Gigabit Testbed West

1. August 1997 – 31. Januar 2000
Abschlußbericht

Forschungszentrum Jülich GmbH

GMD – Forschungszentrum Informationstechnik GmbH

Stiftung Alfred-Wegener-Institut für Polar- und Meeresforschung

Pallas GmbH

März 2000

Federführung:
Zentralinstitut für Angewandte Mathematik
Forschungszentrum Jülich GmbH
52425 Jülich

Ansprechpartner:
Dr. Thomas Eickermann, E-Mail: th.eickermann@fz-juelich.de, Tel.: 02461-61-6596
Dr. Roland Völpel, E-Mail: roland.voelpel@gmd.de, Tel.: 02241-14-2298
Peter Wunderling, E-Mail: peter.wunderling@gmd.de, Tel.: 02241-14-2930

WWW: http://www.fz-juelich.de/gigabit

Dieses FE-Vorhaben wurde vom Verein zur Förderung eines Deutschen Forschungsnetzes e.V. unter dem Kennzeichen „TK 598 – NT041“ mit Mitteln des Bundesministeriums für Bildung und Forschung gefördert.

Beteiligte Institutionen und Personen

An diesem Projekt waren folgende Institutionen beteiligt:

Forschungszentrum Jülich GmbH
John von Neumann – Institut für Computing
Zentralinstitut für Angewandte Mathematik (ZAM)
Prof. Dr. Friedel Hoßfeld

Forschungszentrum Jülich GmbH
Institut für Medizin (IME)
Prof. Dr. med. Karl Zilles

Forschungszentrum Jülich GmbH
Institut für Erdöl und Organische Geochemie (ICG-4)
Prof. Dr. Dr. h.c. Dietrich Welte

Stiftung Alfred-Wegener-Institut für Polar- und Meeresforschung (AWI)
Bereich Wissenschaftliches Rechnen
Dr. Wolfgang Hiller

GMD – Forschungszentrum Informationstechnik GmbH
Institut für Algorithmen und Wissenschaftliches Rechnen (SCAI)
Prof. Dr. Ulrich Trottenberg

GMD – Forschungszentrum Informationstechnik GmbH
Institut für Medienkommunikation (IMK)
Prof. Dr. Martin Reiser

Pallas GmbH, Brühl
Karl Solchenbach

Die folgenden Personen haben mitgewirkt:

Dr. Alfred Arnold, ZAM
Dr. Roland Beucker, ZAM
Kläre Cassirer, SCAI
Dr. Dieter Conrads, ZAM
Jürgen Dammers, IME
Dr. Thomas Eickermann, ZAM
Ursula Eisenblätter, IMK
Dr. Rüdiger Esser, ZAM
Dr. Stephan Frickenhaus, AWI
Wolfgang Frings, ZAM
Dr. Bernadette Fritzsch, AWI
Dr. Martin Göbel, IMK
Dr. Udo Göbel, AWI
Gernot Goebbels, IMK
Daniel Gembris, IME
Georg Glock, IMK
Dr. Thorsten Graf, ICG-4
Arnfried Griesert, IMK
Helmut Grund, SCAI
Matthias G. Hackenberg, SCAI
Anke Häming, ZAM
Horst Hardelauf, ICG-4
Dr. Jörg Henrichs, Pallas
Dr. Olaf Heudecker, AWI
Dr. Wolfgang Hiller, AWI
Jochen Hirsch, IMK
Ferdinand Hommes, IMK
Dr. Wolfgang Joppich, SCAI
Dr. Manfred Kaul, IMK
Udo Keller, Pallas
Dr. Thomas Kentemich, Pallas
Renate Knecht, ZAM
Axel Klier, SCAI
Kirstin Krüger, IMK
Dr. Burkhard Mertens, ZAM
Martha Merzbach, IMK
Prof. Dr. Wolfgang E. Nagel, TU Dresden
Ralph Niederberger, ZAM
Ulrich Nütten, IMK
Klaus-Dieter Oertel, Pallas
Eva Pless, IMK
Priv.-Doz. Dr. Stefan Posse, IME
Dr. Peter Post, SCAI
Johannes Quaas, SCAI
Dr. René Redler, AWI
Martin Sczimarowsky, ZAM
Dr. Jon Shah, IME
Karl Solchenbach, Pallas
Dr. Thomas Störtkuhl, AWI
Alexander Supalov, Pallas
Dr. Peter Tass, IME
Priv.-Doz. Dr. Harry Vereecken, ICG-4
Dr. Roland Völpel, SCAI
Wolfgang Vonolfen, IMK
Sabine Werner, ZAM
Klaus Wolf, SCAI
Peter Wunderling, IMK
Lothar Zier, IMK
Dr. Herwig Zilken, ZAM

Inhalt

1 Einführung

Im Frühjahr 2000 nimmt der DFN-Verein das neue Gigabit-Wissenschaftsnetz (G-WiN) in Deutschland in Betrieb. Die zunehmende Nutzung traditioneller Internet-Dienste sowie neue Anwendungen, die hohe Kommunikationsleistungen benötigen, erfordern die Ablösung des jetzigen Breitband-Wissenschaftsnetzes (B-WiN) durch eine leistungsfähigere Infrastruktur. Zur Vorbereitung dieses Schrittes hat der DFN-Verein zwei Testbeds initiiert und mit Mitteln des Bundesministeriums für Bildung und Forschung (BMBF) gefördert. Als erstes dieser beiden Projekte startete das Gigabit Testbed West am 1. August 1997 mit der Inbetriebnahme einer 622 Mbit/s ATM/SDH-Verbindung zwischen Forschungszentrum Jülich und GMD – Forschungszentrum Informationstechnik in Sankt Augustin. Dieses Projekt wurde planmäßig zum 31. Januar 2000 mit einem zweitägigen Abschlußkolloquium in Jülich beendet, auf dem die Ergebnisse des Projektes in Vorträgen und Live-Demonstrationen der Fachöffentlichkeit vorgestellt wurden.

Schwerpunkte im Gigabit Testbed West waren die Erprobung und Untersuchung der Gigabit-Netztechnik sowie der exemplarische Nachweis ihres Nutzens durch Anwendungen vor allem aus dem Umfeld des wissenschaftlichen Rechnens. Dazu wurden in zunächst fünf Teilprojekten verschiedene Aspekte der Kopplung von Höchstleistungsrechnern (Metacomputing) und der verteilten Visualisierung untersucht. Im Vordergrund stand dabei die Kopplung strukturell unterschiedlicher Ressourcen, um Probleme mit heterogenen Anforderungen effektiv behandeln zu können. Dazu wurden die massiv-parallelen und vektor-parallelen Supercomputer sowie High-End Visualisierungsserver und immersiven Display-Systeme in Jülich und Sankt Augustin an das Testbed angebunden und auf Anwendungsebene miteinander gekoppelt. Genutzt wurde dieser heterogene Metacomputer nicht nur für numerische Simulationen. In Verbindung mit einem Magnetresonanz-tomographen gelang es, Meßdaten in Echtzeit zu analysieren und visualisieren – ein Beispiel eines „computer-enhanced instrument“. Außer zum Datenaustausch zwischen Höchst-leistungsrechnern wurde die Bandbreite des Testbeds auch zur Evaluierung von Geräten für die verteilte Videoproduktion in professioneller Qualität genutzt.

Als überraschend stabil erwiesen sich die WAN-Verbindungen und die eingesetzten Netzwerkkomponenten. Die von der o.tel.o (heute Mannesmann ARCOR AG) angemietete Glasfaserstrecke mit SDH-Technik lief von Anfang an problemlos. Im August 1998 wurde die Strecke von 622 Mbit/s auf 2,5 Gbit/s umgestellt. Mit den beiden ersten außerhalb der USA ausgelieferten Seriengeräten der neuesten ATM-Switches von FORE-Systems (ASX-4000) wurden erstmals in Europa Benutzerdaten mit einer Geschwindigkeit von annähernd 2,4 Gbit/s über ein WAN übertragen. Nach anfänglichen Dämpfungsproblemen auf der Strecke lief auch diese Verbindung bis zum Projektende im Januar 2000 sehr stabil.
Im Jahr 1999 wurden weitere Einrichtungen im Köln-Bonner Raum an das Testbed angebunden. Im April war die Verbindung von der GMD über die DLR zur Universität und der Kunsthochschule für Medien (KHM) in Köln betriebsbereit. Im Dezember waren auch die Universität in Bonn sowie caesar (center of advanced european studies and research) angebunden. Diese, zur Zeit mit 622 Mbit/s betriebenen Strecken, bleiben noch über das Projektende hinaus in Betrieb. Die vier Anwendungsprojekte, welche die neuen Verbindungen nutzen, sind organisatorisch unabhängig von dem im Januar beendeten Projekt „Gigabit Testbed West“ und laufen noch bis Mitte des Jahres 2000.

Das Projekt wurde immer wieder durch die verspätete oder gänzlich ausbleibende Verfügbarkeit von ATM-Adaptern für die Endgeräte behindert. So konnten 1998 lediglich die

Workstations und Server von SUN mit 622 Mbit/s angeschlossen werden. Für die SGI Onyx-2 waren entsprechende Adapter erst Mitte 1999 lieferbar. Für die CRAY-Rechner T90 und T3E und die IBM SP2 der GMD (mit MicroChannel-basierten Knoten) wurde im Projekt eine andere Lösung gefunden. Durch den Einsatz von dedizierten Workstations als HiPPI-ATM-Gateways konnte 1998 der erzielbare Durchsatz gegenüber den standardmäßig verfügbaren 155 Mbit/s-ATM-Adaptern mehr als verdreifacht werden. Dies war eine wesentliche Voraussetzung für den Erfolg der Metacomputing-Teilprojekte.

Trotz dieser Schwierigkeiten und einer insbesondere im ersten Jahr der Projektlaufzeit hohen Personalfluktuation konnten alle Anwendungs-Teilprojekte erfolgreich und termingerecht abgeschlossen werden. Das wurde u.a. dadurch möglich, dass den Anwendern von Beginn an eine stabile Entwicklungsumgebung zur Verfügung stand. Dazu hat außer der Zuverlässigkeit der Netzwerkverbindungen auch die von der Universität Stuttgart entwickelte und frühzeitig zur Verfügung gestellte Kopplungsbibliothek PACX-MPI beigetragen. PACX-MPI ist eine verteilte MPI-Implementierung, die im Testbed bis zur Fertigstellung der projektbegleitend von der Pallas GmbH (im Auftrag von Forschungszentrum Jülich und GMD) entwickelten MetaMPI-Bibliothek eingesetzt wurde. Es mußten lediglich solche Arbeitspakete im Zeitplan nach hinten verschoben werden, die Leistungsmessungen und Optimierungen umfaßten oder die auf bestimmte, nur in MetaMPI enthaltene MPI-Funktionalität angewiesen waren.
In diesem Bericht sind die Arbeiten und Ergebnisse der insgesamt acht Teilprojekte des Gigabit Testbed West dargestellt. Die Resultate der vier noch laufenden Projekte der Universitäten in Bonn und Köln sowie der KHM mit der GMD werden nach ihrem Ende in separaten Berichten vorgestellt.

An den erzielten Erkenntnissen und Ergebnissen aller Projekte, die direkt oder indirekt in den Aufbau des G-WiN des DFN einfließen, können auch weitere zukunftsorientierte FuE-Aktivitäten im Bereich der Netztechnik und den darauf aufsetzenden Anwendungen partizipieren. Die nächste Generation der schnellen Netze – die optischen Netze – kann bereits heute prototypisch eingesetzt werden. Hier bietet sich an, die Netz- und Anwendungsumgebung des GTB-West auf ein optisches Testbed des DFN aufzusetzen. Der Zeitpunkt ist ideal: Anwendungen, Netzinfrastruktur und Knowhow sind vorhanden, um in ein optisches Testbed „durchzustarten“.

Wir danken dem DFN-Verein, insbesondere Herrn Dr. Jürgen Rauschenbach und Frau Dr. Gudrun Quandel, für die Unterstützung und wohlwollende Begleitung und dem BMBF für die Förderung des Projektes.

2 GIGAnet

Beteiligte Partner: *Institut für Medienkommunikation (IMK/GMD)*
Zentralinstitut für Angewandte Mathematik (ZAM/FZJ)

Ansprechpartner: *Ferdinand Hommes (IMK/GMD), Ralph Niederberger (ZAM/FZJ)*
Weitere Beteiligte: *Dr. Dieter Conrads (ZAM/FZJ), Ursula Eisenblätter (IMK/GMD), Jochen Hirsch (IMK/GMD), Martha Merzbach (IMK/GMD), Eva Pless (IMK/GMD), Martin Sczimarowsky (ZAM/FZJ), Sabine Werner (ZAM/FZJ), Peter Wunderling (IMK/GMD), Lothar Zier (IMK/GMD)*

2.0 Projektbeschreibung

Um die geplanten Anwendungen im Gigabit Testbed durchzuführen, war eine professionelle Betreuung der Netztechnik notwendig. Diese Arbeiten wurden im Teilprojekt GIGAnet durchgeführt. Zu den Aufgaben von GIGAnet gehörten:

- die Auswahl und Beschaffung der Netzkomponenten, der Aufbau und Betrieb des Netzes und die Konfiguration der IP-Netze,
- die Durchsatzmessungen und das Tuning der Netzkomponenten für den Einsatz in einem Hochgeschwindigkeitsnetz,
- die Bereitstellung einer optimalen Anbindung der Supercomputer,
- und die Untersuchung des Netzverhaltens im Überlastfall.

In den folgenden Abschnitten werden die durchgeführten Arbeiten und die dabei gewonnenen Ergebnisse im Einzelnen beschrieben.

2.1 Betrieb und Konfiguration des Netzes

2.1.1 Leitungen

Das Gigabit Testbed West ist ein ATM basiertes Hochgeschwindigkeitsnetz, das Forschungseinrichtungen aus dem Köln-Bonner-Aachener Raum miteinander verbindet. Abbildung 2.1 zeigt den Stand des Netzes Ende Januar 2000. Die Bandbreiten betragen 622 Mbit/s und 2,4 Gbit/s. Die Leitungen werden von unterschiedlichen Anbietern und mit unterschiedlichen Techniken zur Verfügung gestellt. Während die Strecke GMD-FZJ auf Grund ihrer Länge eine von o.tel.o betriebene SDH-Verbindung ist, sind alle anderen Verbindungen als Dark Fiber ausgelegt. Die Leitungen im Bonner Raum wurden von der Deutschen Telekom angemietet. Im Bereich Köln wurde als Anbieter Net Cologne ausgewählt. Die GMD und die DLR sind über eigene Glasfaserleitungen miteinander verbunden.

Das Gigabit Testbed West wurde in drei Phasen aufgebaut. In der ersten Phase wurde die Leitung zwischen der GMD und dem FZJ im August 1997 zunächst mit 622 Mbit/s Übertragungsrate geschaltet. Die Leitung lief stabil und ohne Probleme. Anfang August 1998 wurde, nachdem geeignete ATM-Switche zur Verfügung standen, die Leitung als eine der ersten Verbindungen in Europa auf 2,4 Gbit/s hochgerüstet. Zum Einsatz kamen ATM-Switche von FORE (ASX-4000), deren 2,4 Gbit/s-Interfaces noch Prototypen waren und die uns als Beta-Test-System zur Verfügung gestellt wurden. Am 5.8.1998 wurde dann bei ersten Tests eine Leitungsauslastung von 99,97 % bei UDP-Verkehr und 96,4 % bei TCP-Verkehr

erreicht. Im Rahmen einer Presseerklärung wurde auf diesen außergewöhnlichen Erfolg hingewiesen.

Während der Beta-Testzeit führten in den darauffolgenden Monaten Probleme mit Dämpfungsgliedern und dem Timing der Switche untereinander zu einigen Betriebsstörungen. Die beiden Probleme überlappten sich teilweise, so dass die Fehlersuche sehr schwierig war. Die Dämpfungsglieder (10 db) werden benötigt, um die 2,4 Gbit/s Single Mode Interfaces der ASX 4000 Systeme an die SDH-Komponenten von o.tel.o anzuschließen. Bei der 622 Mbit/s-Verbindung zwischen der GMD und dem FZJ gab es keine Timing-Probleme. Beim Übergang zu 2,4 Gbit/s traten jedoch wiederholt SDH-Fehler auf. Mit Hilfe von FORE ist es gelungen, eine Einstellung für die ASX 4000-Systeme zu finden, die fehlerfrei funktioniert, so dass es derzeit keinerlei Probleme mit der Gigabit-Strecke gibt. Am 1.10.1998 wurde dann die Gigabit-Verbindung offiziell in Betrieb genommen und stand bis Ende Januar 2000 zur Verfügung.

In der zweiten Phase wurde das Gigabit Testbed West auf den Kölner Raum ausgedehnt. Seit dem 3.5.1999 gibt es eine durchgehende 622 Mbit/s ATM-Verbindung zwischen der GMD, der DLR, der Universität Köln und der Kunsthochschule für Medien. Als ATM-Switches wurden Systeme von CISCO eingesetzt (MSR8540 und LS1010), da in dieser Phase auch die Gigabit Ethernet Technik eingesetzt werden sollte. Die für die DLR geplanten Systeme wurden leider nicht beschafft, so dass die 50 km lange Strecke von der GMD zur Universität Köln ohne Zwischenverstärker überbrückt werden musste. In der GMD wurde hierzu ein Single-Mode-Multi-Mode Wandler eingesetzt, mit dem sich Entfernungen bis 60 km überbrücken lassen. In Köln werden Single Mode Interfaces eingesetzt, deren maximale Reichweite bei 50 km liegt. Für die Verbindung Universität Köln zur KHM ist dies ausreichend. Die Verbindung zur GMD wird jedoch im Grenzbereich betrieben. Es kam wiederholt zu Sonet-Fehlern, die sich jedoch bisher immer durch Reinigen der Glasfaserverbindungen beheben ließen. In den letzten Monaten kam es zu keinerlei Ausfällen mehr. Die Leitung steht noch bis Ende Juni 2000 zur Verfügung.

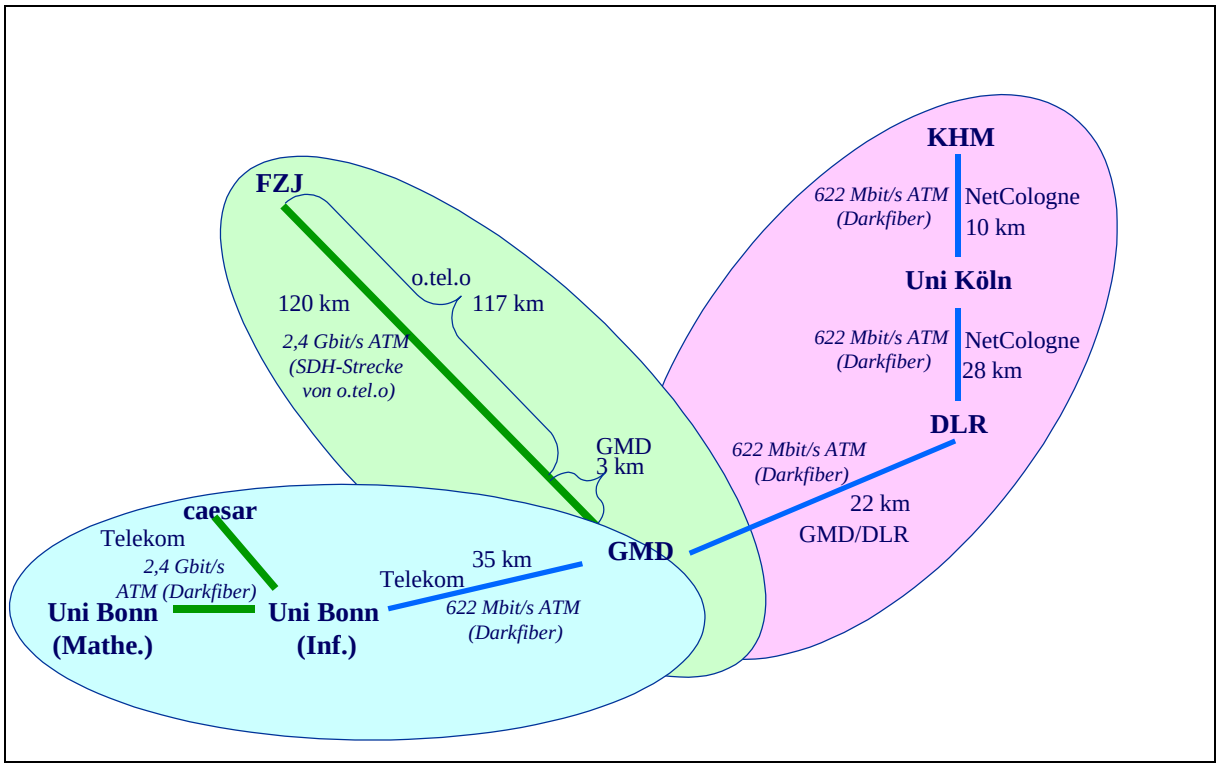


Abb. 2.1: Netztopologie

In der dritten Phase wurde der Bonner Raum an das Gigabit Testbed West angebunden. An der Universität Bonn und bei caesar wurden CISCO MSR8540-Systeme installiert und diese mit 2,4 Gbit/s miteinander verbunden. Die Anbindung an die GMD erfolgt zur Zeit mit 622 Mbit/s. Bisher traten keinerlei Störungen auf. Die Leitungen stehen auch noch nach dem Ende des Gigabit Testbed West bis Ende 2002 zur Verfügung, da sie im Gegensatz zu den anderen Leitungen vollständig von den Projektpartnern selbst finanziert werden.

2.1.2 ATM-Routing

In der Anfangsphase wurde im Gigabit Testbed West zwischen den Netzen der Partner ausschließlich statisches Routing auf IISP-Basis eingesetzt, während lokal schon PNNI als ATM Routing Protokoll verwendet wurde. Einem Einsatz von PNNI als globales Routing Protokoll stand zunächst die mangelnde Unterstützung von hierarchischem PNNI in den FORE ATM-Switches entgegen. In Version 6 der FORE-Switch Software wurde diese Funktionalität zu Beginn des 2. Quartals 1999 zur Verfügung gestellt. Erste Tests innerhalb des ATM-Netzes der GMD ergaben Interoperabilitätsprobleme zwischen der FORE- und CISCO-PNNI-Software, die auf unterschiedliche Auslegung der Standards zurückzuführen sind. Die Fehler wurden an CISCO und FORE gemeldet. Leider wurde die Behebung des Problems zwischen den beiden Herstellern hin und hergeschoben. Zur Zeit liegt die Behebung des Fehlers bei CISCO. Daher konnte hierarchisches PNNI nur zwischen den Peer Groups der GMD und dem FZJ und zwischen den Peer Groups der Universität Bonn und von caesar eingesetzt werden. Abbildung 2.2 gibt die Struktur des derzeitigen PNNI-Netzes wieder.

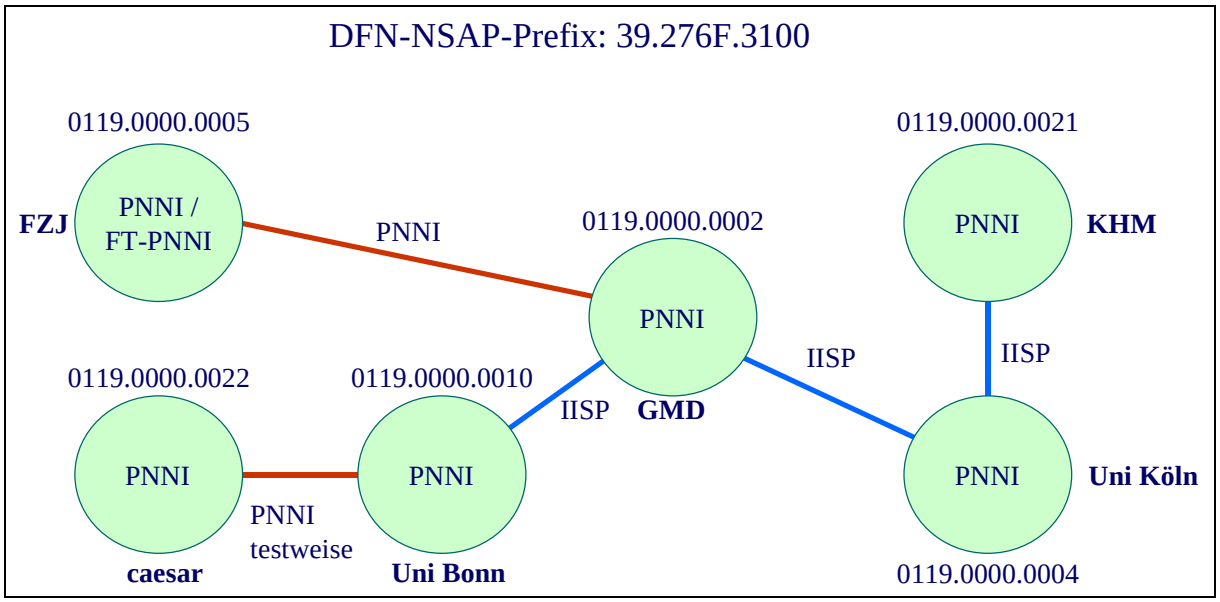


Abb. 2.2: ATM-Routing

2.1.3 Netztechniken

Wie bereits erwähnt basiert das Gigabit Testbed West auf ATM als Übertragungsprotokoll. Neben native-ATM-Anwendungen aus dem Multi-Media-Bereich wird hauptsächlich TCP/IP als Übertragungsprotokoll eingesetzt. Es gibt zwei Techniken TCP/IP über ATM zu realisieren: LAN-Emulation (LANE) und Classical IP (CIP). Beide Techniken kommen im Gigabit Testbed zum Einsatz. Die meisten Endsysteme im Workstation-Bereich wurden anfangs mit 155 Mbit/s oder 622 Mbit/s ATM-Adapterkarten ausgestattet, die beide Protokolle unterstützen. Die Anbindung der Supercomputer über ATM hat sich als schwierig erwiesen (siehe Kapitel 2.3). Ihre Anbindung wurde über Workstations realisiert, die als

Router zwischen HiPPI- und ATM-Netzen arbeiten. In den letzten Monaten wurden Workstations auch verstärkt mit Gigabit-Ethernet-Karten ausgerüstet, da diese bei vergleichbarem Durchsatz ein wesentlich besseres Preisleistungsverhältnis bieten. Die Umsetzung von Gigabit Ethernet auf ATM LAN-Emulation wurde mit Hilfe der CISCO MSR8540-Systeme realisiert, die eine Kombination aus Ethernet-Switch und ATM-Switch mit einem speziellen Routing Modul darstellen. Abbildung 2.3 zeigt die Netztechniken, die bei den einzelnen Organisationen zur Verfügung gestellt werden.

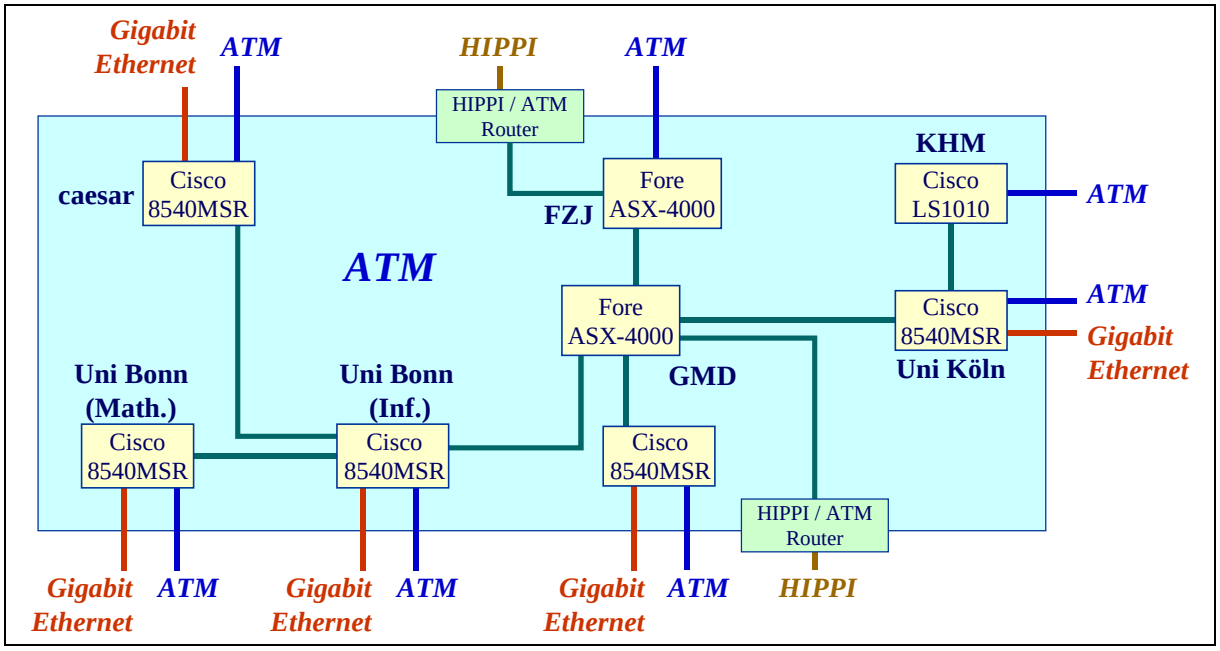


Abb. 2.3: Netztechniken

2.1.4 IP-Netzstruktur

Wie aus der vorhergehenden Abbildung ersichtlich, wird IP im Gigabit Testbed West über unterschiedliche Netztechniken angeboten: IP über ATM, IP über HiPPI und IP über Ethernet. Es wurden unterschiedliche IP-Netze für die einzelnen Anwendungsprojekte eingerichtet, die jeweils für den speziellen Anwendungsfall optimiert waren. Die einzelnen Netze sind zum Teil standortübergreifend, um Routing zu vermeiden, und zum Teil lokal.

Abbildung 2.4 zeigt die unterschiedlichen Netze und ihre Verbindungen untereinander. Die Supercomputer-Gateways und die Highend Workstations wurden über ein gemeinsames Classical IP-Netz miteinander verbunden. In diesem Netz werden IP-Pakete bis 60 000 Byte unterstützt, wodurch ein optimaler Durchsatz erreicht werden kann. Da Classical IP kein Broadcast und Multicast unterstützt, wurde für Anwendungen, die diese Funktionalität benötigen, ein LAN-Emulation-Netz eingerichtet. Da LANE und Gigabit Ethernet aufgrund der im Verhältnis zu Classical IP kleinen maximalen Paketgröße von 1500 Byte einen wesentlich schlechteren Durchsatz bieten (siehe Kapitel 2.2), wurden für Multicast-Anwendungen spezielle Ethernet-Netze eingerichtet, die auf proprietärer Basis (Jumbo Frames) auch größere Paketgrößen unterstützen.

2.2 Durchsatzwerte in Hochgeschwindigkeitsnetzen

Neben der Bereitstellung, dem Betrieb und der Überwachung der Netztechnik wurden Delay- und Performance-Messungen durchgeführt, um Aussagen über das Verhalten der ATM-Strecken und der im Gigabit Testbed West vorgesehenen Rechner zu gewinnen.

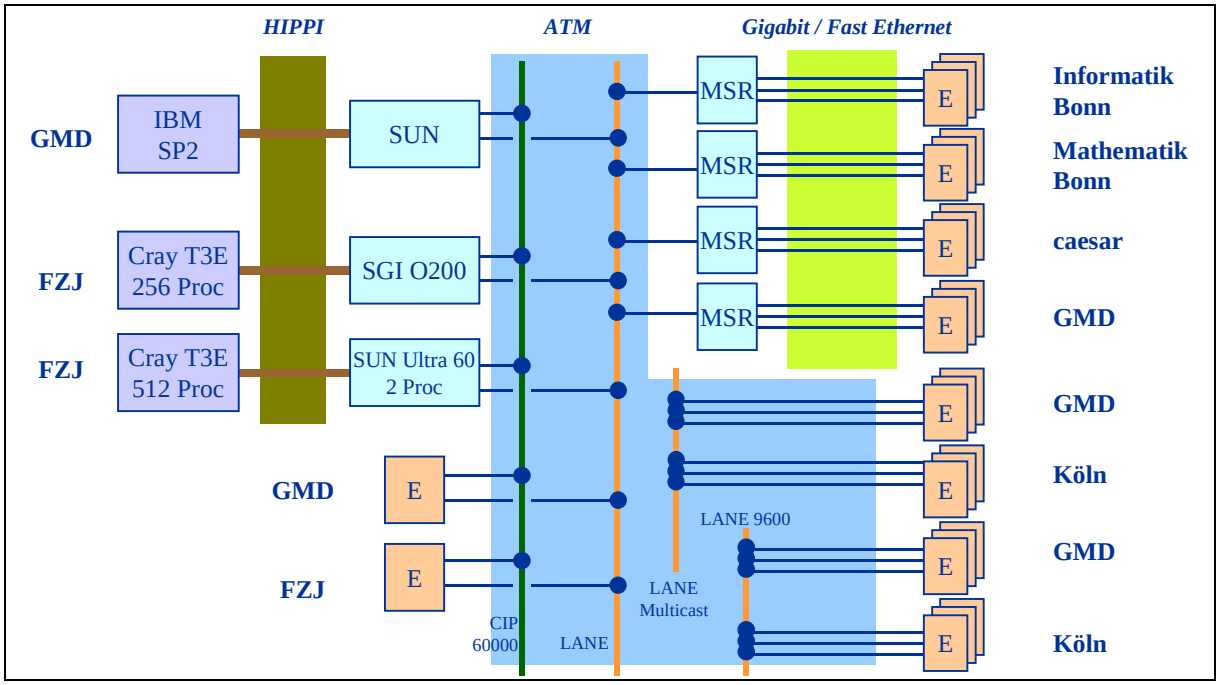


Abb. 2.4: IP Netzstruktur

2.2.1 Delay-Messungen

Die Delay-Messungen wurden in der ersten Hälfte 1998 mit dem ATM-Analyser GN Nettest Interwatch 95000 auf der 622 Mbit/s-Strecke zwischen der GMD und dem FZJ durchgeführt. Hierbei gab es folgende Ergebnisse:

- Der Delay in einem ATM-Switch (switching delay plus transmission delay) beträgt ca. 20-25 µs bei 155 Mbit/s. Bei 622 Mbit/s ist der Wert ca. 5 µs günstiger. Diese Werte entsprechen in etwa den Herstellerangaben.
- Bei einer Port-Auslastung über 95% kommt es zu Schwankungen, da hier switch-interne Parameter wie z.B. Buffer Size eine Rolle spielen.
- Das Delay auf der ATM-Strecke GMD-FZJ (3 ATM-Switches, 3 Line Repeater, 120 km) und zurück beträgt ca. 1,4 ms. Davon entfällt 1,2 ms auf die Verzögerung durch die 240 km Glasfaser.

Die Verzögerung von 1,4 ms hat Auswirkungen auf den maximalen TCP/IP-Durchsatz. Bei Benutzung der Standard TCP/IP Buffer Size/Window Size von 64 KByte (Standard bei den meisten Betriebssystemen) beträgt der maximale Durchsatz 64 KByte/s pro 1,4 ms, d.h. etwa 366 Mbit/s bei 622 Mbit/s. Dieser Wert wurde durch Messungen bestätigt. Die Verzögerung zwischen den Supercomputern betrug etwa 7 ms. Neben der Verzögerung auf der ATM-Leitung kommen hier noch die Durchlaufzeiten durch die HiPPI-Switche und die HiPPI/ATM-Router hinzu. Bei einer Vergrößerung der Buffer Size auf z.B. 1 MByte kann der maximale Durchsatz erreicht werden.

2.2.2 Durchsatz-Messungen

Für die Anfang 1998 durchgeführten Durchsatzmessungen wurde das Programm Netperf benutzt. Auf zwei eigens aufgesetzten gemeinsamen (GMD und FZJ) Classical IP-Netzen und einem gemeinsamen LAN-Emulation-Netz wurden die Messungen sowohl für TCP als auch

für UDP durchgeführt. Als Betriebssystem dienten Solaris 2.6 und Windows NT 4.0. Die PCs waren mit 155 Mbit/s-Adaptern ausgerüstet, während die SUNs mit 155 Mbit/s und 622 Mbit/s-Adaptern angebunden waren. Die theoretisch maximal möglichen Durchsätze sind in Tabelle 2.1 wiedergegeben.

Als Ergebnis der praktischen Messungen (siehe Tabelle 2.2) lässt sich Folgendes festhalten: Bei Maschinen mit einem 155 Mbit/s-Adapter erreicht man den maximalen Durchsatz bei einer Taktfrequenz von ca. 200 MHz und größer. Bei Maschinen mit 622Mbit/s-Adapter lässt sich der maximale Durchsatz nur für Classical IP erreichen. LAN-Emulation bleibt hinter den Erwartungen zurück. Hier liegt der maximale Durchsatz bei ca. 200-300 Mbit/s. Dies ist wahrscheinlich auf die kleinere Paketgröße (1500 Byte gegenüber 9180 Byte) zurückzuführen. Die damaligen Highend-Maschinen mit einer Taktrate von 300-400 MHz bewältigen die Menge der anfallenden IO-Interrupts (ca. 44.000/s bei 622 Mbit/s und LAN-Emulation) nicht. Außerdem wurde festgestellt, dass im 622 Mbit/s-Fall der Durchsatz bei Mehrprozessor-Maschinen höher ist.

	155 Mbit/s	622 Mbit/s	2,4 Gbit/s	%
Kapazität	155,520	622,080	2.488,320	100
SDH Overhead	5,760	23,040	92,160	3,7
ATM Overhead	14,128	56,513	226,053	9,1
ATM Nutzdaten	135,632	542,527	2.170,107	87,2
ATM Zellen	353.208	1.412.830	5.651.321	
CIP TCP 9140 Overhead	1,118	4,474	17,896	0,7
CIP TCP 9140 Nutzdaten	134,513	538,053	2.152,211	86,5
CIP TCP 9140 Pakete/s	1840	7.358	29.434	
LANE TCP 1460 Overhead	6,711	26,844	107,375	4,3
LANE TCP 1460 Nutzdaten	128,921	515,683	2.062,732	82,9
LANE TCP 1460 Pakete/s	11.038	44.151	176.604	

Tabelle 2.1: Maximale Durchsatzwerte

	Adapter	LANE	CIP
Ultra 30, Sol. 2.6, 248 MHz	622	130	408
Ultra 60 (2P), Sol. 2.6, 296 MHz	622	199	530
EP 5000 (8P), Sol. 2.6, 248 MHz	622	155	501
PC, NT4.0, PII, 333 MHz	155	120	131
PC, NT4.0, P Pro, 200 MHz	155	120	130
Onyx2, (12P), IRIX 6.4, 195 MHz	155	82	126

Tabelle 2.2: gemessene Durchsatzwerte für LANE und CIP, Werte in Mbit/s

Bei den Durchsatzmessungen mittels des Programms Netperf handelt es sich um Speicher-zu-Speicher-Messungen, d.h. Delays durch Datenträger usw. spielen keine Rolle. Um die Performance bei Platte-zu-Platte-Anwendungen zu ermitteln, wurden außerdem Messungen mit dem Programm ftp durchgeführt. Hierbei wurde eine 24 MB große Datei zwischen den Maschinen übertragen. Dabei zeigte es sich, dass nicht die Netzwerkadapter der Engpass sind, sondern die angeschlossenen Platten. Die Transferzeiten hingen nur von den Transferraten der Platten ab. Die besten Ergebnisse erzielten wir mit RAID-Array-Platten an SUN-Systemen. Hier erreichten wir einen Durchsatz von bis zu 16 MB/s.

Mitte 1999 wurden die Messungen noch einmal für Gigabit Ethernet und LAN-Emulation wiederholt. Tabelle 2.3 zeigt das Ergebnis der Messungen. Die maximalen Durchsatzwerte liegen heute für die schnelleren Systeme bei 300-400 Mbit/s, stellenweise sogar darüber. Die theoretisch möglichen Durchsatzraten werden nach wie vor nicht erreicht. Hier muss die Taktrate noch weiter erhöht, bzw. müssen Engpässe im Betriebssystem beseitigt werden.

von \ nach		Gigabit Ethernet						LANE
	Typ	PC NT 4.0 PIII, 450 MHz	PC NT 4.0 PIII, 550 MHz	PC NT 4.0 PIII, 450 MHz	PC NT 4.0 PII, 333 MHz	SUN Solaris 2.7 2 Proz., 296 MHz		SUN Soloris 2.6 8 Proz., 248 MHz
Typ	GE-Karte	ACEnic	Intel PRO/1000	Packet Engines	ACEnic	SysKonnect	SUN Gigabit	622 Fore-ATM
PC NT 4.0 PIII, 450 MHz	ACEnic Adapter		339,49	337,19			429,85	387,69
PC NT 4.0 PIII, 550 MHz	Intel PRO/1000	298,35		332,79			388,64	363,63
PC NT 4.0 PIII, 450 MHz	Packet Engines	301,54	290,93				430,62	398,79
PC NT 4.0 PII, 333 MHz	ACEnic Adapter					30,20		280,67
SUN Solaris 2.7 2 Proz., 296 MHz	Sys-Konnect				108,57			184,15
	SUN Gigabit	275,08	312,49	307,43				378,34
SUN Sol. 2.6 8 Proz., 248 MHz	622 Fore ATM	195,35	193,59	205,28	176,34	94,81	240,50	

Tabelle 2.3: gemessene Durchsatzwerte für Gigabit Ethernet und LANE, Werte in Mbit/s

Die Ergebnisse der Durchsatzmessungen lassen sich wie folgt zusammenfassen:

- Die maximal erreichbaren Durchsatzwerte sind abhängig von der Entfernung (Delay), den Paketgrößen, den Buffer-Größen, der Durchsatzrate der angeschlossenen Platten und der maximalen Interrupt-Rate der Endsysteme.
- Bei Platte-Platte-Transfer bilden in der Regel die Transferraten der Platten den Engpass und nicht die Durchsatzraten der Netz-Interfaces.
- Heutige Endsysteme erreichen bei Classical IP und einer Paketgröße von 64 Kbytes den maximalen Durchsatz bei Speicher-zu-Speicher-Anwendungen.
- Heutige Endsysteme erreichen bei Gigabit Ethernet und LAN Emulation auf Grund der kleinen Paketgrößen und der damit verbundenen I/O-Interrupt-Rate bei Speicher-zu-Speicheranwendungen nicht den maximalen Durchsatz.

2.3 Kopplung von Supercomputern über HiPPI/ATM

Ein wesentliches Aufgabengebiet des Gigabit-Projektes war es, vorhandene Anwendungen so zu erweitern, dass sie gleichzeitig auf die ‚Compute-Nodes' der Systeme IBM SP2 (GMD) und eines der Cray-Systeme des FZJ (T90, T3E oder J90) zugreifen konnten. Demzufolge war es erforderlich, diese Systeme mit ausreichender Performance an das Gigabit Testbed anzuschließen.

Im Verlaufe des Projektes wurden verschieden Alternativen der Anbindung diskutiert und getestet.

Zum Zeitpunkt der Projektbeantragung wurden von Cray Research und IBM ATM-Interfaces mit OC-12 oder höheren Geschwindigkeiten für T3E bzw. SP2 in Aussicht gestellt.

Für T3E-Systeme der Firma SGI/Cray konnten diese jedoch nicht zur Verfügung gestellt werden. Die Ursache liegt in einer architekturbedingten Begrenzung der Interruptrate der T3E, die hohe Datenraten nur bei großer Blockung zulässt. Bei Nutzung von IP über ATM ist die maximale MTU von 9180 Byte der begrenzende Faktor. Als Ausweg lies sich die potentiell große Blockung der HiPPI-Technik ausnutzen. Messungen innerhalb des Cray-Komplexes im ZAM ergaben, dass sich mit der T3E bei TCP/IP über HiPPI (MTU 65280) Transferraten bis zu 430 Mbit/s, bei Nutzung des raw-HiPPI-Protokolls und 1 MByte großen Blöcken bis zu 540 Mbit/s erzielen lassen.

Für die Nutzung von HiPPI waren drei Alternativen denkbar:

1. HiPPI – Tunneling via ATM. Ascend bot zur Projektlaufzeit einen Router (GRF400) an, der über eine solche Funktionalität verfügen sollte. Konkrete Nachfragen und Tests der Uni Stuttgart, die einen solchen Router für anderweitige Tests beschafft hatte, ergaben jedoch, dass das zur Verfügung stehende Release keine Datenblöcke über 9180 Byte unterstützt. Zudem war der Einstandspreis mit ca. 150 TDM recht hoch. Insgesamt erschien daher eine Beschaffung nicht sinnvoll.

2. Einsatz eines HiPPI – SoNet Extenders von Essential. Hierzu hätte auf SDH-Ebene ein OC-12 Kanal reserviert werden müssen, der eine anderweitige Nutzung der Gigabitstrecke in Phase 1 des Projektes verhindert hätte. In der Phase 2 hätte die OC-48 Leitung in 4 OC-12 Leitungen aufgespalten und eine davon für diese Verbindung reserviert werden müssen. Auch dies erschien nicht sinnvoll.

3. Einsatz einer spezialisierten Workstation als HiPPI–ATM Proxy Server bzw. Application-Gateway, der die Konvertierung der Blockgröße durchführt. Cray Research hat den Einsatz einer SGI-Octane als Router zwischen HiPPI und ATM untersucht und inzwischen als strategisches Produkt zur Verfügung gestellt.

Aus technischer und finanzieller Sicht erschien die Variante 3 den größten Erfolg zu versprechen. Dies insbesondere, da hier auch mit der Unterstützung der Firma SGI/Cray Research gerechnet werden konnte. Ferner versprach Essential für seine EC-340-SF PCI Short Serial HiPPI NIC Transferraten von bis zu 100 MB/s sustained auf Netzwerkseite und bis zu 132 MB/s zum PCI-Bus hin. Mit vorhandenen Sbus basierten SUN Workstations wurden TCP/IP Datendurchsätze von ca. 60 MB/s bei einer Latenz von 0,12 msec über ATM OC-12 erzielt. Der Durchsatz war somit etwa 10 % höher als der maximal von der T3E zu erwartende Datenstrom (430 Mbit/s, s.o.). Die Latenz war deutlich geringer als die des IP-Stacks der T3E, die bei ca. 3 msec lag. SUN und/oder SGI Lösungen würden somit keinen Kommunikationsengpass darstellen.

Im Forschungszentrum Jülich wurde daher im Rahmen eines Beta-Tests mit der Firma SGI/Cray eine ATM-to-HiPPI Gateway Lösung über eine SGI-O200 Workstation installiert und auf ihre Nutzbarkeit für das Gigabit Testbed West Projekt hin untersucht.

Leistungstests mit Sun-Workstations bei der Etablierung der GMD-FZJ Gigabit-Strecke zeigten einen hohen Datendurchsatz der SUN-Plattform (bis zu 530 Mbit/s). Daher wurde, um einerseits einen Leistungsvergleich mit unterschiedlicher Hardware (SUN, SGI) durchführen zu können und andererseits unabhängig zu sein von herstellerbedingten Entwicklungs- und Lieferverzögerungen, in Jülich zur zweiten Cray-T3E eine Lösung über eine SUN-Ultra/60 als Gateway-System getestet. Beide Lösungen arbeiteten nahezu gleich gut, wobei die SUN-Variante eine etwa 10 % höhere Durchsatzrate lieferte.

In der SP2 sind in 8 Knoten ATM OC-3 Interfaces installiert. Generell konnten für die SP2-Anbindung ebenso die drei folgenden Konfigurationen genutzt werden.

1. Einsatz von mehreren OC-3 Interfaces. Nachteilig war hier, dass die Komplexität der von Pallas zu entwickelnden Kommunikationsbibliothek MPI-2 zugenommen hätte, was zu einem geringeren Durchsatz für die Anwendungen geführt hätte. Vor allem ist hier aber die Bandbreite einer Punkt-zu-Punkt Verbindung auf 155 Mbit/s begrenzt.

2. Einsatz eines PCI-basierten SMP-SP2-Knotens mit einem ATM OC-12 Interface. Derartige Knoten wurden von IBM für 1998 angekündigt, waren aber für die Nutzung im Rahmen dieses Projektes sehr teuer.

3. Einsatz eines IBM High Performance Gateway Nodes (HPGN). Dieser entspricht einem GRF400 mit einer Verbindung zum SP2-internen High-Performance Switch (HPS).

Aus technischer Sicht erschienen die Lösungen 2. und 3. vielversprechend. Alle zwei Alternativen waren kompatibel mit der Application-Gateway Anbindung der T3E. Aus Symmetrie, Komplexitäts- und Kostengründen wurde jedoch auch hier eine SUN-basierte Lösung entsprechend der Cray-Anbindung gewählt. Da bereits eine SUN mit ATM OC-12 Interface, sowie ein parallel HiPPI-Interface für die SP2 vorhanden waren, musste lediglich ein PCI serial-HiPPI und ein HiPPI-Modem (parallel nach serial) beschafft werden.

Es wurde daher parallel zu den Arbeiten in Jülich in der GMD ein entsprechender Zugang (HiPPI-to-ATM Gateway über SUN) zur IBM-SP2 realisiert.

Durch die HiPPI-to-ATM Gateway Lösung (siehe Abbildung 2.5) konnte ein erhöhter Durchsatz zwischen beiden Supercomputern (IBM ⇔ Cray) durch Etablierung einer durchgehenden Verbindung mit einer MTU von 60000 Byte unter Ausnutzung des Path-MTU-Discovery-Algorithmus erreicht werden. (Größere MTU Werte bis maximal 65280 Byte, wie für HiPPI im Standard vorgesehen, führten zu intermittierenden Fehlern, die auch mit Unterstützung von SGI/Cray, Essential und SUN nicht gelöst werden konnten). Verbindungstests zwischen der IBM-SP2 und den Cray-T3E Systemen über das Gigabit Testbed West führten zu Übertragungsraten von maximal 370 Mbit/s.

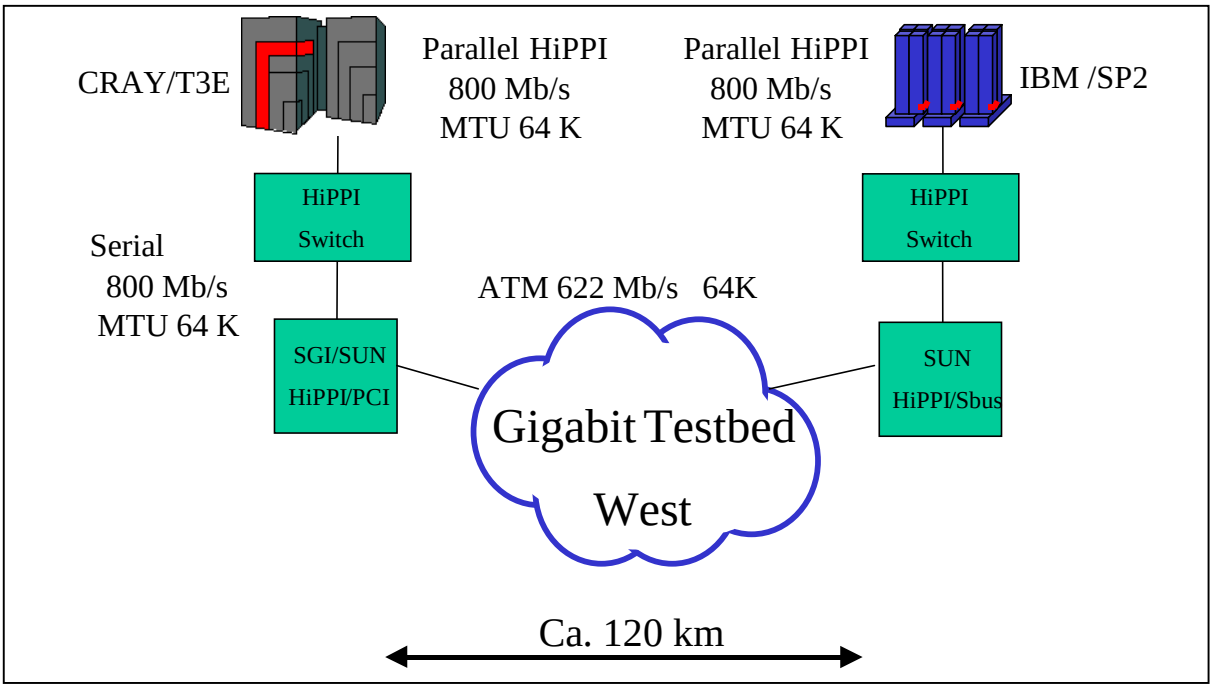


Abb. 2.5: ATM-to-HiPPI Gateway Lösung

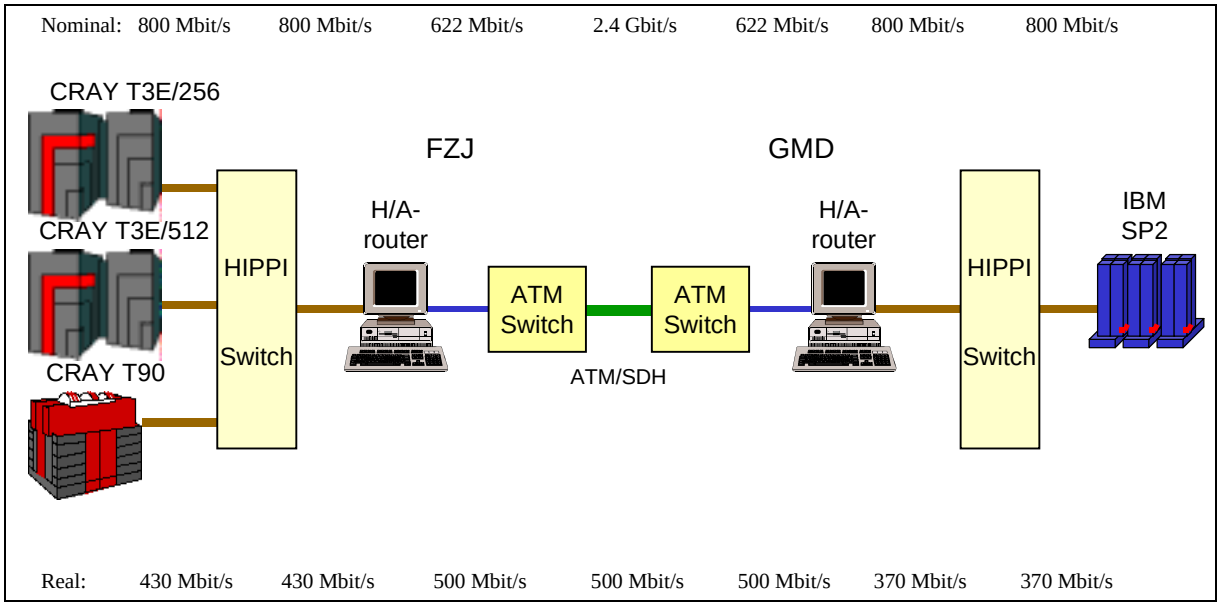


Abb. 2.6: Maximale und erreichte Durchsatzraten auf Teilstrecken

Eine Graphik über die nominelle und bisher erreichte Übertragungsrate auf den einzelnen Teilstrecken zeigt die Abbildung 2.6. Die endgültig maximal erzielten Durchsatzraten, sowie die derzeitige Konfiguration sind in der Abbildung 2.7 wiedergegeben.

In weiteren Projektschritten wurden auch die weiteren CRAY Systeme des Forschungszentrums Jülich in das Gigabit Testbed integriert, indem die HiPPI Produktions-Interfaces dieser Systeme auch für das Gigabit Testbed freigegeben wurden. Insofern konnten auch die CRAY-T90 und die beiden CRAY-J90 Systeme mit hohem Durchsatz von den Gigabit-Anwendungen genutzt werden.

	GMD			KHM	FZJ			
from / to	SP2	SUN	SGI (12)	SGI (2)	SUN U 60 (2)	SGI	CRAY (256)	CRAY (512)
SP2		310	300		270	310	280	290
SUN	330		450	490	460	460	360	450
SGI (12 proc)		470		470	430	330		
SGI (2 proc)		530	540					
SUN U 60 (2 proc)	300	530	520			540	370	420
SGI O200	310	400	410		410		310	320
CRAY (256 proc)	210	370			430	380		
CRAY (512 proc)	280	280			280	380		
CRAY (256,HIPPI)								340
CRAY (512,HIPPI)							385	

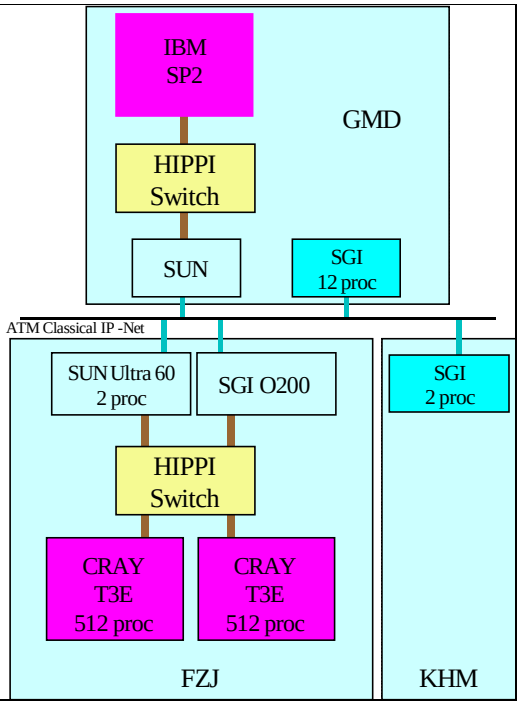


Abb. 2.7: Konfiguration und erreichte Durchsatzraten zwischen Endsystemen

Die in der ersten Projektphase testweise installierten 155 Mbit/s ATM-Interfaces der beiden CRAY T3E Systeme wurden aus der Konfiguration wieder entfernt, da sie zeitweise zu Shutdown-Problemen auf diesen System führten (Treiber ließ sich manchmal nicht stoppen). Es wurde diesbezüglich keine Fehlersuche durchgeführt, da mit den HiPPI-to-ATM-Gateways bereits ein leistungsfähigerer Zugang zur Verfügung stand.

2.4 Überlastmessungen

Im Laufe des Projektes wurden Überlastmessungen an unterschiedlichen Modulen von ATM-Switches durchgeführt. Ziel war es herauszufinden, wie sich die Netzkomponenten im Überlastfall verhalten. Hierzu gehört insbesondere die Auswirkung von Überlast auf die Quality-of-Service-Merkmale von ATM-Verbindungen. Eine der Fragen war: Wie verhalten sich UBR- und CBR-Verbindungen im Überlastfall?

Die Überlast wurde mit einem ATM-Analyser generiert, der über ein 155 Mbit/s und ein 622 Mbit/s-Interface verfügte und Zellen mit der vollen Bandbreite schicken konnte. Auf diese Weise ließ sich gezielt Überlast auf 155 Mbit/s und 622 Mbit/s-Leitungen generieren. Die Untersuchungen ergaben, dass CBR-Verkehr, wie erwartet, nicht gestört wird. Bei UBR-Verkehr war das Ergebnis abhängig von den verwendeten Netzmodulen auf den ATM-Switchen. Bei älteren Netzmodulen, die kein Weighted Fair Queueing (WFQ) unterstützen, wurden kleinere UBR-Datenströme durch einen großen Störstrom gestört. WFQ bedeutet, dass alle Datenströme gleich behandelt werden, d.h. bei *n* Datenströmen bekommt jeder Datenstrom maximal ein *n*-tel der Bandbreite. Hat man wenige kleine Datenströme und einen großen Störstrom, so wird nur der Störstrom zurückgeregelt, solange die kleinen Datenströme ihr Kontingent nicht ausschöpfen.

Schwieriger war die Erzeugung einer Überlast auf der 2,4 Gbit/s-Leitung zwischen der GMD und dem FZJ, da ein Analyser mit einem 2,4 Gbit/s-Interface nicht zur Verfügung stand. Abbildung 2.8 zeigt wie das Problem gelöst wurde. Am Eingang des ATM-Switches wurde der 622 Mbit/s Störstrom mittels einer Point to Multipoint ATM-Verbindung vervierfacht. An unterschiedlichen Messpunkten wurden die Zellverlustraten gemessen.

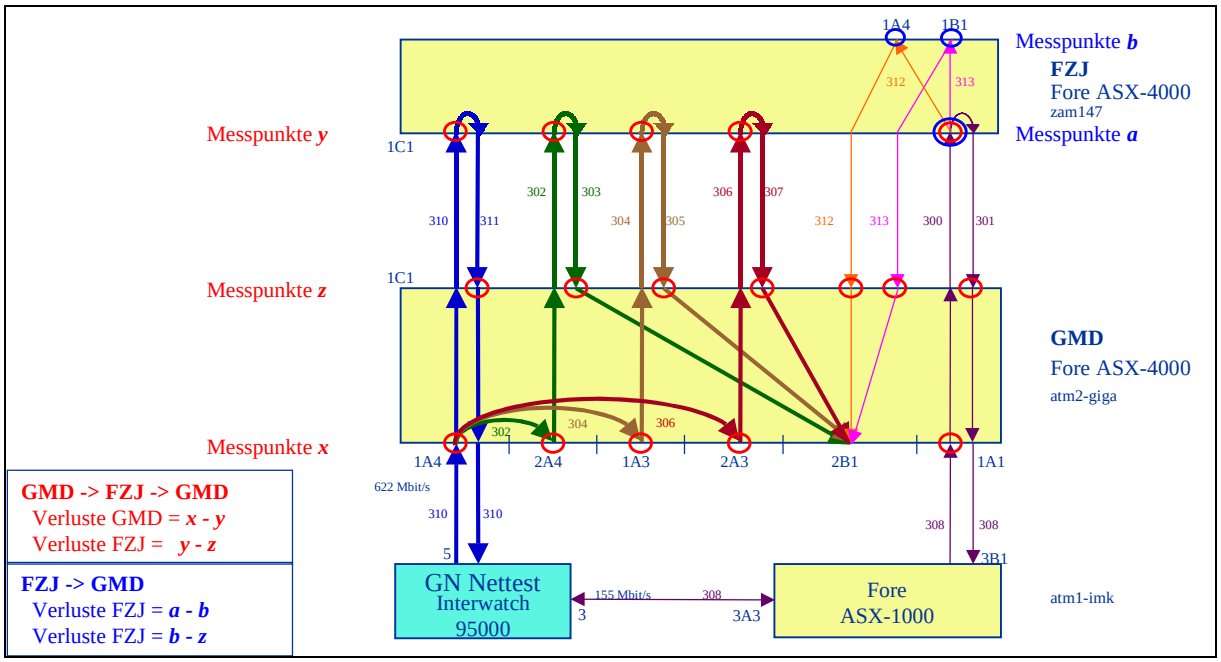


Abb. 2.8: Testszenario für Überlastmessungen

Die Untersuchungen ergaben, dass die eingesetzten 2,4 Gbit/s-Module WFQ nicht beherrschen. Gleichgroße UBR-Datenströme wurden unterschiedlich gestört. CBR-Daten wurden korrekt behandelt. Inzwischen hat FORE eine neue Generation von 2,4 Gbit/s Netzmodulen herausgebracht, die leider aus Kostengründen nicht untersucht werden konnten.

Als Fazit lässt sich festhalten, dass neuere ATM-Switches über ein verbessertes Traffic Management verfügen und die geforderten Verkehrsparameter einhalten.

2.5 Zusammenfassung

Im Teilprojekt GIGAnet des Gigabit Testbed West wurden die einzelnen Teilstrecken des Testbeds aufgebaut, in Betrieb genommen und überwacht. Für die Anwender wurden unterschiedliche IP-Netze konfiguriert, die für ihre Anwendungen optimiert waren. Umfangreiche Durchsatzmessungen und Tuning-Maßnahmen zur Bestimmung der maximalen TCP-Übertragungsraten wurden durchgeführt, um eine optimale Kopplung der Supercomputer in der GMD und im FZJ über HiPPI/ATM-Gateways zu erreichen. Mit Hilfe von Überlastmessungen wurde das Netzverhalten im Grenzbereich untersucht. Die eingesetzte ATM-Technik erwies sich als stabil und zuverlässig. Leider wurde die direkte Anbindung von Endsystemen insbesondere im Supercomputerbereich durch mangelnde Unterstützung von Seiten der Hersteller den Erwartungen nicht gerecht.

3 Methoden- und Werkzeugunterstützung, Software-Beratung

Beteiligte Partner: *Zentralinstitut für Angewandte Mathematik (ZAM/FZJ)*
Institut für Algorithmen und Wissenschaftliches Rechnen (SCAI/GMD)

Ansprechpartner: *Dr. Thomas Eickermann (ZAM/FZJ)), Helmut Grund (SCAI/GMD)*
Weitere Beteiligte: *Dr. Alfred Arnold (ZAM/FZJ), Dr. Jörg Henrichs (Pallas GmbH), Anke Häming (ZAM/FZJ), Dr. Thomas Kentemich (Pallas GmbH), Axel Klier (SCAI/GMD), Alexander Supalov (Pallas GmbH), Dr. Roland Völpel (SCAI/GMD)*

Ziel dieses Teilprojektes war es, den Anwendungen die benötigten Werkzeuge für die Nutzung des Metacomputers zur Verfügung zu stellen und Hilfestellung beim Einsatz dieser Werkzeuge zu leisten. Als wichtigstes Werkzeug wurde bei der Vorbereitung des Projektes eine metacomputing-fähige MPI-Implementierung identifiziert. Damit sollte den Anwendungsentwicklern die transparente Nutzung des Metacomputers über die standardisierte MPI-2-Programmierschnittstelle ermöglicht werden. Da zu Beginn des Projektes keine MPI-Implementierung existierte, die diese Anforderungen erfüllte, haben das Forschungszentrum Jülich und die GMD gemeinsam bei der Pallas GmbH eine Entwicklung in Auftrag gegeben. Die Entwicklung dieser *MetaMPI*-Bibliothek wurde aus Mitteln der beiden Zentren finanziert. Im Rahmen des Gigabit Projektes sollte lediglich die Evaluierung erfolgen.

Als Hauptaufgabe ergab sich damit, als Schnittstelle zwischen dem GIGAnet-Arbeitspaket, den Anwendungen und der Pallas GmbH zu fungieren, um Anforderungen und Fehlermeldungen der Anwender zu kanalisieren und weiterzuleiten. Weitere Aufgaben stellten sich im Verlauf des Projektes. So mußte eine Übergangslösung bis zur Fertigstellung des ersten MetaMPI-Prototyps im Herbst 1998 geschaffen werden. Dazu wurde die im Rechenzentrum der Universität Stuttgart (RUS) entwickelte PACX-MPI Bibliothek eingesetzt, die zu diesem Zeitpunkt eine kleine aber zunächst ausreichende Teilmenge des MPI-1 Standards bereitstellte. Weitere Arbeiten waren die Portierung von MetaMPI auf zusätzliche Hardware-Plattformen, die Entwicklung von ergänzenden Werkzeugen wie einer Bibliothek für die Instrumentierung von MetaMPI-Anwendungen für das Performance-Analysewerkzeug VAMPIR sowie einer graphischen Oberfläche zur Erstellung von MetaMPI-Konfigurationen.

3.1 Einsatz von PACX-MPI

PACX wurde 1995 im RUS ursprünglich für die Kopplung einer Intel Paragon mit einer CRAY YMP entwickelt und seitdem kontinuierlich ausgebaut. Die aktuelle Version unterstützt die Kopplung einer beliebigen Zahl von Rechnern mit dem vollen MPI-1 Standard. Zu Beginn des Gigabit Projektes haben wir die damalige Version 2.0, die eine Kopplung von zwei CRAY T3E Systemen unterstützt, eingesetzt. Im Rahmen des Gigabit Testbeds führten wir folgende Modifikationen an PACX durch:

- Die Bibliothek wurde auf die IBM SP2 portiert. PACX nutzt je einen Knoten für eingehende und ausgehende externe Nachrichten. Da die SP2 der GMD nur über einen HiPPI-Knoten verfügt, wurde die Option geschaffen, die gesamte Kommunikation über einen Knoten abzuwickeln. So konnte das leistungsstarke HiPPI-Interface in beiden Richtungen genutzt werden.

- Die externe TCP/IP-Kommunikation wurde für latenzarme Netzwerke mit hoher Bandbreite optimiert.

3.2 Message Passing Bibliothek MetaMPI

Ziel bei der Entwicklung von MetaMPI war es, eine MPI-Bibliothek für verteilte Anwendungen zu erstellen. Diese sollte

- einen gemeinsamen Kommunikator MPI_COMM_WORLD für verteilte Anwendungen zur Verfügung stellen, und damit eine für den Anwendungsentwickler transparente Kopplung der verteilten Rechner ermöglichen,
- den vollen MPI-1 Standard und ausgewählte Teile von MPI-2 enthalten,
- alle im Testbed relevanten Plattformen, also CRAY T3E, CRAY T90, IBM SP2, sowie SMP-Rechner von SUN und SGI unterstützen,
- hohe Kommunikationsleistung sowohl innerhalb als auch zwischen den zu koppelnden Rechnern erbringen und
- vorhandene Netze durch flexible Konfigurierbarkeit optimal nutzen können.

3.2.1 Design und Implementierung

Um die Performance-Ziele zu erreichen, nutzt MetaMPI innerhalb der Rechner jeweils die „native“ Kommunikationsmethode und TCP/IP Socket-Verbindungen zwischen den Rechnern. Die Kommunikation zwischen den Rechnern wird über dedizierte Prozessoren, sogenannte Router-Knoten, abgewickelt - ein Ansatz, der auch von PACX-MPI verfolgt wird. Bei MetaMPI ist zusätzlich die Anzahl der Router und die Anzahl der Socket-Verbindungen pro Router frei konfigurierbar. Außerdem können – sofern vorhanden – auch mehrere Netzwerkadapter genutzt werden.
Alternativ könnte man auf Router verzichten, indem zwischen je zwei Prozessoren auf verschiedenen Rechnern eine Socket-Verbindung aufgebaut wird, sobald sie kommunizieren müssen. Ein Nachteil dieser, z.B. in MPICH-G, der MPI-Implementierung des globus-Projektes, genutzten Technik ist, dass die Anzahl der offenen Socket-Verbindungen quadratisch mit der Prozessorzahl zunehmen kann, was insbesondere bei Rechnern mit verteiltem Betriebssystem wie der CRAY T3E ein Problem darstellt. Ein weiterer Vorteil des Router-Ansatzes ist die Möglichkeit, auf einzelnen oder wenigen Knoten vorhandene Netzwerkanbindungen optimal für die externe Kommunikation nutzen zu können. Ein Beispiel hierfür ist der HiPPI-Knoten der IBM SP2 in der GMD. Abbildung 3.1 zeigt eine Beispielkonfiguration.

Basis der Entwicklung von MetaMPI ist die public-domain MPI-Implementierung MPICH in der Version 1.1. MPICH ist eine vollständige MPI-1 Implementierung, die für viele Supercomputer-Plattformen sowie SMP-Rechner und Workstation-Cluster verfügbar ist. Die Unterstützung einer solchen Vielzahl unterschiedlicher Plattformen wird dadurch möglich, dass die architekturabhängigen Teile der Bibliothek in sogenannten Devices gekapselt sind, die über eine definierte Schnittstelle (das abstract device interface - ADI) mit den höheren Schichten der Bibliothek verbunden sind. MetaMPI ist in der Lage, mehrere dieser Devices simultan zu nutzen. Für die Kommunikation *innerhalb* der Rechner wird das auf die Architektur optimierte native Device genutzt, für die Kommunikation *zwischen* den Rechnern wurden zwei neue Devices entwickelt und implementiert. Das Gateway-Device stellt die Verbindung zwischen Anwendungs- und Router-Knoten her, es nutzt dazu das native Device. Das Tunnel-Device verbindet die Router-Knoten der verschiedenen Rechner durch Socket-

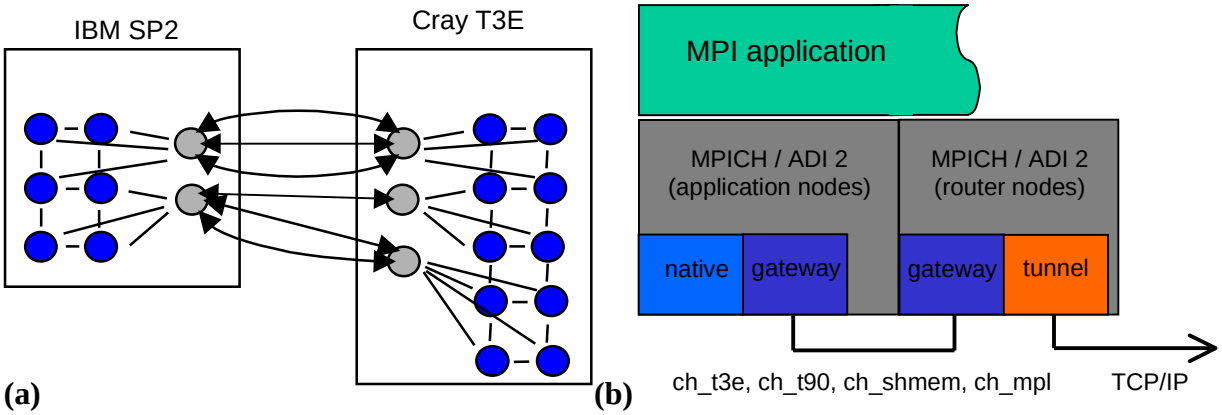


Abb. 3.1 a) Beispielkonfiguration einer SP2 – T3E Kopplung mit MetaMPI: für die MPI-Anwendung sind nur die blau gekennzeichneten Anwendungsknoten sichtbar. Die grau eingezeichneten Router-Knoten sind transparent. Die Anwendungsknoten eines Rechners kommunizieren untereinander und mit den Router-Knoten des selben Rechners über das native Kommunikationsprotokoll. Die Kommunikation zwischen den Rechnern wird von den Router-Knoten, vermittelt über TCP/IP Socket-Verbindungen (Doppelpfeile), ausgeführt.
b) Struktur einer MetaMPI-Anwendung. Auf den Anwendungsknoten sind das native Device für die interne Kommunikation sowie das Gateway-Device für externe Nachrichten aktiv. Die Router-Knoten vermitteln ein- und ausgehende Nachrichten, wobei nach innen das Gateway- und nach außen das Tunnel-Device genutzt wird.

Verbindungen miteinander. Abbildung 3.1 zeigt schematisch diesen Aufbau. Die Entwicklung von MetaMPI durch Pallas baut auf einer Modifikation von MPICH durch Joachim Worringen (RWTH Aachen) auf, die eine transparente Kopplung von SUN SMP-Rechnern erlaubt.

Die Tests mit ersten Prototypen von MetaMPI zeigten, dass die Erstellung komplexer Konfigurationen, wie sie z.B. bei der gekoppelten Klima- und Wettersimulation notwendig ist, erhebliche Anforderungen an die Anwender stellt. So werden Kenntnisse über die Details der Netzwerkkonfiguration (z.B. IP-Adressen in verschiedenen physikalischen und logischen Netzen) benötigt. Um die Anwender von dieser Aufgabe zu entlasten, wurde von Pallas ein Konzept zur Verbesserung der Benutzerfreundlichkeit von MetaMPI erstellt und umgesetzt. Dieses sieht vor, dass die zum Start einer MetaMPI-Anwendung benötigten Informationen über die genutzten Rechner und Netze nicht mehr aus einer Konfigurationsdatei, sondern aus einer relationalen Datenbank gelesen werden. Das ZAM entwickelte dazu ein Werkzeug mit graphischer Benutzeroberfläche (MetaConf), das eine komfortable Erstellung von Konfigurationen ohne Wissen über die zugrundeliegenden Netze und IP-Adressen ermöglicht. Abbildung 3.2 zeigt einen Bildschirmabzug von MetaConf während der Erstellung der in Abbildung 3.1 dargestellten Konfiguration.

Der erste Prototyp von MetaMPI, der eine Kopplung von CRAY T3E und IBM SP2 mit voller MPI-1.2 Funktionalität ermöglicht, wurde von Pallas im Februar 1999 ausgeliefert. Im ersten Halbjahr 1999 wurde die Funktionalität dann schrittweise um ausgewählte Eigenschaften des neuen MPI-2 Standards erweitert, die für Metacomputing-Anwendungen besonders wichtig sind. Dazu zählen vor allem die Funktionen des Kapitels 5 „Process Creation and Management“ des MPI-2 Dokuments. Sie erweitern das statische Prozess-Modell des MPI-1 Standards, indem sie die Kopplung unabhängig voneinander gestarteter MPI-Anwendungen ermöglichen (process attachment). Außerdem kann eine Anwendung selbst neue Prozesse starten und mit ihnen kommunizieren (process spawning). Dadurch können Teile einer verteilten Simulation zeitversetzt gestartet werden oder eine Steering-Komponente kann sich dynamisch an eine Simulation an- und abkoppeln.

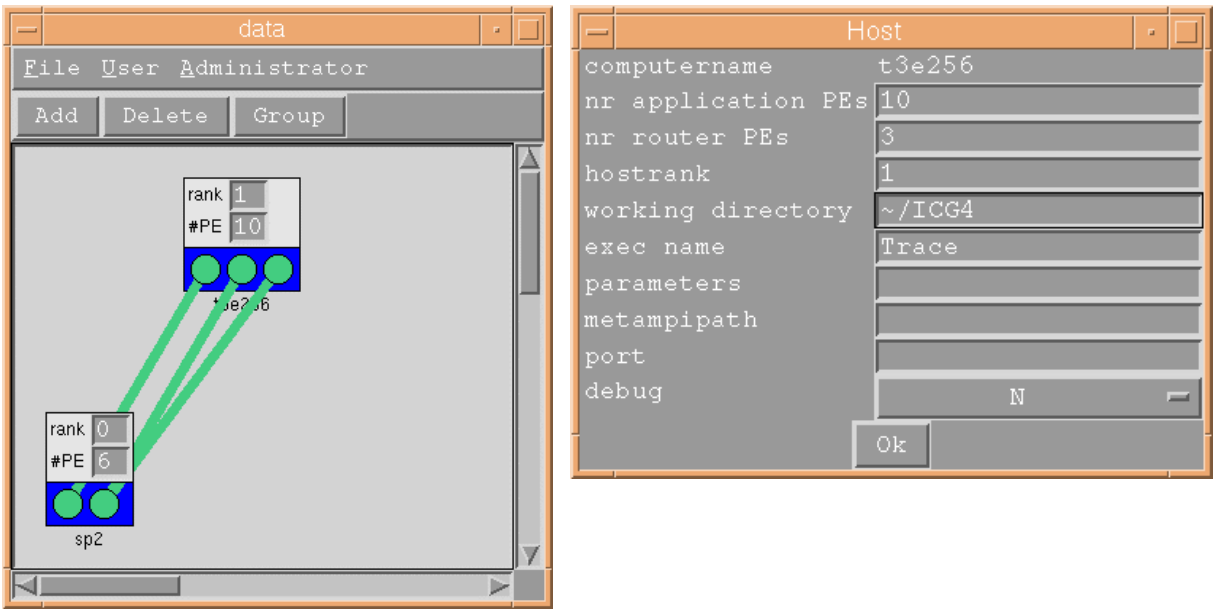


Abb 3.2: Bildschirmabzug von MetaConf, einem graphischen Werkzeug zur Erstellung von Konfigurationen für MetaMPI. Die Topologie des Metacomputers (die Rechner und die Verbindungen zwischen ihnen) werden in dem Hauptfenster (links) festgelegt. Details, wie die Namen und Pfade der auszuführenden Programme, Kommandozeilenparameter und zu nutzende Netze werden in weiteren Fenstern (rechts) festgelegt.

Solche erst zur Laufzeit miteinander verbundenen Anwendungsteile kommunizieren nur über sogenannte Interkommunikatoren. Um dennoch die für viele Anwendungen wichtigen kollektiven Operationen zu ermöglichen, wurden auch die „Extended Collective Operations“ (Kapitel 7) des MPI-2 Standards implementiert.

MetaMPI ist auf den Plattformen IBM SP2, CRAY T3E, CRAY T90 (Portierung durch ZAM), SUN – Solaris und SGI – IRIX (Portierung durch SCAI) verfügbar und erlaubt eine transparente Kopplung einer beliebigen Zahl von Rechnern dieser Typen.

3.2.2 Performance der Punkt-zu-Punkt Kommunikation

Bei der Evaluierung der Performance von MetaMPI konzentrierten wir uns zunächst auf die Punkt-zu-Punkt Kommunikation, da bei der MetaMPI zugrundeliegenden MPI-Implementierung MPICH die kollektive Kommunikation auf der Basis der Punkt-zu-Punkt Kommunikation realisiert ist. Die folgenden Tabellen zeigen die Resultate von Ping-Pong Messungen. In Tabelle 3.1 sind für Nachrichten innerhalb eines Rechners Latenz (halbe Roundtrip-Zeit einer 1 Byte Nachricht) und Bandbreite (für 2 MByte Nachrichten) mit denen der vom jeweiligen Hersteller ausgelieferten MPI-Implementierung verglichen. Mit Ausnahme der Bandbreite der T90 und der Latenz der SP2 zeigt sich MetaMPI den Hersteller-Implementierungen mindestens ebenbürtig. Bei den Cray Systemen wurde sogar eine signifikant geringere Latenz erreicht.

Für Nachrichten, die zwischen Rechnern ausgetauscht werden, werden die Werte mit denen eines TCP-Socket Ping-Pong Benchmarks verglichen. Der deutliche Verlust an Bandbreite ist durch das Router-Konzept bedingt. Jede Nachricht wird zunächst vom Anwendungs-Knoten über das native Protokoll zum Router-Knoten geschickt, von dort via TCP/IP zum Router des entfernten Rechnern und von dort weiter zum Zielknoten. Auf diese Weise addieren sich die

Hersteller-MPI – MetaMPI

Rechner	MPI-Impl	Latenz [usec]	Bandbreite [MB/sec]
T3E	Cray MPI	15.1	314.6
	MetaMPI	7.5	333.5
T90	Cray MPI	43.7	4612.6
	MetaMPI	24.7	2314.1
SP2	IBM MPI	48.8	46.3
	MetaMPI	97.7	45.9
Sun E5000	MetaMPI	28.0	103.0
Sun Ultra 60	MetaMPI	22.7	137.8

Tab 3.1: Mit einem MPI Ping-Pong Benchmark gemessene Latenz und Bandbreite der rechnerinternen Kommunikation für MetaMPI und (soweit vorhanden) der herstellereigenen MPI Implementierung.

Latenz: MetaMPI – TCP/IP

MetaMPI / **TCP/IP**	**E5000**	**T3E**	**T90**	**Ultra 60**
SP2	4976.8	5787.1	3659.3	4987.1
	561.2	3783.0	2656.2	1285.6
E5000		5549.9	4996.6	5056.7
		3533.6	2205.4	825.2
T3E		6402.5	5503.9	5819.3
		4521.5	3723.0	4336.6

Tab 3.2: Mit einem Benchmark für MetaMPI (obere Zeile) bzw. TCP/IP Sockets (untere Zeile) gemessene Latenz in Microsekunden der Kommunikation zwischen zwei Rechnern.

Bandbreite: MetaMPI – TCP/IP

MetaMPI / **TCP/IP** / **max. BW**	**E5000**	**T3E**	**T90**	**Ultra 60**
SP2	16.9	14.4	16.3	17.3
	39.3	28.0	26.3	37.4
	17.6	16.5	16.6	17.9
E5000		19.8	24.1	28.5
		31.4	35.0	59.7
		22.4	25.8	29.7
T3E		18.1	19.8	23.4
		33.9	30.1	35.2
		28.2	27.3	25.9

Tab 3.3: Mit einem MetaMPI (obere Zeile) bzw. TCP/IP Socket (mittlere Zeile) Benchmark gemessene Bandbreite in MByte/s der Kommunikation zwischen zwei Rechnern (für 2 MByte Nachrichten). In der dritten Zeile steht die Bandbreite, die sich ergibt, wenn man die Laufzeiten der rechnerinternen Kommunikation (Anwendungs-Knoten zu Router-Knoten) und der TCP/IP Kommunikation zwischen den Rechnern addiert.

Laufzeiten der 3 „Hops“ zur minimal möglichen Laufzeit. Die sich daraus ergebende maximal mögliche Bandbreite ist in der Tabelle unter „max. BW“ aufgetragen und erklärt zumindest teilweise die gemessenen Verluste. Die größere Latenz ist vor allem dadurch bedingt, dass die Router über Systemaufrufe (select) die Socket-Verbindungen überwachen. Ein solcher Aufruf dauert auf der CRAY T3E (unabhängig vom Ergebnis) mindestens 1 Millisekunde.

3.2.3 Performance der kollektiven Operationen

MetaMPI ist nicht für kollektive Operationen über Rechnergrenzen hinweg optimiert, sondern übernimmt diese Operationen unverändert von MPICH. Dabei werden Nachrichten über einen binären Baum auf die Knoten verteilt oder von ihnen eingesammelt. Unter Umständen werden Nachrichten mehrfach zwischen den Rechnern ausgetauscht. Wie oft dies geschieht, hängt von der Anzahl der Knoten und von der Knotennummer des „root“-Knotens ab.
Die von Thilo Kielmann (FU Amsterdam) entwickelte MagPIe-Bibliothek modifiziert die kollektiven Operationen so, dass unter optimaler Ausnutzung der Topologie jede Nachricht nur höchstens einmal zwischen zwei Rechnern ausgetauscht wird. MagPIe nutzt das Profiling-Interface von MPI und ist kompatibel zu MetaMPI. Durch Hinzubinden dieser Bibliothek lassen sich MetaMPI-Anwendungen, die kollektive Operationen über Rechnergrenzen hinweg ausführen, erheblich beschleunigen. Abbildung 3.3 demonstriert das am Beispiel einer Broadcast-Operation zwischen zwei CRAY T3E-Rechnern. Mit MagPIe dauert dieses Broadcast nicht länger als ein einfaches MPI-Send zwischen den Rechnern.

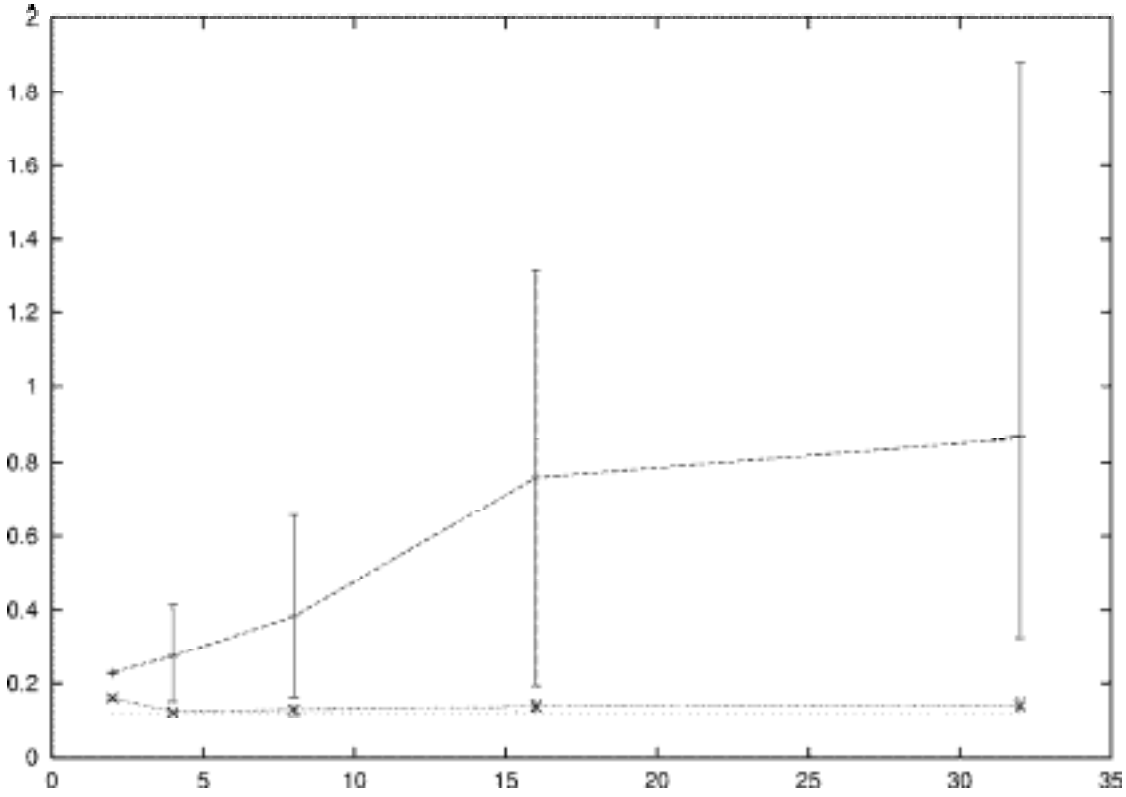


Abb 3.3: Laufzeit einer MPI_Bcast-Operation einer 2 MByte großen Nachricht auf zwei CRAY T3Es in Abhängigkeit von der Knotenzahl. Die Meßwerte auf der unteren Linie stammen von MetaMPI und MagPIe. Zum Vergleich ist gestrichelt die Laufzeit eines MPI_Send einer 2 MByte großen Nachricht zwischen den beiden CRAY T3Es eingetragen. Die Meßwerte auf der oberen Linie sind mit MetaMPI ohne MagPIe aufgenommen. Die Fehlerbalken zeigen die Schwankungen, die sich durch verschiedene root-Knoten des MPI_Bcast ergeben.

3.3 VAMPIR – Instrumentierung

Das im ZAM entwickelte VAMPIR ist ein Werkzeug zur Analyse des Laufzeit- und Kommunikationsverhaltens von MPI-Anwendungen. Eine wichtige Komponente hierin ist eine Wrapper-Bibliothek, die die MPI-Funktionsaufrufe umschließt und Daten über ausgetauschte Nachrichten sammelt. Um VAMPIR für die verteilten MPI-Anwendungen des

Projektes nutzen zu können, wurden sowohl eine PACX-MPI als auch eine MetaMPI-Version dieser Wrapper-Bibliothek implementiert. Darüber hinaus wurde VAMPIR um einige metacomputing-spezifische Eigenschaften erweitert. So lassen sich Knoten gruppieren, um die Zugehörigkeit zu den verschiedenen Rechnern darzustellen. Außerdem gibt es eine halbautomatische Korrektur der auf den Systemen unterschiedlichen Uhren-Stände.

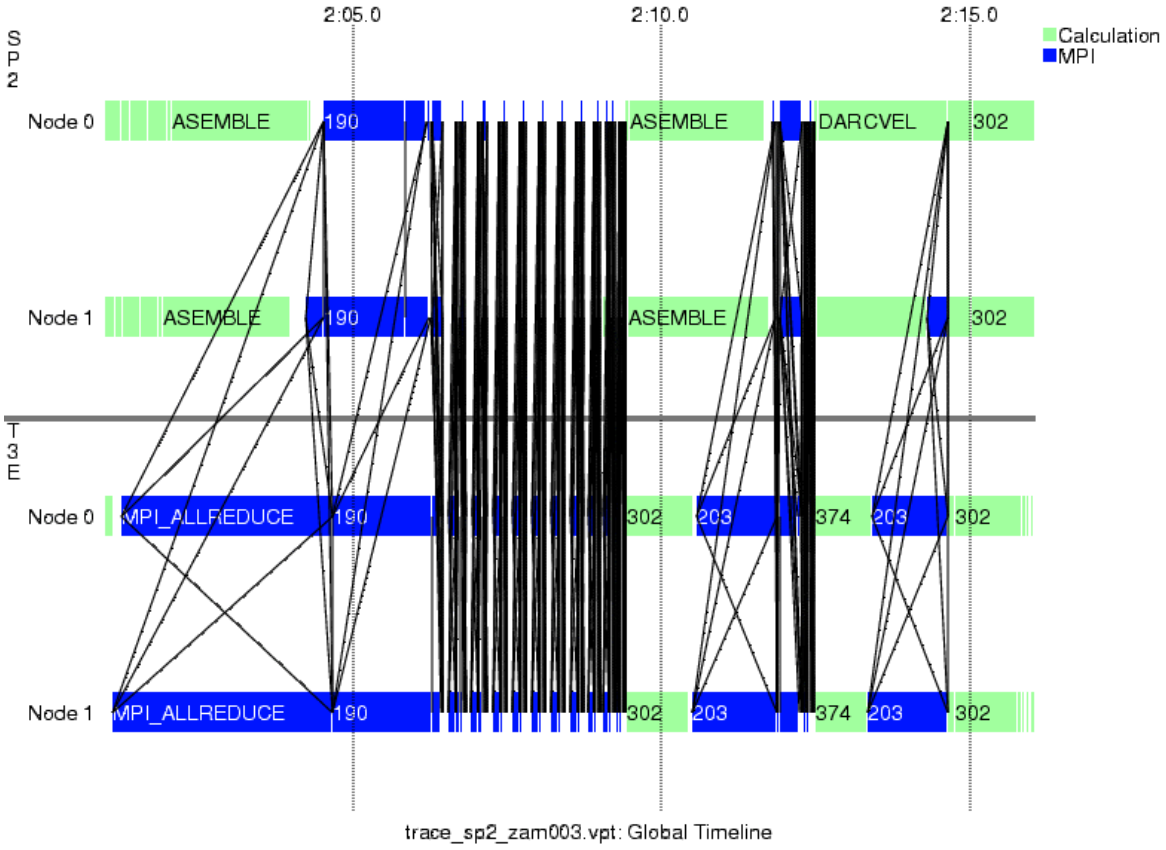


Abb. 31: Mit VAMPIR dargestellter Ausschnitt aus einem verteilten Lauf des Simulationsprogramms Trace (vgl. Kapitel 4) mit je zwei SP2- und T3E-Knoten. Dargestellt sind Berechnung und Kommunikation. Die Linien symbolisieren einzelne ausgetauschte Nachrichten.

3.4 Zusammenfassung

Im Lauf des Projektes hat sich die von der Pallas GmbH im Auftrag von Forschungszentrum Jülich und GMD entwickelte und implementierte MetaMPI-Bibliothek zu einem robusten und flexiblen Werkzeug entwickelt, das sowohl innerhalb als auch zwischen den gekoppelten Rechnern eine hohe Kommunikationssleistung erbringt. Von GMD und FZJ wurden dabei nicht nur – wie ursprünglich geplant – die Evaluierung von MetaMPI durchgeführt, sondern es wurden auch ergänzende Portierungsarbeiten geleistet und zusätzliche Werkzeuge wie MetaConf entwickelt.
Aufgrund des prototypischen Charakters der Anwendungen standen bei der Entwicklung von MetaMPI Performance und Stabilität im Vordergrund. Andere Aspekte, die in einem Produktionsszenario ebenfalls wichtig sind, wurden bewußt ausgeklammert. So verfügt MetaMPI zur Zeit über keine Schnittstellen zu Resource-Scheduling Systemen oder über konsistente Sicherheitsmechanismen. Entsprechende Erweiterungen sind im Rahmen zukünftiger Projekte geplant.

4 Schadstoffausbreitung im Boden

Beteiligte Partner: *Institut für Erdöl und organische Geochemie (ICG-4/FZJ)*
Zentralinstitut für Angewandte Mathematik (ZAM/FZJ)
Institut für Algorithmen und Wissenschaftliches Rechnen (SCAI/GMD)

Ansprechpartner: *Wolfgang Frings (ZAM/FZJ)*
Weitere Beteiligte: *Priv.-Doz. Dr. Harry Vereecken (ICG-4/FZJ),*
Horst Hardelauf (ICG-4/FZJ), Dr. Thorsten Graf (ICG-4/FZJ),
Dr. Thomas Eickermann (ZAM/FZJ)

Viele Probleme der Umweltforschung bedürfen einer integrierten Betrachtung von Boden und gesättigten Grundwasserleitern. Vor allem in der Grundwasserneubildungszone finden viele wichtige chemische und biologische Prozesse statt und Schadstoffe zeigen aufgrund der mit diesen Prozessen verbundenen Wechselwirkungen ein differenziertes Transportverhalten. Für die Simulation der Schadstoffausbreitung im Boden sind im Institut für Erdöl und organische Geochemie (ICG4) des Forschungszentrums Jülich zwei Programme entwickelt worden. Trace berechnet dabei die Strömung des Grundwassers, Partrace die Ausbreitung der Schadstoffe darin.

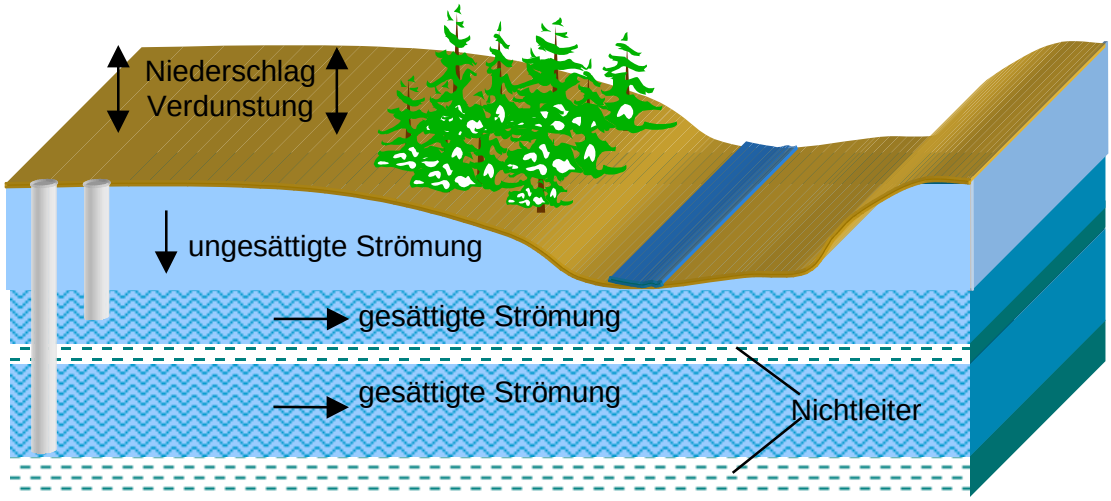


Abb. 4.1: Schema der Grundwasserströmung im Boden. Die einzelnen Strömungsschichten des Grundwassers sind durch Nichtleiter voneinander getrennt. Durch einen ungesättigten vertikalen Wasserfluß geschieht der Austausch mit der Oberfläche.

Im Rahmen des Gigabit Testbed West sind die beiden Programme, die auf je einem massiv-parallelen Rechner ablaufen, miteinander zu einer Metacomputing-Anwendung gekoppelt worden. Die beiden dabei genutzten Parallelrechner CRAY T3E und IBM SP2 sind räumlich getrennt, aber über die Gigabit-Leitung miteinander verbunden.
Zur direkten Überwachung der Simulation ist in AVS/Express eine Visualisierungs- und Steueroberfläche implementiert worden, mit der die Ergebnisse direkt angezeigt und die Parameter der Simulation geändert werden können.

4.1 Kopplung der Programme

Trace berechnet zu einem vorgegebenen Gebiet die Strömung des Grundwassers und liefert als Ergebnis ein Fließfeld, das die Geschwindigkeit und die Richtung der Strömung auf den Gitterpunkten beschreibt. Trace ist auch in der Lage, zeitabhängige Fließfelder zu berechnen.

Partrace benutzt die berechneten Fließfelder und untersucht das Verhalten von Schadstoffpartikeln, die in dieses Gebiet injiziert werden (bzw. hinein diffundieren). Bisher berechnete Trace ein stationäres Fließfeld, das in eine Datei geschrieben und von Partrace bei dessen Start eingelesen wurde.
In der gekoppelten Version beider Programme werden die Felder nicht mehr auf Dateien ausgelagert, sondern von Trace direkt an Partrace weitergegeben. Die Übergabe wird in jedem Simulationsschritt durchgeführt, so dass auch zeitabhängige Fließfelder für die Simulation genutzt werden können. Für die Verteilung der gekoppelten Metacomputing-Anwendung auf die beiden Rechner ist der Ansatz gewählt worden, jeweils ein Programm auf einem Rechner laufen zu lassen. Der zunächst naheliegende Ansatz, statt dessen das Simulationsgebiet auf die beteiligten Rechner aufzuteilen, ist dagegen nicht praktikabel. Dabei müssen zwar wesentlich kleinere Datenmengen übertragen werden (nur die Werte an den Randgebieten) diese aber sehr viel häufiger (ca. 60 mal pro Simulationsschritt). Ein Ergebnis des Vorläuferprojektes im Regionalen Testbed NRW, das der Vorbereitung des B-WINs diente, war, dass die große Latenz eines Weitverkehrsnetzes das Programm in nicht akzeptablem Ausmaß verlangsamt. Effizienter ist es, mehr Daten weniger häufig zu übertragen.

Für die Kopplung der beiden Programme ist die Speicherung der zu übertragenden Daten innerhalb der Programme wichtig. Sowohl Trace als auch Partrace werden parallel ausgeführt und verteilen die Daten auf die einzelnen Knoten. Die beiden folgenden Abschnitte stellen Trace und Partrace genauer vor und untersuchen die Speicherung der Daten.

4.1.1 Trace

Mit Hilfe von Trace kann die Strömung des Wasser in variabel gesättigten porösen Medien modelliert werden. Dazu wird die dreidimensionale generalisierte Richards-Gleichung unter Verwendung eines Finiten-Elementen-Verfahren gelöst. Die Zeitabhängigkeit wird durch ein finites Differenzschema diskretisiert. Verschiedene Randbedingungen können vorgegeben werden. Für die Beschreibung der Feuchtigkeitsänderung stehen verschiedene Ansätze zur Verfügung. Das Programm arbeitet mit heterogenen Parameterfeldern, eine Einbeziehung von Quell- und Senkentermen ist möglich.
Trace ist in Fortran 90 implementiert und benutzt eine dynamische Zeitschrittsteuerung. Die Länge der Zeitschritte wird je nach Konvergenzverhalten der Advektions-Dispersions-Gleichung in einem vorgegebenen Intervall vergrößert oder verkleinert. Die Rechenzeit für einen bestimmten Abschnitt der Simulationszeit ist daher von dieser Zeitschrittlänge abhängig. Zusätzlich variiert auch die Zahl der Iterationen, die pro Zeitschritt ausgeführt werden müssen, so dass eine a priori Vorhersage der Rechenzeit nicht möglich ist.

Trace ist in der Lage, bei Simulationsrechnungen bis zu 10^8 Finite-Elemente-Knoten zu verwenden. Die Parallelisierung beruht auf einem parallelen Konjugierte Gradienten Verfahren, das das gesamte Gitter in einzelne überlappende Teilgitter zerlegt. Jeder Knoten hält jeweils eines dieser Teilgitter im Hauptspeicher. Für die Kommunikation wird MPI verwendet.
Für jeden FE-Knoten müssen für die Simulationsberechnung ca. 50 Double-Werte gespeichert werden. Unter anderen werden der Geschwindigkeitsvektor, der Druck und die Parameter der Bodeneigenschaften an der Position im Simulationsgebiet benötigt.

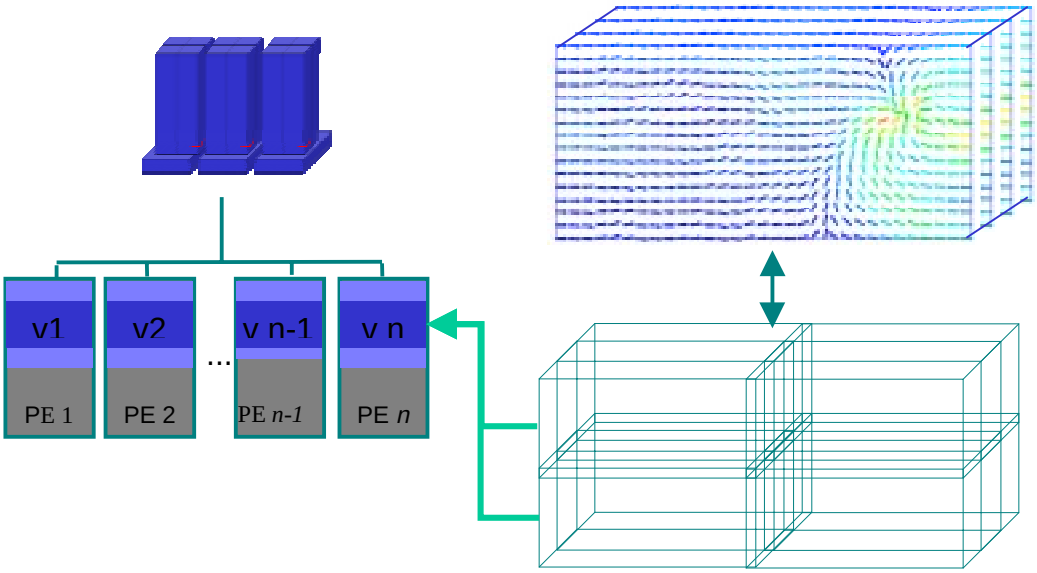


Abb. 4.2: Datenaufteilung bei Trace. Das Geschwindigkeitsfeld wird in einzelne Teilgebiete zerlegt und auf die einzelnen Prozessorelemente verteilt. Die Teilgebiete hängen räumlich zusammen. Die Randgebiete werden während der Simulationsrechnung untereinander ausgetauscht.

4.1.2 Partrace

PARTRACE modelliert den Transport gelöster Stoffe bei einem vorgegebenen Fließfeld. Dazu wird die Particle Tracking (Random Walk) Methode verwendet. Diese Methode beruht auf einer Analogie zwischen der Advektions-Dispersionsgleichung und der partiellen Differentialgleichungen aus der statistischen Physik (Fokker-Planck-Gleichung). Die gelöste Stoffmenge wird durch eine große Anzahl von Partikeln abgebildet, welche sich individuell advektiv und dispersiv bewegen. Die advektive Bewegung wird aus dem von Trace berechneten Geschwindigkeitsfeld ermittelt. Die dispersiven Schritte erhält man durch einem zufälligen Vektor mit Mittelwert 0 und einer Varianz, abhängig vom Dispersionstensor. Zur Berücksichtigung der Wechselwirkung können lineare und nichtlineare Isothermen sowohl im Gleichgewicht als auch kinetisch eingebunden werden. Eine Einbeziehung thermodynamischer Modelle ist derzeit in Arbeit.
Partrace ist in C++ implementiert und benutzt eine feste Zeitschrittsteuerung. In einem Zeitschritt wird die Bewegung eines jeden Partikels berechnet. Die Rechenzyklen, die pro Zeitschritt und Partikel durchgeführt werden müssen, sind stets gleich aufwendig. Solange ein Partikel nicht adsorbiert oder aus dem Simulationsgebiet gewandert ist, ist die Rechenzeit pro Partikel konstant. Die Rechenzeit für einen bestimmten Simulationszeitraum ist damit fest und nur von der Leistung der Maschine abhängig. Da die Partikel gleichmäßig auf alle Prozessoren verteilt werden und nur am Ende eines Zeitschrittes eine Kommunikation zwischen den Prozessoren nötig ist, ist die Last optimal balanciert und eine hohe Skalierbarkeit bei Hinzunahme weiterer Prozessoren gegeben.
Partrace benötigt für die Speicherung der Daten eines Partikels 40 Byte. Da die Partikel auf die Speicher der Prozessorelemente verteilt wird, können Simulationsläufe mit bis zu 10^9 Partikeln gerechnet werden.
Der Speicherbereich für das Geschwindigkeitsfeld wird auch auf die einzelnen Prozessoren aufgeteilt. Der Zugriff erfolgt während der Simulationsrechnung nur lesend. Daher werden für die Lesezugriffe die einseitigen Leseoperationen von shmem, einer Low-level Kommunikationsbibliothek auf der CRAY T3E, benutzt. Für die restliche Kommunikation wird MPI verwendet.

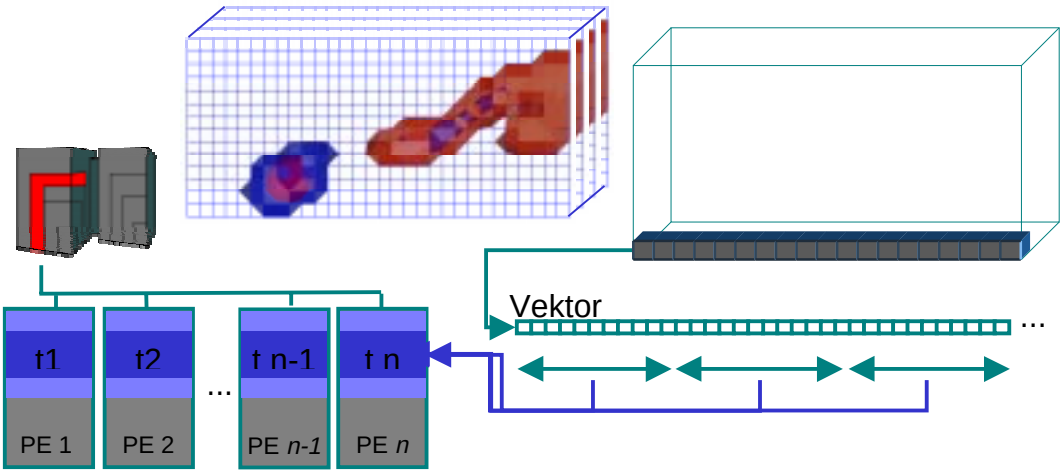


Abb. 4.3: Datenaufteilung bei Partrace. Der Speicherbereich des von Trace gelieferten Geschwindigkeitsfeldes wird in Partrace in Teilstücke zerlegt und auf die Prozessoren aufgeteilt. Eine solche Aufteilung erlaubt einen einfachen Zugriff auf Feldkomponenten mittels shmem.

4.1.3 Datenaustausch

Zur Kopplung der Programme muß das von Trace berechnete Geschwindigkeitsfeld des Grundwassers nach jedem Zeitschritt an Partrace übergeben werden. Da die Aufteilung des Feldes auf die einzelnen Prozessoren unterschiedlich ist und auch die Anzahl der benutzten Prozessoren pro jeweiligem Programm unterschiedlich sein kann, ist eine direkte Übertragung von Knoten zu Knoten nicht möglich. Ein Knoten benötigt also Feldkomponenten, die auf mehreren Trace-Knoten verteilt sein können. Die Kopplung wurde so implementiert, dass alle Trace-Knoten ihre Daten an den ersten Knoten von Partrace schicken. Von dort werden die Daten dann auf die restlichen Knoten von Partrace verteilt. Dazu werden derzeit die Schreiboperationen von shmem verwendet und jede Feldkomponente einzeln verteilt. Alternativ können die Datenblöcke vom ersten Knoten auch mit MPI-Broadcast an die restlichen Knoten verschickt werden, wobei dann jeder Knoten nur die lokal gespeicherten Feldkomponenten selektiert.

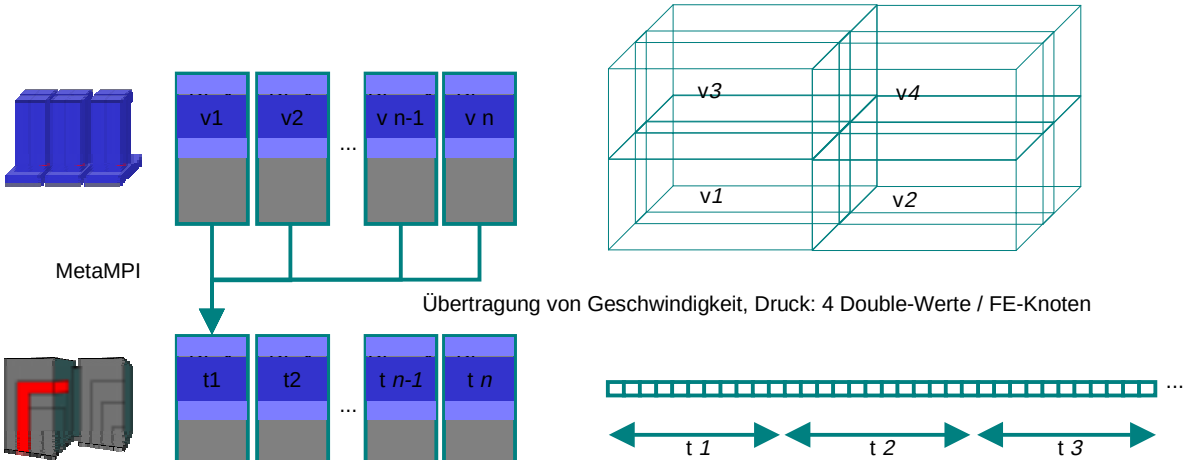


Abb. 4.4: Datenfluß zwischen Trace und Partrace. Die einzelnen Knoten von Trace schicken die dort gespeicherten Teile des Geschwindigkeitsfeldes über die Gigabit-Strecke an den ersten Knoten von Partrace. Von dort werden die Daten an die restlichen Knoten von Partrace verteilt. Bei diesem Verfahren werden die Daten nur einmal über das Weitverkehrsnetz übertragen.

Bei diesem Verfahren werden die Datenblöcke nur einmal über die Gigabit-Strecke zwischen den Rechner übertragen. Zudem ist die Reihenfolge beliebig, in der die einzelnen Trace-Knoten ihre Daten schicken.
Für die Kopplung der Programme wird das im Rahmen des Gigabit-Projekts entwickelte MetaMPI benutzt. Beide Programme befinden sich dabei im gleichen Kommunikator MPI_COMM_WORLD. Für die lokale Kommunikation innerhalb der Programme steht jeweils ein weiterer Kommunikator zur Verfügung.
Da der erste Partrace-Knoten durch die Verteilung der Daten stärker belastet ist als die anderen Rechenknoten, ist hier durch die angepaßte Verteilung der simulierten Partikel eine statische Lastbalance implementiert worden.

4.1.4 Zeitschrittsteuerung

Trace benutzt bei der Simulation eine dynamische Zeitschrittsteuerung. Je nach Konvergenzverhalten wird die Schrittweite reduziert oder verlängert. Auch die Rechenzeit pro Simulationsschritt ist unterschiedlich. Partrace benutzt dagegen eine fest eingestellte Zeitschrittlänge und auch die Rechenzeit, die ein Prozessor pro Partikel benötigt, ist fest.
Da der Datenfluß innerhalb der Kopplung nur in eine Richtung läuft, können Laufzeitschwankungen teilweise durch eine Pufferung der Fließfelder ausgeglichen werden. Dazu werden auf der Seite von Trace die Daten asynchron versendet. Der Speicherbereich für das Geschwindigkeitsfeld auf dem Trace-Knoten wird erst gegen Ende eines Simulationsschrittes wieder benötigt. Das heißt, dass die Daten in dieser Zeit für das Versenden zur Verfügung stehen. Trace muß nur dann auf das Ende der Kommunikation warten, wenn nach Ablauf der Rechnung zum Simulationsschritt die Daten des letzen Simulationsschritts noch nicht versendet sind.
Auf der Seite von Partrace können die Daten dagegen nicht asynchron empfangen werden, da alle Datenpakete auf dem ersten Partrace-Knoten zwischengespeichert und von dort weitergeleitet werden. Der Hauptspeicher des ersten Knoten ist aber zu klein, um dort das komplette Geschwindigkeitsfeld zu speichern. In der Zeit, die mit dem Empfangen der Daten von Trace verbracht wird, können keine Rechnungen durchgeführt werden. Daher spielt die Bandbreite des verwendeten Netzes eine wesentliche Rolle. Je schneller die Kommunikation zwischen den beiden Programmen ist desto mehr Zeit kann von Partrace für die Simulationsrechnung genutzt werden.

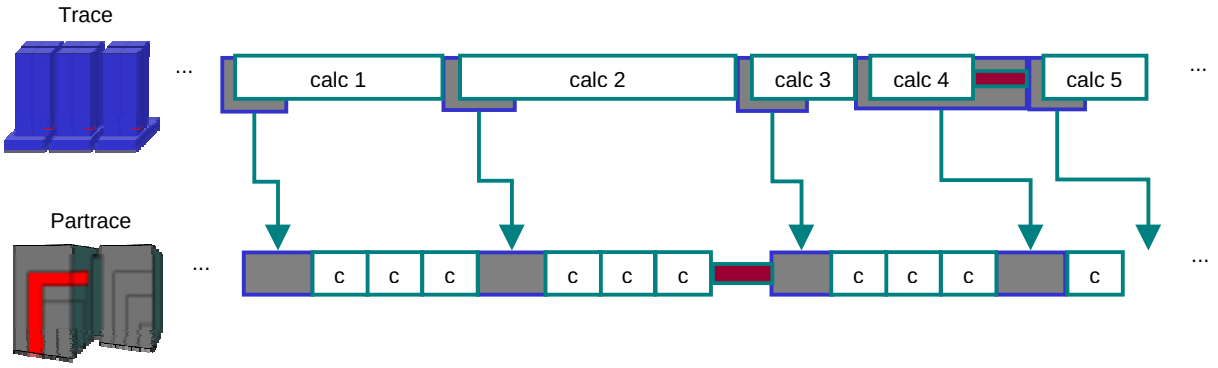


Abb. 4.5: Zeitschrittsteuerung in der gekoppelten Anwendung. Die dynamische Zeitschrittsteuerung und die variable Rechenzeit pro Simulation machen eine synchrone Kopplung beider Programme schwierig. Durch asynchrone Sendebefehle kann ein Teil der Kommunikationszeit durch Simulationsrechnungen überdeckt werden.

4.1.5 Messungen

Partrace benutzt teilweise die CRAY-spezifischen shmem-Befehle für die interne Kommunikation. Diese stehen auf der IBM SP2 nicht zur Verfügung. Für den Einsatz beider Programme im Gigabit Testbed West war damit die Verteilung der Programme auf die Rechner festgelegt. Partrace wird auf der CRAY T3E in Jülich, Trace auf der IBM SP2 in Sankt Augustin eingesetzt.
Für die Messung der maximalen Belastung der Gigabit-Leitung durch die Kommunikation der beiden Programme wurden Eingabedaten definiert, die eine maximale Hauptspeicherauslastung auf der kleineren der beiden Maschinen - der IBM SP2 – hervorrufen. Damit ist die Maximalgröße der zu übertragenden Daten definiert. Auf der SP2 stehen für Trace 32 Prozessoren mit jeweils 100MB freiem Hauptspeicher zur Verfügung. Pro FE-Knoten werden 50 Double-Werte gespeichert. Somit können pro Prozessor 250.000 FE-Knoten gespeichert werden, insgesamt also 8,3 Millionen FE-Knoten. Die durchschnittliche Rechenzeit beträgt dabei 200 Sekunden pro Simulationsschritt. Die maximale Auslastung bei Partrace auf der CRAY T3E sind bei 512 Knoten mit jeweils 512 MByte Hauptspeicher ca. 4 Millionen Partikel pro Rechenknoten. Die verteilte Speicherung des Geschwindigkeitsfeldes ist bei 512 Prozessoren vernachlässigbar. Die Rechenzeit pro Simulationsschritt beträgt dabei ca.10 Sekunden. Zwischen den beiden Programmen werden dabei in jedem Simulationsschritt 287 MByte Daten übertragen. Bei Messungen wurden dabei beim asynchronen Senden ca. 20 Sekunden benötigt. Das entspricht einer Bandbreite von 114 MBit/s.
In einer zweiten Messung wurden untersucht, wie die Kopplung der beiden Programme die Rechenzeit der einzelnen Programme beeinflußt. In der Abbildung 4.5 ist in einem Diagramm die Rechenzeit von Trace für die jeweils ersten 8 Simulationsschritte auf der SP2 aufgetragen. Das Eingabebeispiel hierbei umfaßt ca. 500000 FE-Knoten. Gemessen wurden jeweils die Rechenzeit bei der ungekoppelten Trace-Version mit Hersteller-MPI oder MetaMPI und bei der gekoppelten Version mit MetaMPI. Die Rechenzeit wird durch den Einsatz von MetaMPI nicht beeinflußt und durch die Kopplung nur unwesentlich verlängert.

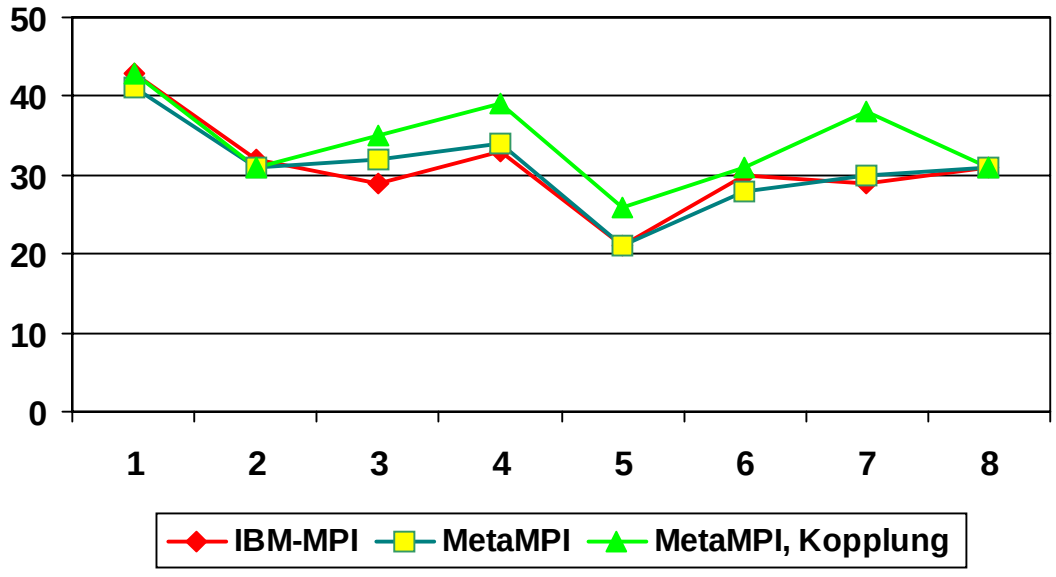


Abb. 4.6: Vergleich der Rechenzeit pro Simulationsschritt. Die drei Kurven zeigen die Rechenzeit der ersten 8 Simulationsschritte für die nicht gekoppelte Trace-Version mit Hersteller-MPI und MetaMPI sowie für die gekoppelte Version mit MetaMPI.

4.2 Online-Visualisierung

Im Rahmen des Projekt ist neben der Kopplung beider Programme auch eine Online-Visualisierung entwickelt worden, mit deren Hilfe Zwischenergebnisse während der laufenden Simulation betrachtet und die Parameter der Simulationsrechnung verändert werden können.
Die Visualisierung der Strömungs- und Konzentrationsfeldern erfolgt mit AVS/Express, das eine dreidimensionale Stereo-Darstellung auf dem Bildschirm oder auch auf der Responsive Workbench ermöglicht.
Für die Kopplung sind in AVS/Express Kommunikationsmodule und für die Simulationsprogramme eine Library implementiert worden. Diese *visit*-Library ermöglicht die direkte Übertragung von Daten vom Simulationsprogramm in das Verarbeitungsnetzwerk von AVS/Express. Dabei werden die Daten über Sockets übertragen und nicht auf Platte zwischengespeichert.

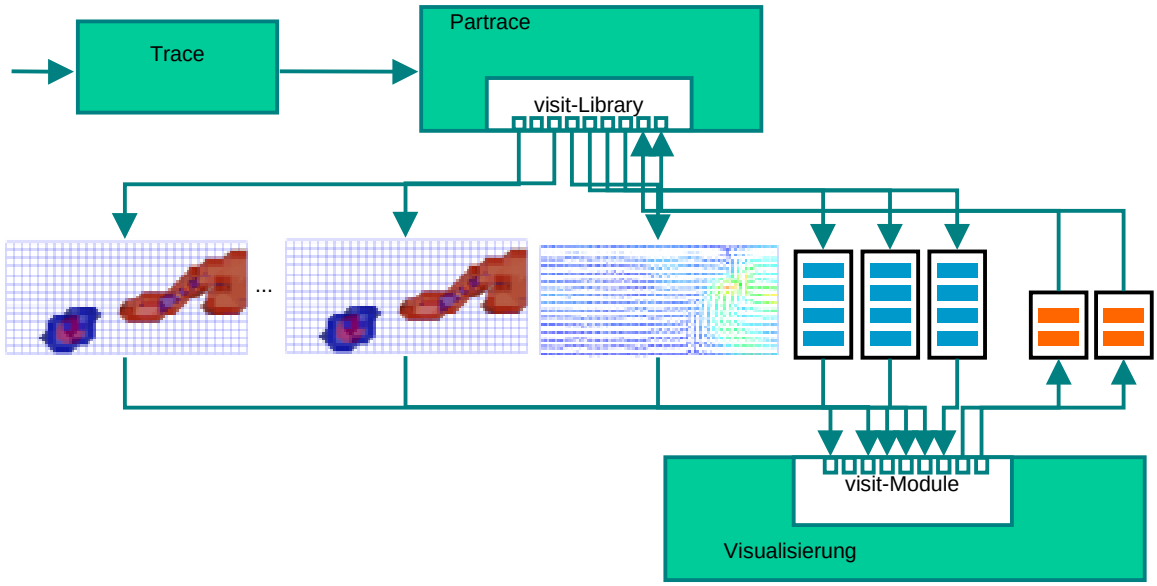


Abb. 4.7: Ankopplung der Visualisierung mit ***visit.*** Die von der Visualisierung benötigten Daten werden von Partrace aus mit Hilfe der visit-Library zu der in AVS/Express implementierten Oberfläche ParView geschickt. Neben den Konzentrationsfeldern und Strömungsfeldern werden auch Steuerungsdaten in beide Richtungen übertragen. Die Datenblöcke sind mit Kennungen versehen und können somit in ParView den verschiedenen Verarbeitungsschienen zugeführt werden.

Die Konzentration der einzelnen Schadstoffe wird mit Volumenoberflächen dargestellt. Dazu wird eine Oberfläche durch alle Volumenelemente gelegt, deren Konzentrationswert einer über das Steuerpanel einstellbaren Größe entspricht. Es können gleichzeitig bis zu drei verschiedene Schadstoffgruppen angezeigt werden. Das Geschwindigkeitsfeld des Grundwassers wird entweder mit Vektoren oder mit Strömungslinien angezeigt, deren Startposition interaktiv bestimmt werden kann. Zusätzlich kann der Wasserdruck über farblich kodierte Schnittebenen und die Leitfähigkeit des Boden mit Volumenoberflächen angezeigt werden.
Die in Partrace und Trace benutzten Felder können in dieser Größe (bis zu 10^8 FE-Knoten) in AVS/Express nicht direkt angezeigt werden. Die Felder werden daher vor der Übertragung zu ParView auf eine für AVS/Express anzeigbare Größe reduziert. Die Größe kann im Steuerpanel eingestellt werden.
In einem zweiten Fenster können in einem Diagramm Durchbruchskurven angezeigt werden, die den zeitlichen Verlauf der Schadstoffkonzentration an einer bestimmten Position im Simulationsgebiet zeigen. Die Position im Simulationsgebiet kann dabei interaktiv verändert

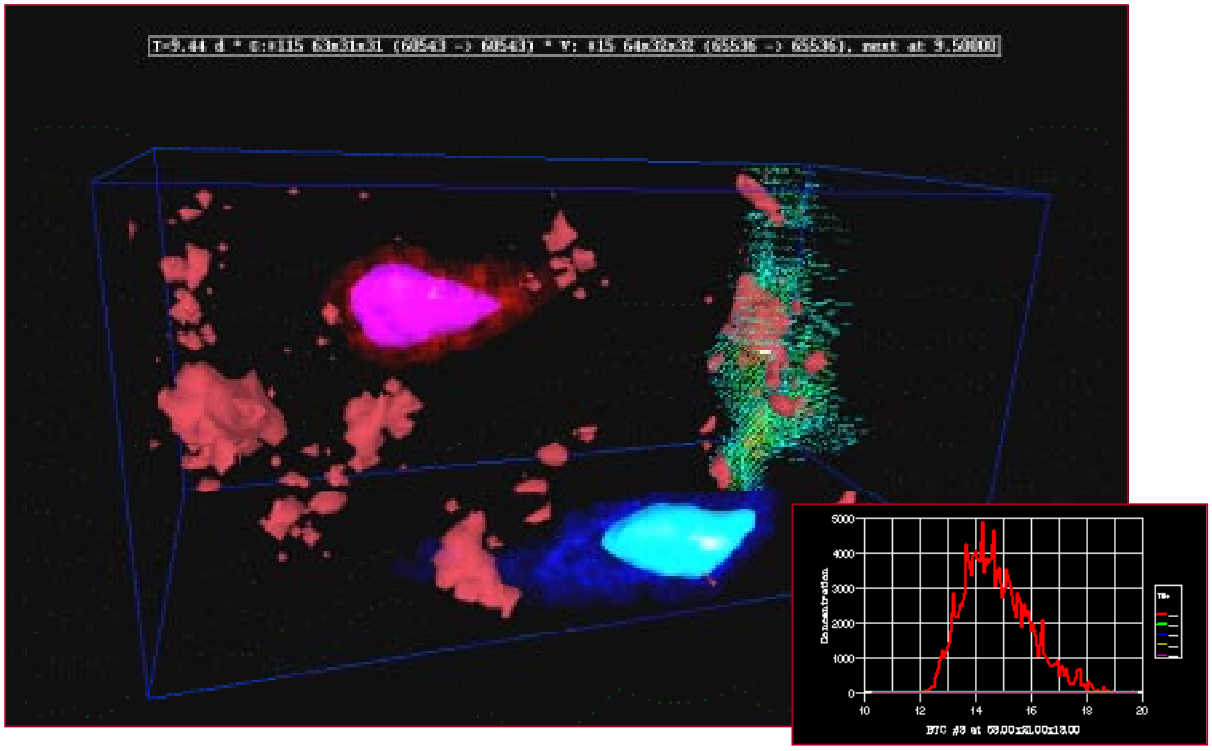


Abb. 4.8: Darstellung der Daten mit ParView. Das größere Bild zeigt die 3D-Darstellung, die neben den Schadstoffwolken auch das Strömungsfeld und die Leitfähigkeit (Porösität) des Boden an. Das kleinere Bild zeigt eine Durchbruchskurve, die den zeitlichen Verlauf der Schadstoffkonzentration an einer bestimmten Position im Simulationsgebiet anzeigt.

werden. Die Informationen, die Partrace zur Bestimmung der Kurven benötigt, werden in jedem Simulationsschritt von ParView an die Simulation übertragen.
Auch die Injektionspunkte für Schadstoffe, die Anzahl der Partikel und der Zeitpunkt der Injektion können interaktiv über das Steuerpanel bestimmt werden.
Die Übertragung von Daten von der Simulation zu ParView wird immer von der Simulation angestoßen. Aktionen innerhalb von ParView führen daher nicht zu einer Unterbrechung der Simulationsrechnung. Die Wartezeiten auf der Seite der Simulationsprogramme sind daher gering.

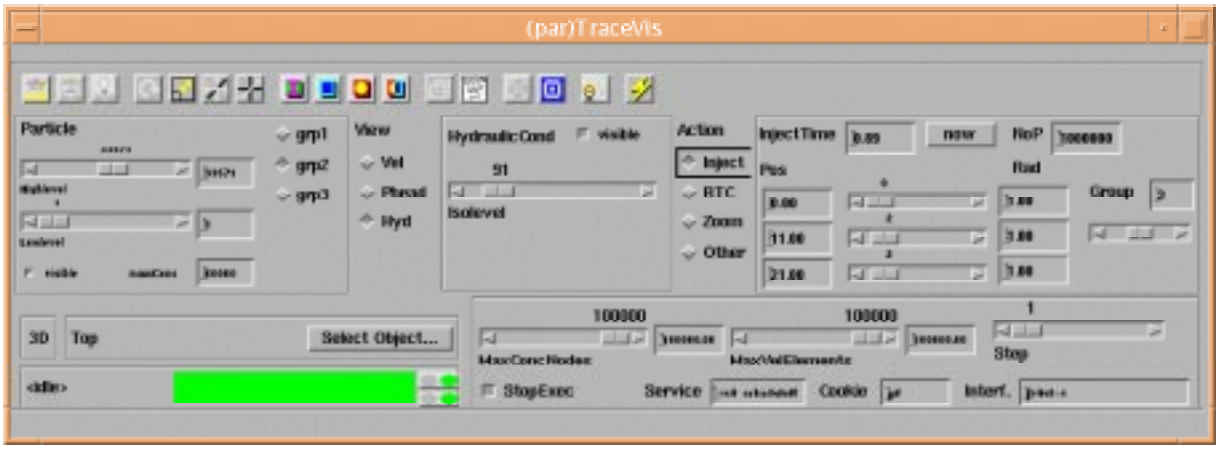


Abb. 4.9: Steuerpanel von ParView**.** Von diesem Panel aus können die wichtigsten Parameter der Visualisierung gesteuert werden. Zudem ist auch die Steuerung der Simulation von hier aus möglich. Im linken oberen Bereich kann die Anzeige der Partikel, im mittleren Bereich die Anzeige des Geschwindigkeitsfeldes, des Drucks und der Leitfähigkeit verändert werden. Im rechten Bereich wird das Einbringen von Schadstoffen und die Anzeige der Durchbruchskurven gesteuert. Der rechte untere Bereich enthält Einstellungen zur Verbindung zu Partrace.

Um auch das Anhängen an eine laufende Simulationsrechnung zu ermöglichen, wird die Verbindung zu ParView auf der Seite von Partrace über eine Konfigurationsdatei gesteuert, die in jedem Simulationsschritt erneut eingelesen wird. Die Konfigurationsdatei beschreibt dabei, wie die Maschine, auf der ParView läuft, erreicht werden kann. Sie enthält zusätzlich einen Schalter, mit dem die Verbindung ein- bzw. ausgeschaltet werden kann. Ändern sich während der Simulationsrechnung die Daten in dieser Datei, wird von Partrace die bestehende Socketverbindung geschlossen und ggf. eine neue Verbindung mit den aktuellen Parametern geöffnet. Dadurch wird es möglich, eine laufende Simulationsrechnung nacheinander von verschiedenen Workstations aus zu betrachten.

4.3 Zusammenfassung

Im Rahmen des Projekt sind zwei wesentliche Erweiterungen an den Anwendungen entwickelt worden. Die erste Erweiterung ist die direkte Kopplung der beiden Programme, durch die nun nicht-stationäre Eigenschaften der Grundwasserströmung bei der Simulation berücksichtigt werden können. Die Kopplung der beiden Programme mit Hilfe von MetaMPI war problemlos.
Die zweite Erweiterung ist die Online-Visualisierung mit ParView, die bei der Kontrolle der Ergebnisse hilft und sogar die Steuerung von Parametern ermöglicht. So können zum Beispiel die Positionen im Simulationsgebiet, an denen Durchbruchskurven abgenommen werden, interaktiv während der Simulation bestimmt werden. Bisher mußten die Positionen vor dem Start der Simulation festgelegt werden. Dies ist schwieriger, da der Weg der Schadstoffwolke im Simulationsgebiet zu diesem Zeitpunkt noch nicht bekannt ist.
Es hat sich gezeigt, dass die hier angewendete Verteilung der Programme auf dem Metacomputer - Trace auf dem einen und Partrace auf dem anderen Rechner – effizient ist. Gegenüber einer Aufteilung des Simulationsgebiets auf die beiden Rechner müssen hier zwar mehr Daten übertragen, diese aber weniger oft übertragen werden. Die vorwiegend in eine Richtung führende Übertragung kann asynchron und damit gleichzeitig mit der Simulations-rechnung erfolgen.

5 Algorithmische Auswertung der Magnetenzephalographie

Beteiligte Partner: *Institut für Medizin (IME/FZJ)*
Zentralinstitut für Angewandte Mathematik (ZAM/FZJ)
Institut für Algorithmen und Wissenschaftliches Rechnen (SCAI/GMD)

Ansprechpartner: *Dr. Thomas Eickermann (ZAM/FZJ)*
Weitere Beteiligte: *Dr. Roland Beucker (ZAM/FZJ), Jürgen Dammers (IME/FZJ), Dr. Peter Tass (IME/FZJ)*

Magnetenzephalographie (MEG) ist ein bildgebendes medizinisches Verfahren, bei dem die durch neuronale Aktivität im menschlichen Gehirn erzeugte magnetische Induktion gemessen wird. Dazu dient eine Anordnung von supraleitenden Magnetfeld-Detektoren (SQUIDs), die helmartig um den Kopf eines Probanden oder Patienten plaziert sind. Aus den Messungen lassen sich Rückschlüsse auf die elektrischen Ströme im Gehirn und damit die neuronale Aktivität ziehen. Die Stärke des Verfahrens ist der direkte Zugriff auf elektrophysiologische Daten in hoher zeitlicher Auflösung. Auch ist MEG mit keinerlei Belastung des Probanden oder Patienten verbunden.
Am Institut für Medizin (IME) des Forschungszentrums Jülich ist ein 148-Kanal Ganzkopfsystem BTi Magnes 2500 MEG der Firma BTi installiert. Mit diesem Gerät werden dort neben grundlagenorientierten Fragestellungen der kognitiven Neurowissenschaft auch Bewegungsstörungen wie z.B. die Parkinsonschen Krankheit untersucht. Durch die Aufklärung biochemischer und elektrophysiologischer Ursachen und deren Zusammenhänge soll ein Beitrag zur Entwicklung neuer Therapien wie eines bedarfsgesteuerten Hirnschrittmachers geleistet werden.
Den Stärken von MEG steht die relativ geringe räumliche Auflösung als Nachteil gegenüber. Die Rekonstruktion der kontinuierlichen Stromdichteverteilung im Kopf aus den Magnetfeldmessungen führt auf ein mathematisch unterbestimmtes inverses Problem, das prinzipiell nicht lösbar ist. Aus der Literatur sind verschiedene Verfahren bekannt, die mit zusätzlichen Annahmen Näherungslösungen liefern. Etablierte Verfahren sind der *Mu*ltiple *Si*gnal *C*lassification (MUSIC) Algorithmus, der die Stromverteilung durch wenige Dipole approximiert, und die Magnetic Field Tomographie (MFT), bei der a-priori Informationen aus simulierten Daten genutzt werden, um eine „wahrscheinlichste“ kontinuierliche Stromdichteverteilung zu berechnen.

Das Ziel in diesem Teilprojekt war die Implementierung eines verteilten parallelen MUSIC-Programms. Die Motivation dafür ist, dass für Anwendungen, wie sie oben skizziert sind, große Mengen von Daten ausgewertet werden. Bei typischen Parametern (einer Sampling-Rate von 1 kHz) fallen je 15 Minuten Messzeit etwa 1 Gigabyte Daten an. Zur Qualitätskontrolle der Experimente und Optimierung der Stimulationsbedingungen wird eine möglichst zeitnahe Auswertung der gemessenen Daten angestrebt. Einschließlich der Vorverarbeitung der Daten (Filterung und Beseitigung von Artefakten, die durch Herzschlag und Augenbewegungen erzeugt werden) nimmt die Auswertung auf einer Workstation heute ein bis zwei Tage in Anspruch. Durch den Einsatz von Supercomputern soll diese Zeit auf weniger als eine Stunde zu reduziert werden, so dass die Resultate noch während einer Sitzung verfügbar sind.
Eine weitere Reduktion der Laufzeit verspricht die Verteilung der Auswertung auf verschiedene Rechner, da Teilschritte des MUSIC-Algorithmus sich besonders effizient auf Rechnern verschiedener Architektur (massiv parallel und vektor-parallel) implementieren lassen.

5.1 Der MUSIC Algorithmus

Dem MUISC-Algorithmus liegt die Annahme zugrunde, dass eine relativ geringe Zahl von stark lokalisierten Stromquellen für das vom Gehirn erzeugte Magnetfeld verantwortlich ist. Diese Quellen werden durch punktförmige Stromdipole approximiert. Die Dipole werden als ortsfest angenommen, können aber ihre räumliche Orientierung ändern. Der Algorithmus plaziert die Dipole so, dass ihr Magnetfeld das gemessene Signal möglichst gut reproduziert. Indem eine Messung in zeitlich überlappende Intervalle zerlegt wird, die dann unabhängig voneinander ausgewertet werden (*Sliding Window* Technik), lassen sich auch zeitliche Änderungen der Erregungsorte erfassen.
Etwas genauer betrachtet läßt sich MUSIC in drei Teilschritte zerlegen, die im folgenden qualitativ beschrieben werden.

5.1.1 Trennung von Signal und Rauschen - SVD

Der erste Schritt dient dazu, Rauschanteile aus den gemessenen Magnetfeldern zu eliminieren und eine Abschätzung zu liefern, wie viele unabhängige Stromquellen für das gemessene Magnetfeld verantwortlich sind. Dazu wird eine Singulärwertzerlegung (*singular value decomposition* - SVD) der Meßdaten durchgeführt. Diese zerlegt das gemessene Signal in unabhängige Komponenten und liefert ein sogenanntes Singulärwertspektrum, das angibt welchen Anteil die jeweiligen Komponenten am Gesamtsignal haben. Bei einem hinreichend hohen Signal-zu-Rausch-Verhältnis (*Signal to Noise Ratio, SNR* z.B. 10:1) gibt es im Spektrum eine deutliche Lücke zwischen den großen Eigenwerten, die dem Signal zuzuordnen sind und den kleineren Eigenwerten, die vom Rauschen herrühren. Für alle weiteren Rechnungen wird nun nur noch der Signalunterraum berücksichtigt. Dessen Dimension (die Zahl der Signal-Eigenwerte) ist zugleich eine obere Schranke für die Zahl der Dipole, die das Signal erzeugen.

In vielen Messungen ist das SNR nicht so hoch, dass eine klare Trennung möglich ist. Eine einfache und robuste Möglichkeit, die Dimension des Signalraumes dennoch nach oben abzuschätzen ist dann, einen festen Anteil, z.B. 90%, des gemessenen Magnetfeldes (in einer geeigneten Normierung) dem Signal zuzuordnen. Ist das tatsächliche SNR niedriger, ist die Dimension des Signalraumes zu hoch abgeschätzt. Der folgende Abschnitt macht deutlich, dass dies unkritisch für die weiteren Schritte des Algorithmus ist.

Numerisch muß hier die SVD einer Matrix durchgeführt werden, deren Größe die Zahl der Magnetfeldsensoren (148 Zeilen) mal der Anzahl der Meßwerte in dem analysierten Zeitintervall (je nach Parametern, typisch ca. 400 Spalten) ist. Diese Dimensionen lassen erwarten, dass auf einem Vektorrechner wie der Cray T90 bereits eine gute Performance zu erzielen ist. Für eine Parallelisierung auf einem Distributed-Memory Rechner wie der IBM SP2 oder der Cray T3E ist das Problem jedoch noch zu klein.

5.1.2 Lokalisierung der Dipole

Im zweiten Schritt des MUSIC-Algorithmus werden Position und Orientierung der Dipole so bestimmt, dass ihr Magnetfeld das gemessene Signal möglichst gut reproduziert. Dazu wird zunächst ein Modell für das sogenannte „Vorwärtsproblem“ benötigt, d.h. für das Magnetfeld, welches ein Dipol an den Orten der 148 SQUIDs erzeugt. Gängig sind dabei Kugelmodelle, bei denen der Kopf durch eine Kugel mit homogener elektrischer Leitfähigkeit approximiert wird. Für dieses Modell kann eine geschlossene Formel für das Vorwärtsproblem angegeben werden. Realistischere Modelle, die den Kopf durch einen Ellipsoid oder durch Kugel- bzw.

Ellipsiod-Schalen mit jeweils homogener Leitfähigkeit annähern sind numerisch erheblich aufwendiger und wurden hier nicht berücksichtigt.
Wenn alle Modellannahmen korrekt wären, würde das vom Dipol generierte Magnetfeld vollständig in dem im ersten Schritt bestimmten Signalunterraum liegen. MUSIC bestimmt die Positionen und Orientierungen der Dipole so, dass der Winkel zwischen dem Dipolfeld und dem Signalunterraum minimal wird. Die Bestimmung der Dipol-Orientierung läßt sich dabei von der des Ortes separieren, so dass effektiv Extremwerte einer skalaren Funktion (dem Winkel) gesucht werden, die von 3 Parametern abhängt (der Dipolposition). Die mit der SVD abgeschätzte Dimension des Signalraumes ist dabei eine Obergrenze für die Anzahl der Dipole. Zusätzlich wird durch eine Obergrenze für den Winkel die Anzahl der akzeptierten Dipole weiter reduziert. Diese Grenze muß nach Sichtung der Resultate manuell eingestellt werden. Typische Werte liegen zwischen 15° und 35°.
Aus der medizinischen Anwendung wird eine Genauigkeit der Dipolposition von ca. einem Millimeter gefordert, wobei Dipole, die 5 Millimeter voneinander entfernt sind, noch unterschieden werden sollen. Um dieses zuverlässig zu erreichen, wird der Winkel zunächst auf einem Gitter mit 5 Millimeter Auflösung berechnet, das dann lokal an den Minima verfeinert wird, bis eine Genauigkeit von einem Millimeter erreicht ist.

Dieses Verfahren läßt sich effizient auf einem massiv-parallelen Rechner implementieren, da im wesentlichen eine große Anzahl von Funktionsauswertungen (hier typisch 20.000), die voneinander unabhängig sind, durchgeführt werden. Zwischen den Prozessoren muß man lediglich die lokale Verfeinerung des Gitters koordinieren.

5.1.3 Zeitverlauf der Dipolstärke

Im letzten Schritt wird der zeitliche Verlauf der Dipolstärke (also der Stärke der neuronalen Erregung) bestimmt. Dazu muß nur noch das gemessene Magnetfeld auf das von den Dipolen generierte Feld projiziert werden, da aus den vorherigen Schritten sowohl der Signalraum als auch die Dipolfelder vorliegen.

5.2 Implementierung auf Parallelrechnern

In einem ersten Schritt wurde der MUSIC-Algorithmus in dem parallelen Programm **pmusic** auf MPI-Basis implementiert. Bei der Parallelisierung wurde ein zweistufiges Konzept verfolgt. Zunächst wurden die oben beschriebenen Teilschritte des Algorithmus in Module gekapselt und diese Module auf verschiedene Prozessoren verteilt. Dies bereitet zum einen die geplante Verteilung der Module auf verschiedene Rechner vor, verringert aber durch das Pipelining der Teilaufgaben bereits die Laufzeit, wenn in einem Programmlauf mehrere unabhängige Datensätze ausgewertet werden, wie beispielsweise bei der Sliding Window Auswertung. Insgesamt wurde **pmusic** in vier Module aufgeteilt:

- **reader** liest Steuer- und Experimentparameter und leitet sie an die anderen Module weiter. Es unterstützt verschiedene Dateiformate für die MEG-Meßdaten und unterschiedliche Auswertemodi (Sliding Window, Beschränkung der Analyse auf Teilintervalle, Verringerung der zeitlichen Auflösung, etc.)
- **svd_ts** führt die Singulärwertzerlegung (SVD) der Meßdaten durch und schätzt anhand des Eigenwertspektrums die Anzahl (*n*) der Strom-Dipole im Gehirn. Ein weiteres Resultat ist der Signalunterraum, d.h. die Wirkung dieser Dipole auf die Spulen des MEG-Gerätes. Diese Ergebnisse werden an das nachfolgende Modul weitergeleitet.

- **b_max** berechnet die Position und Orientierung der Strom-Dipole. Diese Resultate, das Magnetfeld am Ort der Spulen und der Winkel zwischen Dipolfeld und Signalunterraum werden an das nächste Modul übertragen.
- **amp_ts** bestimmt den zeitlichen Verlauf der Amplitude der von **b_max** lokalisierten Strom-Dipole so, dass der gemessene Signalverlauf möglichst gut approximiert wird. Alle Resultate werden in Dateien ausgegeben.

Zunächst wurden in den einzelnen Modulen serielle Algorithmen benutzt. Mit Ausnahme des **reader**-Moduls wird der größte Teil der Rechenzeit aller Module für Operationen der linearen Algebra benötigt, für die hochoptimierte Bibliotheken existieren (NAG, LAPACK, SciLib). In einem zweiten Schritt wurden die Module **svd_ts** und **b_max** parallelisiert. Für die SVD des Moduls **svd_ts** wurde dazu die Bibliothek ScaLAPACK eingesetzt. Diese ist eine für massiv-parallele Rechner optimierte parallele Variante der LAPACK-Bibliothek. Bei der Parallelisierung des Moduls **b_max** wurde ein Farming-Konzept benutzt. Ein Master-Prozess verteilt die Daten, jeder Worker-Prozessor führt einen Teil der Funktionsauswertungen durch. Anschließend sammelt der Master die Funktionswerte und bestimmt die Maxima. Um die gefundenen Orte wird das Gitter lokal verfeinert, wobei die Arbeit wieder auf die Worker-Prozessoren verteilt wird. Abbildung 5.1 zeigt schematisch die Struktur von **pmusic**.

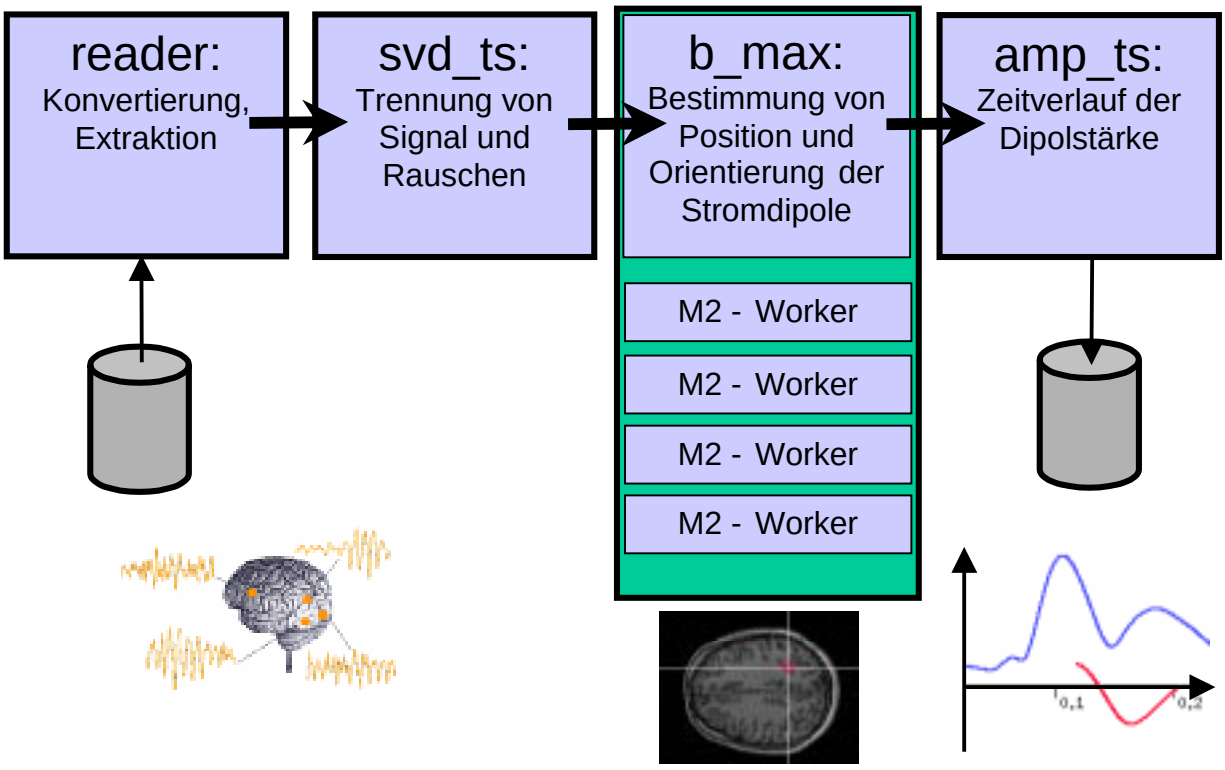


Abb. 5.1: Aufbau des Programms **pmusic**. Die von **reader** gelesenen Daten werden von den anderen Modulen verarbeitet und schließlich in Dateien geschrieben oder direkt an eine einfache 2D-Visualisierung weitergereicht. **reader, svd_ts** und **amp_ts** laufen auf je einem Prozessor, **b_max** kann auf einem oder mehreren Prozessoren parallel ablaufen.

Die Auswahl der Plattformen für die verteilte Version von **pmusic** erfolgte anhand von Leistungsmessungen mit der 1998 fertiggestellten ersten Version des parallelen Programms. Als Testbeispiel diente ein gemitteltes auditorisches Experiment von 1 sec Dauer, bei dem die Meßdaten mit einer Frequenz von 4 kHz gesampelt wurden. Ausgewertet wurde der sogenannte M100-Response, eine neuronale Antwort, die etwa 100 Millisekunden nach dem Reiz beobachtet wird und etwa 100 Millisekunden andauert. Die folgenden Tabellen zeigen

die Ergebnisse für die im Projekt relevanten Rechner-Plattformen CRAY T3E, CRAY T90 und IBM SP2. Zum Zeitpunkt der Messungen stand im ZAM eine T3E-900 zur Verfügung, die inzwischen zu einer T3E-1200 (mit höherem Prozessortakt) aufgerüstet wurde. Erwartungsgemäß zeigte sich der Vektorrechner T90 bei der SVD überlegen. Die Parallelisierung dieses Moduls brachte auf der T3E aufgrund der geringen Problemgröße keine Verbesserung.

Rechner	reader	svd_ts	b_max	amp_ts
T3E-900	0.29	0.41	25.2	0.039
T90	0.21	0.14	39.9	0.050
SP2	0.09	0.82	69.8	- (*)

Tab. 5.1: Laufzeiten der Module von **pmusic** in Sekunden. Jedes Modul nutzt nur einen Prozessor. (*) Eine benötigte Bibliothek (NAG) stand auf der SP2 nicht zur Verfügung.

	seriell	1	2	4	8	16
T3E-900	0.41	0.93	0.78	0.69	0.65	0.65
Speedup	-	0.22	0.52	0.59	0.63	0.63

Tab. 5.2: Laufzeit des Moduls **svd_ts** in Sekunden. Die serielle Version wird mit der parallelen Version (1 bis 16 Prozessoren) verglichen. Wegen der geringen Problemgröße ist die serielle Version effizienter.

Im Gegensatz dazu läßt sich auf allen Plattformen die Laufzeit des Moduls **b_max** erheblich reduzieren. Bei T90 und SP2 begrenzt die Anzahl der verfügbaren Prozessoren die Gesamtleistung. Bei der T3E nimmt die Effizienz bei mehr als 128 Prozessoren deutlich ab. Eine genauere Analyse zeigt drei Hauptursachen: die Zeit, die für das Verteilen und Sammeln der Daten benötigt wird, wächst (in etwa logarithmisch) mit der Prozessorzahl. Die Zeit, die der Master für das Aufsuchen der Maxima benötigt, bleibt konstant. Bei insgesamt abnehmender Laufzeit steigt der Anteil dieser beiden nicht-skalierenden Programmteile ungünstig an. Außerdem sind an den lokalen Gitterverfeinerungen jeweils höchstens 125 Prozessoren beteiligt. Eine denkbare Optimierung zur Verbesserung der Skalierbarkeit ist die Aufteilung des Volumens auf mehrere Master-Prozessoren.

Rechner	seriell	1	2	4	8	16	31	64	128	192
T3E-900	25.2	25.2	12.6	6.4	3.2	1.7	0.91	0.49	0.32	0.27
Speedup	-	1.0	2.0	3.9	7.9	15	28	51	79	93
T90	39.9	39.9	20.0	10.0	5.1					
Speedup	-	1.0	2.0	4.0	7.8					
SP2	69.8	69.9	35.2	17.7	9.1	4.6	2.5			
Speedup	-	1.0	2.0	3.9	7.7	15	24			

Tab. 5.3: Laufzeit des Moduls **b_max** in Sekunden. Auf allen untersuchten Plattformen wird durch Parallelisierung eine erhebliche Reduktion der Laufzeit erreicht.

5.3 Messungen auf dem heterogenen Metacomputer

Aufgrund der Ergebnisse des vorigen Abschnittes wurden für die verteilte Version von **pmusic** die CRAY T90 und die CRAY T3E ausgewählt. Die IBM SP2 der GMD, die noch mit *Thin Nodes* der ersten Generation (66 MHz) ausgestattet ist, zeigte bei keinem Modul eine konkurrenzfähige Leistung. Um die Daten nicht mehrfach zwischen T3E und T90 austauschen zu müssen, wurden **reader** und **svd_ts** auf der T90 und **b_max** und **amp_ts** auf der T3E ausgeführt.

Mit geringen Anpassungen am Quellcode ließ sich **pmusic** von den bisher genutzten systemeigenen MPI-Versionen auf MetaMPI umstellen. Allerdings war die Umstellung mit intensiver Fehlersuche und Erstellung von Testfällen auf Seiten von MetaMPI verbunden, da dies die erste Anwendung war, die eine T90-T3E Kopplung realisierte. Bei den Laufzeiten der einzelnen Module war kein Unterschied zwischen der mit dem Hersteller-MPI und der mit MetaMPI gebundenen Version festzustellen. Dies liegt vor allem daran, dass der Anteil der Kommunikation an der Gesamtlaufzeit des Programms relativ gering ist. Für einen Vergleich zwischen der auf einem Rechner ablaufenden Version mit der Metacomputing-Version wurden zwei Testfälle ausgewählt, die in etwa der geplanten Anwendung von **pmusic** entsprechen: das bereits beschriebene auditorische Experiment (148 Detektoren, 4 kHz Sampling-Frequenz, 1 Sekunde Dauer) wird nun vollständig ausgewertet. In der ersten Variante wird die Messung in 21 überlappende Intervalle von je 100 Millisekunden Dauer zerlegt, die jeweils einzeln ausgewertet werden (*Sliding Window*).

Rechner	CPUs	reader	svd_ts	b_max	amp_ts	gesamt
T3E-1200	4	0.3	**6.9**	218.3	0.8	219.3
	128			2.6		**7.5**
T90	4	0.1	3.2	437.1	2.2	437.9
	6			**147.2**		**149.6**
T2E-1200 + T90	8	0.1	2.5	73.4	0.9	74.2
	128			**2.9**		**3.8**

Tab. 5.4: Laufzeiten der Module von **pmusic** in Sekunden für minimale und optimale Prozessorzahlen, sowohl in verteilter als auch in nicht verteilter Konfiguration für eine Sliding Window Auswertung. Bei der verteilten Konfiguration schließt die Prozessorzahl die von MetaMPI benutzten Router-PEs ein.

Tabelle 5.4 faßt einige der gemessenen Laufzeiten zusammen. Man erkennt, dass die Gesamtlaufzeit wesentlich durch die Laufzeit des jeweils langsamsten Moduls bestimmt wird. Da die Module in einer Pipeline durchlaufen werden, setzt sich die Gesamtzeit aus der Startup-Zeit der Pipeline (also der Zeit, die zur Auswertung eines 100 Millisekunden Intervalls benötigt wird) und der Laufzeit des langsamsten Moduls zusammen. Bei der T3E-Version ist letztlich das Modul **svd_ts** der begrenzende Faktor. Durch dessen Auslagerung auf die T90 wird die Gesamtlaufzeit mehr als halbiert. Für längere Zeitreihen ist ein noch günstigeres Verhältnis zu erwarten, da dann die Startup-Zeit der Pipeline keine Rolle mehr spielt.

Die zweite Variante der Auswertung besteht darin, die gesamte Messung in einem Schritt auszuwerten. Der numerische Aufwand hierfür ist insgesamt geringer als bei der Sliding Window Auswertung, so dass kürzere Gesamtlaufzeiten erreicht werden, wie in Tabelle 5.5 und Abbildung 5.2 gezeigt ist. Da nur noch ein Datensatz die Pipeline der Module durchläuft, ist die Laufzeit nun die Summe der Laufzeiten der einzelnen Module. Hinzuzurechnen ist

auch der Kommunikationsaufwand, da die Kommunikation zwischen den Modulen nun nicht mehr mit Berechnungen überlappt werden kann. Um den Einfluß der Kommunikation zu untersuchen, wurden die Daten einmal lokal über das HiPPI-Netz des Jülicher Cray-Komplexes übertragen und einmal mit einer „Schleife“ Jülich – Sankt Augustin – Jülich über das Gigabit Testbed umgelenkt. Von der T90 zur T3E werden in dem Beispiel etwa 5 MByte in einem Block übertragen. Wie die Tabelle zeigt, wirkt sich die mit der Schleife erzielte geringere Bandbreite bereits messbar auf die Gesamtlaufzeit aus. Ursache ist dabei nicht die Bandbreite im Testbed sondern der Durchsatz der beiden HiPPI-ATM-Gateways im Kommunikationspfad. Auch die Latenz des WANs (ca. 2 Millisekunden) fällt gegenüber der Übertragungszeit von mehr als 200 Millisekunden nicht ins Gewicht.

Rechner	reader	svd_ts	b_max	amp_ts	svd_ts → b_max		gesamt
T3E-1200	0.1	2.4	**0.3**	**0.4**	0.026	180 MB/s	3.8
T90	< **0.1**	**0.3**	7.1	1.8	0.22	20 MB/s	10.5
T90 + T3E / HiPPI	< **0.1**	**0.3**	**0.3**	**0.4**	0.21	22 MB/s	**1.8**
T90 + T3E / Gigabit					0.35	13 MB/s	**2.0**

Tab. 5.4: Laufzeiten der Module von **pmusic** in Sekunden für optimale Prozessorzahlen in verteilter und nicht verteilter Konfiguration für die Auswertung eines 1 Sekunden-Intervalls. Bei der T90 wurden alle 10 Prozessoren eingesetzt, für die anderen Messungen 128 inclusive der Router-PEs.

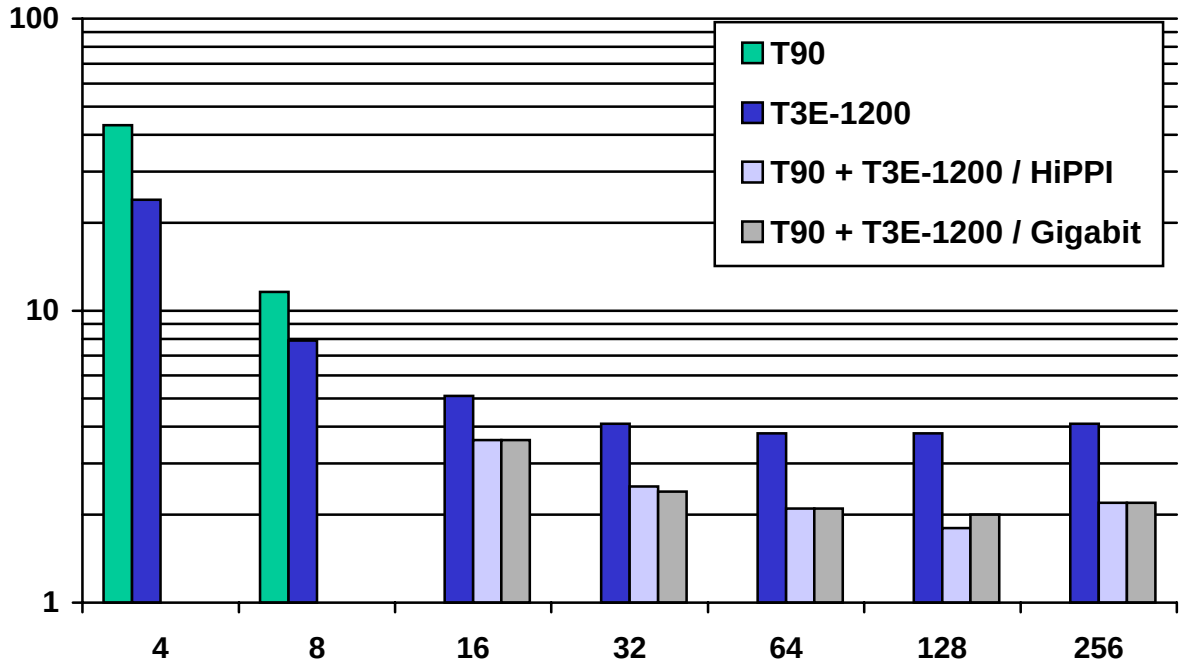


Abb. 5.2: Gesamtlaufzeit des Programms **pmusic** für unterschiedliche Konfigurationen und Prozessorzahlen. Bereits ab 16 Prozessoren ist die verteilte Version deutlich überlegen.

5.4 Visualisierung der Ergebnisse

Begleitend zu der Entwicklung von **pmusic** wurde *DisCo-MUSIC* (Display and Control of MUSIC), ein graphisches Werkzeug zur Darstellung der Ergebnisse, auf Basis von Perl/Tk entwickelt. DisCo-MUSIC ist in der Lage, die von **pmusic** berechneten Dipole mit Kernresonanz-Aufnahmen des Probanden zu überlagern. So wird eine Zuordnung der Aktivitätszentren zu anatomischen Strukturen im Gehirn möglich. Außerdem werden Zeitverläufe der Dipolamplituden dargestellt und der maximal akzeptierte Winkel zwischen Dipolfeld und Signalunterraum kann interaktiv justiert werden. Zur Zeit ist DisCo-MUSIC ein reines Post-Processing Werkzeug. Es ist geplant, Möglichkeiten zur Steuerung von pmusic zu integrieren. Damit soll die Möglichkeit geschaffen werden, auch solche Parameter des Programms interaktiv zu verändern, die Auswirkungen auf den Programmablauf haben. Ein Bildschirmabzug von DisCo-MUSIC ist in Abbildung 5.3 gezeigt.

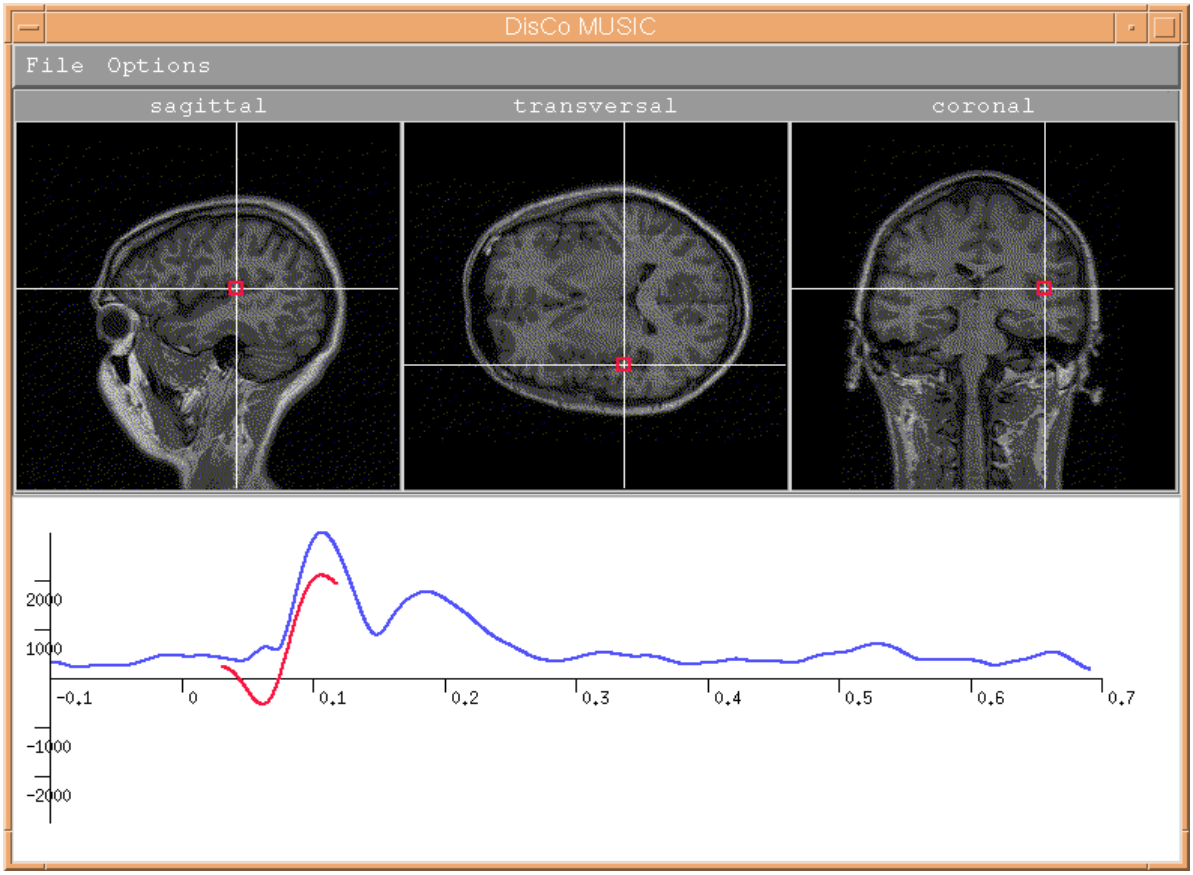


Abb. 5.3: DisCo-MUSIC-Display des Testbeispiels in Sliding Window Auswertung. Im oberen Bereich sind drei orthogonale Schnitte durch eine Kernresonanz-Aufnahme des Probanden gezeigt. Das rote Quadrat markiert die Position des stärksten Dipols, er befindet sich im auditorischen Cortex. Das untere Diagramm zeigt in blau das mittlere quadratische Magnetfeld, das von den SQUIDs gemessen wurde. Man erkennt deutlich den M100 Puls, der etwa 100 Millisekuden nach dem Klick-Laut (zum Zeitpunkt 0) auftritt. Der zeitliche Verlauf der Dipolamplitude des oben dargestellten Dipols ist in rot markiert.

5.5 Zusammenfassung und zukünftige Arbeiten

In diesem Teilprojekt wurde ein verteiltes paralleles Programm zur Auswertung von MEG-Messungen mit dem MUSIC-Algorithmus auf einem heterogenen Metacomputer implementiert. Dabei ist es durch geeignete Aufteilung der Teilschritte des Verfahrens auf ein massiv paralleles und ein vektor-paralleles System gelungen, die Laufzeit des Programms gegenüber einem nicht verteilten parallelen Programm mindestens zu halbieren. Sofern – wie im Gigabit Testbed der Fall – eine ausreichende Bandbreite zur Verfügung steht, kann das Programm auch über ein Weitverkehrsnetz verteilt werden, da es unempfindlich gegenüber Latenzen im Millisekundenbereich ist.
Eine kürzere Laufzeit verspricht Vorteile für den Betrieb des MEG-Tomographen im Institut für Medizin im Forschungszentrum Jülich. Es ist geplant, das MUSIC-Verfahren dort in der Arbeitsgruppe von Dr. Tass bei neurowissenschaftlichen Experimenten einzusetzen, um durch eine schnelle Auswertung laufender MEG-Experimente noch während einer Sitzung mit dem Probanden oder Patienten z.B. die Stimulationsbedingungen optimieren zu können.
Außerdem soll in pmusic das leistungsfähigere RAP-MUSIC (*R*ecursively *A*pplied and *P*rojected MUSIC), eine Erweiterung des klassischen MUSIC-Verfahrens, implementiert werden. RAP-MUSIC verspricht Vorteile bei der Lokalisierung mehrerer unabhängiger oder gekoppelter Dipole.

6 Komplexe Datenvisualisierung über ein Gigabit-WAN

Beteiligte Partner: *Institut für Medienkommunikation (IMK/GMD)*
Institut für Medizin (IME/FZJ)
Zentralinstitut für Angewandte Mathematik (ZAM/FZJ)
Institut für Algorithmen und Wissenschaftliches Rechnen (SCAI/GMD)

Ansprechpartner: *Wolfgang Frings (ZAM/FZJ), Dr. Martin Göbel (IMK/GMD)*
Weitere Beteiligte: *Gernot Goebbels (IMK/GMD), Dr. Thomas Eickermann (ZAM/FZJ), Daniel Gembris (IME/FZJ), Priv.-Doz. Dr. Stefan Posse (IME/FZJ), Dr. Roland Völpel (SCAI/GMD), Dr. Herwig Zilken (ZAM/FZJ)*

Nicht nur in wissenschaftlichen, sondern auch in technischen Arbeitsprozessen ist die Visualisierung eine zentrale Voraussetzung zur verständlichen Darstellung der komplexen wissenschaftlich-technischen Ergebnisse. Sie ist ein unerläßliches Hilfsmittel zum Studium dynamischer Systeme, der Darstellung abstrakter und auch nicht sichtbarer Phänomene. Mit der Verfügbarkeit leistungsfähiger Parallel-/Multivektor-Rechner kommt man den Anforderungen an Echtzeit-Darstellung gerade für große Anwendungen am nächsten.

Insbesondere in Bereichen der medizinischen Datenverarbeitung sind bildgebende Verfahren in der anwendungsorientierten Routine und Forschung nicht mehr wegzudenken. Auch in diesem Bereich ist man in immer stärkerem Maße nicht nur an der reinen Darstellung der Daten interessiert, sondern zusätzlich sollen verschiedene Stufen der Visualisierungspipeline integriert und interaktiv bearbeitet werden. Hierzu gehören neben der eigentlichen Visualisierung vor allem die Teilaspekte Transformation (Filterung, Skalierung, Projektion), Klassifizierung, Segmentierung und Modellbildung.

Mehr und mehr zeigt sich auch, dass die Visualisierung der Rohdaten, ohne vorherige Bearbeitung, oft zu nicht akzeptablen und nicht aussagekräftigen Resultaten führt. Gerade aber das immense Anwachsen der verfügbaren Datenvolumina macht sowohl die Integration verschiedener Stufen als auch deren Ausführung in akzeptabler Zeit auf heutigen Workstations unmöglich. Der limitierende Faktor ist hierbei neben der begrenzten Speicherausstattung vor allem auch die numerische Rechenleistung moderner Einprozessorsysteme. Die verschiedenen Stufen müssen auch für große Datensätze in wenigen Sekunden bearbeitet werden können, so dass eine interaktive Rückkopplung aufgrund des dargestellten Resultats möglich ist.

Zur effizienten Bewältigung der beschriebenen Probleme müssen modernste Technologien aus den Bereichen Hard- und Software bzw. Netze eingesetzt und zu einem permanent verfügbaren Gesamtsystem integriert werden. Grundsätzliches Ziel ist, die Daten der experimentellen Messung über einen örtlich abgesetzten zentralen Visualisierungsserver aufzubereiten, auf welchen von beliebigen, geographisch verteilten Lokationen zugegriffen werden kann. Die eigentliche Darstellung kann dann entweder auf diesem zentralen Visualisierungsserver oder auf lokal vorhandenen speziellen Graphikworkstations erfolgen.

Nur die Nutzung verschiedener Hardware-Plattformen, welche sowohl den Anforderungen an Speicherkapazität als auch an numerischer Rechenleistung und Graphik gerecht werden, gekoppelt über leistungsfähige Breitbandnetze, kann hierbei zu akzeptablen Bearbeitungszeiten der heute und in Zukunft anfallenden Datenmengen führen.

Eine Anwendung, bei der die oben genannten Aspekte eine wichtige Rolle spielen, ist die Auswertung und Anzeige von funktionalen Kernresonanzbildern in Echtzeit (fMRI). Dort werden die mit Hilfe eines Kernresonanz-Tomographen (MR-Scanner) aufgenommenen Schichtbilder des menschlichen Gehirns analysiert und graphisch aufbereitet. Mit diesem Verfahren ist es möglich, die aktiven Bereiche des Gehirns bei z.B. motorischen, visuellen oder akustischen Reizen zu lokalisieren. Da die Aufnahmen in kurzen Zeitabständen erfolgen, bleibt für die Analyse und Visualisierung der Daten nur wenig Zeit.
Im Rahmen des Gigabit Testbed West ist eine im Institut für Medizin (IME) im Forschungszentrum Jülich entwickelte Client-Server-Anwendung (FIRE) zur Auswertung der Daten so erweitert worden, dass auch bei einer aufwendigeren Analyse der Daten die Verarbeitung und die Anzeige in „Echtzeit“ erfolgen kann. Dazu wurden neben dem Kernresonanztomographen weitere im Gigabit Testbed West vorhandene Spezialrechner (CRAY T3E im FZJ und SGI Onyx 2 in der GMD) in den Verarbeitungsprozeß eingebunden. Für die 3D-Darstellung der Ergebnisse wird eine Responsive Workbench benutzt. Während die vom MR-Scanner produzierten Datenvolumina noch über Standard-Netze übertragen werden können, erfordert die Übertragung der in der GMD erzeugten Bilddaten Gigabit-Kapazität.

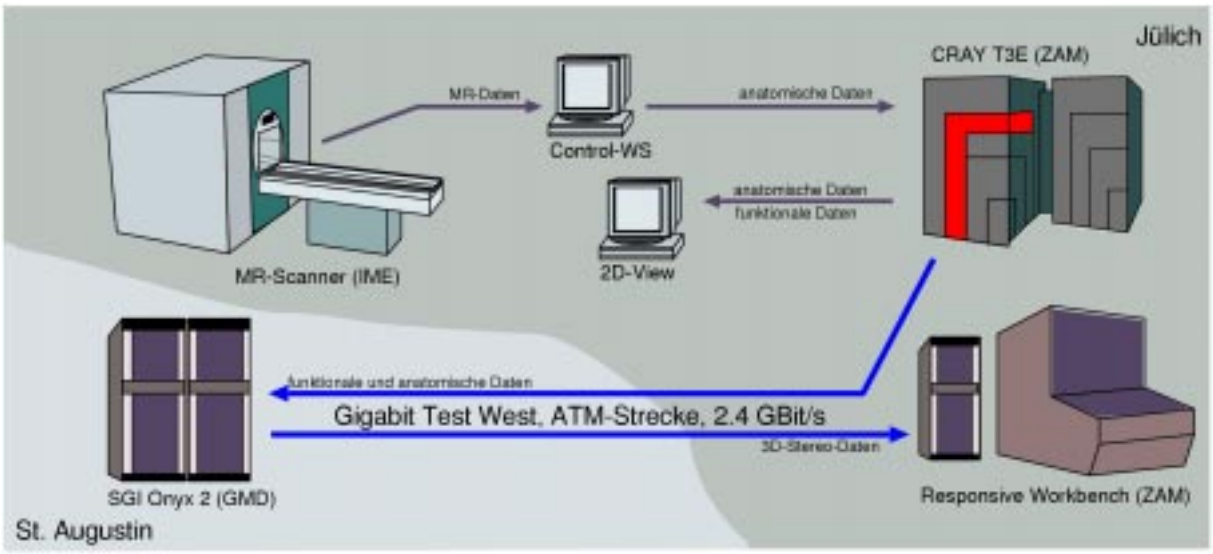


Abb 6.1: Datenfluß der verteilten Analyse und Anzeige der MR-Daten: Die Schichtbilder werden von der Bedienworkstation des MR-Scanners an den Parallelrechner CRAY T3E gesendet, dort analysiert und zusammen mit den Ergebnissen an die FIRE-Oberfläche auf einer lokalen Workstation weitergegeben. Gleichzeitig werden die Daten über die Gigabit-Strecke an den Visualisierungsserver Onyx 2 in der GMD geleitet. Dieser generiert daraus eine 3D-Darstellung in Stereo. Statt diese lokal in der GMD anzuzeigen, werden die gerenderten Bilder (Framebuffer) über die Gigabit-Strecke nach Jülich übertragen und dort angezeigt.

Die folgenden Abschnitte beschreiben die Anwendung und die für die verteilte Implementierung notwendigen Erweiterungen.

6.1 Komponenten von FIRE und die verteilte Implementierung

Das Programmpaket FIRE (**F**unctional **I**maging in **RE**altime) besteht aus den beiden Komponenten RT-Server und RT-Client. RT-Server läuft auf der Control-Workstation des MR-Scanners und leitet die dort aufgenommenen Schichtbilder an den RT-Client weiter. Dieser Client ist für die Steuerung von FIRE, die Analyse und 2D-Anzeige der Schichtbilder zuständig. In dieser ursprünglichen Version ist der Realtime-Client ein monolithisches Programm, das sowohl für die Interaktion mit dem Benutzer (GUI) als auch für die Datenanalyse zuständig ist. Für dieses Projekt wurden diese Aufgaben getrennt. Während die

Berechnungen nun auf der CRAY T3E durchgeführt werden, erfolgt die Steuerung weiterhin von einer Workstation aus. Mit der Auslagerung der Rechenkomponenten auf einen Parallelerechner können in den Analyseprozeß weitere Schritte eingefügt werden, ohne dass die Abarbeitung in Echtzeit gestört wird. Im Rahmen des Projekts sind dazu Bildfilter, eine Bewegungkorrektur und die Referenzvektor-Optimierung in die Bearbeitungschiene eingebunden worden.

6.1.1 Verfahren der Analyse

Ziel der Datenanalyse ist es, die Bereiche des Gehirns zu markieren, die bei einem bestimmten Reiz (z.B. optisch, akustisch oder motorisch) aktiviert werden. Dabei nutzt man den BOLD-Effekt aus, der die Änderung der Blutsauerstoffsättigung bei neuronaler Aktivierung beschreibt.
Diese Änderung führt bei MR-Messungen zu Magnetfeldinhomogenitäten, die sich in Intensitätsänderungen in den Schichtbildern widerspiegeln. Zur Auswertung wird die gemessene Zeitreihe mit einer Modellzeitreihe (Referenzvektor) verglichen. Die Modellzeitreihe wird aus dem Zeitverlauf der Stimulation, die meistens durch ein einfaches On/Off-Paradigma beschrieben wird, und einem Modell der hämodynamischen Responsefunktion (Verlauf der Blutsauerstoffsättigung nach einem Reiz) durch Faltung gebildet. Eine übliche Methode für den Vergleich ist die Berechnung des Korrelationskoeffizienten aus den beiden Zeitreihen. Zur Verbesserung der Ergebnisse können Störanteile über „Detrending-Vektoren“ und andere Filtertechniken vermindert werden. Eine weitere Verbesserung der Ergebnisse erreicht man über eine sogenannte „Referenzvektoroptimierung“, die im Abschnitt 6.1.4 beschrieben wird. Für die statistische Auswertung werden dabei nur die Daten eines sogenannten „Sliding Windows“ benutzt, dessen Länge die Dauer einer Experimentwiederholung ist. Durch diese Technik nehmen der Rechenaufwand und der Speicherbedarf im Verlauf der Messung nicht zu.

6.1.2 Parallelisierung der Datenanalyse

Die Korrelationskoeffizienten werden in der Datenanalyse für jedes Pixel der Schichtbilder getrennt berechnet. Die Korrelationskoeffizienten hängen nur von den Meßdaten des Pixels und der Modellzeitreihe ab. Die Berechnungen für die einzelnen Pixel sind also unabhängig voneinander und können bei der Parallelisierung auf verschiedenen Prozessoren bestimmt werden. Für die Parallelisierung der Datenanalyse wird die Arbeit auf Basis der Pixel aufgeteilt. Jeder Prozessor bearbeitet nur einen Teil der Pixel eines Schichtbildes. Die Schichtbilder umfassen maximal 256x256 Pixel und belegen nur 512 KB Speicher. Die typische Breite eines Sliding Window ist 20-30 Zeitpunkte, so dass die Bilder maximal 15 MB Speicher belegen. Jedem Knoten auf der T3E stehen mindestens 128 MB Hauptspeicher zur Verfügung. Daher können Schichtbilder als ganzes an die Prozessoren verteilt werden. Die Bilder werden von dem Master-Knoten (Knoten 0) auf der T3E vom Socket entgegengenommen und an die die Slaves-Knoten (Knoten 1 bis n) verteilt. Jeder Prozessor bearbeitet nur eine Anzahl von Pixeln und trägt die Ergebnisse an den entsprechenden Stellen im Ergebnisbild ein. Die so berechneten funktionalen Teilbilder werden auf dem Master-Knoten zusammengefaßt und über Sockets an die GUI-Komponente weitergegeben. Jeder Prozessor bestimmt selbst, welche Bereiche des Bildes zum Gehirn gehören (nur diese werden bearbeitet) und welche Teile von ihm analysiert werden.

Bei Messungen mit der verteilten FIRE-Version wurden vom Scanner jeweils 16 Schichtbilder der Größe 64x64 Pixel aufgenommen und zur CRAY T3E übertragen. Dort wurden die Bilder mit einem Filter behandelt, einer Bewegungskorrektur unterzogen und die

Korrelation mit einem für jedes Pixel optimierten Referenzvektor (RVO) berechnet. Tabelle 6.1 zeigt den Zeitaufwand für die jeweiligen Teilschritte und den Speedup, den man beim Einsatz verschieden vieler Prozessoren (PEs) erhält. Die Messung zeigt, dass der Einsatz von bis zu 128 Prozessoren sinnvoll ist (skaliert). Den größten Anteil an der Gesamtzeit verbraucht die Optimierung des Referenzvektors.

Anzahl PE	Filter	Bewegung	RVO	Gesamtzeit	Speedup
1	0.18	1.55	109.27	110.00	1.0
2	0.09	0.91	54.65	55.65	2.0
4	0.05	0.56	27.36	27.97	4.0
8	0.03	0.46	13.74	14.23	7.8
16	0.02	0.35	6.93	7.30	15.2
32	0.02	0.33	3.51	3.86	28.7
64	0.03	0.35	1.85	2.22	50.0
128	0.04	0.34	1.00	1.37	81.1
256	0.04	0.40	0.59	1.01	110.5

Tabelle 6.1: Zeitaufwand für die Teilschritte der Analyse. Die Zeiten beruhen auf Experimenten, die auf der CRAY T3E-600 abliefen. Die Prozessoren sind mit 300 MHz getaktet und es stehen jeweils 128 MByte Hauptspeicher pro Knoten zur Verfügung.

Die Zeitspanne zwischen der Aufnahme der Schichtbilder im MR-Scanner und der Anzeige der Korrelationsbilder am Bildschirm (2D) kann wie folgt abgeschätzt werden. Bei 16 Schichten mit jeweils 64x64 Pixels erhält der RT-Server die Daten ca. 1,5 Sekunden nach der Messung. Die Zeiten für den Datenaustausch zwischen RT-Server, CRAY T3E und RT-Client summieren sich auf 1,1 Sekunden. Weitere 0,6 Sekunden werden für die Anzeige der Schichtbilder am Bildschirm benötigt. Mit 256 Knoten auf der T3E kann somit die Zeit für den kompletten Verarbeitungszyklus auf unter 5 Sekunden gebracht werden. In der aktuellen Version von FIRE werden die einzelnen Schritte nacheinander ausgeführt, d.h. es wird erst dann ein neues Bild vom RT-Server angefordert, wenn die Verarbeitung und Anzeige des vorherigen Bildes abgeschlossen ist. Der Durchsatz von FIRE ergibt sich somit als Summe der Verzögerungen im RT-Client und der CRAY T3E, also 2,7 Sekunden. Da der MR-Scanner unabhängig von FIRE arbeitet, ist auf der Seite des MR-Scanners eine Wiederholrate von 3 Sekunden möglich.

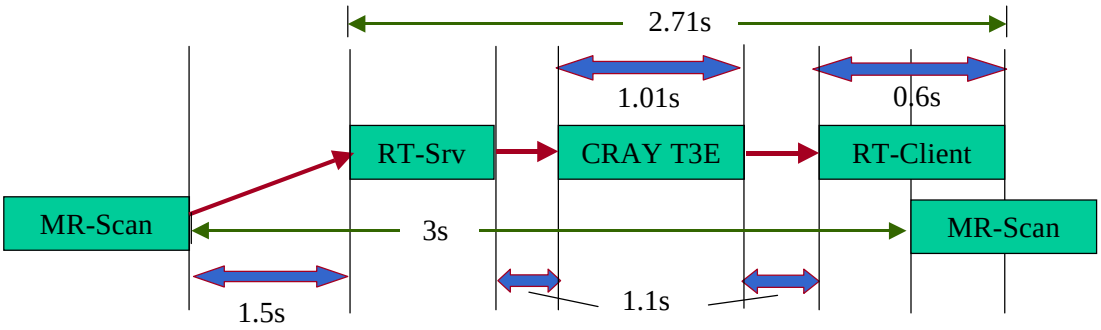


Abb 6.2: Zeitlicher Ablauf bei der Messsung

6.1.3 Filter und Bewegungskorrektur

Für den Test der räumlichen Bildfilterung wurden zwei Filter in FIRE implementiert, die im Calculator auf der T3E laufen. Beide Filter werden auf alle Pixel im Bild angewandt und verändern diese in Abhängigkeit zu deren Umgebung. Der Medianfilter sortiert die in einem Quadrat um das Pixel gelegenen Werte und ersetzt den Wert des Pixels durch den Wert, der an der mittleren Position der sortierten Reihenfolge liegt. Der Mittelwertfilter ersetzt den aktuellen Pixelwert durch das arithmetische Mittel der Pixelwerte im betrachteten Quadrat. Der Medianfilter beläßt im Bild die Kanten, reduziert aber das Rauschen, das durch einzelne Pixel entstanden ist. Er eignet sich daher für eine Filterung vor der Datenanalyse. Der Mittelwertfilter glättet das Bild und eignet sich eher zur Filterung nach der Datenanalyse.

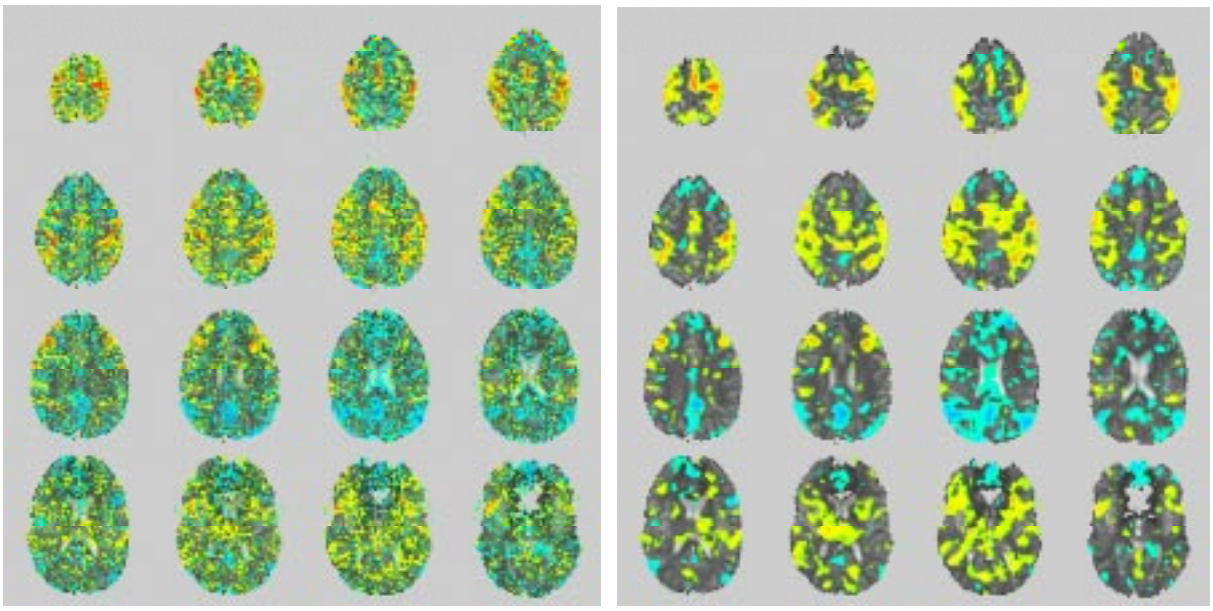


Abb 6.3: Räumliche Bildfilterung: Das linke Bild zeigt die Originaldaten. Das rechte Bild zeigt das Bild, nachdem vor der Korrelation ein 3x3 Median- und nach der Korrelation ein 3x3 Mittelwertfilter angewendet worden ist.

Ein großes Problem bei der Analyse von MR-Zeitreihen ist die Bewegung des Probanden. Die Bewegung des Kopfes schlägt sich natürlich auch in den einzelnen Schichtbildern nieder, so dass es zu Pixelverschiebungen kommen kann. Konkret bedeutet dies, dass ein Pixel nicht immer die gleiche Position im Gehirn des Probanden beschreibt. Eine Analyse kann dann schon bei einer kleinen Verschiebung zu verfälschten Ergebnissen führen (Artefakte). Im IME wurde deshalb ein Algorithmus entwickelt, der über ein Linear-Least-Squares-Verfahren die Bewegung des Kopfes zu einem Referenzbild berechnet. Im Rahmen des Projektes wurde dieser Algorithmus in FIRE integriert und für die CRAY T3E parallelisiert. Durch die Parallelisierung kann die Bewegungskorrektur in Echtzeit durchgeführt werden.

6.1.4 Referenzvektor-Optimierung

Bei der Workstation-Version des Programms wird die Korrelation der Meßdaten mit einem festen Referenzvektor (Modellzeitreihe) berechnet. Diese Modellzeitreihe wird zu Beginn der Messung festgelegt und jedes Pixel wird mit dieser Zeitreihe verglichen. Treten die Reaktionen auf den Reiz zum Beispiel etwas später ein als im Paradigma beschrieben, verringert sich der Korrelationskoeffizient. Mit der RVO werden zwei Parameter eines

Modells der hämodynamischen Responsefunktion variabel gehalten. Diese Parameter (Verzögerung und Dauer der Reaktion auf den Reiz) werden dann für jedes Pixel und zu jedem Meßzeitpunkt an die gemessenen Daten angepaßt. Durch dieses Fitting wird der Korrelationskoeffizient unempfindlich gegen Variation der Verzögerung und Dauer der Reaktion, was die Nachweisempfindlichkeit deutlich verbessert. Vorteilhaft ist zudem, die Referenzvektor-Optimierung jeweils für den Mittelwert einer Gruppe von Pixel (3x3, 4x4) durchzuführen und die einzelnen Zeitreihen der Pixel mit der für die Gruppe optimierten Modellzeitreihe zu vergleichen. So wird das Signal-zu-Rausch Verhältnis weiter verbessert.

Für die RVO wurde zunächst ein Konjugierte Gradienten Verfahren implementiert. Hier traten Probleme mit lokalen Minima auf. Daher wurde auf ein „Grid-Verfahren“ umgestellt, das wesentlich robuster ist. Es berechnet den Korrelationskoeffizienten auf einem hinreichend feinen Gitter im Parameterraum. Zwar ist dieses Verfahren wesentlich rechenintensiver, erreicht aber – bedingt durch wenige Hauptspeicherzugriffe und gute Cache-Ausnutzung – eine sehr hohe MFlops-Rate pro Prozessor. Eine Test mit 64PE der CRAY T3E zeigte, dass auch das Grid-Verfahren noch in Echtzeit eingesetzt werden kann.

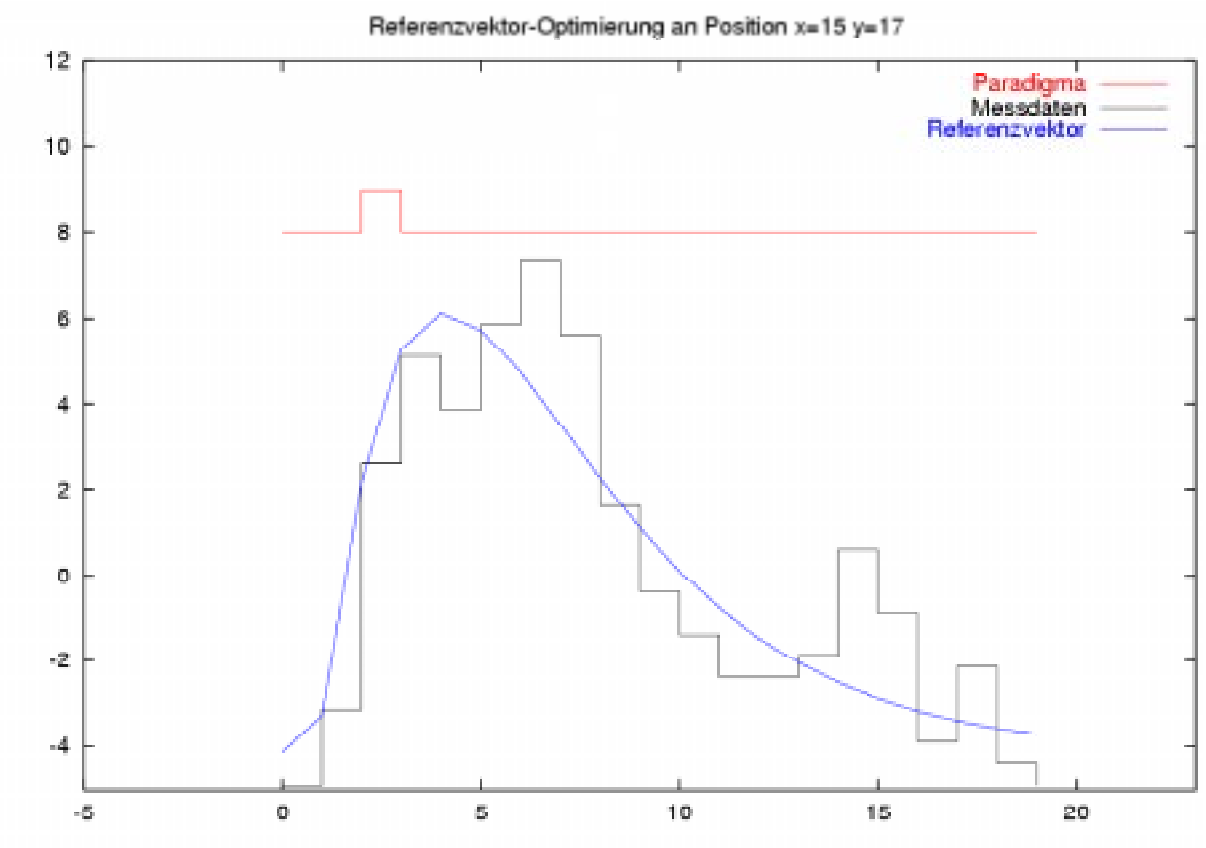


Abb. 6.4: Referenzvektor-Optimierung: Das Diagramm zeigt für ein Pixel den Inhalt des Sliding Window und die daran angepaßte Kurve des Referenzvektors.

6.2 Visualisierung

Wesentlicher Bestandteil des Projektes war die dreidimensionale Darstellung der Meßergebnisse. Neben der reinen Darstellung des Gehirn auf dem Bildschirm sollte auch eine Stereo-Darstellung auf einer Responsive Workbench möglich sein. Dazu wurde in den Visualisierungsprozeß der örtlich entfernte Visualisierungsserver SGI Onyx 2 in GMD eingebunden. Die folgenden Abschnitte stellen neben einem Prototypen für die Visualisierung die Implementierung der Visualisierung mit AVANGO, der Umlenkung der Bilddaten und Steuerinformationen vor.

6.2.1 Prototyp mit AVS/Express

Als Prototyp für die spätere Visualisierung auf der Workbench wurde mit der kommerziellen Graphik-Software AVS/Express (Advanced Visual Systems) eine dreidimensionale Stereodarstellung des Gehirns entwickelt. Diese kann für Demonstrationen eingesetzt werden, bei denen eine Workbench nicht zur Verfügung steht. AVS/Express bietet eine Vielzahl von Modulen zur Analyse und zur Darstellung von 2D- und 3D-Daten an. Für den 3D-Bereich stehen Möglichkeiten des interaktiven „Renderns“ von Volumendatensätzen zur Verfügung.

Die mit dem MRI-Scanner gemessenen Schichtbilder stellen einen solchen Volumendatensatz dar. Neben den für die funktionale Analyse vorgesehenen Schichtaufnahmen werden bei den

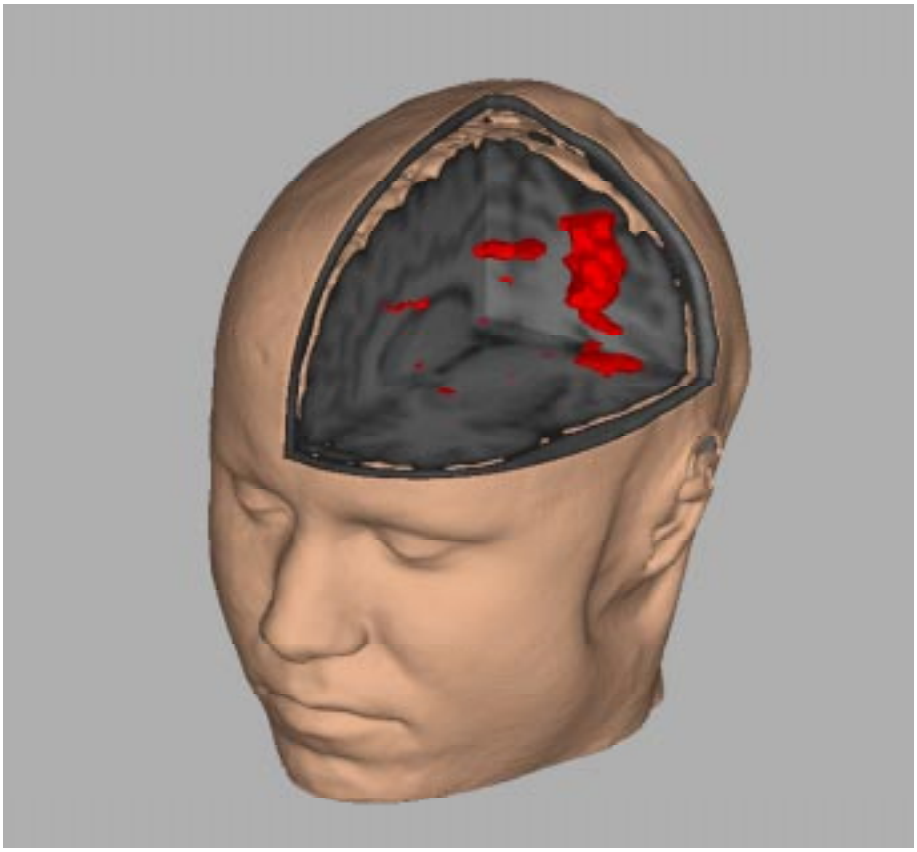


Abb 6.5: Mit AVS generierte Kopf-Darstellung mit MRI-Daten: Die roten Bereiche innerhalb des Gehirnvolumens zeigen die Aktivierung bei einer Bewegung der rechten Hand. Für die anatomische Darstellung stand ein Volumendatensatz mit 8.3 Mio. Voxeln zur Verfügung. Dieser Volumendatensatz wurde vorher mit einem 5x5x3 Mittelwertfilter geglättet und in AVS mit einer ISO-Oberfläche dargestellt. Die von FIRE berechneten funktionalen Daten lagen in 16 Schichten mit jeweils 64x64 Pixel vor. In FIRE war vor der Korrelation eine 3x3 Median- und nach der Korrelation ein 5x5 Mittelwertfilter eingeschaltet.

meisten Messungen vorher anatomische Aufnahmen gemacht, die den ganzen Kopf in einer hohen Auflösung (256x256x128 Pixel) erfassen. Im entwickelten AVS-Modell werden diese anatomischen Daten für die Darstellung des Kopfes benutzt.

Für den Aufbau der kompletten Graphik benötigt eine Sun Ultra 2 mit Creator3D-Graphik ein bis zwei Minuten. Da bei einer Datenänderung in AVS nur die im Netzwerk nachfolgende Module erneut aufgerufen werden, erfolgt die Aktualisierung der Darstellung bei neuen funktionalen Daten in Echtzeit. Manipulationen wie Rotation, Zoomen oder Schneiden des Modells sind hingegen in Echtzeit nicht möglich. Eine Stereo-Darstellung ist möglich, verringert aber zusätzlich die Leistung.

6.2.2 3D-Darstellung mit AVANGO

Die Schichtbilder der anatomischen Aufnahmen und der funktionalen Analyse werden auf dem Visualisierungsserver ONYX 2 IR mit 4 unabhängigen graphischen Subsystemen zu einer 3D-Darstellung aufbereitet. Diese im IMK/GMD installierte Maschine ist über das Gigabit-Netz mit der CRAY T3E und der Workbench im ZAM/FZJ verbunden und bietet genügend Leistung für die Manipulationen wie Rotation, Zoomen oder Schneiden des Modells in Echtzeit. Für die Aufbereitung der Bildinformation wird das in der GMD entwickelte VR-Visualisierungssystem AVANGO genutzt, das von der CRAY T3E in Jülich die Rohdaten über eine Socket-Verbindung erhält. Für diese Anbindung ist im Projekt ein Netzwerk-Modul entwickelt worden, das mit den Verarbeitungsmodulen von AVANGO verknüpft wird. Der Rohdatenstrom besteht aus anatomischen und funktionalen Daten, die mit einer Kennung versehen sind und somit vom Netzwerk-Modul wieder aufgespalten werden kann. Die Datenblöcke werden vom Netzwerk-Modul an die entsprechenden Module zur Weiterverarbeitung weitergegeben.

Der aus den einzelnen Schichten der anatomischen Aufnahme rekonstruierte Volumendatensatz wird in den 3D-Texturspeicher der Graphikhardware abgelegt. Damit ist es möglich, frei postionierbare Schnittebenen durch das Volumen zu legen und in Echtzeit zu bewegen. Für die Visualisierung des Kopfes werden dazu drei orthogonale und zwei frei positionierbarer Schnittebenen zur Verfügung gestellt. Die funktionalen Daten, die die aktiven Bereiche im Gehirn markieren, werden als Volumen in die Schnittebenen eingeblendet und während der Messung laufend aktualisiert.

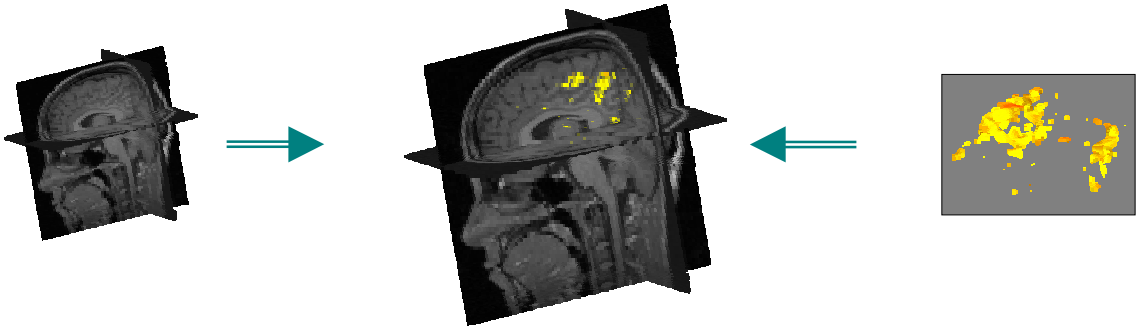


Abb 6.6: In AVANGO werden die anatomischen Daten als Schnittebenen (grau) und die funktionalen Daten als Volumen (gelb) dargestellt.

Mit AVANGO kann dieses Modell auf der Responsive Workbench dargestellt werden. Dazu werden Bilder für jeweils die obere und untere Projektionsfläche generiert. Für die Stereodarstellung, bei der jedem Auge ein einzelnes Bild zugeführt wird, müssen zudem jeweils zwei Bilder erzeugt werden.
Zur Steuerung der Visualisierung werden an der Workbench zwei Eingabegeräte benutzt: ein sogenannter Polhemus Stylus und ein weiteres Eingabegerät mit drei Tasten. Zusätzlich wird

die Augenposition des Betrachters über ein Head-Tracking-System festgestellt und bei der Generierung der Stereobilder berücksichtigt.

Um die Darstellung auf eine räumlich entfernte Workbench über das Gigabit-Netzwerk umzuleiten, müssen nicht nur die fertigen Bilder zu dieser übertragen, sondern auch die Signale der eben beschriebenen Eingabewerkzeuge zur GMD zurückgesendet werden. Kapitel 6.1.3 „Framebuffer-Umlenkung“ beschreibt die Implementierung und die Messungen zum ersten Teil der Aufgabe. Kapitel 6.1.4 „Umlenkung der Steuerdaten“ beschreibt die Implementierung zum zweiten Teil der Aufgabe.

6.2.3 Framebuffer-Umlenkung

In der SGI Onyx 2 wird bei der Stereodarstellung für jedes Auge ein getrennter Framebuffer benutzt. Dieser Speicherbereich der Grafikhardware enthält die Pixel-Information des fertig berechneten Bildes. Der Framebuffer wird für den Aufbau des Bildes auf dem Bildschirm bzw. der Workbench zyklisch ausgelesen. Soll das Bild nun nicht lokal dargestellt werden, muß der Inhalt des Framebuffers zur einer entfernten Workstation übertragen und dort in den entsprechenden Framebuffer abgelegt werden. Für das Auslesen des Framebuffers und das Speichern von Bildern in den Framebuffer stehen OpenGL-Befehle zur Verfügung.
Dabei muß dafür gesorgt werden, dass der Framebuffer während der Leseoperation nicht verändert wird. Daher muß das Auslesen mit dem Generieren neuer Bilder synchronisiert werden. Um optimale Performance zu erzielen, wird bei AVANGO das Lesen und Verschicken der Framebuffer von dedizierten Threads durchgeführt, die simultan mit den eigentlichen

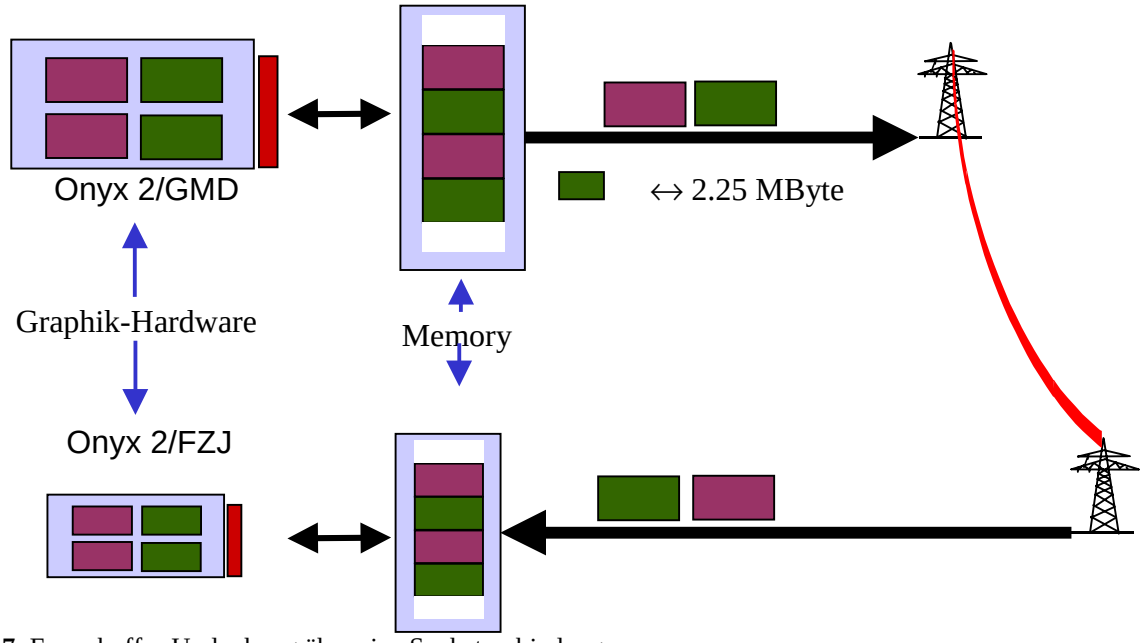


Abb 6.7: Framebuffer-Umlenkung über eine Socketverbindung.

Funktionen von AVANGO ausführt werden. Damit keine Probleme beim Zugriff von konkurrierenden Threads auf den Framebuffers auftreten, werden die Schreib- und Lese-Operationen mit Semaphoren synchronisiert. Wartezeiten werden durch die Swapbuffer-Technikvermieden, die zwei verschiedene Framebuffer für die Lese- und Schreiboperationen bereitstellt.

Zur Anzeige der Bilder im Forschungszentrum Jülich ist ein Receiver entwickelt worden, der die Daten der vier Framebuffer jeweils über einen eigenen Socket empfängt. Der Receiver besteht aus zwei Threads, von den der eine für die Überwachung der Sockets und der andere für die Darstellung der Framebuffer auf dem Bildschirm zuständig ist. Da beide Threads auf jeweils einem Prozessor laufen und die Daten zwischen den Threads gepuffert werden, erhält man eine flüssige Anzeige der Bilder.
Ein Framebuffer enthält bei eine Bildschirmgröße von 1024x768 bei 24 Bit Farbtiefe 2,25 MByte Daten. Für die Übertragung eines kompletten Frame auf der Workbench sind vier solcher Framebuffer nötig, was zu 9 MByte pro Frame führt. In Messungen konnte eine Übertragungsrate von 380 MBit/s erreicht werden. Dies entspricht einer Bildwiederholrate von 5 Frames/s bei der unkomprimierten Übertragung der Bilddaten. Begrenzender Faktor ist hier die aufwendige Generierung der Bilder in AVANGO. Da das Kommunikationsmodul nicht während der Generierung auf die Framebuffer zugreifen kann, entstehen Wartezeiten.
In Versuchen mit vorab gerenderten Bildern lassen sich mehr als 7 Frames/s übertragen und anzeigen, was einer Datenrate von 517 MBit/s entspricht. Dies ist ein Wert, der nahe an die maximalen über eine 622 MBit/s ATM Classical IP Netz 538 MBit/s herankommt. Eine LZO-Komprimierung der Bilddaten führt zwar zu eine erheblich geringeren Datenrate (je nach Bild bis zu einem Zehntel), setzt aber eine höhere CPU-Leistung des Graphikservers voraus. Bei Messungen mit AVANGO konnten hier nur bis zu 3 Frames/s gemessen werden.

6.2.4 Steuerdaten-Umlenkung

Für die Nutzung eines entfernten Graphikservers reicht es nicht aus, das Rendering dorthin auszulagern und die fertigen Bilder lokal anzuzeigen. Zudem müssen die Signale der lokal an der Workbench vorhandenen Eingabegeräte an den entfernten Visualisierungsserver gesendet werden. Im Wesentlichen sind das die Positionen und Rotationwinkel des Stereobrillensensors und der Eingabegeräte (Polhemus Stylus, 3Button Device). Der Stereobrillensensor bestimmt die Position und Blickrichtung des Betrachters und erlaubt so die Generierung einer für den Betrachter perspektivisch richtigen Projektion der Szene. Der Polhemus Stylus ist ein Eingabestift, der in der virtuellen Szene als Laserpointer fungiert. Er besitzt einen Button, mit dem Objekte in der Szene selektiert und bewegt werden können. Das 3Button Device wird meistens fest mit einem Objekt verknüpft, so dass jede Bewegung des Devices zu einer Bewegung des Objekts führt. Mit den drei Buttons können beliebige Aktionen ausgeführt werden.
Zu übertragen sind somit Positionen und Drehwinkel von drei Sensoren sowie die Klick-Event der vier Buttons. Auf der lokalen Seite läuft dazu ein Sender, der die Daten der Eingabegeräte zyklisch ausliest, zu einem Datenpaket zusammenfaßt und über eine Socket-Verbindung zu AVANGO schickt. Dort werden die Daten wieder aufgeteilt und in die entsprechenden Variablen von AVANGO geschrieben. Insgesamt werden hierbei nur wenige Daten übertragen. Trotzdem ist hier eine geringe Latenz wichtig, da die Aktionen mit den Eingabegeräten synchron zu der Anzeige auf dem Bildschirm ablaufen müssen.

6.3 Zusammenfassung

Durch die Aufteilung und Parallelisierung des FIRE-Clients ist es nun möglich, die Analyse der Meßdaten zu verfeinern, ohne den Echtzeit-Charakter der Anwendung zu verlieren. Die im Projekt zusätzlich eingebrachten Analyseschritte, wie die Bewegungskorrektur und Referenzvektor-Optimierung, verbessern die Ergebnisse der Analyse, benötigen dabei aber eine Rechenleistung, die derzeit von handelsüblichen Workstations nicht erbracht wird. Nur

durch den Einbezug des Parallelrechners CRAY T3E und dessen Anbindung über eine schnelle Netzverbindung war die Erweiterung der Analyse möglich.
Der zweite Aspekt in diesem Projekt war, auch für die Visualisierung einen speziellen Visualisierungsserver zu benutzen. Auch dieser ist nicht lokal vorhanden, aber über das schnelle Gigabit-Netzwerk des Testbeds zu erreichen. Gerade bei der Übertragung von Bildinformation ist eine hohe Bandbreite wichtig. Im Prinzip ist eine Software-Komprimierung der Daten zwar möglich. Tests haben aber gezeigt, dass dazu auf beiden Seiten der Übertragung eine hohe CPU-Leistung notwendig ist, die damit nicht mehr der Graphikanwendung zur Verfügung steht. Eine Komprimierung reduziert zwar die Datenrate, aber auch die Rate, mit der die Bilder übertragen werden. Mit einem Gigabit-Netz ist die Komprimierung nicht notwendig. Neben der Bandbreite stellt auch die Latenz bei der Übertragung der Steuersignale ein wichtiges Maß dar. Bei einem schnellen Netz ist diese gering und erlaubt damit ein synchronen Ablauf der Visualisierung.
Das Projekt hat gezeigt, dass in diesem aufgezeigten Fall des heterogenen Metacomputing mit einem schnellen Netz die Verbindung der Einzelkomponenten implementiert und erfolgreich betrieben werden konnten.

7 Multimediale Anwendungen im Gigabit-Testbed

Beteiligte Partner: *Institut für Medienkommunikation (IMK/GMD)*
Zentralinstitut für Angewandte Mathematik (ZAM/FZJ)

Ansprechpartner: *Ulrich Nütten (Leiter Digital Media Productions, IMK/GMD)*
Weitere Beteiligte: *Georg Glock (IMK/GMD), Arnfried Griesert (IMK/GMD), Dr. Manfred Kaul (IMK/GMD), Kirstin Krüger (IMK/GMD), Wolfgang Vonolfen (IMK/GMD)*

7.1 Zielsetzung des Projektes

In der professionellen Videoproduktion werden hohe Ansprüche an die Qualität der produzierten Bilder gestellt. Diesen Qualitätsanforderungen muß von allen Produktions- und Übertragungsverfahren entsprochen werden, wenn sie im professionellen Broadcast eingesetzt werden sollen.
Im Rahmen des Europäischen ACTS-Projekt AC 089 "Distributed Video Production" hat die GMD als Projektleiter zusammen mit 16 internationalen Partnern Anwendungsszenarien mit verteilten Produktionsumgebungen entwickelt und getestet. Wesentliches Beurteilungskriterium waren dabei die hohen Qualitätsansprüche der Anwendungspartner aus der TV-Industrie.

Im Rahmen des Gigabit-Testbeds sollten die gewonnenen Erfahrungen in der Form eingebracht werden, dass im Rahmen von verteilten Videoproduktionen im Gigabit-Netz geeignete Endgeräte wie Encoder, Decoder und ATM-Adapter getestet wurden und dabei die oben angesprochenen Qualitätskriterien zugrunde gelegt werden sollten. Ergebnis dieses Projekts sollte eine begründete Aussage darüber sein, inwieweit die heutigen Geräte in Bezug auf Bildqualität und Delay professionelle Videoproduktionen in verteilten Umgebungen erlauben.

Um eine solche Aussage machen zu können, wurden die am Markt erhältlichen Produkte für die Übertragung von hochqualitativem Video in 622 Mbps-Netzen evaluiert und ausführlichen Versuchen unterzogen. Anwendungsszenario für die Tests war im wesentlichen ein verteiltes Virtuelles Studio, dass auf dem von der GMD entwickelten Virtual Studio-System "3DK" basiert.

7.2 Anforderungen an die Qualität von Videoübertragungen

- Sendetaugliche Signalqualität für Audio-/ Videoübertragung
  Gerade die öffentlich-rechtlichen Rundfunkanstalten in Deutschland sind sehr kritisch was die Einhaltung der vorgegebenen Normen für die Bildqualität angeht. Selbst wenn mit bloßem Auge keine Beeinträchtigungen der Signalqualität wie Artefakte oder Jitter sichtbar waren, war die Qualität für sie nicht akzeptabel, wenn die Meßwerte außerhalb dieser Normwerte lagen.
- Ausfallsicherheit
  Es ist selbstverständlich, dass für Live-Sendungen ein Bildausfall infolge eines Defektes einer der Übertragungskomponenten inakzeptabel ist. Aber auch für aufgezeichnete Produktionen sind solche Störungen kritisch, da die Produktionszeiträume aus Aktualitäts- oder Kostengründen mitunter extrem knapp bemessen sind und technische Ausfälle die

gesamte Produktion gefährden können. Insofern kommt diesem Punkt besondere Wichtigkeit zu.

- Echtzeit: Übertragung ohne produktionsbehindernde Verzögerung
  Eine Videoverzögerung von über 2-3 Frames ist im Produktionsbetrieb hinderlich, da alle anderen Video- und Audioquellen (zum Beispiel von externen Zuspielern) im gleichen Maße verzögert werden müssen. Für den Moderator ist es sehr störend, den eigenen Ton über Lautsprecher verzögert zu hören.
- Interaktiv: Übertragung in beide Richtungen mit sehr kleiner Verzögerung
  Soll der Moderator mit virtuellen Objekten, beispielweise mit einem virtual actor, interagieren, so ist dies nur mit sehr kleiner Verzögerung möglich, da sonst gemeinsame koordinierte Aktionen nicht synchronisiert werden können.

7.3 Durchgeführte Arbeiten

Da für Investitionen keine Projektmittel vorgesehen waren, konnten nur solche Geräte einem Test unterzogen werden, die der Hersteller der GMD leihweise zur Verfügung stellen konnte. Dabei handelte es sich teilweise um Vorserienmodelle und handgelötete Prototypen.

Getestet wurden folgende Geräte:

Codecs:
MMS SCSI Video mit MJEP-Codierung bis 70 Mbps
Fore Nemesys Codecs mit MJEG-Codierung bis 100 Mbps
Sony BDX 1000-Serie mit MPEG 2 4:2:2 profile mit 50 Mbps

ATM-Adapterkarten:
Fore Runner HE 622
SGI-ATM-Karte

7.3.1 Codecs

Alle Codecs konnten in verteilten Produktionsszenarien im Laborbetrieb eingesetzt werden, für sendetaugliche Produktionen eigneten sie sich jedoch nicht:

- Die Bildqualität erreichte nicht professionelle Zuspielqualität
  Die für den Studiobetrieb vorgeschriebenen Pegel und Grenzwerte wurden nicht eingehalten.
- Mangelnde Stabilität einiger Geräte
  Aufgrund des Vorseriencharakters einiger Geräte war ihr Betrieb instabil, es kam in unregelmäßigen Abständen zu Bildausfällen.
- Cell Delay Variation > 0,08 sec
  Unterschiedliche Verzögerungen bei der Übertragung der ATM-Zellen können auf der Decoder-Seite nicht kompensiert werden, da der Pufferspeicher zu gering ist.
- Verzögerung bei En-/Dekodierung ist teilweise zu groß
  Einige Geräte verzögerten das Videosignal infolge der Komprimierung/Dekomprimierung um mehr als 1 Sekunde. Dies ist in verteilten Produktionsumgebungen nicht praktikabel, so sieht z. B. der Kameramann, der das gemischte Bild auf seinem Kamerasucher hat, erst mit dieser großen Verzögerung, wo seine Kamera in der virtuellen Kulisse hinschaut. Zudem muss der Ton um die gleiche Zeitspanne verzögert werden, was auf der Audio-Seite ebenfalls zu Problemen führt. Die Verzögerung durch die ATM-Strecke selber, also die eigentliche Übertragungszeit, ist sehr gering (< 1 Field) und somit vernachlässigbar).

7.3.2 ATM-Adapter

Die ATM-Adapter bilden bei einem Remote Rendering Szenario im Virtuellen Studio (siehe Anhang) die Schnittstelle vom Grafikhochleistungsrechner zum ATM-Netzwerk und damit zum Compositer, sie sind ein wesentliches Element der Produktionskette. Deshalb müssen sie genauso den Anforderungen an eine professionelle Videoproduktion genügen wie Codecs oder andere Geräte. Getestet wurden die beiden oben genannten Karten, die zwar beide als PCI-Boards ausgeführt sind, jedoch unterschiedlich implementiert werden:
Die Fore-Karte wird über einen Adapter, der bis zu drei PCI-Boards aufnehmen kann, in den Grafikrechner gesteckt, während die SGI-Karte ausschliesslich mit einem speziellen Adapter geliefert wird, der nur diese Karte aufnehmen kann.

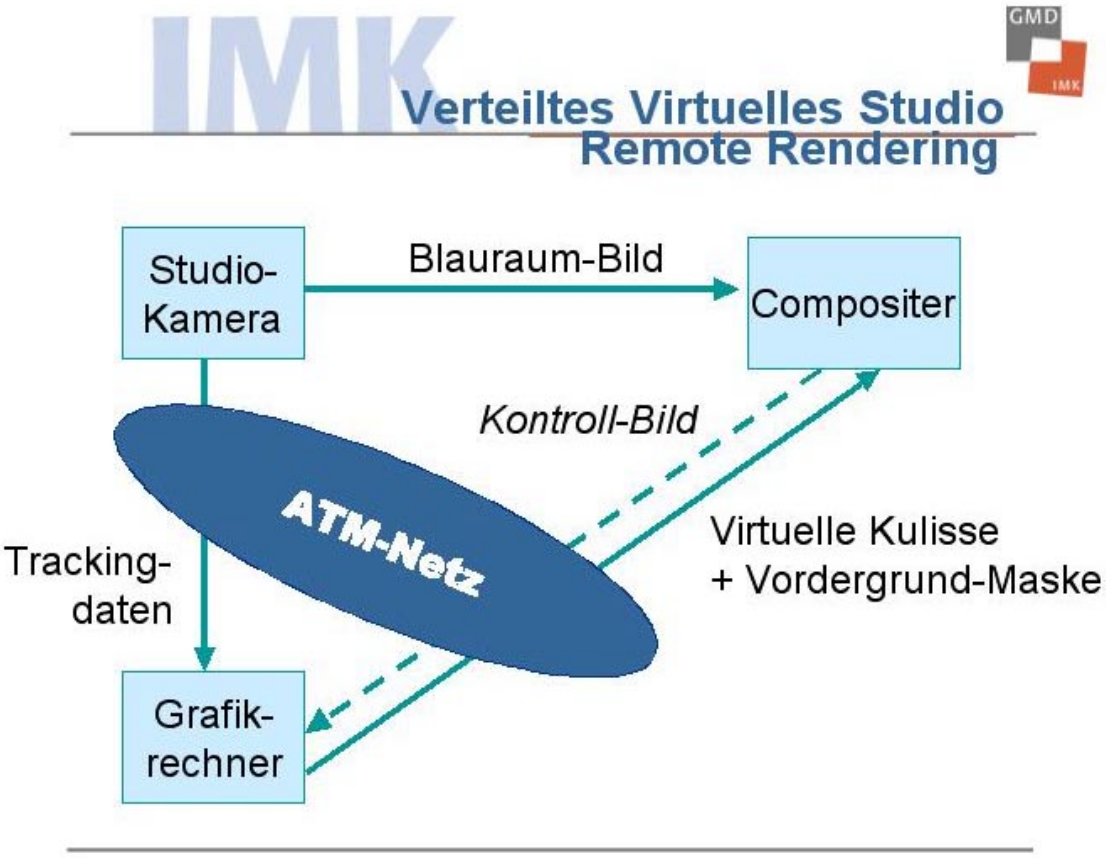
Diese verschiedene Ausführung führte zu unterschiedlichen Ergebnissen bei Dauertests in einem Grafikhochleistungssystem Typ SGI Onyx 2. Die Fore-Karte blockierte nach einigen Minuten das gesamte System und brachte den Rechner in einen undefinierten Zustand. Vermutlich kann die Fore-Karte die Daten nicht schnell genug über den Adapter in den Bus transferieren, was dazu führt, dass der Puffer überläuft und das Board blockiert.

Die unter gleichen Bedingungen getestete SGI-ATM-Karte mit dem speziellen Adapter funktionierte problemlos, sie übertrug einen unkomprimierten digitalen Videostrom (270 Mbps) einwandfrei.

7.4 Zusammenfassung

Mit den getesteten Geräten sind professionelle Produktionen in verteilten Umgebungen nur mit großen Einschränkungen machbar. Die mangelnde Bildqualität und Betriebsstabilität lassen einen Einsatz in diesem Umfeld nicht zu. Die teilweise sehr hohen Verzögerungen durch die Komprimierung und Dekomprimierung sind in der Praxis nicht akzeptabel. Testergebnisse wie bei der Fore-Karte zeigen, dass manche Entwicklungen offensichtlich sehr schnell ohne ausreichende Tests auf den Markt gebracht wurden.

Für eine abschliessende Gesamtbeurteilung ist allerdings folgendes zu beachten:
Zum Zeitpunkt der Durchführung der Tests (Ende 98/1. HJ 99) war die Auswahl an geeigneter Komprimierungs-Hardware noch recht gering, viele Entwicklungen befanden sich noch im Prototypenstadium. Insofern können die Testergebnisse nur eine Momentaufnahme der am Markt befindlichen Geräte zum Zeitpunkt der Testphase widergeben. Die im Rahmen des Projekts durchgeführten Marktbeobachtungen zeigen, dass nachfolgende Entwicklungen deutliche Verbesserungen insbesondere bei Stabilität und Bildqualität aufweisen. Auch sind inzwischen deutlich mehr Geräte erhältlich, so z. B. von Leitch, Tiernan, Cellware u.a.. Diskussionen und Geräte-Demonstrationen auf Messen haben ergeben, dass damit Videoübertragungen in sendetauglicher Qualität möglich sein sollten. Nicht ganz befriedigend gelöst ist nach wie vor das Problem der Verzögerung, hier liegen die Werte noch zu hoch, um wirklich interaktive verteilte Anwendungen durchführen zu können. Es ist aber absehbar, dass die Geräte-Entwicklung mittelfristig durch leistungsfähigere Hardware akzeptable Größenordnungen erreichen kann.


IMK
GMD
IMK
Verteiltes Virtuelles Studio
Remote Rendering
Studio-
Kamera
Blauraum-Bild
Compositer
Kontroll-Bild
ATM-Netz
Virtuelle Kulisse
+ Vordergrund-Maske
Tracking-
daten
Grafik-
rechner

8 Verteilte Berechnung von Klima- und Wettermodellen

Beteiligte Partner: *Alfred Wegener Institut für Polar- und Meeresforschung (AWI)*
Deutsches Klimarechenzentrum (DKRZ)
Institut für Algorithmen und Wissenschaftliches Rechnen (SCAI/GMD)
Zentralinstitut für Angewandte Mathematik (ZAM/FZJ)

Ansprechpartner: *Dr. Wolfgang Hiller (AWI), Dr. Wolfgang Joppich (SCAI/GMD)*
Weitere Beteiligte: *Dr. Stephan Frickenhaus (AWI), Dr. Thomas Störtkuhl (AWI),*
Dr. Bernadette Fritzsch (AWI), Dr. Olaf Heudecker (AWI),
Dr. Udo Göbel (AWI), Dr. René Redler (AWI/GMD),
Johannes Quaas (GMD)

8.1 Überblick

Zur Erforschung des Klimasystems der Erde, insbesondere zur Vorhersage des Klimawandels, dienen als genauestes Werkzeug physikalische Modelle der Klimakomponenten. Zur Berücksichtigung möglichst vieler Einflüsse ist es notwendig, alle Subsysteme miteinander zu koppeln. Im Teilprojekt "Verteilte Berechnung von Klima- und Wettermodellen" sollte gezeigt werden, dass sich eine Kopplung von state-of-the-art-Modellen auf einem Metacomputer mit leistungsfähigen Weitverkehrsnetzen effizient durchführen läßt.
Im folgenden Abschnitt werden die benutzten Werkzeuge noch einmal kurz vorgestellt. Unter 8.3 wird über die im letzten Berichtszeitraum erfolgten Arbeiten berichtet, und im letzten Abschnitt werden die Ergebnisse des Teilprojektes zusammengefaßt.

8.2 Modelle

Atmosphärenmodell

Als Atmosphärenmodell diente das an der GMD aufgrund früherer Arbeiten bekannte Modell IFS ("Integrated Forecast System", in der Météo-France Version "ARPEGE"), das vom Europäischen Zentrum für Mittelfristige Wettervorhersage (ECMWF) in Zusammenarbeit mit dem französischen Wetterdienst Météo-France entwickelt wurde. Das IFS wurde vom ECMWF u.a. im AMIP (Atmosphere Model Intercomparision Project) für Klimasimulationen eingesetzt. Für unser Projekt benutzten wir die Version CY16R2 von 1997. Aufgund von Lastbalance-Überlegungen entschieden wir uns für die relativ geringe, in Klimarechnungen jedoch nicht unübliche Auflösung des globalen Atmosphärenmodells von T21L19, was einem horizontalen Gitter von 64*32 Punkten bei 19 vertikalen Schichten entspricht.

Ozeanmodell und Seeismodell

Das MOM (Modular Ocean Model) wurde am GFDL (Geophysical Fluid Dynamics Laboratory in Princeton) entwickelt und seither an zahlreichen Instituten zur Simulation der ozeanischen Zirkulation eingesetzt. Im Projekt wurde eine am AWI um ein Seeismodell erweiterte Version MOM 2 benutzt, die auf der T3E mit SHMEM parallelisiert vorliegt. Als Auflösung des globalen MOM wurde ein horizontales Gitter von 194*92 Punkten bei 29 Schichten gewählt.

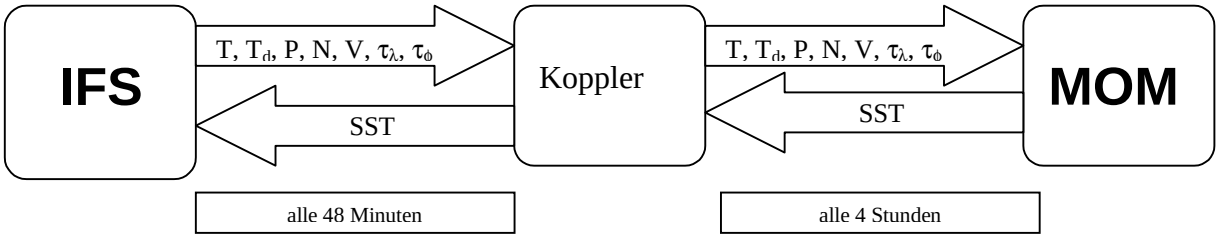


Abb. 8.1: Datenaustausch zwischen den Modellen

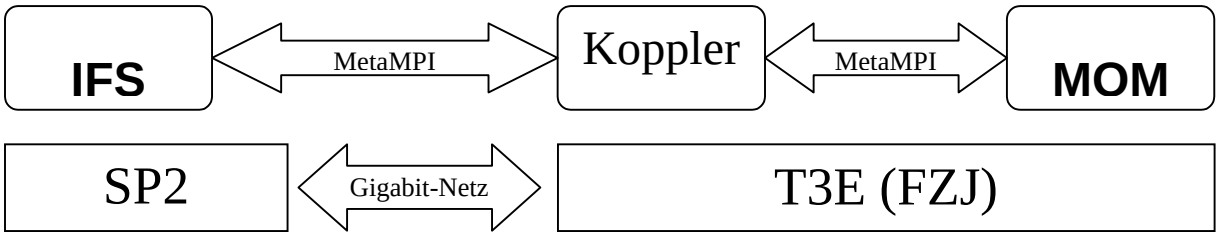


Abb. 8.2: Die Verteilung der Modelle auf die Hardware.

Kopplung

Die beiden Teilmodelle benötigen für die zeitliche Integration der dynamischen Gleichungen Randbedingungen, die an der gemeinsamen Schnittfläche, der Meeresoberfläche, das jeweils andere Modell liefert. Das IFS erhält dabei von MOM die Meeresoberflächentemperatur (sea surface temperature, SST), das MOM den Windantrieb (Windstress in meridionaler und zonaler Richtung sowie Windgeschwindigkeit), Temperatur und Taupunktstemperatur der untersten Modellschicht sowie den Gesamtniederschlag und die Bewölkung (vgl. Abbildung 8.1). Den Datenaustausch und die Interpolation zwischen den unterschiedlichen Gittern steuert ein weiteres Programm, der Koppler. Hierzu wurde der CSM flux coupler des NCAR angepaßt. Die Kommunikation der drei Programme findet mit Hilfe der von Pallas entwickelten MPI-Version "MetaMPI" statt. Die interne Kommunikation des Atmosphärenmodells wird ebenfalls über MetaMPI abgewickelt. Die Parallelisierung von MOM arbeitet mit der CRAY-spezifischen SHMEM-Bibliothek. Zur Verteilung der Modelle auf die zur Verfügung stehende Hardware wird auf die obige Skizze (Abbildung 8.2) verwiesen.

8.3 Arbeitsschritte im Berichtszeitraum

Debugging von MetaMPI

Das Problem, dass man mit MetaMPI beim von IFS benötigten buffered send nur Pakete bis zu einer Größe von 128 KB verarbeiten konnte, wurde gelöst. Dazu wurde das Problem auf ein kleines Testprogramm reduziert, das den Fehler eindeutig identifizierte. Mit dessen Hilfe konnte die Firma Pallas in Zusammenarbeit mit den Entwicklern des IBM-MPIch den Fehler in MetaMPI beheben.
Da das IFS auf der IBM-SP2 auch für die interne Kommunikation MetaMPI benutzt, war erst nach dieser Fehlerbehebung eine Modellkopplung möglich, bei der beide Modelle parallel laufen.

Anpassungen des Kopplers

Die Gitterdefinitionen des MOM und des Kopplers einerseits und des IFS andererseits sind unterschiedlich: bei den ersten laufen die Breiten von Süden nach Norden, beim IFS umgekehrt von Norden nach Süden. Die einheitliche Behandlung der versendeten Felder wurde in die Interpolationsroutinen des Kopplers eingebaut.
Weitere Änderungen in den Interpolationsroutinen des Kopplers wurden dadurch notwendig, dass der NCAR CSM-Koppler ein geschachteltes Gitter erwartet. Das IFS als Spektralmodell benutzt nur ein Gitter, auf dem die Werte in den Knoten vorliegen. Ein zusätzliches „Kanten"-Gitter mußte also definiert werden, um die Routinen des Kopplers weitgehend unangetastet zu lassen. Durch die Definition der Längen des IFS-Gitters, mit dem Nullmeridian beginnend, mußte der Koppler auch für negative Längen angepaßt werden.
Um das gekoppelte Modell nicht nur mit Initialisierungs- sondern auch mit Restart-Datensätzen als Anfangswerten integrieren zu können, waren in den Kommunikationsroutinen des IFS weitere Anpassungen nötig: Das IFS benötigt einen Modellzeitschritt, um sich selbst zu initialisieren, und kann erst dann neue Daten an das Ozeanmodell schicken.

Startup-Skripte

Wegen der unterschiedlichen job-scheduling-Systeme auf der SP2 der GMD und der T3E des FZJ ist ein interaktiver Start der Modelle notwendig. Hierzu wurden Skripte geschrieben, die dem Benutzer einen Großteil der Arbeit abnehmen. Dennoch ließen sich insbesondere bei den längeren Integrationen zahlreiche Schwierigkeiten nicht vermeiden, die z.B. aus Eigenheiten der secure shell resultierten. Problematisch ist die unterschiedliche Handhabung der Zuteilung interaktiver Rechenzeit, so ist z.B. nur nach spezieller Freigabe eine interaktive Nutzung der T3E nachts möglich, andererseits ein längerer Lauf tagsüber wegen der Konfiguration der SP2 nicht möglich. Hier besteht für künftige Metacomputing-Projekte eine wesentliche Verbesserungsmöglichkeit der Produktions- und Entwicklungsumgebungen.

Modellauf

Es wurde eine Integration des gekoppelten Modells über insgesamt acht Monate durchgeführt, indem jeweils nach einem Monat das Modell mit dem Herausschreiben einer restart-Datei beendet und anschließend ausgehend davon neu gestartet wurde, um Probleme mit der Jobverwaltung zu umgehen.
Das IFS wurde dabei mit Analysedaten des ECMWF vom 1.1.1997 initialisiert. Der Zeitschritt des IFS betrug 48 Minuten, nach jedem Zeitschritt fand ein Datenaustausch mit dem Koppler statt. Die Randdaten, die der Koppler dem IFS zur Verfügung stellte, blieben dabei über einen Zeitraum von fünf Modellschritten (entsprechend einem Ozeanzeitschritt) konstant.
Die Temperatur und der Salzgehalt des MOM wurden mit klimatologischen Daten nach Levitus initialisiert, der Ozean befand sich zu Beginn der Integration in Ruhe. Der Zeitschritt des MOM betrug 4 Stunden. Die Randdaten des Ozean wurden daher aus dem Mittelwert der fünf Felder gebildet, die das IFS während eines Ozeanzeitschritts an den Koppler gesendet hatte.
Bei jeder Kopplung mit dem Ozeanmodell wurden vom Koppler jeweils alle versendeten Felder herausgeschrieben, um den Datenaustausch darstellen zu können. In Abbildung 8.3 ist ein Beispiel für die insgesamt acht zwischen den Modellen ausgetauschten Gitterpunktsfelder dargestellt. Die ersten sieben Grafiken zeigen die vom IFS an das Ozeanmodell gesendeten Felder: Die Niederschlagsrate in m/s, die Bedeckung des Himmels mit Wolken als Zahl

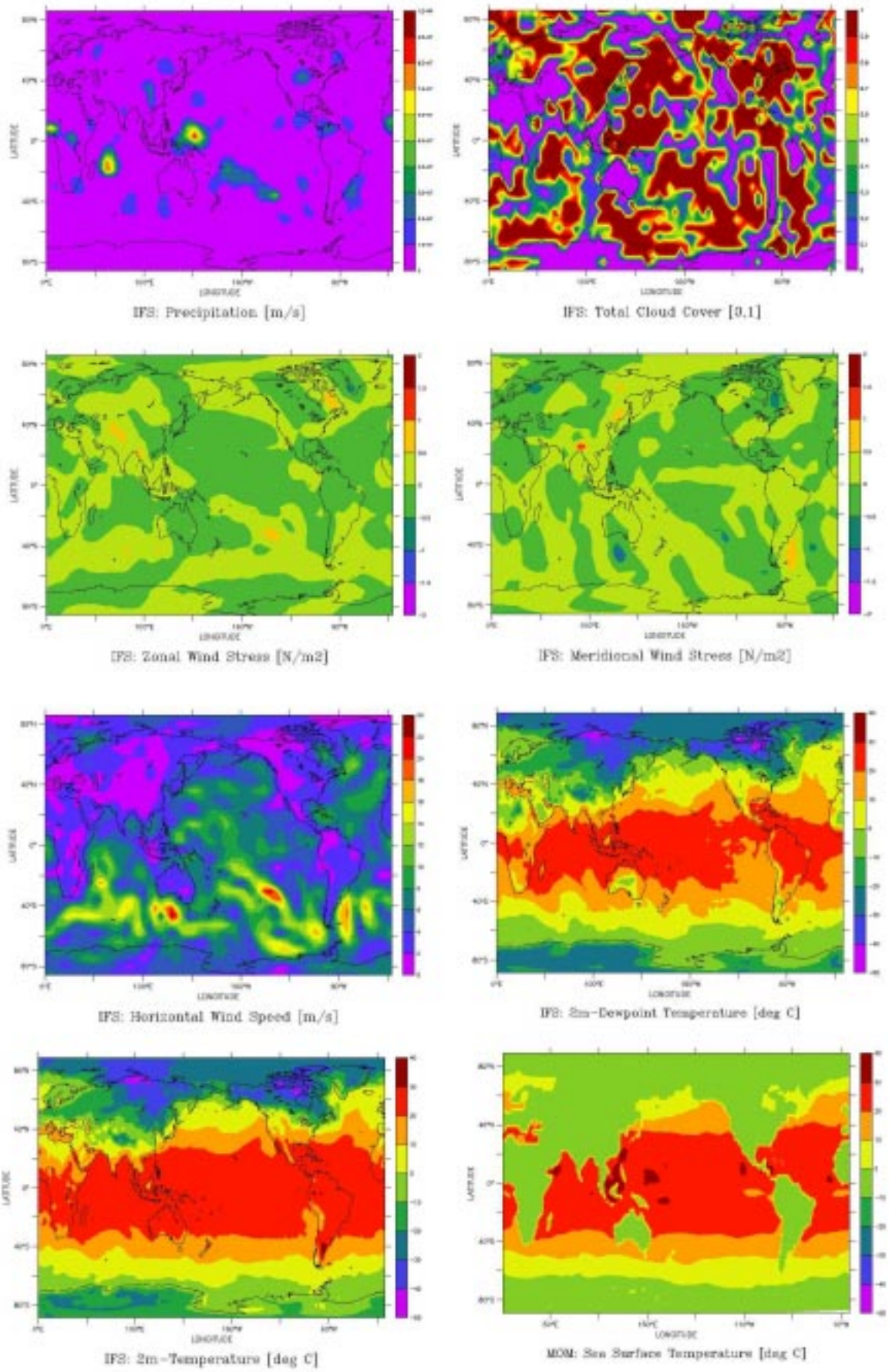


Abb. 8.3 : Die versendeten Felder

zwischen 0 und 1, die Schubspannung in zonaler und meridionaler Richtung, jeweils in N/m², die Geschwindigkeit des Horizontalwindes in m/s, die Taupunktstemperatur in 2 m Höhe über Grund in °C sowie die Temperatur in 2m Höhe in °C. Die achte Grafik stellt die Meeresoberflächentemperatur, die das MOM an das IFS sendet, in °C dar. Hierbei ist die Landmaske durch Werte von 0°C codiert.
Für die spätere Visualisierung wurden zudem vom Ozeanmodell die Oberflächenströmung und vom Atmosphärenmodell ein Datensatz, u.a. mit dem horizontalen Wind und der Temperatur der untersten Modellschicht, jeweils nach 12 Stunden Modellzeit, gespeichert.

8.4 Ergebnis

Nachdem im Laufe des Projektes die rechnerübergreifende Kommunikationsbibliothek MetaMPI entwickelt wurde, standen dem verteilten Rechnen keine großen Hindernisse mehr im Weg – parallel laufende Anwendungen lassen sich relativ leicht auf mehrere Rechner verteilen. Zu Beginn des Projekts stellte dies die größte Hürde dar, da MetaMPI noch nicht entwickelt war, und PACX die spezielle Anforderung des Teilprojekts „Verteilte Berechnung von Klima- und Wettermodellen“, mindestens drei verschiedene Programme zu koppeln, nicht erfüllen konnte. Auch für MetaMPI stellte das Teilprojekt in vielen Hinsichten die Pilotanwendung dar.
Der Aufwand zur Definition der Schnittstellen für die beiden Klimamodelle des Ozeans und der Atmosphäre entsprach den Erwartungen, in vielen Aspekten ließen sich durch die Einschränkung auf serielle Kopplung die Schnittstellen der Modelle für das Einlesen von Randbedingungen aus Dateien adaptieren.
Im Rückblick stellt man fest, dass mit dem NCAR CSM flux coupler ein Koppler gewählt wurde, der sehr stark auf die Anwendung des Climate System Model des NCAR zugeschnitten war. Dies trifft insbesondere auf die Fähigkeit des Kopplers zu, selbst Flüsse zu berechnen, da diese Flußberechnungen feste Bestandteile eines Klimamodells sind und sich nicht ohne weiteres verallgemeinern lassen. Deshalb waren wesentliche Eingriffe in den Koppler notwendig. Im Nachhinein erscheint es als nicht sinnvoll, an dem Konzept eines eigenen Programms, das die Kopplung steuert, festzuhalten; für Folgeprojekte wird daran gedacht, die Modelle direkt zu koppeln und als Kopplungschnittstelle beispielsweise die an der GMD entwickelte Bibliothek MpCCI zu verwenden.

Eine Integration über mehrere Monate wurde erfolgreich durchgeführt, die Ergebnisse sind plausibel. Verbleibende Mängel sind vermutlich darauf zurückzuführen, dass für das MOM aus Zeitgründen kein spinup durchgeführt wurde. Bei diesem würde das Modell zunächst allein bis zu einem gewissen Gleichgewicht integriert. Diese Fragen werden noch in einer aus dem Projekt hervorgegangenen Diplomarbeit untersucht.

9 Portierung von Anwendungen aus dem CISPAR-Projekt

Beteiligte Partner: *Pallas GmbH*
Institut für Algorithmen und Wissenschaftliches Rechnen (SCAI/GMD)
Zentralinstitut für Angewandte Mathematik (ZAM/FZJ)

Ansprechpartner: *Karl Solchenbach (Pallas), Klaus Wolf (SCAI/GMD)*
Weitere Beteiligte: *Kläre Cassirer (SCAI/GMD), Matthias G. Hackenberg (SCAI/GMD), Klaus-Dieter Oertel (Pallas), Dr. Peter Post (SCAI/GMD)*

9.1 Zielsetzung

Ziel des Teilprojekts war die Demonstration einer gekoppelten Simulation auf dem Metacomputer bestehend aus einer CRAY T3E am Forschungszentrum Jülich und der IBM SP2 der GMD (Abb.9.1). Aus dem EU ESPRIT Projekt CISPAR (http://www.pallas.de/outpage/out-cis.htm) wurde hierzu das multidisziplinäre Testbeispiel *Bending Flap* ausgewählt, wobei es sich um die Simulation einer einseitig fixierten, verformbaren Klappe in einem Flüssigkeitskanal handelt. Die Strömungsberechnung wird auf der T3E mit dem parallelen Programm *LiSS-3D* der GMD durchgeführt, da die in CISPAR verwendeten CFD-Programme im Rahmen des Gigabit-Projekts nicht zur Verfügung stehen. Für die strukturmechanischen Berechnungen wird das parallele Programm *PERMAS* des CISPAR-Partners INTES benutzt. Beide Programme führen die gekoppelte Simulationsberechnung mit Hilfe der Kopplungsbibliothek COCOLIB durch. Diese ist seit dem Ende von CISPAR im März 1999 als Binärversion frei verfügbar, sie wurde inzwischen aber von dem Nachfolgeprodukt MpCCI (www.mpcci.org) abgelöst. Auf dem Metacomputer müssen sowohl die parallelen Programme als auch die COCOLIB auf der MetaMPI-Kommunikationsbibliothek basieren, einer Implementierung des MPI-Standards, die auf T3E und SP2 sowohl die interne als auch die externe Kommunikation realisiert („multi-system, multi-protocol“).

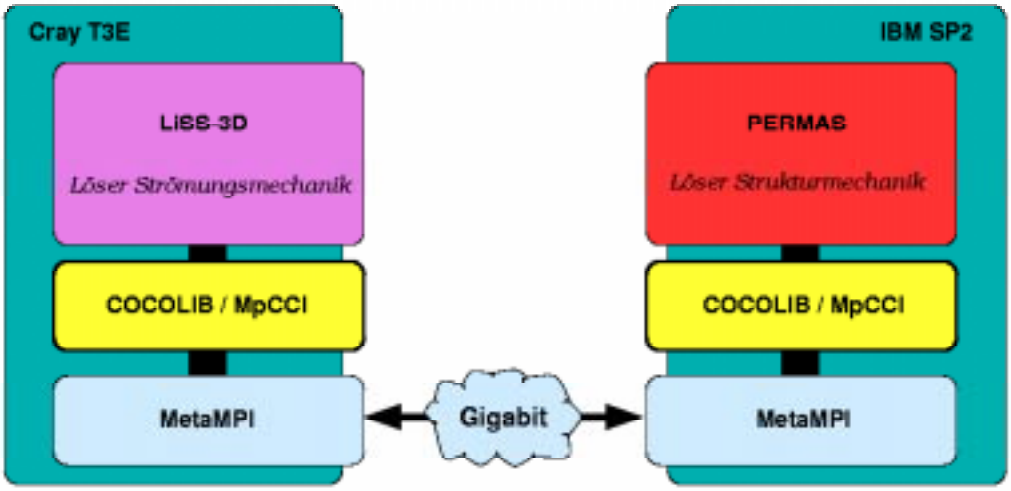


Abb.9.1: Gekoppelte Simulation auf dem T3E-SP2 Metacomputer

9.2 Methodisches Vorgehen

Die gekoppelte Anwendung benötigt die unabhängig voneinander entwickelten Softwarekomponenten MetaMPI, COCOLIB, *LiSS* und *PERMAS*. Deren Integration und damit die Realisierung der Kopplung wurde in drei Arbeitspaketen durchgeführt:

[1] Referenzimplementierung auf einem IBM-Workstationcluster

- Entwicklung von *PseudoFE*:
  Die GMD hat das Programm *PseudoFE* entwickelt, das den Kopplungsalgorithmus und den Datenaustausch von PERMAS via COCOLIB emuliert. Einerseits konnte damit unter Verwendung des *PseudoFE*-Sourcecodes die korrekte Anbindung des Programmpaketes *LiSS* an die Kopplungsbibliothek COCOLIB verifiziert werden. Andererseits wurde der unentgeltliche Aufwand des ehemaligen CISPAR-Partners INTES im Gigabit-Projekt minimiert, denn durch die Verwendung von *PseudoFE* wurde nur auf der SP2 eine einzige Neugenerierung von PERMAS auf der Basis von COCOLIB auf MetaMPI notwendig.

- Erweiterung von *LiSS* um die COCOLIB-Anbindung:
  Nachdem die grundsätzliche Eignung des Programmpaketes *LiSS* zur Berechnung der Strömung im Rahmen der ausgewählten Anwendung demonstriert werden konnte, wurden die COCOLIB-Aufrufe in *LiSS* integriert. Die in *LiSS* benutzte, MPI-basierte Kommunikationsbibliothek CLIC 4.5 wurde auf das in der COCOLIB implementierte Kommunikatorkonzept umgestellt und der eigentliche *LiSS*-Lösungskern mit den entsprechenden Unterprogrammaufrufen aus der COCOLIB versehen. Darüberhinaus mußten neue *LiSS*-interne Kommunikationen implementiert werden.

- Gekoppelte *LiSS-PseudoFE* Simulation auf einem IBM-Workstationcluster:
  Als Referenzimplementierung wurde auf einem IBM-Workstationcluster das Emulationsprogramm *PseudoFE* mittels COCOLIB auf MPICH an das parallele *LiSS* gekoppelt. Beispielhaft wurden damit Testfälle mit vier, acht und zehn seitens *LiSS* an der Rechnung beteiligten Prozessoren durchgeführt.

[2] Gekoppelte *LiSS-PseudoFE* Simulation auf dem Metacomputer T3E-SP2

- Test von MetaMPI auf dem Metacomputer T3E-SP2

- Portierung und Test der COCOLIB auf den Metacomputer T3E-SP2:
  Die COCOLIB wurde auf der Basis von MetaMPI sowohl auf der T3E als auch der SP2 installiert und auf dem Metacomputer getestet. Mit Hilfe der internen Testprogramme wurden Probleme in MetaMPI und COCOLIB bzgl. der unterschiedlichen Datenrepräsentationen der beteiligten Rechner aufgedeckt, die umgehend von den Entwicklern bei Pallas bzw. der GMD behoben wurden.

- Portierung von *LiSS* auf T3E:
  Das Programmpaket *LiSS* wurde auf der T3E installiert. Als Basis diente die auf dem IBM-Workstationcluster laufende Version, wobei das Host-Node-Modell in ein SPMD-Programmiermodell abgeändert wurde.

- Portierung von *PseudoFE* auf SP2

- Gekoppelte *LiSS-PseudoFE* Simulation auf dem Metacomputer T3E-SP2:
  Die Komponenten wurden integriert und gekoppelte *LiSS-PseudoFE* Simulationen auf dem Metacomputer T3E-SP2 durchgeführt.

[3] Gekoppelte *LiSS-PERMAS* Simulation auf dem Metacomputer T3E-SP2

- Portierung von *PERMAS* auf SP2:
  Aus dem CISPAR-Projekt existierte bereits eine auf MPICH basierende Portierung von *PERMAS* auf die SP2. Diese mußte durch einen INTES-Mitarbeiter für das Gigabit-Projekt auf der Basis von COCOLIB auf MetaMPI neu generiert werden.

- Gekoppelte *LiSS-PERMAS* Simulation auf dem Metacomputer T3E-SP2:
  Abschließend wurde auf der SP2 das Emulationsprogramm *PseudoFE* durch *PERMAS* ersetzt und das Testbeispiel erfolgreich mit der *LiSS-PERMAS* Kopplung auf dem Metacomputer gerechnet.

9.3 Gekoppelte Simulation für das Testbeispiel Bending Flap

Aus dem CISPAR-Projekt wurde das multidisziplinäre Testbeispiel *Bending Flap* des Partners Sulzer Innotec ausgewählt. Hierbei handelt es sich um die Simulation einer einseitig fixierten, verformbaren Klappe in einem Flüssigkeitskanal (Abb. 9.2). Da die Strömungsgeschwindigkeit im oberen Zuflußkanal höher ist als im unteren, wird die Klappe nach unten gebogen.

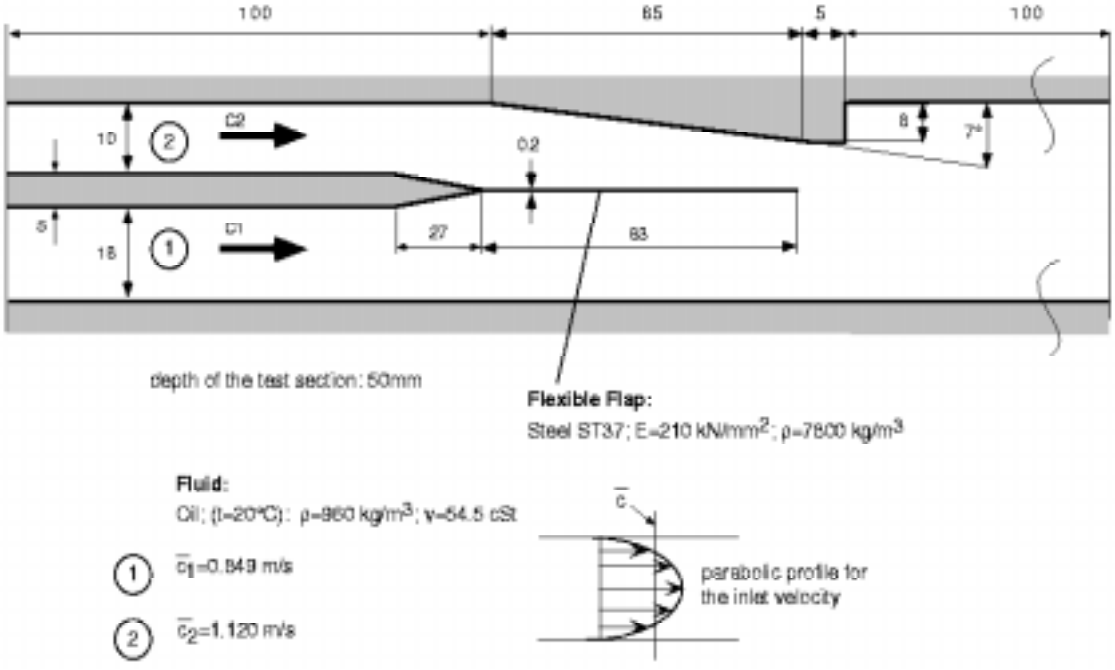


Abb. 9.2: Testbeispiel ***BendingFlap***

Abb. 9.3 zeigt das Ausgangsgitter für die gekoppelte Simulation, die aus technischen Gründen als 3D-Simulation (mit nur einer Elementschicht in der Tiefe) durchgeführt wird. Das Strömungsgitter für *LiSS* besteht aus 6615 Elementen, während das Gitter der Klappe für die Strukturanalyse mit *PERMAS* nur 26 Elemente umfaßt. Die „Non-matching“ Kontaktfläche besitzt auf der Strömungsseite 64 und auf der Strukturseite 26 Elemente, so dass die Kopplungsbibliothek COCOLIB die auszutauschenden Daten intern interpolieren muß.

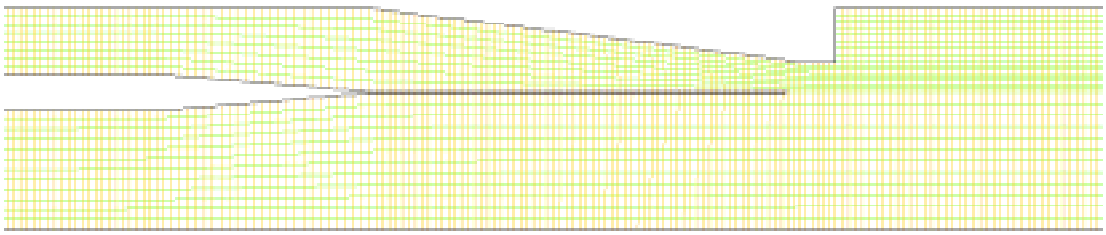


Abb. 9.3: Ausgangsgitter

Die Ergebnisse der gekoppelten Simulation sind in den folgenden Abbildungen dokumentiert. Abb. 9.4a zeigt die Druckverteilung vor dem ersten Kopplungsschritt: im oberen Kanal hat sich ein deutlicher Überdruck aufgebaut. Dieser führt in den ersten Kopplungsschritten zu einer zu weiten Durchbiegung der verformbaren Klappe (Abb. 9.4b). Im konvergierten Zustand in Abb. 9.5a hat sich ein Gleichgewicht zwischen dem Strömungsdruck und der inneren Spannung der Klappe (hier nicht dargestellt) eingestellt. Die zugehörigen Strömungsgeschwindigkeiten dieses 20. Kopplungsschritts sind in Abb. 9.5b dargestellt.

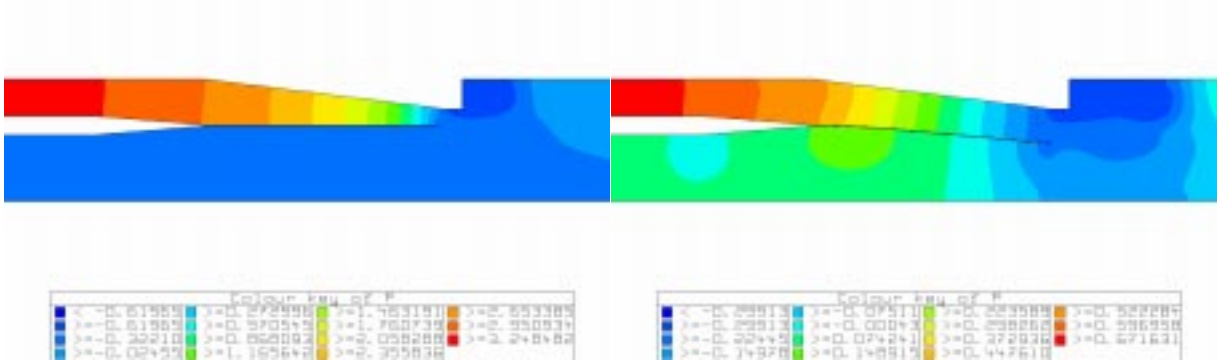


Abb. 9.4a/b: Druckverteilung vor dem 1. und 3. Kopplungsschritt

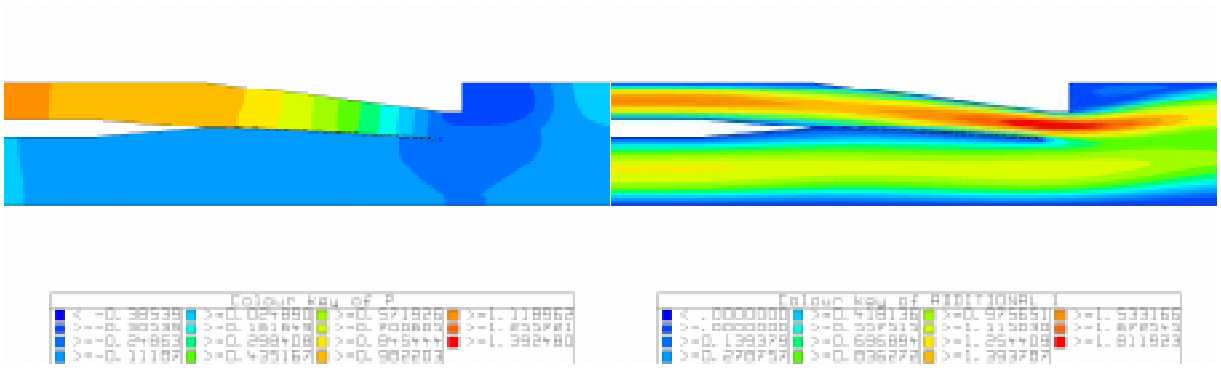


Abb. 9.5a/b: Konvergierte Druckverteilung und Strömungsgeschwindigkeit

Als Kopplungsalgorithmus wurde der Gauß-Seidel-Algorithmus verwendet. Hierbei rechnen das Strömungs- und Strukturprogramm immer abwechselnd (Abb. 9.6). Dies scheint langsamer zu sein, als wenn beide Programme gleichzeitig rechneten (Jacobi-Algorithmus). Jedoch ist die numerische Konvergenz des Gauß-Seidel-Algorithmus i.a. höher als beim Jacobi-Verfahren, was zu einer letztlich kürzeren Rechenzeit führt.

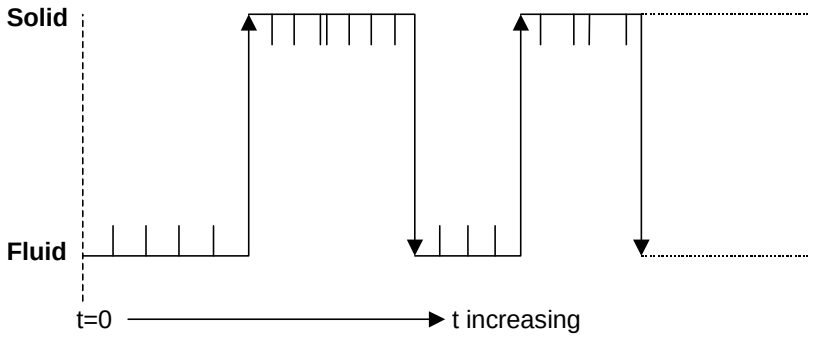


Abb. 9.6: Kopplungsalgorithmus Gauß-Seidel

Aus dem Gauß-Seidel-Algorithmus erklärt sich die Rechenzeitverteilung in Abb. 9.7. Die gekoppelte Simulation wurde hierbei mit einem *PERMAS*-Prozeß auf der SP2 und drei *LiSS*-Arbeitsprozessen auf der T3E durchgeführt (der zusätzliche, im wesentlichen nur steuernde *LiSS*-Masterprozeß ist nicht angezeigt). Auf dem Metacomputer kommt es zu einer echten Überlappung von Kommunikation und Rechnung, wobei die Zeit in der COCOLIB zum größten Teil aus Wartezeiten auf das Berechnungsende des jeweils anderen Programms besteht. Nur ein sehr kleiner Teil wird tatsächlich für den Datentransfer benötigt.

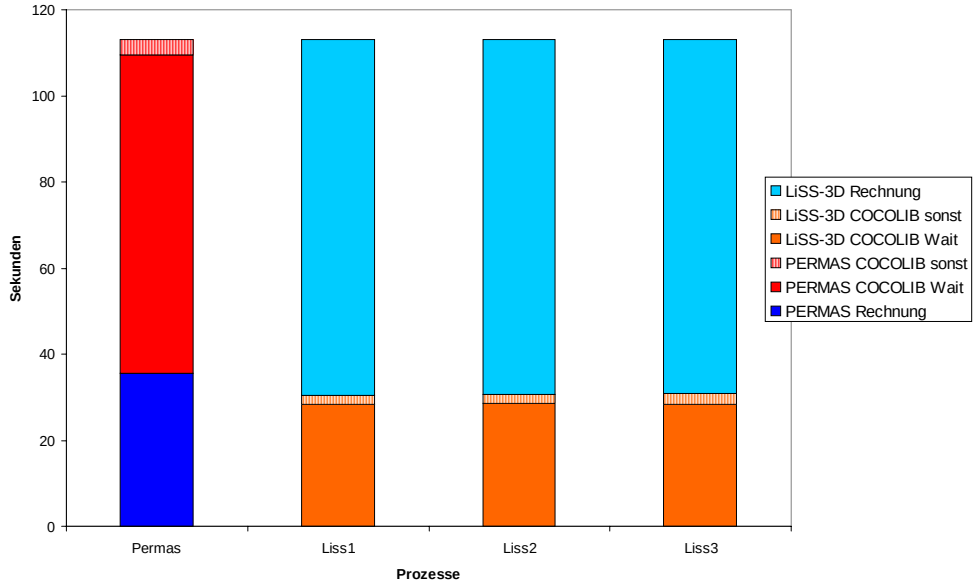


Abb. 9.7: Zeitverteilung auf die Prozesse

9.4 Zusammenfassung

In diesem Teilprojekt wurde erfolgreich demonstriert, dass eine gekoppelte Simulation über eine WAN-Verbindung wie das Gigabit sinnvoll machbar ist. Mit der Kopplungsbibliothek COCOLIB (bzw. deren Nachfolgerin MpCCI) auf MetaMPI stehen die entsprechenden Softwarekomponenten bereit, um eine derartige Anwendung auf einem Metacomputer zu betreiben. Insbesondere erlaubt der verwendete MPI-Standard, die gekoppelte Anwendung lokal vorzubereiten und erst dann auf den Metacomputer zu portieren.

Unser Dank gilt der Firma INTES für ihre Unterstützung bei der *PERMAS*-Portierung.

Publikationen und Vorträge

Referierte Zeitschriften und Konferenz-Proceedings

[1] Th. Eickermann, J. Henrichs, M. Resch, R. Stoy, R. Völpel:
Metacomputing in Gigabit Environments: Networks, Tools, and Applications,
Parallel Computing, 24 1998, Seite 1847-1872.

[2] Th. Eickermann, W. Frings, S. Posse, G. Goebbels, R. Völpel:
Distributed Applications in a German Gigabit WAN,
Proceedings of the Eigth IEEE Interational Symposiu on High Performance Distributed Computing, Redondo Beach, California, USA, August 1999, Seite 143-148.

[3] Th. Eickermann, H. Grund, J. Henrichs:
Performance Issues of Distributed Applications in a German Gigabit Testbed,
Recent Advances in Parallel Virtual Machine and Message Passing Interface,
eds. J. Dongarra, E.Luque, T. Margalev, Lecture Notes in Computer Science, Barcelona, Spanien, Springer Verlag, September 1999, Seite 3-10.

[4] Th. Eickermann, F. Hommes:
Metacomputing in the Gigabit Testbed West,
Workshop on Wide Area Networks and High Performance Computing,
eds. G. Coppermann, E. Jessen, G. Michler, Lecture Notes in Control and Information Sciences, Springer, Essen, September 1999, Seite 119-129.

[5] Th Eickermann, W. Frings, F. Hossfeld, S. Posse, G. Goebbels:
Supercomputer-enhanced functional MRI of the human brain,
IEEE Concurrency, January-March 2000, Seite 11-13.

[6] D. Gembris, J.G. Taylor, W. Frings, S. Wiese, S. Posse, D. Suter:
Sensitivity-enhancement for Functional MR Imaging in Real-Time (FIRE) by reference-vector optimization,
NeuroImage, 9(6) Part 2, 44, 1999.

[7] D. Gembris, J.G. Taylor, S. Schor, W. Frings, D. Suter, S. Posse:
Functional Magnetic Resonance Imaging in Real Time (FIRE): Sliding-window Correlation Analysis with Reference-vector Optimization,
Magnetic Resonance in Medicine, 43, 2000, Seite 259-268.

[8] G. Goebbels, N. Fournier, M. Goebel, H. Zilken, W. Frings, Th. Eickermann, S. Posse:
Remote Visualization of Radiological Data on a Responsive Workbench,
Proceedings of CARS '99, Computer Assisted Radiology and Surgery, 1999.

[9] F. Hommes:
Gigabit-Networking - Ein Erfahrungsbericht,
ONLINE '99, Congressband II, Verlag Online GmbH, Februar 1999, Seite C244.01-C244.09, ISBN 3-89077-193-9.

[10] F. Hommes, E. Pless:
Connecting Heterogeneous Supercomputers in Broadband Networks,
ICATM'99 Conference Proceedings, IEEE, 1999, Seite 324-329, ISBN: 0-7803-5428-1.

[11] R. Niederberger:
Super Computer Communications,
Proceedings of the 41. Cray User Group Meeting, Minneapolis, USA, Mai 1999.

[12] R. Niederberger, H. Grund, E. Pless, F. Hommes:
High speed supercomputer communications in broadband networks,
Terena - Nordunet Networking Conference 1999, June 1999, Computer Networks 31 (1999), Seite 2309-2318.

[13] S. Posse, T. Graf, W. Frings, K. Mathiak, S. Wiese, S. Goebels, H. Zilken, G. Goebbels, V. Kiselev, B. Elghahwagi, Th. Eickermann, D. Gembris:
Functional Imaging in Real Time (FIRE) on a Clinical Whole Body Scanner,
NeuroImage, 9(6) Part 2, 245, 1999.

[14] P. Wunderling:
First German Gigabit Network Testbed,
ERCIM News No. 31, October 1997.

[15] P. Wunderling, F. Hommes:
Gigabit Networking is Reality - 2.4 Gbps via ATM Wide Area Network,
ERCIM News, No. 37, April 1999.

Vorträge und Präsentationen

[16] Th. Eickermann, S. Frickenhaus, W. Frings, O. Heudecker, J. Quaas:
Giga–WAN im Westen, DFN-Stand auf der CeBIT‘99, März 1999.

[17] Th. Eickermann:
Metacomputing Projects in the Gigabit Testbed West,
ZKI-Arbeitskreis Supercomputing, Kiel, 29. April 1999.

[18] Th. Eickermann:
Verteiltes Höchstleistungsrechnen in einem Gigabit Testbed,
NIC-Seminar, Forschungszentrum Jülich, 4. Mai 1999.

[19] T. Eickermann:
Distributed Applications in a German Gigabit WAN
Eigth IEEE Interational Symposium on High Performance Distributed Computing,
Redondo Beach, California, USA, 5. August 1999.

[20] Th. Eickermann:
Metacomputing in High Bandwidth Networks,
Israeli German NGI Workshop, Neveh Ilan, Israel, 7. September 1999.

[21] Th. Eickermann:
Performance Issues of Distributed MPI-Applications in a German Gigabit Testbed,
EuroPVM/MPI’99, Barcelona, Spanien, 27. September 1999.

[22] Th. Eickermann, W. Frings, A. Klier, J. Quaas:
Gigabit Testbed West, DFN-Stand auf der CeBIT 2000, März 2000.

[23] Wolfgang Frings:
Analyse und 3D-Anzeige von Kernresonanzbildern in Echtzeit im Gigabit Testbed West,
Arbeitskreis Medizin, DFN-Symposium, Berlin, 8.März 1999.

[24] Ferdinand Hommes:
Einstieg in die Gigabit-Netze,
21. DECUS Symposium 1998, Karlsruhe, 20.-24. April 1998.

[25] Ferdinand Hommes:
Gigabit-Networking - Ein Erfahrungsbericht,
12. DFN-Arbeitstagung über Kommunikationsnetze, München, 2.-5. Juni 1998,
109. Tagung des GSE AK Telekommunikation, Bonn, 3.-4. September 1998.

[26] Ferdinand Hommes:
Gigabit-Networking - Ein Erfahrungsbericht,
ONLINE '99, Düsseldorf , 1.-4. Februar 1999.

[27] Ferdinand Hommes:
Gigabit-Networking - Ein Erfahrungsbericht,
6. User Conference des ATM Forums, Sankt Augustin, 4. März 1999.

[28] Ferdinand Hommes:
Gigabit Testbed West – Technik,
DFN-Symposium „Fortgeschrittene Kommunikationstechnik“, Berlin, 9.-10. März 1999.

[29] Ralph Niederberger:
(Super Computer (Communications)) ?= (Super (Computer Communications)),
Unicos-AK, DWD Offenbach, Offenbach, 29-30. Oktober 1998.

[30] Ralph Niederberger:
Gigabit Testbed West – Kapazitätsanforderungen, Leistungsmessungen,
DFN-Workshop Leistungsanalyse, Berlin, 19. Februar 1999.

[31] Ralph Niederberger:
Gigabit Testbed West – Technik, Erfahrungen, Ausblick,
DFN-Betriebstagung, Berlin, 23-24 Februar 1999.

[32] Ralph Niederberger:
Super Computer Communications,
41. Cray User Group Meeting, Minneapolis, USA, 24-28. Mai 1999.

[33] Ralph Niederberger:
High speed supercomputer communications in broadband networks,
Terena - Nordunet Networking Conference 1999, Lund, Schweden, 7-10. Juni 1999.

[34] Eva Pless:
Connecting Heterogeneous Supercomputers in Broadband Networks,
2nd International Conference on ATM, ICATM'99- Colmar,
Frankreich, 21-23. Juni 1999.

[35] S. Posse, T. Graf, W. Frings, K. Mathiak, S. Wiese, S. Goebels, H. Zilken, G. Goebbels, V. Kiselev, B. Elghahwagi, T. Eickermann, D. Gembris:
Functional Imaging in Real Time (FIRE) on a Clinical Whole Body Scanner,
Technical Demo auf Human Brain Mapping in Düsseldorf, Juni 1999.

[36] P. Wunderling:
Gigabit Testbed West – Technik,
Arbeitstagung des Arbeitskreises „Heterogene Netze“ der GSE,
Trier, 27. November 1997.

[37] P. Wunderling:
Gigabit Networking – Experiences at GMD,
Poznan Supercomputing and Networking Center, Poznan, Polen, 9. Dezember 1998.

Sonstige Veröffentlichungen

[38] Th. Eickermann, R. Niederberger, P. Wunderling, R. Völpel:
Aufbruch ins Jahr 2000,
DFN-Mitteilungen 45, 1997, Seite 13-15.

[39] Th. Eickermann, P. Wunderling:
Gigabit Schallmauer,
DFN-Mitteilungen 48, 1998, Seite 15.

[40] Th. Eickermann, W. Frings, P. Wunderling, R. Völpel:
Punktlandung im Jahr 2000,
DFN-Mitteilungen 52, 2000, Seite 9-12.

[41] D. Gembris, J.G. Taylor, W. Frings, S. Goebels, S. Schor, S. Posse, V.G. Kiselev, S. Wiese, N.J. Shah and D. Suter:
Sensitivity-Enhancement for fMRI by Reference-Vector Optimization,
Abstract book Intern. Soc. for Magnetic Resonance in Medicine (ISMRM), 1708, 1999.

[42] S. Frickenhaus, W.Hiller:
Coupling parallel atmosphere-and ocean-models in a metacomputing environment: Experience gained from the Gigabit Testbed West,
Berichte aus dem Fachbereich Physik, Report 99, AWI Bremerhaven, 4/2000.

[43] W. Frings, D. Gembris, G. Goebbels:
Beim Denken zusehen,
DFN-Mitteilungen 49, 1999, Seite 9-11.

[44] J. Quaas:
Sensitivitätsstudien zur Flußkopplung von Zirkulationsmodellen des Ozeans und der Atmosphäre,
Diplomarbeit am Institut für Geophysik und Metorologie der Universität zu Köln,
in Vorbereitung.