



□ Dr. Steven Abrams

(E-Mail: [sabrams@us.ibm.com](mailto:sabrams@us.ibm.com))

ist IBM Distinguished Engineer und definiert als Chief Architect für Rational Cloud Computing im IBM Rational Chief Technology Office die Rational Strategie für Cloud Computing und DevOps und ihre Integration in die Unternehmensstrategie der IBM. Als Mitglied des CTO-Teams war er unter anderem an der Entwicklung der Jazz Integration Architecture beteiligt.

## Optimierung des Application Lifecycles in der Cloud

Zurzeit ist viel die Rede von Cloud Computing und dem Ausmaß, in dem es zur Effizienzsteigerung in IT-Unternehmen oder IT-Abteilungen beitragen kann. Indem man sich auf Bereiche konzentriert, in denen die Auslastung der Infrastruktur niedrig ist und die Arbeitskosten hoch sind, kann man erheblich zur Optimierung des Entwicklungszyklus beitragen, was sich in niedrigeren Kosten, besserer Qualität und kürzerem Time-to-Market zeigt. Der Lebenszyklus von Anwendungsentwicklung und -tests ist dafür ein hervorragendes Beispiel. Wir erläutern, wie Cloud Computing eingesetzt werden kann, um den Lebenszyklus einer Anwendung umzugestalten, mit Schwerpunkt auf Implementierung, Installation und Tests von sich in der Entwicklung befindlichen Anwendungen. Außerdem behandeln wir Strategien, wie automatisierte Anwendungsweitergaben in der Pipeline von der Entwicklung über Tests bis hin zur Produktion genutzt werden können, während gleichzeitig die Zusammenarbeit über den Lebenszyklus der Softwareentwicklung und des Softwaredeployments hinweg insgesamt verbessert wird.

Wie Cloud Computing zur Effizienzsteigerung in IT-Unternehmen oder -Abteilungen beitragen kann, wird derzeit häufig diskutiert. Im gleichen Maße herrscht jedoch Verwirrung darüber, was Cloud Computing eigentlich ist und welchen Ansatz Unternehmen wählen sollten, um davon zu profitieren. Der Lebenszyklus von Anwendungsentwicklung und Tests ist für viele Unternehmen ein hervorragender Ausgangspunkt.

Wir werden zeigen, wie Cloud Computing die Implementierung, das Testen und das Deployment von Anwendungen vereinfachen kann. Als Erstes benötigen wir jedoch, um der Verwirrung Herr zu werden, eine gemeinsame Definition des Begriffs „Cloud Computing“. Es ist hilfreich, Cloud Computing aus zwei verschiedenen Perspektiven zu betrachten: „von innen nach außen“, d. h. aus technologischer Sicht, und „von außen nach innen“, d. h. aus geschäftlicher Sicht.

Aus technischer Sicht ist Cloud Computing das Zusammenfließen mehrerer inzwischen weit verbreiteter Technologien, die zusammen eine Vereinfachung, Standardisierung und Automatisierung von großen Teilen der IT-Infrastruktur ermöglichen. Im Kern beruht Cloud Computing auf Virtualisierung, einem Konzept, das bis in die Anfangszeiten der IBM Mainframes zurückreicht. Zwar ist die Allgegenwart der x86-Virtualisierung ein wichtiger Faktor im Cloud Computing, aber Cloud Computing ist wesentlich mehr als nur ein virtualisiertes Rechenzentrum.

Im Allgemeinen umfasst Cloud Computing die Kataloge mit standardisierten Konfigurationen von virtuellen Maschinen, die über eine automatisierte Self-Service-Schnittstelle bereitgestellt werden, verbunden mit Zähler- und Rechnungssystemen, die Leistungsverrechnungen auf Basis der verbrauchten Computerressourcen – z. B. virtuelle Maschinenstunden,

Speicher in GB oder Datenbanktransaktionen – pro Zeiteinheit ermöglichen.

Aus der Tatsache, dass diese ohne großen Aufwand bereitgestellt und gelöscht werden können, ergibt sich die „Elastizität“ des Cloud Computing. Die Abrechnungsmechanismen geben Benutzern Anreize, nicht genutzte Ressourcen freizugeben. Andernfalls müsste Cloud Computing wahrscheinlich als „erweiterungsfähig“ und nicht als „elastisch“ bezeichnet werden.

Aus geschäftlicher Sicht ist Cloud Computing einfach die Bereitstellung von IT-bezogenen Services über das Netzwerk (LAN oder Internet) mit einem verbrauchsbasierten Gebührenmodell. Dazu gehören Infrastructure as a Service (IaaS, z. B. virtuelle Maschinen, Netzkonnektivität und Speicher), Platform as a Service (PaaS, einschließlich Laufzeitstack mit Middleware und Datenbanken), Software

as a Service (SaaS, einschließlich Anwendungsservices wie E-Mail-Programmen und produktivitätssteigernde Office Tools) und auch Business Processes as a Service (BPaaS, z. B. Verarbeitung von Finanztransaktionen).

Ein wichtiges Merkmal ist die Tatsache, dass der Kunde nur für die Services bezahlt, die er in Anspruch nimmt, ohne sich jedoch für die zugrunde liegende Technologie zu interessieren (oder sich ihrer bewusst zu sein). Mit anderen Worten: Es kann sein, dass ein SaaS-Kunde im Bereich für E-Mail-Services für eine Reihe von E-Mail-Accounts bezahlt, aber weder weiß, noch sich dafür interessiert, ob es sich bei den zugrunde liegenden Systemen um virtuelle x86-Maschinen unter KVM (Kernel Virtual Machine) oder um virtuelle Power System-Maschinen unter IBM PowerVM handelt.

Unabhängig davon, wie man Cloud Computing betrachtet, können diese Services mit einer Kombination aus eigener und gemieteter, intern und extern gehosteter Infrastruktur bereitgestellt werden. Sie können von einem Drittanbieter oder von IT-Abteilungen eines Kunden zur Verfügung gestellt werden, wobei die zugrunde liegende Infrastruktur von einem einzelnen Kunden oder mehreren Kunden gemeinsam genutzt werden kann.

Hierbei kann die Infrastruktur innerhalb des Unternehmens oder außerhalb in einem gemeinsam genutzten Rechenzentrum untergebracht sein. Diese Optionen lassen sich auf einem Spektrum zwischen „öffentlichen“ und „privaten“ Clouds einordnen, verbunden mit den entsprechenden Kompromissen zwischen Datenschutz, Kontrolle und Kosten.

Ungeachtet davon, ob die technische oder geschäftliche Sicht zugrunde gelegt und ob eine dedizierte oder gemeinsam genutzte Infrastruktur eingesetzt wird, kann Cloud Computing dazu beitragen, die Effizienz in einem IT-Unternehmen oder einer -Abteilung erheblich zu steigern. Viele Unternehmen sind jedoch unsicher, an welcher Stelle sie mit der Anwendung von Cloud Computing beginnen sollen. Dabei erkennen viele Unternehmen, dass es zwischen dem Hype einerseits und der Unsicherheit andererseits einige Aspekte des Cloud Computing geben muss, die sich ohne großen Aufwand umsetzen lassen.

Wir haben festgestellt, dass der Lebenszyklus von Anwendungsentwicklung und Tests dafür ein hervorragendes Beispiel ist. Die Auslastung der Infrastruktur ist niedrig, die Arbeitskosten sind hoch und eine Automatisierung des Lebenszyklus bietet sich an. Cloud Computing kann erheblich zur Optimierung des Entwicklungszyklus beitragen, was sich in niedrigeren Kosten, besserer Qualität und kürzeren Markteinführungszeiten niederschlägt.

Studien[2008 IDC] haben gezeigt, dass in typischen IT-Unternehmen oder -Abteilungen zwischen 30 und 50 % aller Server dedizierte Testserver sind und die meisten dieser Server eine Auslastung von 10 % oder weniger aufweisen. Angesichts dieser vielen nicht ausgelasteten Server liegt der Schluss nahe, dass Unternehmen genügend Ressourcen für Tests zur Verfügung haben.

Dennoch ist eines der größten Hindernisse auf dem Weg einer Anwendung in die Produktion die begrenzte Verfügbarkeit von Servern, auf denen die Anwendung getestet werden kann, und der Testrückstand ist häufig der größte Faktor, aus dem sich das Deployment neuer Anwendungen verzögert. Das ist ärgerlich – trotz vieler dedizierter Testserver, die größtenteils nicht genutzt werden, können Teams ihre Anwendungen nicht schnell genug einrichten, um sie zu testen und zu implementieren. Die Ursache dafür ist im Allgemeinen die Schwierigkeit, Testumgebungen ordnungsgemäß zu implementieren und zu konfigurieren.

Nun untersuchen wir, wie Cloud Computing eingesetzt werden kann, um die Implementierung und Konfiguration von Anwendungen zu automatisieren. Im Bereich Entwicklung und Tests ist dies einer der ersten Schritte auf dem Weg zum Cloud Computing.

Als Erstes sollten Unternehmen eine relativ kleine Anzahl von gemeinsamen Mustern in Anwendungsinfrastruktur und Middleware ermitteln, die als Basis für eine Standardisierung dienen kann. Dies ist nicht einfach und erfordert die Mitarbeit der für Infrastruktur, Entwicklung und Betrieb zuständigen Teams. Dem gegenüber steht jedoch die Gewissheit, dass Umgebungsanforderungen, die dem Standardkatalog entsprechen, wesentlich schneller bearbeitet werden können.

Das Ziel einer 100 %-igen Abdeckung der Anwendungen in einem Unternehmen ist nicht realistisch. Ausgangspunkt sollten daher Anwendungen sein, die nicht zu den „IT-Kronjuwelen“ des Unternehmens gehören und bei denen der Aufwand für die Einhaltung einer Standardplattform in einem angemessenen Verhältnis zu den Vorteilen durch niedrigere Verwaltungskosten und schnellere Implementierung steht. Es kann sogar hilfreich sein, mit einer Plattformkonfiguration zu beginnen, die eine angemessene Anzahl von Anwendungen abdeckt.

Der nächste Schritt besteht darin, die Plattform und ihre Konfiguration in VM-Images zu erfassen und in den – je nach Anforderungen des Unternehmens privaten oder öffentlichen – Katalog des Cloud-Providers zu laden. Cloud-Verwaltungssysteme, mit deren Hilfe Sie das Deployment mehrerer virtueller Maschinen zusammen orchestrieren können, sind für diesen Zweck besonders gut geeignet.

Danach werden Automatisierungsskripts zum Konfigurieren dieser Umgebungen erstellt und in den Cloud-Katalog eingebunden. Anschließend können die Umgebungen auf Anforderung über eine Self-Service-Schnittstelle automatisch bereitgestellt und gelöscht werden, wenn sie nicht mehr gebraucht werden. Schließlich werden Skripts zum Implementieren der Anwendung auf der bereitgestellten Plattform geschrieben.

Es ist hilfreich, sich den Aufbau der virtuellen Maschinen in drei Ebenen vorzustellen. Auf der untersten Ebene befinden sich die Komponenten, die fest in das Image integriert sind, normalerweise das Betriebssystem und einige Middleware-Komponenten. Diese unterliegen häufig Unternehmensstandards – d. h. Richtlinien zu Betriebssystemen und Patch-Levels und Ähnlichem. Sie sind groß und ihre Installation ist kompliziert, aber sie ändern sich relativ selten.

Als Nächstes folgen die gemeinsamen Komponenten, die nach dem Bereitstellen der Instanzen auf den Images installiert werden. Für diese Komponenten gelten häufig andere Governance-Prozesse und Änderungsregeln als für die zugrunde liegenden Infrastrukturkomponenten. Dazu können beispielsweise Softwarekompo-

nenten gehören, die von anderen Teams innerhalb des Unternehmens oder Libraries, welche von Drittanbietern bereitgestellt werden. Um die Richtigkeit des Komponentenkatalogs zu gewährleisten, ist es sinnvoll, eine Definitive Software Library zu verwenden und dafür zu sorgen, dass die Automatisierungsskripts auf die genehmigten Versionen aus dem Katalog zurückgreifen.

Auf der obersten Ebene befinden sich die Komponenten, die sich am schnellsten verändern, nämlich die eigentlichen Anwendungselemente. Diese Komponenten und die Automatisierungsskripts, mit denen sie installiert und konfiguriert werden, unterliegen den Entwicklungsprozessen (tägliche Builds für Integration, Release-Kandidaten usw.). Dadurch bleibt die Software mit den Skripten, durch die sie bereitgestellt und konfiguriert wird, synchronisiert und beide können zusammen implementiert und getestet werden.

Es ist nicht einfach, diese Umgebungen und die für Bereitstellung und Konfiguration erforderlichen Automatisierungsskripts zu definieren, aber Benutzern stehen Technologien zur Verfügung, mit deren Hilfe sie die Zusammenfügung von virtuellen Maschinen und die Implementierung von Software auf diesen Maschinen modellieren können.

In einigen Fällen können die betreffenden Tools sogar aus den Modellen Skripten zum Automatisieren des Softwaredeployments generieren. Dies ähnelt der modellgestützten Entwicklung der eigentlichen Anwendungstopologien. Für eine weitergehende Automatisierung sind sowohl bei kommerziellen Anbietern als auch im Open Source-Bereich entsprechende Frameworks verfügbar, welche die Konfiguration von Middleware-Komponenten vereinfachen können.

Dies sind die Bausteine einer Cloud-basierten automatisierten Anwendungsimplementierung über den Lebenszyklus. Damit lässt sich der Prozess der Implementierung einer Testumgebung auf eine Self-Service-Anforderung über das Webportal der Cloud reduzieren. Die Cloud holt die vom Unternehmen als notwendig erachteten Genehmigungen ein und stellt danach automatisch die virtuellen Maschinen bereit – einschließlich Betriebssystem und deren Komponenten der unteren

Ebenen – und implementiert die abhängigen Komponenten aus der Software Library.

Die für Anwendungen, Tests oder Betrieb zuständigen Teams können anschließend die Anwendungselemente verteilen und dabei Automatisierungsskripts nutzen, deren Tests und Konfigurationsverwaltung mit den entsprechenden Vorgängen für die Anwendung zusammen ausgeführt werden. Damit kann ein äußerst fehlerträchtiger manueller Prozess, der mehrere Wochen – oder sogar Monate – dauern konnte, innerhalb von Stunden oder sogar Minuten durchgeführt werden.

Sobald ein Satz von Standardumgebungen definiert ist, können die von Entwicklern und Testern benötigten Tools direkt in den Cloud-Katalog integriert werden. Anschließend können Entwickler Anwendungen vom Desktop aus auf dynamisch bereitgestellten Cloud-Servern implementieren und Teams können Anwendungen automatisch direkt von ihren Build-Servern verteilen.

Dadurch, dass alle Teams ohne großen Aufwand Zugriff auf dieselben Umgebungen und Automatisierungen haben, wird sichergestellt, dass alle Benutzer Anwendungen mit derselben Konfiguration verwenden, was die Wahrscheinlichkeit von durch Fehlkonfigurationen verursachten Mängeln reduziert und die Reproduktion von Mängeln erleichtert. Dadurch verbessert sich die Zusammenarbeit zwischen Entwicklern und Testern.

Davon kann auch das für den Betrieb zuständige Team profitieren. Im Betrieb können allerdings nicht immer dieselben Umgebungen genutzt werden wie bei Entwicklung und Tests. Für Produktionssysteme gelten häufig besondere Anforderungen wie Hochverfügbarkeit oder Integration mit Überwachungs- und Sicherheitslösungen.

Indem die für den Betrieb zuständigen Teams Zugriff auf dieselbe Definitive Software Library von Komponenten und dieselben Automatisierungsskripts, wie die Entwicklungs- und Test-Teams haben, wird es ermöglicht, dass die Anwendungen trotz Unterschieden in den zugrunde liegenden Umgebungen richtig installiert und konfiguriert werden.

Unter der Bezeichnung „DevOps“ ist eine Reihe von Best Practices in der Entstehung begriffen, die zeigen, wie Cloud Computing eingesetzt werden kann, um diese Zusammenarbeit über den gesamten Lebenszyklus von der Entwicklung bis hin zum Betrieb auszudehnen. Ein echter DevOps-Ansatz geht weit über standardisierte Cloud-Kataloge und Automatisierungen hinaus.

Die hier beschriebenen Techniken können jedoch IT-Unternehmen und -Abteilungen helfen, in einem gewissen Ausmaß von DevOps zu profitieren, ohne die häufig umfangreichen Organisations- und Verfahrensänderungen durchführen zu müssen, die mit einer vollständigen Umsetzung von DevOps verbunden wären. Indem man mit einem Standardsatz von Anwendungsmustern beginnt, erleichtert man seinem Unternehmen den Weg zu DevOps, falls es diesen einschlagen sollte.

Das Automatisieren von Deployment und Konfigurationen für Anwendungen bietet Vorteile auf verschiedenen Ebenen. Zunächst werden Anwendungszyklen verkürzt, weil Testumgebungen ohne großen Aufwand verfügbar gemacht werden können. Außerdem neigen Entwicklungs- und Testteams weniger dazu, Server zu „horsten“, wenn sie wissen, dass sie ordnungsgemäß konfigurierte Umgebungen auf Anforderung erhalten und besonders, wenn ihnen die Nutzung der Ressourcen in Rechnung gestellt wird.

Dadurch werden Verbesserungen bei der Auslastung erzielt und damit die Kosten gesenkt. Die Qualität wird ebenfalls verbessert: Die schätzungsweise 30 % aller Mängel, die auf falsch konfigurierte Umgebungen zurückzuführen sind, können verringert und möglicherweise sogar vollständig beseitigt werden, wenn Tester und Entwickler in der Lage sind, sich auf wirkliche Qualitätsprobleme zu konzentrieren und die Testabdeckung zu verbessern.

Der Zeitaufwand für das Einrichten und Konfigurieren – und wiederholte Neukonfigurieren – von Testumgebungen ist häufig der wichtigste Faktor im Testzyklus. Wenn sich die Implementierungszeit von Tagen oder Wochen auf Stunden oder Minuten verkürzen lässt, können Tester mehr Zeit mit dem eigentlichen Testen der Anwendung verbringen. Die Wartungs-

kosten werden gesenkt: Mit einem Standardsatz von VM-Images verringert sich der Arbeitsaufwand, der erforderlich ist, um alle Testserver auf den richtigen Patch-Levels – z. B. bei Betriebssystem und Middle-ware – zu halten.

Wir haben uns auf die Frage konzentriert, wie die Technologie des Cloud Computing helfen kann, Konfiguration, De-

ployment und Tests von Anwendungen in Unternehmensumgebungen zu vereinfachen. Sie trägt dazu bei, Kosten zu senken, die Qualität zu verbessern und die Markteinführungszeit zu verkürzen. Da Unternehmen heutzutage von ihren IT-Abteilungen verlangen, mit weniger Kosten mehr zur Wertschöpfung beizutragen, ist eine sorgfältig geplante Cloud-Implementierung, beginnend mit dem Entwick-

lungs- und Testzyklus, eine ernsthaft zu erwägende Lösung. ■

## Referenzen

**[2008 IDC]** - 2008 IDC report: „Industry Developments and Models -- Global Testing Services: Coming of Age“ und „IBM Internal Reports“

---