

SE aus Informatik: High Performance Computing in the Cloud

Erich Mraz,
Maximilian Schorr

8. Juni 2013

HPC Infrastruktur

Cluster Computing

- paralleles und verteiltes System
- Ansammlung eng verbundener Einzelrechner
 - homogene Ressourcen (Hardware und OS)
 - gesammelt an einem Standort
 - über LAN verbunden
- arbeiten zusammen als einheitlich integriertes System
 - Single System Image
 - zentralisiertes Job Management & Scheduling System
- hohe Performance zu geringem Preis und gute Skalierbarkeit

Grid Computing

- paralleles und verteiltes System
- Kombination von Rechenressourcen verschiedener administrativer Domains
 - lose verbunden (Dezentralisation)
 - heterogene Ressourcen
 - haben nicht den selben Standort
 - verbunden über LAN, MAN oder WAN
- autonome Knoten
 - eigener Resource Manager
 - unabhängige Einheit
 - Diversity und Dynamism
 - verteiltes Job Management & Scheduling
- Abrechnung von fixem Betrag pro Service oder verschiedene Organisationen teilen sich untaugige Ressourcen

Teilen, Auswählen und Zusammenlegen (von Ressourcen) dynamisch zur Laufzeit abhängig von ihrer Verfügbarkeit, Leistungsfähigkeiten, Performance, Kosten und QoS Ansprüchen

Cloud Computing

- paralleles und verteiltes System
- Ansammlung zusammenhängender Computer
 - generell homogene Ressourcen (besseres Hardware Management)
 - Daten liegen auf zentralem Server (besseres Daten Management)
- großer Pool von **leicht benutzbaren** und on-demand zugänglichen **virtualisierten** Ressourcen
 - dynamisch bereitgestellt, dargestellt als eine oder mehrere vereinte Rechenressourcen
 - on-demand Bereitstellung von Applikationen
 - **skalieren** entsprechend variabler Last und erlauben optimale Ausnutzung von Ressourcen
- **pay-per-use** Modell
- IT-Kapazität erhöhen (oder Leistungsfähigkeiten ausbauen) "on the fly" und in Echtzeit
- kosteneffektiv

Bereitstellen, Konfigurieren und Verwalten von Infrastruktur und es Forschern ermöglichen sich auf die Wissenschaft zu fokussieren und den Aufbau von Software und Experimenten mit Kollegen zu teilen

HPC & Cloud

HPC Applikationen

- heterogene und dynamische Applikation Workflows
- eng verbunden
- großer Aufwand an Interprozessor Kommunikation
- schnelle Verbindungen zwischen Prozessoren
- benötigen idR große Mengen Rechenpower für kurze Zeitspanne
- large scale

HPC & Cloud

- Cloud Computing Services
 - stellen Plattform für Applikationen bereit, wenn lokale Infrastruktur nicht verfügbar ist
 - ergänzen existierende Plattformen mit zusätzlichen Kapazitäten oder Leistungsfähigkeiten
 - verbessern Usability und Produktivität von traditionellem HPC
- Haupt-Anwendungsmodi
 - ① **HPC in the Cloud** - gesamte Applikation auslagern
 - ② **HPC plus Cloud** - HPC/Grid Ressourcen mit Cloud Services komplementieren
 - ③ **HPC as a Cloud** - HPC/Grid Ressourcen als Cloud zur Verfügung stellen
 - elastische on-demand Cloud Abstraktionen verwenden
 - Flexibilität von Cloud-Modellen mit der Performance von HPC-Systemen verbinden



Abbildung: Hybrid HPC/Grid + Cloud usage modes

Anwendungsmodi & Herausforderungen

HPC in the Cloud

- nur effektiv bei bestimmten Arten von HPC-Applikationen
 - hohe Parallelität (MapReduce)
 - minimale Synchronisation
 - minimale Interprozess-Kommunikation
 - geringe I/O Ansprüche
 - modest scale
- Herausforderungen
 - mangelnde Leistungsfähigkeiten und Homogenität der Cloud-Hardware
 - fehlende High-Speed Netzwerkverbindung
 - hohe Latenzen unterbinden Vorteile in Skalierbarkeit, Flexibilität und Fehler-Toleranz
 - alternative Code-Formulierungen für Applikationen

HPC plus Cloud

- Cloud bursting
 - Applikation läuft auf traditionellen HPC-Ressourcen
 - "bursting" auf die Cloud bei
 - plötzlich eintretendem Bedarf nach Rechenkapazität
 - Bedarf nach spezieller Leistungsfähigkeit, um heterogene Applikations-Ansprüche zu erfüllen
 - neue und effektivere Nutzungsarten
- Herausforderungen
 - Heterogenität und Dynamik
 - Ressourcen mit unterschiedlicher Konfiguration, Performance, Verfügbarkeit, Laufzeitverhalten, Stabilität, Kosten
 - dynamische Anforderungen und Beschränkungen auf Applikationsseite
 - Bereitstellung und Scheduling geeigneter Ressourcen hinsichtlich Kosten/Leistung
 - autonomes Monitoring und Management von dynamischem Verhalten und Anforderungen

HPC as a Cloud

- HPC Ressourcen werden mittels Cloud Computing-Abstraktionen als on-demand Ressourcen angeboten
- speziell für HPC Applikationen
 - Hardware mit schnelleren Prozessoren und Netzwerkverbindungen
 - nicht-virtualisierte Hardware
- zwei Hauptansätze

Large HPC systems	Small HPC clusters connected together	
Physical Solutions	Physical Solutions	Virtualized Solutions
E.g., Blue Gene/P as a Service	E.g., Penguin on Demand, Silicon Graphics, HPCynergy	E.g., EC2 Compute Cluster, Azure, Adaptive Systems, CloudCycle

Tabelle: Cloud Paradigms and Practices for CDS&E, Table 1: HPC as a Cloud classification

- Herausforderungen
 - **Scale & Performance**
 - **Bereitstellung** - Aufbau einer virtuellen Schicht zwecks einfacher Konfiguration und Zugang zu HPC Ressourcen als Cloud
 - **Flexibilität und on-demand Zugang** aufgesetzt auf batch-queue Systeme
 - **Programmierbarkeit** unabhängig von der darunterliegenden Hardware

Summary of the key usage modes

	Advantages	Disadvantages	Typical applications	Sample systems
HPC in the Cloud	Easy to use, infinite resources	Low performance, expensive	E/P, L/S, A, HTC	Amazon EC2, Microsoft Azure
HPC as a Cloud	Elasticity, on demand access	Small scale, limited QoS, medium performance	S, M, HPC, MTC	EC2CC, SGI, POD
HPC plus Cloud	Acceleration, conservation, resilience	Different prices, performance, availability of resources	HPC + Analytics/Visualization	CometCloud, InterGrid, GridWay

Tabelle: Cloud Paradigms and Practices for CDS&E, Table 2: HPC Cloud landscape

Public - Cloud - Vergleiche

Informationen

Wichtige Fragen:

Was wird bereitgestellt?

Wie ist die Performance?

Kosteneffizienz?

Welche Public-Clouds:

Amazon Elastic Cloud Compute (EC2)

Microsoft Windows Azure

Rackspace

Hardware und Softwarekonfigurationen

- **Amazon EC2:**

- **Ram:** 613MByte bis 68GByte
- **Konfigurationen:** drei große mit jeweils 8 CPU Kernen

- **Betriebssysteme:**
 - ① high I/O performance
 - ② high memory size
 - ③ high CPU performance

- **Clustermöglichkeit:**
 - ① 5 oder 8 Knoten
 - ② Spezieller Kluster für hohe I/O performance
 - ③ GPU Cluster

- **CPU:** eingeteilt in Compute Units
 - 2007 Intel Xeon 1.0 - 1.2Ghz
 - 2006 Intel Xeon 1.6Ghz

Hardware and Software Configurations

- Microsoft Windows Azure:

- Ram: 768MByte bis 14GByte
- Konfigurationen: 5
 - ① 1Ghz single Core (Quelle: Azure)
 - ② 1.6Ghz 8 Core (Quelle: Azure)
- Betriebssysteme:
 - ① 4 GNU/Linux Versionen
 - ② 5 Windows Versionen

- Rackspace:

- Ram: 256MByte bis 30GByte
- Konfigurationen: 8
 - Single virtual Core
 - 8 virtual Cores
- Betriebssysteme:
 - ① 20 GNU/Linux Versionen
 - ② 9 Windows Versionen

Instanzen - Setup

① Amazon EC2

- Tool mit acht Schritten
 - Linux Kernel Version
 - Windows
- Mehrere Instanzen gleichzeitig erstellbar

② Windows Azure

- Vier Schritte nötig
- Instanzen müssen einzeln erstellt werden

③ Rackspace

- Drei Schritte
 - ① Betriebssystem
 - ② Name des Systems
 - ③ Größe der Maschine
- Instanzen müssen einzeln erstellt werden

Benchmarks

Benchmarks - NPB

- Numerical Aerodynamic Simulation Parallel Benchmarks (NPB)
 - Benchmark um hochparallele Supercomputer zu vergleichen
 - NASA Advanced Supercomputing (NAS) Division
- OpenMP und MPI Implementierungen
- Benchmarksammlung zum Testen von:
 - ① Floating point performance:
 - BT, EP, SP
 - ② Integer performance:
 - IS
 - ③ Communication performance:
 - CG, FT, LU, MG, UA
 - ④ Data Movement performance:
 - DT
- Programmiersprachen: C und Fortran

Benchmarks - eingesetzte Hardware

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: Eigenschaften der Benchmarkmaschinen

Benchmarkergebnisse - NPB

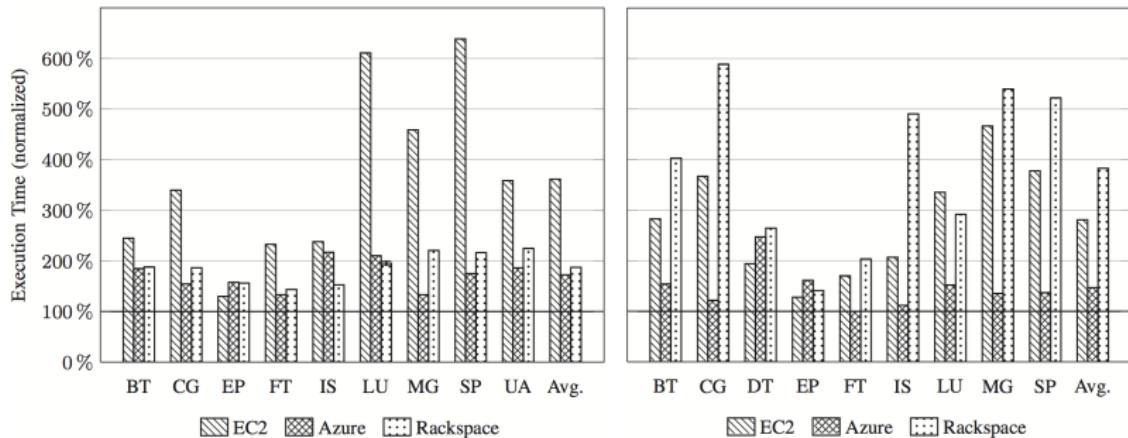


Abbildung: Links: OpenMP bei Benutzung eines Knotens
Rechts: MPI bei Benutzung von zwei Knoten

CPU: Floating Point: BT, EP, SP - Integer: IS
Communication: CG, FT, LU, MG, UA - Data Movement: DT

Benchmarkergebnisse - NPB & Durchschnittsergebnisse

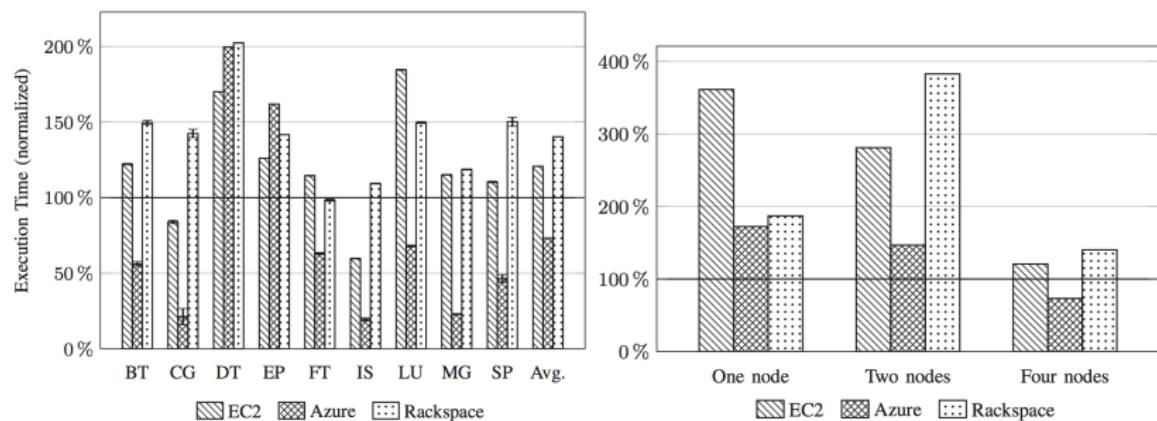


Abbildung: Links: MPI bei Benutzung von vier Knoten
Rechts: Durchschnittsergebnisse

CPU: Floating Point: BT, EP, SP - **Integer:** IS
Communication: CG, FT, LU, MG, UA - **Data Movement:** DT

Benchmarks - Erkenntnisse

Ein Knoten:

Cluster war schnellstes System

Zwei Knoten:

Cluster war schnellstes System
Azure erreicht fast Clusterergebnisse

Vier Knoten:

Azure ist schneller als das Clustersystem
Aufgrund von schnellem Netzwerktransfer?

⇒ **Cloud bringt Vorteile bei großem Rechenaufwand / hoher Parallelität**

Kosteneffizienz

Kosten - Cluster

Clusteranschaffungskosten: **12.000\$**

Kosten pro Stunde bei einer Nutzungsdauer von drei Jahren:

$$\frac{12.000\$}{3 \cdot 365 \cdot 24} = \mathbf{0.457\$/h}$$

Angenommen der Cluster verbraucht bei Vollast 275W

Preis für eine Kilowattstunde: 0.0991\$

$$\Rightarrow 275W \cdot 0.0991\$ = \mathbf{0.0273\$/h}$$

Kosten pro Stunde für den Cluster: 0.484\$/h

Kostenvergleich - Kosten pro Stunde

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: Kosten in USD pro Stunde

⇒ **Cluster (nur) bei hoher Nutzungsdauer im Kostenvorteil**

Kostenvergleich - Kosteneffizienz und Break Even Point

Kosteneffizienz

Kosten pro Stunde bei gleicher Rechenarbeit

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

Break Even Point

Tag an dem die Cloudkosten gleich dem Anschaffungspreis der Cluster sind

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

Clusterkosten: 4239.8\$ pro Jahr

Kostenvergleich - Kosteneffizienz und Break Even Point

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 Konten	74	161	53	-
	2 Knoten	96	187	26	-
	4 Knoten	222	620	71	-

Tabelle: Kosteneffizienz und Break Even

Kosten & Effizienz - Erkenntnisse

Kosteneffizienz

- Cluster meist effizienter als Cloud
- 1 Knoten zu 2 Knoten: kaum Effizienzänderung
- 4 Knoten \Rightarrow Clouds erreichen bessere Kosteneffizienz
- 4 Knoten: Azure ist 41% kosteneffizienter als der Cluster d.h.:

Azure kostet 41% weniger um die gleiche Arbeit zu erledigen als der Cluster

Break Even Point

- **Kleinster Break Even:** Rackspace mit zwei Knoten
26 Tage Break Even, jedoch nur bei vollen 26 Tagen Rechenzeit
- **Größter Break Even:** Azure mit vier Knoten \Rightarrow 620 Tage