

## Cloud Transformation

# Wie der Umzug in die Cloud gelingt

von Bodo Clausen und Pascal Stieber

Wissen Sie noch, wie Ihr letzter Umzug verlaufen ist? Haben Sie in tage- oder wochenlanger Arbeit Kisten vorbereitet, Geschirr verpackt und sich Utensilien wie Sackkarre, Fließdecken und Möbelroller zugelegt? Oder sogar Spezialwerkzeug angeschafft, zum Beispiel für den Aufbau einer Küche oder die Montage von Badezimmerarmaturen? Vielleicht haben Sie sich neben dem offensichtlichen „Doing“ – also dem Umzugsvorgang – auch eine Reihe organisatorischer Gedanken gemacht: Wann würde z. B. der Bodenleger seine Arbeit verrichtet haben, sodass Sie die Wand tapezieren oder den Raum einrichten können? Und konnten Sie für den Umzugstag rechtzeitig eine Reihe zupackender Freunde als Helfer gewinnen? Oder haben Sie sogar ein Umzugsunternehmen beauftragt?

Wenn Sie sich jetzt wiedererkennen, wären wir lieber bei Ihrem Umzug dabei gewesen als bei einigen, die wir miterleben durften. Die waren deutlich weniger gut vorbereitet. Eine solide Organisation hilft nicht nur beim Wohnungsumzug: Auch beim Umzug in die Cloud kann eine laxe Vorbereitung Chaos verbreiten, was sich auf das Unternehmen schnell geschäftsschädigend auswirken kann. Die Einflussfaktoren eines Cloud-Transformationsprojekts auf Organisation, Betrieb und Prozesse des Unternehmens sind vielfältig.

Was ist bei der Vorbereitung zu beachten? Welche Fragen sollten Sie sich unbedingt im Vorfeld stellen? Und welche Tools und Methoden können bei den ersten Schritten in die Cloud weiterhelfen? In diesem Artikel geben wir Ihnen Anregungen aus eigener Erfahrung und nehmen Sie außerdem mit auf einen Exkurs zur Kostenfrage am Beispiel einiger Angebote von Amazon Web Services.

## Welche Systeme eignen sich für die Cloud?

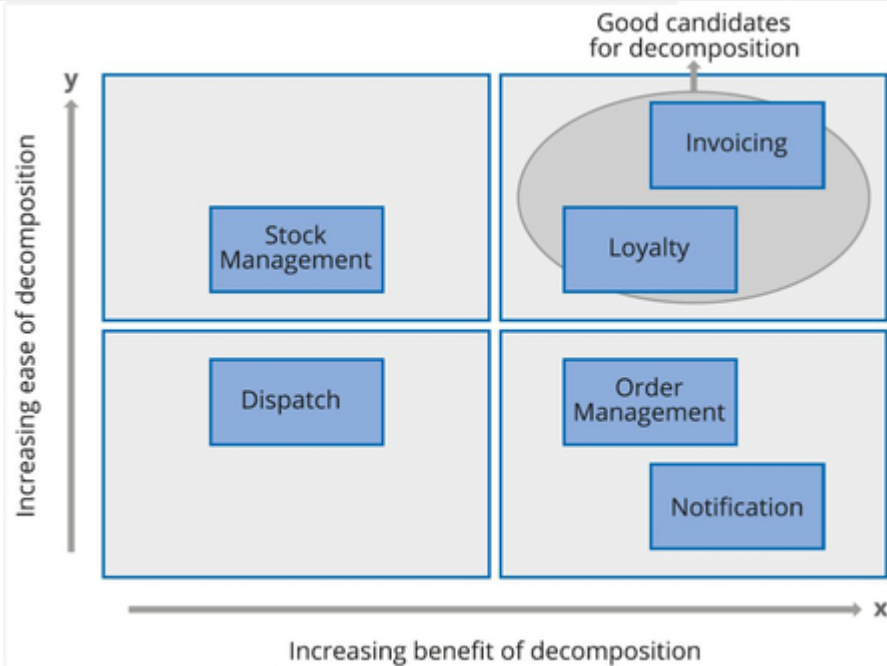


Abbildung 1: Zwei-Achsen-Modell von Newman zur Bestimmung von Komponenten für Transformationsvorhaben (Quelle: [1])

Transformationsprojekte bedeuten in den meisten Fällen Anpassungen der Architektur der bestehenden Systeme. Dabei sollte bei der Menge der zu verändernden Bestandteile Augenmaß bewahrt werden. Zu viele parallele Veränderungen innerhalb eines Unternehmens können riskant sein und den Erfolg eines Projekts gefährden. Durch Abhängigkeiten zwischen den Projekten ergeben sich Wechselwirkungen, die unbeabsichtigt Zeit und Ressourcen binden. Ein bewährtes Hilfsmittel für das Vorgehen bei Auswahl und Priorisierung geeigneter Komponenten stellt das Zwei-Achsen-Modell von Sam Newman [1] dar (siehe **Abbildung 1**).

Demnach eignen sich für die Ablösung monolithischer Systeme Systemkomponenten mit wenig Abhängigkeiten besonders gut. Diese rangieren in Newmans Zwei-Achsen-Modell entlang der Y-Achse relativ weit oben. Zugleich sollte durch die Ablösung ein hoher Benefit zu erwarten sein, der auf der X-Achse gemessen wird. Die Komponenten der sich im Einsatz befindenden Altsysteme sind in diesem Modell rechts oben zu finden. Diese Komponenten könnten beim Umzug in die Cloud also ausgemustert werden und dabei helfen, sogenannte „technische Schulden“ mit hoher Effizienz abzubauen. Bei der Evaluierung von neuen technischen Frameworks kann der Fokus entsprechend auf diesen Teil der Applikation gelegt werden. Welche weiteren Faktoren einen positiven Einfluss auf den Projekterfolg bei der Transformation in die Cloud haben können, wird nachfolgend beschrieben.

## Realistische Ziele zur Erreichung des Erfolgs

Im Vorfeld eines Transformationsprojekts kann ein Proof-of-Concept als Test- und Lernobjekt des abzulösenden Systems dabei helfen, ein erstes Gespür für die neuen Cloud-Technologien zu bekommen.

Dazu kommt die Frage, welche realistischen nichtfunktionalen Qualitätsanforderungen Sie an die zu entwickelnde Applikation stellen. Bei der Antwort ist man schnell bei Merkmalen wie hochverfügbar (mindestens 99,9 %), hochskalierbar und strapazierfähig. Diese Anforderungen erfüllen diverse Cloud Services nativ mithilfe von Best-Practice-Architekturen. Die Amazon Web Services (AWS) haben hier zum Beispiel das Well-Architected Framework im Angebot, das auch die Umsetzung von verteilten Architekturen unterstützt.

## Die Fachabteilungen einbeziehen

Unserer Erfahrung nach wird der Fokus häufig zu sehr auf die Technologie gelegt, der Blick aufs Geschäftsmodell und das zu entwickelnde Produkt hingegen vernachlässigt. Business-Requirement-Methoden wie EventStorming und DomainStorytelling helfen, mittels kollaborativer Ansätze das gewünschte Fachwissen der Domain-Experten zu synthetisieren und zu verteilen. Angereichert mit Domain Driven Design (DDD) kann das erarbeitete Domänenwissen mit geeigneten Frameworks (wie z. B. Vlingo oder Axon) in kompilierbaren Code überführt werden. Der Mehrwert der genannten kollaborativen Ansätze sollte jedoch schon bei dem Wissensaufbau der Business Domain erkennbar werden und kann zusammen mit dem Zwei-Achsen-Modell von Newman

ebenfalls als Werkzeug bei der Identifizierung von geeigneten Systemkomponenten für ein neues Cloud-Native-Produkt dienen.

## Die Mitarbeiter „mitnehmen“

Die Helfer bei Ihren Wohnungsumzügen waren sicherlich meist in bester Stimmung. Schließlich dauert so ein Umzug in der Regel nur einen Tag und Sie haben einen Helferkreis eingeladen, dem Sie vertrauen und der sich bereits kannte. Auch in der Arbeitswelt spielen soziale Einflussfaktoren eine große Rolle, was in soziotechnischen Systemen leider häufig unterschätzt wird. Doch am Ende ist es ein Mitarbeiter, der mithilfe der vielen Technologien, die uns ein Cloud Provider zur Verfügung stellt, das Unternehmensprodukt entwickelt. Die Möglichkeiten, die uns die Cloud anbietet, scheinen schier unendlich. Technisch steht der hochverfügbaren, hochskalierbaren und strapazierfähigen Anwendung also nichts im Wege. Doch die besten technologischen Möglichkeiten der Cloud Frameworks bleiben ungenutzt, wenn Mitarbeiter nicht an diese andocken können.

## IT der zwei Geschwindigkeiten

Neben Skillset und Motivation der Mitarbeiter heißt es jedoch auch, Unternehmensprozesse und Vorgehen auf die Cloud-Native-Entwicklung auszurichten. Eine Organisationsänderung mit dem Fokus auf IT ist gerade in großen Konzernen oder Behörden nicht so einfach möglich. Die Frage, wie die Organisation auf neue Technologien bzw. die mit Cloud Computing einhergehenden Veränderungen vorbereitet werden kann, sollte hier eindringlich gestellt werden.

Ein Lösungsansatz führt zur sogenannten „Two-Speed-Architektur“ [2]. Hier werden der Fortbestand und eventuell kleinere Weiterentwicklungen von bestehenden Systemen gesichert, ohne auf Innovation durch neue Vorgehensmodelle und Technologien verzichten zu müssen. Dafür initiiert man einen parallelen Betrieb mit ausgewählten Ressourcen, der autark von bestehenden Prozessen abläuft. Wem das noch nicht ausreicht, der kann über eine Neuausgründung aus dem bestehenden Unternehmen im Bedarfsfall nachdenken.

## Agile Methoden und DevOps

So unterschiedlich Fitness und Motivation Ihrer Umzugshelfer sein können, so verschieden sind auch Know-how- und Motivations-Level der Mitarbeiter des Transformationsprojekts. Auch hier kann die „Two-Speed-Architektur“ helfen, geeignete Ressourcen aus der Regelorganisation herauszuziehen.

Neben den richtigen Schlüsselpersonen sind kurze Feedbackzyklen essenziell, um das Produkt so früh wie möglich in Betrieb zu nehmen und sich einen Wettbewerbsvorsprung zu sichern. Die Methoden der agilen Softwareentwicklung ermöglichen schnelles Lernen, Wissensaufbau und Wissensvermittlung – und das sind nur einige Vorteile, die den Einstieg in komplexe Themen wie Cloud-Architekturen mithilfe von agilen Entwicklungsmethoden sehr erleichtern können.

Ergänzend zur agilen Softwareentwicklung haben sich hier DevOps-Ansätze bewährt. Der Einsatz und die Verankerung von DevOps in der Organisation lässt sich durch ein Tool wie das Site Reliability Engineering Framework von Google [3] vereinfachen. Mit Service Level Indicators wird eine gewünschte Kennzahl identifiziert (z. B. die Verfügbarkeit). Mit Service Level Objectives lässt sich auf dieser Basis ein Zielwert definieren (z. B. 87 %), der auf die Kennzahl angewendet werden kann. So ist es möglich, ein Produkt stetig zu verbessern und manuellen Aufwand für Betriebsprozesse weitestgehend abzubauen. Das Framework fördert zudem eine offene Fehlerkultur, welche die Identifikation des Teams mit den abgelieferten Softwarekomponenten stärkt und am Ende der Qualität zugutekommt.

## Architekturmodelle in der Cloud

Betrachtet man Cloud-Transformationsprojekte mit dem Fokus auf die Integrationsszenarien ergeben sich drei Möglichkeiten.

### 1. Cloud-Only

Hier sind alle Bestandteile der Architektur des Zielsystems in der Cloud angesiedelt. Dies ist mit Sicherheit eine spannende Option. Viele Transformationsprojekte werden sich mit der Zeit in diese Richtung entwickeln, nur spielt dieses Szenario in der Masse der Projekte zurzeit eine untergeordnete Rolle. Die Umstellungen erfolgen zunächst meist sukzessive.

## 2. Hybrides Cloud-Projekt

In diese Richtung gehen momentan die meisten Projekte. Die Bestandsarchitektur mit vielen Rechnern im eigenen Rechenzentrum ist bereits vorhanden und wird durch Cloud Services erweitert. Die Ziele rangieren von der Modernisierung der Architektur und des Werkzeugkastens über die Steigerung der Flexibilität bis hin zu Kostenoptimierungen. Dieses Szenario liegt auch dann vor, wenn man bereits verschiedene Systeme in der Cloud nutzt und diese erweitert.

## 3. Multi-Cloud-Hybrid

In dieser Architektur werden On-Premises-Bestandteile mit Cloud Services von verschiedenen Herstellern gemischt. Dies kann daran liegen, dass z. B. bei größeren Konzernen verschiedene Bereiche erfolgreich in verschiedenen Clouds arbeiten und eine Migration auf eine einfachere Architektur zu aufwendig wäre. Häufig nutzen unterschiedliche Unternehmensbereiche die Vorteile verschiedener Cloud-Anbieter, je nach Relevanz für den jeweiligen Bereich. Der Nachteil ist eine erhöhte Komplexität vieler Themen, wie beispielsweise verschiedene Abrechnungssysteme und vielfältigere Schnittstellen, wenn man in drei oder vier Welten zu Hause ist. Dementsprechend mehr Know-how müssen auch die Mitarbeiter mitbringen beziehungsweise aufbauen. Ein Vorteil ist die geringere Abhängigkeit von einem Anbieter. Wenn Know-how für zwei oder mehr Cloud-Welten vorhanden ist, kann es natürlich später einfacher sein, von einem Anbieter zum nächsten zu wechseln.

Schlussendlich werden in allen drei Szenarien ein oder mehrere Cloud Services genutzt. Es ist also immer auf die veränderten Rahmenbedingungen durch die Cloud zu achten.

Auch hier bietet sich unsere Umzugsanalogie an: So ließe sich das zweite Szenario des hybriden Cloud-Projekts mit einem Umzug vergleichen, bei dem sich die Familie ein Ferienhaus am See kauft und einen Teil des aktuellen Haushalts dorthin bringen möchte. Dabei stellen sich Fragen wie: Was bleibt im Erstwohnsitz, und was brauchen wir für das Ferienhaus? Wenn es ein Haus am See ist, sollten wohl definitiv Badesachen vor Ort sein. Es gibt allerdings auch Gegenstände wie eine Waschmaschine, die sowohl am Erstwohnsitz als auch im Ferienhaus eine Bereicherung darstellen.

In der Cloud heißt das: Gewisse Funktionalitäten werden sowohl im eigenen Rechenzentrum als auch in der Cloud benötigt. Das können Datenbanken, aber auch Datenverarbeitungssysteme sein. In vielen Projekten werden bestehende On-Premises-Systeme mit der Cloud verbunden. Hier gibt es verschiedene Ansätze und Herausforderungen sowie eine Vielzahl an Tools, die noch weitere Möglichkeiten eröffnen. Womit man beginnt, hängt dabei von den individuellen Erfordernissen ab. Zusätzlich spielen Erfahrungen und Zusammensetzung des Teams eine Rolle, das die Umsetzung durchführt.

Wir haben gute Erfahrungen mit multifunktionalen Teams gemacht, bestehend aus Administratoren, Entwicklern, Testern, Operations-Mitarbeitern bis hin zu den Business Analysten und Data Scientists. Diese Teams können andere Wege zurücklegen als ein Zwei-Personen-Team.

## Exkurs: Die Cloud als Kostensenker!?

Viele denken, dass die Cloud – neben den Möglichkeiten, die sie technologisch bietet – auch preislich sehr attraktiv ist. Doch ist das wirklich so? Und wie ermittle ich die Kosten für ein konkretes Projekt?

Selbst bei kleinen Projekten kann es eine echte Herausforderung sein, die genaue Höhe der Kosten zu ermitteln. Natürlich gibt es Kalkulatoren, die dabei helfen. Doch die Bestimmung der Kosten bleibt aufwendig, weil sehr viele oft unbekannt Parameter berücksichtigt werden müssen. So verursacht der Datentransfer Kosten. Zusätzliche Kosten werden durch Daten verursacht, die im Simple Storage Service gespeichert sind. VPN-Verbindungen, Elastic-Cloud-Compute-Instanzen (EC2) und viele andere Produkte verursachen ebenfalls Kosten.

### *Hauptkostentreiber*

Deshalb identifizieren wir in der Regel zunächst die Hauptkostentreiber. Dazu gehören normalerweise die Rechenleistung, also EC2-Instanzen. In vielen Fällen wird Elastic Map Reduce (EMR) eingesetzt, was die EC2-Instanzkosten um 10 % steigert. Für diese treibenden Kosten schätzen wir die ungefähren Laufzeiten und multiplizieren diese mit den Kosten für EC2 und EMR. Damit haben wir einen ersten Anhaltspunkt. Ein weiterer wichtiger Kostentreiber sind die Datenbanken. Viel mehr lohnt sich an dieser Stelle noch gar nicht zu berechnen.

### *Budgets*

In der AWS Cloud gibt es beispielsweise die Möglichkeit, das Erreichen von selbst definierten Grenzen zu überwachen. Mittels „Budgets“ kann in diesem Fall z. B. eine Information per Mail erfolgen. Wenn die grobe Überschlagsrechnung zu den Dimensionen des vorhandenen Budgets passt, sollte ein entsprechendes Monatsbudget gesetzt werden. Zusätzliche weitere Budgets, die entsprechend auf Wochen umgerechnet sind, werden eingetragen. Somit erfolgt bei Annäherung an eine Grenze des virtuellen Wochenbudgets eine Information per Mail. Wenn dies frühzeitig erreicht wird, muss entsprechend gehandelt werden. So lässt sich verhindern, dass das geplante Monatsbudget bereits nach ein paar Tagen verbraucht ist.

#### *Service-Pakete*

Speziell in der Cloud gibt es mannigfaltige Optimierungsmöglichkeiten durch eine große Anzahl an Services. Durch die Kombination unterschiedlicher Services sind auch verschiedene Wege zu einer Lösung denkbar. Dabei können die unterschiedlichen Lösungswege unterschiedliche Kosten bedeuten. Am Beispiel von EMR lässt sich das sehr gut erläutern:

EMR ist die eigene Hadoop Distribution von AWS. Hier sind verschiedene Tools wie Jupyterhub, HDFS, Spark und mehr auswählbar. Die Tools werden auf einem Cluster installiert und konfiguriert. Die Cluster werden generiert, verrichten ihre Aufgaben, z. B. die Verarbeitung von Daten, und werden danach terminiert.

Dies bedeutet, wenn der Cluster für die Verarbeitung eine Stunde benötigt, dann bezahlt man diese Stunde und natürlich die Zeiten, in denen der Cluster erstellt und terminiert wird. Dazu kommt, dass die grundsätzliche Leistungsfähigkeit der Instanztypen Einfluss auf die Netzwerkanbindung hat. In der On-Premises-Welt war dies nicht in diesem Maße der Fall, da dort jeder Server mindestens eine Netzwerkkarte mit einer Bandbreite von einem Gigabit hatte. In der Cloud hingegen ist es von Vorteil, die Instanzen nach ihrem Grundtyp beispielsweise „speicheroptimiert“ auszuwählen und hierfür eine aus der Erfahrung passende Speichergröße zu verwenden. Im Betrieb kann später via Monitoring ermittelt werden, ob kleinere oder größere Instanzen benötigt werden und wie die Auslastung des Netzwerkes ist.

#### *Dynamische Kostenmodelle*

Die Auslastung ist ein zentraler Punkt: Die Server müssen nicht mehr so groß wie möglich dimensioniert werden, um Lastspitzen abzufangen, sondern sie werden genau passend für die zu erwartende Last eingesetzt. Sollte es zum Monatsende eine größere Verarbeitung geben, dann wird genau an diesen Tagen ein größeres Cluster verwendet als an den übrigen Tagen. Wenn die Lasten nicht planbar sind und schwanken, können dynamischere Lösungen mit Loadbalancern und Autoscaling-Groups verwendet werden. Dann kann die Rechenleistung dynamisch durch die automatische Hinzu- oder Wegnahme von Instanzen angepasst werden.

Bei einem Kunden konnten wir so erhebliche Kosten einsparen. Wir hatten folgende Berechnungen durchgeführt:

Wenn wir bei AWS zwölf xlarge-m5-Instanzen mit Elastic Block Storage in Frankfurt verwenden, kostet dies On-Demand zurzeit 2,76 Dollar pro Stunde. Als Spot-Instanz kostet diese Lösung nur 0,84 Dollar, also 30,4 % des On-Demand-Preises. Spot-Instanzen kommen aus einem Pool an Instanzen, den AWS vorhalten muss, um entsprechend dynamisch reagieren zu können. Dies hat zur Folge, dass Kunden zwar deutlich günstiger an die Instanzen kommen, aber diese mit einer Vorlaufzeit von wenigen Minuten nicht mehr zur Verfügung stehen können.

Doch nicht bei allen Knotentypen machen Spot-Instanzen Sinn. Schauen wir uns die verschiedenen Knotentypen der EMR Cluster dafür einmal genauer an:

**Masterknoten:** Ein oder mehrere Masterknoten steuern die Cluster. Spot-Instanzen wären hier unpassend. Denn fallen alle Masterknoten weg, ist der Cluster nicht mehr funktionsfähig.

**Core-Knoten:** Eine weitere Kategorie sind Core-Knoten. Diese stellen im Hadoop Distributed Filesystem (HDFS) den Speicher zur Verfügung. Da im HDFS standardmäßig dreifach repliziert wird, würde bei einem Wegfall eines Core-Knotens das HDFS die Daten, die auf ihm gespeichert sind, auf andere Core-Knoten verteilen, um die dreifache Replizierung wiederherzustellen. Je nach Größe des Clusters kann dies unterschiedliche Konsequenzen haben, von nahezu keiner Auswirkungen bis hin zu einem Cluster-Ausfall.

**Task-Knoten:** Diese sind am ehesten für eine Spot-Instanz geeignet. Hier werden nur Rechenleistung und Arbeitsspeicher zur Verfügung gestellt. Grundsätzlich funktionieren die Cluster so, dass nach gewisser Zeit ohne Ergebnis die Aufgaben erneut an weitere Task-Knoten gesendet werden. Wenn hier ein oder mehrere Knoten ausfallen, hat dies Auswirkungen auf die Laufzeit, nicht jedoch auf die Funktionsfähigkeit des Clusters. Somit sind Task-Knoten in EMR Clustern eine gute Möglichkeit, Spot-Instanzen einzusetzen und damit Kosten zu sparen.

Eine weitere Dynamisierung stellen Fleets dar. Hier können verschiedene Instanz-Typen genannt werden. Wenn also für den Master kein Instanz-Typ A zur Verfügung steht, kann der Typ B oder auch Typ C verwendet werden. Dies erhöht die Chance, die Verarbeitung mit der gewünschten Rechenleistung zur gewünschten Zeit durchzuführen. Auch Spot-Instanzen können hier beispielsweise adressiert werden: „Wenn innerhalb einer Zeitspanne von x Minuten nicht die gewünschten Spot-Instanzen zur Verfügung stehen, weiche auf On-Demand-Instanzen aus.“

M5 12 xlarge Laufzeit 1 Jahr				
Zahlungsoption	% Vorkasse	Stundensatz in \$	Vergleich in % zu On-Demand	On-Demand Stundensatz in \$
Keine Vorabzahlung	0	1,739	63	2,76
Teilweise Vorauszahlung	50	1,656	60	
Komplette Vorauszahlung	100	1,623	59	

Abbildung 2: Amazon Web Services Kaufoption Reserved Instances xlarge-m5 (Quelle: [4])

#### Sparangebote für Reserved Instances

Eine weitere Option sind Reserved Instances. Eine Instanz wird zumeist für ein Jahr gebucht. Auch eine Buchung für drei Jahre ist möglich. Die größte Ersparnis wird erzielt, wenn die Rechnung zu Beginn der Laufzeit komplett bezahlt wird. **Abbildung 2** zeigt hierzu ein Beispielangebot.

Im Beispielangebot können mit Reserved Instances 41 % der Kosten gegenüber der On-Demand-Option eingespart werden. Die Reserved Instances müssen allerdings bezahlt werden, ob sie laufen oder nicht. Im On-Demand-Modell kann bei verringerten Laufzeiten, z. B. bei einer Laufzeit von zwölf Stunden pro Tag die Hälfte der Kosten eingespart werden. Das ist bei den Reserved Instances nicht möglich und passt damit nicht sonderlich gut zur gewünschten Flexibilität in der Cloud. Obwohl es auch bei den Reserved Instances einen kleinen Trend zur Flexibilisierung gibt. So kann zum Beispiel der Instanz-Typ gewechselt werden. Das Grundproblem wird dabei etwas abgemildert, jedoch nicht komplett gelöst.

An dieser Stelle kommen die Savings Plans ins Spiel. Diese wurden Ende 2019 eingeführt. Bei den Savings Plans können zwei Modelle unterschieden werden:

EC2 Instance Savings Plans (ISP)

und Compute Savings Plans (CSP).

Die ISP beziehen sich auf Instanz-Familien und haben laut AWS ein höheres Sparpotenzial als der flexiblere CSP, in unserem Beispiel die m5-Familie. Es wurde an dieser Stelle der AWS-Kostenkalkulator verwendet, um On-Demand- und ISP-Preis zu vergleichen.

Produktname	% Vorkasse	Kosten/Monat in \$	Vergleich in % zu On-Demand
On-Demand	0	1681,92	100
ISP 1 Jahr	0	1269,47	75,5
ISP 1 Jahr	50	1208,88	71,8
ISP 1 Jahr	100	1184,79	70,4
ISP 3 Jahre	0	870,16	51,7
ISP 3 Jahre	50	805,92	47,9
ISP 3 Jahre	100	757,74	45

Abbildung 3: Preiskalkulation für die ISP-Option im Vergleich

Der On-Demand-Preis wird mit 2,304 Dollar pro Stunde angegeben. Inklusive Mehrwertsteuer entspricht dies einem Preis von ca. 2,74 Dollar. An einigen Stellen wird der On-Demand-Preis mit 2,76 Dollar beziffert, hier gibt es also kleinere Abweichungen. Der ISP-Preis liegt deutlich darunter. Bei der Berechnung des ISP-Preises sind Laufzeit und Zahlungsweise entscheidend. **Abbildung 3** zeigt die Werte für die Laufzeit von einem bis zu drei Jahren im Vergleich. Zur besseren Darstellung wurde dort der Breakeven-

Point in Prozent verwendet. Diese Werte geben an, wie hoch die prozentuale Auslastung der EC2-Instanz sein sollte, damit der ISP-Plan günstiger ist als der On-Demand-Betrieb.

In diesem Beispiel liegt das größte Sparpotenzial bei einem Breakeven Point von 45 %. Hier entstehen nur 45 % Kosten im Vergleich zu einer On-Demand-Nutzung von 100 %. Dafür muss man sich für drei Jahre binden und die Gesamtsumme im Voraus entrichten.

Die Kostenberechnung ist, wie man sieht, nicht ganz trivial und viele Angebote sind in der Praxis nicht ganz so günstig, wie die Anbieter vermuten lassen. Hier gilt es, die verschiedenen Hebel und Einflussfaktoren zu erkennen und dabei die Vorteile flexibler Modelle in der Cloud nicht aus dem Blick zu verlieren. Erfahrungswerte und Einschätzungen von Experten können hier weiterhelfen.

## Fazit

Fassen wir noch einmal zusammen: Die Ablösung bestehender Softwarekomponenten durch Cloud-Native-Technologien stellt vor allem große Unternehmen und Organisationen mit komplexen Strukturen vor besondere Herausforderungen. Eine „Two-Speed-Architektur“ kann dabei helfen, Innovation abseits der Regelorganisation voranzutreiben. Bei der Weiterentwicklung einzelner Cloud-Applikationen kann mithilfe des Site-Reliability-Engineering-Konzepts die Optimierung des Betriebs als fester Bestandteil in die agile Softwareentwicklung aufgenommen werden.

Wir haben die Erfahrung gemacht, dass die Cloud gut zu agilen Projekten passt, sich selbst allerdings auch sehr agil weiterentwickelt. Diese Entwicklung findet auf vielen Ebenen statt. Dementsprechend ist aktuelles Wissen in diesen Bereichen wichtig.

Weitere Möglichkeiten, wie zum Beispiel die Architektur in Richtung Serverless-Technologien auszubauen, eröffnen wieder neue Welten. Als Kunde muss man sich keine Gedanken über Themen wie Servergröße, Betriebssystemversionen u. a. machen. Man verwendet einfach nur den Service und der Anbieter kümmert sich um den Betrieb. Auch kostenmäßig kann das von Vorteil sein, wenn es um tatsächlich verarbeitete oder verwendete Datenmengen geht.

Dabei wird jedoch schnell vergessen, dass ein Team oder auch ein Unternehmen pro Zeiteinheit nur begrenzte Innovationsfähigkeit aufbringen kann und diese auf viele Themen verteilen muss. Diese Innovationsenergie muss bewusst eingesetzt werden. Deshalb ist es erfahrungsgemäß weniger sinnvoll, die Suche nach der besten Architektur und den günstigsten Kosten ständig ohne Punkt und Komma fortzuführen. Das wäre, als würde beim Umzug immer wieder die Route zum neuen Ferienhaus zeitlich und preistechnisch optimiert. Die Zeit und der Preis sind beim Umzug zwar von Bedeutung, am Ende ist die Aufgabe jedoch, die Möbel ohne Beschädigungen in das neue Haus zu befördern.

## Literatur und Links

**[1]** In Anlehnung an Sam Newman; Monolith to Microservices; O`Reilly (2019) S. 62.

**[2]** Ernst Tiemeyer; Handbuch IT-Management; Hanser (2017) S.315 ff.

**[3]** Murphy / Treynor: „In Conversation“, in: Internetseite Site Reliability Engineering, URL: <https://landing.google.com/sre/interview/ben-treynor-sloss/>, Abruf am 26.05.2020.

**[4]** in: Internetseite Amazon EC2 - Preise, URL: <https://aws.amazon.com/de/ec2/pricing/>, Abruf am 26.05.2020.

---



## Bodo Clausen

ist Lead Consultant und arbeitet seit 2007 bei der OPITZ CONSULTING Deutschland GmbH. Seine technische Heimat ist im BI/DWH-, Big-Data- und Cloud-Umfeld. Seit 2014 ist er Teil des Competence Center Big Data und Advanced Analytics und ist in der Rolle des Lead Big Data Integrator zuständig für die Toolauswahl, Planung und Ausführung der entsprechenden Trainings. Seit 2017 arbeitet er im Kundenumfeld in der AWS Cloud mit diversen Technologien. Agile Vorgehen wie Scrum und Kanban gehören seit 2015 zu seinem Arbeitsalltag.

E-Mail: [bodo.clausen@opitz-consulting.com](mailto:bodo.clausen@opitz-consulting.com)

---



## Pascal Stieber

ist Solution Architect und Trainer bei der OPITZ CONSULTING Deutschland GmbH und beschäftigt sich mit technologischen Trendthemen in modernen Softwarearchitekturen. Seine Erfahrung aus über 10 Jahren in der Unterstützung von Design- und Entwicklungsprozessen in Enterprise-Projekten teilt er als Trainer in diversen Schulungen und Workshops mit Kollegen und Kunden.

E-Mail: [pascal.stieber@opitz-consulting.com](mailto:pascal.stieber@opitz-consulting.com)

### **Bildnachweise:**

OPITZ CONSULTING

Online Themenspecial

Impressum

|

Kontakt & Anfrage