogers:
THE EGS4 CODE SYSTEM, SLAC-Report-256, Dec. 1985;
s.auch: <http://ehssun.lbl.gov/egs/egs.html>.
- [2] *MCNP - A General Monte Carlo N-Particle Transport Code*,
LA-12625-M, Version 4B, Manual, March 1997;
s.auch: <http://www-xdiv.lanl.gov/CXI/PROJECTS/MCNP>
- [3] *GEANT - Detector Description and Simulation Tool*,
CERN Program Library, W 5013, Edition March 1994
- [4] A. Fasso et al.: *FLUKA 98 User's Guide*, Manual, 1998
and
A. Fasso et al.: *Total Giant Resonance Photonuclear Cross-Section for light Nuclei: A Database for the FLUKA Monte Carlo Transport Code*,
Proceedings of the III Specialists Meeting on shielding aspects of accelerators, targets and irradiation facilities, Sendaj, Japan, 12-13 May 1997, p. 61-74
- [5] B. Naumann, W. Neubert und D. Pröhl:
Entwicklung von Strahlfängern für den Elektronenbeschleuniger ELBE,
FZR-Report, 1999 (in Druck)
- [6] *Verordnung über den Schutz vor Schäden durch ionisierende Strahlen (Strahlenschutzverordnung - StrlSchV)*, Gesetzblatt I, S. 943, 1989
- [7] *Atomgesetz mit Verordnungen*, 21. Auflage, Nomos Verlagsgesellschaft, Baden-Baden, S. 181, 1998
- [8] NCRP Report No. 51, *Radiation protection design guidelines for 0.1 - 100 MeV particle acceleration facilities*, Washington, 1977
- [9] *Data for Protection Against Ionizing Radiation from External Sources: Supplement to ICRP Publication 15*, ICRP-21, International Commission on Radiological Protection, Pergamon Press, April 1971
- [10] H. Kumpf: *Vorgehen zur Berechnung einer Photoneutronenquelle mit MCNP*, interner Bericht, FZR-FWST, 1998

Neue Hilfsmittel für die Herstellung von Videos

Horst Böttger

1 Vorbemerkung

Bereits im Jahresbericht 1995/96 waren neue Werkzeuge für die Herstellung von Videos vorgestellt worden. Es handelte sich dabei zum einen um Werkzeuge zur Erstellung von Video-Vorspann und -Abspann, zum anderen um neue AVS-Module für die Animation mittels Kamerabewegungen. Neben dem Nutzerinterface (Modul `rotate_camera`) und den Modulen für Operationen mit skalaren Daten zur Bahnsteuerung gehörten hierzu auch Module, mit denen die Dynamik der Kamerabewegungen gesteuert werden kann. Wir bezeichnen derartige Operationen als Harmonisierung und verstehen darunter Aktionen wie Start, Stop, Beschleunigung, Bremsung und Geschwindigkeitstrends.

Die Arbeiten wurden 1997 weitergeführt. Die Ergebnisse sind im folgenden dargestellt.

2 Betrachtung zur Kinematik

Die Animation durch Kamerabewegungen wird realisiert durch die zeitliche Veränderung mindestens eines Kameraparameters, so daß eine Bewegung der Szenerie impliziert wird. Die erforderlichen Werte der Zeitvariablen t erhält man mittels des AVS-Animators durch den Modul `animatead_float`.

Dieser Modul generiert nach seiner Aktivierung eine Folge diskreter t -Werte entsprechend den vorzugebenden Parametern a, b, n :

$$t(i) = a + i * dt, \quad dt = (b - a)/n, \quad i = 0, 1, \dots, n, \quad n > 0 \quad (\text{Standardwerte: } a=0, b=1, n=10).$$

Leitet man diese t -Werte innerhalb des betreffenden AVS-Netzwerkes den zu verändernden Kameraparametern zu, so entsteht eine Sequenz aus $n+1$ Bildern (Animationssequenz). Diese stellen in der Regel einen einzelnen Flugbahnabschnitt dar. Die Übergänge vom Ruhezustand (z.B. einer Stillbildfolge des Anfangszustandes) zur Bewegungsphase und umgekehrt (zur nachfolgenden Sequenz) sind sprunghaft. Dies führt bei der Betrachtung der kompletten Videosequenz zu unnatürlichen Bewegungseindrücken, die nicht akzeptiert werden können. Vielmehr muß eine der Realität entsprechende Glattheit gefordert werden, die Beschleunigungen und Bremsungen enthält.

Mathematisch betrachtet, ist t als Funktion der Echtzeit an den Intervallgrenzen nicht stetig differenzierbar. Bei der simplen Kamerarotation betrifft dies den Drehwinkel $\phi(t)=c*t+d$, beim Kameraflug die Ortsfunktion

$$s(t) = x(t), y(t), z(t)$$

die i.a. für die Kameraposition und den Kamerablickpunkt definiert werden muß.

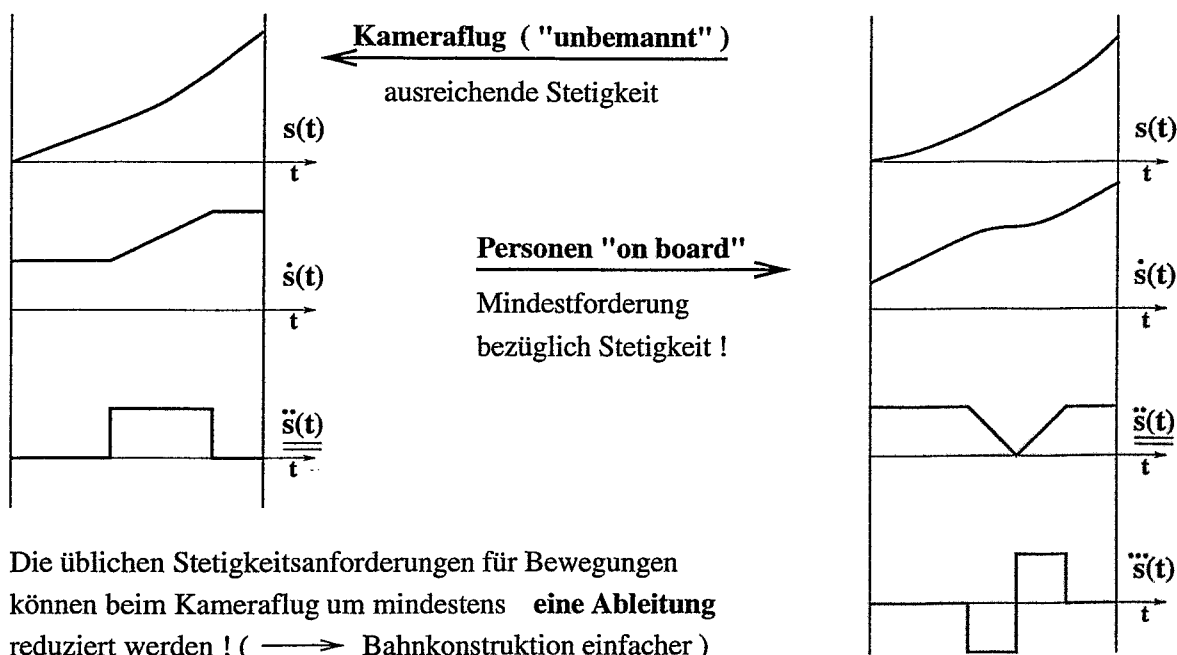
Wie schon im Vorbericht erwähnt, besteht eine komplette Kameraflugbahn in der Regel aus mehreren Abschnitten, an deren Verbindungsstellen ebenfalls gewisse Stetigkeitsbedingungen einzuhalten sind.

Es ergibt sich die Frage, wie hoch diese Forderungen nach Stetigkeit sein müssen. Zur Beantwortung stellen wir die folgende Überlegung an:

In der realen Welt müssen überall dort, wo Bewegungen stattfinden, bestimmte Mindestforderungen bezüglich der auftretenden Längs- und Querbewegungen gestellt werden. Diese als Kraft spürbaren Größen sollen sich nicht sprunghaft ändern, also stetig sein. Da sie mit der zweiten Ableitung der Ortsfunktion $s(t)$ zusammenhängen, erfordert dies eine mindestens zweimal stetig differenzierbare Funktion $s(t)$. Diese Mindestforderung hat Konsequenzen z.B. auf die Ausformung von Kurven für Verkehrswege, auf die Steuerung von Anfahr- und Bremsvorgängen (horizontal und vertikal) u.a.m.

In der hier betrachteten virtuellen Welt nehmen wir die Bewegung nicht physikalisch, sondern nur visuell wahr. Unser Auge ist nicht in der Lage, eine der zweiten Ableitung von $s(t)$ entsprechende Größe zu erkennen. Das bedeutet, es braucht nur eine einmal stetig differenzierbare Ortsfunktion $s(t)$ gefordert zu werden, also nur $s(t)$ und $v(t) = ds(t)/dt$ müssen stetig sein. Der Sachverhalt ist in Abbildung 1 dargestellt.

Visuelle und kinematische Wahrnehmung von Bewegungen sind verschieden !



Die üblichen Stetigkeitsanforderungen für Bewegungen können beim Kameraflug um mindestens **eine Ableitung** reduziert werden ! (\longrightarrow Bahnkonstruktion einfacher)

Abbildung 1: Betrachtung zur Kinematik

Durch diese Erkenntnis vereinfachen sich die Bedingungen zur Verifizierung visuell akzeptabler (harmonischer) Kamerabewegungen:

- Start- und Stoppvorgänge erfordern nur lineare Geschwindigkeitsänderungen.
- Funktionen zur Geschwindigkeitsänderung innerhalb von Bahnabschnitten müssen nicht stetig differenzierbar sein.
- Beim Zusammenfügen von Bahnabschnitten müssen nur Ort und Geschwindigkeit übereinstimmen, was bei der Wahl der betreffenden Modulparameter unmittelbar erreicht und überprüft werden kann.

Da die Ortsfunktion $s(t)$ eine Vektorgröße ist, gelten die aufgeführten Bedingungen natürlich für jede ihrer Komponenten $x(t)$, $y(t)$ und $z(t)$ einzeln.

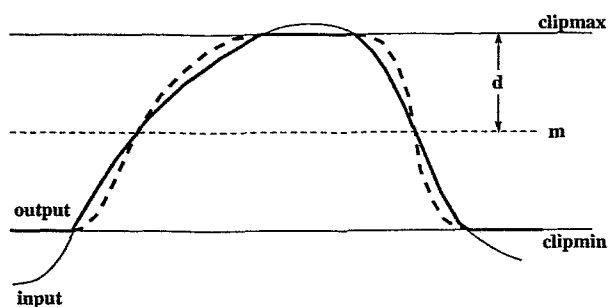
3 Der AVS-Modul float_clip

Der Modul gehört noch zur Gruppe der Module für die Bahnsteuerung, ist aber zusammen mit den Modulen zur Harmonisierung entwickelt worden und enthält ebenfalls Elemente zur Erreichung der Stetigkeit entsprechend den Erkenntnissen im vorangegangenen Abschnitt.

Er dient zur Ausformung spezieller Funktionsverläufe und zur Steuerung besonderer Ereignisse auf Flugbahnabschnitten, wie Bahnversetzungen, Begrenzungen und Kameraschwenks. Die Parameter und Funktionen des Moduls sind in Abbildung 2 aufgeführt.

Der Modul kann als Filter innerhalb derjenigen AVS-Netzwerkeile eingesetzt werden, die zur Berechnung der Ortsfunktionen für Kameraposition und -blickpunkt dienen. Mit der Wahl der Eingangsfunktion werden die Ereignispunkte, mit den Clipgrenzen die Variationsintervalle festgelegt, vgl. Abbildung 2.

Funktion und Anwendung des AVS-Moduls float_clip



Ausformung spezieller Funktionsverläufe zur Ereignissteuerung auf Flugbahnen.

Parameter : **clipmin** , **clipmax** ,
choice (linear | quadratic)

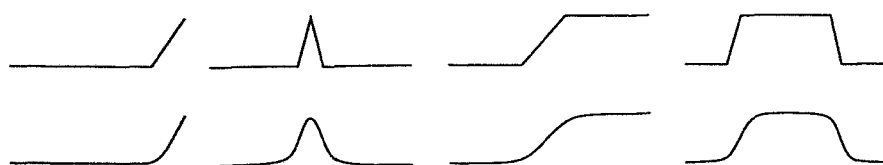
$$\text{output} = \begin{cases} \text{clipmax} & : \text{input} > \text{clipmax} \\ \text{clipmin} & : \text{input} < \text{clipmin} \\ \text{input} \mid f(\text{input}) & : \text{sonst} \end{cases}$$

Anwendung : Begrenzungen, Bahnversetzungen, Kinematik, Kameraschwenk

$$f(\text{input}) = \begin{cases} \text{clipmin} + (\text{input} - \text{clipmin})^2/d & : \text{input} \leq m \\ \text{clipmax} - (\text{clipmax} - \text{input})^2/d & : \text{input} > m \end{cases}$$

(bei choice = quadratic)

Formenvielfalt :



sowie alle Drehungen und Spiegelungen

Abbildung 2: Bahnbeschreibung

Ein Kameraschwenk ist die gezielte Veränderung des Vektors der Blickrichtung, d.h. der Differenz zwischen den Ortsfunktionen für Kamerablickpunkt und Kameraposition (vgl. from und at im Vorbericht, Kapitel 2A). Mit Kameraschwenks kann die Aufmerksamkeit auf hervorragende oder interessante Details im betrachteten Objekt gerichtet und selbst im schnellen Vorbeiflug auf sie fixiert werden.

Mittels eines Auswahlparameters kann festgelegt werden, ob die zwischen den Clipgrenzen liegenden Eingangswerte so transformiert werden sollen, daß beim Erreichen der Grenzen eine Annäherung höherer Ordnung entsteht.

Ein Verzicht auf diesen glatten Verlauf bedeutet zwar den Verlust der Differenzierbarkeit der betreffenden Koordinatenfunktionen an den Knickstellen und damit eine Verletzung der im vorangehenden Abschnitt dargelegten Stetigkeitsforderungen. Doch zeigen die Erfahrungen, daß dies nicht störend ist, wenn es sich dabei nur um Änderungen der Blickrichtung handelt.

In zahlreichen Anwendungen hat der Einsatz des Moduls besonders das Zusammenfitten von aufeinanderfolgenden Bahnabschnitten erleichtert.

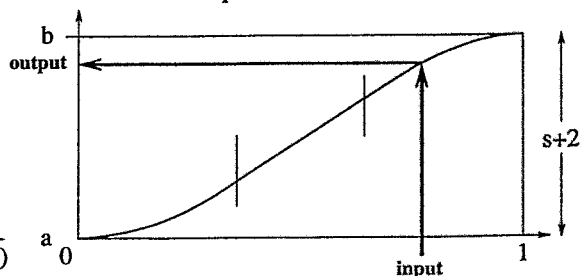
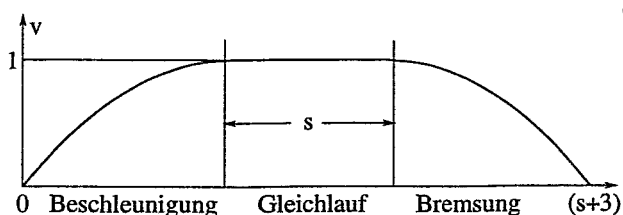
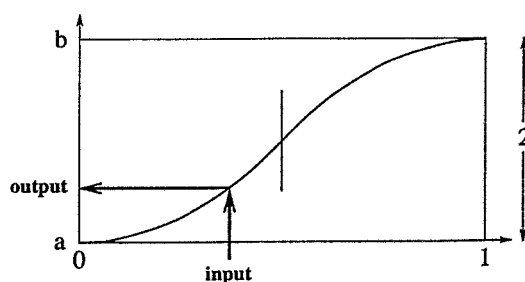
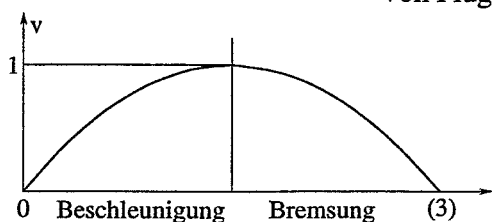
4 Der AVS-Modul `float_harmonic`

Der Modul dient zur Simulation von Beschleunigungen und Abbremsungen auf Flugbahnabschnitten und damit zur Realisierung von Start, Stop und der Richtungsumkehr bei Kamerarotationen und beim Kameraflug.

Als Filter transformiert er die vom Modul `animated_float` des AVS-Animators gelieferte Rechteckfunktion der Geschwindigkeit in die obere Halbwelle der harmonischen Schwingung, vgl. Abbildung 3. Dabei wird die Sinusfunktion durch einen Parabelbogen approximiert, was jedoch visuell nicht wahrnehmbar ist.

Geschwindigkeitsverlauf und Übertragungsfunktion des AVS-Moduls `float_harmonic`

Anwendung : Start , Stop , Richtungsumkehr von Flugbahnen



Filter `float_harmonic` kann mehrfach hintereinander verwendet werden, um Übertragungsfunktionen höheren Grades zu erzeugen.

Der Parameter s steuert den Gleichlaufanteil, $s=0$ ergibt harmonische Bewegung.

Abbildung 3: Harmonisierung (1)

Auf Grund der Ergebnisse im Abschnitt 2 würde anstelle dieses Parabelbogens ein linearer An- und Abstieg für die Stammfunktion (d.h. die transformierte Geschwindigkeit) ausreichen. Doch die o.g. Bedingungen stellen Mindestforderungen dar und die implementierte Lösung erfordert kaum Mehraufwand.

Der Modulparameter s dient zum Einfügen einer Gleichlaufphase zwischen Beschleunigung und Bremsung. Standardmäßig ist s gleich Null.

Die Eingangsdaten müssen im Intervall $[0,1]$ liegen. Davon entfällt der Abschnitt

$[0, 1.5/(s+3)]$	auf die Beschleunigung,
$(1.5/(s+3), (s+1.5)/(s+3))$	auf den Gleichlauf und
$((s+1.5)/(s+3), 1]$	auf die Bremsung.

Die Ausgabewerte entsprechen dem Integral über die Geschwindigkeitsfunktion (von Null bis zum Eingabewert). Sie werden standardmäßig auf das Intervall $[0,1]$ transformiert, können jedoch mittels der Parameter a, b nachträglich verändert werden. Als Wegfunktion überstreichen sie die Teilintervalle

$[0, 1/(s+2)]$	für die Beschleunigung,
$(1/(s+2), (1+s)/(s+2))$	für den Gleichlauf und
$((1+s)/(s+2), 1]$	für die Bremsung.

Mit diesen Werten erfolgt in der Regel die Berechnung der Ortsfunktionen für Kameraposition oder/und Kamerablickpunkt.

Somit ist es möglich, mit nur einem AVS-Netzwerk eine Sequenz zu erzeugen, die eine vom Stillstand ausgehende harmonisch beschleunigte Bewegung, einen Gleichlaufanteil (je nach Wert des Parameters s) und eine bis zum Stillstand harmonisch abgebremste Bewegung enthält. Dabei muß der Eingangswert t das gesamte Intervall $[0,1]$ durchlaufen.

Will man innerhalb eines Flugbahnabschnittes nur von Null bis zu einem gewissen Endwert beschleunigen oder von einer gewissen Anfangsgeschwindigkeit auf Null abbremsen, so hat man den Parameter s auf dem Standardwert ($s=0$) zu belassen und dafür zu sorgen, daß die Eingangswerte jeweils das o.g. Teilintervall, also $[0,0.5]$ oder $[0.5,1]$ überstreichen.

Man kann den Modul auch mehrfach hintereinander einsetzen, um für die Geschwindigkeit Nullstellen höherer Ordnung zu erreichen.

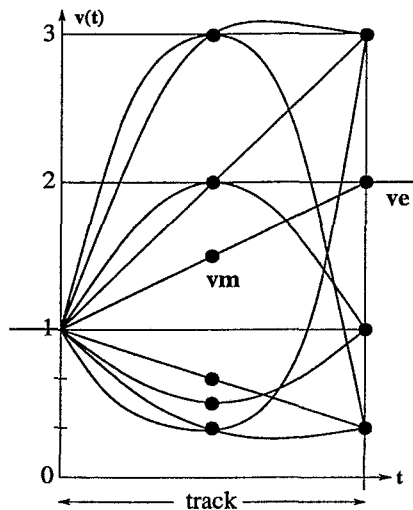
Der Modul `float.harmonic` ist in sämtlichen Videos, die nach seiner Fertigstellung erzeugt wurden, für Starts und Stops von Kamerarotationen und -flügen eingesetzt worden.

5 Der AVS-Modul `modify_velocity`

Der Modul dient zur Veränderung der Fluggeschwindigkeit innerhalb eines Bahnabschnitts, ohne daß dabei eine Stillstandsphase entstehen soll. Dafür kommen folgende Situationen in Betracht:

- Innerhalb des Bahnabschnitts sollen interessante Details der Szenerie genauer inspiziert werden, die Fluggeschwindigkeit ist in diesem Bereich zu verringern.
- Der Bahnabschnitt ist Bestandteil der Kamerabewegung von einer festen Position zur nächsten. Um die Flugzeit zu verkürzen, ist die Geschwindigkeit im mittleren Teil des Abschnitts zu erhöhen.
- Der Bahnabschnitt liegt in einer Phase genereller Geschwindigkeitsänderungen (Geschwindigkeitstrend).

Abb.4 zeigt die Funktionen des Moduls und einige mögliche Verläufe der Geschwindigkeit $v(t)$. Mit den Parametern v_m und v_e sind die Werte in der Mitte und am Ende des Bahnabschnitts vorzugeben, bezogen auf den Anfang des Bahnabschnitts. Die Funktion $v(t)$ ergibt sich dann aus der interpolierenden Parabel. Die Werte v_m und v_e sind auf das Intervall $[1/3, 3]$ begrenzt, um allzu starkes Überschwingen zu vermeiden. Die Standardwerte sind $v_m = 1, v_e = 1$.



Funktion des AVS-Moduls **modify velocity**

Veränderung der Bahngeschwindigkeit im Verlauf eines Abschnitts (track).

$$\text{output} = \int_0^{\text{input}} v(t) dt \quad (\text{zuzügl. Skalierung, Translation})$$

Parameter : v_m und v_e , bezogen auf das Ende des vorangegangenen Abschnittes ($\hat{=} 1$)

Grenzen : $1/3 \leq v_m, v_e \leq 3$

Anwendung:

- Flugzeitverkürzung ($v_m > 1$)
- partielle Verzögerung (z.B. Inspektion) ($v_m < 1$)
- Geschwindigkeitstrend ($v_e \lesseqgtr 1$)

Der Modul **modify velocity** zeigt außerdem die spezifischen Bahnschrittzahlen des aktuellen und nachfolgenden Abschnitts an, bezogen auf den vorangegangenen Abschnitt (empfohlenes Geschwindigkeitsfitting an den Verbindungsstellen).

Abbildung 4: Harmonisierung (2)

Die Eingangswerte t müssen das Intervall $[0,1]$ überstreichen. Die Ausgabewerte entsprechen dem Integral über die Funktion $v(t)$, sie werden standardmäßig auf den Bereich $[0,1]$ transformiert, können aber durch zwei weitere Modulparameter auf ein beliebiges Intervall verändert werden. Sie sind den Modulen zur Berechnung der Ortsfunktionen für Position und Blickpunkt der Kamera zuzuleiten.

Zur Vereinfachung der stetigen Verknüpfung des aktuellen Bahnabschnitts mit seinem Vorgänger und Nachfolger (entsprechend den Bedingungen aus Abschnitt 2) gibt der Modul die empfohlenen spezifischen Bahnschrittzahlen des aktuellen und nachfolgenden Bahnabschnitts, bezogen auf den vorangegangenen, an den Nutzer aus (das sind die Werte n für den Modul `animated_float` des AVS-Animators).

Als Filter kann der Modul auch mehrfach hintereinander eingesetzt oder mit dem Modul `float_harmonic` kombiniert werden. Dabei sind nur die Standardwerte für die Eingabebereiche einzuhalten.

6 Schlußbemerkung

Mit den hier vorgestellten neuen AVS-Modulen sind die Arbeiten zur Problematik Szene-Kamera-Rotation-Positionsveränderungen abgeschlossen worden. Der Einsatz dieser Module in bisher sieben erstellten Videos erbrachte realitätsnahe Kamerabewegungen mit ansprechender Dynamik und hoher Akzeptanz.