



### Arbeitspaket 2: Blaupausen und Beratung

# Kosten-Nutzen-Abschätzung des Aufbaus von neuen Communities<sup>1</sup>

Deliverable	2.1.4 Kosten-Nutzen-Abschätzung Community-Aufbau
Autoren	Arbeitspaket 2: Blaupausen und Beratung
Editoren	Joachim Schultes, Stephan Sandvoss
Datum	23-05-2011
Dokument Version	2.0.0

#### **A: Status des Dokuments**

Deliverable 2.1.4, Version 2.0.0, Release.

#### **B: Bezug zum Projektplan**

Diese Kosten-Nutzen-Abschätzung stellt eine Entscheidungshilfe zur Verfügung, ob bei gewissen Anforderungen und Rahmenbedingungen Grid-Computing verwendet und ein neues Community-Grid aufgebaut werden soll.

---

<sup>1</sup>This work is created by the WissGrid project. The project is funded by the German Federal Ministry of Education and Research (BMBF).

**C: Abstrakt**

Die Kosten-Nutzen-Abschätzung entwickelt Entscheidungshilfen, ob sich eine Arbeitsgruppe mit gewissen Computing-Anforderungen einer bestehenden Community anschließen soll, ob eine neue Community aufgebaut werden soll oder ob eine andere Lösung wie z.B. Cloud-Computing verwendet werden soll.

**D: Änderungen**

Version	Date	Name	Brief summary
0.0.0	28.1.2010	S. Sandvoss	Erstellung des Arbeitsdokumentes
0.0.1	28.1.2010	S. Sandvoss	erste Struktur
0.0.2	21.2.2010	S. Sandvoss	Ausformulierung aller Abschnitte außer Zusammenfassung und Literaturverzeichnis
0.0.3	25.2.2010	S. Sandvoss	Restrukturierung, um Inhalt und Organisation mehr zu betonen
0.1.0	09.03.2010	S. Sandvoss	Restrukturierung, Teilung in allgemeines Grid-Computing und Community-Aufbau
0.1.1	31.03.2010	S. Sandvoss	kleine Korrekturen, erste Abschätzungen für Kostenbeträge hinzugefügt
0.1.2	06.04.2010	S. Sandvoss	Anhang mit Kostenbeträgen fertig gestellt
0.1.3	12.04.2010	S. Sandvoss	Neue Rechtschreibung
1.0.0	04.05.2010	H. Enke	Release
1.0.1	09.03.2011	J. Schultes	Einarbeitung interner Kommentare
1.1.0	29.03.2011	J. Schultes	neues Kapitel 3.5: Datenmodelle, weitere Überarbeitung der Abschnitte
1.1.1	16.04.2011	H. Enke	Typos etc.
2.0.0	23.05.2011	H. Enke	Release

# Inhaltsverzeichnis

- 1 Abstract** **4**
  
- 2 Einleitung** **5**
  
- 3 Schritt 1: Grid-Computing** **7**
  - 3.1 Überblick . . . . . 7
  - 3.2 Gemeinsame Themengebiete . . . . . 8
  - 3.3 Vorgaben des Managements . . . . . 9
  - 3.4 Arbeitsgruppen und Anwender . . . . . 11
  - 3.5 Anforderungen aus dem Datenmodell . . . . . 12
  - 3.6 Softwareorientierte Anforderungen . . . . . 13
  - 3.7 Hardwareorientierte Bedingungen . . . . . 14
  - 3.8 Fazit . . . . . 16
  
- 4 Schritt 2: Community-Aufbau** **18**
  - 4.1 Überblick . . . . . 18
  - 4.2 Gemeinsame Themengebiete . . . . . 18
  - 4.3 Vorgaben des Managements . . . . . 19
  - 4.4 Arbeitsgruppen und Anwender . . . . . 20
  - 4.5 Softwareorientierte Anforderungen . . . . . 20
  - 4.6 Hardwareorientierte Bedingungen . . . . . 21
  - 4.7 Fazit . . . . . 22
  
- 5 Zusammenfassung** **23**

# Kapitel 1

## Abstract

The following chapters provide considerations for a cost-benefit analysis about the usage of the Grid Computing and the establishment of a new Community Grid. The target audience for this paper are all people of an (academic) institution considering Grid Computing, both decision makers and users and technical staff. Financial, juridical, organisational, political, thematic and technical aspects are considered and discussed.

# Kapitel 2

## Einleitung

Akademische Nutzergemeinschaften mit ähnlichen Problemstellungen betreiben üblicherweise einen regen Erfahrungsaustausch. Manche arbeiten wohlorganisiert zusammen, z.B. im Rahmen einer Community. Institutionen, die noch nicht Mitglied einer Community sind, stehen nun nach der Erfindung der Grid-Technologie vor der Entscheidung, weiterhin auf sich selbst gestellt zu arbeiten, einem Community-Grid beizutreten oder mit Gleichgesinnten ein neues Community-Grid zu gründen. Dieses Dokument betrachtet zur Kosten-Nutzen-Abschätzung zuerst die Rahmenbedingungen beim Einsatz des Grid-Computings (Kapitel 3), dann beim Aufbau einer neuen Community (Kapitel 4). Wenn also schon die generelle Entscheidung für den Einsatz des Grid-Computing gefallen ist, kann der Leser direkt mit Kapitel 4 beginnen.

Beim Grid-Computing stellen sich global verteilte Arbeitsgruppen ihre lokalen Ressourcen via Netzwerk gegenseitig zur Verfügung, um gemeinsame Problemstellungen zu lösen oder durch den Austausch unterschiedlicher Daten zu neuen Erkenntnissen zu gelangen. Ausgehend von unterschiedlichen Inhalten und Anforderungen wurden verschiedene Community-Grids gegründet, z.B. AstroGrid-D [AG10], C3 Grid [C3G10a], HEP Grid [HG10], InGrid [IG10], MediGRID [MG10a], Text-Grid [TG10a].

Diese Communities unterscheiden sich im wesentlichen durch folgende Elemente: Themengebiete, Leitung, Arbeitsgruppen, Struktur, Software-Repository und Hardware. Außer Computinghardware mit hoher Rechenleistung und großem Speicherplatz stellt ein Community-Grid also auch eine gewisse Infrastruktur in den Bereichen Software und Personal bereit. All diese Ressourcen inklusive sämtlicher Daten werden üblicherweise gemeinsam, aber kontrolliert genutzt. Ebenso werden oftmals auch die (Zwischen-)Ergebnisse zentral bereitgestellt. Daher bietet sich Grid-Computing bei räumlich verteilten Kollaborationen, vor allem auch bei internationalen Großkollaborationen an. Wie oben angedeutet können jedoch auch viele voneinander unabhängige Arbeitsgruppen aufgrund thematischer Nähe ein Community-Grid gründen, um damit insgesamt schneller und effizienter zu arbeiten.

Vom Grid-Computing unterscheiden sich z.B. Desktopgrid- und Cloud-Computing. Desktopgrid-Computing besteht normalerweise aus gleichberechtigten Rechnern, die sowohl Dienste in Anspruch nehmen, als auch diese zur Verfügung stellen können. Die Bereitstellung und Inanspruchnahme erfolgen in der Regel spontan und ohne Absprache zwischen den Nutzern. Cloud-Computing entspricht dagegen eher dem Client-Server-Modell und verfolgt eine zum Teil sogar flexible und dynamische Auslagerung von Rechenlasten und Datenmengen je nach Nutzungsgrad auf gemeinsam genutzte

beziehungsweise fremde (kommerzielle) Ressourcen. Bei beiden Ansätzen werden eigene Ressourcen in der Regel fremden Nutzern nicht zur Verfügung gestellt.

In den folgenden Kapiteln werden mögliche Anforderungen zum Einsatz des Grid-Computings und zur Gründung eines Community-Grids sowohl aus Sicht der einzelnen Institutionen als auch aus Sicht der Community vorgestellt. Danach folgt eine Diskussion aller in Frage kommenden Aspekte mit dem Ziel, eine Kosten-Nutzen Abschätzung vornehmen zu können, die aus einer Entscheidung für das Grid-Computing oder ein neues Community-Grid resultieren. Gegebenenfalls wird eine Empfehlung für eine andere Lösung gegeben, z.B. sich einem bestehenden Community-Grid anzuschließen, Cloud-Computing oder eine lokale Lösung zu verwenden.

Dedizierte Vergleichsstudien zwischen Cloud- und Grid-Computing sind in [EGEE08, ARXIV08] zu finden. Üblicherweise ist jedoch — wie wir später sehen werden — bei wissenschaftlichen oder nicht kommerziellen Projekten der Aufbau eines neuen Community-Grids die erste Wahl, sofern ein hoher Bedarf an Ressourcen besteht und noch kein inhaltlich ähnliches Community-Grid besteht.

# Kapitel 3

## Schritt 1: Grid-Computing

### 3.1 Überblick

Um zu entscheiden, ob sich für eine Nutzergemeinschaft die Verwendung des Grid-Computings lohnt, muss folgende Frage beantwortet werden: Soll mit anderen, thematisch nahe stehenden Institutionen wohlorganisiert zusammengearbeitet werden? Diese Frage betrifft nicht nur inhaltliche, sondern auch politische, finanzielle und organisatorische Aspekte, vor allem hinsichtlich der gemeinsamen Nutzung von individuellen Ressourcen und Daten. Um die Frage zu beantworten, muss eine Kosten-Nutzen-Analyse durchgeführt werden, die all jene Gesichtspunkte bewertet und gegeneinander abwägt. Ein ökonomisches Modell für eine solche Abschätzung ist in Abbildung 3.1 zu sehen. Da es aber Gesichtspunkte gibt, deren Kosten nicht bestimmbar sind, kann der Wert einer Zusammenarbeit über Grid-Computing mittels der Kosten-Nutzen-Analyse nicht mit einer einzigen Zahl (einem Geldbetrag) beziffert werden. Es muss viel mehr auf die individuelle Situation bezogen ein differenziertes Bild gezeichnet werden, das zunächst auf K.O.-Kriterien hin untersucht wird und dessen Gesamteindruck letztendlich den Ausschlag gibt.

Bevor mit der Analyse begonnen wird, sollte jedoch zuerst überprüft werden, wie weit der Wille vorhanden ist, mit anderen Institutionen intensiv zusammenzuarbeiten. In welchem Maß Verantwortungen, Hardware, Software, Personal, Daten oder Berechnungen geteilt oder ausgetauscht werden, hängt nämlich von der Bereitschaft aller Arbeitsgruppenmitglieder ab — vor allem auch von der Leitung — dies zu tun. Ist eine generelle Bereitschaft zur Zusammenarbeit vorhanden, kann die Suche nach Gleichgesinnten beginnen.

Danach muß von der Nutzergemeinschaft (und den Gleichgesinnten) ein genauer Anforderungskatalog erstellt werden, der alle für die Abschätzung relevanten Quantitäten und Qualitäten spezifiziert. Diejenige Lösung, die am besten dem Anforderungskatalog gerecht wird, birgt damit auch den größten Nutzen. Unter Berücksichtigung der Kosten jeder möglichen Lösung kann entschieden werden, ob die den größten Nutzen bringende Lösung realisiert wird oder doch eine andere Option aufgrund von Kostenoptimierung gewählt wird. Nach Abstimmung mit den Gleichgesinnten kann entschieden werden, ob die Zusammenarbeit als Beitritt oder Neugründung organisiert wird, siehe hierzu Kapitel 4. Schließlich sind Blaupausen zur Realisierung der beschlossenen Zusammenarbeit in [WG10b] zu finden.

In den folgenden Abschnitten wird auf die fünf Aspekte Themengebiete, Managementvorgaben, Anwenderbedürfnisse, Software- und Hardwareanforderungen näher eingegangen, ihre nach sich



Abbildung 3.1: Modell einer Kosten-Nutzen-Analyse [P04, modifiziert]

ziehenden Kosten und Nutzen für ausgewählte Lösungen abschätzen und vergleichen, sowie anschließend ein Fazit ziehen.

## 3.2 Gemeinsame Themengebiete

Eine inhaltliche Übereinstimmung oder Nähe zu anderen Arbeitsgruppen ist meistens der Hauptgrund, Grid-Computing zu verwenden. Gerade um Daten, Berechnungen oder Forschungsergebnisse auszutauschen ist diese Form der Zusammenarbeit erfunden worden. Diese dürfte in der Zukunft mehr und mehr lokale Lösungen unter einem Dach vereinen, wodurch zahlreiche Synergieeffekte zu erwarten sind. Dahingegen scheinen das Peer2Peer- und Cloud-Computing eher für Projekte geeignet zu sein, die nicht wissenschaftlich sind oder keines Austauschs bedürfen.

Durch die wohlorganisierte Zusammenarbeit über das Grid-Computing steigt in der Regel die Gesamteffizienz, d.h. der Aufwand zur Erzielung von Forschungsergebnissen sinkt. Dies kann aber nur gelingen, wenn ein gegenseitiges, effizientes Geben und Nehmen zwischen den Mitgliedern gepflegt wird, so dass der Nutzen der Zusammenarbeit den zusätzlich entstehenden Organisationsaufwand überwiegt. Dies bedeutet auch, dass standardisierte Architekturen von Hard- und Software existieren und die einzelnen Institutionen über leistungsfähige Schnittstellen zusammenarbeiten. Außerdem gibt es Großprojekte, die sich gar nicht anders als mit Grid-Computing bewältigen lassen.

### 3.3 Vorgaben des Managements

Es gibt Bereiche im allgemeinen Management, die meistens unter der Verantwortung der Arbeitsgruppenleitung liegen. Hierzu gehören:

- das Themengebiet
- die Finanzen
- die Einhaltung von Gesetzen
- die Realisierung der Computing-Lösung
- das Personalwesen
- die Sicherheit
- der Außenauftritt
- die Hierarchiestruktur
- die Politik

Diese Aspekte haben den größten Einfluß auf die Entscheidung für eine bestimmte Lösungsoption (Desktopgrid-, Cloud-, Grid- oder lokales Computing).

Die zu bearbeitenden Themenbereiche werden von der Arbeitsgruppenleitung festgelegt, und es besteht eine starke Korrelation zwischen diesen und den anderen Bereichen. Je aufwendiger z.B. ein Forschungsvorhaben ist, desto mehr Geld und Personal wird benötigt. Je erfolgreicher die Projekte sind, desto größer ist in der Regel das zur Verfügung stehende Budget.

Umgekehrt legt die Menge der zur Verfügung stehenden Geldmittel im Prinzip die Größe des Projekts und den Umfang der bewältigbaren Berechnungen fest — und damit auch die Quantität und Qualität der Ressourcen. Oft wird daher der Aufbau einer lokalen Lösung ausgeschlossen, die beträchtliche Investitions- und Unterhaltskosten verursacht, gerade wenn das Projekt keine langfristige Laufzeit hat. In diesem Fall ist es günstiger, für die Nutzung externer Ressourcen aufzukommen, z.B. im Rahmen eines Cloud-Computing. Falls das Projekt jedoch nur kurz ist, oder es noch nicht absehbar ist, ob es (mehrfach) verlängert wird, bieten sich Desktopgrid- und Grid-Computing an, die beständig und günstig erweiterbar sind. Um aber unvorhergesehene Kosten zu vermeiden, sollten die Modalitäten für die Nutzungsabrechnungen vorher genau geklärt werden. Ein Beispiel für einen Kostenvergleich zwischen Cloud- und lokalem Computing ist in [LCG08], zwischen Cloud- und Grid-Computing in [EGEE08] zu finden.

Da die Jurisdiktion von Staat zu Staat verschieden ist, kann z.B. eine Ablage von Daten in einem anderen Staat aufgrund dessen oder der eigenen Gesetze verboten sein. Außerdem können Übertragungs- oder Datenzugriffssicherheit in anderen Staaten z.B. durch fehlende Gesetze hinter üblichen Standards zurückbleiben, so dass eine Datenablage dort nicht zu Gebote steht. Während lokale Lösungen hiervon nicht betroffen sind, muss die Realisierung der übrigen Optionen gegebenenfalls auf ausgewählte Staaten beschränkt werden, was vor allem beim Desktopgrid- und auch Grid-Computing einen erheblichen Aufwand bedeuten kann.

Bevor die Entscheidung für eine Option fällt, müssen Aspekte hinsichtlich der Realisierung bedacht werden. Der Zeitraum bis zum Beginn der produktiven Nutzung wird in der Regel durch Planungszeiten, rechtliche Fristen (z.Bsp. Ausschreibungsfristen), Liefer- und Installationszeiten beträchtlich sein. Gerade der Neuaufbau einer lokalen Lösung dürfte generell am zeitaufwendigsten sein. Aber selbst das Outsourcing ins Cloud-Computing kann sich durch Finanzierungsfragen, Verhandlungen mit dem Anbieter sowie evtl. notwendiger Infrastrukturinstallation (z.Bsp. Netzwerkanbindung) in die Länge ziehen. Alle drei Ansätze können jedoch, wenn eigene Ressourcen schon ausreichend vorhanden sind in der Regel relativ zeitnah nach der Entscheidung realisiert werden, sofern die Anpassung der genutzten Software dies ermöglicht.

Ebenfalls können die notwendigen Investitionen in die Akzeptanz der Nutzer der favorisierten Lösung eine beträchtliche Rolle spielen, z.B. bei der Favorisierung einer Lösung oder aufgrund einer bestehenden Expertise in diesem Bereich. Außerdem benötigt jede Option eine unterschiedliche Menge von (Unterstützungs-) Personal, das auch verschieden qualifiziert ist. Gerade hinsichtlich des Angebots auf dem Arbeitsmarkt, aber auch der Arbeits(tarif)verträge sowie der Einstellungs- und Entlassungsmöglichkeiten können große Differenzen bestehen. Lokale Lösungen sind hiervon weit mehr betroffen als externe wie das Cloud-Computing, während Desktopgrid- und Grid-Computing dazwischen liegen.

Es gibt mehrere Sicherheitsaspekte, die beim Clustercomputing berücksichtigt werden müssen, vornehmlich die physikalische Datensicherheit, die Übertragungssicherheit, die Datenzugriffssicherheit und die Unfallsicherheit. Hardwaredefekte und Katastrophen wie Brand, Überschwemmung, Erdbeben, etc. können zu einem Datenverlust führen. Gerade bei lokalen Lösungen ist diese Gefahr deutlich erhöht, während es extrem unwahrscheinlich ist, dass weltweit verteilte Daten gleichzeitig durch solche Vorfälle verloren gehen. Die Übertragungssicherheit impliziert, dass Daten sowohl physikalisch korrekt (ohne Bitfehler) als auch ohne Zugriff Dritter, sei es lesend oder schreibend, von einem Ort zum anderen übertragen werden. Um dies zu gewährleisten, werden fehlertolerante Hard- und Software sowie Datenverschlüsselung benötigt. Da lokale Lösungen keine Datenübertragung über externe Verbindungen benötigen, sind sie im Falle von Hochsicherheitsanforderungen die einfachste Option. Die Zugriffssicherheit schließt an die Übertragungssicherheit an und betrifft (nicht) vorsätzliche Datenmodifikation wie Hinzufügen, Löschen und Ändern von Informationen. Das Auftreten versehentlicher Modifikationen kann durch Schulungen vermindert und durch Datensicherungen rückgängig gemacht werden. Weitaus seltener sind mutwillige Zugriffe durch Angriffe. Externe Angreifer sind relativ leicht in Schranken zu halten, indem z.B. Zertifikate mit starker Verschlüsselung und sichere Passwörter verwendet werden, die regelmäßig ausgetauscht werden. Interne Angreifer sind dagegen weitaus gefährlicher. Nur mit klaren Sicherheitsrichtlinien und Strafandrohungen bei Verstößen, hierarchischen Befugnissen und einer restriktiven Informationspolitik sowie regelmäßigen Schulungen für die Mitarbeiter können solche Risiken minimiert werden. Dies ist umso wichtiger, je sensibler die Daten, je verstreuter deren Ablage und je größer die Anzahl der Nutzer mit Zugriff sind. Eine lokale Lösung bietet hier wieder die höchste Sicherheit, während Cloud-Computing gerade aufgrund seines Konzepts, Ressourcen an Drittanbieter auszulagern, am unsichersten ist, vgl. hierzu auch [ARXIV08]. Die Unfallsicherheit schließlich betrifft die technischen Anlagen, die aufgrund von Software- oder Hardwarefehlern zu Sach- oder Personenschäden führen können. Dieses Thema betrifft jedoch kaum das Clustercomputing, sondern eher die Systeme, die die Computing-Infrastruktur bereitstellen und verwalten.

Eine Entscheidung für das Grid-Computing bedeutet aufgrund der Zusammenarbeit mit anderen Institutionen, dass zusätzlich zum eigenen Außenauftritt einer für die Community erstellt und gepflegt werden muss. Der Auftritt in der Öffentlichkeit ist nicht zu unterschätzen, und entsprechend dessen

Qualität können sich positive oder negative Konsequenzen ergeben. Somit besteht hier ein großer Unterschied zum Desktopgrid- und Cloud-Computing, wo dies unüblich und auch nicht notwendig ist.

Die Hierarchiestruktur beim Grid-Computing ist in der Regel eher partnerschaftlich, d.h. es besteht eine gegenseitige, intensive Kooperation zwischen Gleichgesinnten. Im Idealfall kommt diese Lösung ohne allzu viele Regeln für diese Zusammenarbeit und den Umgang miteinander aus. Praktisch werden aber auch hier Interessenkonflikte zu einer Reglementierung und einer Art Abrechnung führen. Dies unterscheidet sich sowohl vom Desktopgrid-Computing, wo es im allgemeinen nur interne Regeln gibt, und dem Cloud-Computing, bei dem genau festgelegte Leistungen von einem externen Ressourcenanbieter erbracht werden (sofern diese benutzt werden).

Schließlich gibt es noch politische Aspekte, aufgrund derer häufig die Entscheidung für eine gewisse Option getroffen wird. Diese betreffen alle Lösungsoptionen unterschiedlich, da diese Faktoren in der Regel lokal abzugrenzen sind. Beispiele für politische Faktoren sind die Schaffung oder Erhaltung von Arbeitsplätzen, die Aufwertung einer Region durch den Standortfaktor IT, Zusammenarbeit von Wissenschaft/Forschung und Industrie, der Föderalismus in Deutschland, etc.

Auch wenn eine Kosten-Nutzen-Abschätzung hinsichtlich der behandelten Aspekte meist nur für die Arbeitsgruppenleitung möglich ist, sollten Belange der übrigen Share- und Stakeholder aufgrund der dargestellten Implikationen nicht unberücksichtigt bleiben.

### 3.4 Arbeitsgruppen und Anwender

Die Anwender haben einige Anforderungen an ihre Arbeitsumgebung. Diese umfassen:

- einen inhaltlichen Austausch zwischen den Nutzern
- einen möglichst geringen Arbeitsaufwand für jeden einzelnen
- eine Einrichtung von Arbeitsgruppen
- eine schnelle und qualifizierte Unterstützung bei technischen Fragen

Diese Aspekte haben manchmal auch direkten Einfluss auf die Wahl und das Design von Hard- und Software. Wieder können verschiedene Lösungsoptionen in vielen Punkten die gleichen Leistungen erbringen, aber in einigen Bereichen gibt es große Unterschiede, sowohl bei der Realisierung als auch bei den Kosten.

Den intensiven inhaltlichen Austausch gibt es nur beim Grid-Computing, nicht aber beim Cloud-, Desktopgrid- oder lokalen Computing. Gerade akademische Nutzergruppen sind aber auf den Erfahrungsaustausch angewiesen und damit die typischen Kandidaten für das Grid-Computing.

Der Arbeitsaufwand für jeden einzelnen Nutzer steigt im Allgemeinen mit der Komplexität der Systeme, die zur Bewältigung der Berechnungen benötigt werden, und ist damit von der Lösungsoption weitestgehend unabhängig. Sicherlich sind lokale Lösungen am wenigsten aufwendig, gerade was z.B. die Notwendigkeit von Sicherheitszertifikaten angeht. Allerdings ist häufig bei kleinen Arbeitsgruppen nicht genügend Personal oder Geld vorhanden, um Datenablagen, Softwareanwendungen, etc. effizient und professionell zu organisieren, so dass der Vorteil der Einfachheit oft durch einen

Mehraufwand z.B. wegen unzureichender Dokumentation überwogen wird. Cloud-Computing dürfte in Bezug auf den Arbeitsaufwand für den individuellen Nutzer am günstigsten sein, sofern die Benutzung der externen Ressourcen leicht ist und Schulungen durchgeführt werden. Allerdings erfordert das Cloud-Computing, dass die Nutzer die zur Verfügung gestellten Systeme konfigurieren. Das kann weitaus zeitaufwendiger sein, als bestehende Grid-Ressourcen zu verwenden, die meist über Standardbibliotheken und -konfigurationen verfügen. Dafür fällt in der Regel ein erheblicher Mehraufwand in der „Gridifizierung“ der eingesetzten Software an. Beim Desktopgrid-Computing bedarf es meistens einer weiteren Abstraktionsschicht, die die Systembenutzung über Grenzen ermöglicht, die durch verschiedene Softwareversionen und -konfigurationen entstehen können.

Um umfangreiche Fragestellungen gemeinsam effizient zu bearbeiten, ist eine organisierte Aufteilung der Arbeit auf spezialisierte Anwender notwendig. Üblicherweise werden Arbeitsgruppen zu bestimmten Themenkomplexen eingerichtet, in denen jeweils mehrere Anwender zusammenarbeiten und sich gegebenenfalls mit anderen Arbeitsgruppen austauschen. Gerade bei Großkollaborationen ist eine solche Untergliederung unumgänglich. Für einen Anwender eines kleinen Instituts, der mit einem Kollegen einer anderen Einrichtung über Grid-Computing zusammenarbeitet, stellt dies jedoch in der Regel eine unnötige Bürokratisierung dar, weil er nur mit wenigen Personen direkt und schnell in Kontakt treten möchte. In einem solchen Fall reicht es, eine virtuelle Arbeitsgruppe einzurichten, so dass zwar themenfremde oder außenstehende Personen einen strukturierten Überblick erhalten und bei Bedarf Ansprechpartner schnell finden können, aber sich für die zusammenarbeitenden Forscher der Arbeitsalltag nicht unnötig verkompliziert.

Das Unterstützungspersonal schult die Nutzer und hilft ihnen bei aktuellen Fragestellungen weiter. Bei lokalen Lösungen fehlen oft die Mittel für solches Personal, so dass die Nutzer ihre Probleme ohne professionelle Hilfe gemeinsam oder allein lösen müssen. Manchmal gibt es jedoch eine hochspezialisierte Nutzerbetreuung, die seit langem mit der lokalen Lösung vertraut ist und präzise Hilfe leisten kann. In diesem Fall könnte eine lokale Lösung erste Wahl sein. Günstiger ist das Grid-Computing, bei dem in jeder Arbeitsgruppe mindestens ein interner, professioneller Ansprechpartner zur Verfügung steht, der mindestens die Strukturen auf seiner Seite genau kennt. Genau dies kann aber auch einen Nachteil darstellen, wenn die Ressourcen mehrerer Grid-Sites aufgrund ihrer Heterogenität nicht über ein einzelnes Tool in Anspruch genommen werden können. Dies ist beim Desktopgrid-Computing sogar der Regelfall. Liegt jedoch eine Homogenität vor, sind die Effizienz und die Qualität der Nutzerbetreuung maximal, so dass das Grid-Computing das beste Verhältnis von Nutzen zu Kosten hat.

Anforderungen, die minimalen Aufwand für die Nutzer beinhalten, sind nur mit lokalen Lösungen zu bewältigen, die dementsprechend kostenaufwendig sind. Liegen jedoch die Bedürfnisse bei Austauschbarkeit und Synergieeffekten, bietet Grid-Computing das beste Nutzen-zu-Kosten-Verhältnis. Desktopgrid- und Cloud-Computing liegen in der Regel bei den Anwenderanforderungen hinter dem Grid-Computing oder lokalen Lösungen zurück.

### 3.5 Anforderungen aus dem Datenmodell

Um eine Entscheidung für oder gegen die Nutzung eines Grids zu fällen, darf das zugrundeliegende Datenmodell nicht außer acht gelassen werden. Dabei ist vor allem die benötigte Datenmenge von großer Bedeutung. Konkret sind die wichtigsten Kriterien: Datenmenge, Datenraten, Latenzzeiten, Abhängigkeiten der Daten voneinander, Datensicherheit und Datenverfügbarkeit. Für die nachfolgenden Teilprobleme der Datenverfügbarkeit gibt es bereits Lösungen, welche die genannten

Grenzen des Speicherortes verschwimmen lassen (Hadoop, Google File System). Nichts desto trotz werden im folgenden die grundlegenden Kriterien für zwei Fälle

Handelt es sich im konkreten Fall um eine kleine Datenmenge (wenige GBytes), so ist es einfach möglich, diese Daten auf jeder Grid-Site bzw. innerhalb der Cloud vorzuhalten, sofern die Datenintegrität gewährleistet ist (Daten befinden sich am Ausführort des Job, bzw. werden zum Ausführort des Job transportiert). Dies erhöht automatisch die Datenraten, Datensicherheit (im Sinne von Verlust) und die Datenverfügbarkeit und verringert die Latenzzeiten durch einen lokalen Zugriff. Bestehen jedoch Abhängigkeiten der Daten voneinander, so ist sehr wahrscheinlich nur eine zentrale Speicherlösung ratsam bzw. evtl. sogar technisch nur möglich (z.Bsp. Datenbanken). Diese kann dann ebenfalls von allen Ansätzen (Grid, Desktopgrid, Cloudcomputing) benutzt werden, solange die Nutzung im Rahmen dessen bleiben, was die mögliche Datenrate und die notwendigen Latenzzeiten der WAN Anbindung nicht überfordert.

Im Gegensatz dazu besteht im Falle einer großen Datenmenge (mehrere TBytes) die Notwendigkeit, primär die Jobs zu den Daten zu schicken. Dies schließt eine Nutzung eines Desktopgrids aus. Dabei können, sofern nicht Abhängigkeiten dagegen sprechen, die Daten beliebig auf die Grid-Sites oder innerhalb der Cloud verteilt werden, wie es zum Beispiel im WLCG der Fall ist. Durch Replikation der Daten erreicht man zudem direkt eine Erhöhung der Datensicherheit (im Sinne von Verlust) und mit Einschränkungen der Datenverfügbarkeit. Problematisch kann dieser Ansatz werden, wenn die Daten unterschiedlich genutzt werden, so dass es auf einzelnen Grid-Sides bzw. in Teilen der Cloud zu einer verlängerten Umlaufzeit durch lange Wartezeiten führt. Dieses Problem kann durch eine (automatisierte) Erhöhung der verfügbaren Replikas der problematischen Daten gelöst werden.

Für mittlere Datenmengen müssen ebenfalls die oben genannten Kriterien berücksichtigt werden, wobei die entsprechende Wichtung individuell festgelegt werden muss. Aber auch hier beschränkt in der Regel die benötigte Datenmenge für einen Job eine Nutzung des Desktopgrids erheblich.

### 3.6 Softwareorientierte Anforderungen

Über die allgemeinen Anforderungen in Sektion 3.4 hinaus haben die Anwender einige Anforderungen, die die Wahl und das Design der Software betreffen:

- die Automatisierbarkeit der Berechnungen bzw. die Interaktivität der Software
- die Austauschbarkeit
- das Ordnungssystem der Daten

Wieder können verschiedene Lösungsoptionen in vielen Punkten die gleichen Leistungen erbringen, aber in einigen Bereichen gibt es große Unterschiede, sowohl bei der Realisierung als auch bei den Kosten.

Cluster-Computing, sei es in Form von Desktopgrid, Cloud, Grid oder lokalem, benötigt automatisierbare Software, die im Stapelverarbeitungsmodus (Batch-Modus) betrieben wird. Sind aber die Anwendungen interaktiv, kommt nur eine lokale Lösung in Frage.

Hinsichtlich der Austauschbarkeit sind mehrere Aspekte zu berücksichtigen. Software muss möglichst unabhängig von der Hardware austauschbar sein, falls z.B. ein Softwareproduzent die Unterstützung

einstellt. Außerdem müssen zur Verbesserung Updates einspielbar sein. Schließlich müssen die verwendeten Software- und Hardwareplattformen Schnittstellen zu anderen Systemen bereitstellen, um Informationen global mit anderen austauschen zu können und um über ein erweiterbares System zu verfügen. Damit langfristig die Daten von den Nutzern ungehindert verarbeitet und ausgetauscht werden können, muss ein wohldefiniertes und offen gelegtes Datenformat verwendet werden. All diese Aspekte hinsichtlich der Austauschbarkeit können von den verschiedenen Lösungsoptionen gleichermaßen abgedeckt werden, es gibt somit keine Option, die gegenüber einer anderen klar im Vorteil ist. Auch die Kosten für Austauschbarkeit sollten damit bei den verschiedenen Lösungen ähnlich sein. Es dürfte jedoch einen erheblichen Aufwand bedeuten, zu einer anderen Lösung zu wechseln, z.B. vom Cloud-Computing zum Grid-Computing, weil massive Umstellungen in Software und Arbeitsabläufen zu erwarten sind.

Ordnungssysteme werden primär für einen einfachen und schnellen Zugriff auf die Daten benötigt. Darüber hinaus fallen aber oft auch Metadaten oder weitere Organisationssysteme wie Nutzerverwaltung an. Sollen alle Daten in einer Datenbank abgelegt werden, auf die sehr schnell zugegriffen werden muss, kommt nur eine lokale Lösung mit einer zentralen Datenbank in Frage. Sind so hohe Anforderungen nicht notwendig, kann es gerade bei internationaler Zusammenarbeit und zur Lastverteilung sinnvoll sein, die Daten dezentral zu organisieren, sei es in Datenbanken oder geordneten Dateiablagen. Vor allem für das Grid-Computing ist eine solche Dezentralität typisch, aber auch beim Cloud- oder Desktopgrid-Computing besteht diese Möglichkeit. Gegebenenfalls kann durch Redundanz ständige Verfügbarkeit gesichert werden. Um aber einen schnellen und effizienten Zugriff zu gewährleisten, muss ein einheitliches Software-System existieren, das unabhängig von dem Ort und der Hardware ist, wo die Daten liegen. Genau dies ist ein wesentlicher Aspekt des Grid-Computings, weil durch die Gründung von Communities Infrastrukturen geschaffen werden, mit denen ein einheitlicher und leichter Zugriff auf (schon vorhandene) verteilte Daten ermöglicht wird.

Anforderungen, die Interaktivität oder schnelle Zugriffe auf fein geordnete Datenablagen beinhalten, sind nur mit lokalen Lösungen bewältigen, die dementsprechend kostenaufwendig sind. Liegen jedoch die Bedürfnisse bei Austauschbarkeit und Synergieeffekten, bietet Grid-Computing das beste Nutzen-zu-Kosten-Verhältnis. Desktopgrid- und Cloud-Computing liegen in der Regel bei diesen Aspekten hinter dem Grid-Computing oder lokalen Lösungen zurück.

### 3.7 Hardwareorientierte Bedingungen

Ähnlich wie Software muss Hardware austauschbar sein, d.h. reparierbar oder ersetzbar, am besten im laufenden Betrieb, um Ausfälle schnell beheben zu können und eine Langzeitarchivierung zu ermöglichen. Darüber hinaus sollte Hardware leicht erweiterbar sein, um den steigenden Anforderungen der Software gerecht zu werden.

Schließlich gibt es Bedingungen an Hardwareressourcen, die sich allein aus der thematischen Aufgabenstellung ergeben: die Menge der Daten, Berechnungen und Nutzer. Diese fallen pro Zeit an bzw. sind in einem geforderten Zeitintervall zu bewältigen, d.h. entsprechende Geschwindigkeiten und Zeitverläufe sind anzugeben. Darunter fallen: Datenzugriffszeit, Datenübertragungsrate und Nutzerrate. Diese können voneinander und vom Systemzustand abhängen und der Langzeitverlauf kann sehr unterschiedlich sein. Da Grid-Computing (globale) Verteilung von Berechnungen und Datenablage bedeutet, hängen diese Aspekte vom Ort ab.

Zu den konkreten, quantitativen Anforderungen ("Sollwerte") gehören:

- die absolute Datenmenge, die auf dem Storage abgelegt werden soll
- die Datenmenge, auf die pro Zeiteinheit lesend oder schreibend zugegriffen werden soll, d.h. die Verarbeitungsrate auf dem Storage und die Bandbreite zwischen ihm und dem Computingelement
- die Datenmenge, die pro Zeiteinheit vom oder zum Nutzer übertragen werden soll, d.h. die Datenübertragungsrate zwischen dem Storage und dem Nutzerelement
- die Datenzugriffszeit, d.h. die Zeit von der Anfrage bis zur Lieferung der Daten, sei es an das Computing- oder Nutzerelement
- die Zahl der Berechnungen, die pro Zeiteinheit durchgeführt werden sollen
- der Bedarf an Hauptspeicher, um die Berechnungen durchzuführen
- der Bedarf an temporärem Festplattenspeicher, um gegebenenfalls Ein- oder Ausgabedaten auf dem Computing-Element zwischenspeichern
- die Zahl der Nutzer, die pro Zeiteinheit die Ressourcen verwenden sollen
- der Bedarf an Information, z.B. über Systemzustand oder Systemgebrauch, der pro Zeiteinheit an die Nutzer fließen soll, d.h. die Nutzerunterstützungsrate
- Abhängigkeit der Daten voneinander

Im folgenden wird davon ausgegangen, dass die quantitativen Anforderungen schon ermittelt worden sind. Jede Lösungsoption (lokales, Peer2Peer-, Cloud- oder Grid-Computing) kann zwar in vielen Punkten die gleichen Sollwerte bereitstellen, aber in manchen Bereichen gibt es Unterschiede, sowohl bei der technischen Realisierung als auch bei den Kosten. Außerdem unterscheiden sich die Optionen vor allem hinsichtlich eines potentiellen Ausbaus der Ressourcen und des Zeitverlaufs der Istwerte.

Jede Lösung kann etwa die gleichen Sollwerte bereitstellen für fünf der neun quantitativen Anforderungspunkte: die absolute Datenmenge, die Verarbeitungsrate auf dem Storage und die Bandbreite zu den Computingelementen, die Berechnungsrate, der Hauptspeicher, der temporäre Festplattenplatz. Dies ist in der Tatsache begründet, dass jede Lösung äquivalente Hard- und Software verwenden kann. Somit sind auch die entsprechenden Kosten in etwa gleich groß.

Es gibt jedoch beträchtliche Unterschiede zwischen den Lösungsoptionen bei den übrigen 4 Punkten: Datenrate zwischen Storage und Nutzer, Datenzugriffszeit, Nutzerrate und Nutzerunterstützungsrate. Bei einer lokalen Lösung sind sicherlich die Datenrate zwischen dem Nutzer und dem Storage, die Datenzugriffszeit und die Nutzerunterstützungsrate besser als bei den anderen Optionen. Da beim verteilten Rechnen die Istwerte von Ort zu Ort variieren und eventuell von den Sollwerten abweichen können, ist zumindest bei datenintensiven Operationen z.B. ein Echtzeitbetrieb nicht möglich. Kleine Nutzerraten, die vor allem beim Storagezugriff erforderlich sind, sind bei lokalen Lösungen nicht garantierbar. Verteiltes Rechnen wie das Grid-Computing führt jedoch per constructionem zu keiner übermäßigen Nutzerlast. Im Gegenzug leiden Desktopgrid-, Cloud- und Grid-Computing öfters an langen Zugriffszeiten, einem zähen Informationsfluss zum Nutzer sowie potentiell geringeren Datenraten zwischen Storage und Nutzer. Eine lokale Lösung ist also bei Echtzeitberechnungen

erforderlich und scheint auch wegen der Vorteile bei jenen drei quantitativen Aspekten meist die bessere Wahl gegenüber verteiltem Rechnen zu sein.

Allerdings können lokale Lösungen normalerweise spontane Spitzenlasten nicht in einem engen Zeitfenster bewältigen, weil sie dann für den Normalbetrieb überdimensioniert und damit unrentabel wären. Ähnlich verhält es sich beim Cloud-Computing, bei dem nur eine feste Menge an externen Ressourcen zur Verfügung stehen. Da beim Desktopgrid-Computing das Bereitstellen von Ressourcen spontan erfolgt, variiert das Potenzial zur Spitzenlastbewältigung. Das Grid-Computing kann jedoch Spitzenlasten meist gerecht werden, weil Ressourcen von anderen Nutzern, die momentan nicht benötigt werden, unter Absprache zeitnah in Anspruch genommen werden können. Dieser Vorteil kann noch verstärkt werden, wenn mehrere Communities sich Ressourcen teilen.

Ein Community-Grid kann außerdem relativ leicht seine Ressourcen stark erweitern, indem z.B. neue Mitglieder mit eigenen Ressourcen aufgenommen werden. Da bestehende Mitglieder in gewissen Zeitintervallen ihre Ressourcen der technologischen Entwicklung anpassen, wachsen beim Grid-Computing in der Regel die zur Verfügung stehenden Mittel beständig. Eine große Erweiterung einer lokalen Lösung ist jedoch oft langwierig und aufwendig. Beim Cloud-Computing dürfte eine Erweiterung meist zwar einfach, aber nicht schnell möglich sein, weil mit externen Anbietern verhandelt werden muss. Die zur Verfügung stehenden Ressourcen können beim Desktopgrid-Computing aufgrund des lockeren Zusammenschlusses kaum vorhergesagt werden oder sogar zeitweilig abnehmen.

Sind also nur wenige Nutzer vorhanden, die an einer geringen Zahl von benachbarten Orten kleine Datenmengen mit sequenziellen Berechnungen verarbeiten, ist die Verwendung des Grid-Computings nicht notwendig bzw. von Nachteil. Wenn schneller Zugriff auf gewisse Informationen notwendig ist, z.B. in Echtzeit, ist eine lokale Ablage unerlässlich, etwa in einer Datenbank mit kurzer Antwortzeit. Fallen für global verstreute Nutzer viele Berechnungen in einer kontinuierlichen Menge an, dürfte Cloud-Computing vorteilhaft sein, wenn die Nutzer wenig Unterstützung bedürfen. Bei vielen (wissenschaftlichen) Projekten entsteht jedoch großer Rechenbedarf spontan. Außerdem werden Daten und Berechnungen üblicherweise von vielen Nutzern geteilt. Dementsprechend ist auch ein beträchtlicher Informationsaustausch zwischen den Nutzern erforderlich. Somit ist ein Community-Grid naheliegend, das all diese Möglichkeiten bietet.

### 3.8 Fazit

Bei der Diskussion der Anforderungen haben wir gesehen, dass einige Aspekte bei der Entscheidung für oder gegen die Verwendung des Grid-Computings nur eine untergeordnete Rolle spielen, weil sie z.B. von der Lösungsoption unabhängig sind. Dazu gehören:

- 5 Aspekte bezüglich der Hardware:  
die absolute Datenmenge, die Verarbeitungsraten auf dem Storage und die Bandbreite zu den Computingelementen, die Berechnungsraten, der Hauptspeicher und der temporäre Festplattenplatz
- 2 Aspekte bezüglich der Software:  
die Automatisierbarkeit, die Austauschbarkeit

Bei folgenden Punkten gibt es z.T. große Unterschiede zwischen den Lösungsoptionen:

- 4 Aspekte bezüglich der Hardware:  
die Datenrate zwischen Storage und Nutzer, die Datenzugriffszeit, die Nutzerrate und die Nutzerunterstützungsrate
- 1 Aspekt bezüglich der Software:  
die Ordnungssysteme
- alle Aspekte hinsichtlich der Anwender:  
der inhaltliche Austausch zwischen den Nutzern, der Arbeitsaufwand für den einzelnen, die Unterstützung bei technischen Fragen, die Einrichtung von Arbeitsgruppen
- alle Aspekte bezüglich des Managements:  
die juristischen Aspekte, die Realisierungsaspekte, die Personalaspekte, die finanziellen Aspekte, die Sicherheitsaspekte, der Außenauftritt, die Hierarchiestruktur, die politischen Aspekte
- Ähnlichkeiten oder Übereinstimmungen bei Themengebieten

Obwohl einige Punkte ein K.O.-Kriterium für den Einsatz des Grid-Computings darstellen, sind meistens alle anderen Lösungsoptionen weniger geeignet, gerade im akademischen Umfeld.

# Kapitel 4

## Schritt 2: Community-Aufbau

### 4.1 Überblick

Nach der Entscheidung für die Verwendung des Grid-Computings bleibt noch die Wahl zwischen der Gründung einer neuen oder dem Beitritt zu einer bestehenden Community. Ist die Institution thematisch einer bestehenden Community benachbart, so scheint ein Beitritt naheliegend. Zu manchen Themengebieten gibt es aber noch keine Community, und somit kommt nur ein Neuaufbau in Frage.

Um Kosten und Nutzen abzuschätzen, orientieren wir uns an der Vorgehensweise des vorhergehenden Kapitels, indem wir nicht einfach eine Zahl als Ergebnis liefern (einen Geldbetrag), sondern ein differenziertes Bild zeichnen, das wir auf K.O.-Kriterien untersuchen und dessen Gesamteindruck den Ausschlag gibt. Ebenso verwenden wir zur Analyse die Aspekte des vorhergehenden Kapitels: die Themengebiete, das Management, die Arbeitsgruppen und Anwender, die Software- und Hardwareanforderungen. Der Vorteil eines Beitritts liegt auf der Hand: Da die Community schon existiert, sind die bestehenden Strukturen in der Regel ausgereift und können oft ohne großen Aufwand übernommen werden. Allerdings ist ein Beitritt nur erfolgreich, wenn die bestehende Community die Arbeitsgruppe akzeptiert und wenn die Community zu den Anforderungen der Arbeitsgruppe passt. Weiterhin muss der eventuell anfallende Mehraufwand (durch zusätzlichen Supportbedarf, etc.) in der bestehenden Community ausgeglichen werden. Ist dies nicht der Fall, bietet sich ein Neuaufbau an, bei dem alle dedizierten Mitglieder jeden Punkt verhandeln können.

### 4.2 Gemeinsame Themengebiete

Eine Entscheidung für das Grid-Computing impliziert eine intensive und wohlorganisierte Zusammenarbeit von Institutionen mit gemeinsamen Themengebieten. Bei einem Beitritt zu einer bestehenden Community kann es jedoch Überlappungen zwischen den Themen der Community und der beitretenden Institution geben, so dass zwar ein Erfahrungsaustausch stattfinden kann, aber die beitretende Institution manchmal neue Themenbereiche wählen muss.

Wird eine neue Community gegründet, haben meistens die konstituierenden Mitglieder verschiedene Themenbereiche, die sie dann zu ihrer jeweiligen Bearbeitung behalten können.

In beiden Fällen, Beitritt und Neugründung, findet eine gegenseitige, thematische Bereicherung statt. Ein wohldefinierter und -kontrollierter Zugriff auf die Daten und Berechnungen anderer Mitglieder kann erfolgen, so dass etwaige, bisher unentdeckte Zusammenhänge gefunden werden können. Außerdem können Forschungsergebnisse gemeinsam schneller erzielt werden, nicht nur durch den Datenaustausch, sondern auch durch den Erfahrungsaustausch.

### 4.3 Vorgaben des Managements

Sowohl beim Beitritt als auch bei der Gründung einer neuen Community werden in der Regel nicht nur Daten und Erfahrungen ausgetauscht, sondern auch Kompetenzen abgegeben und gewonnen. Dies betrifft meistens auch viele Bereiche, die unter der Verantwortung der Leitung stehen: Gesetzliche Rahmenbedingungen, Realisierung, Personal, Finanzen, Sicherheit, Außenauftritt, Hierarchiestruktur, Politik.

Würde der Beitritt zu einer bestehenden Community den gesetzlichen Rahmenbedingungen oder einer Betriebsordnung, denen der Kandidat oder die Community unterliegt, zuwiderlaufen, kommt nur eine Neugründung in Frage.

Hinsichtlich der Realisierung des Grid-Computings unterscheiden sich Beitritt und Neuaufbau massiv. Bei ersterem werden einfach bestehende, ausgereifte Strukturen und Prozesse übernommen, was abgesehen von manchen Softwareschnittstellen recht schnell von statten gehen kann. Bei einem Neuaufbau müssen die Strukturen und Prozesse zunächst definiert und erstellt werden, was beträchtliche Zeit in Anspruch nehmen kann.

Der Einstieg ins Grid-Computing verursacht auch Änderungen beim Personal: Ansprechpartner für externe Nutzer müssen zur Verfügung gestellt werden, umgekehrt können aber Nutzer auch externe Berater zu Hilfe nehmen. Hier unterscheiden sich Beitritt zu und Neugründung einer Community nicht wesentlich. Zusätzliche Stellenmittel können gewonnen werden, aber auch die Arbeitszeitbelastungen können ansteigen.

Von der finanziellen Seite aus betrachtet kann das Grid-Computing ein Gewinn- oder Verlustgeschäft sein. Wenn eigene Ressourcen nicht ausgelastet sind, sondern anderen evtl. sogar gegen Geld zur Verfügung gestellt werden, stellt dies einen Mehrwert für beide Seiten dar. Falls jedoch die Nutzungsabrechnung nicht gerecht ist oder die Nutzer wegen Fehlorganisation weniger Rechenzeit oder Ablageplatz als vorher haben, überwiegen die Nachteile für die eine oder andere Seite. Ein Beitritt zu einer bestehenden Community dürfte zwar recht leicht sein, aber das Nutzungsabrechnungsschema ist dann auch meist vordefiniert und wohl kaum verhandelbar. Eine Neugründung hingegen berücksichtigt oft besser individuelle Gegebenheiten.

Zahlreiche Sicherheitsaspekte spielen beim Grid-Computing eine Rolle, z.B. die physikalische Datensicherheit, die Übertragungssicherheit und die Datenzugriffssicherheit. Während die physikalische Datensicherheit durch die Verwendung des Grid üblicherweise stark erhöht wird (globale Redundanz), hängen die Übertragungssicherheit und die Datenzugriffssicherheit von der Implementierung ab. Bei einem Beitritt zu einer bestehenden Community sind die entsprechenden Standards bereits festgelegt, und kurzfristige Änderungen sind unwahrscheinlich. Eine Neugründung kann jedoch diese Standards an die Bedürfnisse aller konstituierenden Mitglieder anpassen.

Hinsichtlich des Außenauftritts unterscheiden sich Beitritt und Neugründung massiv. Bei einer bestehenden Community ist der gemeinsame Außenauftritt schon etabliert und wird üblicherweise

höchstens in Details an das neue Mitglied angepaßt werden. Bei einer Neugründung müssen die Mitglieder gemeinsam einen Außenauftritt kreieren und können hierzu ihre volle Kreativität einbringen.

Bei der Hierarchiestruktur sind die Unterschiede analog zum Außenauftritt: Eine bestehende Community ist mit Hierarchien strukturiert, in die der Beitrittskandidat eingegliedert wird. Bei einem Neuaufbau hingegen können die Hierarchiestufen von den Mitgliedern in Übereinstimmung festgelegt werden.

Schließlich spielen politische Aspekte beim Einsatz des Grid-Computings eine Rolle, vor allem bei der Entscheidung, mit wem zusammengearbeitet wird. Dabei darf die politische Beständigkeit und die Laufzeit der Finanzierung der in Frage kommenden Partner(länder) nicht außer acht gelassen werden. Gerade bei bestehenden, langjährigen Zusammenarbeiten bietet sich eine Intensivierung der Beziehungen an, sei es zum Beitritt oder zur Neugründung.

#### **4.4 Arbeitsgruppen und Anwender**

Wenn die Nutzer des Beitrittskandidaten schon Mitglieder der (konstituierenden) Community kennen, z.B. durch bestehende Zusammenarbeit, fällt der Grid-Einstieg besonders leicht. Dennoch muss der inhaltliche Austausch intensiviert werden, damit die Zusammenarbeit effizient ist, was gerade beim Grid-Computing wichtig ist. Kennen sich die Nutzer nicht, dürfte eine Ausweitung der Zusammenarbeit aufgrund der allseitigen Aufbruchstimmung leicht gelingen. Tritt jedoch eine Institution einer Community bei, ist es oft schwierig, aufgrund der dort bestehenden Arbeitsbeziehungen eine deutliche Sichtbarkeit und intensive Zusammenarbeit zu erlangen. Hiervon ist auch die Gründung von Community-internen Arbeitsgruppen betroffen, die gemeinsam Themen bearbeiten. Wird jedoch eine neue Community aufgebaut, werden zur internen Organisation und thematischen Zusammenarbeit Arbeitsgruppen gegründet, die Nutzer aus möglichst allen Institutionen umfassen.

Anwender akzeptieren neue Lösungen und Prozesse wie den Einsatz des Grid-Computings bereitwillig, wenn es für sie keinen Arbeitsmehraufwand bedeutet bzw. ihnen genügend Vorteile bietet. Neugründung und Beitritt unterscheiden sich hier wohl kaum, außer dass eine Neugründung im allgemeinen länger dauert, aber sicherlich auch Potential hat, die Benutzbarkeit zu verbessern.

Falls technische Fragen oder Probleme auftreten, ist es unerlässlich, qualifizierte Unterstützung zu haben, die den Nutzern schnell und erfolgsorientiert hilft. Es dürfte utopisch sein, dass Hilfe persönlich und sofort zur Verfügung steht, weil die Anwender im Rahmen des Grid-Computings (global) verteilt zusammenarbeiten. Dies betrifft jedoch einen Beitritt wie eine Neugründung gleichermaßen, wenn man von wie oben erwähnten, etwaigen Integrationsproblemen absieht. Allerdings lassen sich mit Hilfe einer großen Vielfalt von Kommunikationsmitteln wie E-Mail, Chat, Telefon, etc., die heutzutage zur Verfügung stehen, zufriedenstellende Supportzeiten erreichen. Dies klappt sogar um so besser, je internationaler die Community ist, da in der Regel aufgrund der Zeitverschiebung zu jeder Zeit genügend Expertise vorhanden ist.

#### **4.5 Softwareorientierte Anforderungen**

Für den Beitritt zu wie die Neugründung einer Community gilt gleichermaßen, dass die Automatisierbarkeit der EDV unumgänglich ist. Grid-Computing kann riesige Datenmengen und/oder lange,

aufwändige Berechnungen, die interaktiv nicht bewältigbar sind umfassen. Dementsprechend muss die zur Datenverarbeitung verwendete Software ein Design und eine Implementierung haben, die dem Nutzer möglichst viel Arbeit abnimmt und einen schnellen Erfolg beschert.

Ebenso sollte die Software maximale Austauschbarkeit gewährleisten. Sowohl weitgehende Unabhängigkeit von der Hardware als auch wohldefinierte und offen gelegte Datenformate sind Voraussetzung für eine erweiterbare Zusammenarbeit und den langfristigen Austausch von Informationen. Dementsprechend ist auch eine wohldokumentierte Software notwendig, um Schnittstellen zwischen den Institutionen einrichten zu können. Hier treten erfahrungsgemäß große Probleme bei einem Beitritt ebenso wie bei einer Neugründung auf, weil die Systeme oft inkompatibel zu einander sind. Ob eine neu zu schaffende technische Lösung möglich ist wird je nach Komplexität nur durch einen Proof of Concept festzustellen sein, was die Übergangszeit zusätzlich verlängert. Dabei ist von vornherein zu berücksichtigen, in wie weit die Skalierbarkeit und Verfügbarkeit zentraler Dienste gewährleistet ist, da diese eine schwer austauschbare Basis für den Betrieb bedeutet. Tritt eine Institution einer bestehenden Community bei, sind Softwarestandards üblicherweise nicht verhandelbar, d.h. der Kandidat hat sich anzupassen und die Lösungen zu übernehmen bzw. geeignete Schnittstellen zur Verfügung zu stellen. Wird dagegen eine neue Community gegründet, stehen die Institutionen vor der Wahl, ihre Systeme über zu schaffende Schnittstellen miteinander zu verbinden oder gleiche Systemstandards für alle festzulegen, was explizit eine Runderneuerung der Software darstellt. Mit der Bereitstellung der Zugriffsmöglichkeit müssen aber auch Zugriffsrechte eingerichtet und angepasst werden, um Datenschutz und -integrität zu gewährleisten.

Schließlich muss ein effizientes Ordnungssystem der Daten existieren, das einen schnellen, gridweiten, aber kontrollierten Zugriff ermöglicht. Dies impliziert auch ein standardisiertes Metasystem, damit Informationen über alle Datensätze verschiedener Institutionen gleichzeitig, homogen, leicht und schnell erhältlich sind. Hier gibt es oft sogar noch mehr Probleme als bei der Software selbst, weil Datenformate und -beschreibung beliebig sein können, während für Anwendungen üblicherweise standardisierte Programmiersprachen und Designmuster verwendet werden.

## 4.6 Hardwareorientierte Bedingungen

Die Hardware hat in der Regel den geringsten Anteil an der Entscheidung zwischen Beitritt zu und Gründung einer Community. Oftmals kann sie durch eine Abstraktionsschicht (zum Beispiel durch Virtualisierung) vor der Anwendung verdeckt werden. Zudem besteht beim Grid-Computing in der Regel noch keine starken Einschränkungen hinsichtlich der Wahl der Hardwareplattform, während bei Softwareschnittstellen üblicherweise viel größere Schwierigkeiten auftreten, wie wir oben gesehen haben. Jedoch wird der allgemeine Trend zu GPGPU (General Purpose Computation on Graphics Processing Unit) für die Berechnung von parallelen Algorithmen diese Abhängigkeit zunehmend erhöhen, so dass für diese Zwecke oftmals nur eine lokale Lösung bzw. eine homogene Lösung in Betracht kommen wird.

Ein Beitritt ist günstiger, wenn der Kandidat über keine eigenen Hardwareressourcen verfügt und die Community auch keinen Beitrag von Hardware verlangt; dementsprechend wird nur eine Nutzungsabrechnung erfolgen. Sind beim Kandidaten Ressourcen vorhanden und hat die Community einen Hardwareüberschuss im Verhältnis zum Nutzerbedarf, ist eine Neugründung in der Regel günstiger, weil sich die Hardware schneller amortisiert.

Falls eine bestehende Community über große Hardwareressourcen verfügt, die niedrig ausgelastet

sind, dürfte ein Beitritt günstiger sein, andernfalls sollte eine Neugründung erwogen werden, damit die Wartezeiten für den Erhalt der Berechnungsergebnisse minimiert werden.

Sowohl bei einer Neugründung als auch einem Beitritt muss überprüft werden, ob die Dimensionierung und Organisation der Metasysteme, die Informationen über vorhandene Daten schnell liefern müssen, der zu erwartenden Nutzungssteigerung entspricht. Andernfalls dürfte es zu einem massiven Performancerückgang oder sogar einem Zusammenbruch der Metasysteme kommen.

Schließlich können sich Beitritt und Neugründung hinsichtlich der Langzeitarchivierung von Daten unterscheiden. Wenn bestehende Communities nicht die Bedürfnisse des Kandidaten erfüllen, z.B. bei der Planung oder den Verträgen mit Hardwarelieferanten, dann ist eine Neugründung wohl angemessen. Andererseits gibt es Kandidaten, die diesen Aspekt bisher nicht berücksichtigt oder keinen Bedarf haben. In diesem Fall ist ein Beitritt wohl günstiger.

## 4.7 Fazit

Bei der Diskussion der fünf Anforderungstypen haben wir gesehen, dass einige Aspekte bei der Entscheidung für oder gegen den Aufbau eines neuen Community-Grids nur eine untergeordnete Rolle spielen. Dazu gehören:

- 4 Aspekte bezüglich der Hardware:  
Langzeitarchivierung, Schnittstelle, Plattform, Dimensionierung der Metasysteme
- alle Aspekte bezüglich der Software:  
Automatisierbarkeit, Austauschbarkeit, Ordnungssystem
- 2 Aspekte hinsichtlich der Anwender:  
Arbeitsmehraufwand, Unterstützung
- 2 Aspekte bezüglich des Managements:  
Personal, Finanzen
- Bereicherung und Erfahrungsaustausch bei den Themengebieten

Bei folgenden Punkte gibt es z.T. große Unterschiede zwischen einem Beitritt und einer Neugründung:

- 2 Aspekte bezüglich der Hardware:  
Nutzungsabrechnung/Einbringen eigener Hardware, Menge/Auslastung der Ressourcen
- 1 Aspekt hinsichtlich der Anwender:  
Zusammenarbeit/Sichtbarkeit,
- 4 Aspekte bezüglich des Managements:  
Gesetzeslage, Realisierung, Sicherheitsaspekte, Außenauftritt, Hierarchiestruktur, Politik

Trotz dieser Unterschiede dürfte aber oft die Faustregel zutreffen, dass sich nach der Entscheidung für das Grid-Computing der Beitritt zu einer Community lohnt, wenn diese thematisch nahe liegt und die Nutzungsprofile sich ergänzen, und dass ansonsten wohl nur ein Neuaufbau in Frage kommt.

# Kapitel 5

## Zusammenfassung

Das Grid-Computing ist dabei, weltweite Standardtechnologie für bestimmte Anwendungsfälle zu werden. Gerade bei der akademischen Zusammenarbeit sind Community-Grids unverzichtbar geworden. Wegen des laufenden starken Wachstums auf diesem Sektor sind Beitritte von Institutionen zu bestehenden Communities und Gründungen von weiteren Grids zu erwarten. Auch Austritte, Ausgründungen oder Communitywechsel dürften die nächsten Jahre gehäuft auftreten, bis das Grid-Computing vollständig etabliert ist und eine Konvergenz bei den Communities erreicht ist. Für Institutionen mit Interesse am Grid-Computing ist es wichtig, vor Entscheidungen für oder gegen das Grid-Computing bzw. für oder gegen einen Communityaufbau den Nutzen und die Kosten abzuschätzen, die daraus resultieren. Viele Faktoren müssen berücksichtigt werden, sowohl hinsichtlich Hardware, Software, Anwendern, Management als auch Themengebieten. Hierbei dient dieses Dokument als Übersicht und Entscheidungshilfe. Bei einer positiven Wahl stehen zur Realisierung z.B. die Publikationen der WissGrid-Arbeitsgruppe "Blaupausen für neue Community-Grids" zur Verfügung [WG10b]. Darüber hinaus ist eine persönliche Kontaktierung von Fachberatern (fachberater@wissgrid.de) anzuraten und dürfte wohl sogar für eine schnelle und reibungslose Realisierung unumgänglich sein.

# Literaturverzeichnis

- [AG10] *AstroGrid-D – German Astronomy Community Grid (GACG)*. [www.astrogrid-d.org/](http://www.astrogrid-d.org/)  
(Stand: April 2010)
- [C3G10a] *C3-Grid – Collaborative Climate Community Data and Processing Grid*. [www.c3grid.de/](http://www.c3grid.de/)  
(Stand: April 2010)
- [DFN10] *Deutsches Forschungsnetz Entgelte* <http://www.dfn.de/dienstleistungen/dfninternet/entgelte/> (Stand: April 2010)
- [HG10] *D-Grid Initiative: HEP Grid*. [www.hepcg.org/](http://www.hepcg.org/) (Stand: April 2010)
- [IG10] *D-Grid Initiative: In-Grid*. <http://www.ingrid-info.de/> (Stand: April 2010)
- [MG10a] *MediGRID – Home*. [www.medigrid.de/](http://www.medigrid.de/) (Stand: April 2010)
- [P04] *Pawellek, Irene: Spielregeln für Netzwerke: Wenn Unternehmen zusammenarbeiten; wie Unternehmen zusammenarbeiten; ein Praxisleitfaden. 2004*
- [TG10a] *TextGrid – Vernetzte Forschungsumgebung in den eHumanities*. [www.textgrid.de/](http://www.textgrid.de/)  
(Stand: April 2010)
- [WG10b] *WissGrid: Deliverables*. <http://www.wissgrid.de/publikationen/deliverables/wp2.html> (Stand: April 2010)
- [LCG08] *Ian Bird, Tony Cass, Bernd Panzer-Steindel, Les Robertson - CERN/IT: Summary of the Plan for ensuring the Infrastructure needed to meet the Computing Needs for Physics at CERN in the Next Decade* <http://lcg.web.cern.ch/LCG/documents/PlantomeetLHCExperimentRequirementsatCERN-Summary.pdf>
- [EGEE08] *AN EGEE COMPARATIVE STUDY: GRIDS AND CLOUDS- EVOLUTION OR REVOLUTION?* <https://edms.cern.ch/file/925013/3/EGEE-Grid-Cloud.pdf>
- [ARXIV08] *Ian Foster, Yong Zhao, Ioan Raicu, Shiyong Lu: Cloud Computing and Grid Computing 360-Degree Compared* <http://arxiv.org/pdf/0901.0131>
- [CO09] *Edward Walker: The Real Cost of a CPU Hour* <http://www.computer.org/portal/web/csdl/doi/10.1109/MC.2009.135>