

# Ausgewählte Kapitel der Systemsoftware: Cloud Computing

*„Zunächst heiter bis wolzig, später dauerhaft bedeckt“*

Timo Höning

Friedrich-Alexander-Universität Erlangen-Nürnberg  
Lehrstuhl Informatik 4 (Verteilte Systeme und Betriebssysteme)

<http://www4.informatik.uni-erlangen.de>

Sommersemester 2010



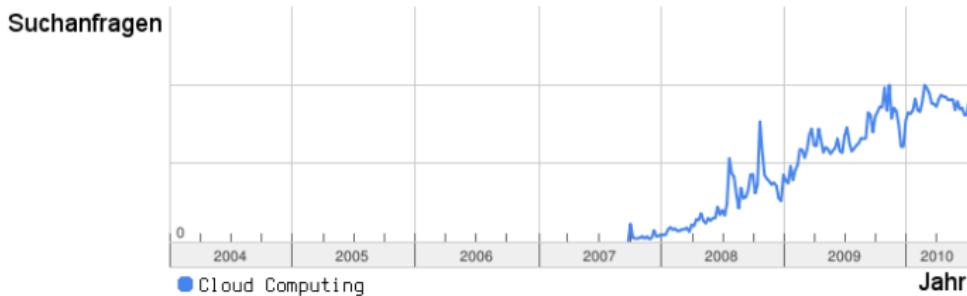
*„Zunächst heiter bis wolkig, später dauerhaft bedeckt“*

- Semesterabschluss
  - **Keine** Wiederholung
  - **Aktive** Teilnahme erwünscht
- Themen
  - Phänomen Cloud Computing: Ursprung und Wurzeln
  - Prinzip der Offenheit
  - Fallstudie Twitter



»We call it *cloud computing* ( . . . ) «

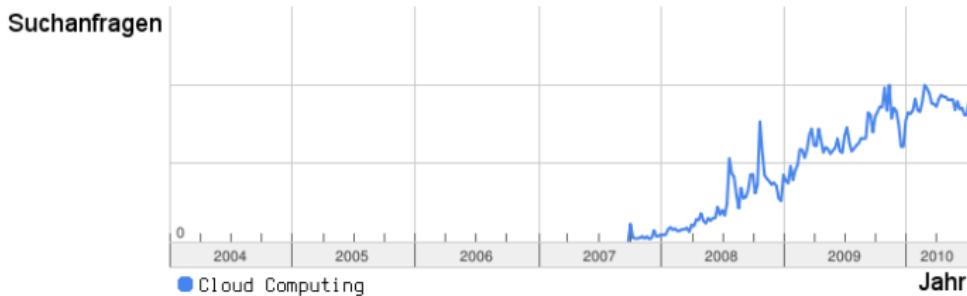
9. August 2006



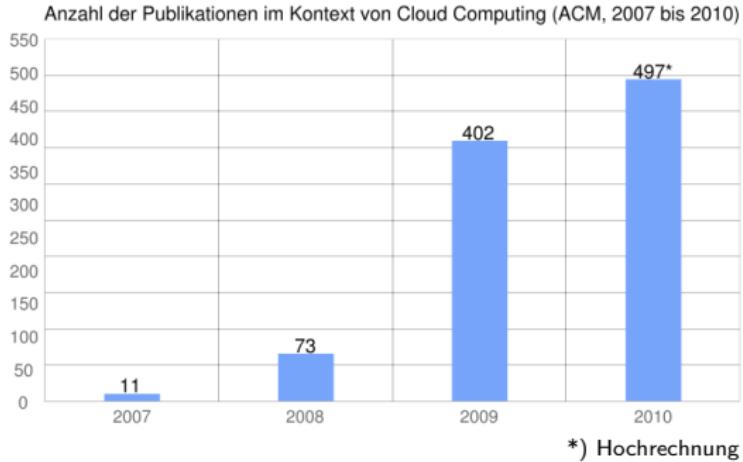
»We call it *cloud computing* (...) «

Eric Schmidt ([Google](#))

Search Engine Strategies Conference, San Jose, 9. August 2006



# Cloud Computing: . . . und seine Folgen.



- Gravierende Auswirkungen des Modeworts „Cloud Computing“
  - **Forschung**
  - Wirtschaft
- Cloud Computing
  - Fokus auf **Technik**
  - Cloud  $\hat{=}$  Internet, Cloud Computing  $\hat{=}$  Internet + ?



# Cloud Computing: Zeitpunkt, Grundprinzipien

---

- Wieso entstand Cloud Computing zu dieser Zeit? Wieso nicht früher?



- Wieso entstand Cloud Computing zu dieser Zeit? Wieso nicht früher?
  - **Infrastruktur**, Hard- und Software-Technologie ✓
    - Datenübertragungsrate, Commodity Hardware, Virtualisierung
  - **Systemsoftware**, Verteilte Algorithmen ✓
    - Google File System/Cassandra, MapReduce/Hadoop, Zookeeper
  - **Dienstleistungsprinzip**, Geschäftsmodell („... as-a-Service“) ✓
    - Software as a Service, *Service-Oriented Architecture* (SOA)
- Erfüllbarkeit der Grundprinzipien des Cloud Computings
  - **Skalierbarkeit**
    - unter Wahrung von: Hochverfügbarkeit, Verlässlichkeit, Effizienz
  - **On-Demand**
    - zum Ermöglichen von: dynamischer Zuordnung der Ressourcen, Abrechnung nach tatsächlichem Verbrauch



- Bob Bemer (IBM)
  - „**How to Consider a Computer**“: Erste wissenschaftliche Arbeit zum Thema Timesharing im Jahre 1959
  - „**The Father of ASCII**“: hat u. a. Backslash und Escape beigetragen
  - Weitblick: Erste Veröffentlichung zur Jahr-2000-Problematik im Jahre 1971 („**Time and the Computer**“)



- John McCarthy (Stanford University)
  - „**Utility Business Model**“ (1961)
  - Rechenleistung und Spezialanwendungen sollen verkauft werden wie andere Ressourcen – wie Wasser und Strom



- Gemeinsamkeiten
  - Teilen von Ressourcen
  - Abrechnungsmodell
  - Aufteilung von Anwendungslogik und Benutzerschnittstelle
- Unterschiede
  - Transparenter Zugriff, ortsunabhängig
  - *Unendlichkeit* der Ressourcen
  - Grad der Verteilung, Netzstruktur
- Resultate
  - Transfer bekannter Grundlagen und Prinzipien
  - Erfüllen der Anforderungen von Technologie *und* Wirtschaft
  - Grundlage vieler **offener** Forschungsprojekte *und* Produkte



- Cloud Computing und Open Source

- Alle Unternehmen sind *wolkig*; die einen mehr, die anderen weniger
  - <http://www.{amd,dell,ibm,microsoft,novell,vmware}.com/cloud>
- Optimierung und Wiederverwendung
  - Teilen der Implementierungsarbeit, gemeinsame Problemfindungsprozesse
- Beschleunigt die Umsetzung eigener Ziele
  - Hemmschwelle auf ein Minimum reduzieren

- Vorteile von Open Source im Kontext von Cloud Computing

- Open-Source-Projekte lösen das Problem der Anbieterabhängigkeit („Vendor Lock-In“)
- Einsicht in Quellcode beschleunigt Forschungsprojekte *und* Produktentwicklung
- Garantiert Heterogenität



- Paradebeispiele (losgelöst von ihren Unternehmenskonzepten)
  - Facebook
    - Apache Cassandra, Apache Hive, HipHop for PHP
  - Twitter
    - FlockDB, Murder, Clients für Cassandra und Zookeeper
  - Rackspace
    - OpenStack Initiative
- Explizite Kritik an Google (losgelöst vom Unternehmenskonzept)
  - **Minimales** Engagement an *offenen* Cloud-Computing-Projekten
  - **Maximales** Engagement um Wettbewerbsvorteil voll auszuschöpfen
  - Wenn überhaupt werden Errungenschaften nur ansatzweise geteilt
  - Google würde ohne Open-Source-Software **nicht** existieren; vgl. GPLv3



## »Twitter Loves Open Source.«

<http://engineering.twitter.com/>

- Wieso Twitter?
  - **Kein** typisches Cloud-Unternehmen
  - **Minimalistischer Ansatz** (Internet-SMS, „one-to-many“)
  - **Offen**
- Fakten
  - Ohne Cloud Computing würde Twitter nicht existieren
  - Amazon Simple Storage Service (Amazon S3):  
[http://s3.amazonaws.com/twitter\\_production/a/1279322210/images/fronts/logo.png](http://s3.amazonaws.com/twitter_production/a/1279322210/images/fronts/logo.png)
- Probleme
  - **Extrem** schnelles Wachstum, Skalierbarkeit
  - Stößt **häufig** an Technologiegrenzen



# Cloud Computing: Fallstudie Twitter

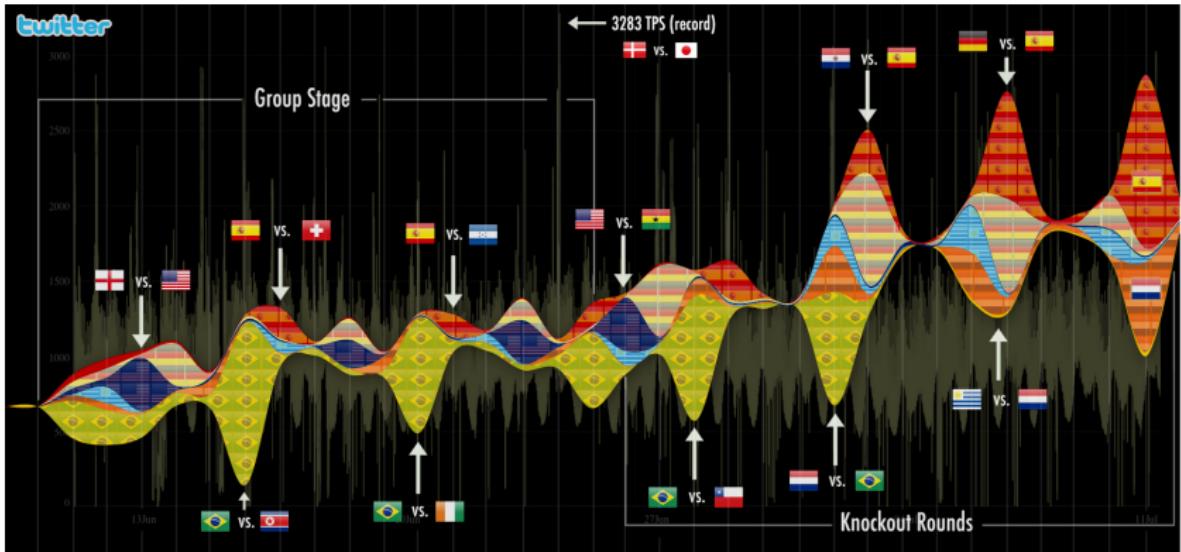
- Fakten (in Zahlen, Stand Mitte 2010)
  - 253 Mitarbeiter
- Twitter und Cloud Computing
  - Anwendungslogik komplett in der Cloud
  - Nutzt Cloud-Technologie (Amazon S3, Cassandra, Zookeeper etc.)
  - Transparenter Zugriff
  - Veröffentlicht Eigenentwicklungen als Open Source (z. B. Murder)



- Fakten (in Zahlen, Stand Mitte 2010)
  - **253** Mitarbeiter
  - **55 Millionen** „Tweets“ pro Tag
  - **100 Millionen** Benutzer (pro Tag 300.000 neue Benutzer)
  - **3 Milliarden** API-Zugriffe pro Tag, das entspricht dem gesamten Netzverkehr von Yahoo
  - **19 Milliarden** Suchanfragen pro Monat. Zum Vergleich:
    - Google: 88 Milliarden
    - Yahoo: 9,4 Milliarden
    - Bing (Microsoft): 4,1 Milliarden
- Twitter und Cloud Computing
  - Anwendungslogik komplett in der Cloud
  - Nutzt Cloud-Technologie (Amazon S3, Cassandra, Zookeeper etc.)
  - Transparenter Zugriff
  - Veröffentlicht Eigenentwicklungen als Open Source (z. B. Murder)



# Cloud Computing: Technologiegrenzen



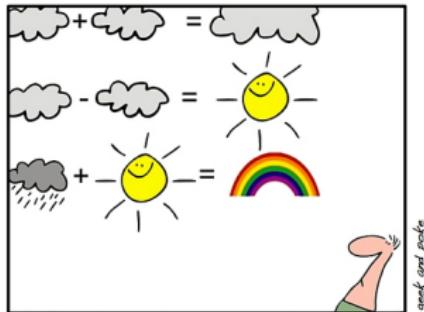
- Höchstwert: 3283 „Tweets“ pro Sekunde (TPS)
- Zahlreiche Ausfälle **vor** dem Finale der Weltmeisterschaft
- Das WM-Finale war bis dato das Ereignis mit den meisten „Tweets“ (kumuliert), **kein** Ausfall von Twitter



- **Cloud Computing** ist das Resultat paralleler, teilweise unabhängiger Entwicklung; **nicht geplant, aber auch kein Zufall**
- Vorhandene **Cloud-Projekte** sind **erste Prototypen**; nicht weniger, aber auch nicht mehr
- Die Anforderungen von Cloud Computing wird das Gebiet der **Systemsoftware nachhaltig beeinflussen**; auch in der Zukunft
- Cloud Computing bildet die **Grundlage für Unternehmen ohne Infrastruktur**; ansatzweise wie Twitter



„Zunächst heiter bis wolkig, später dauerhaft bedeckt“



SIMPLY EXPLAINED - PART 17:  
CLOUD COMPUTING

