



F5 White Paper

Herrscher über die Wolke: Anforderungen an Cloud-Computing

F5 Networks' Standpunkt zu Cloud-Computing:
Definition, Architektur und Entwicklung

von Lori MacVittie

Technical Marketing Manager, Application Services

Alan Murphy

Technical Marketing Manager, Virtualization

Peter Silva

Technical Marketing Manager, Security/Optimization

Ken Salchow

Manager, Technical Marketing



Inhalt

Einleitung	3
Das große Ganze	4

Cloud-Design	5
Dynamische Steuerebene	5

Cloud- Architektur	7
Utility-Computing	7
Framework-Computing	9
Unternehmens-Computing	11
Zusammenstellung	12
So wird die Cloud IT-fähig	13

Schlussfolgerung	14
-------------------------	-----------

Einleitung

Inzwischen ist es so gut wie unmöglich, eine Fach- oder Wirtschaftszeitschrift, Website oder Zeitung zu lesen, ohne dass einem eine Bemerkung zu Cloud-Computing in die Augen springt - um was es dabei geht, wie IT und Wirtschaft dadurch verändert werden. Die Auswirkungen von Cloud-Computing werden noch viele Jahre spürbar sein, egal, was letztendlich daraus werden wird. Trotzdem gibt es immer noch keine einheitliche Definition von Cloud-Computing, ein Manko, das sich aufgrund fehlender Diskussion, Planung und Implementierung in moderne Unternehmensnetzwerke auch nicht so schnell ändern dürfte. Definitionsversuche zu Cloud-Computing und Konzepte "der Cloud" sind äußerst unklar und schwer zu fassen. Nur wenige Hersteller sind bereit, über den Marketing-Hype und das "Cloud-Washing" vorhandener Produkte hinaus ihren Standpunkt zu vertreten und darzulegen, was für unter echtem Cloud-Computing verstehen, was es zur Zeit auf dem Markt gibt, was fehlt und welche Merkmale gebraucht werden, damit die Unternehmen von diesem dynamischen Ideologiewechsel beim Computing profitieren können.

Cloud-Computing ist keine Revolution, sondern eine Weiterentwicklung vorhandener Unternehmens-Computing-Architekturen und hat seinen Ursprung in den Anfängen des Netzwerk-Computing. Der Unterschied besteht darin, dass inzwischen gigantische Fortschritte bei der Virtualisierung praktisch aller Bereiche eines Datenzentrums gemacht wurden. Darüber hinaus hat sich inzwischen ein dynamisches Verständnis und das Bedürfnis entwickelt, steuern zu wollen, welche Services die Cloud wann und wie für die Nutzer dieser Services bereitstellt. Dieses neue dynamische Modell muss in der Lage sein, den Anwendungs- und Datenverkehr zu unterbrechen, den aktuellen Kontext zu interpretieren und der Cloud-Infrastruktur Anweisungen dazu zu geben, wie die Anforderung am effizientesten übertragen wird.

Die Cloud ist keine Punkt- oder Komplettlösung aus einer Hand, sondern ein ganzheitliches, aus verschiedenen Komponenten bestehendes Ökosystem, das einigen grundlegenden Anforderungen entsprechen muss, um die Bedürfnisse von Unternehmen erfüllen zu können. Zu diesen Anforderungen zählen Skalierbarkeit, Anpassungsfähigkeit, Erweiterbarkeit und Verwaltbarkeit. Darüber hinaus muss die Cloud zusätzliche Funktionalität aufweisen, die den Best-in-Class-Anforderungen eines Unternehmens entspricht, z.B. Funktionen für Sicherheit, Verfügbarkeit in Echtzeit und Performance.

Es bleibt jedoch die Frage, wie diese neue dynamische Computing-Architektur eigentlich aussieht und was es braucht - über die bereits heute verfügbaren Standardwerkzeuge hinaus -, damit sich etwas als "Cloud" qualifiziert.

Das große Ganze

Von erheblicher strategischer Bedeutung ist die Frage, wie die verschiedenen Teile der Infrastruktur integriert werden, um die Cloud zu bilden. Dazu gehört alles, von der nackten Hardware über die Benutzer bis hin zu sämtlichen Elementen dazwischen. Außerdem gibt es je nach Rolle unterschiedliche Möglichkeiten, die Interaktion verschiedener Operationen innerhalb der Architektur zu sehen.

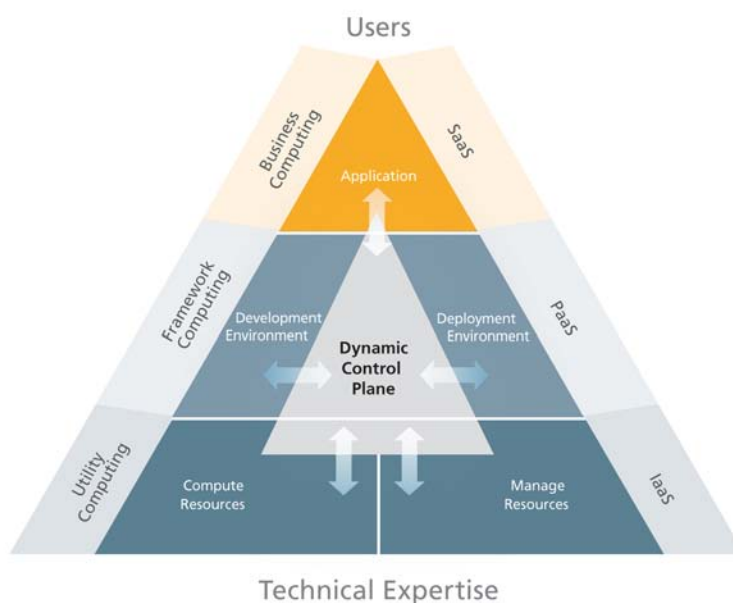


Abbildung 1: Cloud-Architektur

Die Cloud-Computing-Architektur besteht aus mehreren Funktionskomponenten (z.B. Rechenressourcen oder Bereitstellungsumgebungen), die in den verschiedenen Ebenen einer Pyramide organisiert sind. Die Breite dieser Ebenen steht für die Tiefe des technischen Know-hows, das erforderlich ist, um die jeweilige Ebene zu erstellen und/oder bereitzustellen. Die Pyramidenebenen entsprechen in etwa den Konzepten von Infrastructure as a Service (IaaS), Platform as a Service (PaaS) und Software as a Service (SaaS). An der Spitze der Pyramide stehen die Benutzer, die auf die Anwendungen zugreifen; im Zentrum befindet sich eine dynamische Steuerebene, die sich mit allen anderen Ebenen überschneidet und für Konnektivität in Echtzeit, Datenkoordination und Flusskontrolle zwischen den Ebenen sorgt.

Um optimal von einer Cloud-Architektur profitieren zu können, muss jede Komponente in irgendeiner Form vorhanden sein. Elemente der dynamischen Steuerebene z.B. sind auf jeder Ebene der Cloud-Architektur Voraussetzung dafür,



White Paper

Herrscher über die Wolke: Anforderungen an Cloud-Computing

dass die Cloud-Umgebungen effizient und on-Demand arbeiten können. Dazu ist ein Grad an Automatisierung und Orchestrierung erforderlich, der nur durch die Integration von Komponenten in die gesamte Architektur erreicht werden kann. Ohne diese Funktionalität können Unternehmen nicht von den Vorteilen dieses Modells profitieren. Wenn auch nur eine der Kernkomponenten nicht implementiert wird, ist aufgrund der fehlenden Zusammenarbeit keine echte Cloud-Architektur möglich.

Cloud-Design

Die Komponenten für den Bau der Cloud ähneln den Komponenten, die für den Bau einer herkömmlichen Architektur benötigt werden. Der Unterschied liegt in der Art und Weise der Integration, der Kommunikation und der Funktionsweise.

Man kann natürlich zu Recht einwenden, dass Web-Hosting-Services schon vor zehn Jahren die erste Implementierung von Software as a Service (SaaS) gewesen sind. Einige werden sogar argumentieren, dass dies auch die erste Platform as a Service (PaaS) gewesen ist, da dadurch eine HTML-Plattform zum Bau von individuellen Anwendungen zur Verfügung gestellt wurde. Und viele werden gleich noch dazusagen, dass die Rack-and-Power -Anbieter der Dot-Com-Ära Infrastructure as a Service (IaaS)-Lösungen angeboten haben.

Cloud-Computing entstand aus einem Single-Server, der für einen einzelnen Kunden bereitgestellt wurde und entwickelte sich dann über einen Hosting-Anbieter zu einem Anbieter für Business-Continuity und Disaster-Recovery. Irgendwann im Lauf dieser Entwicklung überwandene neue Technologien mit Geräten wie Load-Balancern, mit WAN-Optimierung, Komprimierung, Caching und Content Delivery Networks (CDNs) die physischen Einschränkungen. Dann war man so weit, dass die Geräte integriert werden konnten: eingebaute APIs, konsolidierte Server-Racks in Blade-Racks. Und schließlich wurde das Ganze automatisch bereitgestellt. Die Cloud-Architektur ist lediglich der logische Abschluss einer zehn Jahre dauernden Entwicklung. Inzwischen haben wir ein noch nie da gewesenes Maß der Virtualisierung von Hardware, Software, Netzwerk und Speicher erreicht und stehen kurz davor, das alles zu einem großen Ganzen zusammenstellen zu können.

Wie also sieht der letzte Schritt aus? Was ist der Unterschied zwischen einer Cloud und einer Pseudo-Cloud?



Dynamische Steuerebene

Bei herkömmlichen Traffic- und Computing-Systemen ist die Verarbeitung häufig in zwei getrennte Ebenen aufgeteilt: Datenebene und Steuerebene. Aufgabe der Datenebene ist die Beschaffung von Daten - sei es Input von einem System, oder, wie in unserem Fall, Anforderungen vom Benutzer - und die Rückgabe von Daten (Output, Dateien oder Antworten). Die Datenebene ist die grundlegende Konnektivität, die den Datenverkehr zu und von den Zielen abwickelt. Die Steuerebene dagegen beschäftigt sich eher damit, die Daten entsprechend Kontext und Richtlinien zu verwalten - sie verändert das "Wie" der Datenebene.

Die grundlegende Idee der Cloud-Architektur besteht darin, Benutzer - die unter Umständen mobil sind und zwischen LANs, WLANs und Internetverbindungen wechseln - und Services mit den Anwendungen zu verbinden, die diese benötigen. Auch die Anwendungen können je nach Anforderungen des Unternehmens von einem Cloud-Zentrum zum nächsten wechseln. Wenn Hardwareressourcen und Server anlaufen oder stillgelegt werden, wenn Anwendungen von der Entwicklung in die Produktion gehen, oder wenn komplette Anwendungen vom internen Datenzentrum zu einem Cloud-Anbieter verschoben werden, braucht die Cloud-Architektur eine dynamische Steuerebene, die die Daten überwacht und sicherstellt, dass diese kontinuierlich und bestmöglich verbunden sind. Die dynamische Steuerebene muss in der Lage sein, den Datenverkehr beim Durchqueren der Cloud abzufangen, die Daten zu interpretieren und die Cloud-Architektur anzuweisen, wie sie den Benutzer am besten mit der entsprechenden Anwendungsinstanz verbindet.

Abfangen

Die dynamische Steuerebene muss so platziert sein, dass der gesamte Datenverkehr zwischen dem Benutzer und der Anwendung und über die gesamte Cloud-Plattform hinweg für sie sichtbar ist. Ohne die Fähigkeit, Datenverkehr und Datenanforderungen abzufangen, kann die dynamische Steuerebene ihre eigentliche Aufgabe nicht richtig erfüllen. Sie muss nicht nur den Datenfluss selbst sehen können, sondern auch in der Lage sein, die Metadaten oder den Kontext des Datenverkehrs abzufangen. Die Datenebene und alle Komponenten, die innerhalb der Datenebene laufen, müssen für die dynamische Steuerebene sichtbar sein.

Interpretieren

Es genügt nicht, alle Informationen über den Daten- und Anwendungsfluss zu haben. Die dynamische Steuerebene muss in der Lage sein, die Elemente des Kontexts in Bezug auf die jeweilige Anforderung und Geschäftsrichtlinie



White Paper

Herrscher über die Wolke: Anforderungen an Cloud-Computing

und anderen Anwendungs- und Cloud-Verkehr zu verstehen. Die dynamische Steuerebene muss kontinuierlich Kontext und Richtlinie bewerten, um jederzeit intelligente Entscheidungen treffen zu können.

Anweisen

Wenn der dynamischen Steuerebene alle verfügbaren Informationen vorliegen und sie den Kontext analysiert hat, muss sie der Architektur Anweisungen geben, wie die beiden Endpunkte am besten miteinander verbunden werden. Darüber hinaus muss die Steuerebene mit der Infrastruktur - der Datenebene - kommunizieren, um das aktuelle Bereitstellungsmodell je nach identifizierten Bedürfnissen zu ändern. Dazu kann es erforderlich sein, Anforderungen an eine neue Instanz der Anwendung oder ein neues Datenzentrum zu schicken, die Komprimierung und Verschlüsselungseinstellungen zu ändern oder sogar andere Komponenten in der Architektur anzuweisen, Ressourcen zu schaffen oder herauszunehmen, die zur Bereitstellung dieser Anwendung oder Daten gebraucht werden. Darüber hinaus kann es auch notwendig sein, dass die dynamische Steuerebene auf der Grundlage von Richtlinien und Kontext schlicht und einfach den Zugriff verweigert.

Diese drei Punkte machen die Integration, auf die wir bereits hingewiesen haben, zwingend erforderlich und unterstreichen die Notwendigkeit, auf jeder Ebene und innerhalb jeder Komponente der Cloud-Architektur eine Schnittstelle zur dynamischen Steuerebene vorzusehen. Es spielt keine Rolle, ob es sich dabei um eine native, individuelle Integration oder lediglich um einen offenen Standard handelt, der vom Verbraucher verwendet werden kann, **die dynamische Steuerebene muss integriert werden, um jederzeit abfangen, interpretieren und anweisen zu können - sonst ist es keine Cloud.** Je besser diese Services bereitgestellt werden können, desto besser kann die dynamische Steuerebene intelligent und völlig automatisiert operieren, ohne ein manuelles (menschliches) Eingreifen.

Cloud-Architektur

Utility-Computing

Die Basisschicht der Cloud-Architektur besteht aus den Kernkomponenten, zu denen die Rechenressourcen und die Verwaltungsressourcen gehören, und erfordert das meiste technische Know-how für Bau und Bereitstellung. Auf diesem Fundament wird eine Cloud gebaut, und wie schon erwähnt besteht sie aus den Komponenten, die fast immer von Anbietern stammen, die ihren Kunden IaaS-Lösungen liefern.



White Paper

Herrscher über die Wolke: Anforderungen an Cloud-Computing

IaaS

Infrastructure as a Service (IaaS) ist ein Cloud-Computing-Modell, bei dem vorausgesetzt wird, dass die gesamte Infrastruktur als On-Demand-Modell bereitgestellt wird. Dieses Modell wird so gut wie immer in Form einer virtualisierten Infrastruktur mit virtualisierten Infrastruktur-Services umgesetzt, sodass der Kunde virtuelle Maschinen als Komponenten bereitstellen kann, die über eine Konsole verwaltet werden. Die physischen Ressourcen - Server, Speicher und Netzwerk - werden vom Cloud-Anbieter zur Verfügung gestellt, während die Infrastruktur, die auf diesen Komponenten bereitgestellt wird, vom Benutzer verwaltet wird. Hier ist anzumerken, dass der Benutzer von IaaS fast immer ein Team ist, das aus IT-Experten für die benötigten Infrastrukturkomponenten besteht.

IaaS nutzt die dynamische Steuerebene, um On-Demand-Skalierbarkeit durch das schnelle, automatische Provisioning von Rechenressourcen zu ermöglichen. Bei einer virtualisierten Architektur - die häufigste Form der IaaS-Architektur - beinhaltet dies die automatische Bereitstellung und das automatische Starten neuer Instanzen auf einer virtuellen Maschine. Es werden so viele Instanzen gestartet, wie gebraucht werden, um die Kapazitätsanforderungen zu erfüllen, wobei diese Instanzen wieder stillgelegt werden, wenn die Nachfrage zurückgeht.

In dieser Schicht der Architektur sind alle Komponenten dafür verantwortlich, umsetzbare Daten für die anderen Komponenten zur Verfügung zu stellen und bestimmte Aufgaben auszuführen, um ein automatisches Provisioning oder Decommissioning erfolgreich durchzuführen.

IaaS wird häufig als Utility-Computing angesehen, da es Computerressourcen ähnlich wie Strom, Gas und Wasser (engl. utilities) behandelt. Wenn die Nachfrage nach Kapazitäten steigt, werden vom Anbieter mehr Rechenressourcen vorgesehen. Sinkt die Nachfrage, wird die Menge an verfügbaren Rechenressourcen entsprechend reduziert. Dadurch kann die Cloud-Architektur sowohl "on-Demand" als auch "Pay-Per-Use" sein.

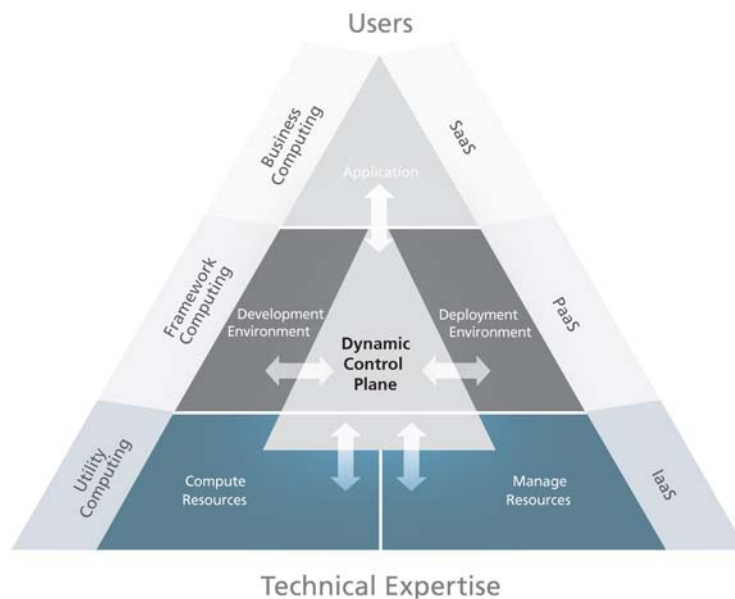


Abbildung 2: IaaS und Cloud-Architektur

Rechenressourcen

Rechenressourcen zählen zu den Basiskomponenten der Cloud - "nackte" Ressourcen wie CPU, Arbeitsspeicher und Festplatte -, die letztendlich dafür sorgen, dass Anwendungen, die innerhalb der Cloud gebaut wurden, auch funktionieren. Das kann z.B. ein Hosting-Service-Anbieter mit Hunderten oder Tausenden von installierten Serversystemen sein, die darauf warten, von Abonnenten benutzt zu werden, oder ein einzelnes Blade-Chassis mit extrem dichten Ressourcen zur virtuellen Segmentierung.

Diese Schicht weckt in der Regel Erinnerungen an einen herkömmlichen Server. Die modernen Systeme sind natürlich erheblich komplizierter und vielseitiger. Sie können eine große Menge an Prozesskernen und Arbeitsspeicher enthalten, aus denen sich virtuelle Systeme bauen lassen; Netzwerkschnittstellenkarten für Auto-Provisioning, die dynamisch von 10 MB bis zu mehreren Gigabit konfiguriert werden können; und direkt (Direct Attached Storage, DAS) oder an das Netzwerk (Network Attached Storage, NAS) angeschlossene Speichersysteme, um die Anforderungen der Anwendungssoftware zu erfüllen, die darauf installiert wird

Verwaltungsressourcen

Verwaltungsressourcen sind die Komponenten, die man braucht, um aus nackter Hardware betriebsfähige Serverplattformen mit entsprechender CPU, Arbeitsspeicher und Plattenressourcen zu machen, die zur Unterstützung der darauf installierten Anwendungen erforderlich sind. Darüber hinaus haben Verwaltungsressourcen auch die Aufgabe, die Ressourcenanforderungen zu überwachen und sicherzustellen, dass die Anwendung alle Rechenressourcen erhält, die sie braucht. Außerdem verschieben diese Ressourcen bei Bedarf die Anwendung oder beschaffen zusätzliche Ressourcen.



White Paper

Herrscher über die Wolke: Anforderungen an Cloud-Computing

Diese Komponente ist meist gleichbedeutend mit Systemen für VM-Verwaltung oder Software-Provisioning, die die nackte Hardware mit Betriebssystemen, Patches und Anwendungslogik ausstatten und für übergeordnete Netzwerk-Konnektivität (IP-Adressierung und mehr) sorgen können.

Zwischen der Schicht mit Rechenressourcen und den Verwaltungsressourcen müssen Daten ausgetauscht werden, damit keine Rechenressourcen verschwendet werden, die besser für eine andere Anwendung genutzt werden könnten.

Framework-Computing

Viele Anwendungen sind auf Software-Plattformen erstellt, die auf Infrastruktur-Services laufen. Diese Plattformen können Umgebungen wie z.B. Oracle oder ASP.NET sein und stellen für Unternehmen eine bequeme Möglichkeit dar, individuelle Anwendungen zu erstellen, ohne sich dabei um die mit den Plattformen zusammenhängenden Details kümmern zu müssen. Viele Plattformen basieren auf Standards - z.B. Java EE -, andere wiederum sind proprietär, u.a. Google AppEngine und Architektur-Frameworks, die von Unternehmensarchitekten entwickelt und bereitgestellt werden.

PaaS

Platform as a Service (PaaS) ist ein Cloud-Computing-Modell, bei dem eine bestimmte Entwicklungs- und Bereitstellungsplattform - z.B. Java EE, IBM WebSphere, Oracle, Google Apps, .NET, BizTalk - die Basis für die Bereitstellung bildet. Diese Clouds sind insofern proprietär, als nur Anwendungen, die für die jeweilige Plattform entwickelt wurden, in der Cloud bereitgestellt werden können.

PaaS ist eine Art Framework-Computing, da es sich bei der zur Verfügung gestellten Plattform um das Kern-Framework handelt, in dem Anwendungen entwickelt werden. Diese Anwendungen laufen auf keiner anderen Plattform und enthalten häufig plattformspezifische Erweiterungen oder Services, z.B. SimpleDB von Amazon, die nicht auf andere Umgebungen portiert werden können. Das Konzept des Framework-Computing stammt aus Architekturen, bei denen eine Schicht mit Funktionalität und Services zur Verfügung gestellt wird, sodass sich die Entwickler nicht um die damit zusammenhängenden Details kümmern müssen. Dieser Ansatz hat den Vorteil, dass Anwendungen schneller entwickelt und bereitgestellt werden können.

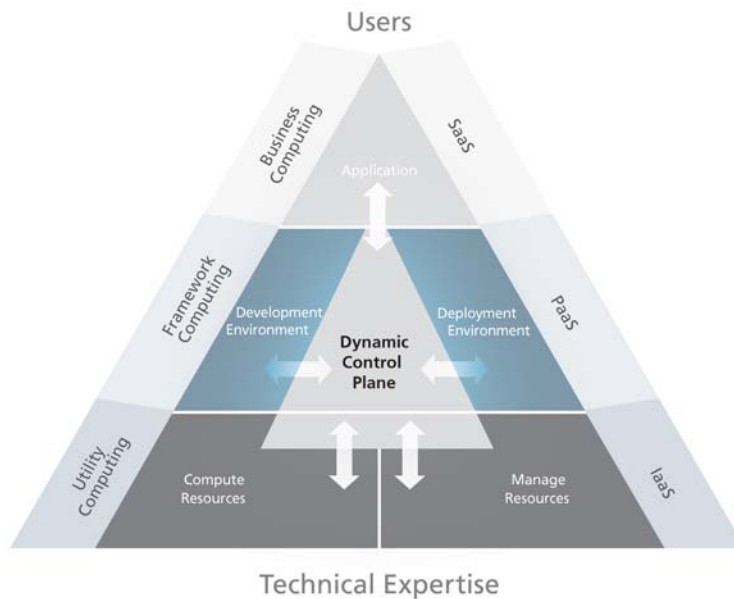


Abbildung 3: PaaS und Cloud-Architektur

Entwicklungsumgebung

Alle Plattformen benötigen eine Entwicklungsumgebung, in der die Anwendungen entwickelt, gebaut, getestet und validiert werden - außerhalb der Produktionsumgebung. Diese Entwicklungsumgebungen können herkömmliche integrierte Entwicklungsumgebungen (IDE) sein, die so konfiguriert sind, dass sie Ressourcen innerhalb einer PaaS-Umgebung bereitstellen, oder direkt als Teil des PaaS-Angebots integriert sein. In Umgebungen, die von Microsoft Visual Studio abgeleitet wurden, ist eine Verbindung zu internen und externen Instanzen einer Microsoft-spezifischen Plattform möglich, was die "Offline"-Entwicklung von Anwendungen ermöglicht, die in einer PaaS-Umgebung bereitgestellt werden sollen. Immer mehr Angebote, die weniger Standard und eher proprietär sind - Angebote, die vollständig von Ressourcen abhängen, die nur in der PaaS-Umgebung existieren, z.B. Force.com von Salesforce.com - stellen eine PaaS-gehostete Entwicklungsumgebung zur Verfügung, über die Entwickler ihre Lösungen bauen, testen und bereitstellen können.

Bereitstellungsumgebung

Die zweite Komponente in dieser Schicht wird benötigt, um die fertige Anwendung in der Produktion bereitzustellen, damit die Endbenutzer mit ihr arbeiten können. Im Grunde genommen ist dies die Run-Time-Umgebung, in der die Anwendungen bereitgestellt werden. Der Unterschied zwischen der Run-Time-Umgebung der PaaS und der von gehosteten oder sogar herkömmlichen, vom Unternehmen bereitgestellten Plattformen ist die On-Demand-Skalierbarkeit von PaaS, die bei den anderen Plattformen fehlt. Diese On-Demand-Skalierbarkeit ergibt sich z.B. aus der Bereitstellung der Umgebung auf einer generischen IaaS oder dadurch, dass Entwicklung, Bereitstellung und dynamische Steuerungsebene individuell aufgebaut und verbunden werden. Der zweite Ansatz führt zur Erstellung einer



White Paper

Herrscher über die Wolke: Anforderungen an Cloud-Computing

plattformspezifischen IaaS, bei der die benötigten Komponenten so angeordnet werden, dass sie Ressourcen abhängig von den individuellen Anforderungen der Bereitstellungsumgebung bereitstellen und stilllegen können.

Unternehmens-Computing

Ganz oben in der Pyramide steht das allgemeine Unternehmens-Computing. Dort finden sich die meisten Organisationen - vor allem Unternehmensorganisationen - wieder; sie können zwar eine Geschäftsanforderung identifizieren, aber weder eine Anwendung bauen noch eine Infrastruktur, auf der diese läuft. Anstatt sich beim Bau und/oder der Bereitstellung von Infrastruktur und Plattformen auf eine interne IT-Abteilung zu verlassen, suchen sich die Verantwortlichen einfach eine Anwendung und lassen diese laufen. Die meisten Unternehmen entscheiden sich für diese Option, da sich der finanzielle und personelle Aufwand zur Implementierung standardisierter Anwendungen nicht lohnt, die IT-Ressourcen von solchen Anwendungen nicht effizient genutzt werden oder schlicht das erforderliche Know-how im Unternehmen fehlt.

SaaS

Software as a Service (SaaS) ist ein Cloud-Computing-Modell, bei dem den Kunden vorgefertigte Anwendungen (z.B. CRM, SFA, Textverarbeitung, Tabellenkalkulation und HRM) über Web-Browser oder andere lokale Schnittstellen wie z.B. Anwendungen für mobile Geräte angeboten werden. Diese Anwendungen können in der Regel individuell angepasst werden, allerdings braucht sich der Kunde nicht um die zugehörige Infrastruktur, die Entwicklungsplattform oder die eigentliche Implementierung zu kümmern.

Obwohl es so aussieht, als wären die in einer SaaS-Cloud-Architektur bereitgestellten Anwendungen lediglich gehostete Anwendungen - z.B. das ASP-Modell der Dotcom-Ära/-Blase -, sind diese auf Unternehmens-Computing spezialisierten Architekturen häufig auf einer PaaS gebaut, die über eine IaaS bereitgestellt wird. Die Plattform, auf der die Anwendung bereitgestellt wird, kann vor dem Kunden verborgen bleiben, aber sie ist die Grundlage der mandantenfähigen Eigenschaften von SaaS und der Möglichkeit der individuellen Anpassung an Kundenwünsche. Die Plattform für Salesforce.com z.B. wurde in den letzten Jahren als getrennte PaaS namens Force.com vermarktet, die Kunden zusätzliche Optionen zum Bau individueller Lösungen bietet.

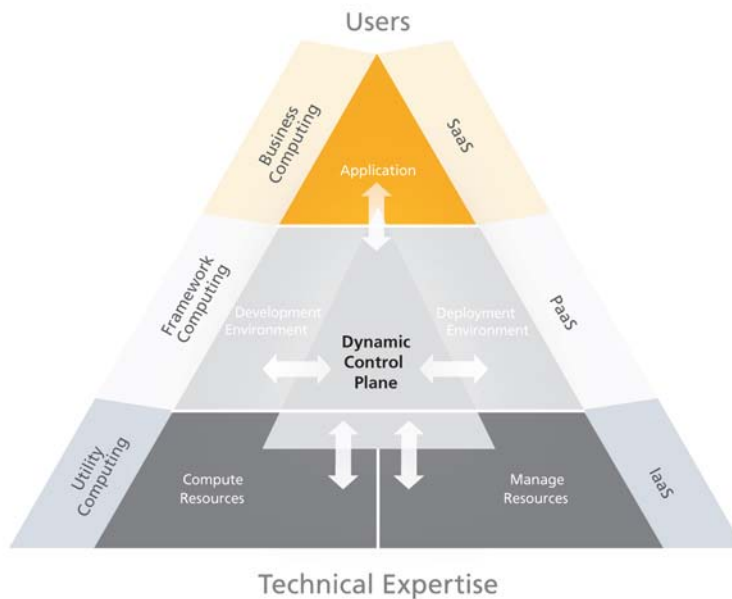


Abbildung 4: SaaS und Cloud-Architektur

Anwendung

Hier sind Anwendungen die einzige Komponente. Egal, ob es sich bei der Anwendung um das Ergebnis des Baus einer Utility-Computing handelt, gefolgt von einer Plattform, oder einfach um eine auf einem Server bereitgestellte Anwendung, das ist der Teil, mit dem die Benutzer interagieren. Den Benutzern ist egal, wie die Anwendung gebaut wurde, wo sie gespeichert ist oder welche Rechenressourcen für die Bereitstellung erforderlich sind. Sie erwarten einfach, dass die Anwendung verfügbar ist, wenn sie sie brauchen, dass sie reaktionsschnell ist und gut funktioniert, und dass sie sicher ist, egal, wo, wann und wie darauf zugegriffen wird.

Zusammenstellung

Je weiter wir von den Bestandteilen der Infrastruktur ganz unten in der Pyramide bis zur Spitze der Pyramide nach oben gehen, desto weniger Know-how wird gebraucht, um die einzelnen Komponenten zu erstellen. Das liegt ganz einfach daran, dass jede Schicht auf der vorherigen aufgebaut werden kann, ohne dass man die Schicht darunter voll und ganz verstehen muss. Ein Unternehmen mit begrenzten Infrastruktur-Fähigkeiten kann IaaS von einem Anbieter kaufen und auf dieser Infrastruktur seine eigene Plattform (oder gleich mehrere Plattformen) bauen, ohne über das Know-how zu verfügen, das man braucht, um eine Infrastruktur komplett neu zu erstellen. Im Managed-Hosting-Bereich ist diese Entwicklung schon seit Jahren zu beobachten. Das Unternehmen braucht keine eigenen Experten für Hardware oder Netzwerke und damit weniger technisches Know-how.

Idealerweise erstellt eine IT-Abteilung eine Reihe von Services, die die Anforderungen des Unternehmens erfüllen. Sie kann eine IaaS-Lösung innerhalb ihres eigenen Datenzentrums bauen, darauf eine (oder mehrere) PaaS erstellen und



White Paper

Herrscher über die Wolke: Anforderungen an Cloud-Computing

sogar fertige Anwendungen in einem Cloud-Kontext bereitstellen, was Flexibilität und Agilität ermöglicht. Auf diese Weise kann die Abteilung schnell und problemlos auf die wechselnden Anforderungen des Geschäftsbetriebs reagieren.

Auf die gleiche Weise können dann auch einzelne Geschäftsbereiche Lösungen auf der Grundlage ihrer individuellen Anforderungen und technischen Kompetenz bereitstellen. Sie können SaaS-Lösungen über einen externen Cloud-Anbieter bereitstellen oder mit intern verfügbaren Lösungen arbeiten. Eine andere Möglichkeit besteht darin, Anwendungen auf Plattformen zu erstellen und als eigene IaaS-Lösung bereitzustellen.

Mit der Cloud-Architektur können Unternehmen Lösungen an der Schnittstelle von IT und Geschäftsbetrieb bereitstellen. Angesichts der dynamischen, nicht-statischen Zuordnung von Anwendungen und Ressourcen - unabhängig davon, ob die IT-Abteilung die Anwendung komplett neu erstellt hat oder ob das Unternehmen lediglich seine eigenen Lösungen bereitstellt - kann ein Unternehmen mit Cloud-Architektur eine nahtlose Integration an dem Punkt bewerkstelligen, der für das Unternehmen am besten geeignet ist. Das Unternehmen kann daher die Steuerung in verschiedenen Bereichen selbst übernehmen (z.B. im Bereich Sicherheit und Compliance), ist aber gleichzeitig in der Lage, ein Höchstmaß an Agilität im Geschäftsbetrieb beizubehalten.

So wird die Cloud IT-fähig

Die neuere dynamische Cloud-Architektur hat zwar jede Menge Vorteile, muss aber trotzdem die grundlegenden Anforderungen von Unternehmens-IT erfüllen und - damit interne und externe Systeme zusammenarbeiten können - einheitliche und wiederverwendbare Verfahren vorsehen, mit denen voneinander abweichende Implementierungen auch bei den Cloud-Anbietern miteinander verbunden werden können. Wenn sich die Cloud und die dynamische Steuerebene in weiten Teilen der IT-Branche durchsetzen sollen, müssen sie eine unternehmenstaugliche Lösung sein und nicht nur bloße Theorie oder ein Sammelsurium an Hilfskonstruktionen und Einzelimplementierungen.

Bei der ursprünglichen Diskussion über die Konzepte von Infrastructure 2.0 (inzwischen ist daraus eine ständige Arbeitsgruppe entstanden, die sich mit der Formalisierung der Anforderungen an eine Cloud-Infrastruktur beschäftigt) gab es mehrere Schlüsselkomponenten, die als notwendig erachtet wurden, um die dynamische Steuerungsebene - ein entscheidendes Element von Infrastructure 2.0 - erstellen und verwalten zu können. Dazu gehörte eine Infrastruktur, die skalierbar,

anpassungsfähig, erweiterbar und verwaltbar sein muss. Es gibt jedoch noch weitere grundlegende Anforderungen der Unternehmens-IT, die berücksichtigt werden müssen. Jede Komponente, die einen derart massiven Einfluss auf Anwendungen und Daten ermöglicht, muss auch sicher sein und in Echtzeit operieren können; Abstriche bei der Sicherheit oder der Performance darf es nicht geben.

Die dynamische Steuerungsebene muss jederzeit verfügbar sein, um Benutzer mit den entsprechenden Ressourcen zu verbinden, was in Echtzeit und ohne Auswirkungen auf die Performance zu erfolgen hat. Trotz der Tatsache, dass die dynamische Steuerungsebene jede Benutzersitzung, jeden Datenzugriff und jedes Verschieben von Anwendungsverbindungen vermitteln und belegen muss, um unternehmenstauglich zu sein, darf dies nur mit wenig bis überhaupt keiner zusätzlichen Latenz erfolgen. Wie wir bereits ausgeführt haben, geht es bei einer IT-Architektur letztendlich um die Anwendung und die Benutzererfahrung dieser Anwendung. Das Endziel der Cloud-Architektur ist ein ausgewogenes Verhältnis zwischen dieser Benutzererfahrung und den Steuerungselementen und Richtlinien, die für den Geschäftsbetrieb erforderlich sind.

Schlussfolgerung

Cloud-Computing ist keine Revolution, sondern eine Entwicklung, die seit über zehn Jahren, wenn nicht schon seit den Anfängen der elektronischen Datenverarbeitung andauert. Die Cloud ist schlicht und einfach ein Architekturmodell, das viele der Komponenten, die heute weltweit in Datenzentren verwendet werden, auf flexiblere, reaktionsschnellere und effizientere Art einsetzt. Der grundlegende Unterschied besteht darin, dass diese Komponenten über eine dynamische Steuerungsebene miteinander verbunden sind, die die Architektur über die schnell wechselnden Anforderungen moderner Anwendungen, Daten und Clients informiert.

Die dynamische Steuerungsebene muss den Datenfluss beim Durchqueren der Cloud **abfangen**, die Daten **interpretieren** und die Cloud-Architektur **anweisen** können, wie sie den Benutzer am effizientesten mit der entsprechenden Anwendungsinstanz verbindet. Um jedoch problemlos im Unternehmen bereitgestellt werden zu können, muss sie auch skalierbar, anpassungsfähig, erweiterbar, verwaltbar und sicher sein, sowie Echtzeit-Performance ermöglichen. Und um diese dynamische Umgebung zu unterstützen, muss die Cloud mit diesen Vorgaben im Hinterkopf gebaut werden. Alle Komponenten - wie IaaS, PaaS, SaaS, Benutzer und Anwendungen - müssen so konzipiert sein, dass sie zusammen und als Teil der dynamischen Steuerungsebene arbeiten können.



White Paper

Herrscher über die Wolke: Anforderungen an Cloud-Computing

Es gibt keinen Zweifel daran, dass sich eines der Cloud-Computing-Konzepte in den nächsten Jahren zum bevorzugten Verfahren für die Bereitstellung von kritischen Anwendungen entwickeln wird. Und es gibt nur wenig Zweifel daran, dass der Schritt in Richtung Cloud-Architektur auch in Zukunft die Entwicklung der Werkzeuge fördern wird, die Unternehmen brauchen, um Geschäftsbetrieb und IT besser aufeinander abzustimmen. Abzuwarten bleibt lediglich, ob Anbieter und Hersteller eine unternehmenstaugliche, dynamische Steuerungsebene entwickeln können, damit sich das Bild zusammensetzt und Unternehmen von den angesprochenen Vorteilen profitieren können.

F5 Networks, Inc. 401 Elliott Avenue West, Seattle, WA 98119 888-882-4447 www.f5.com

F5 Networks, Inc.
Corporate Headquarters
info@f5.com

F5 Networks
Asia-Pacific
info.asia@f5.com

F5 Networks Ltd.
Europe/Middle-East/Africa
emeainfo@f5.com

F5 Networks
Japan K.K.
f5j-info@f5.com

WP-Dynamisch-IT 05/10

© 2010 F5 Networks, Inc. All rights reserved. F5, F5 Networks, the F5 logo, BIG-IP, FirePass, iControl, TMOS, and VIPRION are trademarks or registered trademarks of F5 Networks, Inc. in the U.S. and in certain other countries.