

Fachbereich Computerwissenschaften  
Universität Salzburg

Seminar aus Informatik:

## **High Performance Computing in the Cloud**

Erich Mraz, B.Eng.  
Maximilian Schorr, B.Eng.

**LV Leiter:**  
Prof. Dr. Wolfgang Pree

19.Juli.2013

Universität Salzburg  
Fachbereich Computerwissenschaften  
Jakob-Haringer-Straße 2  
A-5020 Salzburg  
Austria

# Inhalt

<b>1</b>	<b>Einleitung</b>	<b>1</b>
<b>2</b>	<b>Allgemeiner HPC Infrastrukturen Vergleich</b>	<b>1</b>
2.1	Cluster Computing . . . . .	2
2.2	Grid Computing . . . . .	2
2.3	Cloud Computing . . . . .	2
<b>3</b>	<b>HPC Applikationen und die Cloud</b>	<b>3</b>
3.1	Eigenschaften von HPC Applikationen . . . . .	3
3.2	Einsatz der Cloud bei HPC Applikationen . . . . .	3
3.3	Anwendungsmodi . . . . .	4
<b>4</b>	<b>Benchmarks</b>	<b>7</b>
4.1	Einleitung . . . . .	7
4.2	Setup . . . . .	8
4.3	Hardware und Software Konfigurationen . . . . .	9
4.4	Performance Analyse . . . . .	10
4.5	Kosteneffizienz . . . . .	14
<b>5</b>	<b>Zusammenfassung</b>	<b>17</b>
<b>6</b>	<b>Referenzen</b>	<b>18</b>

## 1 Einleitung

Cloud Computing bewirkt derzeit einen Umbruch im IT-Bereich, durch das Anbieten von on-demand Zugang zu einer Abstraktion jederzeit verfügbarer, unlimitierter Ressourcen. Diese Ressourcen passen sich flexibel der benötigten Kapazität an und bezahlt wird gemäß dem Umfang der Inanspruchnahme. Die virtuellen Ressourcen werden letztendlich gemietet und erlauben somit eine einfache Kontrolle der Kosten, sowie ein ansprechendes Kosten/Leistungs Verhältnis.

Gleichzeitig ermöglicht die allgemein zunehmende Datenverarbeitungs- und Rechenleistung neue Herangehensweisen in der wissenschaftlichen und industriellen Forschung, insbesondere im HPC-Bereich (High Performance Computing). Inzwischen können riesige Informations-/Datenmengen zur Lösung komplexer Probleme (teilweise auch kollaborativ) verarbeitet werden.

Clouds stellen diese Rechenleistung on-demand zur Verfügung und lassen Anschaffung, Einrichtung und Pflege der HPC-Infrastruktur hinfällig werden. Der Anwender kann sich einzig und allein auf seine eigentliche Aufgabe konzentrieren. Forschungsergebnisse können auf einfache Art und Weise über die Cloud anderen Personen zugänglich gemacht werden. Dadurch wird die Produktivität erheblich gesteigert. Neben all den Vorteilen, existieren auch einige Limitierungen und Stolpersteine beim Einsatz von HPC in der Cloud.

Die folgende Abhandlung beinhaltet einen Überblick über die verschiedenen gebräuchlichen HPC-Infrastrukturen, eine Beschreibung der Charakteristika und Limitierungen verschiedener Konzepte von HPC auf einer Cloud, sowie einen detaillierten Kosten/Leistungs Vergleich der gängigsten public Clouds untereinander und mit einem Cluster.

## 2 Allgemeiner HPC Infrastrukturen Vergleich

In den Quellen [3] und [8] wird das Konzept Cloud Computing mit herkömmlichen HPC-Infrastruktur Alternativen verglichen. Es wird eine generelle Definition, sowie eine Abgrenzung von Cloud Computing zu den genannten Alternativen erarbeitet. Viele von der Cloud verwendete Technologien und Ideen sind keinesfalls neu, sondern wurden von bisheriger HPC-Infrastruktur übernommen und teilweise weiterentwickelt.

## 2.1 Cluster Computing

Ein Cluster ist ein verteiltes, paralleles System, bestehend aus einer Ansammlung eng gekoppelter Einzelrechner. Die einzelnen (homogenen) Rechereinheiten befinden sich an einem zentralen Standort. Sie sind üblicherweise durch ein sehr schnelles LAN verbunden und arbeiten als einheitlich integriertes System zusammen. Mittels zentralisiertem Job-Management und Scheduling wird ein SSI System (Single System Image) bereitgestellt. Dieses Konzept erlaubt den Einsatz von hoher Performance zu geringem Preis und guter Skalierbarkeit. Ein Cluster ist gegenüber einem einzelnen (Super-) Computer mit vergleichbarer Performance eine ökonomische, kosteneffektivere Lösung für HPC-Anwendungen.

## 2.2 Grid Computing

Als Grid wird ein paralleles, verteiltes System bezeichnet, welches Rechenressourcen unterschiedlicher administrativer Domains kombiniert. Hier befinden sich die zum Großteil heterogenen Ressourcen an verschiedenen Standorten und werden über LAN, WAN und MAN lose gekoppelt miteinander verbunden. Die Heterogenität und Dynamik der vorhandenen Ressourcen erfordert ein (über die Knoten) verteiltes Job-Management und Scheduling. Im Gegensatz zum Cluster sind die einzelnen Knoten autonom, sie besitzen einen eigenen Ressourcen-Manager und agieren als unabhängige Einheit. Die Ressourcen werden also dynamisch zur Laufzeit geteilt, ausgewählt und zusammengelegt, abhängig von ihrer aktuellen Verfügbarkeit im Netz, Leistungsfähigkeiten, Performance, Kosten und QoS Ansprüchen. Die einzelnen Organisationen, die ihre Ressourcen dem Pool zur Verfügung stellen, teilen sich die vorhandenen Kapazitäten. Aussenstehende können zu fixen Kosten bestimmte Leistungen in Anspruch nehmen.

## 2.3 Cloud Computing

Hintergrund der Cloud ist ebenfalls ein paralleles, verteiltes System. Wie beim Cluster sind die Ressourcen örtlich zusammengefasst und es wird zumindest eine teilweise Homogenität erreicht, um besseres Hardware-Management und somit bessere Performance zu gewährleisten. Alle Anwendungsdaten liegen auf einem zentralen Server. Cloud-Kapazitäten sind ähnlich dem Grid on-demand über Internet zugänglich. Durch eine Virtualisierung der vorhandenen bzw. benötigten Ressourcen wird die Zugänglichkeit und Usability hier allerdings stark vereinfacht. Die Cloud stellt die benötigten Kapazitäten dynamisch bereit, dargestellt als eine oder mehrere unabhängige Recheneinheiten. Diese Virtualisierung erlaubt eine Skalierung entsprechend variabler Last und optimale Ressourcenausnutzung. Infolgedessen ermöglicht die Abrechnung über ein pay-per-use Modell es jedem Interessenten kosteneffektiv, on-the-fly und in Echtzeit seine IT-Kapazität zu erhöhen

(oder Leistungsfähigkeiten auszubauen). Die Cloud übernimmt das Bereitstellen, Konfigurieren und Verwalten der Infrastruktur und ermöglicht es Forschern sich auf die Wissenschaft zu fokussieren und Kollegen einfach und schnell Zugang zu Aufbau von Software und Experimenten in der Cloud zu gestatten.

## 3 HPC Applikationen und die Cloud

Nachdem bereits die Vorteile der Cloud hinsichtlich Usability und Kosteneffizienz herausgearbeitet wurden, betrachten wir anschließend die Cloud-Performance speziell bei HPC-Anwendungen. Dazu muss man zunächst die Eigenschaften einer HPC-Anwendung kennen. Quelle [5] geht auf die Möglichkeiten und Herausforderungen des momentanen Standes von HPC in der Cloud ein.

### 3.1 Eigenschaften von HPC Applikationen

HPC-Applikationen zeichnen sich aus durch enge Kopplung (einzelner Prozesse) mittels einer großen Menge an Interprozessorkommunikation. Sie benötigen typischerweise großen Aufwand an Rechenleistung innerhalb einer verhältnismäßig kurzen Zeitspanne. Es sind folglich Ressourcen in großem Umfang und schnelle (Netzwerk-)Verbindungen zwischen den einzelnen Prozessoren erforderlich.

### 3.2 Einsatz der Cloud bei HPC Applikationen

Cloud Computing Services können HPC in mehrerlei Hinsicht unterstützen. Sie stellen eine Plattform für HPC-Applikationen bereit, wenn lokale Infrastruktur nicht (ausreichend) verfügbar ist oder ergänzen existierende lokale Plattformen mit zusätzlichen Kapazitäten und Leistungsfähigkeiten. Darüber hinaus verbessern Cloud Services Usability und somit Produktivität von traditionellem High Performance Computing, indem sie beispielsweise bereits funktionsfähig eingerichtete und vorkonfigurierte Arbeitsumgebungen für HPC zur Verfügung stellen. Dessen ungeachtet sind die bereitgestellten Ressourcen aufgrund ihrer Virtualisierung hochgradig elastisch und leicht konfigurierbar mit der Fähigkeit problemlos zu skalieren und sich variabler Last anzupassen. Die Einfachheit der von der Cloud angebotenen Abstraktionsschichten entschärft einige Probleme mit denen HPC auf herkömmlichen Infrastrukturen konfrontiert wird. In vielen Fällen reduziert der Einsatz der Cloud zugleich die anfallenden Kosten.

### 3.3 Anwendungsmodi

Manish Parashar et al. arbeiten in ihrer Publikation [5] drei Haupt-Anwendungsformen heraus, dargestellt in Abbildung 1. Jeder Modus bringt gewisse Vor- und Nachteile, bzw. Einschränkungen mit sich.

1. **HPC in the Cloud** - gesamte HPC-Applikation auf eine gängige public und/oder private Cloud Plattform auslagern
2. **HPC plus Cloud** - bestehende herkömmliche HPC/Grid Ressourcen mit Cloud Services komplementieren und unterstützen
3. **HPC as a Cloud** - HPC/Grid Ressourcen als speziellen Cloud Service zur Verfügung stellen

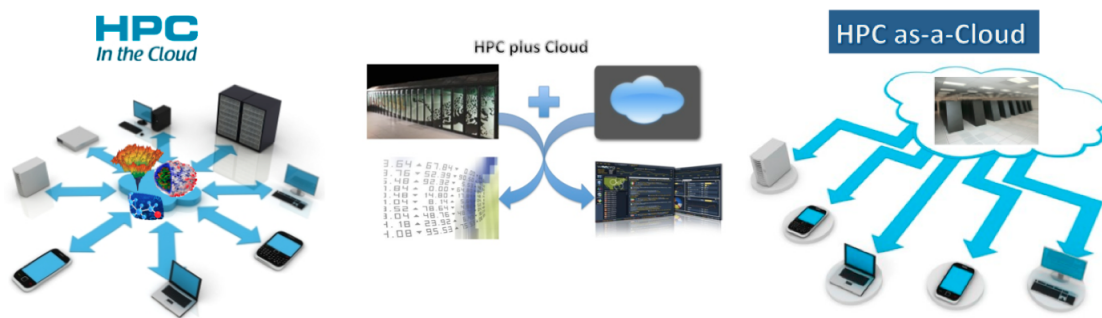


Abbildung 1: Hybrid HPC/Grid & Cloud-Anwendungsformen

#### 3.3.1 HPC in the Cloud

In die gängigen bekannten Cloud Plattformen können nur bestimmte Arten von HPC-Applikationen effektiv ausgelagert werden. Dies sind Applikationen mäßigen Umfangs mit minimalen Ansprüchen an Synchronisation und (Interprozessor) Kommunikation, sowie geringen I/O Ansprüchen. Die Einschränkungen ergeben sich vor allem aus der Beschaffenheit des Netzwerks in dem die einzelnen Cloud Ressourcen lokal verbunden sind. Mit abnehmender Parallelität der Applikation, müssen die auf verschiedene Einzelrechner verteilten Prozesse vermehrt Daten austauschen. Dementsprechend wachsen die Anforderungen an Bandbreite und Latenz des Netzwerks. Deshalb sind herkömmliche Clouds nur geeignet für HPC-Applikationen mit sehr hoher Parallelität, die voneinander unabhängige Datenpakete analysieren oder Simulationen durchführen. Um die einzelnen Teilergebnisse zusammenzuführen werden dann MapReduce-artige Algorithmen verwendet.

Verschiedene Varianten von MapReduce Berechnungen lassen sich aufgrund ihrer hohen Parallelität gut auf der Cloud umsetzen.

Verschiedene Aspekte der herkömmlichen kommerziellen Clouds verhindern nach wie vor eine generalisierte Umsetzung von HPC-Applikationen auf der Cloud. Neben dem bereits genannten Fehlen von High-Speed (Netzwerk-)Verbindungen, können mangelnde Leistungsfähigkeiten oder Homogenität der Cloud-Hardware ebenso HPC erschweren. Hohe Latenzen hemmen die Vorteile der Cloud in Skalierbarkeit, Flexibilität und Fehler-Toleranz. In vielen Fällen müssen geeignete alternative Code-Formulierungen gefunden werden, um eine spezielle HPC-Applikation erfolgreich auf eine Cloud zu portieren.

### 3.3.2 HPC plus Cloud

Eine weitere attraktive Einsatzmöglichkeit der Cloud bei HPC ist das sogenannte *Cloud Bursting*. Dabei wird die HPC-Applikation zunächst auf einer oder mehreren traditionellen HPC-Ressourcen ausgeführt. Bei plötzlich eintretendem Bedarf (spike in demand) nach mehr Rechenkapazität oder speziellen Leistungsfähigkeiten, zum Beispiel um heterogene Applikationsanforderungen zu erfüllen, wird ein Teil der Berechnungen auf die Cloud ausgelagert. Diese Hybrid-Infrastruktur birgt das Potential zu effektiveren Anwendungsformen, bei denen die Cloud zusätzlichen Speedup, Stabilität und ein besseres Kosten/Leistungsverhältnis ermöglicht. Kapazitäten und/oder Kapabilitäten können dynamisch zur Laufzeit hinzugefügt bzw. entfernt werden.

Die dadurch entstehende Dynamik und Inhomogenität des Systems bewirkt substanzielle Herausforderungen. Hardware-Ressourcen mit unterschiedlicher Konfiguration, Performance, Verfügbarkeit, Laufzeitverhalten und Stabilität müssen aufeinander abgestimmt werden. Auf Applikationsseite ändern sich die Anforderungen und Einschränkungen ebenfalls stetig. Die Bereitstellung und das Scheduling geeigneter Ressourcen, um die gesamte Laufzeit hindurch einen optimalen Kompromiss aus Kosten und Leistung zu erreichen, kann nicht manuell erfolgen und erfordert autonomes Monitoring und Management des dynamischen Verhaltens und der Anforderungen.

### 3.3.3 HPC as a Cloud

Aufgrund der Limitierungen von gewöhnlichen Cloud Plattformen in Bezug auf HPC, haben Cloud Provider inzwischen den Bedarf nach speziell für HPC entworfenen Cloud Konzepten erkannt. Sie bieten Hardware mit schnelleren Prozessoren und Verbindungen an und verzichten zum Teil sogar wieder auf eine Virtualisierung

der Ressourcen, um für maximale Performance auf die Hardware optimierte Programmierung zu erlauben. Diese Cloud Services werden üblicherweise als *HPC as a Cloud* bezeichnet. Sie bieten on-demand HPC-Ressourcen mittels Cloud Computing-Abstraktionen an und erreichen so die Vorteile des Cloud-Modells ohne Abstriche bei der Performance in Kauf nehmen zu müssen. Es gibt zwei Ansätze zu *HPC as a Cloud*, welche in Tabelle 1 zusammengefasst werden.

Große HPC Systeme	Kleine verbundene HPC Cluster	
Physikalische Lösung	Physikalische Lösung	Virtualisierte Lösung
z.B. Blue Gene/P as a Service	z.B. Penguin on Demand, Silicon Graphics, HPCynergy	z.B. EC2 Compute Cluster, Azure, Adaptive Systems, CloudCycle

**Tabelle 1:** HPC as a Cloud Klassifikation

Während *HPC as a Cloud* Usability und Produktivität von HPC wesentlich verbessert und einige Probleme herkömmlicher HPC Infrastruktur entschärft, gibt es auch hier einige Limitierungen:

- **Scale & Performance** - Die Größe aktueller kommerzieller *HPC as a Cloud* Systeme ist stark eingeschränkt (128 Prozessoren auf einer Amazon HPC Instanz gegenüber 294912 Prozessoren auf einem IBM Blue Gene/P Supercomputer). Auch die Performance speziell für HPC designer lokaler Cluster wird noch nicht erreicht.
- **Bereitstellung** - Der Aufbau einer virtuellen Schicht zur einfachen Konfiguration und Zugang zu HPC Ressourcen als Cloud ist anspruchsvoll. Die Schicht muss verschiedene HPC Ressourcen umschließen und dabei vollen Nutzen aus den spezifischen Ressourcen ziehen.
- **Flexibilität und on-demand Zugang** - In aktuellem HPC werden gewöhnlich batch-queue Systeme verwendet. Diese Systeme sind jedoch nicht on-demand zugänglich und schränken somit die Flexibilität stark ein.
- **Programmierbarkeit** - Web- bzw. Cloud-Anwendungen werden üblicherweise unabhängig von der zugrundeliegenden Hardware programmiert. Auf die Hardware zugeschnittene Programmierung kann die Performance erheblich steigern, beispielsweise indem mittels Datenlokalität die Kommunikationsansprüche gesenkt werden. Deshalb muss die Möglichkeit hardwareunabhängiger Programmierung ohne daraus resultierenden Performanceverlust weiter forciert werden.



### 3.3.4 Zusammenfassung der drei Haupt-Anwendungsmodi

	Vorteile	Nachteile	Typische Anwendungen	Beispiel-Systeme
<b>HPC in the Cloud</b>	Einfache Nutzung, unendlich Ressourcen	Niedrige Performance, teuer	E/P, L/S, A, HTC	Amazon EC2, Microsoft Azure
<b>HPC as a Cloud</b>	Flexibilität, on-demand Zugang	kleiner Umfang, QoS begrenzt, mittlere Performance	S, M, HPC, MTC	EC2CC, SGI, POD
<b>HPC plus Cloud</b>	Beschleunigung, Einsparungen, Belastbarkeit	unterschiedliche Kosten, Leistung, Verfügbarkeit von Ressourcen	HPC + Analyse/ Visualisierung	Comet-Cloud, InterGrid, GridWay

**Tabelle 2:** HPC Cloud Landschaft

**E/P** - Embarassingly Parallel

**S** - Synchron

**M** - Metaprobleme

**HTC** - High Throughput Computing

**L/S** - Lose Synchron

**A** - Asynchron

**HPC** - High Performance Computing

**MTC** - Many Task Computing

## 4 Benchmarks

In diesem Kapitel werden drei bekannte Cloud-Anbieter und ein Cluster in den Punkten Performance und Kosteneffizienz miteinander verglichen.

### 4.1 Einleitung

Aktuell haben Clouds das Potential eine Alternative zu traditionellen Cluster- und Gridsystemen darzustellen. Die Motivation eigene Implementierungen in die Cloud zu verlegen sind die niedrigeren Beschaffungs- und Wartungskosten sowie eine hohe Verfügbarkeit und eine sehr gute Skalierbarkeit. Durch die Kombination dieser Eigenschaften eignet sich die Cloud um HPC Anwendungen nicht nur im privaten Bereich einzusetzen. Das Kerngeschäft der Cloud-Anbieter ist jedoch im Normalfall das Anbieten von Content und nicht der HPC-Bereich.

Das Hauptaugenmerk in diesem Kapitel liegt im Vergleich von drei wichtigen HPC Aspekten, dem *Einsatz*, der *Performance* und der *Kosteneffizienz* bei Anwendung in der Cloud. Als Quelle wurde hauptsächlich das Paper [7] herangezogen, welches diese drei Punkte in Bezug auf die Cloudanbieter Amazon (Amazon Elastic Cloud Compute (EC2)), Microsoft (Microsoft Windows Azure) und Rackspace evaluiert. Die Benchmarkergebnisse werden anschließend mit einem selbst konfigurierten Cluster verglichen.

## 4.2 Setup

Um die jeweiligen Benchmarks durchführen zu können, werden Instanzen mit einem, zwei und vier Knoten benötigt. Die MPI Versionen des NPB Benchmarks werden auf der Instanz mit einem Knoten, die OpenMP Version auf Instanzen mit zwei und vier Knoten durchgeführt. In diesem Kapitel wird der Setup-Prozess einer Instanz, das Erstellen von Instanzen, die Startzeit und der Zugriff auf die Instanzen beschrieben.

### 4.2.1 Amazon EC2

Das Setup-Tool, welches von Amazon zur Konfiguration der EC2 Instanzen angeboten wird, besteht aus acht Schritten in welchen u.a. die Linux Kernel Version bestimmt wird. Eine Voraussetzung zur Erstellung einer Instanz ist es eine Security-Gruppe erstellt zu haben. Dieser Schritt ist allerdings nur bei Erstellung der ersten Instanz von Nöten. Zur Erleichterung trägt bei, dass es bei EC2 möglich ist mehrere Instanzen auf einmal zu erstellen. Diese Eigenschaft ist sehr komfortabel, da bei HPC Berechnungen meist mehrere gleichwertige Instanzen benötigt werden.

### 4.2.2 Microsoft Windows Azure

Um eine Instanz in Azure zu erstellen, sind vier Schritte zu erledigen. Um mehrere Instanzen zu verbinden ist es einmalig nötig eine Affinitätsgruppen zu erstellen. Es kann immer nur eine Instanz bearbeitet werden. Zum Löschen oder Erstellen einer Instanz ist ein exklusiver Zugriff auf die Instanz Voraussetzung. D.h. soll eine Instanz bearbeitet (gelöscht, erstellt, geändert) werden, muss die Bearbeitung der vorherigen Instanz abgeschlossen sein.

### 4.2.3 Rackspace

Die komfortabelste Methode um Instanzen zu erstellen bietet Rackspace an. Es muss nur das gewünschte Betriebssystem, der Name der Maschine und die Maschinengröße angegeben werden. Als Benutzer kann immer nur eine Instanz erstellt werden, vom Anbieter können diese jedoch parallel erstellt werden.

### 4.2.4 Bearbeitungszeiten

In Tabelle 3 werden Start- und Löschzeiten von einer und zwei Instanzen dargestellt. Eine möglichst kurze Lösch- und Erstellzeit ist erwünscht, da auch diese Zeit vom Benutzer bezahlt werden muss. Kurze Bearbeitungszeiten tragen also zu einer besseren Kosteneffizienz und höheren Flexibilität bei.

EC2 besitzt eine geringe Lösch- und Startzeit verglichen mit den anderen Anbietern. Azure bietet keine Möglichkeit Instanzen parallel zu löschen oder zu erstellen. Diese Eigenschaft erhöht die Bearbeitungszeiten von mehreren Instanzen dramatisch. Rackspace besitzt einen langsamen Startvorgang bei einer Instanz, wird jedoch mit wachsender Instanzenzahl nicht bedeutend langsamer. Des Weiteren besitzt Rackspace die kürzesten Löschzeiten.

Anbieter	Start 1 Instanz	Start 2 Instanzen	Löschen 1 Instanz	Löschen 2 Instanzen
EC2	2:34	2:34	0:33	0:33
Azure	3:06	12:43	3:07	12:26
Rackspace	12:37	12:57	0:23	0:23

**Tabelle 3:** Zeit (Min:Sec) zum Verwalten der Cloud Instanzen

## 4.3 Hardware und Software Konfigurationen

Ein wichtiger Punkt für die Entwicklung von HPC Anwendungen in der Cloud ist die Flexibilität der Hard- und Softwarekonfigurationen, welche der Provider anbietet. In diesem Kapitel werden die verfügbaren Hardwarekonfigurationen sowie die Anzahl der bereitgestellten Betriebssysteme beschrieben. In Tabelle 4 werden die Hardwareeigenschaften der Einzelmaschinen aufgelistet.

### 4.3.1 Amazon EC2

Amazon bietet elf verschiedene Konfigurationen für Einzelmaschinen mit einer großen Menge von Einstellungen an. Die leistungsschwächste Konfiguration ist

ein Single-Core mit 613 Mbyte Arbeitsspeicher. Für Anwendungen die mehr Leistung erfordern, werden drei speziell große Konfigurationen mit jeweils acht CPU-Kernen angeboten, mit dem Focus auf I/O Performance, hohen Speicherverbrauch und GPU Nutzung (GPGPU = General Purpose Computation on Graphics Processing Unit) angeboten. Als Betriebssysteme werden vier GNU/Linux und zwei Microsoft Windows Versionen zur Verfügung gestellt.

In EC2 ist es möglich die Instanzgröße zu verändern. Jedoch muss die Instanz zur Bearbeitung abgeschaltet, nach Wunsch verändert und wieder gestartet werden.

Als Virtualisierungssystem wird Xen von Amazon eingesetzt. Eine genaue Beschreibung der Hardware wird nicht angeboten, jedoch unterteilt Amazon die Rechenleistung der Hauptprozessoren in (virtuelle) Compute Units. Eine Compute Unit entspricht laut Amazon einem Intel Xeon mit einer Taktfrequenz von 1.0 bis 1.2 Ghz aus dem Jahr 2007 oder einem Xeon aus dem Jahr 2006 mit 1.6 Ghz Taktung.

#### 4.3.2 Microsoft Windows Azure

Microsoft verwendet für ihren IaaS (Infrastructure as a Service) als Virtualisierung Hyper-V. Azure bietet fünf Hardwarekonfigurationen für Einzelmaschinen an. Von einer leistungsschwächeren Single-Core Instanz mit 768 Mbyte Arbeitsspeicher bis zu einer leistungsstarken Instanz mit acht Kernen und 14 Gbyte RAM reicht das Angebot des Providers. Als Betriebssysteme werden fünf GNU/Linux und vier Windows Versionen angeboten. Bei einer Instanzänderung muss ähnlich der EC2 die Instanz abgeschaltet und wieder gestartet werden.

#### 4.3.3 Rackspace

Rackspace bietet acht Hardwarekonfigurationen für Einzelmaschinen an. Von einem Einzelkern bis Achtkern-Hauptprozessoren und von 256 Mbyte bis hin zu 30 Gbyte Arbeitsspeicher reicht die Spannweite an verfügbaren Konfigurationen von Instanzen. Zur Virtualisierung verwendet Rackspace Openstack. Angeboten werden 20 GNU/Linux und neun Microsoft Windows Versionen als Betriebssystem. Im Gegensatz zu Amazon EC2 und Microsoft Windows Azure können bei Rackspace Instanzen nicht geändert werden. Wird eine Änderung der Instanzen nötig, muss diese gelöscht und neu erstellt werden.

### 4.4 Performance Analyse

Zum Vergleich der Performance der drei Anbieter im HPC Bereich werden Ein-, Zwei- sowie Vier-Knoten-Instanzen verwendet. Die Intra-Knoten-Kommunikation (ein Knoten) wird mit einer OpenMP (Open Multi-Processing) Version, die Inter-Knoten-Kommunikation (zwei und vier Knoten) wird mit einer MPI (Message Passing Interface) Version der NPB Benchmarks getestet.

Eigenschaft	EC2	Azure	Rackspace
Anzahl verschiedener Hardware Konfigurationen	11	5	8
Möglichkeit der Änderung der Instanzgröße	Ja	Ja	Nein
Min/Max Arbeitsspeichergröße (GByte)	0.613-68	0.768-14	0.256-30
Min/Max Anzahl an CPU Kernen		1 bis 8	
Unterstützte Betriebssysteme	GNU/Linux, Windows		

Tabelle 4: Eigenschaften einer Einzelmaschinen

#### 4.4.1 NAS Parallel Benchmarks (NPB)

Der NAS Parallel Benchmark [1] ist eine Sammlung kleiner Programme, welche helfen die Leistungsfähigkeit von Supercomputern zu evaluieren. Die Benchmarks lehnen sich an Anwendungen im CFD (Computational Fluid Dynamics) Bereich an.

Folgende Benchmarks wurden zum Evaluieren der Clouds benutzt [7]:

1. **IS** - Integer Sort, random memory access
2. **EP** - Embarrassingly Parallel
3. **CG** - Conjugate Gradient, irregular memory access and communication
4. **MG** - Multi-Grid on a sequence of meshes, long- and short-distance communication, memory intensive
5. **FT** - Discrete 3D fast Fourier Transform, all-to-all communication
6. **BT** - Block Tri-diagonal solver
7. **SP** - Scalar Penta-diagonal solver
8. **LU** - Lower-Upper Gauss-Seidel solver
9. **UA** - Unstructured Adaptive mesh, dynamic and irregular memory access
10. **DT** - Data Traffic

Des Weiteren ist es nötig, je nach Größe der zu testenden Maschine, die Problemgröße des Benchmarks einzustellen. Zum Evaluieren der Cloudmaschinen wurde in [7] Problemgröße B eingesetzt, da diese komplett in den Arbeitsspeicher der zu testenden Knoten passt.

### 4.4.2 Benchmarkmaschinen

Die IaaS Anbieter mussten im Paper [7] die Benchmarks mit folgenden Spezifikationen durchlaufen und werden mit einem entsprechenden Cluster verglichen.

#### Amazon EC2

Ausgewählt wurde die High-CPU Extra Large On-Demand Konfiguration, welche zu einem Preis von 0.66\$ pro Stunde angeboten wurde. Als aktueller Preis am 13.Juni 2013 wird auf der Amazon Webseite [2] 0.58\$ pro Stunde angegeben.

#### Microsoft Azure

Die Extra-large Compute Konfiguration von Azure mit einem Preis von 0.64\$ pro Stunde, welcher auch dem aktuellen Preis am 13.Juni 2013 entspricht [4].

#### Rackspace

Eingesetzt wurde die größte verfügbare Konfiguration mit einem Preis von 1.8\$ pro Stunde. Der aktuelle Preis einer entsprechenden Instanz entspricht am 13.Juni 1.20\$ (unmanaged Cloud) bzw. 1.32\$ (managed Cloud) und kann auf [6] eingesehen werden.

Eine Übersicht der Maschinen kann in Tabelle 5 eingesehen werden.

Parameter	EC2	Azure	Rackspace	Cluster
Instanzname	High-CPU Extra Large	Extra Large VM	-	-
CPU Typ	2006/07 Intel Xeon	-	-	Intel Xeon E5530
CPU Takt (GHz)	~2.5 - 3.0	1.6	2.4	2.33
CPU Kerne	8	8	8	8
RAM/Maschine (GByte)	7	14	30	12

**Tabelle 5:** Eigenschaften der Benchmarkmaschinen

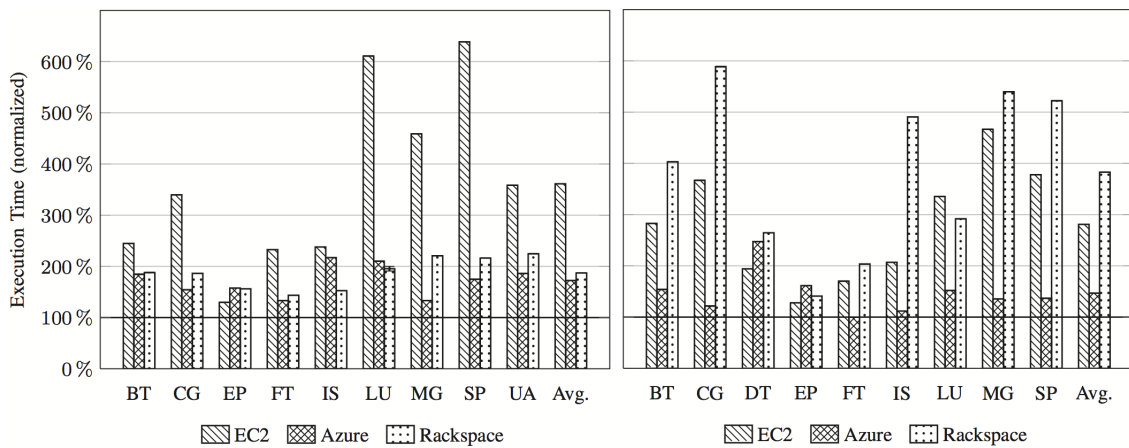
### 4.4.3 NAS Parallel Benchmarks - Ergebnisse

Die Benchmarks wurden von den Ein-, Zwei- und Vier-Knoten-Systemen der drei Anbieter durchlaufen. Im linken Bild in Abbildung 2 kann die Performance der Ein-Knoten-Systeme betrachtet werden. Das Ergebnis dieses Tests stellt sich wie folgt dar: Azure und Rackspace erreichen vergleichbar gute Ergebnisse mit einem kleinen Vorteil für die Microsoft Cloud. Etwas überraschend ist, dass EC2, des in diesem Vergleich die schlechtesten Ergebnisse erreicht, im EP Benchmark Spitzenwerte liefert. Das lässt sich jedoch mit der CPU Abhängigkeit des EP-Benchmarks

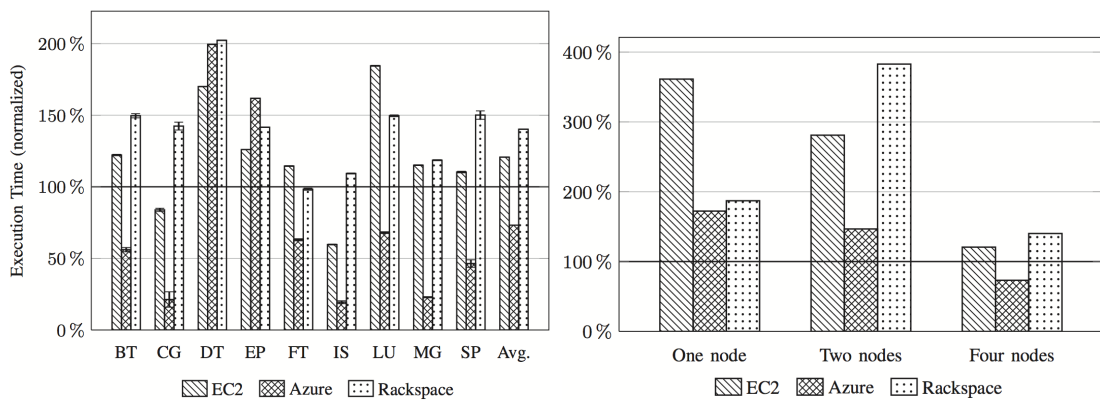
und der leistungsfähigen CPU im EC2 Knoten erklären. Das Clustersystem erreicht jedoch keiner der drei Cloud-Anbieter.

Ein klareres Ergebnis ergibt sich bei Betrachtung des rechten Bildes in Abbildung 2. In der Open MP Version des NPB stellt Microsoft den Benchmarksieger bei den Zwei-Knoten-Systemen. Das Clustersystem mit zwei Knoten wird nur von einem der Cloud-Anbieter in manchen Benchmarks annähernd erreicht.

Im linken Bild in Abbildung 3 werden die Ergebnisse der Vier-Knoten-Systeme gezeigt. Microsoft Azure gewinnt wie im Zwei-Knoten-Benchmark. Azure erreicht in der Vierknotenkonfiguration eine höhere Performance als der Vier-Knoten-Cluster.



**Abbildung 2:** Links: OpenMP bei Benutzung eines Knotens  
Rechts: MPI bei Benutzung von zwei Knoten



**Abbildung 3:** Links: MPI bei Benutzung von vier Knoten  
Rechts: Durchschnittsergebnisse

## 4.5 Kosteneffizienz

In diesem Kapitel werden Kosten und Performance der Cloudanbieter sowie des Clusters in Relation gestellt und somit die Kosteneffizienz betrachtet.

### 4.5.1 Systemkosten

Um die Systemkosten des Clusters bestimmen zu können, werden die Kosten pro Stunde bei einer Nutzungsdauer von drei Jahren berechnet. Als Anschaffungskosten der Clustermaschine werden 12.000\$ angenommen. Berechnet werden die Kosten pro Stunde unter Berücksichtigung der anfallenden Stromkosten, welche beim Ablauf der Benchmarks gemessen wurden, unter Annahme einer Laufzeit von 24 Stunden pro Tag.

Berechnung der Kosten pro Stunde ohne anfallende Stromkosten aus [7]:

$$\frac{\text{Anschaffungskosten}}{\text{Jahre} * \text{Tage} * \text{Stunden}} = \text{Kosten (\$ pro Stunde)}$$

Wird die Formel auf die gegebenen Daten angewandt, ergibt sich folgendes:

$$\frac{12000}{3 * 365 * 24} = 0.457\$/h$$

Die Kosten pro Stunde des Clusters betragen ohne Stromkosten bei einer Laufzeit von 24 Stunden auf drei Jahre 0.457 \$ pro Stunde.

Die anfallenden Stromkosten müssen jedoch miteinbezogen werden. Bei Durchlauf der Benchmarks wurden durchschnittlich 275W Stromverbrauch gemessen. Bei einem durchschnittlichen Strompreis von 0.0991\$ für eine kW/h in den USA im Jahre 2010 können nachfolgende Berechnungen erstellt werden:

$$\text{Stromverbrauch} * \text{Strompreis} = \text{Stromkosten (\$ pro Stunde)}$$

Wird obige Formel angewandt,

$$275W * 0.0991\$ = 0.0273\$/h,$$



ergeben sich Gesamtkosten von 0.484\$ pro Stunde bei Benutzung des Clusters.

Die Ergebnisse im Vergleich mit den Cloudmaschinen können in Tabelle 6 betrachtet werden. Auffallend ist, dass keiner der drei Anbieter mit einem kleineren Preis pro Stunde als der Cluster aufwarten kann. Bei Evaluierung der Ergebnisse muss jedoch in Betracht gezogen werden, dass die Clusterkosten pro Stunde nur bei einer Nutzung von 24 Stunden pro Tag über drei Jahre berechnet wurden. Die niedrigsten Kosten pro Stunde im Cloudbereich erreicht die Microsoft Azure dicht gefolgt von Amazons EC2.

Als Schluss kann gefolgert werden, dass die Cloudanbieter bei einer Nutzung welche nicht 24 Stunden 7 Tage die Woche beträgt beim Faktor Kosteneffizienz durchaus als eine sehr gute Alternative zu einem Cluster betrachtet werden können.

Parameter	EC2	Azure	Rackspace	Cluster
Cloudkosten	0.66	0.64	1.80	-
Hardwarekosten	-	-	-	0.475
Stromkosten	-	-	-	0.0273
Kosten pro Node	0.66	0.64	1.80	0.484
Kosten für zwei Nodes	1.32	1.28	3.60	0.968
Kosten für vier Nodes	2.64	2.56	7.20	1.936

**Tabelle 6:** Kosten in USD pro Stunde

#### 4.5.2 Kosteneffizienz und Break Even Point - Berechnung

Da sich die Preise und die Performance der Cloud-Anbieter unterscheiden, wäre es nicht sinnvoll Kosten und Performance getrennt zu betrachten.

Als Kosteneffizienz wird in *"High Performance Computing in the Cloud: Deployment, Performance and Cost Efficiency"* [7] die Skalierung von Performance mit dem Preis pro Stunde festgelegt. Dies ergibt einen fairen Vergleich.

Um die Kosteneffizienz bestimmen zu können wurden in [7] zwei Metriken - Kosteneffizienz und Break Even Point - eingeführt.

##### 1. Kosteneffizienz

Die Metrik Kosteneffizienz gibt an, wieviel die jeweilige Maschine pro Stunde kosten würde, um die selben Benchmarks zu bestreiten. Das Ergebnis wird in USD pro Stunde angegeben.

Berechnet wird die Kosteneffizienz wie folgt:

$$\text{Kosteneffizienz} = \text{durchschnittliche Performance} * \text{Kosten pro Stunde}$$

## 2. Break Even Point

Der Break Even Point gibt an, ab welchem Tag es kosteneffizienter ist den Cluster anstelle einer der Cloudmaschinen zu benutzen.

Definiert wird der Break Even Point folgend:

$$\text{Break Even Point} = \frac{\text{Jahreskosten}}{24 * \text{Kosteneffizienz}}$$

Parameter		EC2	Azure	Rackspace	Cluster
Ausführungszeit der Benchmarks (in %)	1 Knoten	361.3	172.3	187.1	100.0
	2 Knoten	281.0	146.7	382.8	100.0
	4 Knoten	120.7	73.2	140.2	100.0
Kosteneffizienz (in \$ pro Stunde)	1 Knoten	2.39	1.10	3.37	0.484
	2 Knoten	3.71	1.89	13.78	0.968
	4 Knoten	3.19	1.14	10.09	1.936
Break Even Point (in Tagen)	1 Knoten	74	161	53	-
	2 Knoten	96	187	26	-
	4 Knoten	222	620	71	-

**Tabelle 7:** Kosteneffizienz und Break Even

### 4.5.3 Erkenntnisse

Die gewonnenen Ergebnisse aus [7] können in Tabelle 7 auf Seite 16 betrachtet werden. Im Falle der Kostenffizienz konnte nur einer der Cloudanbieter dem Cluster die Stirn bieten. Einzig die Microsoft Azure in ihrer Vierkernkonfiguration ist kosteneffizienter als der Cluster und gewinnt mit 1,14\$/h gegen den Cluster mit 1,936\$/h. In allen anderen Konfigurationen sind die Cloudanbieter kostenineffizienter als der Cluster im HPC Bereich.

Mit in Betracht gezogen werden muss jedoch die Tatsache, dass zur Berechnung der Clusterkosten eine Clusterlaufzeit von drei Jahren angenommen wird.

Auch im Bereich des Break Even Point stellt sich die MS Azure in ihrer Vierknotenkonfiguration als herausragend dar. Ganze 620 volle Tage Rechenzeit des Clusters wären nötig, um die MS Azure zu erreichen. Mit 26 Tagen den schnellsten Break Even Point erreicht Rackspace in ihrer Zweiknotenkonfiguration.

Am Ende des Kapitels kann das Fazit gezogen werden, dass Clouds im Moment teilweise kosteneffizienter als Cluster sein können. Dies zeigt auch das Ergebnis der Microsoft Azure im Paper [7]. In Zukunft kann jedoch erwartet werden, dass auch andere Cloudanbieter vergleichbar gute Performance zu einem erschwinglichen Preis anbieten werden und sich so als eine ernsthafte Alternative zum herkömmlichen Cluster erweisen werden.

## 5 Zusammenfassung

Das Ziel dieses Papers war es verschiedene HPC Infrastrukturen gegenüberzustellen und voneinander abzugrenzen. Insbesondere die Vor- und Nachteile der Cloud gegenüber anderen Infrastrukturen bei HPC wurden herausgearbeitet.

Möglichkeiten, Nutzen, aber auch Limitierungen der Cloud im HPC-Bereich zum gegenwärtigen Zeitpunkt sind aufgezeigt worden und die derzeit verbreitetsten Clouds untereinander und mit einem Cluster hinsichtlich Kosten und Leistung wurden verglichen. Während die Cloud viele Services (Contentbereich, Client/Server Webanwendungen, Backup und Filesharing) bereits nachhaltig verändert hat, ist der Durchbruch im HPC Bereich noch nicht in vollem Umfang vollzogen. Zum gegenwärtigen Zeitpunkt sind nicht alle der gängigen Clouds in der Lage einen Großteil der HPC-Anwendungen mit zufriedenstellender Performance / Kostenefizienz umzusetzen. Eine effektive Nutzung der Cloud-Ressourcen für HPC bedarf mitunter alternativer Applikationsformulierungen. Es liegt viel ungenutztes Potential im HPC Sektor bei Cloudnutzung brach und es Bedarf weiterer Forschung sowie eine

Verbesserung des Cloud-Modells. In naher Zukunft werden sich die Clouds jedoch zu einer ernsthaften Alternative zu Clustersystemen im HPC Bereich entwickeln.

## 6 Referenzen

- [1] NAS Parallel Benchmarks. <http://www.nas.nasa.gov/publications/npb.html>, 2013. [Online; abrufbar am 13-Juni-2013].
- [2] Amazon. Amazon EC2 On-Demand-Instances Prices. <http://aws.amazon.com/de/ec2/>, 2013. [Online; abrufbar am 13-Juni-2013].
- [3] Gandotra Indu, Abrol Pawanesh, Gupta Pooja, Uppal Rohit, and Singh Sandeep. Cloud computing over cluster, grid computing: a comparative analysis. *Journal of Grid and Distributed Computing*, 1:01–04, 2010.
- [4] Microsoft. Microsoft Azure Preisdetails. <http://www.windowsazure.com/de-de/pricing/details/cloud-services/>, 2013. [Online; abrufbar am 13-Juni-2013].
- [5] Manish Parashar, Moustafa AbdelBaky, Ivan Roderer, and Aditya Devarakonda. Cloud paradigms and practices for cds&e. *Rep. Print*, 2012.
- [6] Rackspace. Cloud Servers Pricing. <http://www.rackspace.com/cloud/servers/pricing/>, 2013. [Online; abrufbar am 13-Juni-2013].
- [7] Eduardo Roloff, Matthias Diener, Alexandre Carissimi, and Philippe OA Navaux. High performance computing in the cloud: Deployment, performance and cost efficiency. In *Cloud Computing Technology and Science (CloudCom), 2012 IEEE 4th International Conference on*, pages 371–378. IEEE, 2012.
- [8] Luis M Vaquero, Luis Roderer-Merino, Juan Caceres, and Maik Lindner. A break in the clouds: towards a cloud definition. *ACM SIGCOMM Computer Communication Review*, 39(1):50–55, 2008.