

Hochschule Karlsruhe
Technik und Wirtschaft
UNIVERSITY OF APPLIED SCIENCES

Cloud Computing

Osman Aksoy, Marc Fischer,
Christian Kolodziej, XXX

Hochschule Karlsruhe - Technik und Wirtschaft
University of Applied Sciences
Moltkestraße 30, D-76133 Karlsruhe

29. Januar 2010

Rechtlicher Hinweis

Dieses Werk ist urheberrechtlich geschützt. Das Urheberrecht und die Verfügungsgewalt über Verwertungsrechte obliegen einzig den Autoren. Sie gestatten der Hochschule Karlsruhe die Nutzung zum Zweck der Korrektur und Bewertung innerhalb des dafür vorgesehenen Zeitraums sowie das Recht, die in das Eigentum der Hochschule Karlsruhe übergegangenen körperlichen Kopien über diese Frist hinaus zu archivieren und gegebenenfalls zum Nachweis der Notengebung zu gebrauchen. Jede darüber hinausgehende Nutzung oder Verwertung, insbesondere die Wahrnehmung einfacher oder besonderer Verwertungsrechte – was die hochschulinterne Veröffentlichung in Teilen oder als Ganzes einschließt – wird der Hochschule Karlsruhe ausdrücklich nicht gestattet. Die Autoren weisen ferner ausdrücklich darauf hin, dass es kein zur Herleitung eines solchen Anspruchs geeignetes Rechtsinstrument gibt, dessen Gültigkeit sich auf dieses Werk erstreckt. Die Autoren behalten sich jedoch für ihren jeweiligen Anteil dieses Werks vor, nach eigenem Ermessen sowohl gegenüber der Hochschule Karlsruhe als auch gegenüber Dritten von den vorgenannten Rechten Gebrauch zu machen. Dies bedarf gegenüber der Hochschule Karlsruhe jedoch in jedem Fall der ausdrücklichen, schriftlichen und zweckgebundenen Einwilligung der Autoren.

Inhaltsverzeichnis

1	Einleitung	1
2	Definition	3
2.1	Das 3-Ebenen-Modell	4
2.2	Organisationsformen von Clouds	5
2.2.1	Public Clouds und Private Clouds	5
2.2.2	Hybrid Clouds	6
2.3	Cloud-Computing-Angebote	6
3	IaaS und die technischen Grundlagen des Cloud Computings	8
4	Platform as a Service	9
4.1	Funktionsweise und Bedeutung von PaaS	9
4.2	Beispiele für PaaS in einer Anwendungsarchitektur	10
4.2.1	Ein Webshop als Beispiel	10
4.2.2	Aufbau Inhouse	11
4.2.3	Aufbau mit IaaS	12
4.2.4	Aufbau mit PaaS	13
4.3	Beispielhafte PaaS-Ergänzung durch SaaS	15
4.3.1	Aufgabe Anwendungsfall	16
4.3.2	Aufgabenrealisierung mit Tomcat und MySQL	16
4.3.3	Google App Engine	17
4.3.4	Amazon Web services	24
4.4	Andere Definitionen	25
5	Software as a Service (SaaS)	27
5.1	SaaS Vorgehensweise	28
5.2	SaaS-Anwendungsbeispiele	29
5.2.1	Salesforce CRM	29
5.2.2	Sipgate Team	32
6	Unternehmerische und rechtliche Einordnung	34
6.1	Geschäfts-, Nutzungs- und Abrechnungsmodelle	34
6.1.1	Infrastructure as a Service (IaaS)	35
6.1.2	Platform as a Service (PaaS)	37

6.1.3	Software as a Service (SaaS)	38
6.2	Investitions- vs. Betriebskosten	39
6.3	Von der Wertschöpfungskette zum Wertschöpfungsnetz	40
6.4	Sourcing	42
6.5	Datensicherheit und Datenschutz	44
6.5.1	Transparenz	44
6.5.2	Rechtlicher Rahmen nach BDSG	45
6.6	Vergleich zwischen dem deutschen und dem weltweiten Markt	45
7	Fazit	48
	Literaturverzeichnis	49

Kapitel 1

Einleitung

In den vergangenen Jahren hat die moderne Welt der Unternehmens-IT rasante Veränderungen durchlebt. Einerseits haben sich IT-Systeme von einfachen monolithischen Gebilden hin zu komplexen verteilten Anwendungen mit hohen Anforderungen entwickelt. SOA ist eines der aktuellen Themen der Gegenwart. Parallel zu den IT-Systemen hat sich auch ihre Bedeutung für Unternehmen verändert. IT-Systeme sind nicht mehr nur technische Unterstützung. Für viele Unternehmen sind sie geschäftskritisch oder stellen sogar die Geschäftsgrundlage dar. In diesem Fall bedeutet die Nichtverfügbarkeit dieser Systeme das Ende des jeweiligen Geschäftsmodells. Mängel in der eigenen IT stellen je nach Konkurrenzsituation auf den heutigen Märkten einen erheblichen Wettbewerbsnachteil dar.

Die Bedeutung der IT-Systeme hat direkte Auswirkungen auf die Anforderungen, die ein Unternehmen an seine IT unweigerlich stellen muss. Die Systeme müssen immer höheren Standards entsprechen, was sich in den Anforderungen hinsichtlich der Verfügbarkeit und Performance sowie der Flexibilität widerspiegelt. Um im Wettbewerb zu bestehen, müssen die IT-Systeme jederzeit im geforderten Maße einsatzfähig sein, was insbesondere für die Unternehmen ein Problem darstellt, bei denen der IT-Betrieb nicht das Kerngeschäft ist. Aus Sicht dieser Unternehmen ist es naheliegend den Betrieb der eigenen IT an spezialisierte externe Dienstleister abzugeben, die aufgrund ihrer Erfahrung diese Aufgabe effizienter und qualitativ hochwertiger erfüllen können.

Da trotz des Outsourcings des IT-Betriebs der eigentliche Geschäftsbetrieb beim Unternehmen verbleibt, ist eine Kopplung der verbleibenden internen Systeme und Anwendungen an die externen Leistungen nötig. Für die Anwendungen bedeutet dies, dass sie verteilt ausgeführt werden und deshalb Schnittstellen zur systemübergreifenden Kommunikation bieten müssen. Durch die Entwicklung hin zu verteilten Systemen und insbesondere zu serviceorientierten Architekturen erfüllen die gängigen Paradigmen der Softwareentwicklung diese Anforderungen bereits.

Auf Seite des externen Leistungserbringers ist es wiederum entscheidend, die IT-Systeme der Kunden einfach zu verwalten und die vertraglich vereinbarte Leistung jederzeit zur Verfügung stellen zu können und gleichzeitig eine möglichst hohe Auslastung seiner Systeme zu erzielen. Hier haben insbesondere die verschiedenen Virtualisierungstechniken auf Basis der x86-Architektur, die in vergangenen Jahren entwickelt wurden, neue Möglichkeiten eröffnet.

In den vergangenen Jahren haben sich somit verschiedene Techniken parallel entwickelt, die den zentralen Betrieb von IT-Diensten bei darauf spezialisierten Dienstleistern erleichtern. Die Integration dieser Techniken zu einem geschlossenen Produkt war somit nur eine Frage der Zeit und wird nun in verschiedenen Formen unter dem Begriff „Cloud Computing“ gehandelt.

Cloud Computing wird seither vielerorts – oft unreflektiert – als Revolution und Allheilmittel gepriesen. In dieser Ausarbeitung möchten wir zunächst den Begriff und die Features des Cloud Computings greifbar machen, um daraufhin die technische Funktionsweise einer Cloud darzulegen. Darauf aufbauend zeigen wir die Integration von Cloud-Diensten in eigenen Anwendungen sowie die Nutzung fertiger Anwendungen aus der Cloud. Abschließend erfolgt die Betrachtung der aktuellen Marktsituation sowie des rechtlichen Rahmens.

Auf Grundlage dieser Informationen soll der Leser in die Lage versetzt werden, die tatsächlichen Möglichkeiten des Cloud-Computings für sich selbst beurteilen zu können, um eine technisch fundierte Entscheidung bezüglich des konkreten Einsatzes von Cloud Computing zu treffen.

Karlsruhe, im Dezember 2009

Kapitel 2

Definition

von Osman Aksoy

Cloud Computing ist die echtzeitige Bereitstellung von IT-Diensten, die über Internet flexibel und bedarfsgerecht zur Verfügung gestellt werden und damit „ein Geschäfts- und Betriebsmodell zur Bereitstellung und Verwendung von IT-Ressourcen“¹. Damit hat man aus Unternehmenssicht die Möglichkeit, an eine Vielzahl von Kunden hoch standardisierte und massiv skalierbare IT-Fähigkeiten über die „Wolke“ bereitzustellen. Aus Nutzeransicht ermöglicht Cloud Computing eine Ressourcenflexibilität und somit eine Umverteilung von Investitions- zu Betriebsaufwand.

Die IT-Leistungen beziehen sich auf die folgenden drei Punkte (siehe [BIT09]):

- Basisinfrastruktur,
- Plattformen für Anwendungsentwicklungen und -betrieb oder
- Anwendungen

Da man die IT-Dienstleistungen auf drei Ebenen betrachtet, gibt es in der Praxis auch zahlreiche Definitionen für den Begriff „Cloud Computing“. Je nach Interessenlage betonen Software-, Service- oder Infrastrukturspezialisten unterschiedliche Aspekte. Im Folgenden werden einige dieser Definitionen erwähnt, die durch eine Anbieterbefragung der Forrester Research entstanden ist (vgl. [Her09]).

Cloud Computing steht für einen Pool aus abstrahierter, hochskalierbarer und verwalteter IT-Infrastruktur, die Kundenanwendungen vorhält und nach Verbrauch abgerechnet wird. Das amerikanische Marktforschungs- und Beratungshaus Saugatuck Technology fasst den Begriff weiter: „Cloud Computing umfasst On-Demand-Infrastruktur (Rechner, Speicher, Netze) und On-Demand-Software (Betriebssysteme, Anwendungen, Middleware, Management- und Entwicklungs-Tools), die jeweils dynamisch an die Erfordernisse von Geschäftsprozessen angepasst werden. Dazu gehört auch die Fähigkeit, komplette Prozesse zu betreiben und zu managen.“ Bei Cloud Computing geht es im Kern darum, Ressourcen dynamisch zur Verfügung zu stellen, erläutert Alfred Zollar, Chef der Tivoli-Sparte in IBMs Software Group. Im Vergleich zu älteren Konzepten könnten Nutzer mit erheblich schnelleren Reaktionen auf Kapazitätsanfragen rechnen.

¹Siehe [Mas09].

Die Online-Enzyklopädie Wikipedia erklärt das Konzept so: „Cloud Computing (...) ist ein Begriff aus dem Bereich der Computertechnologie, der in engem Zusammenhang mit der Vision des Grid Computing steht. Die Anwender einer Software betreiben die Software-Applikationen und die dazu notwendige Hardware (Server etc.) nicht mehr selbst, sondern beziehen sie über einen Anbieter, der beides für ihn und andere Nutzer betreibt“ [Wik09].

2.1 Das 3-Ebenen-Modell

Es ist weitgehend akzeptiert, dass drei Ebenen zusammen die Serviceumgebung des Cloud Computing ergeben. Diese drei Ebenen haben die Gemeinsamkeit, dass die IT-Leistungen als Dienste bzw. „as a Service“ bereitgestellt werden.

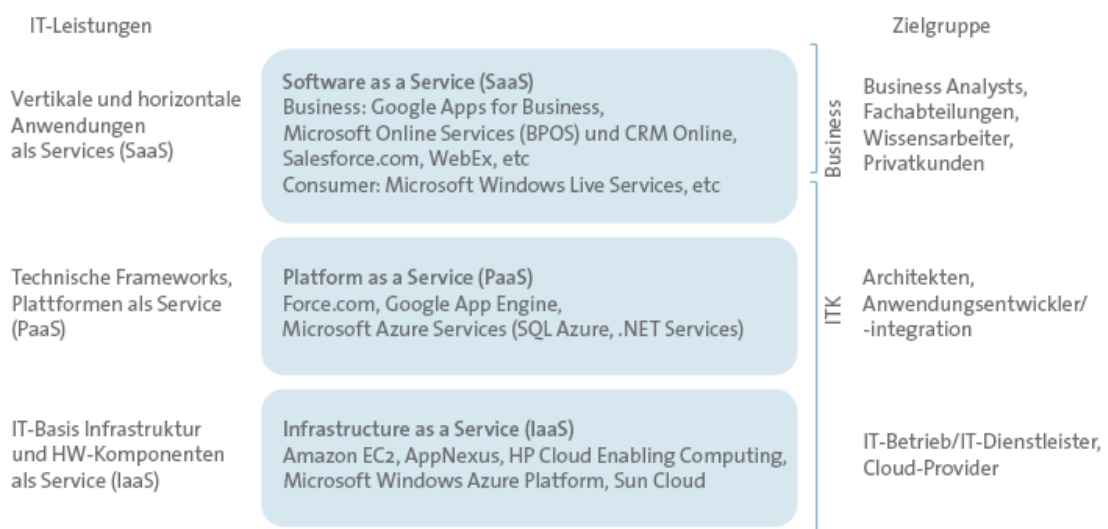


Abb. 2.1: 3-Ebenen-Modell, Quelle: [BIT09]

Auf der untersten Ebene befinden sich die IT-Leistungen der Basisinfrastruktur (IaaS). Sie bilden das Tätigkeitsfeld der Spezialisten für den IT-Betrieb sowie der IT-Dienstleister. Auf technologischer Ebene wird hier im wesentlichen wenig veredelte Rechen- und Speicherleistung auf virtualisierten Servern sowie Netzwerkinfrastruktur-Funktionalität mit hohem Standardisierungsgrad und intelligentem System- Management als Service bereitgestellt. Dabei können die einzelnen Funktionalitäten auch eng verbunden sein und als integrierter Service angeboten werden.

Eine Ebene darüber liegen IT-Leistungen für Entwickler-Plattformen (PaaS). Mit den Cloud Services dieser Ebene befassen sich System-Architekten und Anwendungsentwickler. PaaS beschreibt Services auf der Anwendungs-Infrastruktur-Ebene (Middleware- Layer), die auf Basis von technischen Frameworks, also Entwicklungs-Plattformen, angeboten werden. Mit ihnen lassen sich Anwendungskomponenten entwickeln und integrieren. Hier finden sich je nach Hersteller Cloud Ser-

vices für Datenbank-Funktionalitäten, Zugriffskontrolle, Workflow-Steuerung und für die Synchronisation von Anwendungen und Endgeräten in vernetzten Systemen.

Die oberste Ebene umfasst Anwendungen. Diese werden im Markt als Software as a Service (SaaS) bezeichnet. Diese Dienste richten sich an Anwender. Geschäftsanwendungen werden als standardisierte Services von einem Dienstleister bereitgestellt. Dabei sind ihre Anpassungs- und Integrationsmöglichkeiten oft eingeschränkt. Ein bekanntes Beispiel dafür ist Salesforce CRM. Aber auch Desktop-, Kollaborations- und Kommunikations-Anwendungen sowie industriespezifische Geschäftsabläufe, die vollständig von der Technologie abstrahiert sind, fallen in diesen Bereich (siehe [BIT09]).

2.2 Organisationsformen von Clouds

Nachdem wir das 3-Ebenen-Modell definiert haben steht die Frage offen, wie solche virtualisierten und standardisierten Cloud-Ressourcen zur Verfügung gestellt und genutzt werden. Unter Betriebs-, Eigentums- und Organisationsaspekten unterscheidet man zwischen den Private Cloud und den Public Cloud. Diese beiden Cloud-Formen unterscheiden sich nicht in der technischen Realisierung, sondern sie unterscheiden sich nur organisatorisch. Virtualisierung, Service-Orientierung und Nutzung der Services über das Internet sind dabei wesentliche Elemente der technischen Implementierung.

2.2.1 Public Clouds und Private Clouds

Public Cloud befindet sich im Eigentum des IT-Dienstleisters. Der Zugriff erfolgt über Internet. Viele Kunden teilen sich eine virtualisierte Infrastruktur. Die Nutzung erfolgt flexibel und schnell durch Subskription. Eine Public Cloud stellt eine Auswahl von hochstandardisierten Geschäftsprozess-, Anwendungs- und/oder Infrastruktur-Services (Service-Modulen) auf einer variablen, nutzungsabhängigen Basis zur Verfügung. Auf die Form und den physischen Ort der Datenhaltung, auf Compliance- und Sicherheitsaspekte hat der Nutzer normalerweise keinen Einfluss (siehe [BIT09]).

Private Cloud wird vom Unternehmen selbst betrieben und es dürfen nur autorisierte Personen auf den Cloud zugreifen, der Vorteil bei Private Cloud ist natürlich die erhöhte Sicherheit. Der Zugriff erfolgt meist über Intranet. Bei einer Private Cloud handelt es sich um eine effiziente, standardisierte und sichere IT-Betriebsumgebung unter Kontrolle des Unternehmens, die aber eine individuelle, auf die Geschäftsprozesse eines Unternehmens zugeschnittene Anpassung erlaubt. Mehr Informationen zu den Datenschutzaspekten von Cloud Computing finden sich in Kapitel 6.5.

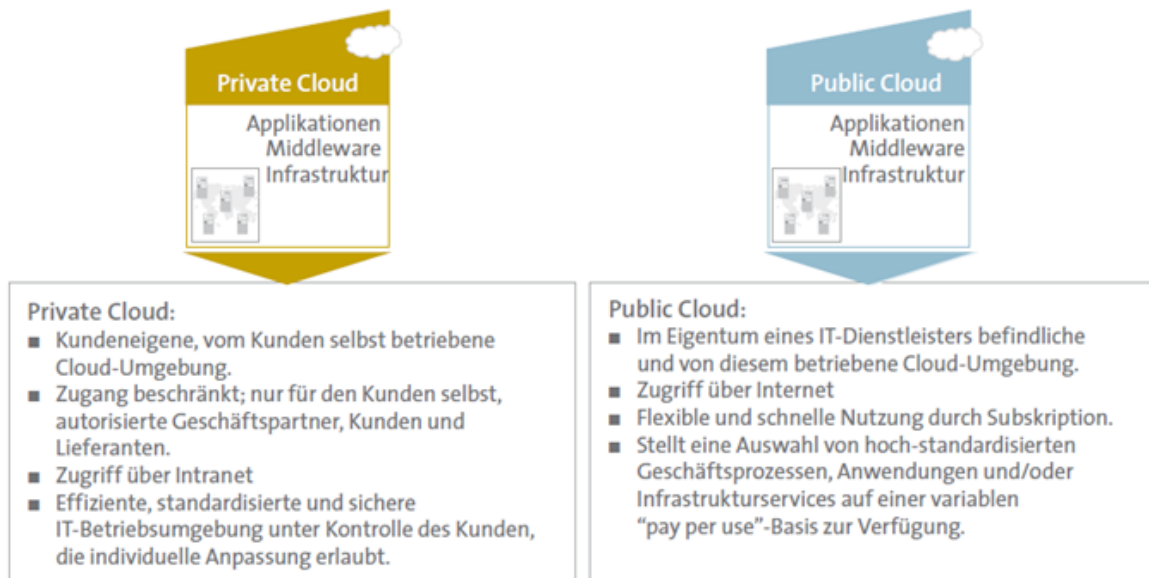


Abb. 2.2: Unterscheidung in Private und Public Cloud, Quelle: [BIT09]

2.2.2 Hybrid Clouds

In der Realität werden auf absehbare Zeit überwiegend Mischformen (Hybrid Clouds) vertreten sein. Hybrid Clouds sind mögliche Nutzungskombinationen von Private Clouds, Public Clouds und traditioneller IT-Umgebung. Die Herausforderung besteht darin, traditionelle IT-Umgebung, Private Cloud und/oder Public Cloud auf der Applikations-, der Middleware- und der Infrastruktur-Ebene in Bezug auf Services und Sicherheit so zu integrieren, dass sich eine heterogene Umgebung für den Nutzer homogen darstellt.

Der Endanwender an seinem Eingabegerät erwartet eine integrierte, den Geschäftsprozess optimal unterstützende Anwendung, unabhängig davon, ob diese Applikation oder Teile davon in einer Private Cloud, in einer Public Cloud oder auf der traditionellen IT-Umgebung zur Verfügung stehen. Letztendlich werden die Punkte Sicherheit und Service-Integration (Interoperabilität) über die Akzeptanz und damit über den Erfolg des Cloud-Computing-Modells entscheiden (vgl. [BIT09]).

2.3 Cloud-Computing-Angebote

Derzeit beobachten wir eine steigende Zahl der Angebote von On-Demand-Internetdiensten. Prominente Dienstleister wie Amazon, Google, SUN, IBM, Oracle, Salesforce etc. erweitern ihre Computing-Infrastrukturen und Plattformen zu einem zentralen Element für die Bereitstellung von Spitzendiensten für Computerberechnungen, Speicherung, Datenbanken und Anwendungen. Anwendungsdienste könnten E-Mail, Office-Anwendungen, Finanzdienste, Video-, Audio- und Datenverarbei-

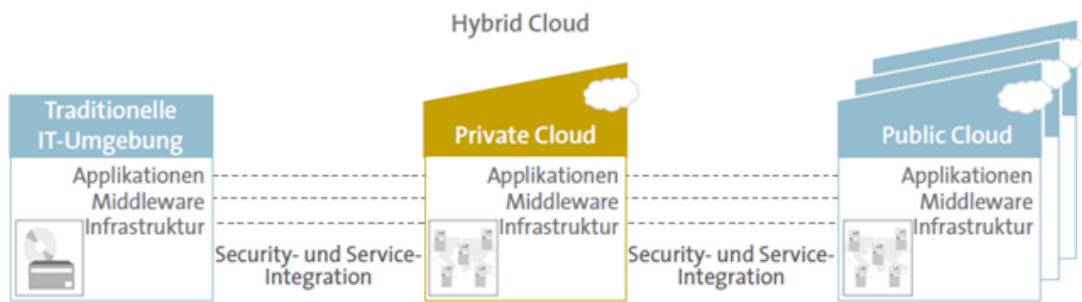


Abb. 2.3: Hybrid Cloud als Mischform zwischen Private und Public Cloud, Quelle: [BIT09]

tung sein. (siehe [Com09]).

Bereits 2002 begann Amazon, einen Web Service für andere Webseiten oder Client-Anwendungen anzubieten. Seit 2006 werden die Amazon Web Services (AWS) als Infrastruktur- Service-Plattform in der Cloud angeboten. Es handelt sich dabei um eine Sammlung verschiedener Webservices, die auf dem Webportal von Amazon.com im Internet angeboten werden. Nach Angaben von Amazon.com haben sich bereits mehr als 330.000 Entwickler für die Nutzung von AWS registriert. Dabei arbeitet Amazon auch mit anderen Software- oder SaaS-Anbietern wie Adobe Möglichkeiten von Cloud Computing zu eröffnen. Wollen Unternehmen, ISVs oder Entwickler neue webbasierte Geschäftsapplikationen programmieren und betreiben, können sie nun sowohl Datenbank, Logik- und User-Interface-Funktionen von Force.com als auch Speicher- und Rechenleistung der Amazon S3 und Amazon EC2 Dienste nutzen(siehe [BIT09]).

Abschließend möchte ich kurz auf Azure , das Cloud Computing-Konzept von Microsoft eingehen. Im Gegensatz zu den vorher genannten Cloud Computing Services ist Azure größtenteils bisher noch eine Vision, die in den nächsten Wochen und Monaten mit Leben gefüllt werden soll. Microsoft folgte kürzlich Salesforce.com mit seiner eigenen Plattform für den Aufbau Cloud-basierter Applikationen namens Windows Azure. David Thompson, Vice President von Microsoft Online, verkündete zudem, dass das Unternehmen plane, seine gesamte Enterprise Software in die Cloud zu verlagern.

Kapitel 3

IaaS und die technischen Grundlagen des Cloud Computings

Der Autor dieses Kapitels hat der Veröffentlichung leider widersprochen.

Kapitel 4

Platform as a Service

von Marc Fischer

4.1 Funktionsweise und Bedeutung von PaaS

IaaS stellt dem Kunden zunächst nur ein virtuelles System zur Verfügung. Bezüglich der Installation von Anwendungen ist der Kunde einerseits völlig frei, kann jedoch andererseits nicht auf höhere Dienste, wie etwa Anwendungsserver zurückgreifen.

Anders verhält es sich auf der nächst höheren Produkt-Ebene des Cloud Computings, der Platform as a Service, kurz PaaS. Hier erhält der Kunde nicht mehr nur mehr oder minder „nackte“ virtuelle Systeme sondern eine vollständige Laufzeitumgebung, die er nur noch durch seine eigene Software ergänzen muss.

Die Laufzeitumgebung entspricht dabei in der Regel üblichen Standards, kann also beispielsweise Java-, Ruby-, Python- oder .NET-Anwendungen beziehungsweise die entsprechenden Bytecode-Kompilate ausführen. Diese Anwendungen sind durch den Kunden selbst zu schreiben. Dazu bieten die meisten Cloud-Anbieter die Integration verschiedener Funktionen in bestimmte IDEs an, so dass beispielsweise Anwendungen mit wenigen Klicks in der Cloud ausgeführt oder getestet werden können.

Der Entwicklungsprozess ändert sich durch Cloud Computing jedoch nicht grundlegend. Die gängigen Paradigmen, insbesondere hinsichtlich verteilter Systeme, gelten auch hier. Sie werden hier besonders gefordert, da die einzelnen Komponenten einer Anwendung möglicherweise entfernt voneinander ausgeführt werden, sind Schnittstellen, insbesondere SOA auf Basis von Webservices, von besonderer Relevanz hinsichtlich der Interprozesskommunikation.

Die Infrastruktur eines PaaS-Dienstes kapselt dabei die Komplexität der darunterliegenden IaaS-Dienste und Anwendungsservern. Um die Leistung der Plattform beliebig zu skalieren muss die Anwendung über mehrere virtuelle Systeme, wie sie auf IaaS-Ebene verwendet werden, verteilt werden. Dazu ist das Deployment und Clustering von mehreren Anwendungsservern notwendig. Diese Aufgaben werden durch den PaaS-Dienst übernommen und sind in der Regel über eine

Schnittstelle ansprechbar, indem beispielsweise zusätzliche Ressourcen allokiert oder freigegeben werden. Der Kunde kann hier die Auslastung seiner Anwendung kontrollieren und die von ihm beim Cloud-Anbieter gebuchten Ressourcen entsprechend anpassen.

PaaS vereint also die Leistung mehrerer IaaS-Instanzen zu einer homogenen Ebene, die für den Kunden und die Anwendung in sich geschlossen erscheint und direkt für eigene Anwendungen genutzt und über einfache Schnittstellen gesteuert werden kann. Dies ist die wesentliche Leistung im Produkt „PaaS“.

Welche Funktionen die jeweilige Plattform bietet ist von Anbieter zu Anbieter unterschiedlich, auch die Integration in IDEs erstreckt sich auf eine Bandbreite von „nicht vorhanden“ bis hin zu eigenen IDEs für die Arbeit in der Cloud.

Die Grenze zwischen PaaS und SaaS ist nicht immer einfach zu ziehen. Neben PaaS bieten die meisten Anbieter auch SaaS-Komponenten ergänzend zu PaaS an. So kann ein Entwickler selektiv entscheiden, einzelne Anforderungen seiner Anwendung nicht selbst zu implementieren, sondern an einen solchen Dienst auszulagern. Ebenso werden bestimmte weitere Dienste, insbesondere Datenbanken und Speicherserver als Teil von PaaS angeboten.

4.2 Beispiele für PaaS in einer Anwendungsarchitektur

Im Folgenden sollen verschiedene Möglichkeiten, PaaS in eine Anwendung zu integrieren, besprochen. Die Beispiele sind dabei nicht an einen bestimmten PaaS-Anbieter gekoppelt, da ihre Angebote und Schnittstellen sich stark unterscheiden und es an dieser Stelle um das grundlegende Prinzip geht. Für die konkrete Implementation bieten alle Anbieter entsprechende Dokumentation.

4.2.1 Ein Webshop als Beispiel

Als Beispiel-Anwendung dient an dieser Stelle das Standard-Beispiel eines einfachen Webshops. Der Webshop stellt ein Produktsortiment auf einer Website zur Verfügung, die durch Kunden online erworben werden können. Dieser Services wird vollständig durch ein Unternehmen abgedeckt. Das heißt, sowohl die Datenaufbereitung des Produktkataloges, die Entwicklung des Webshops und der Businesslogik und der infrastrukturelle Betrieb der erforderlichen Komponenten werden durch das Unternehmen abgewickelt.

Der Webshop an sich ist ein System, das nach der 3-Tier-Architektur konzipiert ist. Die Schichten setzen sich wie folgt zusammen:

- 1. Tier / Persistenz-Schicht** In dieser Schicht werden alle Daten mit den dazugehörigen Selektierungs- und Persistierungsverfahren behandelt.
- 2. Tier / Business-Schicht** Gegebene Geschäftsprozesse mit deren Abhängigkeit und der Zuordnung

zur Persistenz-Schicht.

3. Tier/Präsentations-Schicht Die visuelle Darstellung von Daten und das Agieren zwischen Benutzern und der Business-Schicht sind der Kern dieses Teiles.

Nach diesen Schichten ergibt sich eine Aufteilung eines Webshops mit Katalog und Bestellverwaltung in die in Abbildung 4.1 auf der nächsten Seite gegliederten Serverkomponenten Datenbankserver, Webserver und Anwendungsserver. Im Datenbankserver befindet sich die komplette Datenbasis (Produktivdaten), die Datenbankanwendung und die administrativen Funktionen für den Betrieb einer Datenbank. Im Webserver befinden sich die, wie im Datenbankserver, administrativen Funktionen und zudem die Bereitstellung der Webanwendung in der Webserverinstanz und die Instanz selbst. In der letzten Serverkomponente konzentrieren sich die geschäftsrelevanten Prozesse wie auch der Administrationsaufwand.

Ein möglicher Aufbau der Anwendung wird in Abbildung 4.2 auf Seite 13 aufgezeigt. Die aufgezeigten Packages in den Klassendiagrammen sind mit den Serverkomponenten verknüpft. Dies bedeutet, dass für die Webanwendung alle im *com.example.webanwendung* befindlichen Klassen in dieser Serverkomponente zugeordnet sind und sich der Einsatz darin konzentriert. Für die Komponenten Datenbank und Anwendung ist dies vergleichbar.

4.2.2 Aufbau Inhouse

Wie oben beschrieben stellt das betreibende Unternehmen das Produkt Inhouse zur Verfügung. Der genaue Aufbau dieser Inhouse-Lösung ist nun Inhalt im nächsten Abschnitt.

Wesentlicher Punkt der Inhouse-Lösung ist, dass alles durch das Unternehmen gehostet und gewartet wird. Abbildung 4.1 auf der nächsten Seite zeigt einen physikalischen Server, auf dem die unterschiedlichen Serverkomponenten gehostet sind. Einige Vor- und Nachteile sind hierfür in Tabelle 4.1 aufgelistet.

Die Anwendung ist ansich nicht in verschiedene Systemen getrennt. Alle Anwendungsteile sind quasi eine einzige Anwendung und laufen nur auf einem System, beziehungsweise in einem internen Verbund ohne konkrete Trennung nach SOA, ab. Das Klassendiagramm in Abbildung 4.2 auf Seite 13 zeigt hierzu den Aufbau des Systems als geschlossene Einheit.

Vorteile	Nachteile
eigene Verwaltung	großer Verwaltungsaufwand
sind im eigenen Besitz	hohe Wartungskosten
totale Kontrolle	Fachwissen erforderlich
	Skalierbarkeit schwer umzusetzen

Tabelle 4.1: Vor-/Nachteile Inhouse

Mit diesem Aufbau eines Systems kann man zufrieden sein, jedoch wird nur unter hohem Kostenaufwand und viel Fachwissen das System am Laufen gehalten. Um dies abzuwenden wird nun der

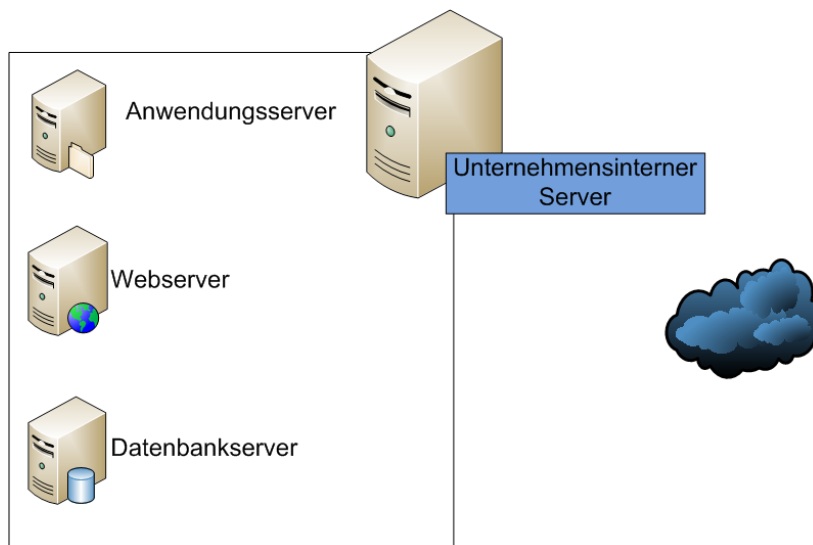


Abb. 4.1: Aufbau Webshop Inhouse

erste Schritt in Richtung Cloud vollzogen. Das heißt im Genauen, das Beispielunternehmen setzt auf einen Aufbau mittels IaaS auf.

4.2.3 Aufbau mit IaaS

Durch den Einsatz eines IaaS wird nun der Datenbankserver in die Cloud verschoben. Das Unternehmen muss sich nur noch um die Datenaufarbeitung und einem lokalen Testsystem, wenn es ein lokales Testsystem wünscht, kümmern. Die Verlagerung benötigt zudem nun strukturelles Umdenken. Die Erreichbarkeit des Datenbanksystems muss nun, unter der Berücksichtigung der Datensicherheit, für das System jederzeit erreichbar sein. Das heißt, es wird eine Abhängigkeit zum IaaS-Anbieter gebildet.

Vorteile

Kostenreduktion
Skalierbarkeit

„unbegrenzte“ Ressourcenausnutzung

Nachteile

Abhängigkeit zum Cloudbetreiber
Höhere Responsezeiten
zwischen Webshopanwendung und Datenbasis
Datenschutz
nur begrenzt Handlungsfähig bei Fehlerfällen

Tabelle 4.2: Vor-/Nachteile IaaS

In der Klassenstruktur des IaaS ändert sich das Ansprechverhalten des Datenbankdienstes. Die Kommunikation zwischen den Webserver- und Businessserver-Anwendungen mit der Datenbankanwendung müssen auf Schnittstellen definiert werden, da der Anwendungsteil der Datenbank mit in die Cloud transferiert wurde. Für die Business und Webanwendungen werden Datenstrukturen und Funktionsbeschreibungen bereitgestellt, um mit der Cloud-Datenbank zu kommunizieren.

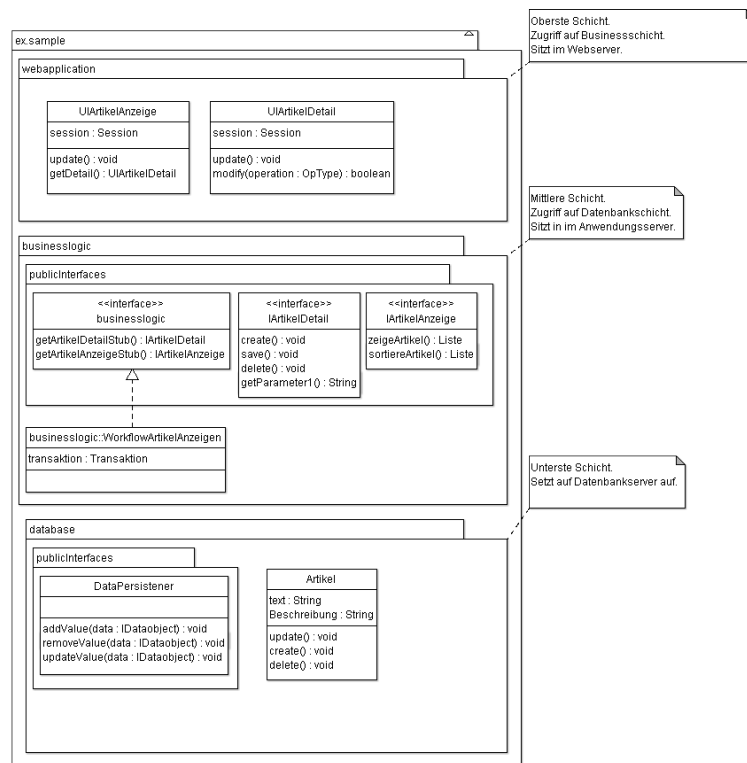


Abb. 4.2: Klassendiagramm Webshop Inhouse

Im Beispiel einer Webservicespezifikation werden grundlegende Klassen spezifiziert. Das heißt, spezifische Datenstrukturen, aber auch Methoden zur Verarbeitung, müssen zur Verfügung gestellt werden. In Abbildung 4.4 auf Seite 15 ist ein separates Package entstanden, dass auf dem IaaS eingebunden wird. Dieses Package stellt einen Webservice zur Verfügung, welches von der Businesslogik im Anwendungsserver konsumiert wird. Grundlegende Technik der entfernten Kommunikation ist das Anbinden von Interfaces. Diese Anbindungen können zum Beispiel durch RMI oder durch WSDL-Webservices mit Axis umgesetzt werden. Allgemein ist dies als Webservice definiert.

Im nächsten Schritt wird zu dem extrahieren der Datenbank auch die Webanwendung in die Cloud verschoben. Das heißt, die Cloud erweitert sich zu einer Platform as a Service

4.2.4 Aufbau mit PaaS

In der Platform as a Service befinden sich nun, wie in Abbildung 4.5 auf Seite 16 zu sehen, der Datenbankserver und der Webserver. Die Anwendung an sich bleibt beim Softwarehersteller/anbieter, das heißt, der Anbieter kümmert sich nur um die anwendungsspezifischen Lösungen, darunter unter anderem das Definieren und Umsetzen von Workflows. Der Anbieter muss sich zudem nicht

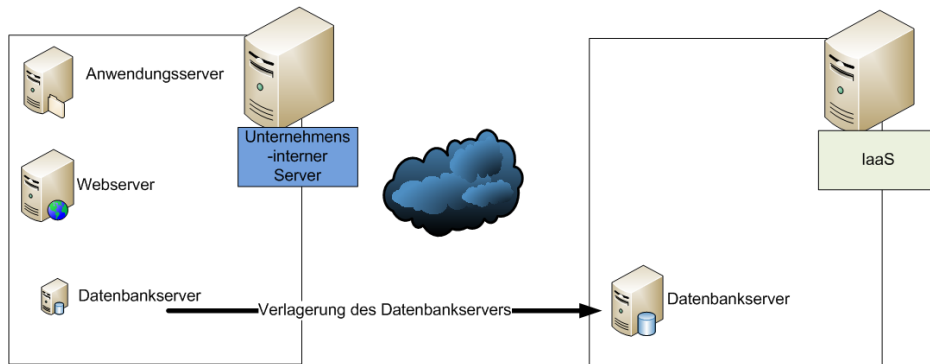


Abb. 4.3: Aufbau Webshop mit IaaS

mehr um datenbank- und webspezifische Probleme kümmern und kann diese an den Cloudbetreiber abgeben. Dieser kümmert sich um die Ressourcenbereitstellung und die Verknüpfung zwischen den IaaS und dem PaaS. Ziel dieses Aufbaues ist dabei hauptsächlich die Last effizienter auszunutzen. Wie schon erwähnt, entlastet dies nun den Softwareanbieter der sich um sein Hauptgeschäft, die Anwendungsproblematik, kümmern kann, ungemein. Nach Abbildung 4.6 auf Seite 17 ist nun folgender Sachverhalt dargestellt. Die Datenbank ist in die Cloud ausgelagert worden und kann mittels Webservicetechniken zugegriffen werden. Die Datenbankanwendung hat die Möglichkeit über einen Servicepoint Ressourcen anzufordern, die an das PaaS weiterdelegiert werden können. Zudem Datenbankserver ist auch der Webserver in der Cloud beherbergt und greift auf die Services des Inhouse-gehosteten Anwendungsserver zu. Auch der Webserver kann wie der Datenbankserver auf die PaaS Service Points zugreifen um Ressourcen anzufordern. Die Möglichkeit auf noch weitere Softwarekomponenten ist in diesem Beispiel nicht gegeben, da der Anbieter dies nicht zur Verfügung stellt. Der Aufbau einer PaaS-orientierten Lösung besteht somit aus verschiedenen Schnittstellen. Zum einen die Schnittstellen des eigenen Anwendungsprojektes, zum anderen das Zurückgreifen auf weitere Schnittstellen von anderen Komponenten, sowohl von der PaaS zur Verfügung gestellten, wie auch von anderen. Es ergibt sich sozusagen eine Verzahnung der Anwendungsteile unabhängig von deren tatsächlichem Standort.

Vorteile

- Kostenreduktion
- Skalierbarkeit
- geringere Responsezeiten
- „unbegrenzte“ Ressourcenausnutzung
- kompaktere Softwarelösung
- Absicherung der Daten durch anbieter
- kaum Wartungsarbeiten

Nachteile

- Abhängigkeit zum Cloudbetreiber
- Komplexitätssteigerung zwischen Webshopanwendung und Datenbasis
- Datenschutz
- nur begrenzt Handlungsfähig bei Fehlerfällen
- Verlust von Fachwissenden (Beratungskosten)

Tabelle 4.3: Vor- und Nachteile PaaS

Für kritische Prozesse, die einen sehr hohen Ressourcenbedarf benötigen, bietet das PaaS ein stimmiges Konzept um diese zu hosten. Bei Lastzeiten kann das PaaS zudem sehr variabel reagieren und entsprechen der Anforderungen diesen entgegensetzen. Dies erleichtert den reibungslosen Betrieb

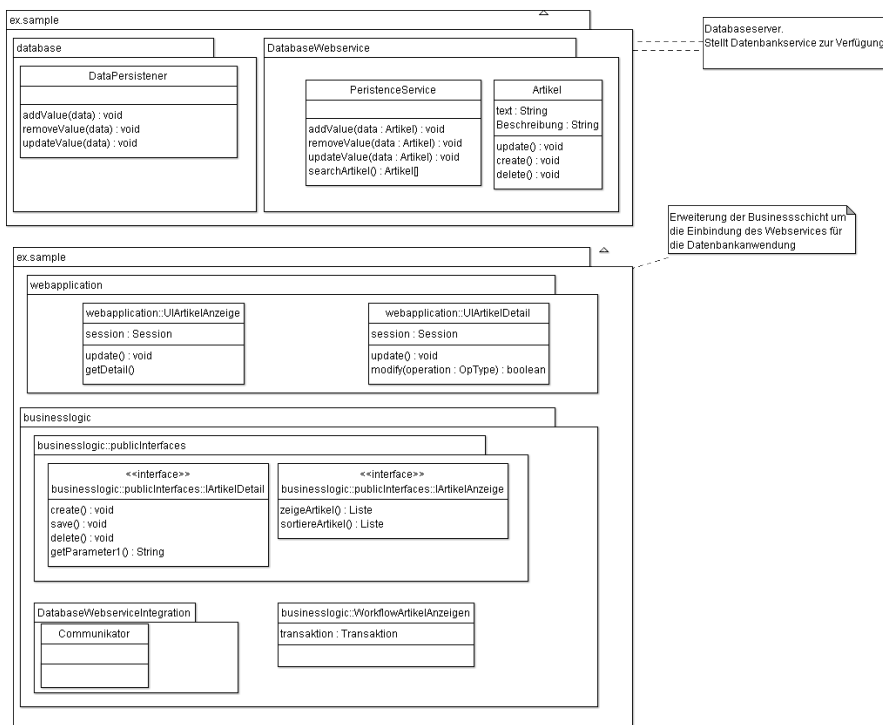


Abb. 4.4: Klassendiagramm nach IaaS

ungemein und entlastet den Softwareanbieter obendrein an administrativem Aufwand. Jedoch sind die Komplexitätssteigerungen, der Datenschutz und die Abhängigkeit von der Cloud beziehungsweise vom Cloudbetreiber nicht zu verachten. Alles in allem aber bietet dieses Konzept ein sehr gutes Angebot, Softwarelösungen so zur Verfügung zu stellen, dass dieses nahezu perfekt verwendet werden können.

4.3 Beispielhafte PaaS-Ergänzung durch SaaS

PaaS selbst ist zunächst nur die Grundlage für eigene Anwendungen. Bei der Entwicklung von cloud-basierten Anwendungen stellt sich jedoch neben der Auslagerung einzelner Komponente die Möglichkeit, gezielt auf SaaS-Leistungen zurückzugreifen. Dies wurde im vorherigen Kapitel bereits gezeigt. Dabei erfolgte die Betrachtung zunächst technisch abstrakt, die der Aufbau einer eigenen Architektur auf PaaS-Ebene ebensowenig standardisiert ist wie die PaaS-Leistungen verschiedener Anbieter selbst. Hier soll dies nun dahingehend konkretisiert werden, dass die Integration eines SaaS-Dienstes am Beispiel von Google AppEngine dargestellt wird.

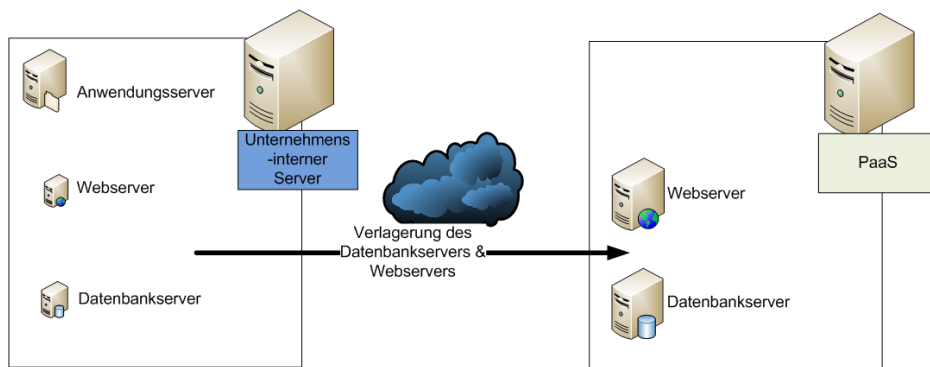


Abb. 4.5: Aufbau Webshop mit PaaS

4.3.1 Aufgabe Anwendungsfall

In diesem Beispiel wird ein Gästebuch-Website erstellt. Die Datensicherung findet in einer Datenbank statt, die für den einfachen Fall, vom Entwickler bereitgestellt und konfiguriert wird. In dem zur Verfügung gestellten Datenbankschema wird eine Entity-Relationship-Modell nach Abbildung 4.7 auf Seite 18 verwendet.

Das Gästebuch setzt auf eine simple Rechteverwaltung mit vielen verschiedenen Benutzern auf. Hierzu ist jeder Benutzer, Tabelle User, mit einer definierten Rolle versehen. Diese ermöglicht es einen Gästebucheintrag, assoziiert mit DB-Tabelle Article, einzufügen. Ein Gästebucheintrag kann vom erstellten Benutzer angelegt, gelöscht oder upgedated werden, hierzu wird die Metainformation „updatedate“ gesetzt, beziehungsweise „creationdate“ bei der Erstellung.

Die Administration und Pflege wird wie oben erwähnt durch den Entwickler verwaltet.

4.3.2 Aufgabenrealisierung mit Tomcat und MySQL

Zum Vergleich verschiedener Architekturen soll im folgenden zunächst der Aufbau der Applikation in einer konventionellen Javaumgebung dargestellt werden. Der Apache Tomcat und die Datenbank MySQL sind für diesen Anwendungsfall bestens geeignet und durch den Entwickler bereitgestellt und entsprechend der Anforderungen konfiguriert worden.

Die Komponentenlogik basiert auf dem UML-Klassendiagramm aus Abbildung 4.8 auf Seite 18. Wesentliche Klassen sind dabei die Klassen „GuestbookView“, „Guestbookmanager“, den Pojo-Klassen „Article“ und „User“. Die GuestbookView beschreibt die Benutzeroberfläche und handelt alle Aktionen mit dem Benutzer. Der Benutzer wird durch die Klasse User abgebildet, in der dieser eine Berechtigung (Role) zugeteilt bekommt. Der Benutzer kann auf der GuestbookView mit Hilfe des Views einen Article, der einen Guestbookeintrag widerspiegelt, anlegen, beziehungsweise updaten oder wieder löschen.

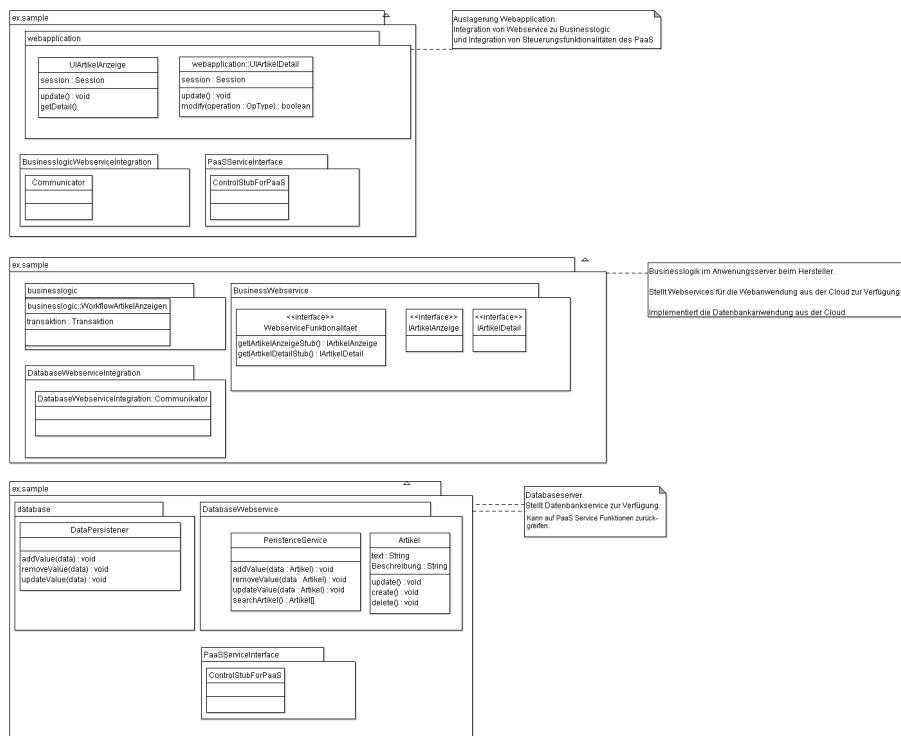


Abb. 4.6: Aufbau Webshop mit PaaS

Am Beispiel „Anlegen eines neuen Gästebucheintrages“ wird mit Hilfe der Abbildung 4.9 auf Seite 19 dieser Prozess im Detail aufgezeigt.

Ziel des beschriebenen Prozesses ist es, die Abhängigkeiten aufzuzeigen, dabei wird ersichtlich, dass für das Handeln der Datenbankverbindung eine separate Klasse zur Verfügung gestellt wird, die den Aufbau zur DB und die entsprechenden Statements bearbeitet. Mit Hilfe des Persistence-manger wird eine abstrakte Klasse als Möglichkeit realisiert, die eine Erleichterung für den Entwickler bereitstellt, vgl. Hibernate/Java Persistence.

Schlüsselstelle dieser Beispielanwendung sind die Datenbankabhängigkeiten und Benutzerverwaltung, die sehr schnell an Administrationstätigkeiten zunimmt, wenn die Webseite weiter ausgebaut werden soll. Wie jedoch solche Programme einfacher gestaltet werden können wird im nachfolgenden Abschnitt am Beispiel der Google App Engine erläutert.

4.3.3 Google App Engine

Die Google App Engine ist ein Projekt der Google Inc. und hat das Ziel, Webanwendungen in einer Cloud abspielen zu lassen. Zudem stellt die Engine eine Entwicklungsumgebung mit Entwicklungswerkzeugen zur Verfügung. Der Service ist zur Zeit kostenlos, hat jedoch eine nur begrenzte

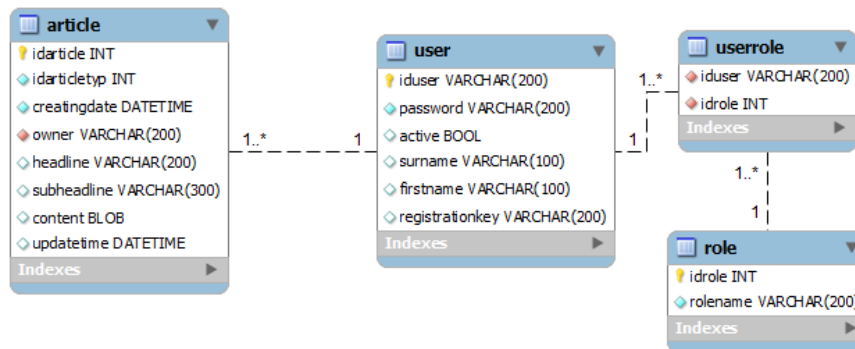


Abb. 4.7: ER Diagramm Guestbook

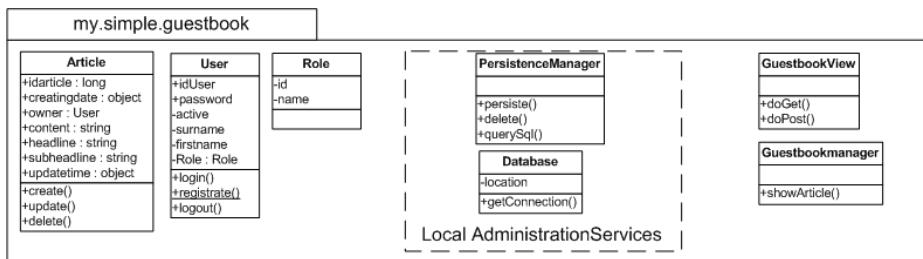


Abb. 4.8: UML-Klassendiagramm Guestbook und Benutzer

Anzahl verwendbarer Ressourcen. Das heißt, unter anderem das eine Seite maximal 7.400 mal pro Minute aufgerufen darf und der Ein/Ausgehende Datenfluss darf maximal ein Gigabyte pro Tag betragen. Das Prinzip besteht darin, einen Benutzeraccount bei Google zu erwerben und bei Überbeanspruchung der Ressourcen diese zuzukaufen.

Technologie

Die Engine kann in der Python und Java Sprache verwendet werden und bietet hierfür ein Standard Entwicklungs Framework. Für die einfache Handhabung des Systems stellt Google eine Ser-verengine zur Verfügung, die das Verhalten der Cloud widerspiegelt. Zudem werden Persistenz, Authentifizierung, URL Fetching, Mail, Memory Caching und diverse Bildmanipulationen zur Verfügung gestellt. Mit der Persistenz werden Query- und Transaktionssicherheits Modelle zur Verfügung gestellt, die ohne relationale Datenbankschemas auskommen (vgl. Google „BigTable“). Bei der Authentifizierung können basierend auf Google Accounts User mit Google-Konten an der selbsterstellten Anwendungen anmelden. Zudem ist über das URL Fetching das zugreifen auf die hochperformante Google-Infrastruktur insbesondere auf Internetinhalte möglich.

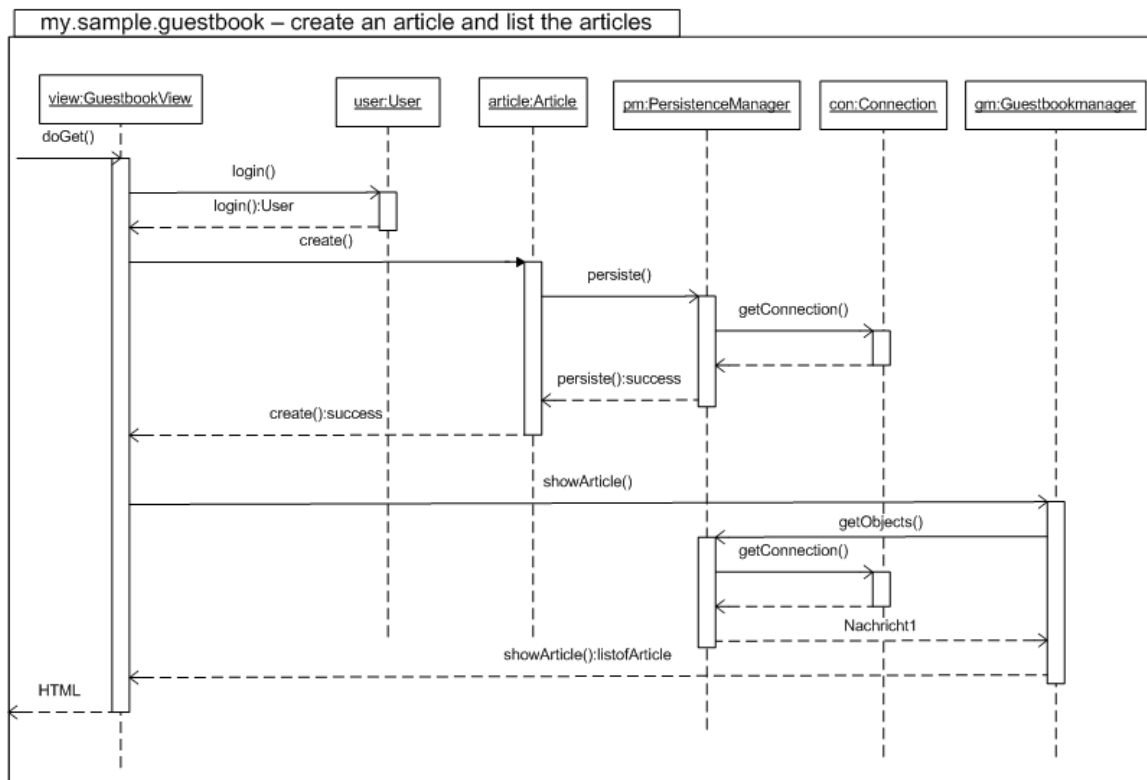


Abb. 4.9: Sequenzdiagramm Guestbook - Anlegen und Auflisten von Gästebucheinträgen

Aufbau Google App Engine - Lokal

Wie oben schon beschrieben stellt Google eine Standard Entwicklungs Framework zur Verfügung. Ein genauer Blick in das Java SDK der App Engine ermöglicht es, einen Blick in die Funktionsweise zu gewähren. Das Framework unterteilt sich in Scriptdateien zum starten eines Java Applikations-servers, der mit dem Google Server in der Funktionsweise vergleichbar ist. Für den lokalen Test wird der beliebte Apache Geronimo [Apa09] verwendet. Um den Server auszuführen wird lediglich in der Kommandozeile folgender Befehlskonstrukt benötigt. [h]

```

1 /appengine - java - sdk /bin /dev_appserver <Sourcepath of JEE Project >

```

Listing:Starten des Google App Servers

Der Server startet per Default auf 127.0.0.1 auf Port 8080 und kann über einen frei gewählten (aktu-ellen) Webbrowser über die URL `http://localhost:8080/<J2EE-Project>` aufgerufen werden.

Für die Datenbanksimulation wird eine objektorientierte Datenbank verwendet, da diese ohne rela-tionale Datenbankschemas auskommt. Der Zugriff auf die Datenbank ist für den Entwickler unin-teressant und wird nur durch die Google Java Funktionen gesteuert. Um ein Beispiel der Persistenz von Daten insbesondere der Datenbank zu verdeutlichen wird im nachfolgenden Abschnitt darauf

eingegangen.

Das Framework beinhaltet einige Java-Librarys die, wie oben schon erwähnt, Methoden zur Verfügung stellt um diverse Funktionalitäten zu verwenden. In einem separaten Ordner befindet sich dazu eine ausführliche Dokumentation, die nach dem JavaDoc (vgl. [SUN09a]) Prinzip erstellt wurde. Als Zusatz zu der Dokumentation werden Beispielanwendungen mitgeliefert, die als Referenz verwendet werden können.

Aufbau der Google App Engine Anwendung

Auch in diesem Fall, um eine vergleichbare Alternative zu bieten, wird ein Gästebuch entwickelt. Die Klassen wurden, wie Abbildung 4.10 zu sehen, leicht modifiziert. Durch die API von Google wird eine Benutzerverwaltung zur Verfügung gestellt wird, die auf der Basis der Googleaccounts aufbaut und eine sehr mächtige Benutzerverwaltung besitzt. Die Klasse „User“, im Diagramm grau markiert, ist eine von Google fest definierte Struktur und kann über den GoogleService „UserService“ abgefragt werden. Die Informationsbeschaffung wird durch das Framework verwaltet und bedingt keiner Anpassung von Entwicklerseite aus.

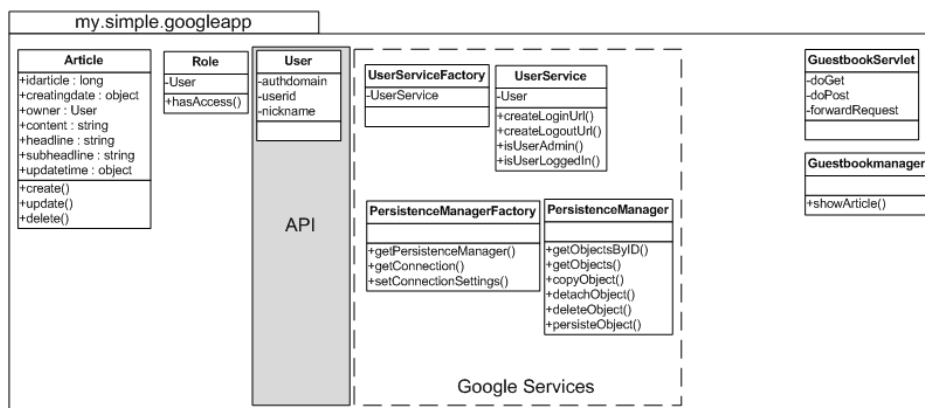


Abb. 4.10: UML-Klassendiagramm Guestbook mit Hilfe der Google Apps Engine

Auch hier wird wie im obigen Beispiel das anlegen eines Gästebucheintrags an einem Sequenzdiagramm, siehe Abbildung 4.11 auf der nächsten Seite, beschrieben. Die Besonderheit in diesem Fall sind die Services von Google, die, wie schon erwähnt, unabhängig vom Entwickler gehandelt werden und nur durch ServiceFactorys initialisiert werden, vgl. einer Implementierung eines Webservices mit dem axis-Framework for Java.

Datenspeicherung

Die Datenspeicherung in einer Google App Engine Anwendung wird mit Hilfe von Annotation [SUN09b] gesteuert. Als Beispiel wird hierfür ein Gästebuch entwickelt. Das Gästebuch besteht aus

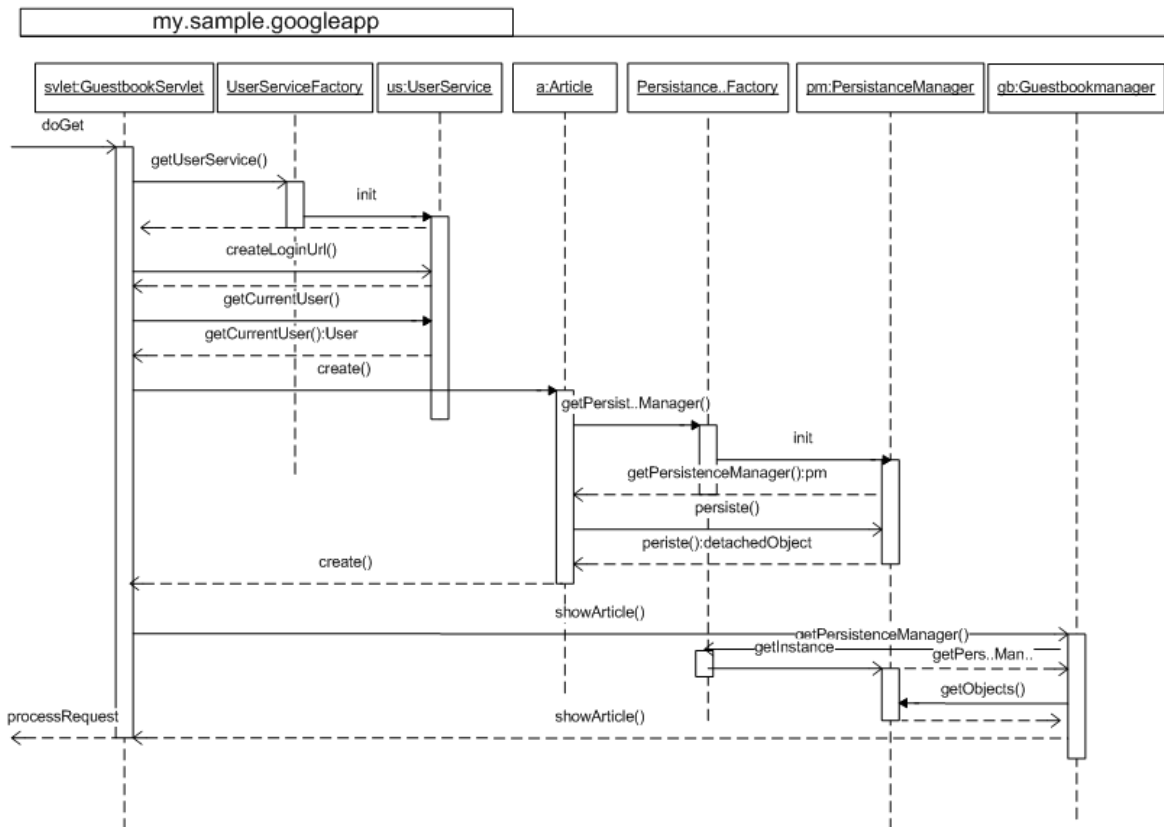


Abb. 4.11: Google Apps Engine Sequenzdiagramm Guestbook - Anlegen und zeigen von Gästebucheinträgen

einer beliebigen Anzahl von Grußbotschaften. Eine Botschaft kann sowohl Anonym wie auch durch das verwenden eines Google Accounts eingetragen werden. Eine Grußbotschaft wird mit der Klasse Greeting abgebildet. Greeting besteht aus einer eindeutigen Nummer (id) einem Benutzer (author) der eigentlichen Botschaft (content) und einem gesendeten Datum (creationdate).

```

1  @PersistenceCapable(identityType=
2      IdentityType.APPLICATION)
3  public class Greeting {
4      @PrimaryKey
5      @Persistent(valueStrategy=
6          IdGeneratorStrategy.IDENTITY)
7      private Long id;
8
9      @Persistent
10     private User author;
11
12     @Persistent
13     private String content;
14
15     @Persistent
    
```

```
16     private Date creationdate;
17     ...
```

Listing:Die Klasse Greeting

In obigem Listing werden die Datenfelder mit der Anotation „@Persistent“ markiert. Das ID-Feld wird zudem mit der Eigenschaft „@PrimaryKey“ und über „IdGenerationStrategy.IDENTITY“ als durch die Datenbank generierte ID definiert. Die Zuordnung und das Handling der Datenfelder wird damit für den Entwickler out of scope behandelt. Lediglich über einen Persistence Manager muss das Datenobjekt, hier Greeting, gespeichert werden. In diesem Beispiel wird dies durch das Abschicken einer Grußbotschaft an den Server zur Verarbeitung dargestellt, siehe Listing::Speichern von Daten mit Hilfe der Persistencefunktionalität.

```
1     ...
2     PersistenceManager pm=PMF.get().getPersistenceManager();
3     try {
4         pm.makePersistent(greeting);
5     } finally {
6         pm.close();
7     }
8     ...
```

Listing:Speichern von Daten mit Hilfe der Persistencefunktionalität

Das Auslesen der Daten ist vergleichbar mit dem obigen Beispiel nur wird hierfür eine Queryobjekt verwendet.

Zugriff auf Google Accounts

Das Verwenden von Goole Accounts ist eine sehr hilfreiche Anwendungsmöglichkeit, wenn man keine eigenen Benutzerverwaltung erstellen möchte. Hierfür bietet Google eine Schnittstelle, die UserService, an. In einer Google App Engine ist die Klasse User definiert, die alle Informationen über einen Benutzer, der bei Google registriert ist, zur Verfügung stellt. Der Zugriff auf den Informationsservice ist durch eine User Service Factory möglich, die eine Schnittstelle zu den Google Accounts bereitstellt.

```
1     UserService userService=UserServiceFactory.getUserService();
2     User user = userService.getCurrentUser();
3     if (user != null) {
4         \\\\...
5     } else {
6         resp.sendRedirect(userService.createLoginURL(req.
7             getRequestURI()));
8     }
```

Listing: Einloggen mithilfe des Google User Services

Mit Hilfe des User Services wird das Login auf das Minimum reduziert, man kann sich nun mehr auf die Webanwendung konzentrieren.

Google App Engine - Eclipse Plugin

Für die Entwicklung einer Webanwendung bietet Google ein Plugin (Abbildung 4.12) für die Entwicklungsumgebung Eclipse [Fou09] an. Durch dieses Plugin können dadurch komplexere Projekte mit Hilfe der Vorgaben von Google umgesetzt werden. Es kann auf dem SDK-Webserver der eigene Programmcode debugt werden. Des Weiteren kann direkt aus der Entwicklungsumgebung das Webprojekt an die Google Cloud übertragen werden.

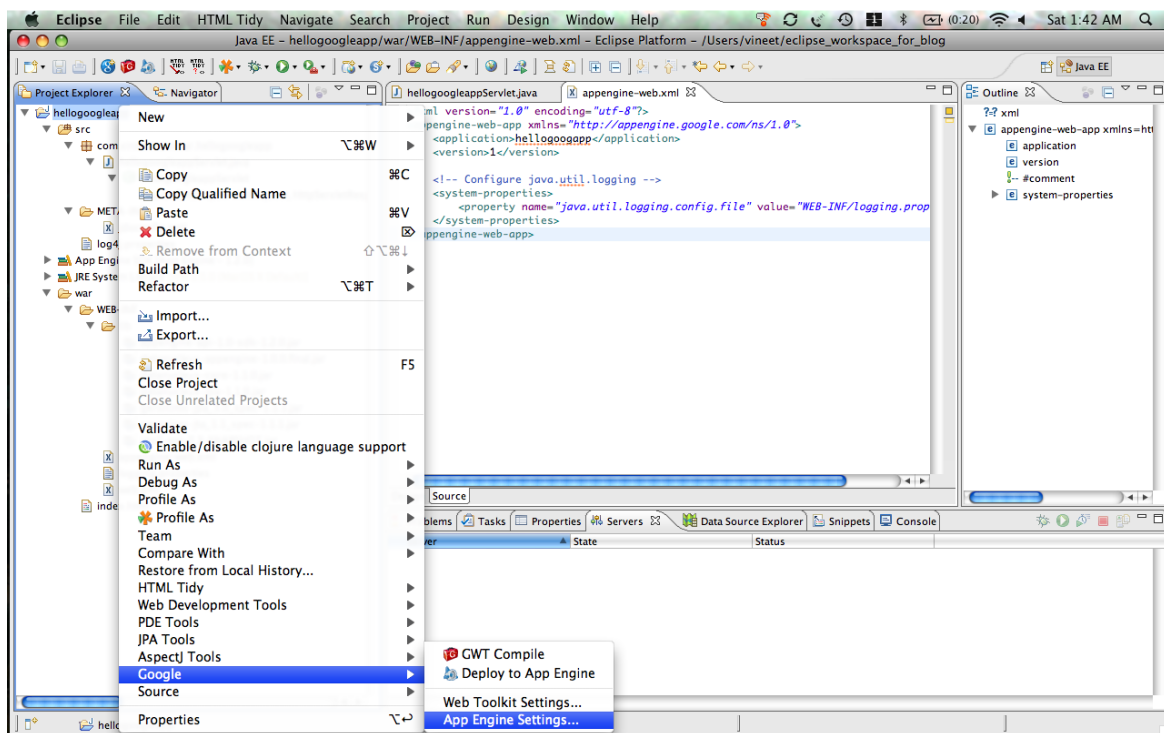


Abb. 4.12: Google App Engine Plugin für Eclipse

Publishing einer Anwendung

Um eine Webanwendung zu Veröffentlichen ist es nötig, erstens, einen Google Account zu besitzen/erstellen und zweitens, eine eindeutige Applikationsid zu erstellen/erzeugen. Die Applikationsid ist eindeutig und existiert nur einmal und ermöglicht die Anwendung bei Google zu registrieren und hochzuladen. Der Uploadvorgang wird durch das Framework selbst unterstützt. Um

nun bei Google auf die Anwendung zuzugreifen, bietet Google zwei Konzepte an. Zum einen kann eine Anwendung an eine Top-Leveldomains gebunden werden. Zum anderen kann durch einen kostenlosen Service von Google an der Domäne <http://appspot.com> eine Subdomäne verwendet werden. Die Subdomäne ist dabei die Applikationsid der Anwendung.

4.3.4 Amazon Web services



Abb. 4.13: Vergleich Google und Amazon

Im Vergleich zur Google App Engine sind die Amazon Web Services eine Sammlung von Webservices. Daraus lässt sich in etwa folgender Vergleich in Abbildung 4.13 schließen.

Zwischen der Googles App Engine und den Amazons Web Services (siehe Abbildung 4.13) wird deutlich, dass das Verständnis für PaaS sehr unterschiedlich umgesetzt werden kann. Amazon bietet eher ein Imagehosting mit Zusatzfunktionen wie zum Beispiel die verschiedenen Webservices an. Google dagegen versteht sich, Programmcode beziehungsweise Google Projekte zu hosten.

4.4 Andere Definitionen

Platform as a Service ist, wie aus Abschnitt ?? auf Seite ?? bereits bekannt, eine hierarchische Einordnung in das Modell nach Abbildung ?? auf Seite ?. Diese Plattform setzt auf der Infrastruktur auf und fügt ihr ein Softwarepaket hinzu. In einem solchen Softwarepaket werden Server zur Verfügung gestellt. Diese Server können unter anderem Applikationsserver oder sonstige weitere Funktionsserver, wie zum Beispiel einen Datenbankserver, Mailserver, sein. In der Plattform sind zudem verschiedene Prozesse definiert. Nach [BIT09] wird die Plattform als „Services auf der Anwendungs-Infrastruktur-Ebene (Middleware-Layer)“ definiert. Dabei ist die Basis ein technisches Framework, auch Entwicklungs-Plattformen.

Diese Definition wird in der Literatur jedoch nicht durchweg geteilt. Insbesondere im angelsächsischen Raum finden sich verschiedene weitere, zumeist weit weniger differenzierte, Nutzungen des Begriffs. Eine Platform as a Service erscheint dabei viel mächtiger. Ziel einer Paas sind dann einige wenige ausschlaggebende Kriterien: Development, deployment, Integration, Design, Storage, und Operation.

Hierzu folgende Definitionen der einzelnen Aspekte nach [Lin09]:

Deployment The ability to test, bundle, and deliver the PaaS created applications. This means hosting the applications, typically accessing them visually, through a browser, or as Web services.

Integration The ability to integrate the applications developed on your PaaS provider, with SaaS applications, or applications that may exist within your enterprise.

Design The ability to design your application and user interfaces.

Storage The ability to provide persistence for the application, meaning an on-demand database or on-demand file storage.

Operations The ability to run the applications over a long period of time, dealing with backup, restore, exception handling, and other things that add value to operations.

Des Weiteren sei auf die auf die Artikel [Bar09], [tec09] und [Mit09] verwiesen.

Als Resume dieser Informationen lässt sich folgende alternative Definition daraus schließen: Als PaaS versteht man die Bereitstellung einer Systemplattform für die Entwicklung einer Softwarelösung. Die Kosten, das Verständnis über die Komplexität und das managen der Systemressourcen wird dabei gezielt minimiert. Softwarelösungen aus den Bereichen Webanwendungen und Services, in Verbindung mit dem Internet, sind das Ziel von PaaS. Das PaaS soll zudem Workflows in den Bereichen Softwaredesign, Entwicklung, Testen und das Veröffentlichen der Lösung erleichtern. Das Hosten der Anwendung wird zudem in die Wolke, zum Beispiel durch einen anderen, fachspezifischeren Lieferanten, übertragen. Zudem können durch die Bereitstellung eines Frameworks eines Lieferanten verschiedene Bereiche, zum Beispiel die Datensicherheit, Schnittstellen zu anderen Systemen, bereitgestellt werden und sehr einfach in die Anwendung integriert werden.

Ein weiterer möglicher Definitionsansatz ist mehr clientbasiert. Hierfür kann PaaS als kosteneffiziente, cloudbasierte Anwendung betrachtet werden, die zum Beispiel verschiedene Datenverarbeitungsschritte implementiert. Diese werden dann durch den Kunden an dessen Endsystem angebunden. Es werden dadurch Produkte möglich, die sich in der freien Marktwirtschaft sonst nicht eigenständig tragen könnten.

Zusammenfassend läßt sich damit festhalten, dass bei allen Dokumente darauf geachtet werden sollte, wie die Begriffe des Cloud Computings verwendet werden. Die höchst unterschiedlichen Bedeutungen der selben Begrifflichkeiten kann schnell zu Missverständnissen führen.

Kapitel 5

Software as a Service (SaaS)

von Osman Aksoy

Software as a Service (SaaS) ist die allgemein verbreitete Form von Cloud Diensten. Mit SaaS erreicht eine einzige Applikation an tausende von Nutzern von den Servern des Softwarelieferanten. Die Nutzer greifen auf die Applikation über einen API zu. IBM hat folgende Definierung über SaaS veröffentlicht;

„Software as a Service (SaaS) ist ein neues, schnell wachsendes und zukunftsweisendes Geschäftsmodell für Softwarehersteller (Independent Software Vendors, kurz ISVs). Über das SaaS-Modell können Sie als ISV Ihren Kunden die eigens entwickelten Anwendungen auf Mietbasis zur Verfügung stellen. Ihre Kunden erhalten so eine kostengünstige Gesamtlösung bequem über das Internet und sparen Anschaffungs- und Lizenzkosten“

In der Praxis wird SaaS häufig als „Mietsoftware“ bezeichnet, wobei diese Bezeichnung nicht passt. Der Kunde mietet keine Software, vielmehr bezieht er einen Anwendungsservice mit allen Eigenschaften, die ein Service bietet:

- Abnahme nach Bedarf
- leichte Erweiterbarkeit
- Bezahlung nach Abnahmemenge.

Dieser Service beinhaltet alle für die Nutzung notwendigen Komponenten:

- Hard- und Software (Lizenzen)
- Wartung und Betrieb.

Die Anbindung an den Dienstleister sowie alle kundenseitigen Komponenten und Aufwände sind nicht enthalten. Anders als beim Application Service Providing bietet der Dienstleister beim SaaS-Modell nicht für jeden Kunden eine eigene Installation an. Hier nutzen alle Kunden dieselbe Anwendung und Infrastruktur, die sich bei einem Dienstleister befindet(siehe [BIT09]).

Das SaaS Konzept ist vergleichbar mit einer Bank, die die Privatsphäre jedes Kunden schützt und

auf breiter Ebene einen zuverlässigen und sicheren Service bietet. Die Kunden der Bank nutzen die Selben Finanzsysteme und Technologien, ohne zu befürchten, dass Unbefugte auf ihre Daten zugreifen.

Mandantenfähige Architektur: Die mandantfähige Architektur hat die Eigenschaft, dass sich alle Benutzer und Anwendungen eine gemeinsame Infrastruktur und Codebasis teilen. Diese gemeinsame Infrastruktur und Codebasis werden zentral verwaltet und gewartet. Die Benutzer sparen so Zeit, die bisher für die Wartung der veralteten Codes aufgebracht werden musste.

Müheleose Anpassung: Alle Benutzer können Anwendungen auf eigene individuelle Geschäftsprozesse anpassen, ohne dabei die gemeinsame Infrastruktur zu beeinträchtigen. Die Architektur der SaaS ermöglicht, dass diese Anpassungen durch Aktualisierungen auch beibehalten bleiben. Somit können SaaS-Anbieter auch mit geringen Kosten und Risikofrei häufiger Updates durchführen.

5.1 SaaS Vorgehensweise

Beim regulären „Kauf“ einer Software erwirbt man normalerweise nicht die Software selbst, sondern lediglich die Lizenz zur Nutzung der Software. Der Nutzer ist dafür verantwortlich, die für den Betrieb der Software notwendige EDV-Umgebung bereitzustellen und zu warten. Für weitere Installationen, Updates oder die Anschaffung neuer Hardware ist er ebenfalls zuständig.

Bei SaaS Produkten ist dies nicht der Fall. Hier wird die Software direkt bei einem Dienstleister betrieben, der Nutzer benötigt dabei lediglich eine minimale IT-Infrastruktur vor Ort. Dies erspart dem Kunden in der Regel Investitionen in speziell geschultes IT-Personal und in den Erhalt des zum Betrieb der Infrastrukturnotwendigen Know-hows. Der SaaS Dienstleister ist bei diesem Modell verpflichtet, die Software in der vertraglich vereinbarten Infrastruktur zu betreiben und die entsprechende Verfügbarkeit der Applikation sicherzustellen.

Damit verbundene Risiken und Aufwendungen, wie z. B. Hardwareausfälle, Wartung und Datensicherung, entfallen somit auch auf den Dienstleister. Im Gegensatz zum Outsourcing bestimmt jedoch der Anbieter, und nicht der Kunde, den entsprechenden Servicelevel, also den Umfang der vertraglich vereinbarten Leistungen. Der Zugriff auf die SaaS Lösung erfolgt in der Regel über das Internet mittels eines handelsüblichen Webbrowsers. Dies hat den Vorteil, dass der Zugriff auf die SaaS Anwendung ortsunabhängig, von nahezu jedem PC mit Internetzugang aus, erfolgen kann. Gerade bei dezentralen Unternehmensstrukturen erweist sich dies als Vorteil. Bei im Vorfeld nicht bekannten oder stark schwankenden Nutzerzahlen zeichnen sich SaaS Produkte außerdem durch eine erhöhte Flexibilität aus.

SaaS Lösungen werden in der Regel für eine monatliche Pauschale angeboten. Die monatliche Pauschale deckt normalerweise alle anfallenden Kosten, also inkl. Wartung und Upgrades, ab. Manche Produkte werden jedoch auch in Abhängigkeit von der Nutzung („Pay-per-Use“) berechnet, was sich vor allem für Gelegenheitsnutzer auszahlen kann. In beiden Fällen bietet die SaaS Lösung eine

erhöhte Kostentransparenz im Vergleich zu vielen herkömmlichen Softwarelösungen. Darüber hinaus besteht bei vielen Produkten die Möglichkeit, diese zunächst kostenlos für eine bestimmte Zeit zu testen, um vorab entscheiden zu können, ob diese den eigenen Anforderungen entsprechen. Auf der anderen Seite besteht bei SaaS Produkten während der Vertragslaufzeit eine verstärkte Abhängigkeit gegenüber dem SaaS Dienstleister und dem jeweiligen Internet-Provider, da deren Zuverlässigkeit ausschlaggebend für den Betrieb der SaaS Lösung ist. Fällt die Internetverbindung oder die IT-Infrastruktur des SaaS Dienstleisters aus, ist das jeweilige SaaS Produkt in der Regel nicht nutzbar. Besonders bei unternehmenskritischen Daten ist außerdem ein solides Vertrauensverhältnis zwischen Dienstleister und Kunde notwendig, da die Kundendaten nicht vor Ort, sondern direkt beim Dienstleistungsgespeichert werden. Gerade bei personenbezogenen Daten sollte die Datenspeicherung unter Achtung der jeweiligen Datenschutzbestimmungen erfolgen.

5.2 SaaS-Anwendungsbeispiele

Heutzutage gibt es eine sehr große Auswahl an SaaS Produkten. Nahezu alle namhaften Hersteller von betriebswirtschaftlicher Standardsoftware haben entsprechende Applikationen in ihrem Programm. Am populärsten sind dabei bisher IT-Landschaft integrieren lassen.

Zu nennen sind daher vor allem die Bereiche CRM (Customer-Relationship-Management) und SCM (Supply-Chain-Management), während komplexere Lösungen wie ERP (Enterprise-Resource-Planning) sich bisher geringerer Beliebtheit im SaaS Bereich erfreuten. Doch SaaS beschränkt sich keinesfalls nur auf betriebswirtschaftliche Standardsoftware. Auch andere, auf ein breiteres Publikum ausgelegte Dienste, wie z. B. webbasierte E-Mail Dienste wie Google Gmail oder auch Online Office-Applikationen wie Google Apps zählen zu den SaaS Anwendungen.

Allgemein kann man jedoch sagen, dass SaaS Anbieter sich bisher verstärkt darauf fokussieren, möglichst standardisierte Applikationen mit eingeschränkten Anpassungsmöglichkeiten anzubieten (siehe [Bor09]).

5.2.1 Salesforce CRM

Wie schon erwähnt, nehmen die SaaS Anbieter von Ihren Kunden den Last von IT-Gebundenen Investitionen ab. Somit können sich die Unternehmen auf Ihre Kerngeschäfte konzentrieren. CRM ist einer der wichtigsten Erfolgsfaktoren für ein Unternehmen, die natürlich erstens an mehreren IT-Diensten verbunden ist, zweitens durch die enorme Konkurrenz eine Schnelligkeit erfordert.

Daher ist Salesforce CRM besonders bei kleinen und mittelständischen Unternehmen sehr schnell beliebt geworden, denn es ist kostensparend, schnell einsetzbar, unterstützt viele CRM-Prozesse und ist leicht bedienbar. Mit Salesforce CRM kann man Kunden, Kampagnen, Aufträge verwalten, Analysen durchführen und den Kunden auf verschiedene Kommunikationswege Support anbieten.

Die Benutzeroberfläche lässt sich auch nach Bedürfnissen leicht anpassen.



Abb. 5.1: Screenshot der Oberfläche von Salesforce im Browser

Durch den Einsatz von Salesforce CRM erreichen die Unternehmen eine hohe Produktivität in dem sie sich auf Ihre Kerngeschäfte konzentrieren. Mit knapp 70000 Kunden verspricht Salesforce CRM den Interessenten Kompetenz und Vertrauen.

Salesforce CRM bietet Dienste in einem breiten Spektrum.

Vertrieb Lösungen aus dem Bereich CRM Software-as-a-Service (SaaS CRM) ermöglichen es Unternehmen, die Produktivität des Vertriebs anzukurbeln, mehr Transparenz zu erzielen und den Umsatz auszubauen.

Kundenservice und Kundensupport Mit einer CRM-Lösung steht Unternehmen innerhalb weniger Wochen eine bei Mitarbeitern äußerst beliebte Anwendung für Callcenter und den Selbstbedienungsbereich für Kunden zur Verfügung, mit der die Kundentreue neue Dimensionen erreicht.

Partner Relationship Management CRM on demand ermöglicht es Partnern, mühelos auf Leads zuzugreifen, bei Geschäften zusammenzuarbeiten und all die Informationen ausfindig zu machen, die sie benötigen, um erfolgreich zu sein. Endlich haben Unternehmen eine einheitliche Ansicht der Vertriebskette, sowohl für direkte als auch indirekte Vertriebskanäle.

Marketing Mithilfe von CRM Software-as-a-Service (SaaS CRM) ist ein geschlossenes Marketing zur Ausführung, Verwaltung und Analyse von Kampagnen möglich, die über mehrere Vertriebswege laufen. Führungskräfte im Marketing können die Kapitalrendite ihrer Budgets



Abb. 5.2: Funktionsumfang von Salesforce CRM

messen, Einkünfte für spezielle Marketingprogramme zurücklegen und Anpassungen in Echtzeit vornehmen.

Inhalt Mit CRM on demand steht die Benutzerfreundlichkeit von Web 2.0 für Ihre Geschäftsinhalte zur Verfügung - so können Sie sie effektiver für die gemeinsame Nutzung freigeben und die Zusammenarbeit optimieren. Versetzen Sie Ihre Mitarbeiter in die Lage, genau die Daten zu finden, die sie benötigen - und zwar über die CRM Software-as-a-Service-Anwendung, mit der sie Tag für Tag arbeiten.

Ideen Dank den CRM-Lösungen von salesforce.com können Sie die geballte Leistungsfähigkeit Ihrer Community optimal nutzen und dynamisch mit Ihren Kunden, Partnern und Mitarbeitern interagieren. In Ihrer Online-Community können Mitglieder Beiträge veröffentlichen, sich austauschen und über Ideen abstimmen. Die besten Ideen kommen zum Vorschein und Sie können Feedback skalierbar einordnen und verwalten.

Analysen CRM Software-as-a-Service (SaaS CRM) bietet Kundenbeziehungsmanagement-Geschäftsanwendern auf allen Ebenen die Möglichkeit, relevante Einblicke zu gewinnen und Analysen durchzuführen. Mithilfe von Berichten in Echtzeit, Berechnungen und Dashboards können Unternehmen Ihre Leistungen verbessern sowie Entscheidungsfindungsprozesse und die Ressourceneinteilung optimieren.

Benutzerdefinierte Anwendungen Sobald Unternehmen mit Cloud Computing-CRM-Lösungen auf den Geschmack gekommen sind, möchten sie in der Regel über CRM Software-as-a-Service (SaaS CRM) hinaus auch noch andere Geschäftsbereiche in Angriff nehmen. Mit der Force.com-Plattform können Sie sowohl CRM on demand als auch alle Geschäftsanwendungen Ihres Unternehmens in einer einzigen Umgebung, mit einem Datenmodell, einem Freigabemodell und einer Benutzeroberfläche erstellen und ausliefern.

5.2.2 Siptgate Team

Siptgate bietet seit 2004 unter dieser SaaS, VOIP Dienste im Cloud für Geschäftskunden an. Die von Siptgate im Cloud angebotener VoIP Dienst, ist für mehrere Geschäftskunden ohne Installations- und Konfigurationsaufwand zugänglich. Als Benutzer von Siptgate Team können Unternehmen Ihre Telefonanlagen unkompliziert ins Internet verlagern und somit haben Sie die Möglichkeit die Telefonanlage bequem über Webbrowser zu steuern.



Abb. 5.3: Screenshot der Siptgate-Oberfläche im Browser

Somit sparen die Benutzer den Aufwand von teuren Telefonanlagen und Wartungsverträgen. Siptgate Team kalkuliert monatliche Pauschalpreise abhängig von der Mitarbeiteranzahl. Somit haben die Benutzer einen vollen Kostentransparenz und ein geringeres Ausfallrisiko im Vergleich zur klassischen Telefonanlagen.

Durch die Benutzeroberfläche von Siptgate kann man Benutzer anlegen und verwalten, Standorte anlegen und verwalten, Rufnummern an bestimmte Benutzer zuweisen oder Rufnummern sperren. Die Benutzer können auch Standortunabhängig Ihre Anrufe und Kontakte nachsehen und ggf. kontaktieren. Das ganze wäre sicherlich ein erheblicher Zeitaufwand bzw. nicht möglich bei klassischen Telefonanlagen.

In folgender Abbildung sind weitere Funktionen von Siptgate Team zu finden.



Abb. 5.4: Funktionsumfang von Sipgate Team

Kapitel 6

Unternehmerische und rechtliche Einordnung

von Christian Kolodziej

Nachdem nun aus technischer Sicht umfangreich behandelt wurde, was man sich unter der Cloud vorstellen muss und wo mögliche Einsatzgebiete liegen, geht es nun um die finanziellen, rechtlichen und qualitativen Aspekte des Cloud Computing. Was alles muss in den Vorstandsetagen besprochen und gegeneinander abgewogen werden, bevor eine solch weitreichende Entscheidung, die die bisherigen Strukturen der IT eines Unternehmens massiv verändert, getroffen werden kann? Welche Risiken und Potentiale sind speziell bei der geschäftlichen Nutzung von Cloud Computing zu bedenken? Und gibt es Unterschiede zwischen Kleinunternehmern, Mittelständlern und Großkonzernen?

Cloud Computing steht noch am Anfang der Marktdurchdringung, hat aber das Potenzial, „mittel- bis langfristig einen beträchtlichen Teil der traditionellen IT-Leistungsangebote zu ersetzen“¹. Insbesondere das Potenzial der Kostensenkung ist oft ausschlaggebend für die Entscheidung pro Cloud Computing. Als weitere Motive nennt der deutsche Branchenverband BITKOM in einem im Oktober 2009 veröffentlichten Leitfaden „die Verlagerung von Investitions- zu variablen Kosten, die schnelle Realisierbarkeit, eine größere Flexibilität und Skalierbarkeit der IT-Ressourcen sowie die nutzungsabhängige Bezahlung der IT-Services“². Die immer noch nicht vollständig überstandene Wirtschaftskrise ist sicherlich auch „ein Treiber für den [...] Einsatz im Unternehmen“³.

Den Kosteneinsparungen gegenüber stehen allerdings oft noch die Unternehmens-Compliance, moralische Vorbehalte sowie datenschutzrechtliche Probleme.

6.1 Geschäfts-, Nutzungs- und Abrechnungsmodelle

Die Tatsache, dass die IT-Infrastruktur eines Unternehmens immer auf Lastspitzen ausgelegt sein muss und außerhalb dieser Zeiten große Kapazitäten der Systeme nicht ausgelastet sind, hat tra-

¹Siehe [BIT09, Seite 13].

²Siehe [BIT09, Seite 13].

³Siehe auch [Gro09].

ditionell hohe Kosten zur Folge. Strategien, ressourcenfressende Routineaufgaben in lastschwache Zeitfenster zu legen oder bestimmte Hardware nur während Spitzenzeiten zuzuschalten, sind zwar erste Ansatzpunkte, die resultierenden finanziellen Entlastungen jedoch eher gering. Was unverändert blieb, waren die immer hohen Anschaffungs- und Fixkosten für Administration, Wartung und Betrieb. Den Spagat zwischen Flexibilität bei gleichzeitig geringen (Fix-)Kosten versucht nun Cloud Computing zu leisten. Dabei sind auch völlig neu Bezahl- und Abrechnungsmodelle möglich.

Um die derzeit üblichen Geschäfts- und Abrechnungsmodelle zu betrachten, ist eine Differenzierung des Begriffs Cloud Computing in seine 3 Schichten Infrastructure as a Service (IaaS), Platform as a Service (PaaS) und Software as a Service (SaaS) nötig. In allen Fällen werden grundsätzlich IT-Leistungen flexibel und bedarfsgerecht zur Verfügung gestellt, was auch Einfluss auf die Abrechnungsmodelle hat. Aus früheren Fixkosten werden durch das Cloud Computing nun variable Kosten. Allerdings entfallen in der Private Cloud noch viele der genannten Vorteile, da weiterhin Hardware beschaffen und betrieben werden muss. Erst in der Public Cloud, wenn auch die gesamte Hardware Eigentum des Dienstleisters ist, sind maximale Skaleneffekte zu erwarten und die Kosten komplett variabel. Auf die Unterschiede zwischen privater, öffentlicher und hybrider Cloud wird in Kapitel 6.4 auf Seite 42 noch detaillierter eingegangen.

6.1.1 Infrastructure as a Service (IaaS)

IaaS ist die Bereitstellung von Rechenleistung und Speicherplatz als „grundlegende Komponenten für Cloud-Computing-Ökosysteme“⁴ ⁵. Dabei wird die in Anspruch genommene Leistung – benötigter Speicherplatz oder verbrauchte Prozessorzeit – entweder nach dem Pay-per-Use- (auch *Pay as you go*) oder Abonnement-Verfahren abgerechnet. Oft steht ein gewisses Inklusiv-Volumen zur Verfügung (insbesondere bei Backup-Diensten, die auch den Privatnutzer adressieren) und nur darüber hinaus genutztes Volumen wird gesondert in Rechnung gestellt wird. Flatrates, wie sie insbesondere im DSL- und Telefonmarkt gang und gäbe sind, finde man bei Cloud-Computing-Angeboten nicht. Diese für Anwender wie Anbieter einfache Abrechnungsart basiert auf dem Prinzip der Mischkalkulation: einige Nutzer verbrauchen weniger als der Durchschnitt, andere Benutzer verbrauchen mehr, aber insgesamt reicht die zur Verfügung stehende Kapazität wie beispielsweise die Bandbreite immer für alle Nutzer. Beim Cloud Computing allerdings fehlt eine Begrenzung der Kapazität. Diese Elastizität des Services ist demnach auch der Grund warum sich Pauschaltarife und Cloud Computing zumindest momentan noch ausschließen⁶.

Bei Inanspruchnahme von IaaS-Diensten bleibt für den Benutzer in der Regel transparent, wo seine Daten gespeichert und seine Prozesse ausgeführt werden, was datenschutzrechtlich bedenklich ist und die Datensicherheit beeinflussen kann (mehr zu diesem Thema in Kapitel 6.5). Über zur Verfügung gestellte Schnittstellen oder Desktop-Programme werden dann Daten in die Cloud geschrieben beziehungsweise Rechenoperationen in der jeweiligen virtuellen Maschine ausgeführt.

⁴Als Ökosystem wird die Zusammenarbeit aller beteiligten Komponenten innerhalb einer Cloud bezeichnet.

⁵Siehe [Com09, Seite 458]

⁶Siehe auch [Bro09].

Die lokale Software-Lösung und der dahinter liegende SaaS-Service müssen dabei nicht zwangsläufig von ein und demselben Anbieter kommen, ganz im Sinne von Cloud Computing können die Anbieter wiederum die Angebote anderer Anbieter nutzen. So greift etwa der Dienst DropBox⁷ für die Datenspeicherung den Service Amazon S3 zurück, der sich mit hohen Sicherheitsstandards auch an Unternehmen mit hohen Anforderungen bezüglich der Datensicherheit richtet und bewährt hat.

Bei Amazon S3 wird jedoch nicht ausschließlich der genutzte Speicherplatz berechnet, die Summe erhöht sich zudem um Kosten für den ein- und ausgehenden Traffic sowie die Anzahl der Anfragen. Um eine grobe Vorstellung über die Konditionen zu bekommen, zeigt die Tabelle 6.1 die von Amazon berechneten Kosten.

Benutzter Speicherplatz	0,15 \$	pro GB-Monat
eingehender Traffic	0,10 \$	pro GB
ausgehender Traffic	0,18 \$	pro GB für die ersten 10 GB
	0,16 \$	pro GB zwischen 10 und 50 GB
	0,12 \$	pro GB über 50 GB
Anfragen	0,01 \$	für 1.000 PUT- und LIST-Requests
	0,01 \$	pro 10.000 GET-Requests

Tabelle 6.1: **Kosten bei der Nutzung des Amazon-Speicherdienstes S3**

Trotz immer weiter sinkender Hardware-Preise sind und bleiben die Angebote von IaaS-Anbietern für Unternehmen sehr interessant, da nicht mehr auf einen Schlag hohe Summen für Hardware-Investitionen anfallen, sondern die Kosten kontinuierlich und über die Zeit verteilt anfallen. Zudem entfallen Administrationskosten und auch defekte Hardware muss nicht mit mehr oder minder großem finanziellen Aufwand von Zeit zu Zeit ersetzt werden. Bei einem Backup-Volumen von 200 GB mit einer sich monatlich ändernden Datenmenge von 5 GB etwa fallen monatliche Kosten in Höhe von 30,90 \$ für die sichere Datensicherung an – verglichen mit den Angeboten dedizierter Webserver ein Schnäppchen wenn man noch dazu den entfallenen Administrationsaufwand berücksichtigt.

Neben der klassischen On-Demand-Abrechnung bietet Amazon EC2 auch noch die sogenannten Reserved Instances an. Dabei ist der Stundenpreis geringer, allerdings ist eine zusätzlich Einmalzahlung erforderlich⁸. Und zuletzt überraschte Amazon Anfang Dezember 2009 mit einem neuen und bisher einmaligem Preismodell mit der Bezeichnung EC2 Spot Instances⁹. Das neue Bezahlmodell basiert auf Angebot und Nachfrage der Kunden und reguliert sich somit selbst. Während dies für die Kunden niedrigere Preise bedeuten kann (aber nicht muss!) profitiert Amazon als Anbieter weil die Systeme besser ausgelastet werden¹⁰. Neben dem Maximalgebot gehen auch noch weitere Faktoren in die Bieterunde mit ein, darunter die EC2-Region sowie Art und Anzahl der gewünschten Instanzen. Da EC2 Spot Instances die Verfügbarkeit von Ressourcen zu bestimmten Zeitpunkten nicht garantieren kann, eignet sich dieses Modell vor allem für nicht zeitkritische Aufgaben.

⁷www.dropbox.net

⁸Siehe auch [Gol09].

⁹Siehe aws.amazon.com/ec2/spot-instances/.

¹⁰Siehe auch [New09].

Die Speicher- und Rechenkapazitäten von IaaS-Lösungen werden überdies eher selten direkt von Endnutzern genutzt, aber im Gegenzug umso mehr von Anbietern anderer Cloud-Computing-Dienste. In vielen Fällen sind die Speicher- und Rechenkapazitäten von IaaS-Lösungen demnach als Komponente in Dienste eingebettet, die den Schichten PaaS und SaaS zuzuordnen sind.

6.1.2 Platform as a Service (PaaS)

Eine Abstraktionsstufe höher sind die Anbieter angesiedelt, die Anwendungsentwicklern eine technische Plattform für die Entwicklung und den Betrieb von Anwendungen bieten. Die bei IaaS noch ausschlaggebenden Datenvolumina rücken in den Hintergrund, stattdessen interessieren jetzt die zur Verfügung stehenden Services, die je nach Art unterschiedlich abgerechnet werden müssen. PaaS-Dienste sind von allen drei Schichten am schwersten zu greifen, da man sich hier am schwersten vorstellen kann, was genau hier genau abgerechnet wird¹¹.

Die Marktführerschaft liegt aktuell zweifelsohne bei dem erst sieben Jahre alten Unternehmen salesforce.com. Auch wenn eher die gleichnamige SaaS-CRM-Lösung bekannt ist, bietet der Anbieter im Wesentlichen PaaS-Dienste an. Unter der Bezeichnung force.com werden zahlreiche Basisdienste für SaaS-Lösungen wie salesforce.com zur Verfügung gestellt. In den angebotenen Services und Entwicklerwerkzeugen wie die Webservice APIs, VisualForce für die GUI-Entwicklung oder Funktionen und Methoden zur Benutzerverwaltung, Datenmanagement, Workflowsteuerung oder Reporting sieht Marc Benioff, CEO und Gründer von salesforce.com, das Ende klassischer Software und verkündet gleichzeitig „die Ära des Cloud Computing auf Basis von PaaS“¹². salesforce.com bietet dabei nicht nur rund 41000 Kunden Kunden (davon knapp 7000 aus Europa) die Plattform für die Entwicklung eigener Software, sondern mit AppExchange auch noch einen Marktplatz, auf dem die fertigen SaaS-Lösungen von anderen Kunden problemlos eingebunden werden können. Auch wird dadurch die Nutzung von Datenobjekten über die Grenzen einzelner SaaS-Lösungen hinweg ermöglicht, insofern diese alle bei salesforce.com betrieben werden. So nutzt beispielsweise auch Google – seinerseits mit der GoogleAppEngine ebenfalls PaaS-Provider – salesforce.com in seinen Office-Anwendungen¹³. Als erste PaaS-Lösung wurde force.com von Gartner in den Application Server Magic Quadrant aufgenommen. Der Begriff „Magic Quadrant“ wurde 2005 von der Gartner Group urheberrechtlich geschützt und ist die „grafische Darstellung eines Marktes in einem bestimmten Zeitraum“ und veranschaulicht, wie sich bestimmte Anbieter nach von Gartner definierten Kriterien in diesem Markt positionieren¹⁴.

salesforce.com bietet neben einem kostenlosen Einsteiger-Paket „Force.com Free“ noch die kostenpflichtigen Pakete „Force.com Enterprise“ sowie „Force.com Unlimited“ für 50 € bzw. 75 € pro Benutzer und Monat an¹⁵. Die Pakete unterscheiden sich in den Punkten Anzahl Nutzer, zur Verfü-

¹¹Zu den technischen Diensten auf PaaS-Ebene siehe Kapitel ?? auf Seite ??

¹²Siehe auch [IR09].

¹³Siehe auch [IR09].

¹⁴Siehe auch [Gmb05].

¹⁵Siehe auch [sal09].

gung stehender Speicherplatz, Anzahl nutzbarer Applikationen, Anzahl Datenbank-Objekte sowie dem Support.

Andere Anbieter neben salesforce.com stehen nicht unbedingt in direkter Konkurrenz, sondern adressieren oftmals andere Zielgruppen. So ist die GoogleAppEngine auf Python-Anwendungen hin ausgerichtet, die Amazon-Angebote bedienen die LAMP-Community¹⁶ und Facebook zielt ganz auf die Privatanwender¹⁷. Entsprechend unterschiedlich gestalten sich auch die Preis- und Geschäftsmodelle. Aufgrund der Beschränktheit dieser Ausarbeitung kann an diese Stelle nicht detaillierter auf diese eingegangen werden. Allen Angeboten gemein ist aber, dass IaaS-Leistungen von PaaS-Angeboten gekapselt und von den PaaS-Gebühren abgedeckt werden und damit transparent im Hintergrund bleiben.

In der Praxis dienen PaaS-Dienste als Basis für zahlreiche SaaS-Anwendungen und werden auch diesem Grunde von Endnutzern in der Regel nur als solche wahrgenommen.

6.1.3 Software as a Service (SaaS)

Cloud Computing hat vor allem auch für Software-Anbieter weitreichende Konsequenzen, schließlich ist die Anwendungsschicht auch diejenige, mit denen die Endnutzer am meisten in Kontakt kommen. Das klassische Lizenzgeschäft von Software ist bei Software as a Service nicht mehr praktikabel, stattdessen muss es durch neue passende Abrechnungsalternative ersetzt werden. Neben der Programmierung der Software muss der Hersteller nun auch für eine adäquate Betreuung der Software über das Internet und die entsprechenden hardwaretechnischen Voraussetzungen sorgen. Nicht selten greifen SaaS-Anbieter selbst wieder auf PaaS- oder IaaS-Lösungen zurück, um flexibel auf die in den meisten Fällen nicht einfach einzuschätzende Nachfrage reagieren zu können.

Der Zugang für Benutzer ist bei der Nutzung von SaaS leichter, da ortsunabhängig, und problemlos mit diversen (auch mobilen) Endgeräten möglich. Zudem entfällt der Installationsaufwand sowie das regelmäßige Einspielen von Updates. Auch erwartet der Kunde, dass er sich schnell bei einem Dienst registrieren kann, um diesen direkt in vollem Umfang zu nutzen. Anstelle der früheren einmaligen Kosten beim Software-Erwerb, fallen nun regelmäßige Gebühren an, die an das Benutzerkonto gekoppelt sind. Flexibilität ist auch hier das Hauptargument, damit von Kundenseite schnell und unkompliziert auf sich ändernde Rahmenbedingungen reagiert werden kann. Bei Bedarf muss sich die Anzahl der Benutzerkonten leicht erweitern aber auch wieder reduzieren lassen.

Auch wenn bei der Nutzung von SaaS Daten anfallen, die anbieterseitig gespeichert werden müssen, werden Datenvolumen in der Regel nicht zusätzlich abgerechnet, sondern sind inklusive oder es besteht eine Limitierung. So werden beispielsweise innerhalb eines CRM-Systems oft viele Daten und Dokumente zu Kunden-Datensätzen gespeichert. Ausschlaggebend für die spätere Rechnung sind jedoch die Anzahl der Benutzerkonten sowie die freigeschalteten Features. Dabei lassen sich je

¹⁶LAMP = Linux, Apache, MySQL und PHP

¹⁷Siehe auch [IR09].

nach Rolle des Benutzers im Idealfall die zur Verfügung stehenden Funktionen anpassen. Anbieter tun gut daran, eine solche Flexibilität anzubieten, weil dies auch die Akzeptanz und Zufriedenheit der Kunden erhöht, die nicht für brach liegende Funktionalitäten zahlen müssen, wie dies beispielsweise bei weit verbreiteten Desktop-Programmen wie Microsoft Office immer der Fall ist. Auch für SaaS-Anbieter gilt hier, dass sich Kundenzufriedenheit langfristig positiv auf den Unternehmenserfolg auswirkt. Genauso wie bei IaaS-Kunden nicht mehr große Investitionen in Hardware, sondern sukzessive anfallen, verdienen SaaS-Anbieter ihr Geld auch über die Zeit und nicht mehr auf einen Schlag. Die großen Peaks beim Umsatz mit Software-Lizenzen, wie etwa bei einem neuen Release, gehören damit der Vergangenheit an.

Als prominentes Beispiel sei an dieser Stelle die SaaS-Lösung BusinessByDesign¹⁸ angeführt, mit der die SAP versucht im Mittelstand Fuß zu fassen und damit neue Kundensegmente zu erschließen. Dieser Schritt wurde nötig nachdem bei großen Unternehmen und Konzernen eine gewisse Sättigung eingetreten ist, das klassische SAP-Produktportfolio aber für den Mittelstand in der Regel zu teuer ist. Die Software-Lösung wird von SAP auf eigenen Serverfarmen betrieben, der Kunde nutzt diese über das Internet als SaaS. Für ihn entfallen damit eigene Investitionen in Hardware, die bei SAP gehostete Software kann noch durch das sog. Scoping im eingeschränkten Umfang an die jeweiligen Bedürfnisse angepasst werden. Pro Anwender und Monat fallen nach ersten Plänen 50 € an. Doch die ursprünglich für 2008 geplante Einführung verzögert sich aktuell, da SAP mit Performance-Problemen zu kämpfen hat. Dies zeigt wiederum, wie sehr SaaS-Angebote von der darunter liegenden Infrastruktur abhängig sind.

6.2 Investitions- vs. Betriebskosten

Der Kostenaspekt beim Einsatz von Cloud Computing ist bereits mehrfach angeklungen. Über alle drei Schichten hinweg ist zu beobachten, dass eine Verschiebung von Fixkosten hin zu variablen Kosten stattfindet. Statt hoher Investitions- und Innovationskosten zu bestimmten Zeitpunkten, wenn es beispielsweise um die Anschaffung oder Erweiterung von Hard- und Software geht, fallen die Kosten nun über die Nutzungszeit verteilt als Betriebskosten an. Das Thema der Abschreibungen tritt damit zunehmend in den Hintergrund weil es schlichtweg immer weniger Hardware gibt, die abgeschrieben werden könnte. Während bei IaaS und PaaS die Berechnung in der Regel rückwirkend anfällt, da die Kosten wie gesehen auf dem tatsächlichen Verbrauch basieren, können bei SaaS Rechnungen im Voraus gestellt werden. Jeder Account mit einem bestimmten Funktionsumfang kann mit Beginn der Nutzung direkt für eine vorab vereinbarte Vertragslaufzeit in Rechnung gestellt werden.

Insbesondere bei Startups, kleinen und mittelständischen Unternehmen dürfte relativ schnell klar sein, dass diese Art der anfallenden Kosten eine große Chance darstellt. Häufig fehlt es diesen Firmen am nötigen Kapital für große Investitionen und insbesondere Startups mit neuen Ideen und

¹⁸Siehe auch www.sap.com/germany/sme/solutions/businessbydesign/index.epx.

Geschäftsmodellen können die tatsächlich benötigte Rechen- und Speicherleistung wenn überhaupt nur sehr schwer einschätzen. Schnell übersteigt die Nachfrage alle Erwartungen und zwingt die IT-Umgebung in die Knie (was wiederum ein negatives Licht auf die Firma wirft). Auch lassen sich in hohem Maße Personalkosten für die Administration sparen, da dies ganz im Verantwortungsbereich des Dienstleisters liegt.

Etwas differenzierter sieht die ganze Sache bei Großunternehmen und Konzernen aus, die über gewachsene IT-Systeme und teilweise sehr große IT-Abteilungen verfügen. Oft handelt es um Systemlandschaften aus vielen unterschiedlichen Systemen mit vielen unterschiedlichen Datenquellen. Aufgrund der hohen Komplexität und Heterogenität gestaltet sich ein Paradigmenwechsel als kompliziert und teuer. Die Tatsache, dass durch eine solche Umstellung schlagartig sehr viele Angestellte ihren Job verlieren würden, ist dabei eher zweitrangig zu betrachten. Viel mehr *Angst* als um Jobabbau – der auf die Rendite und damit den Shareholder Value des Unternehmen sogar noch positiven Einfluss hätte – haben Unternehmen vor Datenschutzproblematiken, wie sie in Kapitel 6.5 noch im Detail erklärt werden. Im Gegensatz zu einer Hybrid oder Public liegen bei Nutzung einer Private Cloud die Daten zwar noch auf den eigenen Systemen, trotzdem haben aber auch hier Dritte die Möglichkeit des Einblicks. Somit ist die Kostenfrage und die Abwägung zwischen Innovations- und Betriebskosten nur ein Faktor unter vielen. Selbst wenn in dieser Hinsicht große Einspar- und Verbesserungspotentiale möglich sind, können andere Faktoren zu einer anderen Entscheidung führen.

6.3 Von der Wertschöpfungskette zum Wertschöpfungsnetz

Cloud Computing sorgt für massive Veränderungen und ermöglicht Szenarien, die heute noch kaum vorstellbar erscheinen aber in 10 - 15 Jahren schon Standard sein könnten. Kann ein Unternehmen auch ganz ohne hausinterne IT und ohne die gewohnte Serverlandschaft auskommen?

Unabhängig davon, ob das genannte Szenario realistisch erscheint und mittelfristig eintritt oder nicht, verändert der Einsatz von Cloud Computing die klassische vertikale Wertschöpfungskette innerhalb von Unternehmen. Der Weg geht hin zu komplexen Wertschöpfungsnetzen, in denen große Teile transparent über die gesamte Welt verteilt sind – das Unternehmen sieht nur die IT-Services und die Ergebnisse. Grundlage für das Funktionieren dieser Zusammenarbeit über Unternehmens- und Netzwerkgrenzen hinaus sind „partnerschaftliche Beziehungen auf der Basis von Vertrauen in Netzwerken“¹⁹. Nur damit lassen sich die „Skaleneffekte und die hochgradige Automatisierung von Dienstleistungsangeboten“²⁰ erreichen. Eine sehr große Bedeutung kommt innerhalb der Zusammenarbeit den Themen Performance, Verfügbarkeit und Sicherheit zu.

Die Frage, die sich Unternehmen stellen müssen, ist: Selber machen, kaufen oder kooperieren? Aus den möglichen Kombinationen der 3 Ebenen von Cloud Computing ergeben sich insgesamt 7 Geschäftsmodelle (siehe Grafik 6.1). Es ist also nicht nötig die gesamte Wertschöpfungskette in die

¹⁹Siehe [BIT09, Seite 33].

²⁰Siehe [BIT09, Seite 33].

Wolke zu verlagern, stattdessen können einzelne Teilschritte herausgegriffen und an IaaS-, PaaS- oder SaaS-Dienste ausgelagert werden. Welche Teile dies sind, dürfte von Unternehmen zu Unternehmen unterschiedlich sein, da auch die Geschäftsprozesse und Geschäftsprozessschritte zu verschieden sind, um eine allgemeingültige Aussage oder Empfehlung treffen zu können.

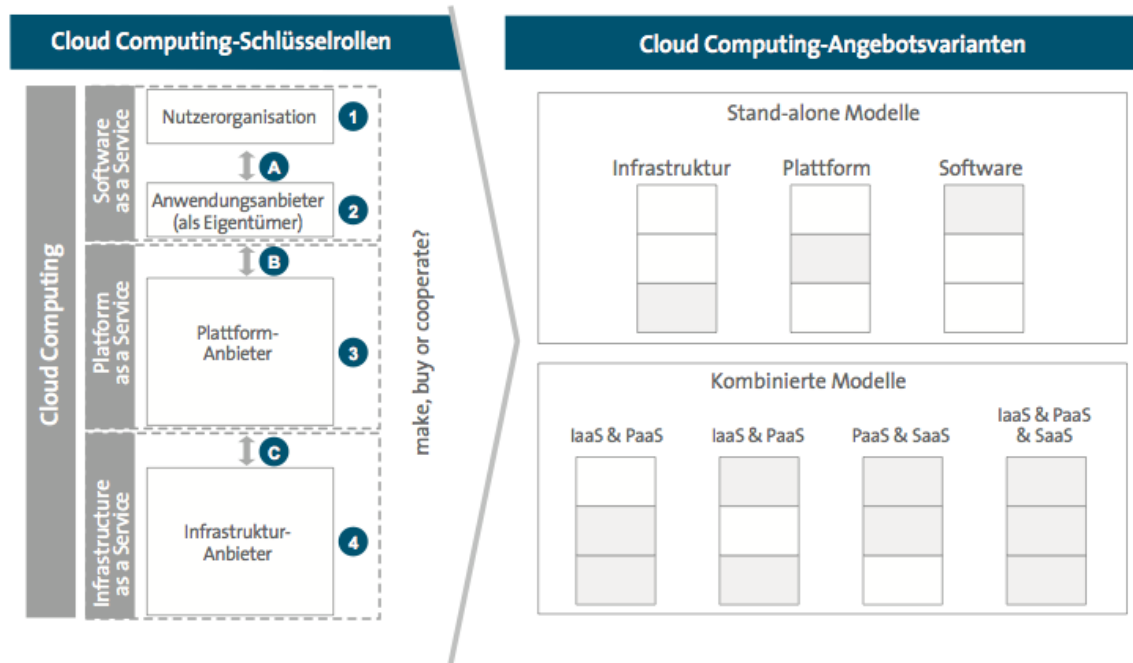


Abb. 6.1: Geschäftsmodelle beim Einsatz von Cloud Computing, Quelle: [BIT09]

Auch müssen entsprechende Voraussetzungen geschaffen werden, um den Unternehmen das Auslagern attraktiv zu machen. Die Datenschutz-Thematik und sonstige rechtliche Themen erstrecken sich über alle Schichten hinweg und entlang der gesamten Wertschöpfungskette. Beim Einsatz von SaaS ist vor allem die optimale Unterstützung der Geschäftsprozesse nötig. Viel zu oft ist zu beobachten, dass Unternehmen die Prozesse an die Software anpassen, das Gegenteil sollte der Fall sein. PaaS-Lösungen müssen ihre Dienste über einfache Schnittstellen zur Verfügung stellen und naht- und komplikationslos mit den schon vorhandenen Systemen zusammenarbeiten. Hierfür müssen sich auch noch Standards entwickeln und etablieren, damit Software-Systeme nicht mit viel Aufwand und Kosten an proprietäre Schnittstellen angepasst werden und die Kunde damit zusätzlich noch in die Abhängigkeit von einem Anbieter geraten. Der erste Schritt ist mit dem Open Cloud Manifesto²¹ gemacht, das zahlreiche namhafte Hersteller AT&T, die Eclipse Foundation, EMC, SAP und VMWare unterschrieben haben. Leider fehlen auf der Liste der Unterstützer die Goblal Player wie Amazon, Google, Microsoft oder Salesforce.com. Dem Anfang müssen jetzt weitere Taten folgen, um möglichst noch weitere Unterstützer zu finden. Denn insbesondere im Hinblick auf die Hybrid Cloud ist die Kompatibilität zwischen Systemen unterschiedlicher Hersteller eminent wichtig. Bei

²¹Siehe auch www.opencloudmanifesto.com.

IaaS-Lösungen sind abschließend vor allem die Verfügbarkeit und jederzeit auch die tatsächliche Bereitstellung der benötigten Leistung wichtig.

Ansonsten kann eine Komponente des neu geschaffenen Wertschöpfungsnetzes schnell zum Single Point of Failure werden und den gesamten Wertschöpfungsprozess und damit die Arbeit in einem Unternehmen zum Erliegen bringen. Mit dem unschönen Nebeneffekt, dass nun der Verantwortliche, der früher ein paar Büros weiter saß, nicht mehr so einfach zu fassen ist und der Handlungsspielraum der Unternehmen in diesem Worst-Case-Szenario sehr eingeschränkt ist. Schlimmstenfalls steht im Unternehmen die Arbeit still weil ich kritisches System oder obligatorische Daten in die Cloud ausgelagert wurden und aufgrund von Problemen nicht zur Verfügung stehen. Im Gegenzug ist dies aber auch mit ein entfallender Kostenblock, der durch das interne Monitoring entsteht.

6.4 Sourcing

Cloud Computing wird auch als IT-Sourcing-Modell charakterisiert, Unternehmen können mit dieser Hilfe ihre Business-Anforderungen umsetzen. Eines der größten Hemmnisse beim Umstieg auf Cloud Computing betrifft Datenschutzaspekte sowie das fehlende Vertrauen in den Datenschutz sowie den Schutz vor Datendiebstahl der nicht selten unternehmenskritischen Daten. Auch die Gefahr, dass die Daten den europäischen Wirtschaftsraum verlassen und dann anderen schwächeren Gesetzen unterliegen, ist nicht zu verachten. Darüber hinaus gibt es oft auch die Befürchtung, zu sehr in die Abhängigkeit von einem Anbieter zu geraten, was nur durch Standards zu verhindern wäre, die sich allerdings erst noch bilden müssen.

All diese Punkte erweisen sich in der stärksten Ausprägung des Cloud Computing – alle Daten in der Public Cloud – durchaus als begründet. Allerdings lässt auch Cloud Computing diverse Sourcing-Optionen zu, die den Grad der Interoperabilität frei wählbar machen. Zunächst einmal gibt es mit den Private Clouds und den Public Clouds zwei reine Cloud-Varianten. Während bei der Private Cloud die Hardware auch Eigentum des eigentlichen Kunden ist und der Zugriff für den Kunden selbst sowie dessen Kunden und Geschäftspartner über das Intranet erfolgt, ist bei einer Public Cloud die gesamte IT-Infrastruktur im Besitz des Dienstleisters, der damit auch die vollständige Kontrolle über die anfallenden Daten hat. Das Internet ist im letzteren Fall das obligatorische Kommunikationsmedium, während bei der Private Cloud alle Funktionen auch unabhängig vom Funktionieren des Internets verfügbar sind.

In der Realität haben die beiden Reinformen allerdings Seltenheitswert und der Nutzer von Cloud-Computing-Angeboten kann den Grad zwischen Eigen- und Fremdbetrieb selbst bestimmen (siehe Grafik 6.2). Für die daraus resultierenden Mischformen hat sich der Begriff der „Hybrid Cloud“ gebildet.

Während bei einer Private Cloud die gesamte Hardware und alle Services im Besitz des Unternehmens sind und auch von eigenem Personal betrieben werden, ist in der Managed Private Cloud

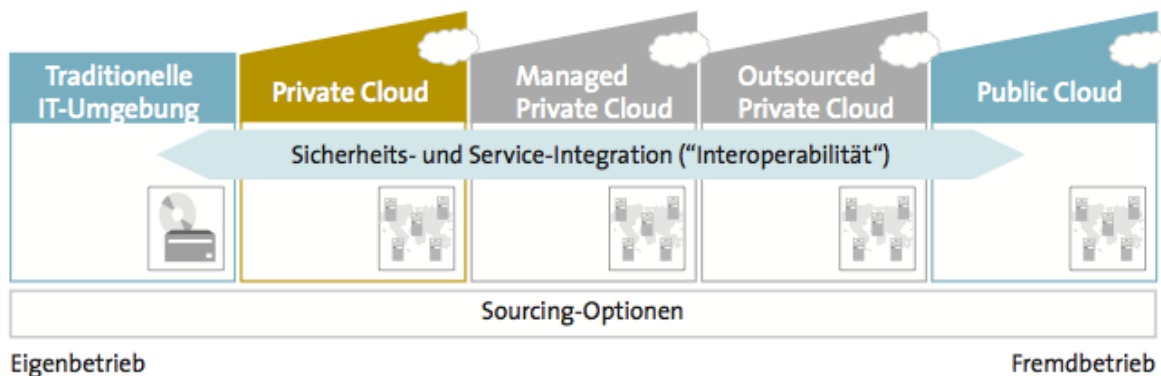


Abb. 6.2: Die Sourcing-Optionen erlauben Unternehmen die Positionierung zwischen Private und Public Cloud

Quelle: [BIT09]

der Dienstleister für den Betrieb verantwortlich, ohne dass sich die Eigentumsverhältnisse der IT-Ressourcen ändern. Innerhalb einer Outsourced Private Cloud ist der Kunde wieder selbst verantwortlich für den Betrieb, allerdings gehören sich die IT-Ressourcen hier im Besitz des Dienstleisters und stehen in der Regel auch physisch bei diesem oder zumindest außerhalb der Räumlichkeiten des Kunden. Die Public Cloud schließlich ist die Endstufe des möglichen Outsourcings. Alle IT-Ressourcen und Administrationsaufgaben liegen in der Verantwortung des Dienstleisters, der Kunde ist nur noch Nutzer mit (vertraglich vereinbarten) eingeschränkten Nutzungsmöglichkeiten. Während der Anbieter mit seinem System parallel mehrere Kunden bedienen kann, haben letztere keine Informationen mehr über physische Standorte oder die IT-Infrastruktur, sondern erhalten lediglich ein temporäres Nutzungsrecht. Ebenso steht durch die gemeinsame Nutzung durch verschiedene Kunden die Individualisierung im Konflikt mit der für den Anbieter vorteilhaften Standardisierung, über die er seinerseits Kosten sparen kann.

Wie schon im Kapitel zuvor kann auch beim Thema Sourcing keine allgemeine Empfehlung getroffen werden, welche Sourcing-Option nun die „beste“ ist. Es gibt aber quantitative Bewertungsmodelle, die dabei bei der Entscheidung über das Outsourcing von IT-Infrastrukturen an einen Cloud-Computing-Anbieter helfen können²². Eine dieser Möglichkeiten ist die Analyse des Discounted Cash Flow (DCF) oder das von Gartner entwickelte Modell Total Cost of Ownership (TCO). Letzteres erfasst alle direkten und indirekten verursachten Kosten. Diese Kosten lassen sich dann mit den aktuellen Kosten vergleichen. Neben den Kosten gilt es aber auch die Vor- und Nachteile bei den wesentlichen Punkten Eigentum, Verantwortung für den Betrieb und Möglichkeiten des Datenzugriffs gegeneinander abgewogen werden, um die für das eigene Unternehmen richtige Entscheidung zu treffen. Neben der rein quantitativen Analyse müssen auch die qualitativen Aspekte in die Entscheidungsfindung für oder gegen den Cloud-Computing-Einsatz mit einbezogen werden²³.

²²Siehe auch [Gro09].

²³Sihe auch [Gro09].

6.5 Datensicherheit und Datenschutz

Die idealtypische Vorstellung von Cloud Computing verspricht dem Nutzer zu jeder Zeit unbegrenzte Rechenleistung, unbegrenzten Speicherplatz sowie beim Einsatz von SaaS die Verfügbarkeit und immer gleiche Performance bei der Nutzung. In der Praxis dürfte allerdings einleuchtend sein, dass die vorhandenen Kapazitäten jedes Anbieters irgendwann einmal erschöpft sind. Dem Idealbild des Cloud-Computing-Paradigmas folgend, werden nun die fehlenden Leistungen über andere Anbieter aus der Cloud bezogen – und das alles vollkommen transparent für den Benutzer.

6.5.1 Transparenz

Doch genau diese propagierte Transparenz stellt sich sowohl aus psychologischen als auch rechtlichen Gesichtspunkten als Hemmschuh dar und dürfte mit zum entscheidenden Faktor werden wenn es darum geht, ob sich Cloud Computing über die diversen Branchen und Anwendungsszenarien hinweg durchsetzt oder nicht. Große Unternehmen werden sich allen Vorteilen zum Trotz sicherlich solange gegen das Cloud-Konzept wehren, bis es eine ausreichende „Transparenz bei der geographischen Lage der gespeicherten Daten und deren Schutz gibt“ [Com09]. Wer gibt schon gerne Daten, die die Geschäftsgrundlage darstellen, aus der Hand ohne zu wissen wo diese gespeichert und mit welchen Sicherheitsmaßnahmen sie vor unberechtigtem Zugriff geschützt werden? Schließlich birgt dies ja auch das Risiko, dass irgendwann einmal unternehmenskritische Daten auf Systemen eines direkten Konkurrenten landen, der seinerseits überschüssige Kapazitäten in der Cloud zur Verfügung stellt – eine Horrorvorstellung für die Verantwortlichen im Unternehmen. Das nötige Vertrauen in die Cloud-Computing-Komponenten muss ich demnach erst noch bilden, was durch ein gewisses Mehr an Transparenz unterstützt werden könnte. Doch das wiederum „würde in gewisser Weise der Idee des Cloud Computing selbst widersprechen“²⁴.

Neben der Sicherheit bei der Speicherung der Daten ist auch der Sicherheit beim Datentransport über das Internet wichtig. Die sicherste Datenspeicherung nützt nichts wenn die Daten unverschlüsselt übertragen werden oder Angriffe wie etwa eine Man-In-The-Middle-Attacke oder Sniffing-Attacken möglich sind. Anders als die gerade thematisierte Problematik der transparenten Datenspeicherung, lässt sich die Sicherheit beim Datentransport gut herstellen und auch mittels geeigneter Zertifikate belegen. Trotz der vorhandenen Risiken kann man sich aber der Professionalität des Cloud-Computing-Partners sicher sein, der „die Unternehmensdaten wie seinen Augapfel hüten [wird], denn davon hängt sein Geschäftserfolg ab“²⁵. Die Vorkehrungen zur Sicherstellung der Datensicherheit dürfte aus diesem Grunde auch deutlich besser sein, als dies die meisten Unternehmen mit angemessenem Aufwand selber leisten können. So sind die Daten innerhalb von Amazon S3 automatisch mehrfach redundant gesichert²⁶.

²⁴Siehe [Com09].

²⁵Siehe [Gro09].

²⁶Siehe auch [Gro09].

6.5.2 Rechtlicher Rahmen nach BDSG

Andere Länder, andere Sitten. Dieses bekannte Sprichwort bezieht zwar eher auf die kulturellen Unterschiede und Gewohnheiten zwischen verschiedenen Nationen, aber auch die Gesetzgebungen unterscheiden sich teilweise signifikant – und das auch schon innerhalb der Europäischen Union. Im vorherigen Kapitel wurde grundsätzlich auf die Thematik der Datensicherheit und des Datenschutzes eingegangen. Deutsche Unternehmen sind in erster Linie den Gesetzen und Reglementierungen des Bundesdatenschutzgesetzes (BDSG) unterworfen. Liegen die Daten aber auf Servern im Ausland, unterliegen die Daten schnell anderen Gesetzen, was den ungewollten Fremdzugriff beispielsweise von Behörden oder staatlichen Institutionen ermöglichen könnte.

Deutschland hat trotz aller Veränderungen der jüngsten Vergangenheit immer noch einen sehr hohen Standard in Sachen Datenschutz, insbesondere im Hinblick auf personenbezogene Daten²⁷. In § 28 Abs. 1 BDSG ist geregelt, dass „das Erheben, Speichern, Verändern oder Übermitteln personenbezogener Daten oder ihre Nutzung als Mittel für die Erfüllung eigener Geschäftszwecke [...] zulässig“ ist. Allerdings muss den Benutzer dem zunächst zustimmen, nachdem er über den Umfang der erhobenen Daten aufgeklärt und auch dahingehend unterrichtet wurde, in welcher Weise und zu welchem Zweck die Speicherung stattfindet. Üblicherweise findet sich diese Zustimmung in den Allgemeinen Geschäftsbedingungen (AGB) der Unternehmen.

Juristisch gesehen stellt die Auslagerung der Datenspeicherung im Rahmen einer Cloud-Computing-Strategie eine wesentliche Veränderung dar, was bedeuten würde, dass der Kunde der Speicherung seiner Daten in dieser neuen Art und Weise erneut zustimmen müsste. Sicherlich ein Punkt, der nur sehr schwer umzusetzen wäre und die Unternehmen mit einer Nutzung von Cloud Computing gegen den Datenschutz verstoßen ließe.

6.6 Vergleich zwischen dem deutschen und dem weltweiten Markt

Nachdem nun das zweifelsohne vorhandene technische Potential von Cloud Computing sowie die bereits vorhandenen Abrechnungsmodelle besprochen worden, soll nun darauf eingegangen werden, wie sich der Cloud-Computing-Markt zum aktuellen Zeitpunkt darstellt und welche Entwicklung zu erwarten ist. Laut einer Studie von Gartner belief sich das Volumen des Cloud-Computing-Markts im Jahr 2008 weltweit auf 46 Milliarden US-Dollar. Bis zum Jahr 2013 soll es auf über 150 Milliarden US-Dollar ansteigen²⁸.

Der deutsche Markt folgt dem Trend des Cloud Computing mit einer gewissen zeitlichen Verzögerung, sein Volumen von 285 Millionen € im Jahre 2008 ist im Vergleich noch sehr gering. Der erwartete Anstieg auf über 560 Millionen € bis zum Jahresende 2011 bedeutet allerdings einen

²⁷ und dazu müssen mittlerweile auch IP-Adressen gezählt werden

²⁸Siehe auch [BITKOM].

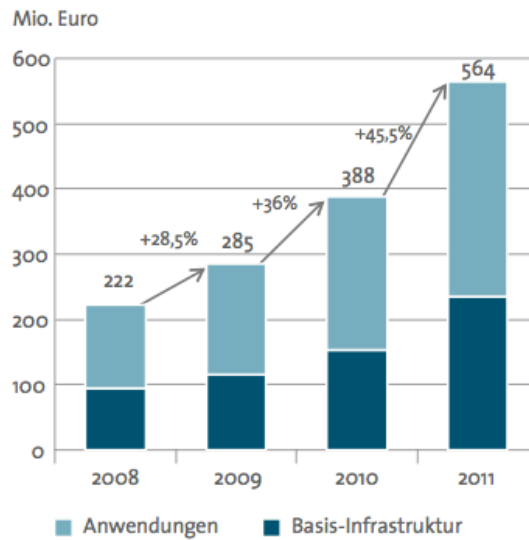


Abb. 6.3: Die Entwicklung des Cloud-Computing-Markts in Deutschland
Quelle: [BIT09]

durchschnittlichen Zuwachs von nahezu 37 % – Zuwachsraten, von denen andere Branchen aber auch andere Bereiche der IT angesichts der momentanen Wirtschaftslage nicht einmal zu träumen wagen.

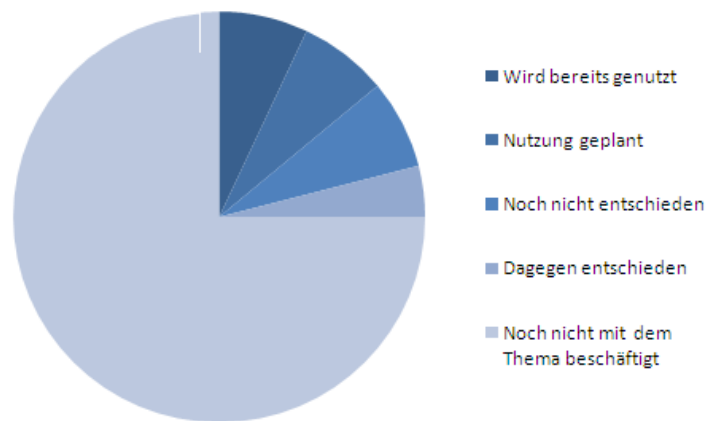


Abb. 6.4: Cloud Computing – Einsatzgrad in Deutschland 2009
Quelle: IDC, Cloud Computing und Services – Status quo und Trends in Deutschland 2009 via [Hei09]

Cloud Computing ist in zahlreichen deutschen Unternehmen noch nicht angekommen, wie die IDC-Studie „Cloud Computing und Services - Status quo und Trends in Deutschland“ zeigt. So haben sich 75 % von 805 befragten IT-Verantwortlichen noch nicht einmal mit dem Thema beschäftigt. Von den übrigen Befragten haben sich allerdings auch nur 16 % gegen eine Nutzung von Cloud entschieden – die Hauptgründe waren hierbei Sicherheitsbedenken –, was die praktische Relevanz

unterstreicht, zumindest als „ergänzende Möglichkeit zur Beschaffung von IT“²⁹. Während in einer von Business Technology Services in Auftrag gegebenen globalen Studie weltweit 40 Prozent der 500 befragten Manager und IT-Verantwortlichen Sicherheitsbedenken bei der Nutzung von Cloud Computing äußerten, waren es in Deutschland satte 64 Prozent.

Alles in allem wird der deutsche Cloud-Computing-Markt aus geschilderten Gründen wohl eine deutlich langsamere Entwicklung nehmen als beispielsweise der amerikanische. Gründe hierfür wurden auf den vorherigen Seiten schon erörtert. Neben der Zurückhaltung der potentiellen Nutzer, zögern aber auch die designierten Anbieter teilweise noch entsprechende Angebote zur Verfügung zu stellen. So war es dem Autor beispielsweise möglich am Rande des World Usability Day am 12. November 2009 in Stuttgart mit einem Anbieter von Dokumentenarchivierungs- und Dokumenten-Workflow-Systemen über die Thematik ins Gespräch zu kommen. Tatsächlich arbeitet man auch dort im Hintergrund mit den Vertriebspartnern bereits an entsprechenden Möglichkeiten, „um technisch gerüstet zu sein“. Allerdings erwartet man keinen besonderen Run sobald ein entsprechender Dienst zur Verfügung steht und begründet dies mit der langjährigen Erfahrung auf dem Markt für Dokumenten-Management-Systeme. So sei es eben eine deutsche Mentalität, Daten nicht aus der Hand geben zu wollen und diese stattdessen innerhalb der eigenen Systeme zu speichern: „Die Deutschen wollen ihre Daten bei sich haben“.

Neben den hinlänglich diskutierten Datenschutzthemen kommen aber im gerade geschilderten Fall der Dokumentenarchivierung auch noch andere gesetzliche Vorgaben und Rahmenbedingungen ins Spiel, wie etwa die Grundsätze ordnungsgemäßer DV-gestützter Buchführungssysteme (GoBS). Bei strikter Einhaltung dieser Grundsätze ist es insbesondere nicht erlaubt, die Daten außerhalb des europäischen Wirtschaftsraumes zu speichern, was erneut das Thema der Transparenz aufwirft. Verstöße dagegen könnten verheerende Konsequenzen für die betroffenen Unternehmen zur Folge haben. Das kann schlimmstenfalls zu einem schlechteren Rating führen, was wiederum zu Schwierigkeiten und schlechteren Konditionen bei Kreditverhandlungen mit den Banken führen kann.

Somit sind es vor allem auch die gesetzlichen Bestimmungen, die insbesondere in Deutschland die Bremse einer umfangreichen Verbreitung von Cloud Computing. Gesetze, die unsere Daten eigentlich „schützen“ sollen, verhindern den Umstieg auf flexiblere und kostengünstigere Möglichkeiten der IT-Beschaffung und -Nutzung. Somit könnte der Hype Cloud Computing auch gesetzliche Änderungen und Erweiterungen nötig machen, um die neuen Möglichkeiten nicht mehr zu blockieren und sowohl für die Anbieter als auch die potentiellen Nutzer Rechtssicherheit zu schaffen.

²⁹Siehe [Hei09].

Kapitel 7

Fazit

Cloud Computing ist wie gesehen nichts grundlegend Neues, sondern fasst bereits verfügbare technische Möglichkeiten lediglich unter einem neuen Begriff zusammen. Auch ist Cloud Computing kein fertiges Produkt, das direkt eingesetzt werden kann. Vielmehr muss der Nutzen für die sehr unterschiedlichen Rahmenbedingungen in den verschiedenen Unternehmen stets individuell betrachtet werden.

Keinesfalls ist Cloud Computing ein Allheilmittel oder die Lösung für alle Leistungs- und Skalierungsprobleme, aber es kann sich für bestimmte Teilbereiche hinsichtlich der Leistung und Kostenaspekten lohnen. Hat man innerhalb seines Unternehmens eben solche Teilbereiche identifiziert, sind die rechtlichen Rahmenbedingungen und die Unternehmens-Compliance zu berücksichtigen, um eine abschließende Beurteilung treffen zu können.

Gerade vor dem Hintergrund regelmäßig bekannt werdender Datenskandale und Sicherheitslücken in kritischen Systemen muss jedes Unternehmen für sich selbst beurteilen, welche Daten es in die alleinige Obhut softwareseitiger Sicherung geben und welche es lieber im eigenen physischen Einflussbereich behalten möchte. In jedem Fall wird es Daten und Prozesse geben, die für das Unternehmen so wichtig sind, dass es die Hoheit über diese Daten nicht abgeben möchte. Die Vision, dass ein Unternehmen komplett ohne eigene IT auskommen kann, sehen wir auf absehbare Zeit als nicht praktikabel. Für die weniger kritischen Daten und Prozesse kann Cloud Computing jedoch als sinnvolle Ergänzung oder in Teilen auch als Alternative zu interner IT darstellen werden.

Das erwartete Wachstum begründet sich unserer Meinung nach aus dem Segment semikritischer Komponenten und wird spätestens dann eine Sättigung erfahren, wenn nur noch kritische IT im Unternehmen vorhanden ist. Ab hier ist es eine Frage des Vertrauens, das die jeweiligen Unternehmen in die Cloud setzen, ob und inwieweit sie weitere Interna in die Cloud verschieben.

Gleichwohl ist zwischen KMUs und Großunternehmen zu differenzieren. Für Erstere stellt Cloud Computing aufgrund seiner Dynamik und Kosteneffizienz eine nicht zu unterschätzende Chance dar. Großunternehmen werden hingegen weiterhin ihre IT selbst betreiben wollen, da sie aufgrund ihrer Größe einerseits die nötigen Kapazitäten für den eigenen Betrieb stellen können sowie andererseits IT-Leistungen in einem Umfang benötigen, in dem Skaleneffekte durch den anteiligen Zukauf von Leistung nicht mehr möglich sind.

Literaturverzeichnis

- Apa09** APACHE: *Apache Geronimo*. <http://geronimo.apache.org/>, 2009. – Letzte Abfrage: 26. November 2009
- Bar09** BARNETT, Alex: *Time to Define „Platform as a Service“ (or PaaS)*. <http://alexbarnett.net/blog/archive/2008/02/19/time-to-define-quot-platform-as-a-service-quot-or-paas.aspx>, 2009. – Letzte Abfrage: 2. Dezember 2009
- BIT09** BITKOM: *Cloud Computing – Evolution in der Technik, Revolution im Business*. http://www.bitkom.org/de/themen/36129_61111.aspx, 2009. – Letzte Abfrage: 28. Oktober 2009
- Bor09** BORN, Achim: *Himmlicher Service*. In: *iX* 11 (2009)
- Bro09** BROSOWSKI, Jan: *ICC: Warum keine Flatrates?* <http://blog.brosowski.biz/2009-04/icc-warum-keine-flatrates/>, 2009. – Letzte Abfrage: 6. Dezember 2009
- Com09** COMPUTERWOCHE: *Die Wahrheit über Cloud Computing*. <http://www.computerwoche.de/management/cloud-computing/1881561/index7.html>, 2009. – Letzte Abfrage: 26. November 2009
- Fou09** FOUNDATION, The E.: *Eclipse IDE*. <http://www.eclipse.org/>, 2009. – Letzte Abfrage: 26. November 2009
- Gmb05** GMBH, Symnatec (: *Symantec im Leader-Segment des Gartner Magic Quadrant*. <http://www.pressebox.de/pressemeldungen/symantec-deutschland-gmbh/boxid-40158.html>, 2005. – Letzte Abfrage: 21. Dezember 2009
- Gol09** GOLEM.DE: *Cloud Computing - Amazon öffnet Spot-Markt für EC2-Instanzen*. <http://www.golem.de/0912/71847.html>, 2009. – Letzte Abfrage: 15. Dezember 2009
- Gro09** GROHMANN, Werner: *Business-Applikationen in der Cloud – Ein Marküberblick über SaaS-Lösungen*. In: *t3n* 18 (2009), S. 24–27
- Hei09** HEINS, Eberhard: *CRM-Clouds ziehen auf*. In: *is report* 11 (2009)
- Her09** HERRMANN, Wolfgang: *Cloud Computing – Die geplante Revolution*. <http://www.computerwoche.de/management/cloud-computing/1904964/>, 2009. – Letzte Abfrage: 26. Oktober 2009

- IR09** IT-REPUBLIK: *Software ist tot - Es lebe die Plattform.* <http://it-republik.de/business-technology/news/Software-ist-tot-%96-es-lebe-die-Plattform-043214.html>, 2009. – Letzte Abfrage: 21. Dezember 2009
- Lin09** LINTHICUM, David: *Are We Moving to 'Platform as a Service?'* . http://intelligent-enterprise.informationweek.com/blog/archives/2007/07/are_we_moving_t.html, 2009. – Letzte Abfrage: 2. Dezember 2009
- Mas09** MASTERSON, Mark: *Cloud-Computing – Mehr als nur ein Hype.* In: *t3n* 18 (2009), S. 21–23
- Mit09** MITCHELL, Dave: *Defining Platform-As-A-Service, or PaaS.* <http://blogs.bungeeconnect.com/2008/02/18/defining-platform-as-a-service-or-paas/>, 2009. – Letzte Abfrage: 2. Dezember 2009
- New09** NEWS t3n: *Amazo EC2 Spot Instances: „ Wie viel würden Sie zahlen?“ - Neues Bezahlmodell für Amazons Cloud Computing.* <http://t3n.de/news/amazon-ec2-spot-instances-263214/>, 2009. – Letzte Abfrage: 15. Dezember 2009
- sal09** SALESFORCE.COM: *Cloud Computing Model Editions & Pricing.* http://www.salesforce.com/platform/platform_edition/, 2009. – Letzte Abfrage: 21. Dezember 2009
- SUN09a** SUN: *Core Java - Java Doc.* <http://java.sun.com/j2se/javadoc/>, 2009. – Letzte Abfrage: 26. November 2009
- SUN09b** SUN: *Java Persistence Annotation.* <http://java.sun.com/javaee/technologies/persistence.jsp>, 2009. – Letzte Abfrage: 26. November 2009
- tec09** TECHTARGET: *Platform as a Service (PaaS).* <http://whatis.techtarget.com/definition/platform-as-a-service--paas-.html>, 2009. – Letzte Abfrage: 2. Dezember 2009
- Wik09** WIKIPEDA: *Die Wahrheit über Cloud Computing.* http://de.wikipedia.org/wiki/Cloud_Computing, 2009. – Letzte Abfrage: 26. November 2009