

Eine Taxonomie und Bewertung von Cloud Computing-Diensten aus Sicht der Anwendungsentwickler

Michael Kretzschmar und Mario Golling

Universität der Bundeswehr München

Fakultät für Informatik

D-85577 Neubiberg

Email: {michael.kretzschmar, mario.golling}@unibw.de

Abstract: Die Idee des Cloud Computings ist eine der meistversprochensten der jüngeren IT-Geschichte. Die Kombination aus bekannten Konzepten wie das der verteilten Systeme mit neuen technischen Möglichkeiten auf den Gebieten der Virtualisierung und Datenübertragung ermöglichen eine neue Stufe der Leistungsfähigkeit und Wirtschaftlichkeit. Abseits der unumstrittenen Vorzüge der Cloud gibt es viele offene Fragen auf diesem recht jungen Gebiet der IT. Basierend auf der Tatsache, dass die meisten Anbieter beim Implementieren eigene Normen entwickeln, existieren zur Zeit nur sehr eingeschränkt Standards bzw. Konventionen bezüglich Datenübertragung, Sicherheit und Interoperabilität etc.

Neben Nutzern und Anbietern von Cloud-Diensten gibt es die Anwendungsentwickler. Die Anbieter von Cloud-Diensten stellen Entwicklern Anbindungen unter anderem in Form von Application Programming Interfaces (APIs) zur Verfügung, durch die Programme den jeweiligen Dienst einbinden können. Doch eben diese Anbindungen sind ebenfalls nicht standardisiert und werfen bei Entwicklern grundsätzliche Fragen bezüglich Anbieterabhängigkeit, Interoperabilität, Programmiersprachen- und Protokollauswahl etc. auf. Zudem fehlt es aus Entwicklersicht an Übersichten und Entscheidungshilfen zu den verschiedensten Anwendungsspektren.

Spezifische Antworten auf diese Fragen zu geben ist nur sehr eingeschränkt möglich, da jeder Entwickler eigene Ziele und Voraussetzungen hat, die die Entscheidungen bezüglich einzelner Punkte maßgeblich beeinflussen. Im Sinne einer Entscheidungsunterstützung wird in diesem Beitrag ein Überblick über aktuelle Implementierungen von Cloud-Diensten gegeben. Hierbei wird aufbauend auf einem erweiterten SPI-Modell betrachtet, welche Arten von Diensten gegenwärtig existieren und welche speziellen Verfahren für allgemeine Aktionen, insbesondere den Zugriff auf die Dienste, benutzt werden können. Es werden Optionen gegenübergestellt und deren Implementierungen evaluiert. Dabei spielt der Vergleich von SOAP- mit REST-basierten Anbindungen eine große Rolle. Als Basis für Entwickler wird zusammenfassend zu jedem untersuchten Spektrum (Computing, Speicher, Plattform und Software) eine Entscheidungshilfe mit den jeweiligen Optionen präsentiert.

1 Motivation

In zahlreichen White Papers verschiedener Anbieter von Cloud-Diensten sind die Vorteile des Arbeitens in der Cloud aufgezählt [TSW, GGW09, SFW10]. So sparen beispielsweise Großfirmen auf Grund der geringeren Kosten im Vergleich zur Nutzung/Instandhaltung

eigener Rechenzentren. Insbesondere kleine, mittlere oder neugegründete Unternehmen können sich auf Grund der Philosophie vertragsungebundener Dienstleistungen (*Pay-as-You-Go*) risikofrei IT-Infrastrukturen leisten.

Neben den Endanwendern, die die Cloud verwenden, gibt es die Entwickler, die Software - die wiederum einen Cloud-Dienst als Grundlage nutzt - entwickeln. Hierzu stellen die Anbieter von Cloud-Diensten Entwicklern Anbindungen unter anderem in Form von Application Programming Interfaces (APIs) zur Verfügung, durch die Programme den jeweiligen Dienst einbinden können. Doch eben diese Anbindungen sind ebenfalls nicht standardisiert und werfen bei Entwicklern Fragen auf:

- Welche Programmiersprache soll verwendet werden?
- Welche Standards unterstützt der Cloud-Anbieter?
- Wird das Programm auch mit anderen Anbietern funktionieren?

Im Sinne einer Entscheidungsunterstützung wird daher im Folgenden ein Überblick über aktuelle Implementierungen von Cloud-Diensten gegeben. Dabei wird betrachtet, welche Arten von Diensten gegenwärtig existieren und welche speziellen Verfahren für allgemeine Aktionen, insbesondere den Zugriff auf die Dienste, benutzt werden können.

2 Das erweiterte SPI-Modell

Mit der Entwicklung des Cloud Computings begannen verschiedene Anbieter unterschiedliche Arten von Diensten anzubieten. Bereits sehr schnell deckte das Gesamtportfolio von Cloud-Diensten eine große Spannweite ab. Vom Angebot *einfacher* Dienste, wie der Nutzung von sogenannten *Speicherdiensten*, die im einfachsten Fall mehr oder weniger einer in das Internet ausgelagerten Festplatte gleichen (mit all den Vorteilen, wie weltweiter Verfügbarkeit, redundanter Speicherung etc.) bis hin zu komplexen Customer-Relationship-Management-Software (CRM). Zur besseren Übersicht hat sich sehr schnell ein Service Modell, das sogenannte SPI-Modell (SPI als Abkürzung für die drei Ebenen *Software as a Service*, *Platform as a Service* und *Infrastructure as a Service*) [Bia08] [MKL09], in der Fachschaft durchgesetzt, welches allgemein sämtliche Cloud-Dienste einer von drei aufeinander aufbauenden Ebenen zuteilt:

2.1 Infrastructure as a Service (IaaS)

IaaS umfasst Dienste, die grundlegende Ressourcen zur Verfügung stellen. Dabei kann es sich um Speicherplatz oder Rechenleistung handeln, die meist in Form von virtuellen Servern auf der physischen Hardware der Anbieter laufen und auf die - durch Abstraktion - in Form von Diensten zugegriffen werden kann. Sie kann weiter unterteilt werden in:

Cloud Computing - Der Bereich, der dem großen Konzept seinen Namen gibt, beschreibt

insbesondere Rechenleistung und Arbeitsspeicher. Hierzu werden die überwiegend in Rechenzentren befindlichen Ressourcen virtualisiert. Das heißt im Cloud-Kontext: Zusammenfassung als “großes Ganzes” und danach wieder Aufteilung in einzelne virtuelle Server. Verwendet werden hierzu Virtualisierungssysteme wie Microsofts Hyper-V [War08] oder der Xen Hypervisor von Citrix [Chi07] [Hes10].

Cloud-Speicher - Speicherressourcen lassen sich bei genauer Analyse des Marktes noch einmal in zwei verschiedene Ausprägungen unterteilen:

- *Blockspeicher* - Als Block angebotene persistente Speicherressourcen, die sich - vergleichbar mit Network-Attached Storage (NAS) - im Rechner als Laufwerk einbinden lassen, werden als Blockspeicher bezeichnet. Hierbei ist zu beachten, dass die Größe des Blocks, also die Größe des Laufwerks, im nachhinein nicht mehr verändert werden kann. Ein Beispiel für solch einen Dienst ist Zettas Enterprise Network Storage Service [Zet10].
- *BLOB-Speicher* - Im Vergleich mit Blockspeicher stellt BLOB-Speicher (BLOB als Akronym für Binary Large Object) ein agileres Konzept dar, da die Größe der Speicherressourcen hier praktisch unbeschränkt ist. Aufgrund der Möglichkeit Dateien öffentlich mittels Web-Link publik zu machen, muss in der Regel darauf geachtet werden, dass global einzigartige Namen verwendet werden. Amazons Simple Storage Service (S3) [S310], Googles Storage for Developers [SfD11] sowie Windows Azure Storage [AZS11] sind nur einige Beispiele.

2.2 Platform as a Service (PaaS)

PaaS beschreibt Dienste, die eine Laufzeitumgebung für Anwendungen bereitstellen und erlaubt es dem Verbraucher, eigene Anwendungen oder Webseiten in der Cloud zu betreiben. Auf variable Last der Anwendungen durch Berechnungen oder Zugriffe kann schnell reagiert werden, indem Ressourcen auf IaaS-Ebene flexibel genutzt werden (*automatische Skalierung*). Ein weiterer Vorteil ist, dass hier der Anbieter sicherstellt, dass keinerlei Arbeiten für das Erstellen einer Laufzeitumgebung durchgeführt werden müssen, da z. B. Frameworks, Patches schon installiert sind.

2.3 Software as a Service (SaaS)

SaaS beinhaltet Programme, die auf einer Cloud-Infrastruktur ausgeführt werden und beschreibt in sich geschlossene Anwendungen, die lediglich in einigen vorgegebenen Optionen konfiguriert werden. Dazu zählen Cloud-basierte E-Mail- oder CRM-Dienste wie die Sales Cloud 2 von Salesforce [SFS11]. Hier kann sich der Benutzer mit seinem Browser anmelden und mit dem Sales Cloud-Dienst arbeiten, ohne weitere Software auf seinem Rechner zu installieren oder zu konfigurieren. Sämtliche Abläufe finden in der Cloud statt. Lediglich die Darstellung der Benutzeroberfläche im Browser belastet das System des

Endanwenders. Xignite beispielsweise dient mit Finanzmarktdaten, die Entwickler mittels einer bereitgestellten Anleitung in ihre eigenen Programme integrieren können [Xig10].

3 Zugriffsarten auf Cloud-Dienste

Für den Zugriff auf die jeweiligen Dienste existieren verschiedene Konzepte:

- *Web GUIs* - Intuitive grafische Benutzeroberflächen werden praktisch von sämtlichen Anbietern zur Verfügung gestellt und sind mittels Browser zu bedienen.
- *Remote Administration* - Im Gegensatz zu Web GUIs werden hier Dienste von einem entfernten Ort, ohne wahrnehmbare Zwischenschicht zwischen Benutzer und Dienst, gesteuert. Zur Kommunikation werden beispielsweise Befehle über eine Terminal-Sitzung übertragen.
- *API Anfragen* - Die APIs der Cloud-Dienstanbieter stellen Entwicklern Vorgaben und Beispiele für Anfragen, sogenannte Requests, zur Verfügung, um Ihren Dienst entsprechend nutzen zu können. Die Herausforderung dabei besteht für die Entwickler darin, jeweils abhängig vom konkreten Cloud-Dienst individuell korrekte Anfragen zu generieren.

Der elementare Unterschied im Bereich der API Anfragen ist die Formulierung sowie die Übertragungsart. Cloud-Dienstanbieter stellen APIs bereit, die meist auf REST oder SOAP basierend beschrieben werden. REST ist eine Abkürzung für Representational State Transfer und beschreibt eine Sammlung von Vorgaben, die den Informationsaustausch vereinheitlicht, erleichtert sowie die Leistung dieser Systeme erhöht. So wird durch REST z.B. gefordert, dass jede Anfrage sämtliche Informationen enthält, die vom Server benötigt werden (Stateless) oder dass Informationen, die den Client interessieren könnten, in der Repräsentation der Ressource verfügbar sein sollen. SOAP, anfangs ein Akronym für Simple Object Access Protocol, hingegen ist ein auf XML basierendes Kommunikationsprotokoll für Anwendungen, das seine Nachricht standardisiert verpackt und über HTTP versendet.

- *Bibliotheken* - Um Entwicklern einen Teil der Arbeit für die Erstellung der Anfrage abzunehmen, existieren für die meisten Programmiersprachen Bibliotheken, sogenannte Libraries. Hier übersendet der Entwickler den durch die Bibliothek bereitgestellten Methoden die benötigten Informationen und diese erstellt und versendet eine korrekte Anfrage. Der Vorteil dabei ist, dass die Bibliotheksfunktionen zum einen intuitiver zu verwenden und zum anderen unabhängiger von Veränderungen auf Seiten der Cloud-Dienstanbieter sind. Ändert sich der Methodenaufruf auf Seiten des Cloud-Anbieters wird lediglich die Bibliothek geändert, nicht aber der Methodenaufruf des Programms an die Bibliothek.

Innerhalb der Bibliotheken stellen die *Multi Cloud Bibliotheken*, die sogenannten Multi Cloud Libraries (MCLs), eine Besonderheit dar. Insbesondere fehlende API

Standards sorgen dafür, dass viele Nutzer die gleiche Methoden (mit den gleichen Parametern) bei unterschiedlichen Cloud-Providern nutzen möchten. Eine solche MCL für Java ist z.B. jClouds [jc110], das eine Großzahl an Speicherdiensten ansprechen kann. Ein Problem dabei ist, dass die Dienste keine einheitliche Menge an Funktionen anbieten. Beispielsweise unterstützen nicht alle Anbieter das Umbenennen von Dateien. Dabei gibt es verschiedene Lösungsansätze. Der einfachste Weg ist es, lediglich von allen unterstützte Methoden in die MCL einzubinden, wobei individuelle Vorteile der verschiedenen Dienste zu Gunsten der Einheitlichkeit verloren gehen. Alternativ könnten sämtliche Funktionen implementiert werden, von denen einige jedoch nur in Verbindung mit bestimmten Anbietern nutzbar sind.

- *Abstraktionsschicht* - Hier wird die Generierung der Anfrage durch Einfügen einer Abstraktionsschicht komplett aus der Programmierenebene ausgelagert und so das Verwenden vollkommen generischer Anfragen ermöglicht. Im Gegensatz zu den MCLs, die einen eingeschränkten Funktionsumfang mit Bezug zu einer bestimmten Ressourcenart (Rechen, Speicher, etc.) aufweisen, unterstützt die Abstraktionsschicht ein viel breiteres funktionales Spektrum und den generischen Aufruf von komplexer aufgebauten Cloud-Diensten. Zudem kann eine Abstraktionsschicht auch von einem anderen Cloud-Anbieter in Form eines Cloud Broker zur Integration angeboten werden. Des Weiteren ist die Abstraktionsschicht - im Vergleich mit einer MCL - unabhängig von einer konkreten Programmiersprache.

Abbildung 1 fasst die Zugriffsarten noch einmal zusammen. Ein konkreter Dienst kann dabei selbstverständlich nicht nur eines der dargestellten Konzepte realisieren, sondern u.U. auch mehrere (so z.B. den Zugriff via Browser und Remote Desktop).

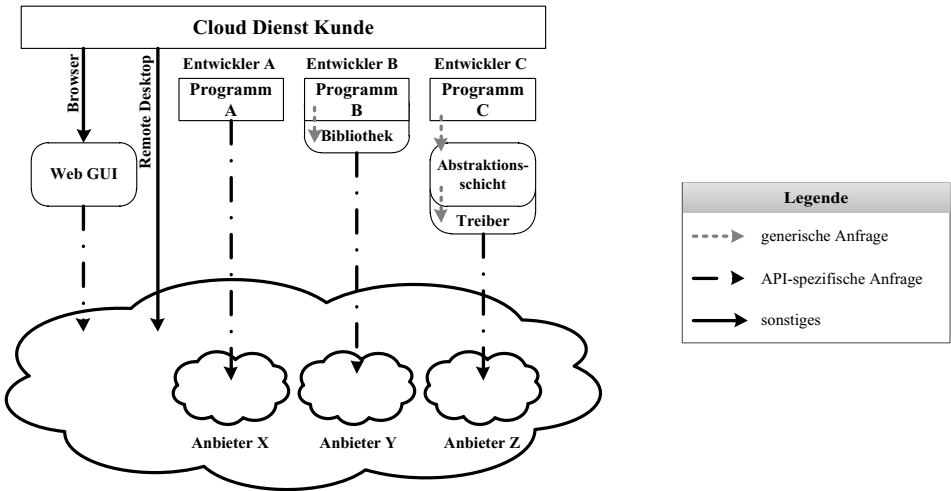


Abbildung 1: Zugriffsarten im Überblick

4 Cloud Computing

Im Folgenden werden die generischen Zugriffsarten, die im vorherigen Abschnitt definiert wurden, auf Ihre Anwendung und Umsetzung im Bereich Cloud Computing untersucht.

Hier sind insbesondere folgende Fragen von speziellem Interesse:

- *REST oder SOAP* - RESTful APIs haben hier das größte Potenzial. Im Vergleich mit SOAP besteht ein Vorteil darin, dass einige RESTful APIs die Möglichkeit bieten, Informationen in von XML verschiedenen Formaten zu versenden, was das eventuell nötige Übersetzen der Nachrichten überflüssig macht.
- *API oder Bibliothek* - Die Bibliotheken sind sehr simpel in der Verwendung, generieren sie doch durch einfache Methodenaufrufe selbständig korrekte Anfragen. Eine direkte API Anfrage ist etwas aufwändiger zu erstellen, da auf Details geachtet werden muss. In der Praxis hängt die Entscheidung zu allererst von der Verfügbarkeit einer gesuchten Bibliothek ab.
- *Wahl der Programmiersprache* - Hier steht Java an erster Stelle. Wenn es für einen Cloud Computing-Dienst eine Bibliothek gibt, ist eine für Java in den meisten Fällen vertreten. Dazu kommen die Multi Cloud Bibliotheken jClouds und Dasein sowie die Java Version von libcloud. Für Python, C# bzw. .NET, Ruby, PHP und Perl bieten ebenfalls einige Anbieter Bibliotheken an.

			Zugriffsarten			Formate			Bibliotheken						Multi Cloud Bibliotheken							
			Virtualisierung	Web GUI	Terminal-Sitzung	API	XML	JSON	C/S	text/plain	Java	Python	Ruby	C#/.NET	PHP	Perl	erlang	Delcloud	jclouds [Java]	Dasein [Java]	libcloud [Java]	libcloud [Python]
Anbieter	Produktname																					
Cloud Computing	Amazon	Elastic Compute Cloud	Xen	x	x	SOAP/Query	x				x	x	x	x	x	x	x	x	x	x	x	x
	Cloud Central	Servers	Xen	x		REST					x	x	x	x	x							
	ElasticHosts		KVM	x	x	REST		x		x												
	Flexiant	FlexiScale	Xen	x	x	SOAP/Query	x				x	x	x	x	x	x						
	GoGrid	Cloud Servers	Xen	x	x	Query	x	x	x		x	x	x	x	x	x			x	x		x
	IBM	Smart Business (Dev. and Trust)	RHEV	x	x	REST					x										x	x
	Rackspace	Cloud Servers	Xen	x	x	REST	x	x			x								x	x	x	x
	ReliaCloud	Cloud Servers		x		Query	x	x													x	
	Verizon	Computing as a Service	VMWare HA	x	x	REST	x															
	voxel	VoxCLOUD	Xen	x		REST	x	x			x	x	x		x	x						x
VPS.NET	Daily Nodes	Xen			REST		x														x	
Software	Cloud.com	CloudStack																		x		
	Enmaly	Elastic Computing Platform																				x
	Eucalyptus																		x	x	x	x
	Nimbula	Direktor																		x		
	OpenNebula																					x
	Red Hat	Enterprise Virtualization Manager																	x			
	VMWare	vCloud [Express]																	x	x		x
	VMWare	vSphere																			x	
verschiedene VPS-Dienste																			x	x	x	

Abbildung 2: Zugriffsarten und verfügbare Bibliotheken für Cloud Computing-Dienste

Neben den generischen Zugriffsarten sind in der Praxis sind die vorkonfigurierten Betriebssysteme, die der jeweilige Anbieter bereitstellt, sowie der physische Standort ebenfalls wichtig (siehe Abbildung 3).

		Betriebssysteme																		Regionen					
		Red Hat Enterprise Linux	Ubuntu	CentOS	SUSE Linux Enterprise	CloudLinux	Debian	Fedora	KnoPIX	FreeBSD	OpenSolaris	Arch	Gentoo	ClearOS	Oracle Enterprise Linux	MS SQL Server 2005	MS SQL Server 2008	MS Windows Server 2003	MS Windows Server 2008	US (Ost)	US (Zentral)	US (West)	Europa	Australien	Silberstein
Cloud Computing	Anbieter	Produktname	x	x	x		x	x			x	x	x		x	x	x	x	x	x		x	x		
	Amazon Elastic Compute Cloud		x	x	x																				
	Cloud Central Servers																								
	ElasticHosts			x	x			x	x	x	x							x	x				x		
	Flexiant FlexiScale			x	x			x												x					
	GoGrid Cloud Servers		x	x	x																				
	IBM Smart Business (Dev. and Trust)		x			x																	x		
	Rackspace Cloud Servers		x	x	x			x	x																
	ReliaCloud Cloud Servers			x	x			x				x													
	Verizon Computing as a Service		x	x	x																				
voxel VoxCLOUD			x	x	x		x	x																	
VPS.NET Daily Nodes			x	x		x	x																		

Abbildung 3: Verfügbare Betriebssysteme und Regionen von Cloud Computing-Diensten

5 Cloud-Speicher

Für die vorgestellten Zugriffsarten wird in diesem Abschnitt untersucht, ob und wie weit diese für vorhandene Cloud-Speicher Dienste umgesetzt wurden. Insbesondere folgende Fragen sind interessant:

- **BLOB oder Block** - Aufgrund der Flexibilität ist BLOB-Speicher für die meisten Aufgaben besser geeignet als Blockspeicher. Viele Vorteile von Blockspeicher (Einbindbarkeit/Ordnerhierarchie) können durch Tools auf BLOB-Speicher übertragen werden.
- **REST oder SOAP** - Im Vergleich der Übertragungskonzepte ist REST klar zu bevorzugen. „85 Prozent des Marktes wendet sich REST zu während SOAP verschwindet“ [Wen09], da RESTful APIs einfacher zu verwenden sind als SOAP APIs, ohne dabei wesentlich an Funktionalität einzubüßen.
- **API oder Bibliothek** - Die Abwägung zwischen den beiden Zugriffsarten ist im Grunde ähnlich wie bei Cloud Computing-Diensten, da ebenfalls nicht für alle Dienst-Programmiersprache-Kombinationen passende Bibliotheken verfügbar sind.
- **Wahl der Programmiersprache** - Gegenwärtig gibt es für Java die meiste Unterstützung, sei es in Form von Bibliotheken für einzelne Dienste oder Multi Cloud Bibliotheken. Mit Dasein, CloudLoop und jClouds basieren drei Multi Cloud Bibliotheken auf Java. Daneben ist PHP die Sprache, die sich durch ihre Verwendung in der simplecloud Bibliothek von den Sprachen C# bzw. .NET, Ruby und Python absetzt.

Abbildung 4 stellt die wichtigsten Eigenschaften der untersuchten Cloud-Speicheranbieter gegenüber.

			Zugriffsarten			Formate			Bibliotheken						Multi Cloud Bibliotheken						
			Web GUI	Terminal-Sitzung	API	XML	JSON	text/plain	Java	Python	Ruby	C#/.NET	PHP	JavaScript	Objective C	simplecloud [PHP]	iclouds [Java]	Dasein [Java]	CloudLoop [Java]	SMEStorage [REST]	
Anbieter	Produktname																				
BLOB-Speicher	Amazon	Simple Storage Service	x	x	REST/SOAP	x			x		x	x	x			x	x	x	x	x	
	Google	Storage for Developers	x	x	REST	x				x							x				
	Mezeo	Cloud Storage Platform		x	REST	x	x		x	x					x					x	
	Microsoft	Windows Azure Storage		x	REST	x			x				x	x			x	x	x		
	Nirvanix	Storage Delivery Network		x	REST/SOAP	x	x	x	x	x	x	x	x	x		x			x		
	Rackspace	Cloud Files	x	x	REST	x	x	x	x	x	x	x	x				x	x	x	x	
	weitere BLOB-Speicherdienste																	x	x		
Software	EMC	Atmos																x	x	x	
	Eucalyptus	Walrus																x			
	Oracle	Sun Cloud (eingestellt)																		x	

Abbildung 4: Zugriffsarten und Bibliotheken für Cloud-Speicheranbieter

6 Platform und Software as a Service

Im Vergleich zu Cloud Computing und Cloud-Speicher ist eine Differenzierung bei PaaS und SaaS schwieriger. Grund dafür ist zum einen, dass beide Ebenen darauf ausgelegt sind dem Benutzer Zugriffsarten, Ressourcenverwaltung und Konfiguration weitestgehend vorzuenthalten und zum anderen, dass hier individuelle Produkte einzelner Anbieter vertreten sind, die es nur begrenzt ermöglichen allgemeine Konzepte zu finden.

Von besonderem Interesse ist hier die Wahl der Programmiersprache. PaaS-Dienste können zwar prinzipiell ein breites Spektrum von Applikationen hosten. Die praktische Untersuchung (Abbildung 5) jedoch hat gezeigt, dass sehr wohl eine gewisse Sprachabhängigkeit besteht. Es ist zu erkennen, dass Java sowie Ruby überwiegend zu empfehlen sind. Entwicklerwerkzeuge für Python, PHP und für das .NET-Framework sind jedoch ebenfalls verfügbar.

			Zugriffsarten			Verfügbare Sprachen				
			Web GUI	Terminal-Sitzung	Remote Desktop	Java	Python	Ruby	C#/.NET	PHP
	Anbieter	Produktname								
PaaS	Microsoft	Windows Azure Storage	x		x	x		x	x	x
	Google	Google App Engine	x			x	x			
	Salesforce	Cloud Storage Platform	x			x				
	Engine Yard	Windows Azure Storage	x					x		
	Heroku	Storage Delivery Network	x	x				x		

Abbildung 5: Übersicht einiger PaaS-Dienste

7 Fazit

Das Anfangs vorgestellte erweiterte SPI-Ebenen-Modell spaltete die einfache Aufteilung von Cloud-Diensten in IaaS, PaaS und SaaS weiter auf, indem IaaS in Computing und Speicher aufgeteilt wurde. Letzterer wurde dabei zusätzlich in die verschiedenen Konzepte BLOB- und Blockspeicher unterteilt. Hier war festzustellen, dass BLOB-Speicher empfehlenswerter sind, da dort die Konzepte der Cloud besser umgesetzt werden.

Als Schwerpunkt wurden Zugriffsmöglichkeiten auf Dienste und ihre Umsetzungen untersucht. Bei der Untersuchung von IaaS-Diensten konnte festgestellt werden, dass neben dem Zugriff via Web GUI pro Anbieter zum Teil sogar mehrere APIs und Bibliotheken für verschiedene Programmiersprachen verfügbar sind. Durch Untersuchen von RESTful und SOAP-basierten API Anfragen konnte festgestellt werden, dass sich REST-konforme Anfragen besser für die Kommunikation mit Cloud-Diensten eignen, da sie tendenziell simpler zu verfassen und zu verarbeiten sind. (Multi Cloud) Bibliotheken sind weit verbreitet und unterstützen in erster Linie Java-Programmierer. Daneben ist PHP die Sprache, die sich durch ihre Verwendung in der simplecloud Bibliothek von den Sprachen C# bzw. .NET, Ruby und Python absetzt. Mit PaaS- und SaaS-Ebene endete die Untersuchung von Implementierungen und Zugriffsarten von und auf Cloud-Dienste, wobei hier aufgrund der Verschiedenartigkeit der Dienste nur begrenzt allgemeine Aussagen getroffen werden können.

In Bezug auf Standards bleibt festzuhalten, dass es momentan für die jeweilige SPI-Ebene keinen Standard gibt, der von allen Cloud-Anbietern umgesetzt wird. Dies stellt eines der größten Probleme dar. An vielen Stellen der Kommunikation zwischen Kunden und Anbietern gibt es Hilfsmittel oder Abstraktionsebenen, die diese fehlenden Konventionen kaschieren. Auch wenn bereits ein Annähern der Dienste aneinander zu erkennen ist und die Entwicklung von Standards langsam nennenswerte Ergebnisse hervorbringt [CDM10] [MKL09] ist davon auszugehen, dass es noch einige Jahre dauern wird, bis die Entwicklung und das Anerkennen von Standards abgeschlossen sind.

8 Danksagung

Besonderen Dank gilt der Professur für Kommunikationssysteme und Internet-Dienste der Universität der Bundeswehr München, allen voran Frau Prof. Dr. Gabi Dreö Rodosek sowie dem Munich Network Management Team für die hilfreichen Diskussionen und wertvollen Kommentare bei der Erstellung dieser Publikation. Darüber hinaus möchten wir uns bei der NATO SC/4 SMI AHWG und der Europäischen Verteidigungsagentur für die Unterstützung und Mitwirkung bedanken. Unsere Forschung in diesem Bereich wurde auch teilweise in Kooperation mit dem Bundesamt für Sicherheit in der Informationstechnik durchgeführt.

Literatur

- [AZS11] Windows Azure Storage — Windows Azure Platform. Online-Quelle, 2011. <https://www.microsoft.com/windowsazure/storage/default.aspx>.
- [Bia08] Randy Bias. Virtual, Cloud, Datacenters? Online-Quelle, April 2008. <http://cloudscaling.com/blog/cloud-computing/virtual-cloud-datacenters>.
- [CDM10] Storage Networking Industry Association. Online-Quelle, 2010. <http://cdmi.sniacloud.com/>.
- [Chi07] D. Chisnall. *The definitive guide to the xen hypervisor*. Prentice Hall Press Upper Saddle River, NJ, USA, 2007.
- [GGW09] Scaling Your Internet Business. Online-Quelle, März 2009. <http://www.gogrid.com/downloads/GoGrid-scaling-your-internet-business.pdf>.
- [Hes10] Kenneth Hess. Top 10 Virtualization Technology Companies. Online-Quelle, April 2010. <http://www.serverwatch.com/trends/article.php/3877576/Top-10-Virtualization-Technology-Companies.htm>.
- [jcl10] jclouds. Online-Quelle, 2010. <http://jclouds.org/>.
- [MKL09] T. Mather, S. Kumaraswamy und S. Latif. *Cloud security and privacy: An enterprise perspective on risks and compliance*. O'Reilly Media, Inc., 2009.
- [S310] Amazon Simple Storage Service (Amazon S3). Online-Quelle, 2010. <http://aws.amazon.com/s3>.
- [SfD11] Google Storage for Developers. Online-Quelle, 2011. <https://code.google.com/apis/storage>.
- [SFS11] Sales Cloud 2. Online-Quelle, 2011. <http://www.salesforce.com/crm/sales-force-automation/>.
- [SFW10] Force.com: Build and run business apps 5 times faster at half the cost. Online-Quelle, 2010. http://www.salesforce.com/assets/pdf/datasheets/DS_Forcedotcom.pdf.
- [TSW] White Paper Cloud Computing. Online-Quelle. http://download.sczm.t-systems.de/t-systems.ch/de/StaticPage/62/79/00/627900_White-Paper-Cloud-Computing.
- [War08] Keith Ward. More Azure Hypervisor Details. Online-Quelle, November 2008. <http://virtualizationreview.com/blogs/mental-ward/2008/11/more-azure-hypervisor-details.aspx>.
- [Wen09] Jerome Wendt. Archiving data to cloud storage: How to choose the right cloud storage provider. Online-Quelle, August 2009. http://searchstorage.techtarget.com/tip/0,289483,sid5_gci1365069_mem1,00.html.
- [Xig10] Getting Started - Market Data Feed - Financial Web Service - On-Demand - Xignite. Online-Quelle, 2010. <http://www.xignite.com/Support/GettingStarted.aspx>.
- [Zet10] Zetta Enterprise Network Storage Service. Online-Quelle, 2010. <http://www.zetta.net/zettaStorageService.php>.