Ein interdisziplinäres Grid-Anwenderpraktikum basierend auf Instant-Grid

Thomas Rings

Institut für Informatik, Georg-August-Universität Göttingen

Fred Viezens

Institut für Biometrie und Medizinische Informatik, Otto-von-Guericke-Universität Magdeburg

Jörg Meyer

II. Physikalisches Institut, Georg-August-Universität Göttingen

Andreas Aschenbrenner

Staats- und Universitätsbibliothek, Georg-August-Universität Göttingen

1. Einleitung

Die Idee eines interdisziplinären Anwenderpraktikums für Grid-Computing [2] entstand durch eine Überschneidung von Interessen in den Disziplinen Physik, Medizin und Geisteswissenschaften. Hierbei spielte die Informatik

die Verbindungsrolle, da die drei Disziplinen ein wichtiges Forschungsgebiet der Informatik, das Grid-Computing, benutzen, um deren jeweilige Forschung voranzutreiben.

Das Praktikum fand im Sommersemester 2008 an der Universität Göttingen statt. 16 teilnehmende Studierende aus den o. g. Disziplinen wurden in vier Gruppen unterteilt. Jede Disziplin stellte Tutoren zur Betreuung der Studierenden. Zudem wurden jeder Gruppe vier Arbeitsrechner aus dem Rechnernetz zum Aufbau eines lokalen Grids zu Verfügung gestellt. Mit diesem Grid sollten die Aufgaben der verschiedenen D-Grid Projekte praktisch gelöst werden.

Die Funktionsweise von Grid-Technologien wurde den Studierenden im Praktikum durch deren Anwendung erklärt. Mit Hilfe der lokal gestarteten Grid-Software Instant-Grid [8, 14] konnte eine eigenständige Grid-Experimentierumgebung zur Bearbeitung der Aufgaben aus den Disziplinen aufgebaut und benutzt werden.

Das nachfolgende Kapitel ist folgendermaßen gegliedert: In Abschnitt 2 wird die technische Grundlage, die Konfiguration des Instant-Grid, für das Praktikum beschrieben. In Abschnitt 3 werden die Aufgaben der verschiedenen Disziplinen vorgestellt. Abschließend in Abschnitt 4 werden kurz die Ergebnisse diskutiert und ein Ausblick gegeben.

2. Technischer Aufbau

Die praktische Realisierung der Aufgaben des Praktikums benötigt eine Grid-Software. Als Grid-Software wurde Instant-Grid [8,14] ausgewählt, da es vereinfachte Möglichkeiten zum Starten von lokalen Computer-Grids [2] bietet. Instant-Grid basiert auf Linux-Knoppix [9] und beinhaltet die Grid-Middleware *Globus Toolkit 4* (GT4) [3]. Somit kann mit Instant-Grid ohne aufwändige Installation ein Computer-Grid von CD gestartet werden. Trotzdem benötigt Instant-Grid, wie im folgenden Abschnitt beschrieben, Anpassungen um in einem lokalen Rechnernetz zu starten.

2.1 Konfiguration in einem lokalen Rechnernetz

Instant-Grid benötigt ein abgeschlossenes lokales Netzwerk, um Konflikte mit Diensten in existierenden Rechnernetzwerken zu vermeiden. Deswegen müssen Rechner, welche das Instant-Grid starten, von ihrem Rechnernetz getrennt werden. Dazu werden die Rechner eines Instant-Grid auf Netzwerk-Switch-Ebene in einem *Virtual Local Area Network* (VLAN) zusammengefasst und somit vom Rechnernetz getrennt. Wie in Abbildung 1 beispielhaft dargestellt, gibt es zwei Projektgruppen, VLAN 1 und VLAN 2, welche

jeweils unabhängig voneinander ein Instant-Grid mit eigenem Server und drei Clients starten können. Um ein Routing zwischen den Netzen und ggf. Firewallregeln auf dem Rechnernetzserver zu ermöglichen, verlässt das eigentliche Rechnernetz den Switch direkt (untagged) und die Instant-Grid-Netze mit ihrer jeweiligen VLAN-Kennung (tagged).

Die Benutzung des Instant-Grid erfolgt nur zu vorgegebenen Zeiten, da in dem Switch die jeweiligen VLANs durch einen Tutor manuell eingerichtet oder entfernt werden müssen. Dies ist nötig, damit die Rechner zusätzlich ihr Standardbetriebssystem starten können. In unserer Konfiguration wurden jeder Gruppe vier Rechner zugeteilt. Auf diesen konnten die Studierenden ihr eigenes Grid aufbauen, starten und konfigurieren.

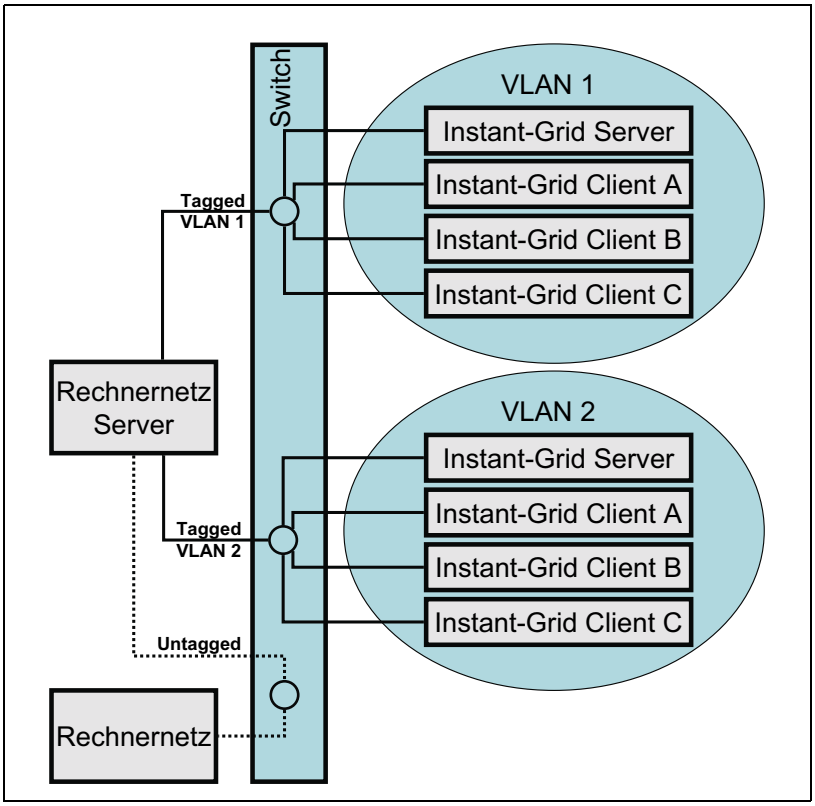

Abb 1: VLAN-Konfiguration

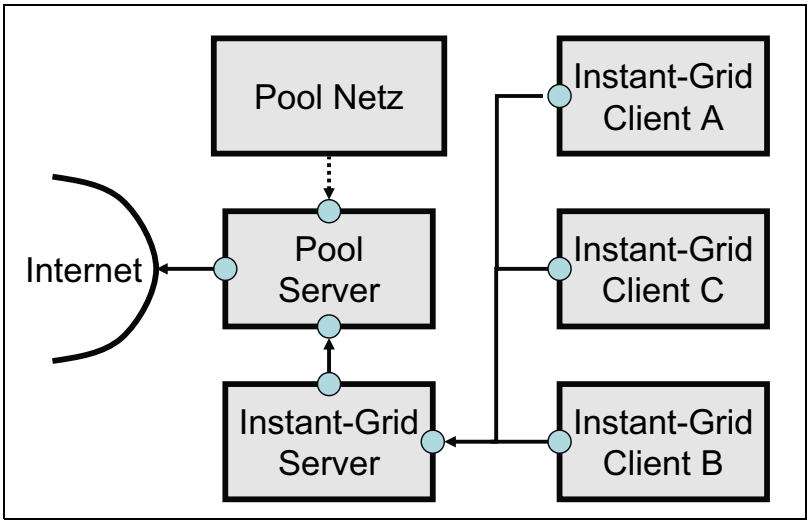

Abb. 2: NAT-Konfiguration

Damit von Instant-Grid-Rechnern auf externes Netz (Zugang zum Internet) zugegriffen werden kann, sollten auf dem Rechnernetzserver sowie auf dem Instant-Grid-Server *Network Address Translation* (NAT)-Gateways eingerichtet werden, wie in Abbildung 2 dargestellt ist.

2.2 Technische Herausforderungen des Instant-Grids

Durch die Benutzung der Linux-Knoppix-Live-CD fehlt es Instant-Grid an einer Komponente zum Management von Benutzer-Accounts. Konfigurationsdaten können nicht permanent geschrieben werden. Somit können Benutzer nicht permanent erstellt werden. Deswegen können die Studierenden nur vor Ort arbeiten und werden zusätzlich von einem Tutor betreut, da die Studierenden als Administrator agieren können und das Rechnernetz schädigen können.

Die Instant-Grid-Netzwerkumgebung erfordert ein lokales Netzwerk. Deshalb ist, wie in Abschnitt 2.1 beschrieben, die Konfiguration des Instant-Grid in einem Rechnernetz sehr umständlich. Studierende sollten die Konfiguration des VLANs nicht selbst vornehmen können. Bei unserer Konfiguration wurde ein Script von einem Tutor gestartet, welches das VLAN aktiviert bzw. deaktiviert. Wird Instant-Grid ohne die VLAN-Aktivierung gestartet, kann es zu erheblichen Konflikten mit dem Rechnernetz kommen.

3. Aufgaben aus den Disziplinen

Das Ziel des Praktikums war es, den Studierenden unabhängig von deren Studienfach Konzepte des Grid-Computings durch Anwendung von Grid-Technologie zu vermitteln. Der Praxisbezug konnte durch die drei in Göttingen ansässigen Projekte des D-Grids [7] der Disziplinen Physik, Medizin und Geisteswissenschaften hergestellt werden. Somit befassen sich die Aufgaben mit HEP Grid [5] aus der Physik, MediGRID [11] aus der Medizin und TextGrid [13] aus den Geisteswissenschaften.

Jede Disziplin erstellte einen Aufgabenblock, welcher sich auf das jeweilige D-Grid Projekt bezog und von jeder Gruppe rotierend bearbeitet wurde. Die drei Aufgabenblöcke wurden folgendermaßen unterteilt:

1. Einführung / Grundlagen
2. Übung / Praxis
3. Reflexion / weiterführende Fragen

Teil 1 gab eine Einführung in die jeweilige Disziplin und behandelte die Voraussetzungen und den Stand des D-Grid Projektes. Die Studierenden recherchierten die Besonderheiten des jeweiligen Themas und arbeiteten Beziehungen der Disziplin zur Grid-Technologie heraus. Zudem sollten Optimierungsvorschläge zu den Grid-Projekten bezüglich Grid-Technologie gegeben werden.

Im zweiten Teil sollten diese Vorschläge und weitere Aufgabenstellungen in einem Grid praktisch umgesetzt werden. Hierzu zählte unter Anderem der Versuchsaufbau des Instant-Grid und die technische Umsetzung der Fragestellungen.

Die Vor- und Nachteile dieser Lösung sowie weitere Optimierungsvorschläge wurden im dritten Teil diskutiert und wiederum umgesetzt. Hierbei wurde die entwickelte Grid-Software vorgestellt und ausgeführt sowie Herausforderungen bei der Entwicklung erörtert und diskutiert. Im Folgenden werden die Aufgaben bezüglich der D-Grid Projekte näher erläutert.

3.1 MediGRID [11]

Innerhalb des MediGRID Aufgabenblocks erfolgte eine Einweisung in verschiedene biomedizinische Anwendungen. Hierzu konnten die Studierenden Anwendungen zur Bildverarbeitung, Bioinformatik und klinischen Forschung über das MediGRID-Portal [12] als Gast benutzen. Um sensible Bereiche dieser Anwendungen zu benutzen, mussten die Studierenden die Beantragungsprozedur von D-Grid-Zertifikaten durchlaufen. Diese Zertifi-

kate ermöglichten durch Eintrag in der Virtual Organisation Education-VO, sensible Grid-Ressourcen und Grid-Anwendungen des MediGRID zu benutzen.

Zudem wurden Programme gridifiziert und in das Instant-Grid eingebunden. Die Verteilung im Grid erfolgte durch Kommandozeilenaufrufe. Diese Aufrufe wurden in ein Script übertragen und ausgeführt. Der Ablauf der Kommandos in diesen Script entsprach einem Grid-Arbeitsablauf (Workflow). Dieses Script wurde exemplarisch in der Applikation für Genvorhersagen angewendet. Die Ausführungen der von den Studierenden entwickelten Workflows im Instant-Grid spiegelten die realen Abläufe auf den Ressourcen des D-Grid wieder. Die Ergebnisse durch Anpassungen und Lösungen aus den thematischen Praktikumsaufgaben fanden Berücksichtigung in den laufenden MediGRID-Entwicklungen.

3.2 HEP Grid [5,6]

Der Praktikumsteil der *Hochenergie-Physik* (HEP) hatte als Ziel, einen Einblick in die typischen Anforderungen moderner Teilchenphysik-Experimente an das Computing zu geben und die verschiedenen Herausforderungen zu untersuchen. Als Beispiel wurde das DØ Experiment am Proton-Antiproton-Beschleuniger Tevatron am Fermilab [1] in den USA betrachtet. Am Tevatron finden Kollisionen von Protonen- und Antiprotonen-Paketen bei den derzeit höchstmöglichen Energien mit einer Rate von etwa 2,5 MHz statt. Die Signale der bei einer Kollision gestreuten und neu produzierten Teilchen wurden nach einer Ereignis-Vorselektion mit einer Rate von 50 Hz vom DØ Detektor aufgezeichnet. Die Rohdaten-Menge pro Ereignis beträgt etwa 250 KB. Das Experiment läuft seit 2001, so dass bereits eine enorme Menge an Daten gemessen und gespeichert wurde. Aus den Rohdaten wurden Physik-Objekte rekonstruiert und ebenfalls gespeichert, um diese in verschiedenen Analysen weiter zu verarbeiten. Neben den gemessenen Daten werden für die Physik-Analysen Simulationen von Kollisionen, so genannte Monte-Carlo-Ereignisse, benötigt. Dazu wurde etwa die gleiche Menge an simulierten und gemessenen Daten gebraucht.

Im Praktikum wurde die Simulation von DØ Ereignissen schrittweise durchgeführt. Der Workflow der Simulationsschritte sowie das Workflow-Skriptsystem wurden im Einzelnen betrachtet. Aus der Dauer der Simulation einzelner Ereignisse und aus den Dateigrößen der verschiedenen Ausgabedaten wurde extrapoliert, wie viel CPU- und Speicherbedarf das Experiment in etwa hat. Im Jahr 2005 hatte das DØ Experiment beispielsweise 250 TB aus gemessenen Rohdaten neu rekonstruiert. Diese gewaltige Anforderung an Computer-Ressourcen lässt sich nur mit weltweit verteilten Grid-Clustern

bewältigen. In einem weiteren Praktikumsteil wurde durch die Studierenden die interaktive Generation von Monte-Carlo-Ereignissen auf ein solches Grid-System übertragen. Hierzu wurde wie in anderen Versuchsteilen Instant-Grid verwendet. Die Studierenden schickten über Kommandozeilen-Befehle des Globus-Toolkits Simulationsereignisse ins Instant-Grid. Die Verwendung von Kommandozeilen und Skripten ist typisch für die Anwendung von Grid-Computing in der Hochenergie-Physik.

Im Praktikum wurden Aspekte wie Daten-Archivierung, Datensicherheit, Aufbau eines Grid-Systems, Anforderungen an die verwendete Middleware sowie mögliche künftige Weiterentwicklungen diskutiert sowie praktisch betrachtet und umgesetzt. Die Computing-Anforderungen der weiteren Experimente am *Large Hadron Collider* (LHC) [10] am CERN in Genf erhöhen den Bedarf an Grid-Computing in der Hochenergie-Physik. Da die Hochenergie-Physik Grid-Computing zur häufigen und ständigen Erstellung von Simulationen und Analysen verwendet, ist eine stabile, standardisierte und getestete Umgebung wichtiger als die Verwendung neuester Entwicklungen in der Grid-Technolgie.

3.3 TextGrid [13]

Der Aufgabenblock zu TextGrid vermittelte den Studierenden die Chancen einer Grid-Infrastruktur für die Geisteswissenschaften. Die Diskussionen mit den Studierenden bauten dabei auf den konkreten Arbeiten des Projektes TextGrid auf, gingen aber weit darüber hinaus.

Der erste Teil vermittelte ein grundlegendes Verständnis für digitale Kulturgüter (z. B. Digitalisate) und darauf basierende geisteswissenschaftliche Forschung. Anforderungen in Bezug auf den wissenschaftlichen Workflow (speziell wie er in TextGrid unterstützt wird), sowie auch in der Verwaltung von digitalen Objekten (z. B. Metadatenverwaltung, Langzeitarchivierung) wurden daraus abgeleitet.

Mechanismen zur Datenreplikation und verteilten Speicherung von Digitalisaten in einer Grid-Umgebung sind wichtige Grundlagentechnologien für die Geisteswissenschaften. Der zweite Teil demonstrierte den Studierenden daher verteilte Datenvorhaltung sowie die Ausführung von Grid-Jobs wie z. B. zur verteilten Konvertierung von Digitalisaten von 100 Megabyte TIFF-Bildern in 1,4 Megabyte große JPEG-Bilder.

In der abschließenden Diskussion im dritten Teil arbeiteten die Studierenden die Chancen einer verteilten digitalen Infrastruktur für die Geisteswissenschaften heraus. Umgekehrt zeigten sie auch einige mögliche Beiträge geisteswissenschaftlicher Technologien in eine disziplinübergreifende Grid-

Infrastruktur auf – von interaktiven Tools bis zur kollaborativen Forschungsumgebung.

Die Studierenden lernten in diesem Aufgabenblock die durchaus kniffligen Anforderungen der Geisteswissenschaften und dahingehend schon entwickelte Technologien und Konzepte kennen. Interaktion und Vergleich mit den Physikern, Medizinern und Informatikern in GoeGrid [4] haben den Studierenden in besonderen Maße die unterschiedlichen Anwendungsgebiete näher gebracht und die zukünftigen e-Humanities greifbar werden lassen.

4. Ergebnisse und Ausblick

Das Praktikum wurde von den Studierenden gut evaluiert und war somit ein Erfolg. Aus technischer Sicht hat sich das Instant-Grid als ein ideales Werkzeug für die Durchführung des Praktikums erwiesen. Instant-Grid ist ein Instrument für ad-hoc-Grid-Anwendungen, welches erweiterbar und gut skalierbar ist. Es sind auch Anpassungen im Sinne kurzfristiger Entwicklungen wie Portlets bis hin zum Aufbau vollständiger Entwicklungsumgebungen möglich. Dem Charakter des Praktikums Rechnung tragend, sind Lösungswege eruiert worden, um Problemstellungen der Disziplinen zu lösen.

Auch zur Vermittlung von tiefer gehenden Grid-Programmierkenntnissen, speziell für die Studierenden der Informatik, ist Instant-Grid das Werkzeug der Wahl. Der Bedarf nach Grid-Entwicklern aus den Reihen der Informatik sowie der anderen Fachdisziplinen ist klar erkennbar. Die Vermittlung von Grid-Programmiertechniken ist in einem interdisziplinären Anwenderpraktikum nicht möglich. Deshalb ist ein Programmierpraktikum für Grid-Technologien in Vorbereitung.

In der abschließenden Diskussion und Nachbereitung dieses interdisziplindren Praktikums wurde sich mit den Interaktionen und der Zusammenarbeit zwischen den Studierenden der verschiedenen Fachrichtungen näher auseinander gesetzt. Es ist nicht Ziel dieses interdisziplindren Praktikums, dass Geisteswissenschaftler programmieren oder Informatiker die chemischen Prozesse der Genvorhersage lernen, sondern dass sie gemeinsam und sich gegenseitig ergänzend die Aufgaben lösen. Idealerweise würde z. B. der Physiker die richtigen Fragen stellen, der Geisteswissenschaftler die Methodik zur Beantwortung der Frage einbringen, der Mediziner das Hintergrundwissen und der Informatiker zur Umsetzung beitragen. In jedem Aufgabenblock wurde die jeweils zur Disziplin passende Rolle durch die Studierenden wahrgenommen und dieser somit einander ergänzend gelöst.

In diesem Sinne gehen die Fertigkeiten, die dieses interdisziplinäre Praktikum vermittelt, weit über das „Grid“ hinaus. Es geht um Kooperation mit

Kollegen, die ganz anderes Wissen einbringen und vielleicht ganz anders an Fragestellungen herangehen. Ebenso geht es um Kommunikation, in der Anforderungen und Vorschläge nicht alleine mit Fachbegriffen und Konzepten gespickt sind, die nur Kollegen der eigenen Disziplin verstehen können. Vor allem aber geht es darum, die unterschiedlichen Disziplinen auch in der Universität stärker zusammen zubringen und die Studierenden auf eine spätere Berufswelt vorzubereiten, in der die Disziplinen meist verschmelzen. Gerade die Vermittlung solcher Eigenschaften ist daher für die Studierenden eine Erfahrung für das Leben, wie sie nur eine interdisziplinäre Lehrveranstaltung vermitteln kann.

Literatur

[1] Fermilab. [Online; http://www.fnal.gov überprüft am 14.10.2008].

[2] Ian Foster. What is the Grid? A Three Point Checklist. Grid Today, 1(6):22, 2002.

[3] Ian Foster. A globus primer, An Early and Incomplete Draft. Technical report, Globus Alliance, 2005.

[4] Göttinger Grid-Ressourcen-Zentrums (GoeGrid). [Online; http://www.d-grid.de/index.php?id=438 überprüft am 10.10.2008].

[5] HEP-Grid. [Online; http://www.d-grid.de/index.php?id=44 überprüft am 29.09.2008].

[6] Worldwide LHC Computing Grid. [Online; http://lcg.web.chern.ch/LCG überprüft am 29.09.2008].

[7] D-Grid Initiative. [Online; http://www.d-grid.de/ überprüft am 29.09.2008].

[8] Instant-Grid. [Online; http://instant-grid.de überprüft am 29.09.2008].

[9] KNOPPER.NET. Knoppix. [Online; http://www.knoppix.org überprüft am 29.09.2008].

[10] The Large Hadron Collider (LHC). [Online; http://lhc.web.cern.ch/lhc/ überprüft am 14.10.2008].

[11] MediGRID. [Online; http://www.d-grid.de/index.php?id=42 überprüft am 29.09.2008].

[12] MediGRID-Portal. [Online; https://portal.medigrid.de überprüft am 09.10.2008].

[13] TextGrid. [Online; http://www.textgrid.de/, http://www.d-grid.de/index.php?id=167 überprüft am 29.09.2008].

[14] Alexander Willner. Entwurf und Implementierung einer Ressourcen-Datenbank für das Instant-Grid-Projekt der GWDG. Master's thesis, Georg-August-Universität Göttingen, 2006.