

# Diffusions-Tensor-Imaging als Gridanwendung – Perfomanzsteigerung und standortunabhängiger Zugang zu leistungsfähigen Ressourcen

Frank Hertel<sup>1</sup>, Dagmar Krefting<sup>2</sup>, Ralf Lützkendorf<sup>1</sup>, Fred Viezens<sup>1</sup>, Andreas Thiel<sup>3</sup>,  
Kathrin Peter<sup>4</sup>, Johannes Bernarding<sup>1</sup>

<sup>1</sup> Institut für Biometrie und Medizinische Informatik  
Otto-von-Guericke-Universität Magdeburg  
Leipziger Str. 44  
39120 Magdeburg  
frank.hertel@med.ovgu.de

<sup>2</sup> Charité - Universitätsmedizin Berlin, Institut für Medizinische Informatik

<sup>3</sup> Oldenburger Forschungs- und Entwicklungsinstitut für  
Informatik-Werkzeuge und -Systeme e. V.

<sup>4</sup> Konrad Zuse Institut Berlin

**Abstract:** Für die nicht-invasive Ermittlung des Verlaufes von Nervenfasern im menschlichen Gehirn, gewinnt die Analyse Diffusions-gewichteter Magnet-Resonanz-Bilder zunehmend an Bedeutung, da sie die Erforschung der inneren Strukturen des zentralen Nervensystems am lebenden Objekt ermöglicht. Aus den Analyseergebnissen lassen sich Aussagen zur Kopplung verschiedener funktionaler Bereiche im Gehirn ableiten oder Zusammenhänge zwischen Deformationen von Nervenfasern und bestimmten Krankheitsbildern erkennen bis hin zur Lokalisierung von Tumorgewebe. Die Modellierung der lokalen Diffusionsparameter kann eine Laufzeit von bis zu mehreren Tagen auf einem klassischen Arbeitsplatzrechner haben, da sie sehr rechenaufwändig ist. Die Implementation dieser Anwendung in einer Gridinfrastruktur inklusive Schicht-basierter Parallelisierung der statistischen Algorithmen mit webbasierten Portalzugang ermöglicht es, Diffusionstensenoren standortunabhängig auf leistungsfähigen Gridressourcen in mehreren Instanzen beschleunigt zu berechnen und Eingangs- wie Ergebnisdaten kollaborativ und sicher innerhalb der Gridinfrastruktur zu verwalten.

## 1 Einleitung

Der zunehmende Ausbau von Grid-Ressourcen innerhalb der D-Grid-Initiative im Bereich biomedizinischer Anwendungen eröffnet neue Wege und Strategien für das verteilte Rechnen [Gen06]. Die Nutzung dieser Strategien ist im Umfeld des *Diffusion Tensor Imaging* (DTI) von starker Bedeutung, da zum einen komplexe Auswertungen mit großen Datenmengen viel Rechenzeit in Anspruch nehmen und zum anderen die Nutzung lokaler Rechenressourcen stark abhängig von deren Verfügbarkeit ist.

Diffusions-gewichtete Magnet-Resonanz-Tomographie (MRT) ermöglicht einen bedeutenden Vorteil bei der Diagnose metabolischer Veränderungen im menschlichen Gehirn. Eine Erweiterung dieser Vorgehensweise stellt die Ermittlung der Diffusionstensoren von Nervenfasern im menschlichen Gehirn dar, die allgemein als Diffusions Tensor Imaging (DTI) bekannt ist. Da die Nervenfasern, welche die weiße Substanz im Gehirn darstellen, stark ausgerichtet und umschlossen von fettreichem Myelin sind, geschieht die Diffusion von Wasserprotonen entlang dieser Fasern schneller. Diese sogenannte Anisotropie der Diffusion kann untersucht werden, indem starke Gradienten räumlich unterschiedlich ausgerichtet werden und somit die Diffusionstensoren ermittelt werden können. Dieser Ansatz zur Untersuchung des Verlaufes von Nervenfasern ermöglicht nicht-invasive Studien der drei-dimensionalen Anatomie der weißen Substanz sowie die Rekonstruktion der axonalen Bahnen im zentralen Nervensystem, was im klinischen sowie wissenschaftlichen Umfeld von großer Bedeutung ist.

In dem vorliegenden Ansatz wurde die Open-Source-Anwendung *FMRIB Diffusion Toolbox* aus der FSL-Bibliothek verwendet [Oxf09]. Diese basiert auf sehr rechenaufwändigen Ansätzen der Wahrscheinlichkeitsrechnung, welche geeignet sind, sich kreuzenden Fasern innerhalb eines Voxels zu verarbeiten. Der komplette Prozess von den Rohdaten bis hin zur Traktografie unterteilt sich in drei Stufen: Zuerst werden die Rohdaten von Rauscharakterfaktoren befreit und die wichtigen Segmente im Gehirn ermittelt. Dieser Schritt wird Vorverarbeitung (preprocessing) genannt. Danach findet der eigentliche und rechenintensivste Schritt, die Modellierung der Diffusionsparameter, durch die Applikation *bedpostX* statt. *Bedpost* ist die Abkürzung für *Bayesian Estimation of Diffusion Parameters Obtained using Sampling Techniques*. Zuletzt wird die Fasertraktografie zur Abschätzung des Verlaufes der Nervenfasern innerhalb der weißen Substanz durchgeführt, welche die Nachverarbeitung (postprocessing) darstellt. Die Ausführungszeit des *bedpostX*-Algorithmus eines typischen diffusionsgewichteten MR-Scans benötigt etwa 24 Stunden. Hochauflösende Bilddaten, wie sie zum Beispiel in Forschungsprojekten verwendet werden, können mehrere Tage zur Berechnung beanspruchen.

Das workflowbasierte Absenden einer DTI-Berechnung innerhalb einer Grid-Infrastruktur löst die aufkommenden Schwierigkeiten dahingehend, dass mit Hilfe der Workflow-Management-Services die Daten in Schichten zerlegt und unabhängig voneinander berechnet werden können, um diese dann verteilt und parallel auf Gridressourcen auszuführen. Des Weiteren sorgt der *Grid Workflow Execution Service (GWES)* dafür, dass die Jobs nur an verfügbare Grid-Ressourcen weitergeleitet werden. Zusätzlich helfen Queuing-Policies dabei, Jobs mit geringer Berechnungszeit eine höhere Priorität zu verleihen, um schneller die Berechnungsergebnisse zurückzuerhalten. Die DTI-Anwendung *bedpostX* aus der FSL-Toolbox bietet gute Voraussetzungen für parallele Ausführung auf verteilten Gridressourcen.

## 2 Methoden

### 2.1 System-Architektur

Bei der Nutzung von MediGRID-Ressourcen im Rahmen der nationalen D-Grid-Initiative kommen verschiedene generische Middlewarekomponenten zum Einsatz [KBB<sup>+</sup>08]. Momentan verfügt MediGRID über 2000 CPU's und etwa 1 PByte Datenspeicher verteilt auf 6 Standorte und wird kontinuierlich ausgebaut. MediGRID basiert auf der Middlewarekomponente *Globus Toolkit 4 (GT4)* mit seinem serviceorientierten Layout, Sicherheitsmechanismen und generischer Job-Übermittlung, wie zum Beispiel die Web-Service-Komponente des *Grid Resource Allocation Management (WS-GRAM)* [Fos05]. Die darunter liegende Architektur von MediGRID wird in Abbildung 1 dargestellt. Die Erweiterungen, die auf die Kern-Infrastruktur des D-Grid aufbauen, bestehen aus dem *Storage Resource Broker (SRB)* und dem *Grid Workflow Execution Service (GWES)*.

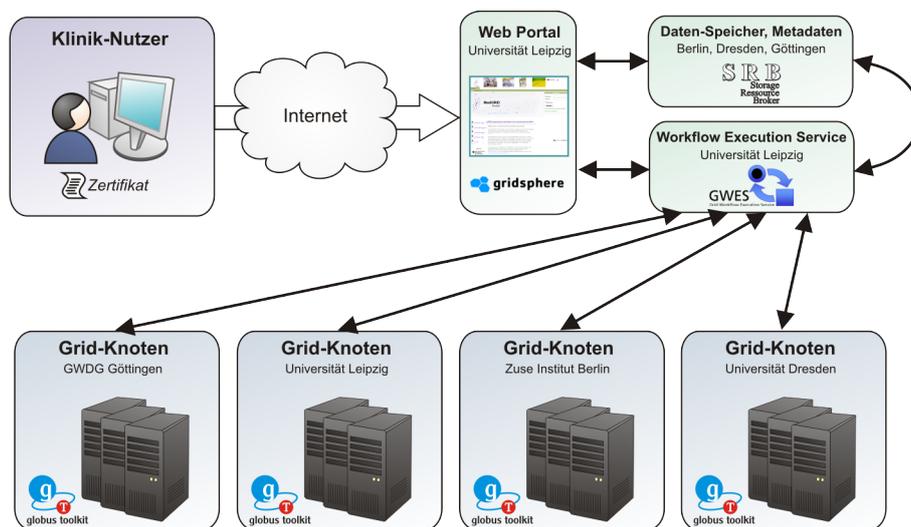


Abbildung 1: Die Architektur der MediGRID-Infrastruktur

In der Grid-Daten-Management-Komponente SRB werden die zu berechnenden Daten sowie die Berechnungsergebnisse zentral hinterlegt. Sie ermöglicht einen transparenten Zugriff auf eine hohe Anzahl an verteilten heterogenen Speicher-Ressourcen [RWM<sup>+</sup>03]. Die zu berechnenden Daten werden beim Start eines Berechnungs-Jobs vom SRB auf die Knoten im Grid geschoben, die die Berechnung durchführen. Jeder Nutzer hat ein eigenes Heimverzeichnis im SRB, welches einem Heimverzeichnis in einem lokalen Dateisystem ähnelt. Nutzer- und Gruppen-basierte Zugriffsrechte können für Dateien und Verzeichnisse definiert werden, um so Daten vor Fremdzugriff zu schützen oder für die kollaborative Nutzung freizugeben. Der Zugriff auf die Daten im SRB erfolgt durch ein Web-Portal.

Der Grid Workflow Execution Service (GWES) verwaltet den mit Hilfe der *Grid Workflow*

*Description Language* (GWDL) formulierten Workflow [Hoh07]. Der Workflow wird als Petrinetz in einer XML-Datei beschrieben. Abbildung 2 zeigt die Petri-Netz-Darstellung eines einfachen Workflow zur bedpostX-Berechnung eines einzelnen Verzeichnisses.

GWES arbeitet auf Basis von High-Level-Petri-Netzen (HLPN), die dazu genutzt werden können Transfer- und Speichervorgänge von Input- und Outputdaten ebenso wie Kontrolldaten, wie zum Beispiel der Bearbeitungsstatus eines Workflow-Schrittes, zu modellieren [HA06]. So definierte Workflows können mit Hilfe der HLPN-Formalismen algorithmisch auf Deadlocks geprüft werden.

Mit Hilfe der *Grid Resource Description Language* (GRDL) wird die auszuführende Anwendung und die für sie nutzbaren Ressourcen beschrieben. Das heißt, in einer XML-Datei werden die Gridressourcen aufgelistet, auf denen bedpostX installiert ist. Zu diesen Ressourcen kann dann der WS-GRAM die Berechnungs-Jobs weiterleiten.

## 2.2 Applikation

Wie eingangs erwähnt, besteht FDT aus drei Teilen, der Vorverarbeitung, der Berechnung der Diffusionstensenoren und der Faserverlauffindung (fiber tracking), wobei die Berechnung der Diffusionstensenoren der rechen-intensivste Teil der Abarbeitung ist. Er basiert auf den Markov-Chain-Monte-Carlo-Sampling-Algorithmus. FDT stellt eine grafische Benutzerschnittstelle bereit, kann aber ebenso Kommandozeilen-basiert ausgeführt werden. BedpostX erwartet eine feste Anzahl von Dateien, die in einem Verzeichnis zusammengefasst sind, einschließlich der original MRI-Bilder, die im NifTI-Format vorliegen müssen, sowie die Ergebnisse der Vorverarbeitung [CAB<sup>+</sup>04]. Die Angabe des Verzeichnisses ist ein notwendiger Parameter beim Aufruf von bedpostX, die drei weiteren Parameter sind optional zu verwenden. Die Ergebnisdaten werden in einem neuen Verzeichnis abgespeichert. Die vorliegende Implementation konzentriert sich auf die Anwendung von bedpostX mit Standardparametern, während die Vor- und Nachverarbeitung lokal auf einem Rechenknoten vorgenommen wird.

## 2.3 Lokale Clusterimplementation

Als Vergleichsbasis zur Gridimplementation soll die lokale Implementation der parallelisierten bedpostX-Anwendung auf einem Cluster dienen. Der Cluster besteht aus zehn 32-Bit-Desktop-PC's mit jeweils einem 3 GHz Intel Pentium 4 Prozessor und 512 MByte RAM [LMH<sup>+</sup>07]. Das Betriebssystem ist Windows XP, weshalb bedpostX unter *cygwin*, eine Wrapper-Bibliothek, die Windows die Unix-API zur Verfügung stellt, betrieben wird. Die Kommunikation wird via ssh-Protokoll realisiert.

### 3 Ergebnisse

#### 3.1 Evaluierungskriterien

Die Motivation für eine Gridimplementation besteht in der Bestrebung nach eine zuverlässigen und schnellen Berechnung von Diffusionstensenoren aus MR-Daten, sowie die Möglichkeit der Kollaboration auf gemeinsamen Daten und Prozesse. Die Evaluierung umfasst daher die Laufzeitmessung und die Bedienbarkeit. Die Applikation wird als monolithische und parallelisierte Version auf jeweils einem lokalen Cluster und im Grid betrieben. Der Zugang erfolgt über ein Webportal.

#### 3.2 Vorgehensweise

Auf den Grid-Knoten selbst wird zusammen mit den Programmkomponenten von bedpostX ein Bash-Skript hinterlegt, in welchem initiale Schritte, die zur Ausführung von bedpostX benötigt werden, aufgelistet sind. Der Workflow beschreibt nun in welcher Reihenfolge die Kommandos ausgeführt werden, um die einzelnen Phasen der Verarbeitung der Initialdaten zu durchlaufen. Die Phasen unterteilen sich wie folgt. Zunächst wird der Datensatz in die einzelnen Schichten zerlegt. Nachdem ermittelt wurde wie viel Rechenknoten der Anwendung zu Verfügung stehen, die Anzahl sei hier  $n$ , werden die einzelnen Schichten zu  $n$  Datenpaketen zusammengefasst. Diese werden anschließend auf die  $n$  Rechenknoten verteilt und der Berechnungsprozess, also die eigentliche bedpostX-Anwendung,  $n$ -mal gestartet. Die Entscheidung wieviel und welche Rechenknoten der Anwendung zur Verfügung stehen wird dem GWES überlassen. Nachdem alle Berechnungsergebnisse vorliegen, werden die Ergebnispakete wieder in die einzelnen Schichten aufgeteilt und zu einem Gesamtergebnispaket zusammengeführt, welches dann wieder zurück in den SRB übertragen wird.

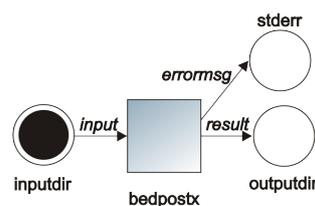


Abbildung 2: Ein einfacher Workflow der monolithischen Ausführung von bedpostX als Petri-Netz

GWES führt, wie oben erwähnt, das Brokering und Scheduling der Ressourcen auf Basis der Informationen der GRDL-Datei aus. Dabei bietet GWES eine durchgängige Überwachung der Kontrollpunkte (Stellen im Petrinetz). Sollte ein Anwendungsschritt (Transition im Petrinetz), fehlschlagen so wird der Fehler in einen Bericht geschrieben und der Anwendungsschritt auf eine anderen freie Ressource umgeleitet und neugestartet. Abbildung 3 zeigt den oben beschriebenen komplexen Workflow der parallelen Verarbeitung

eines Datenpaketes mit mehreren Schichten durch bedpostX.

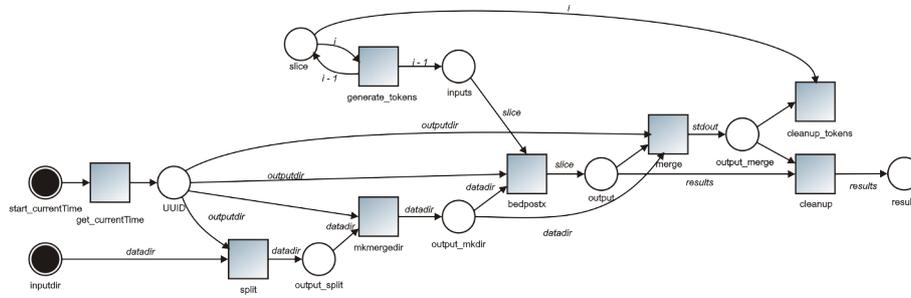


Abbildung 3: Ein komplexer Workflow für die parallelisierte Abarbeitung als Petri-Netz

Gestartet und überwacht wird der Workflow im MediGRID-Portal. Dazu bietet das Portal neben der Komponente, durch welche sich die zu berechnenden Daten in den SRB hochladen lassen, ebenfalls ein Portlet mit dem die den Workflow beschreibende XML-Datei in den SRB übertragen werden kann, um anschließend diesen Workflow im Grid abzusetzen. Die Parallelisierung der Diffusionstensor-Berechnung wird durch das Aufteilen der Ausgangsdaten in mehrere Schichten erreicht. Diese Schichten werden dann unabhängig von einander gleichzeitig auf entsprechend vielen Prozessoren im Grid berechnet und die Ergebnisse zentral zusammengeführt. Schließlich lassen sich die Ergebnisdaten wieder durch das SRB-Portlet abrufen.

### 3.3 Performanz

Klassische Datensätze in der wissenschaftlichen Anwendung bestehen aus 10 bis 70 Schichten. Beispielweise benötigt die Berechnung der Diffusionstensoren eines Datensatz bestehend aus 56 Schichten 10 Tage auf einem einfachen Arbeitsplatzrechner. Die Berechnung des selben Datensatzes auf dem oben beschriebenen lokalen Cluster bestehend aus 10 Arbeitsplatzrechnern war nach etwas weniger als 24 Stunden abgeschlossen. Bei der parallelen Verarbeitung skalierte die Anzahl verwendeter Rechner nahezu linear mit der Verkleinerung der Berechnungszeit bis zu einer Anzahl von 10 Rechnern. Darüber hinaus war kein signifikanter Performanzgewinn zu verzeichnen, da die nichtparallelisierbaren Vorgänge wie Vor- und Nachverarbeitung, Zusammenführung der Daten und Kommunikation über das Netzwerk statisch sind und bei der Performanzsteigerung den Flaschenhals darstellen. Bei der Verwendung aller zehn Rechner im Cluster konnte die bedpostX-Berechnung um das zehnfache beschleunigt werden. Anzumerken ist hierbei, dass die schwache Hardwareausstattung die langen Berechnungszeiten verursacht, dies allerdings nicht untypisch ist für eine Forschungsgruppe in einer aus öffentlichen Mitteln finanzierten Universität. Tabelle 1 zeigt die Gegenüberstellung der Berechnungszeit des aus 56 Schichten bestehenden Datensatzes auf dem Cluster mit 10 Knoten und der Gridimplementierung. Zusätzlich wurden in der Gridimplementierung verschiedene andere Datensätze mit unterschiedlicher Anzahl an Schichten berechnet. Die Varianz bei den Berechnungszeiten resultiert aus der

Heterogenität der Gridressourcen, der Anzahl an Gridressourcen, die vom GWES für den Berechnungsjob allokiert werden konnten, und der unterschiedlich lange Aufenthalt eines Berechnungsjobs in der Warteschlange der allokierten Gridressource. Insgesamt konnte bei Datensätzen mit vielen Schichten durch die monolithische Gridimplementation eine Performanzsteigerung bis zu einem Faktor von 11 erreicht werden, was sich hauptsächlich in der besseren Hardwareausstattung der Gridressourcen begründet. Durch die Parallelisierung der Gridimplementation konnte die Berechnung zusätzlich um einen Faktor bis zu 5,7 beschleunigt werden.

Tabelle 1: Laufzeit Messung

System	Anzahl Schichten	Monolithisch in $h$ insgesamt (pro Schicht)	Parallelisiert in $h$ insgesamt (pro Schicht)	Beschleunigungsfaktor
Grid	11	$6.5 \pm 0.7$ (0.6)	$3.5 \pm 2.5$ (0.3)	1.9
	14	$3.0 \pm 0.8$ (0.2)	$3.5 \pm 1.7$ (0.25)	0.9
	20	$8.9 \pm 3.8$ (0.4)	$3.6 \pm 1.2$ (0.18)	1.6
	56	$22.0 \pm 8.0$ (0.4)	$3.8 \pm 2.2$ (0.06)	5.7
Cluster	56	240.0 (4.2)	23 (0.4)	10.4

## 4 Zusammenfassung

Die Gridimplementation der DTI-Anwendung FSL bedpostX birgt in vielerlei Hinsicht Vorteile in sich. Zunächst konnte die Berechnungszeit verkürzt werden, indem voneinander unabhängige Schichten eines Datensatzes parallel verarbeitet werden. Die Steigerung der Performanz der DTI-Berechnung wird begrenzt durch statische Vorgänge, wie das Übertragen der Daten und ihre Vorverarbeitung, die nicht parallelisierbar sind. Die Nutzung von Gridressourcen entlastet lokale Ressourcen wie Arbeitsplatzrechner und Workstations. Der Nutzer muss sich hierbei keine Gedanken machen, welche Ressourcen die Berechnungen durchführen. Er setzt den Berechnungs-Job über das Portal ab und die Gridmiddleware übernimmt die Suche und Reservierung freier Gridressourcen. Die zu berechnenden Daten werden im SRB verwaltet und automatisch an die allokierte Ressource verteilt. Der durch ein Portal realisierte webbasierte Zugang zum Grid, ermöglicht es dem Nutzer, standortunabhängig und kollaborativ Eingangs- wie Ergebnisdaten zu verwalten. Das heisst ein Nutzer muss weder über eigene Speicherressourcen noch über eigene Rechenressourcen verfügen, um aufwändige DTI-Berechnungen durchführen zu können.

## Literatur

- [CAB<sup>+</sup>04] R W Cox, John Ashburner, Hester Breman, Kate Fissell, Christian Haselgrove, Colin J Holmes, Jack L Lancaster, David E Rex, Stephen M Smith, Jeffrey B Woodward und Stephen C Strother. A (Sort of) New Image Data Format Standard: NIfTI-1. In *Poster on the OHBM Meeting*, 2004.

- [Fos05] I Foster. Globus toolkit version 4: Software for service-oriented systems. In *IFIP International Conference on Network and Parallel Computing*, Seiten 2–13, 2005.
- [Gen06] W Gentsch. D-Grid, an E-Science Framework for German Scientists. In *Proceedings of The Fifth International Symposium on Parallel and Distributed Computing (ISPDC'06)*, Seiten 2–13, 2006.
- [GO08] T Glatard und S D Olabariaga. From gridified scripts to workflows: the FSL Feat case. In *Proceedings MICCAI*, Seiten 33–42, 2008.
- [HA06] A Hoheisel und M Alt. *Petri nets*, Kapitel Workflows for e-Science - Scientific Workflows for Grids, Seiten 110–128. Springer, 2006.
- [Hoh07] A Hoheisel. Grid workflow execution service - dynamic and interactive execution and visualization of distributed workflows. In Academic Computer Centre CYFRONET AGH, Hrsg., *Proceedings of the Cracow Grid Workshop 2006*, Jgg. 2, Seiten 13–24, 2007.
- [KBB<sup>+</sup>08] D Krefting, J Bart, K Beronov, O Dzhimova, J Falkner, M Hartung, A Hoheisel, T A Knoch, T Lingner, Y Mohammed, K Peter, E Rahm, U Sax, D Sommerfeld, T Steinke, T Tolxdorff und M Vossberg, F Viezens und A Weisbecker. MediGRID: Towards A User Friendly Secured Grid Infrastructure. *Future Generation Computer Systems*, Seiten 326–336, 2008.
- [KVBT08] D Krefting, M Vossberg, K Beronov und T Tolxdorff. Standortunabhängiger Zugang und interaktive Steuerung von gridbasierten Diensten zur medizinischen Bildverarbeitung über ein Gridportal. In *Proceedings GMDS*, Seiten 335–337, 2008.
- [KVT08] D Krefting, M Vossberg und T Tolxdorff. Simplified Grid Implementation of Medical Image Processing Algorithms using a Workflow Management System. In *Proceedings MICCAI*, Seiten 23–32, 2008.
- [LMH<sup>+</sup>07] R Lützkendorf, T Mönch, M Hollmann, S Baecke, S Mulla-Osman und J Bernarding. Diffusion Tensor Imaging (DTI) at 3T and 7T. In *Proceedings DS ISMRM*, Seiten 1–13, 2007.
- [NRW04] J Novotny, M Russell und O Wehrens. Gridsphere: An advanced portal framework. In *Proceedings of the 30th Euromicro Conference*, Seiten 412–419, 2004.
- [Oxf09] Analysis Group FMRIB Oxford. FMRIB Software Library. <http://fsl.fmrib.ox.ac.uk/fsl>, Datum des Abrufs: 10.04.2009.
- [RWM<sup>+</sup>03] A Rajasekar, M Wan, R Moore, W Schroeder, G Kremenek, A Jagatheesan, C Cowart, B Zhu, S-Y Chen und R Olschanowsky. Storage Resource Broker - managing distributed data in a grid. *Computer Society of India Journal*, 33(4):42–54, 2003.
- [Smi04] S M Smith. Advances in functional and structural MR image analysis and implementation as FSL. *NeuroImage*, 23:208–219, 2004.