

PG Intercloud

Projektgruppe des Lehrstuhls für Programmiersysteme (LS 5)

- 1. Thema** Entwurf, Entwicklung und Evaluierung einer Abstraktionsschicht, welche Angebote verschiedener Cloud-Anbieter vereint
- 2. Zeitraum** Wintersemester 2012/2013 und Sommersemester 2013
- 3. Umfang** Jeweils 8 Semesterwochenstunden
- 4. Veranstalter** Markus Doedt, OH14, R.130, Tel. 7756, markus.doedt@tu-dortmund.de
Maik Merten, OH14, R.128, Tel. 7759, maik.merten@tu-dortmund.de

5. Aufgabe

Die Projektgruppe hat zum Ziel, Strategien zu entwickeln und zu evaluieren, die dazu dienen, ähnliche Angebote verschiedener Cloud-Anbieter austauschbar nutzbar zu machen. Wie ist es möglich, eine Zwischenschicht über die sehr unterschiedlichen Technologien und Schnittstellen der jeweiligen Anbieter zu etablieren, die eine gleichzeitige Nutzung bzw. eine schnelle Portierung zwischen verschiedener Cloud-Plattformen ermöglicht? Untersuchen soll dies die PG anhand der Entwicklung einer webbasierten Gruppenverwaltung, mit der Musikbands in Kontakt zu ihren Fans treten können.

5.1 Motivation

Immer mehr Anwendungen werden in Form von Webapplikationen entwickelt und sind einem wachsenden Anwenderkreis zugänglich. Applikationen, die vormals bevorzugt auf Desktopcomputern betrieben wurden sind nun als Webanwendungen zugänglich und sind bestrebt, traditionelle Desktopanwendungen zu ersetzen. Bekannte Beispiele hierfür sind Google Docs oder Google Mail. Diese bekannten Anwendungen müssen die Arbeitslast von Millionen von Benutzern bewältigen können, auch zu Spitzenzeiten. Dementsprechend betreiben große Anbieter wie Amazon, Google oder Microsoft sehr große Rechenzentren an einer Vielzahl von Standorten.

Demgegenüber nimmt sich übliche Ausstattung an Rechenmitteln bei kleinen oder mittleren Unternehmen deutlich bescheidender aus. Um eine neue Webanwendung im Markt platzieren zu können, müssen üblicherweise neue Ressourcen beschafft werden. Wächst die Nutzeranzahl rasant, so müssen diese Ressourcen ebenso ausgebaut werden, was hohe Investitionen und Verzögerungen nach sich zieht und weiterhin bedeutsame Nachfolgekosten verursacht. Dies stellt Unternehmen vor große Herausforderungen, die nicht immer mit den zur Verfügung stehen Mitteln zu bewältigen sind. Dies kann die Marktchancen der entwickelten Lösungen erheblich beeinträchtigen.

Cloud-Computing [1, 2] sieht nach einer vielversprechenden Lösung aus, um kosteneffizient Ressourcen von Drittanbietern einzukaufen. Durch Cloud-Computing kann es vermieden werden, eigene Rechenzentren auszustatten, zu betreiben und zu erweitern. Stattdessen werden die benötigten Ressourcen, wie beispielsweise Rechenzeit und Speicherplatz, von den großen Cloud-Anbietern auf Knopfdruck bereitgestellt und nach Nutzung abgerechnet. Hierzu werden Schnittstellen angeboten, um die eigene Anwendung zur Nutzung dieser Ressourcen vorzubereiten.

Diese Schnittstellen sind bisher nicht normiert und jeder Cloud-Anbieter kocht hier sein eigenes Süppchen. Dementsprechend müssen Anwendungen für jede Cloud-Plattform einzeln vorbereitet werden, was aufwendig und somit kostenintensiv sein kann.

In dieser Projektgruppe soll erarbeitet werden, inwieweit heutige Cloud-Angebote eingesetzt werden können, ohne sich in komplette Abhängigkeit von einzelnen Cloud-Anbietern zu begeben.

5.2 Cloud-Technologien

Cloud-Technologien lassen sich im Wesentlichen in drei Kategorien einteilen: SaaS (Software as a Service), IaaS (Infrastructure as a Service) und PaaS (Plattform as a Service). Bei SaaS handelt es sich häufig um nichts anderes als um Webapplikationen, die durch den Einsatz von AJAX-Technologien sehr ähnlich zu Desktopapplikationen sind. Ein gutes Beispiel hierfür ist zum Beispiel Google Calendar, welches in Konkurrenz zu Desktopanwendungen mit Kalenderfunktion (beispielsweise Microsoft Outlook) tritt. SaaS-Angebote sind jedoch auch häufig mit Schnittstellen ausgestattet, um in andere Dienste integriert werden zu können.

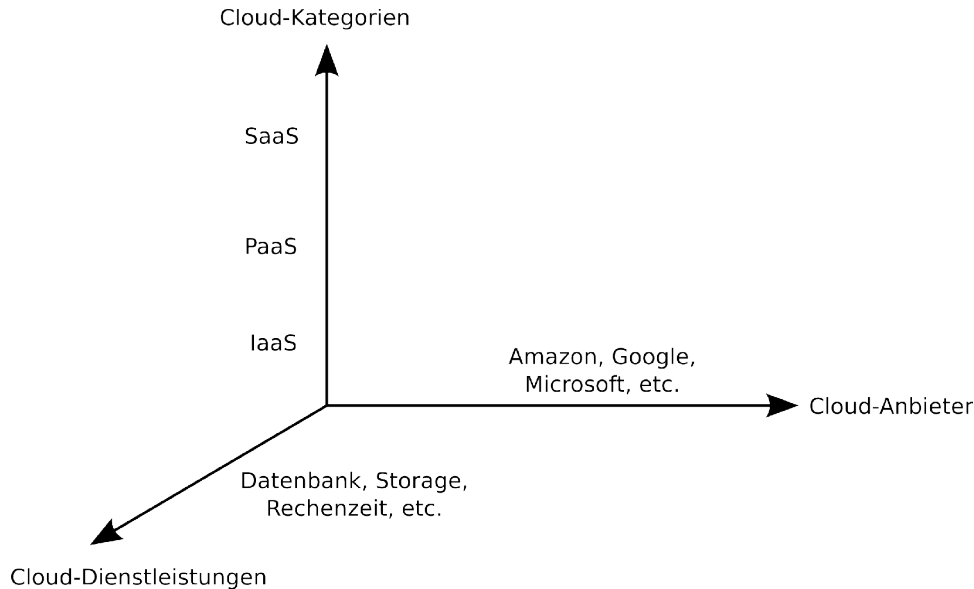


Abbildung 1: Drei wählbare Dimensionen beim Cloud-Computing

Eine weitere Kategorie bei Cloud-Diensten ist IaaS. Hier werden spezielle Infrastrukturdienste im Web angeboten, wie zum Beispiel Amazon S3 [3] zur Datenspeicherung oder Amazon EC2 [4] zur Nutzung von Rechenleistung. Auch einfache Dienste, wie z.B. Dropbox, kann man dieser Kategorie zuordnen.

Eine zwischen IaaS und SaaS einzuordnende Cloud-Kategorie ist PaaS, wie es zum Beispiel von Google mit der App Engine [5] angeboten wird. Hier stehen komplette Plattformen zur Verfügung, auf die man einfach seine eigenen Applikationen deployen kann, ohne sich über die dahinterliegende, konkrete technische Infrastruktur Gedanken zu machen.

5.3 Probleme bei der Portierung von Anwendungen

Wie bereits angesprochen bieten die großen Cloud-Anbieter im Kern ähnliche Lösungen an, die sich jedoch in technischen Details, Schnittstellen und Abrechnungsverfahren unterscheiden. Die folgende Aufstellung stellt beispiel- und lückenhaft eine kleine Auswahl ähnlicher Angebote der großen Anbieter Amazon und Google gegenüber.

	amazon.com®	Google™
Storage	S3	Google Cloud Storage
NoSQL Datenbank	DynamoDB	Google Big Table
Anwendungshosting	AWS Elastic Beantalk	Google App Engine

Hierbei stellen die einzelnen Dienste folgende Funktionen bereit:

- Storage: Ablegen von Dateien, Replizierung der Daten über mehrere Standorte und Backup
- NoSQL: Auf Geschwindigkeit optimierte, nichtrelationale Datenbanksysteme
- Anwendungshosting: Der eigentliche Betrieb von Webanwendungen

Da die jeweiligen Schnittstellen unterschiedlich sind, ist ein Wechsel zwischen Anbietern ein aufwendiges Unterfangen. Dies steht in einem krassen Missverhältnis zum im Cloud-Hype propagierten Flexibilitätsanspruch. Vielmehr scheint die Gefahr gegeben, dass sich Nutzer von Cloud-Plattformen in eine einseitige Abhängigkeit begeben können, einem „vendor lock-in“.

5.4 Ein Anwendungsszenario

Aufgrund der verteilten Natur [6] der Cloud-Services ist es naheliegend, dass sich insbesondere Webapplikationen diese Angebote zunutze machen. Im Rahmen der PG soll beispielhaft anhand einer zu entwickelnden Web-basierten Gruppenverwaltung untersucht werden, wieviel Freiheit durch die entwickelten vereinheitlichten Schnittstellen auf den Entscheidungsachsen von Abbildung 1 geschaffen werden kann. Die Gruppenverwaltung soll darauf ausgelegt sein, die Kommunikation zwischen Musikbands und ihren Fans zu unterstützen und hierbei verschiedenartige Cloud-Dienste nutzen: Terminaustausch für Tourdaten beispielsweise über einen SaaS Kalender (Google Calendar etc.), Dateiablage für Konzertmitschnitte über IaaS-Storage (Amazon S3 etc.) und Betrieb der Anwendung auf einem PaaS-Angebot (Heroku etc.). Die genaue Ausgestaltung obliegt der PG, wobei im Blick behalten werden sollte, was derzeitige Plattformen wie MySpace.com zu attraktiven Angeboten in diesem Anwendungsbereich macht – und wo Verbesserungspotential besteht.



Anhand der Beispielanwendung sollen folgende Fragen beantwortet werden: Ist es möglich, einen gemeinsamen Nenner zwischen verschiedenen Angeboten zu finden und eine einheitliche Schnittstelle zu schaffen? Kann das Wechseln zu einem anderen Anbieter praktisch „auf Knopfdruck“ ermöglicht werden? Wie können mehrere ähnliche Angebote im Sinne einer service-orientierten Orchestrierung [7, 8] genutzt werden, um z.B. ein „vendor lock-in“ zu vermeiden?

5.5 Generelle Vorgehensweise

Am Anfang der Projektgruppe wird die Einarbeitung in die theoretischen Grundlagen serviceorientierter Architekturen, der Entwicklung von Webanwendungen und des Cloud-Computings stehen. Anschließend werden bestehende Cloud-Lösungen betrachtet und dahingehend untersucht, inwieweit sie dazu dienen können, die üblichen Ressourcenanforderungen von Webapplikationen zu bedienen. Nachdem gängige benötigte Dienste identifiziert wurden, kann der Entwurf von vereinheitlichten Schnittstellen und die Implementierung einer entsprechenden Funktionsbibliothek geschehen. Es soll mindestens eine am obigen Anwendungsszenario orientierte Anwendung entstehen, die über die vereinheitlichte Schnittstelle auf Cloud-Lösungen zugreift. Hierbei soll darauf geachtet werden, dass verschiedenartige Cloud-Angebote (wo möglich auch von verschiedenen Anbietern) untersucht und genutzt werden. Jedes Angebot ist dabei auf seine spezifischen Stärken und Schwächen hin zu untersuchen, da nicht zu erwarten ist, dass sich für jede Cloud-Dienstleistung eine vereinheitlichte Schnittstelle vereinbaren lässt, ohne auf interessante Funktionen einzelner Angebote verzichten zu müssen.

6. Teilnahmevoraussetzungen

Erforderlich

- Fundierte Kenntnisse mindestens einer objektorientierten Programmiersprache (z. B. Java)
- Kenntnisse im Gebiet Software-Design und Implementierung

Wünschenswert

- Grundlegende Kenntnisse serviceorientierter Architekturen
- Grundlegende Kenntnisse zur Geschäftsprozessmodellierung
- Datenbankkenntnisse

7. Minimalziele

- Einarbeitung in grundlegende Webtechnologien.
- Eingehende Recherche im Bereich Cloud-Computing und Einarbeitung in aktuelle Technologien aus diesem Bereich.
- Systematischer Vergleich verschiedener Cloud-Dienste und Ansätze.
- Erfolgreiche Implementierung einer vereinheitlichten Schnittstelle für mindestens eine Dienstart (z.B. Storage oder Datenbank).
- Implementierung einer Anwendung aus dem Anwendungsszenario, die von der entwickelten Technologie Gebrauch macht.

8. Literatur

- [1] Brian Hayes. *Cloud computing*. In Commun. ACM 51, 7 (July 2008), 9-11.
<http://dl.acm.org/citation.cfm?doid=1364782.1364786>
- [2] National Institute of Standards and Technology. *The NIST Definition of Cloud Computing*
<http://www.nist.gov/itl/cloud/upload/cloud-def-v15.pdf>
- [3] Amazon Simple Storage Service (Amazon S3)
<http://aws.amazon.com/de/s3/>
- [4] Amazon Elastic Compute Cloud (Amazon EC2)
<http://aws.amazon.com/de/ec2/>
- [5] Google App Engine
<https://developers.google.com/appengine/>
- [6] Andrew S. Tanenbaum, Maarten van Steen - *Distributed Systems: Principles and Paradigms*
 Prentice Hall International; Auflage: 2nd rev. ed. (23. Oktober 2006)
 ISBN 0136135536
- [7] Tiziana Margaria, Bernhard Steffen: *Middleware: just another level for orchestration*. MNCNA 2007: 4
- [8] Tiziana Margaria, Bernhard Steffen, Christian Kubczak: *Evolution support in heterogeneous service-oriented landscapes*. J. Braz. Comp. Soc. 16(1): 35-47 (2010)

9. Rechtlicher Hinweis

Die Ergebnisse der Projektarbeit und die dabei erstellte Software sollen der Fakultät für Informatik uneingeschränkt für Lehr- und Forschungszwecke zur freien Verfügung stehen. Darüber hinaus sind keine Einschränkungen der Verwertungsrechte an den Ergebnissen der Projektgruppe und keine Vertraulichkeitsvereinbarungen vorgesehen.