

Entwicklung eines Vorgehensmodells für Cloud-Migrationen

Eine Betrachtung aus Sicht eines Independent Software Vendors

Bachelorthesis

Claus Steffen Pegenau (1933040)



TECHNISCHE
UNIVERSITÄT
DARMSTADT

ise.
Information Systems
and Electronic Services

Technische Universität Darmstadt

Fachbereich Rechts- und Wirtschaftswissenschaften

Fachgebiet Wirtschaftsinformatik – Information Systems & Electronic Services

Prof. Dr. Alexander Benlian

Betreuer: Prof. Dr. Alexander Benlian

Bachelorthesis zu dem Thema:

Entwicklung eines Vorgehensmodells für Cloud-Migrationen

Eine Betrachtung aus Sicht eines Independent Software Vendors

Bearbeitet von: Claus Steffen Pegenau

Matr.-Nr.: 1933040

Studiengang: Wirtschaftsinformatik

Eingereicht am: 15.03.2017

Förmliche Erklärung

Hiermit erkläre ich, Claus Steffen Pegenau, geboren am 16.03.1990, an Eides statt, dass ich die vorliegende Bachelorthesis ohne fremde Hilfe und nur unter Verwendung der zulässigen Mittel sowie der angegebenen Literatur angefertigt habe.

Die Arbeit wurde bisher keiner anderen Prüfungsbehörde vorgelegt und auch noch nicht veröffentlicht.

Darmstadt, den 15.03.2017

(Unterschrift)

Inhaltsverzeichnis

Abbildungsverzeichnis	vi
Tabellenverzeichnis	vii
1 Einleitung	1
2 Grundlagen	4
2.1 Migration und Cloud-Migration	4
2.2 Softwarehersteller: Independent Software Vendor (ISV)	8
2.3 Die Migration bestimmende Faktoren	10
2.3.1 Wirtschaftliche Faktoren	10
2.3.2 Technische Faktoren	11
2.4 Beteiligte Unternehmen und das On-Premise-Produkt iFMS	12
2.5 Das Fünf-Phasen-Wasserfallmodell	14
3 Vorgehensmodell für Cloud-Migrationen	16
3.1 Chancen der Cloud	17
3.2 Risiken der Cloud	19
3.3 Phase I: Entwicklung einer Vision	20
3.4 Phase II: Machbarkeitsstudie	22
3.5 Phase III: Agiler, iterativer Entwicklungsprozess	23
4 Forschungsmethoden	26
4.1 Theorie: Systematische Literaturübersicht	26
4.2 Praxis: Workshops und Einzelgespräche	26
5 Das Projekt: iFMS@Salesforce	28
5.1 Phase I: Entwickeln einer Vision	28
5.1.1 Realisierung von Chancen	28
5.1.2 Vermeidung von Risiken	30
5.1.3 Geschäftsmodell & Strategie	31
5.2 Phase II: Machbarkeitsstudie	32
5.3 Phase III: Agiler, iterativer Entwicklungsprozess	33
6 Diskussion	34
7 Zusammenfassung und Ausblick	35
7.1 Offene Forschungsfragen	35
7.2 Schwierigkeiten	35
7.3 Abgabedokument	35

8 Ideen	36
8.1 Einleitung	36
8.2 Gamechanger Cloud	36
8.2.1 Lifecycle	36
8.2.2 Towards Modelling a Cloud Applications Life Cycle	37
8.2.3 Service Migration – Herausforderungen	38
8.3 Charakteristika Enterprise Applications	38
8.4 Assessment und Guidelines	39
8.4.1 Assessment	39
8.4.2 Guidelines	39
8.5 Migrating to – or away from the public cloud	39
8.6 Towards an Understanding of Cloud Computing’s Impact on Organizational IT Strategy	40
8.7 Inhaltsbeschreibungen	41
Literatur	I

Abbildungsverzeichnis

Abbildung 1:	Prognose zum Umsatz mit Cloud Computing weltweit von 2009 bis 2015 mit geschätztem Wert für 2016 entnommen aus Statista (2016)	1
Abbildung 2:	Total Cost of Ownership/Server aus Harms & Yamartino (2010)	5
Abbildung 3:	XaaS im Vergleich: Umfang von Dienstleistung und Eigenverantwortung. Aus Harms & Yamartino (2010)	6
Abbildung 4:	Das Beziehungsdreieck aus ISV, Plattformbetreiber und Kunde. Angelehnt an die Darstellung für SOAs aus Schäfer & Lichter (2016).	8
Abbildung 5:	Der Einfluss des Preises auf das Geschäft. Aus Chappell (2012)	9
Abbildung 6:	Das Reply Unternehmensnetzwerk. Eigene Grafik.	12
Abbildung 7:	Screenshot iFMS aus der Benutzerdokumentation (Reply (2011))	13
Abbildung 8:	Das Fünf-Phasen-Wasserfallmodell aus Rai; Sahoo; Mehfuz (2012)	14
Abbildung 9:	Darstellung des Vorgehensmodells. Der iterative Software Engineering Kreislauf wurde Schäfer & Lichter (2016) entnommen.	16
Abbildung 10:	Funktionsumfang vor und nach der Migration im Wasserfallmodell. Selbsterstellte Grafik.	22
Abbildung 11:	Funktionsumfang vor und nach der Migration nach dem in dieser Arbeit vorgestellten Vorgehensweise. Selbsterstellte Grafik.	23
Abbildung 12:	Schematische Darstellung der Zuordnung von Epics	32
Abbildung 13:	Kosten-Nutzen-Analyse der Anforderungen	32

Tabellenverzeichnis

Tabelle 2: Unterschiede im Requirementsengineering. Entnommen aus Schäfer & Lichter (2016)	24
Tabelle 3: Forschungsfragen und zugehörige Rechercheausdrücke. Angelehnt an Rai; Sahoo; Mehfuz (2015)	26
Tabelle 4: Literaturdatenbanken und für welche Fragen sie herangezogen wurden. Quellen: Rai; Sahoo; Mehfuz (2015) und Benlian (2016)	27

1 Einleitung

Die Cloud ist im Mainstream angekommen. (Chase et al. 2014) Wie in Abbildung 1 dargestellt, wuchs der weltweite Umsatz mit Cloud-Computing von 58,6 Milliarden US-Dollar im Jahr 2009 auf geschätzte 203,9 Milliarden US-Dollar im Jahr 2016, was einem durchschnittlichen jährlichen Wachstum von 16,87%¹ entspricht. (Statista 2016)

Auch deutsche Unternehmen drängen zunehmend in die Cloud. Das Marktforschungsunternehmen PAC schätzt, dass der deutsche Cloud-Markt im Jahr 2016 eine Größe von 12,5 Milliarden Euro hatte und mit durchschnittlich jährlich 20,9%² auf 31,4 Milliarden im Jahr 2020 anwächst. Im deutlichen Gegensatz dazu prognostiziert PAC für den Markt der traditionellen IT-Dienstleistungen ein negatives Wachstum von -1,7%³. (Dufft 2016)

Prognose zum Umsatz mit Cloud Computing weltweit von 2009 bis 2016

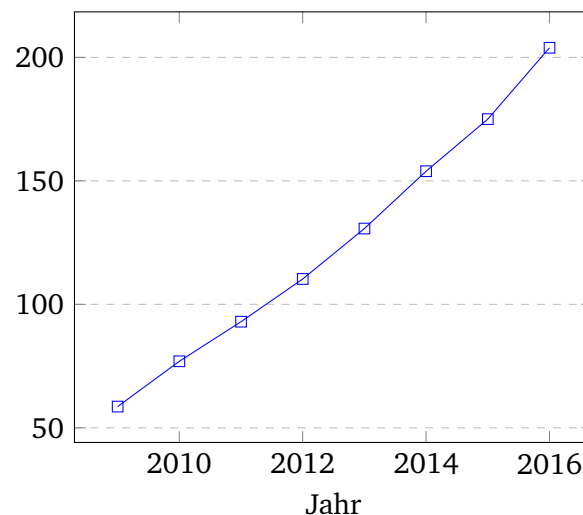


Abbildung 1: Prognose zum Umsatz mit Cloud Computing weltweit von 2009 bis 2015 mit geschätztem Wert für 2016 entnommen aus Statista (2016)

Gerade Softwarehersteller aus diesem traditionellen Bereich der IT-Dienstleistungen sehen sich unter Druck gesetzt, ihre Unternehmungen von diesem schrumpfenden Markt weg, in den stark wachsenden Cloud-Markt zu verlagern. Dabei ist es intuitiv vernünftig, vorhandene Kernkompetenzen durch Migrationen bestehender Produkte zu nutzen, um Wettbewerbsvorteile auf dem neuen Markt zu erlangen.

Doch nicht nur die Umsatzentwicklung des Marktes setzt die Softwarehersteller unter Druck: Ihre Kunden haben sich an Anwendungen in der Cloud gewöhnt und erwarten sich - von ihr - eine günstigere, schnellere, einfachere, flexiblere und effizientere IT im Allgemeinen. (Harms & Yamartino 2010)

¹ CAGR(2009,2016) = 16,87%

² CAGR(2016,2020) = 20,9%

³ CAGR(2015,2019) = -1,7%

Diese bei den Nutzern geweckten Erwartungen sorgen bei den Softwareherstellern für zusätzlichen Migrationsdruck, sie setzen aber auch einen neuen, höheren Maßstab für Software im Allgemeinen.

Entschließt sich ein Softwarehersteller dazu, seine Produkte als Dienstleistungen in der Cloud anzubieten, ändert sich nicht nur die technologische, sondern auch die wirtschaftliche Umgebung erheblich, da ein neuer Markt erschlossen wird und bedacht werden muss, wie und wo die Cloud-Lösung auf dem Markt zu positionieren ist. Die Position im Wettbewerb hat nicht nur für das Vertriebsmodell Auswirkungen - man denke an Fragen der Lizenzierung und Preismodelle - sondern auch den Leistungsumfang. Denn je standardisierter eine Software ist, je geringer die nötige Anpassbarkeit, desto wahrscheinlicher lassen sich bei einem Betrieb in der Cloud die genannten Vorteile realisieren. (Buxmann; Hess; Lehmann 2008) Dies hängt damit zusammen, dass sich bei standardisierter Software Stellschrauben vor dem Nutzer verbergen lassen. Im Optimalfall spielen Netzwerktopologie, Betriebssystem, Laufzeitumgebung und Datenbanken keine Rolle; der Anwender sieht und arbeitet lediglich mit der Software. In diesem Fall spricht man von „Software as a Service“ (SaaS). (Harms & Yamartino 2010, S. 11)

In der Literatur finden sich zahlreiche Vorgehensmodelle, die versuchen, die Migration eines Altsystems beziehungsweise einer Altsoftware in die Cloud im weitesten Sinne zu strukturieren. In einigen, wie in dem von Ahmad & Babar (2014) entwickelten Modell wird ein technischer Schwerpunkt gesetzt; wirtschaftliche Aspekte bleiben weitgehend unberücksichtigt. Andere fokussieren sich lediglich auf den Kosten-Faktor (Rai; Sahoo; Mehruz (2012)). Umfassendere Modelle (Council (2016)) betonen die markt-strategischen Folgen einer Cloud-Migration für Unternehmen.

Zielgruppe der Modelle sind in der Regel Unternehmen, die Teile ihrer IT-Landschaft in die Cloud migrieren wollen. Dies kann zum Beispiel ein Unternehmen sein, das seine Lagerhaltung in die Cloud verlagern will. Eine in dieser Arbeit durchgeführte Literaturrecherche ergab, dass Softwarehersteller als Zielgruppe bisher vernachlässigt wurden. Der Unterschied zwischen den beiden Gruppen ist gravierend: Während im oben genannten Beispiel die Lagerhaltung in Form einer Software lediglich das Kerngeschäft unterstützt, besteht für den Softwarehersteller das Kerngeschäft in der zu migrierenden Software. Für den Erfolg im Wettbewerb ist eine strategische Neuausrichtung unabdingbar, unterscheidet sich in ihrer Tragweite und ihrem Umfang jedoch erheblich von den strategischen Überlegungen, die ein Unternehmen einer anderen Branche bei der Cloud Migration treffen muss.

Um diese Forschungslücke zu schließen, soll ein Modell entwickelt werden, das Softwarehersteller bei der Migration in die Cloud unterstützen soll. Grundlage ist eine Literaturrecherche, die die folgenden Fragen klären soll:

1. Welche Merkmale (Chancen und Risiken) der Cloud sind bei der Migration zu berücksichtigen?

-
2. Wie beeinflussen die gefundenen Merkmale die strategische Ausrichtung und das Geschäftsmodell eines Softwareherstellers?
 3. Wie beeinflussen die gefundenen Merkmale den Migrationsprozess eines Softwareherstellers?

Bevor diese Fragen in Kapitel 3 beantwortet werden und in der Entwicklung eines Vorgehensmodells resultieren, wird in Kapitel 2 auf grundlegende Fragen eingegangen: Welche Vorteile verspricht die Cloud und wie lassen sie sich erzielen? Auf welche Weise unterscheidet sich eine Cloud-Migration von herkömmlichen Migrationen in der IT? Was wird in dieser Arbeit unter einem Softwarehersteller verstanden und welche Annahmen werden getroffen? Wie wirken sich dessen Charakteristika auf die Migration aus? Um Erfahrungen aus der Praxis in das Modell einzubeziehen, wurde das entwickelte Modell in Zusammenarbeit mit einem Beratungsunternehmen auf ein reales Projekt angewendet; Unternehmen und Projekt werden ebenfalls in diesem Kapitel vorgestellt. Das Grundlagenkapitel schließt mit der Vorstellung des Fünf-Phasen-Modells, das als Basis und Orientierung für das in dieser Arbeit entwickelte Modell dient.

Kapitel 4.1 beschreibt die Literaturrecherche mit der die Forschungsfragen beantwortet wurden. Um praktische Erfahrungen einfließen zu lassen wurden Gespräche mit Mitarbeitern geführt, deren Ablauf ebenfalls in diesem Kapitel geschildert werden.

In Kapitel 5 wird das Vorgehensmodell beispielhaft auf das zu migrierende Projekt aus der Praxis angewendet; es werden Möglichkeiten dargestellt, um Chancen der Cloud zu realisieren und Risiken zu meiden – auch in Bezug auf das Geschäftsmodell.

In Kapitel 6 werden die Unterschiede zwischen vorgeschlagenem und tatsächlichem Vorgehen herausgearbeitet und diskutiert.

Die Ergebnisse dieser Diskussion werden anschließend in Kapitel 7 zusammengefasst, offene Fragestellungen, Herausforderungen und Forschungsfragen formuliert.

2 Grundlagen

In dieser Arbeit soll für „Independent Software Vendors“, also unabhängige Softwarehersteller ein Vorgehensmodell für Cloud-Migrationen zu Salesforce entwickelt werden. Aus diesem Ziel ergeben sich grundlegende Fragen, die in diesem Kapitel beantwortet werden sollen:

- Welche Vorteile bietet die Cloud und wie lassen sie sich erzielen? Inwiefern unterscheiden sich Cloud-Migrationen von herkömmlichen Migrationen in der IT und wie lassen sie sich zum Outsourcing abgrenzen?
- Was wird in dieser Arbeit unter „Independent Software Vendors“ verstanden? Welche Besonderheiten weisen sie und ihre Projekte auf und welchen Einfluss haben diese Besonderheiten auf Cloud-Migrationen?
- Warum ist Salesforce als Zielplattform für Migrationen geeignet?
- Welche Faktoren beeinflussen die Cloud-Migration und ihren Erfolg?

Das Kapitel wird von der Vorstellung des Fünf-Phasen-Modells abgeschlossen, auf dem das entwickelte Vorgehensmodell basiert. Auch wird die Wahl für dieses Modell als Arbeitsgrundlage begründet.

2.1 Migration und Cloud-Migration

Die Migration von Software, wird in der Norm ISO/IEC 14764 „Software Engineering – Software Life Cycle Processes – Maintenance“ als spezielle Form der Wartung, nämlich als anpassende Wartung („Adaptive Maintenance“) und Weiterentwicklung definiert, bei der eine Software nach Auslieferung geändert wird, um das Produkt in geänderten oder sich ändernden Umgebungen nutzbar zu halten. (Williams & Carver 2010)

Beispiel für eine solche Migration ist der Umzug einer Software auf neue Hardware mit aktualisiertem Betriebssystem. Grund für den Umzug könnte ein technischer sein, wie etwa dass keine Ersatzteile mehr für die vorhandene Hardware angeboten werden, dass der Support des Herstellers für das laufende Betriebssystem ausläuft. Aber auch wirtschaftliche Gründe, wie etwa eine geforderte, höhere Leistungsfähigkeit oder ein geringerer Energieverbrauch des neuen Systems sind denkbar.

Um die Software lauffähig zu halten, muss sie gegebenenfalls an das aktualisierte Betriebssystem oder die Laufzeitumgebung angepasst werden. Je standardisierter und verbreiteter die Anforderung der zu migrierenden Software, desto geringer werden die nötigen Anpassungen an der Software ausfallen, womit die Migration einfacher und kostengünstiger wird. Eventuell wird lediglich eine virtuelle Maschine auf einen neuen Server gestartet.

Eine solche virtuelle Maschine ließe sich auch in der Cloud starten. Auf diese Weise ließen sich jedoch nicht die folgenden Vorteile realisieren, die Rai; Sahoo; Mehruz (2015) als Hauptgründe für eine Migration in die Cloud identifiziert haben:

Kosteneinsparungen Cloud-Anbieter, die große Rechenzentren mit vielen Servern betreiben können die Total Cost of Ownership pro Server um bis zu 80% reduzieren (Vgl. Abbildung 2). Hauptsächlich geschieht dies über die Faktoren Energie-, Lohn- und Hardwarekosten. En-

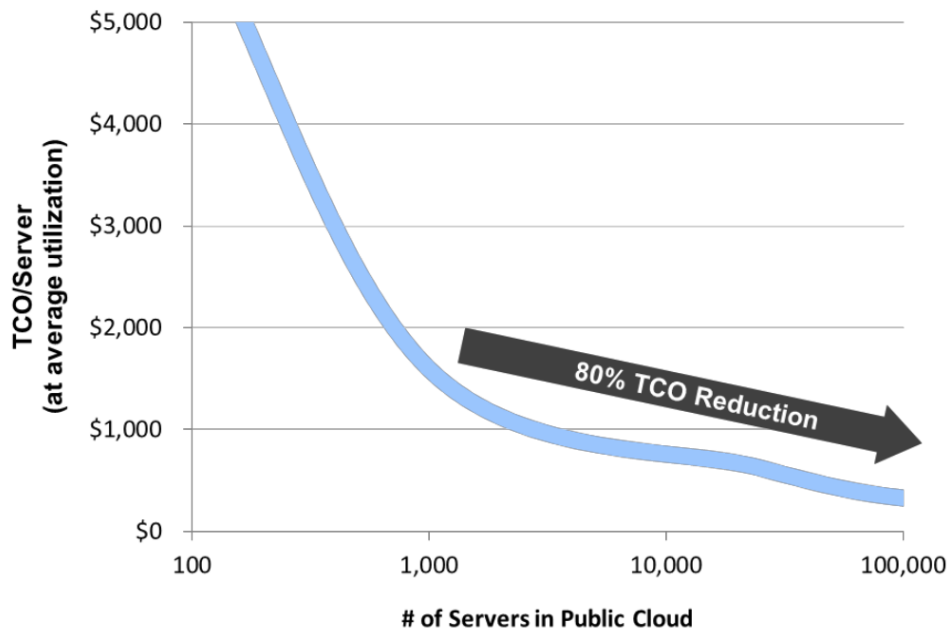


Abbildung 2: Total Cost of Ownership/Server aus Harms & Yamartino (2010)

ergieekosten lassen sich durch die Wahl eines Standortes mit günstigem Strom niedrigen Umgebungstemperaturen senken. Durch Automatisierungstechniken lässt sich die Zahl der Server, die ein Administrator betreibt verzehnfachen, wodurch die Lohnkosten sinken. Hardwarekosten fallen durch bessere Verhandlungspositionen großer Cloud-Anbieter niedriger aus. (Höllwarth 2012)

Die niedrigeren Kosten werden über den Markt teilweise an die Kunden weiter gegeben, die die kapitalintensive Beschaffung der Hard- und Software sowie die Vorhaltung eigener Personalressourcen zur Entwicklung und Wartung reduzieren können. (Repschläger; Pannicke; Zarnekow 2010)

Effiziente Ressourcennutzung In der konventionellen IT-Welt werden geschätzte 80% der Ressourcen dazu verwendet, bestehende Infrastruktur und Dienstleistungen zu betreiben, wodurch wenig Kapazität für wertstiftende Tätigkeiten, wie die Entwicklung neuer Geschäftsfelder oder die Umsetzung von Kundenwünschen bleibt. Cloud-Nutzung kann dieses Verhältnis der Innovation verschieben. (Harms & Yamartino 2010)

Automatisierte, schnelle und unbegrenzte Skalierbarkeit der Ressourcen Die Zuteilung zusätzlicher, auf den Kunden unbegrenzt wirkender Rechenleistung erfolgt in der Cloud entweder über ein einfaches Webformular oder ausgerichtet am momentanen Bedarf voll automatisch. Da der Kunde nur das bezahlen muss, was er benötigt, ist die Cloud gerade für Start-ups, kleine und mittlere Unternehmen, sowie Unternehmen interessant, die entweder einen stark



Abbildung 3: XaaS im Vergleich: Umfang von Dienstleistung und Eigenverantwortung. Aus Harms & Yamartino (2010)

schwankenden oder nicht abschätzbaren Bedarf haben; die Gefahr zu viel Rechenleistung vorzuhalten, was zu unnötig hohen Kosten führt oder zu wenig Rechenleistung, was Kunden abschrecken könnte, sinkt. (Harms & Yamartino 2010; Johnson & Qu 2012; Azeemi; Lewis; Tryfonas 2013)

Das schnelle Deployment in der Cloud ermöglicht es, neue Ideen sehr schnell zu implementieren, zu testen (falls gewünscht sogar im Vergleich zu anderen Lösungen) und auszuwerten. Dadurch kann die Innovationsgeschwindigkeit, Flexibilität und die „Time to Market“ beachtlich gesteigert werden. (Sill 2014; Martens & Teuteberg 2011; Höllwarth 2012)

Geringerer Wartungsaufwand Die Cloud treibt die Standardisierung von Schnittstellen voran, sodass bestehende Funktionalitäten leichter wiederverwendet werden können. Dadurch wird die Weiterentwicklung und Wartung von Software reduziert. (Harms & Yamartino 2010)

Der Grad, in dem diese Vorteile realisiert werden, davon ab, in welchem Umfang Leistungen in der Cloud in Anspruch genommen werden, was wiederum Einfluss auf den Migrationsprozess hat. (Pahl; Xiong; Walshe 2013, S. 213) Zur Kategorisierung des Leistungsumfangs dienen „as a Service“-Modelle. Es gibt recht viele dieser Modelle - Schaffer (2009) zählt 35 Varianten - die mehr oder weniger verbreitet sind und mit „Everything as a Service“, abgekürzt als EaaS oder XaaS zusammengefasst werden. Dabei werden nicht nur Technologien als Dienstleistung angeboten, sondern auch Ziele; so gibt es beispielsweise auch „Hardware-as-a-Service (HaaS)“ und „Business-Process-as-a-Service (BPaaS)“. (Bouvry 2014; Benlian; Hess; Buxmann 2010)

Das National Institute of Standards and Technology sieht drei Service Modelle anerkannte vor, die sich im Anteil der selbst zu verwaltenden, technologischen Anteile (Integrationstiefe) unterscheiden und in dieser Hinsicht in Abbildung 3 mit der herkömmlichen IT verglichen werden. (Mell & Grance 2011; Chase et al. 2014)

Infrastructure as a Service (IaaS) Dem Kunden werden Netzwerk, Speicher und Rechenleistung zur Verfügung gestellt. Über die darauf laufende Software, teilweise sogar über das Betriebssystem kann selbst verfügt werden.

Kunden versprechen sich von IaaS vor allem Flexibilität in der Bereitstellung und dem Betrieb von Servern, Datenspeichern und Netzwerkressourcen. (Pahl; Xiong; Walshe 2013, S. 213)

Platform as a Service (PaaS) Der Kunde kann bereitgestellte oder selbst entwickelte Software in der Cloud laufen lassen und eventuell Einfluss auf die Anwendungsumgebung nehmen.

Software as a Service (SaaS) Der Kunde kann eine vom Anbieter bereitgestellte Software nutzen und in begrenztem Maße konfigurieren.

Das für die Migration genannte Beispiel, eine virtuelle Maschine zu verschieben, lässt sich mit Abbildung 3 für den Cloud-Fall dem Modell IaaS zuordnen. Der Cloud-Kunde muss sich in diesem Fall neben der eigentlichen Anwendung nach wie vor um das virtualisierte Betriebssystem, Middleware, Laufzeitumgebung und Datenhaltung kümmern. Um die Vorteile der Cloud besser auszuschöpfen, müsste die Anwendung entweder so angepasst werden, dass sie auf einer zur Verfügung gestellten, fremdverwalteten Laufzeitumgebung läuft (PaaS). Oder es wird eine bestehende Anwendung in der Cloud genutzt und gegebenenfalls angepasst (SaaS). Beide Optimierungen machen das Treffen konsequenzenreicher, kritischer Entscheidungen nötig, die nicht nur technische, sondern auch betriebswirtschaftliche und strategische Aspekte betreffen. (Pahl & Xiong 2013) Aus diesem Grund ist eine Cloud-Migration⁴ (Pahl; Xiong; Walshe 2013) in ihrem Umfang viel mehr als eine Weiterentwicklung herkömmlicher Migrationen und erfordert eine dementsprechende, viel umfangreichere Planung. (Ahmad & Babar 2014; Alkhalil; Sahandi; John 2016)

Für die zu migrierende Software gibt es zahlreiche Namen, zum Beispiel „Software as a Product“ (SaaP) (Schäfer & Lichter 2016), „Software as a Good“ (Olsen) oder „Legacy System“ (Ahmad & Babar 2014). In dieser Arbeit wird der Begriff „On-Premise-Software“ verwendet.

Der Ablauf von Cloud-Migrationen lässt sich in die folgenden Kategorien unterteilen: Rai; Sahoo; Mehruz (2015)

Ersetzen Einzelne Komponenten einer bestehenden Anwendung werden durch Dienste in der Cloud ersetzt. Andere Komponenten, die mit diesen ersetzten Komponenten interagieren werden umkonfiguriert. Buzzwörter zu dieser Kategorie sind „Service Oriented Architecture“

⁴ Der Vollständigkeit halber eine formale Definition der Cloud-Migration: „Prozess der anteiligen oder vollständigen Bereitstellung von digitalen Gütern, Dienstleistungen, IT-Ressourcen oder Anwendungen in der Cloud“

(SOA) und „Microservices“: Anstatt Anwendungen als Monolithen gewaltigen Umfangs zu entwerfen, werden sie aus wiederverwertbaren Diensten konstruiert, die über Schnittstellen verbunden werden.

Partielle Migration Anstatt Komponenten durch Cloud-Dienste zu ersetzen, werden sie in die Cloud migriert.

Migration der ganzen Anwendung Diesem Typus entspricht das Beispiel von oben: Eine Anwendung wird in ihrer virtuellen Maschine in die Cloud umgezogen.

Cloudify Die Anwendung wird in Gänze an die Cloud und ihre Charakteristika angepasst und mit Cloud Diensten neu entwickelt. Dieser Maxime folgt das in dieser Arbeit vorgestellte Modell.

2.2 Softwarehersteller: Independent Software Vendor (ISV)

Unter dem Begriff „Independent Software Vendor“ (ISV) werden Softwarehersteller zusammengefasst, die Software für von ihnen unabhängige Plattformen, zum Beispiel Microsoft Windows, Google oder Salesforce entwickeln. Sofern der ISV in der Cloud agiert, wird er in der Literatur als „Cloud Carrier“ (Bohn et al.), „Cloud Service Partner“, „Cloud Service Developer“ (Wan & Wang 2014) oder „Cloud Service Vendor“ (Chappell 2012) bezeichnet. Diese Unterscheidung wird getroffen, weil der Wechsel in die Cloud nicht nur mit technischen sondern auch wirtschaftlichen Neuerungen einhergeht. Da sich noch kein Begriff durchsetzen konnte, wird der Softwarehersteller in dieser Arbeit auch wenn er im Cloud-Bereich agiert, als ISV bezeichnet.

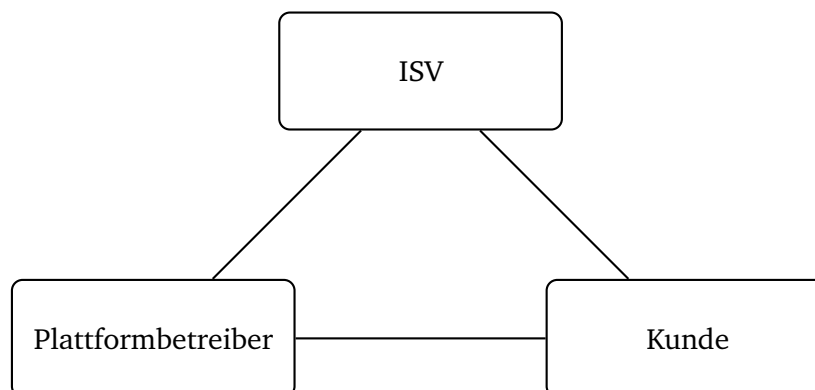


Abbildung 4: Das Beziehungsdreieck aus ISV, Plattformbetreiber und Kunde. Angelehnt an die Darstellung für SOAs aus Schäfer & Lichter (2016).

Wie in Abbildung 4 dargestellt ist der ISV Teil einer Dreiecksbeziehung. Er entwickelt Software für die Plattform des Betreibers, veröffentlicht sie gegebenenfalls auf einem Marktplatz der Plattform. Beispielhaft dafür seien hier Apples App Store oder AppExchange von Salesforce genannt. Die Wahl der passenden Plattform ist essentiell für den Unternehmenserfolg, da Funktionsausfälle und Performanzprobleme können auf den ISV zurückfallen können. Um den passenden Betreiber zu wählen, schlägt Chappell (2012) vier zu berücksichtigende Aspekte vor: Anbietergröße, geo-

grafische Lage, Zuverlässigkeit und gesetzliche Vorgaben.

Der Kontakt zum Kunden hängt vom Markt und insbesondere vom Verkaufspreis ab (Vgl. Abbildung 5). Im Beispiel des App Stores wird es bei kostenlosen oder sehr günstigen Apps keinen oder kaum Kundenkontakt in Form von Anwerbungen, Verkaufsgesprächen, Support oder Training geben. Pauschal gesagt, steigen mit dem Preis die Erwartungen des Kunden in Bezug auf Ansprechbarkeit und Anpassbarkeit der Software. (Chappell 2012)

Average Selling Price	Customer Acquisition Cost	Sales Force	Customer Relationship	Allowable User Training	Example Customer
\$\$\$\$\$	High	External	Personal	Significant	Enterprise
\$\$\$	Moderate	Inside	Telephone	Moderate	Department, SMB
\$	Low	None	Web	Minimal	SMB, Individual

Abbildung 5: Der Einfluss des Preises auf das Geschäft. Aus Chappell (2012)

Die Erschließung des Cloud-Marktes wirkt sich auf das Geschäftsmodell des ISV aus (Chappell 2012):

Markt Das Angebot einer Cloud-Lösung könnte darauf abzielen, bestehende Kunden und Märkte besser zu erreichen; möglicherweise unter Kannibalisierung des bestehenden On-Premise-Produktes. Ziel könnte aber auch sein, neue Märkte zu erschließen. Beide Ziele sind nicht exklusiv. Es ist jedoch wichtig, sich der Möglichkeit der Schwerpunktbildung in dieser Frage bewusst zu sein.

Preisgestaltung Die meisten Preismodelle haben eines der drei folgenden Modelle als Grundlage:

- Abonnement – Nutzer pro Monat oder Gerät pro Jahr. Dies ist das verbreitetste Modell. Durch Vergünstigungen bei Vorauszahlung für einen längeren Zeitraum, können früh höhere Umsätze generiert werden.
- Pro Einheit – Zahlung für jede Transaktion, jedes gespeichertes/übertragenes Gigabyte oder für eine andere messbare Einheit, die einen Nutzungsumfang beschreibt
- Free(mium) – Das Basisprodukt ist kostenlos nutzbar oder wird durch Werbeeinblendungen finanziert; Premiumfunktionen müssen bezahlt werden

Die Wahl der Preisgestaltung hat bedeutenden Einfluss auf den Cashflow des Unternehmens. Gerade die Anfangsinvestition ist bei Cloud-Anwendungen häufig größer als bei On-Premise-Anwendungen, gerade wenn das Unternehmen noch keine Erfahrungen mit der Umsetzung hat. Diese Anfangsinvestition müssen ISV häufig leisten, bevor Kunden akquiriert wurden. Der Einfluss, den die Höhe des Preises auf das Geschäft hat, ist in Abbildung 5 dargestellt. Bei der Migration in die Cloud sollte darauf geachtet werden, dass Preis und die Ausprägungen der anderen Aspekte in der gleichen Zeile liegen.

Verkauf Bei On-Premise-Anwendungen kann der Kunde die Anwendung häufig erst nach dem Kauf in seiner Unternehmensumgebung testen und damit den wahren Produktwert feststellen, weil die Inbetriebnahme für Tests vor dem Kauf zu aufwändig wäre. Bei Cloud-Anwendungen ist ein ausführliches Testen schon vor Abschluss eines großen Lizenzvertrages möglich. Für den ISV kann dies ein Vorteil sein, da die Einstiegshürden für Kunden vermindert sind: Mögliche Fehlinvestitionen lassen sich reduzieren oder vermeiden, weshalb Nutzungsentscheidungen vom Kunden von einer kleineren Gruppe von Managern getroffen werden können. Ist das Produkt im Unternehmen, lassen sich benötigte oder gewünschte Features leichter ermitteln. Der Kunde erhält ein Produkt nach seinen Wünschen, der ISV Projektverträge um die Anwendung anzupassen oder weiterzuentwickeln. Dieser Ansatz wird als „land and expand“ bezeichnet.

Die Erschließung der Cloud kann mit drei Organisationsstrukturen erfolgen (Chappell 2012). Die erste Option besteht darin, die Struktur nicht zu verändern, womit der organisatorische Aufwand zunächst minimiert wird. Soll das On-Premise-Geschäft jedoch weiter betrieben werden, ist es schwierig, beides gut zu machen. Bei Option zwei wird eine Gruppe im Unternehmen gebildet, die sich ausschließlich mit dem Thema Cloud befasst. Die letzte Option besteht darin, diese Gruppe in eine wirtschaftlich selbstständige Einheit auszulagern. Mit dieser letzten Option ist der größte Aufwand verbunden, da mit dem neuen Unternehmen erheblicher Overhead in der Verwaltung (zum Beispiel Personalmanagement und Buchhaltung) einhergeht.

2.3 Die Migration bestimmende Faktoren

Rai; Sahoo; Mehruz (2012) identifizieren wirtschaftliche und technische Faktoren, die sowohl die Migrationseignung einer Anwendung als auch die Migration selbst beeinflussen.

Martens & Teuteberg (2011) sehen viele Parallelen zwischen dem Outsourcen von IT zu externen Dienstleistern und dem Betrieb in der Cloud, weshalb sich Erfahrungen bei Outsourcing-Projekten bei Cloud-Migrationen anwenden lassen. Aus diesem Grund sind die folgenden Faktoren auch aus der Literatur über IT-Outsourcing zusammengetragen.

2.3.1 Wirtschaftliche Faktoren

Bereits getätigte IT-Investitionen: In der Regel wachsen die bereits getätigten IT-Ausgaben mit dem Unternehmen und mit ihnen die Komplexität der Migration. Deshalb ist es in kleinen Unternehmen eher möglich, direkt zu migrieren, während bei größeren Unternehmen

der Übergang in die Cloud wesentlich mehr Planung und gegebenenfalls einen parallelen Betrieb erforderlich macht.

Kosten: In der herkömmlichen IT bestehen Kosten aus der „kapitalintensiven Beschaffung der Hard- und Software sowie der Vorhaltung eigener Personalressourcen“ (Repschläger; Pannicke; Zarnekow 2010). Diese Kosten sind zwar hoch, aber aufgrund der langjährigen Erfahrung auch vorhersagbar und in Budgets eingeplant. Die Migration in die Cloud dagegen bedeutet den Umstieg zu einem „pay per use“-Modell (Khan 2014), von einem von Fixkosten dominierten Modell zu einem, das von variablen Kosten bestimmt ist. Um zu verhindern, dass Kosteneinsparungen aufgezehrt werden, ist es nötig den Umfang der Anwendungsnutzung und die Migrationskosten abzuschätzen.

Datensicherheit: Bevor eine Anwendung in die Cloud migriert wird, sollte bedacht werden, wie kritisch die zugehörigen Daten für den Unternehmenserfolg sind.

Rechtliche Restriktionen: Vor der Migration sollte geprüft werden, ob rechtliche Bestimmungen auch bei einem Betrieb in der Cloud eingehalten werden können.

Zuteilung von Rechenleistungen: Anwendungen, die kurzzeitig große Rechenleistungen benötigen und gut skalierbar sein sollen, lassen sich in der Cloud wahrscheinlich kostengünstiger betreiben als auf eigenen Servern, die ganzjährig reserviert sind und die meiste Zeit im Leerlauf verbringen.

2.3.2 Technische Faktoren

Bestehende Infrastruktur: Bereits die Migration einer einzigen Anwendung kann Änderungen in der internen IT-Infrastruktur erforderlich machen. Zum Beispiel wenn Daten zwischen verschiedenen Diensten ausgetauscht werden soll.

Sicherheitsarchitektur: Um die Daten im Cloud-Umfeld zu schützen, muss das bestehende Sicherheitskonzept an die Gegebenheiten der Cloud angepasst werden.

Komplexität: Während einfache, standardisierte Anwendungen womöglich bereits in der Cloud angeboten werden, steigt mit der Komplexität auch der Planungs-, Implementierungs und Testbedarf bei der Migration.

Netzwerk und Support: Je mehr Daten in der Cloud liegen, desto höher ist die Abhängigkeit von einer funktionierenden Internetverbindung. Hier können zusätzliche Kosten für redundante Verbindungen, Verbindungen mit höheren Kapazitäten oder Verträge mit garantierten Reaktionszeiten im Störfall anfallen.

IT-Fähigkeiten: Auch wenn im Cloudbetrieb auf existierende Technologien und idealerweise existierende Software zurückgegriffen wird, erfordert die Migration dem IT-Team Fähigkeiten und Kenntnisse in den Bereichen Architekturen, Implementierung, Entwicklung und Betrieb ab. Hinzu kommt, dass der Umfang, in dem Kontrolle über die Systeme im Cloudbetrieb ab-

gegeben wird, von den verantwortlichen IT-Mitarbeitern eine „kulturelle“ Herausforderung darstellen kann.

Service Level Agreements (SLAs): Geprüft werden sollte auch, ob Cloud-Anbieter SLAs bieten können, die zum unternehmerischen Bedarf hinsichtlich Verfügbarkeit, Vertraulichkeit und Integrität passen. Auch sollte geregelt sein, welche Verantwortlichkeiten der Anbieter trägt und welche Vertragsstrafen bei Nichteinhaltung drohen.

2.4 Beteiligte Unternehmen und das On-Premise-Produkt iFMS

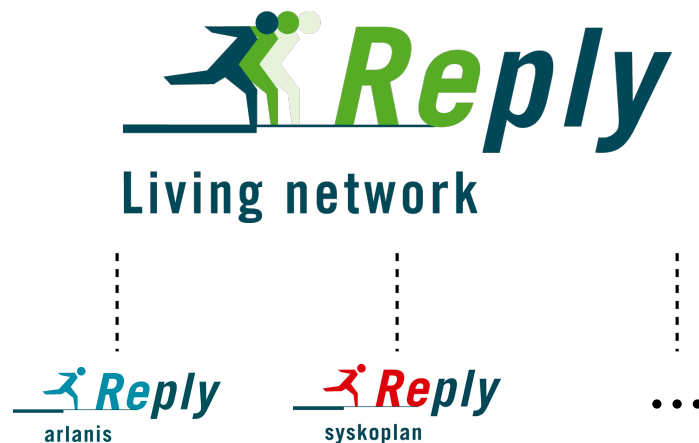


Abbildung 6: Das Reply Unternehmensnetzwerk. Eigene Grafik.

Reply ist ein an der italienischen Börse gehandeltes IT-Beratungsunternehmen und betrachtet sich als „Living network“ aus hochspezialisierten Tochterunternehmen. Seit der Gründung 1996 konnte Reply seinen Umsatz auf über 705 Millionen Euro bei 5.245 Angestellten im Jahr 2015 steigern. Das Netzwerk wuchs und wächst rasch: 2016 wurden bis November drei neue Firmen akquiriert. Zwei Tochtergesellschaften, die schon seit mehreren Jahren Teil von Reply sind, möchte ich genauer vorstellen, da ihre Unternehmensprofile das Migrationsprojekt in besonderem Maße beeinflussen.

Die vormalige syskoplan AG, seit dem Erwerb 2010 (Reply 2016, S. 12) Syskoplan Reply, ist ein Spezialist für SAP-Applikationen und -Plattformen (Reply 2016, S. 10) und entwickelt seit 1999 das integrierte Facility Management System (iFMS). iFMS verbindet die in SAP hinterlegten Daten mit Gebäudeplänen und versucht Prozesse rund um die Verwaltung von Immobilien zu unterstützen. Die gewachsene Java-Anwendung mit einer Client-Server-Architektur lässt sich inzwischen nur noch schwer um von Kunden gewünschte Funktionen erweitern. Auch die Bedienung über eine zusätzlich zu installierenden Anwendung wirkt in Zeiten, in denen Nutzer es gewohnt sind, auch umfangreiche Software über den Webbrowser zu bedienen, anachronistisch. Beide Aspekte schränken die zukünftige Wettbewerbsfähigkeit der Software ein.

Die ehemalige Arlanis Software AG wurde 2012 von Reply übernommen und ist Spezialist für Lösungen auf Basis des Cloud Anbieters Salesforce. Mit Salesforce lassen sich Lösungen häufig ganz ohne eigenen Code auf einer SaaS-Basis zusammen klicken. Ist doch Code erforderlich

lässt er sich auf der PaaS-Plattform (Force.com und Heroku) entwickeln, wobei auf per SaaS angelegte Datenmodelle und Daten zugegriffen werden kann, auf die externe Anwendungen per Web Service zugreifen können. Mit Salesforce1 steht eine App für mobile Geräte zur Verfügung; jede Salesforce-Anwendung lässt sich über diese App bedienen.

Über das Unternehmensnetzwerk von Reply sollen die Stärken beider Tochterunternehmen verbunden werden; mit dem Know-How im Bereich Facility Management und SAP soll auf Salesforce eine innovative, wettbewerbsfähige Anwendung geschaffen werden: iFMS@Salesforce.

Die On-Premise-Anwendung iFMS dient dem Facility Management. Mit ihr lassen sich Gebäude, Etagen und Räume verwalten, mieten und vermieten. Dabei werden in SAP hinterlegte Informationen wie Raumgrößen, Nutzungsarten, Adressen und Kontaktinformationen mit CAD Plänen verknüpft. Um iFMS zu nutzen, müssen Unternehmen einen Server bereitstellen und warten, auf dem der Serverteil der Anwendung läuft. Außerdem wird ein Datenbankserver benötigt, auf dem iFMS seine Daten speichern kann. Auf dem Rechner eines jeden Nutzers muss ebenfalls eine Anwendung installiert werden.

Ein Screenshot aus Anwendersicht ist in Abbildung 7 dargestellt. Sie soll nicht im Detail erklärt werden, sondern nur die Art und Komplexität der Anwendung andeuten. Auf der linken Seite

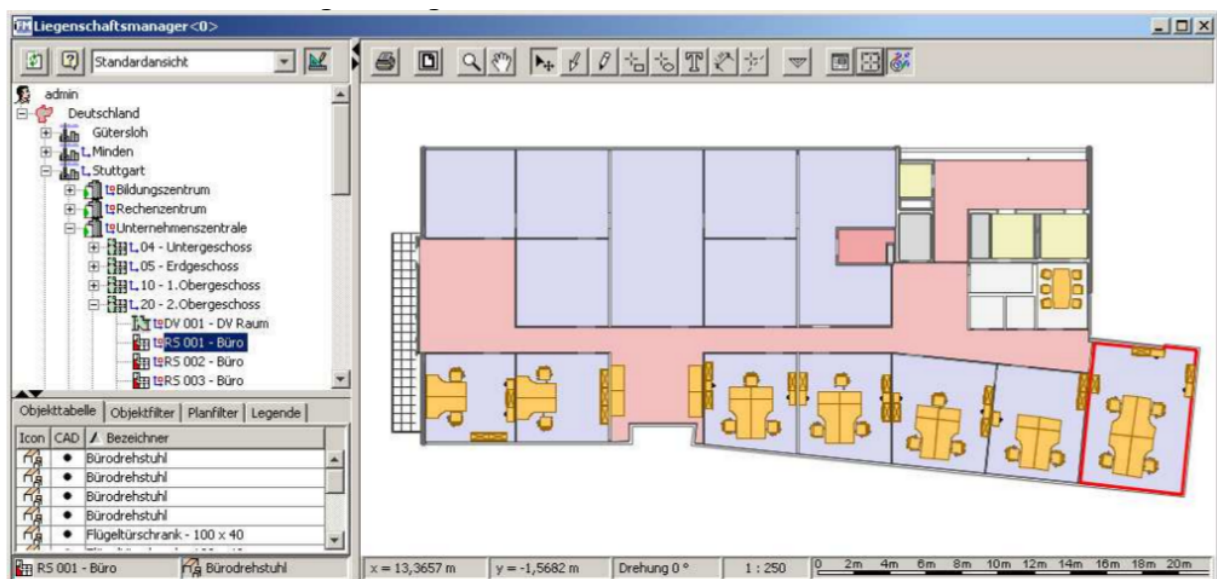


Abbildung 7: Screenshot iFMS aus der Benutzerdokumentation (Reply (2011))

lassen sich die Immobilien in einer quasi beliebig verzweigten Baumstruktur organisieren. So könnten zu den Kategorien in der Abbildung Land, Stadt, Gebäude, Etagen und Räume zum Beispiel noch Kontinente, Ländergruppen, Halbetagen oder Raumteile kommen. Auf der rechten Seite ist der CAD-Plan des Gebäudes bis auf Arbeitsplatzebene dargestellt. Arbeitsplätzen können Mitarbeiter, Telefonanschlüsse und beliebige Ausrüstungsgegenstände zugeordnet sein. Der Raumplan lässt sich ändern: Zwischenwände können gezogen oder entfernt werden. Vertragszugehörigkeiten, Raumnutzungsarten und Bodenbeläge lassen sich - auch für jeden Zeitpunkt in der Vergangenheit - visualisieren.

2.5 Das Fünf-Phasen-Wasserfallmodell

Das in (Rai; Sahoo; Mehfuz 2012) vorgeschlagene Vorgehensmodell zur Migration einer Anwendung in die Cloud ähnelt dem aus der Softwareentwicklung bekannten, iterativen Wasserfallmodell und besteht aus den folgenden fünf Phasen, die in Abbildung 8 dargestellt sind.

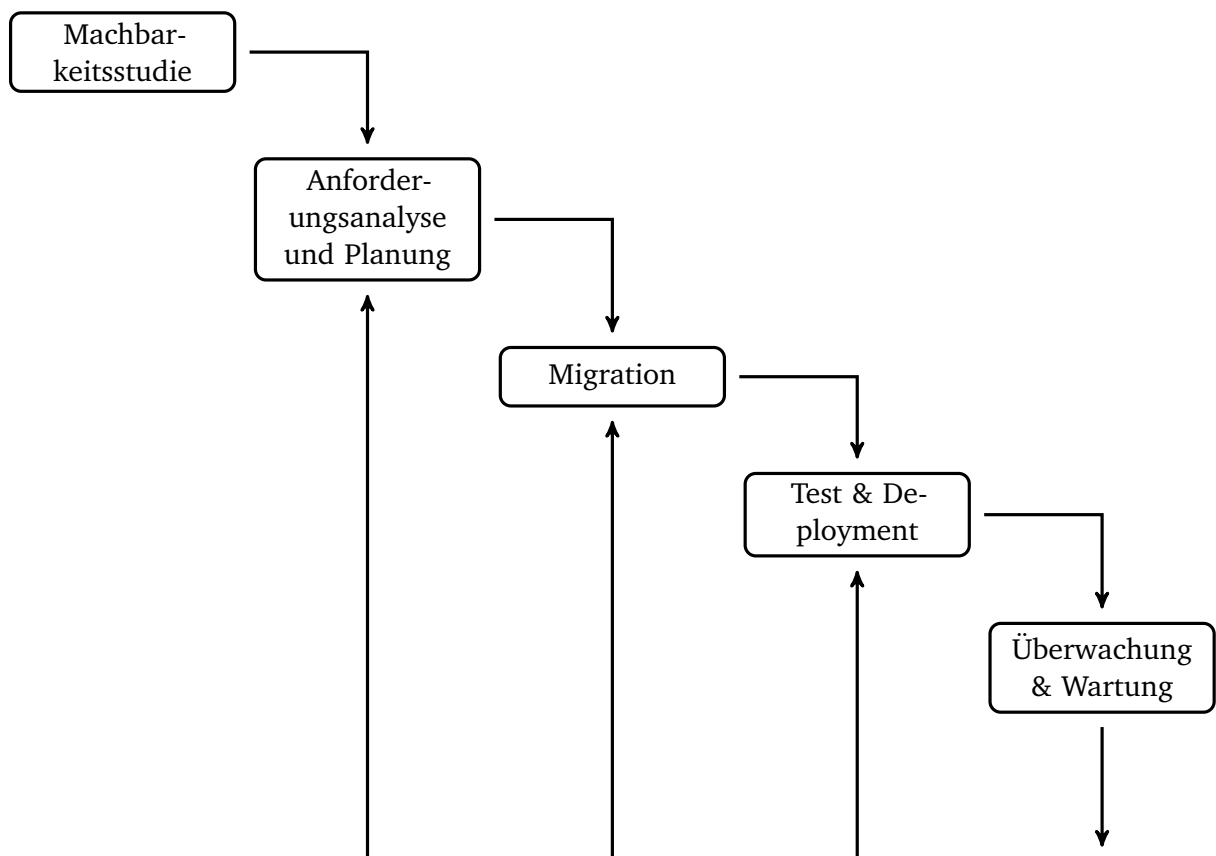


Abbildung 8: Das Fünf-Phasen-Wasserfallmodell aus Rai; Sahoo; Mehfuz (2012)

Phase 1 - Machbarkeitsstudie In dieser Phase wird ergebnisoffen geprüft, ob die Migration einer Anwendung technisch und wirtschaftlich möglich und sinnvoll ist. Dabei wird nicht nur die Anwendung selbst analysiert, sondern auch alle Rahmenbedingungen, die Einfluss auf das Verhalten des Systems ausüben können. Außerdem wird eine detaillierte Kosten-Nutzen-Analyse erstellt.

Phase 2 - Anforderungsanalyse und -Planung Um die zu migrierende Anwendung und ihre Anforderungen zu verstehen, wird in der Planungsphase die bestehende IT-Umgebung unter Berücksichtigung der genannten, die Migration beeinflussenden Faktoren (siehe Kapitel 2.3) genau begutachtet. Gilt die Anwendung auch nach Begutachtung als zur Migration geeignet, werden der Return on Investment (ROI) sowie die Total Cost of Ownership (TCO) berechnet, um die durch die Migration entstehende Kostenvorteile zu verstehen.

Phase 3 - Migration Die existierende Anwendung wird in die Cloud portiert und in Hinblick auf Leistungsfähigkeit und Performanz strukturiert getestet. Zum Schluss wird die neue Plattform in einem User Acceptance Testing (UAT) validiert.

Phase 4 - Tests und Auslieferung Die Daten aus der Produktion werden in die Cloud portiert. Anschließend wird die Software erneut getestet und freigegeben. In dieser Phase ist ein hoher Grade an Überwachung und Support nötig, um unvorhergesehene Probleme auffangen zu können. Unter Umständen wird parallel zum Start der Cloud-Anwendung die Altsoftware zunächst weiter betrieben.

Phase 5 - Überwachung und Wartung Nach der Migration in die Cloud ist es naturgemäß notwendig, die Leistungserfüllung durch den Anbieter in Hinblick auf Leistungsfähigkeit, Verfügbarkeit und Sicherheit zu überwachen um gegebenenfalls Gegenmaßnahmen einleiten zu können.

Das Fünf-Phasen-Wasserfallmodell, eine Weiterentwicklung des allgemeinen Wasserfallmodells, wird in dieser Arbeit als eine dem Leser vermutlich bekannte Grundlage genutzt; durch Vergleiche lassen sich die Ideen des in dieser Arbeit entwickelten Modells ordnen.

Um in dieses Modell Erfahrungen aus der Praxis einfließen lassen zu können

3 Vorgehensmodell für Cloud-Migrationen

In diesem Kapitel wird ein dreistufiges Vorgehensmodell (Vgl. Abbildung 9) für Cloud-Migrationen entwickelt, das sich an Independent Software Vendors mit ihren Bedürfnissen und Anforderungen richtet.

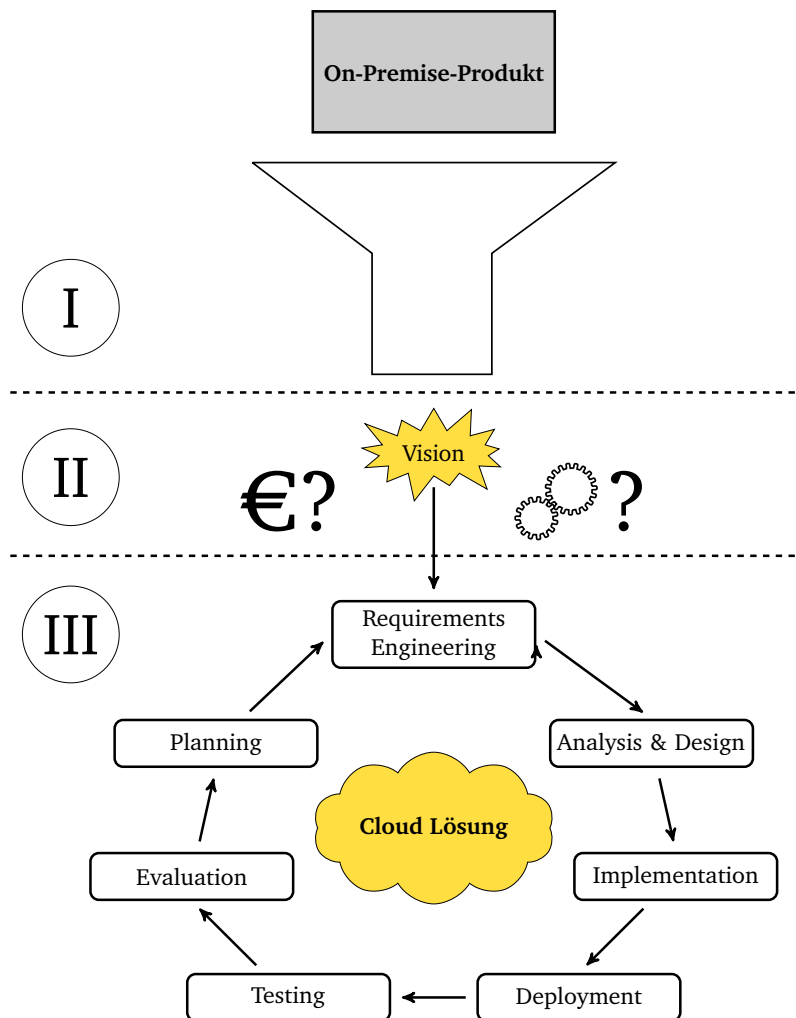


Abbildung 9: Darstellung des Vorgehensmodells. Der iterative Software Engineering Kreislauf wurde Schäfer & Lichter (2016) entnommen.

Das Vorgehen in jeder Phase ist von dem grundlegenden Ziel geprägt, Chancen der Cloud zu realisieren und spezifische Risiken zu vermeiden. Deshalb werden zunächst Chancen und Risiken identifiziert und mit den aus ihnen resultierenden Fragen, Problemstellungen und Herausforderungen vorgestellt. Um der Cloud mit ihren Chancen und Risiken bestmöglich zu begegnen sind auch weitreichende Änderungen am Unternehmen selbst zu bedenken.

In Phase I wird anhand des bestehenden On-Premise-Produktes eine Vision geformt, die unter Umständen einen sehr viel kleineren Funktionsumfang als die On-Premise-Software aufweisen wird, dabei aber Chancen und Risiken des Cloud-Marktes berücksichtigt und so Wettbewerbsfähigkeit und einen schnellen Markteintritt zu ermöglicht.

Anschließend wird in Phase II die Machbarkeit der Vision in technischer und wirtschaftlicher Hinsicht geprüft. Maßgeblich ist die Frage, ob innerhalb eines bestimmten Kostenrahmens auf gegebener Plattform ein konkurrenzfähiges Produkt geschaffen werden kann.

Schließlich wird die Vision in Phase III in einem an die Cloud angepassten, agilen, iterativen Entwicklungsprozess realisiert. Es wird gezeigt, in welchen Aspekten bekannte agile Modelle angepasst werden müssen, um mit der besonders hohen Agilität der Cloud Schritt halten zu können.

Bevor sie im Detail vorgestellt wird, wird jede Phase zunächst begründet und zur Orientierung vom Fünf-Phasen-Modell abgegrenzt.

3.1 Chancen der Cloud

Soziales Element

Beschreibung: Das soziale Element entsteht durch neue Möglichkeiten des Austausches, die durch die Nutzung einer gemeinsamen Cloud-Plattform entstehen: zwischen Nutzern innerhalb eines Unternehmens, zwischen verschiedenen Unternehmen oder mit den Entwicklern. Die Weiterentwicklung wird dadurch inklusiver, Nutzer lassen sich einbeziehen.

Quelle(n): (Sarkar et al. 2012; Schäfer & Lichter 2016; Höllwarth 2012)

Fragen & Herausforderungen: Wie lässt sich der Kontakt zu und zwischen Kunden und ihren Fachabteilungen so direkt, einfach und produktiv wie möglich gestalten? Lassen sich Communities aufbauen?

Analysemöglichkeiten

Beschreibung: Da sich alle Benutzer auf der Cloud bewegen, fallen viel mehr Informationen an, die analysiert werden können

Quelle(n): (Sarkar et al. 2012; Schäfer & Lichter 2016; Chappell 2012)

Fragen & Herausforderungen: Wie lassen sich künftige Entscheidungen mit den gewonnenen Informationen fundierter treffen?

Mobilität

Beschreibung: Lösungen aus SaaS-Bereich sind häufig bereits im Standard auf mobile Bedienbarkeit ausgelegt.

Quelle(n): (Sarkar et al. 2012)

Fragen & Herausforderungen: Um bezüglich Mobilität nicht nur Erwartungen zu erfüllen, sondern Begeisterung zu wecken, sollte geprüft werden, wie die gewonnene Mobilität im konkreten Fall den Kundenwert steigern kann.

Reduzierte Markteintrittskosten, Skalierte Märkte

Beschreibung: Durch pay-per-use-Modelle sind die Markteintrittskosten drastisch reduziert.

Quelle(n): (Marston et al. 2011; Chappell 2012)

Fragen & Herausforderungen: Mit welchen Produkten lassen sich neue Märkte erschlie-

ßen? Welche unerschlossenen Märkte gibt es? Wie lassen sich geographisch weit entfernte Märkte erschließen?

Skalierung der Leistung

Beschreibung: In der Cloud stehen – dynamisch an den aktuellen Bedarf angepasst – unbegrenzte Ressourcen bereit.

Quelle(n): (Marston et al. 2011)

Fragen & Herausforderungen: Wie lassen sich die Ressourcen nutzen, um gegenüber On-Premise-Anwendungen im Vorteil zu sein?

Time to market, kürzere Releasezyklen

Beschreibung: Die Software und auch Updates lassen sich schneller auf den Markt bringen.

Quelle(n): (Schäfer & Lichter 2016; Qian & Palvia 2013; Höllwarth 2012)

Fragen & Herausforderungen:

Alternativen testen

Beschreibung: In der Cloud lassen sich alternative Implementierungen testen und direkt auswerten.

Quelle(n): (Schäfer & Lichter 2016)

Fragen & Herausforderungen:

Sparen der Wartung älterer Versionen

Beschreibung: Kapazitäten werden frei, da es in der Cloud nur eine aktuelle Version gibt, in die alle Entwicklungsarbeit fließen kann und keine Altversionen gewartet und berücksichtigt werden müssen.

Quelle(n): (Schäfer & Lichter 2016; Olsen; Chappell 2012)

Fragen & Herausforderungen:

Standardisierte Komponenten

Beschreibung: Mit in der Cloud verfügbaren, standardisierten Komponenten lassen und loser Kopplung lassen sich Funktionen relativ leicht ergänzen

Quelle(n): (Qian & Palvia 2013; Sarkar et al. 2012)

Fragen & Herausforderungen:

Stetigere Umsätze

Beschreibung: Statt eines Kaufvertrages kommt ein Mietvertrag zu Stande, was zwar kleinere, aber stetigere Umsätze mit sich bringt

Quelle(n): (Höllwarth 2012; Chappell 2012)

Fragen & Herausforderungen:

Verkauf an Fachabteilungen

Beschreibung: Lösungen können direkt an Manager verkauft werden; Verhandlungen mit den IT Abteilungen des Kunden werden minimiert. Inkompatibilitäten in Hard-und Software beim Kunden werden vermieden.

Quelle(n): (Chappell 2012)

Fragen & Herausforderungen:

3.2 Risiken der Cloud

Hohe Kosten durch Pay-per-Use

Beschreibung: Die migrierte Software ist durch die für sie erforderliche dauerhafte Hochlast sehr viel teurer

Quelle(n): ()

Fragen & Herausforderungen:

Lock-in Effekte

Beschreibung: Der ISV ist von zwei Seiten vom Lock-in-Effekt betroffen: 1. Auf Plattformen, die er nutzt um für Kunden Software zu entwickeln. 2. Lock-in-Effekte, die der ISV seinen Kunden aufbürdet.

Quelle(n): (DaSilva et al. 2013)

Fragen & Herausforderungen:

Komplexität unbedacht gekoppelter Komponenten

Beschreibung: Auch wenn die Ankopplung standardisierter Cloud-Komponenten relativ leicht ist, erhöht sich mit jeder Komponente die Komplexität

Quelle(n): (Sarkar et al. 2012)

Fragen & Herausforderungen: Welche Kopplungen sind absehbar? Wie sieht eine Architektur aus die (absehbare) Kopplungen begünstigt?

Datenmigration

Beschreibung: Datenmigrationen oder der parallele Betrieb mit der alten Lösung können Inkonsistenzen in den Daten hervorrufen

Quelle(n): (Qian & Palvia 2013)

Fragen & Herausforderungen:

Leistungstransparenz

Beschreibung: Da der Kunde die Cloud-Lösung in der Regel vor Vertragsabschluss ausprobieren kann, muss der Anbieter einen echten Wert bieten.

Quelle(n): (Chappell 2012)

Fragen & Herausforderungen:

Geringere Umsätze

Beschreibung: Umsätze kommen zwar stetig aber in viel geringerer Höhe als bei einem Kauf.

Quelle(n): (Chappell 2012)

Fragen & Herausforderungen:

Geringere Anpassbarkeit

Beschreibung: Durch die in der Cloud übliche Standardisierung kann die Anpassbarkeit

leiden

Quelle(n): (Chappell 2012)

Fragen & Herausforderungen:

Organisatorische/Strukturelle Umbrüche

Beschreibung: Der Wechsel in der Cloud erfordert viele neue Ansätze, zum Beispiel bei der Lizenzierung oder dem Verkauf und stellt einen großen Umbruch dar.

Quelle(n): (Chappell 2012)

Fragen & Herausforderungen:

Updatefrequenz erfordert Agilität

Beschreibung: Da SaaS-Anwendungen werden typischerweise sehr viel häufiger geupdatet, als On-Premise-Anwendungen. Die hohe Updatefrequenz macht agile Entwicklungsmodelle erforderlich

Quelle(n): (Chappell 2012)

Fragen & Herausforderungen:

3.3 Phase I: Entwicklung einer Vision

Als „disruptive innovation“ ist die Cloud auch ein neuer Markt, auf dem qualitativ hochwertige, teilweise hoch spezialisierte IT-Dienstleistungen gehandelt werden. (Butterfield et al. 2016)

Durch das pay-per-use-Preismodell sind auch kleine Unternehmen in der Lage, diese Technologien und Dienstleistungen zu nutzen. (Höllwarth 2012, S. 156) Dementsprechend ist die Nutzung der Cloud per se weder innovativ – da jeder Mitbewerber die Technologie nutzen kann – noch automatisch ein Wettbewerbsvorteil: Sie ist ein wirtschaftliches Erfordernis, um den Wettbewerb nicht zu verlieren. (Oredo & Njihia 2014)

Um mit der Cloud-Migration nicht nur das Erfordernis zu bewältigen, sondern einen Wettbewerbsvorteil zu erringen, muss nach der Migration ein Wert für den Kunden geschaffen sein. Die Software muss unter aktuellen und möglichen zukünftigen Konkurrenten einzigartig oder wenigstens selten sein, annähernd unnachahmlich und schwer substituierbar sein. (Wright & McMahan 1992) Es bedarf es der geschickten Integration standardisierter, in der Cloud verfügbarer Komponenten zu einer innovativen Gesamtlösung.

Rai; Sahoo; Mehruz (2012) gehen in ihrem Fünf-Phasen-Modell von einer vollständigen Migration mit der Gesamtheit aller Funktionalität aus. Davon wird in dem hier vorgestellten Modell bewusst abgewichen. Bei einer vollständigen Migration dauert es aufgrund der höheren Entwicklungsdauer länger, bis Umsätze generiert werden; benötigte Investitionen steigen immens und mit ihnen das Risiko. Des Weiteren wäre es mit einer 1:1-Migration weder möglich die Chancen der Cloud zu realisieren, noch die Risiken zu umschiffen. Dies betrifft das Produkt selbst – so kann es beispielsweise versäumt werden, zu bedenken, wie mobile Nutzung einen besonderen Wert für den Kunden schaffen kann – aber auch den Entwicklungsprozess. Möglichkeiten, wie die der schnellen, unkomplizierten Releases und der frühen Nutzerrückmeldung und die besondere Agilität die sich daraus gewinnen lässt, werden verschenkt.

Aus diesen Gründen wird eine sprintweise Migration und Weiterentwicklung vorgeschlagen, die mit der Vision eines Produktes beginnt. Das Verständnis des Begriffes „Produktvision“ ist ein Strategisches und wurde aus den Eigenschaften abgeleitet, die Kantabutra & Avery (2010) Unternehmensvisionen attestieren.

Um ein wettbewerbsfähiges Produkt zu schaffen, sollte eine Produktvision...

- ...die genannten Chancen und Risiken - angelehnt an die Methode der SWOT-Analyse (Homburg 2017, S. 501), (Marston et al. 2011, S. 6) - berücksichtigen.
- ...den Umfang des Produktes so klein wie möglich, doch so umfangreich wie nötig halten, um frühestmöglich einen auslieferbaren Wert zu schaffen.
- ...stabil, zukunftsorientiert und langfristig ausgelegt sein.
- ...herausfordernd aber erreichbar sein.
- ...abstrakt genug sein, um Freiheiten bei der Umsetzung zu lassen.
- ...das Potential haben, Kunden und Mitarbeiter zu begeistern.
- ...klar und deutlich zu verstehen sein.

Damit die Vision realisiert werden kann, müssen Prozesse und Unternehmensorganisation an ihr ausgerichtet werden; sie muss kommuniziert werden und Mitarbeiter in die Lage versetzt werden, sie zu erreichen. (Kantabutra & Avery 2010)

Die Vision muss auch Einfluss auf das Geschäftsmodell nehmen. Gerade beim Auftreten von „disruptives innovations“ scheitern viele Unternehmen, weil sie nicht in der Lage oder willens sind, ihr Geschäftsmodell in ausreichendem Maße zu ändern. (DaSilva et al. 2013) Um ein wirtschaftlich nachhaltiges, an die Realitäten des Marktes angepasstes, wettbewerbsfähiges Geschäftsmodell für den Cloud-Markt zu entwickeln, sollte der ISV unter Berücksichtigung der Chancen und Risiken die folgenden Fragen beantworten DaSilva et al. (2013):

1. Wie entsteht für den Kunden Wert in Form eines Produktes oder einer Dienstleistung?
2. In welcher Form und Höhe und mit welchem Preismodell wird Umsatz generiert?
3. Wie können bereits bestehende und kommende standardisierte Komponenten und Dienstleistungen in das Produkt integriert werden?
4. Wie lassen sich bestehende oder zu erwerbende Ressourcen und Fähigkeiten (andersartig) nutzen, um neue Produkte oder Dienstleistungen zu erzeugen?
5. Mit welchen strategischen Entscheidungen lassen sich Wettbewerbsvorteile (auch gegenüber On-Premise-Lösungen und zugehörigen Preismodellen) erlangen?

3.4 Phase II: Machbarkeitsstudie

Phase II entspricht weitgehend der ersten Phase aus Rai; Sahoo; Mehfuz (2012), nämlich der Durchführung einer technischen und wirtschaftlichen Machbarkeitsstudie. Objekt der beiden Machbarkeitsstudien ist hier jedoch die Produktvision in der Cloud, keine Übersetzung der On-Premise-Software in die Cloud. Dies wirkt sich vor allem auf die Prüfung der Wirtschaftlichkeit aus. Der Vorschlag des Fünf-Phasen-Modells an dieser Stelle eine „detaillierte Kosten-Nutzen-Analyse“ durchzuführen, wird hier nicht befolgt. Um diese Entscheidung zu begründen, sollen die beiden Migrationsansätze noch einmal verglichen werden.

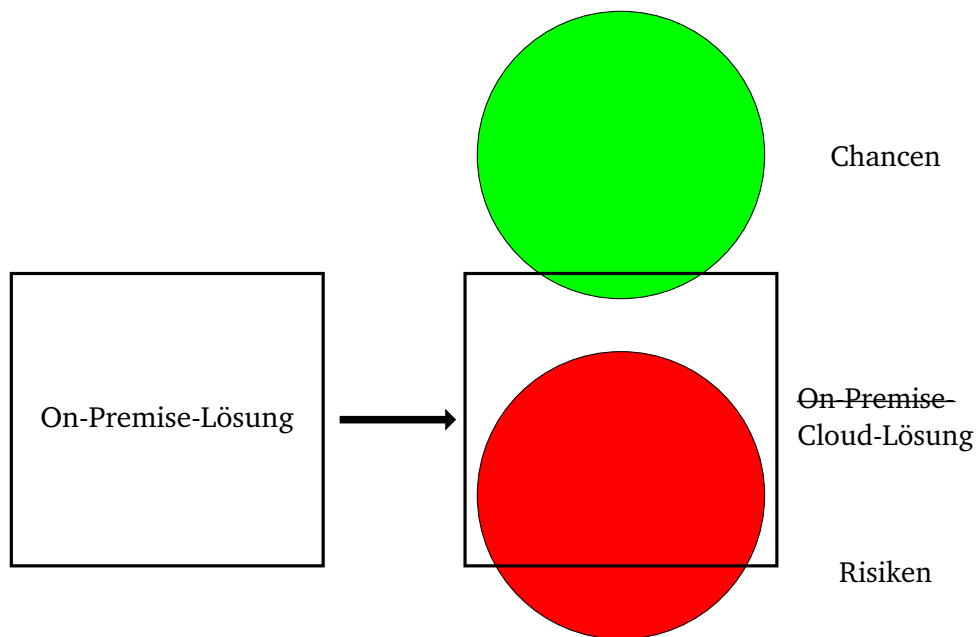


Abbildung 10: Funktionsumfang vor und nach der Migration im Wasserfallmodell. Selbsterstellte Grafik.

In Abbildung 10 ist der Ansatz aus dem Fünf-Phasen-Modell dargestellt. Dabei wird die bestehende Anwendung in ihrem gesamten Umfang in die PaaS/SaaS-Cloud migriert und erst daran anschließend vermarktet. Zumindest die Chance, eine agile Cloud-Lösung zu schaffen, die bereits früh einen Wert für Kunden schafft und sich schnell und an Kundenwünschen orientiert weiterentwickelt, wird verschenkt. Der ISV läuft aber auch Gefahr weitere Vorteile nicht zu realisieren, wenn er bei der Migration zunächst die Herstellung aller Funktionalitäten der Altsoftware in den Vordergrund stellt und dabei vernachlässigt, über Möglichkeiten nachzudenken, die Vorteile und Chancen der Cloud umzusetzen. Deshalb ist die Menge der Chancen nur zu einem geringen Teil in der Cloud-Lösung enthalten – im Gegensatz zu den Risiken, die Teil der Cloud-Lösung werden, wenn man sich ihrer nicht bewusst ist und sie daher nicht vermeiden kann.

Abbildung 11 zeigt die in dieser Arbeit vorgeschlagene Vorgehensweise. Hier ist die Cloud-Lösung in ihrem Umfang kleiner als die bestehende On-Premise-Software, sodass sie gerade das enthält, was für die Vermarktung nötig ist. Die Beschränkung des Umfangs begünstigt eine schnelle Fertigstellung und Vermarktung. Fehlende Features werden agil nachentwickelt und in

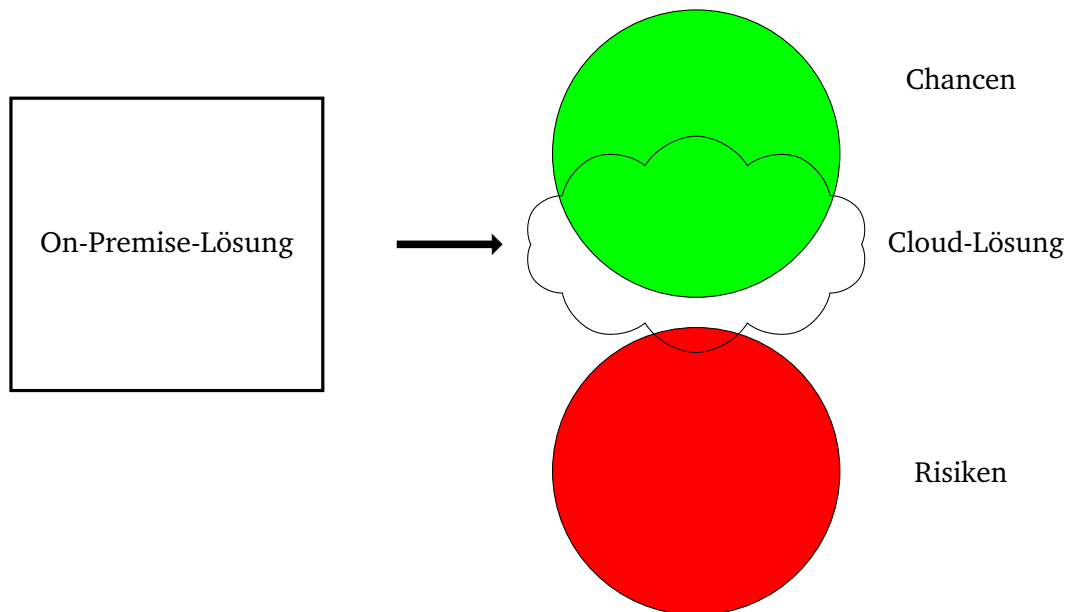


Abbildung 11: Funktionsumfang vor und nach der Migration nach dem in dieser Arbeit vorgestellten Vorgehensweise. Selbsterstellte Grafik.

kurzen Abständen freigegeben. In der Cloud-Lösung werden von Anfang an Chancen und Risiken berücksichtigt und maximiert beziehungsweise minimiert.

Im Ergebnis ist das hier vorgestellte Vorgehen weniger riskant, da nötige Investitionen geringer sind und Fehler wie Erfolge frühzeitig sichtbar werden. Da in diesem Modell zudem nicht die Kosteneinsparungen sondern strategische Ziele im Vordergrund stehen und der zeitliche Horizont viel kleiner ist, kann die Wirtschaftlichkeitsprüfung weniger detailliert ausfallen. Es sollte lediglich geprüft werden, ob nötige Umsätze mit dem Funktionsumfang der Vision generiert werden können.

Auf diesen letztgenannten Aspekt sollte ebenfalls die technische Machbarkeitsstudie abzielen: Ist mit der gewählten Cloud-Plattform die Umsetzung eines vermarktbaren Leistungsumfangs der Vision möglich? Dabei soll gerade nicht die gesamte Vision direkt umgesetzt werden. Sobald Kunden eine erste Version der Cloud-Lösung von Kunden genutzt wird, lässt sich erfragen oder ermitteln, wie die Entwicklung weitergehen soll; womöglich wird festgestellt, dass ein in der Vision vorgesehenes Feature gar keinen Anklang fände oder aber, dass die Vision in eine neue Richtung weiter gedacht werden muss.

3.5 Phase III: Agiler, iterativer Entwicklungsprozess

Die dritte Phase dieses Modells fasst die Phasen zwei bis fünf (Anforderungsanalyse und -Planung, Migration, Testen und Go-Live sowie Überwachung und Wartung) aus dem Fünf-Phasen-Modell zu einer Phase zusammen. In dieser Phase wird anstatt eines iterativen Wasserfallmodells ein agiles, sprintweise iteratives Modell verfolgt. Der Modellwechsel wird notwendig, weil sich in der Cloud die Art und Weise sowie der Rhythmus ändern, in der neue Versionen und Updates ausgeliefert werden. Bei konventioneller Software wurden Fehlerbehebungen häufig in Form

kleiner Updates in festen Zyklen, beispielsweise halbjährlich ausgeliefert. Neue Funktionalitäten kamen hingegen in Form von neuen Softwareversionen (Upgrades) zum Kunden, für die erneut gezahlt werden musste. Im Cloud-Betrieb gibt es regelmäßig nur eine Version der Software, in der nicht nur kontinuierlich Fehler behoben sondern auch neue Funktionen ergänzt werden, ohne, dass dies für die Nutzer mit Unterbrechungen verbunden wäre. (Schäfer & Lichter 2016; Olsen) Unter dieser Anforderung ist ein agiles Vorgehen im Sinne der Wettbewerbsfähigkeit besonders wichtig.

Der agile, iterative Entwicklungsprozess beginnt, wie in Abbildung 9 dargestellt, mit dem Requirements Engineering und der Produktvision als Input.

Das Requirements Engineering (Anforderungsmanagement) hat das Ziel Anforderungen eines Softwaresystem oder eines Features zu erheben, zu strukturieren, priorisieren und zu koordinieren. (Schäfer & Lichter 2016) Auch die Identifizierung von Diskrepanzen zwischen Ergebnis und Ziel gehört dazu. (Tariq; Khan; Iftikhar) Auch wenn sich an dieser Definition bei der Entwicklung in der Cloud nichts Grundsätzliches ändert, machen die Charakteristika der Cloud doch ein Überdenken der Prozesse im Rahmen des Requirements Engineering notwendig. (Schäfer & Lichter 2016; Olsen) Die Unterschiede zwischen Requirements Engineering bei On-Premise-Projekten im Vergleich zu SaaS-Projekten ist in Tabelle 2 aufgeführt.

On-Premise-Software	Software as a Service
Überschaubare Anzahl von Stakeholdern	Viele verschiedene Stakeholder
Keine oder geringe Einbeziehung des Kunden in die Entwicklung	Starke Einbeziehung
Geschäftsbeziehung zum Kunden endet mit einmaliger Zahlung	langfristige Geschäftsbeziehung
Nutzungserfahrungen nur über spezielle Erhebungen	Direkte Rückmeldung, ggf. motiviert durch Hoffnung auf Fehlerbehebung oder neue Features
Regelmäßige, geplante Fehlerbehebung	Sofortige Fehlerbehebung
Keine neuen Features ohne Versionsupgrade	Andauernde Auslieferung neuer Features ohne größere Verzögerung
Updates und Upgrade erfordern Downtime	Nahtloser Updateprozess ohne Unterbrechung
Upgrades haben größere Auswirkungen und machen Schulungen erforderlich	Kontinuierliche Auslieferung weniger disruptiv
Prognose und Test der Akzeptanz schwierig	Alternativen lassen sich an kleineren Nutzergruppen testen

Tabelle 2: Unterschiede im Requirementsengineering. Entnommen aus Schäfer & Lichter (2016)

Um die Änderungen im Requirements Engineering zu reagieren, empfehlen Schäfer & Lichter (2016) die folgenden Schritte:

-
1. Paradigma sich schnell ändernder Anforderungen etablieren. Die Nähe zur agilen Entwicklung und neuen Methoden der Anforderungsermittlung sorgen für eine Volatilität der Anforderungen.
 2. Integrieren der Anforderungsermittlung in einen iterativen, inkrementellen Software Engineering Prozess. Nicht typisch für die Cloud, aber besonders erforderlich aufgrund der volatilen Anforderungen.
 3. Identifizierung und Priorisierung der Stakeholder mit systematischen Methoden
 4. Einbeziehung der Kunden in die Anforderungsermittlung mittels Feature Requests und Bug Reports sind essentiell um Vorteil aus der Cloud-Migration zu ziehen. Nutzer müssen das Gefühl bekommen, Einfluss auf die Entwicklung nehmen zu können.
 5. Implementierung von Feedbackmöglichkeiten. Entweder automatisiert über die Auswertung von Daten oder über Formulare.
 6. Nutzen von Mechanismen für unterbrechungsfreie Updates. Nicht Administratoren, sondern Entwickler bestimmen den Updatezeitpunkt.
 7. Entwicklung von Möglichkeiten, Software an kleineren Nutzergruppen zu testen.

4 Forschungsmethoden

4.1 Theorie: Systematische Literaturübersicht

Grundlage des in dieser Arbeit entwickelten Vorgehensmodells ist eine systematische Literaturübersicht wie von Kitchenham (2004) vorgeschlagen. Dabei werden zunächst für jede Forschungsfrage Schlüsselwörter und ihre Synonyme identifiziert und anschließend mit booleschen Operatoren verknüpft. Die entstandenen Ausdrücke (Vgl. Tabelle 3) dienen anschließend der Recherche in Literaturdatenbanken (Vgl. Tabelle 4)

#	Frage	Rechercheausdruck
1	Welche Merkmale (Chancen und Risiken) der Cloud sind bei der Migration zu berücksichtigen?	(chances OR risks) AND (migration OR adoption OR transition) AND cloud
2	Wie beeinflussen die gefundenen Merkmale die strategische Ausrichtung und das Geschäftsmodell eines Softwareherstellers?	(strategy OR market OR business OR cloudonomics) AND (migration OR adoption OR transition) AND cloud
3	Wie beeinflussen die gefundenen Merkmale den Migrationsprozess eines Softwareherstellers?	(methods OR standards OR framework OR toolkit OR 'life cycle') AND (migration OR adoption OR transition) AND cloud

Tabelle 3: Forschungsfragen und zugehörige Rechercheausdrücke. Angelehnt an Rai; Sahoo; Mehruz (2015)

Bei der Literaturübersicht wurden Ergebnisse berücksichtigt, die

- den Rechercheausdrücken entsprachen
- in deutscher oder englischer Sprache vorlagen
- vollständig vorlagen
- in Abstract oder Fazit einen Zusammenhang zu den Forschungsfragen aufwiesen
- seit einschließlich dem Jahr 2010 erschienen sind

Die fachliche Disziplin der Fragen beeinflusste die Wahl der Literaturdatenbanken und ist in Tabelle 4 dargestellt. So wurde beispielsweise darauf verzichtet in technischen Datenbanken nach Literatur zu marktstrategischen Fragen zu suchen.

4.2 Praxis: Workshops und Einzelgespräche

Name und URL	Forschungsfragen		
	1	2	3
ACM Digital Library http://dl.acm.org/	✓	?	?
Science Direct http://www.sciencedirect.com/	?	?	?
Wiley http://eu.wiley.com/	Keine booleschen Ausdrücke möglich		
Elektronische Zeitschriftenbibliothek (EZB) http://rzblx1.uni-regensburg.de/ezeit/fl.phtml?bibid=TUDA	Keine booleschen Ausdrücke möglich		
Compendex https://www.elsevier.com/solutions/engineering-village/content/compendex	Keine booleschen Ausdrücke möglich		
AIS Electronic Library (AISEL) http://aisel.aisnet.org/	Keine booleschen Ausdrücke möglich		
Zeitschriftendatenbank (ZDB) http://dispatch.opac.ddb.de/DB=1.1/srt=YOP/	?	?	?
IEEE Xplore http://ieeexplore.ieee.org/Xplore/dynhome.jsp?tag=1	?	?	?
Springer-Online: Bücher/Beiträge des Springer Verlags http://www.springerlink.com	✓		
Rechercheangebot der ULB http://www.ulb.tu-darmstadt.de/recherche/	✓	✓	✓

Tabelle 4: Literaturdatenbanken und für welche Fragen sie herangezogen wurden. Quellen: Rai; Sahoo; Meh-fuz (2015) und Benlian (2016)

5 Das Projekt: iFMS@Salesforce

Mit dem entwickelten Vorgehensmodell soll nun beispielhaft die Cloud-Migration des in Kapitel 2.4 vorgestellten iFMS theoretisch geplant werden. In Kapitel 6 wird der entstandene Plan mit dem Vorgehen in der Praxis verglichen und diskutiert.

Die Cloudversion dieser Software wird hier „iFMS@Salesforce“ genannt und von iFMS abgegrenzt. Aus dem Namen geht die gewählte Cloud-Plattform hervor – eine strategische Entscheidung, die im selben Kapitel begründet wurde.

5.1 Phase I: Entwickeln einer Vision

Um zu der Produktvision zu gelangen, dienen die in Kapitel 3 identifizierten Chancen und Risiken als Checkliste bei einem Brainstorming. Das Ergebnis sind Epics, grobgranulare, funktionale und nicht-funktionale Anforderungen an die neue Software. (Pichler 2016) Im Wechselspiel mit den gesammelten Epics werden Geschäftsmodell und Strategie definiert. Einige Chancen und Risiken ziehen strategische Implikationen mit sich, andere Chancen werden durch eine gewählte Strategie ausgeschlossen. Zum Beispiel können Chancen die Erschließung neuer Märkte ermöglichen oder aber eine Realisierung derselben aus strategischen Gründen ausgeschlossen sein.

In dieser Arbeit liegt der Schwerpunkt auf dem Modell, das den Weg zur Vision beschreibt – nicht auf der iFMS@Salesforce-Vision selbst, die aus Sicht des Autors keinen allgemeinen Mehrwert besitzt. Deshalb wird dieser Weg hier dargestellt, auf eine vollständige Beschreibung der Vision jedoch verzichtet.

5.1.1 Realisierung von Chancen

Soziales Element: Vernetzung und Einbeziehung von Nutzern Die Vernetzung zwischen den Nutzern einer Firma erfolgt über Chatter, dem Salesforce Äquivalent von Facebook. Der ISV bietet Kundensupport über die Salesforce Service Cloud. Entwickler sind für Kunden direkter erreichbar. Durch die dort gesammelten Erfahrung können zukünftige Entwicklungsarbeiten zielgerichteter erfolgen. Mitarbeiter des Kunden sollen das Gefühl bekommen, dass ihre Anliegen durch den Support schnell und zufriedenstellend gelöst werden. Durch die schnelle Umsetzung von eingebrachten Verbesserungsvorschlägen, soll sich der Eindruck festigen, dass jeder Mitarbeiter am Produkt mitentwickeln kann und sich selbst im Produkt wieder findet.

Analysemöglichkeiten Der ISV hat Zugriff auf die Salesforceinstanzen seiner Kunden. Dadurch kann er Fehlkonfigurationen und Fehlbedienungen erkennen, Supportangebote verbessern oder Benutzeroberflächen intuitiver gestalten. Der ISV kann ebenfalls analysieren, welche Funktionen besonders häufig genutzt werden und die zugehörigen Abläufe verbessern oder automatisieren.

Mobile Nutzung Mit der Mobile App Salesforce1 ist die Salesforce Anwendung auch auf Mobilgeräten nutzbar. Damit ein echter Mehrwert für den Kunden und seine Mitarbeiter entsteht, sollen sich

- Rauminformationen wie Belegungen und Ansprechpartner abrufen lassen.
- Räume buchen lassen.
- Schadensmeldungen eingeben lassen und nach Freigabe durch einen verantwortlichen Mitarbeiter in einem Auftrag an einen Dienstleister resultieren.
- über Anwesenheitserkennung Raumtemperatur und Licht regeln lassen.
- Räume über ein Indoornavigationssystem finden lassen.
- Reinigungspläne einsehen und ändern sowie Sonderreinigungen anfordern lassen.
- erfolgte Dienstleistungen wie Reinigungen, Reparaturen oder Caterings bewerten lassen, um die Auswahl künftiger Dienstleister besser zu gestalten.

Reduzierte Markteintrittskosten & Skalierte Märkte Um die Hürden für potentielle Kunden gering zu halten und das Produkt auf diese Weise auch für kleine und mittlere Unternehmen attraktiv zu machen, ist die SAP- und CAD-Anbindung optional. Durch die Nutzung des Salesforce Marktplatzes für Apps können Kunden das Produkt leicht beziehen und in einer Basisversion testen beziehungsweise nutzen.

Die Erschließung globaler Märkte erfolgt einerseits über den Marktplatz, andererseits über die gezielte Ansprache großer, internationaler Unternehmen, um iFMS@Salesforce langfristig international zu etablieren.

Skalierung der Leistung Raumbelegungen lassen sich über die verfügbare Leistung in akzeptabler Zeit in verschiedenen Dimensionen optimieren.

CAD-Pläne werden nicht mehr lokal sondern in der Cloud konvertiert.

Time to market & kürzere Releasezyklen Eine vermarktbare Basisversion mit einem Bruchteil der Funktionalitäten der komplexen Altsoftware soll rasch entwickelt, vermarktet und genutzt werden um bei Verhandlungen mit Neu- und Altkunden Interesse zu wecken.

Die langjährige Erfahrung mit CAD und SAP sollen in Zusammenarbeit mit in Salesforce erfahrenen Kollegen die Kernkompetenzen zügig in die Cloud übertragen werden.

Alternativen am Kunden testen Die Kultur des Mitentwickelns (siehe Abschnitt zur Einbeziehung von Nutzern oben) soll genutzt werden, um es interessierten Powerusern in Sandboxes den aktuellen Entwicklungsstand zu testen und direktes Feedback zu geben, das in die nächsten Sprintplanungen einfließen kann.

Wartung einer einzigen Version Der Wartungsaufwand soll reduziert werden, indem es jeweils nur eine Version der Basis-, SAP- und CAD-Komponenten gibt. Anpassungen werden dem

Kunden als Zusatzdienstleistung verkauft, sodass zusätzlicher Wartungsaufwand zu zusätzlichen Umsätzen für den ISV führt.

Standardisierte Komponenten Standardisierte Komponenten lassen sich auf dem Salesforce App Marktplatz beziehen. Beispiel dafür ist die nahtlose Anbindung an Amazons Dateispeicher S3, um es Kunden zu ermöglichen Dateien zu Liegenschaften oder Verträgen zu hinterlegen. Da der ISV keine Kernkompetenz beim Bearbeiten von CAD-Plänen besitzt, soll eine entsprechende Komponente aus dem Marktplatz bezogen werden, sodass der Kunde für keinen Prozess iFMS@Salesforce verlassen muss.

Stetige Umsätze Durch ein Preismodell in dem pro Nutzer und Monat bei einer Mindestlaufzeit von einem Jahr abgerechnet wird, lassen sich stetige Umsätze generieren, die dem ISV mehr Planungssicherheit geben.

Verkauf an Fachabteilungen Im Marketing soll ein Schwerpunkt darauf gelegt werden, Fachabteilungen direkt zu erreichen, zum Beispiel bei Fachtagungen einschlägiger Themen. Idealerweise lassen sich kleine Firmen erreichen, die das Thema Liegenschaftsmanagement bisher ohne Softwareunterstützung bewältigt haben. Sind Fachabteilungen überzeugt lässt sich das Produkt ohne Eingriffe in die IT-Landschaft vor Ort nutzen; eine Einbeziehung der IT-Abteilung ist nicht mehr oder nur in geringem Umfang nötig.

5.1.2 Vermeidung von Risiken

Hohe Kosten durch Pay-per-Use Salesforce rechnet pro Nutzer und Monat ab, sodass weder für den ISV noch für den Kunden mit unerwartet hohen Kosten zu rechnen ist.

Lock-in Effekte Arlanis Reply hat sich strategisch an Salesforce gebunden; der Lock-in Effekt wird in Kauf genommen.

Komplexität unbedacht gekoppelter Komponenten Bei Bezug von Komponenten aus dem Salesforce Marktplatz wird darauf geachtet nur aktiv gepflegte Komponenten mit einem hohen Reifegrad auszuwählen. Außerdem soll vor der Wahl einer bestimmten Komponente die Komplexität des Gesamtsystems in einer Kosten-Nutzen-Analyse betrachtet werden.

Datenmigration Zahlen- und Mengenmäßige Zugriffsbeschränkungen auf die Datenbank durch Salesforce könnten eine Schrittweise Migration der Daten erforderlich machen. Um den Parallelbetrieb mit Altsystemen so kurz wie möglich zu halten, soll ein leicht zu verstehendes Konzept erarbeitet und implementiert werden, mit dem ein zügiger Umzug von logischen Teilkomponenten, wie Etagen, Standorten oder Ländern ermöglicht wird.

Leistungstransparenz Um potentielle Kunden, die iFMS@Salesforce in der Basisversion testen vom Produkt zu überzeugen, sollen bei der Entwicklung Schwerpunkte auf eine intuitive Bedienung und begeisternde Features gelegt werden.

Geringere Umsätze Auch das bisherige Geschäftsmodell sah einmalige Zahlungen für Installation und Anpassung und fortlaufende Zahlungen für die Nutzung vor. Daher besteht dieses Risiko nicht.

Geringere Anpassbarkeit Mit Force.com und Heroku wird Salesforce zu einem PaaS-Dienstleister; Features, die sich mit der Kernfunktionalität oder Erweiterungen von Salesforce nicht umsetzen lassen, lassen sich auf diese Weise nahezu ohne Einschränkungen realisieren.

Organisatorische und strukturelle Umbrüche Die Einarbeitungszeit des alten iFMS-Teams in die Cloud-Technologie soll durch eine Team-Zusammensetzung mit erfahrenen Salesforce-Entwicklern und -Beratern reduziert werden.

Da sowohl Arlanis Reply und Syskoplan Reply als Tochtergesellschaften bereits eng vernetzt sind, ergeben sich wenige strukturelle Änderungen. Es muss allerdings darauf geachtet werden, dass das alte iFMS-Team trotz totaler Abkehr von der bisher genutzten Technologie im neuen Team und im neuen Produkt aufgeht.

Updatefrequenz erfordert Agilität Bisherige Ansätze der agilen Entwicklung werden systematisch mit neuen Entwicklungen im Bereich des Software Engineerings und des Projektmanagements verglichen und gegebenenfalls angepasst. Die Einbeziehung der Nutzer in die Entwicklung macht eine Berücksichtigung ihrer Anliegen in der Springplanung in besonderem Maße erforderlich.

5.1.3 Geschäftsmodell & Strategie

Die strategischen Implikationen einzelner Chancen und Risiken treten deutlich hervor; sie beeinflussen die klassischen „Vier Ps“ des Marketingmixes und spiegeln sie wider (Homburg 2017, S. 1003):

Product – Produktpolitik Wie soll das zukünftige Produkt iFMS@Salesforce aussehen? Mit welcher Technik, in welcher Qualität und in Verbindung mit welchen zusätzlichen Dienstleistungen soll es umgesetzt und angeboten werden?

Price – Preispolitik Welchen Preis sind die Kunden für die Basis-Version, welchen für CAD- und SAP-Unterstützung zu zahlen? Welches Preismodell wird gewählt (Siehe Kapitel 2.2)?

Promotion – Kommunikationspolitik Wie werden (Neu-)Kunden erreicht und dauerhaft gewonnen? Wie wird die Kommunikationspolitik durch das soziale Element der Cloud unterstützt?

Place – Distributionspolitik Gelangt iFMS@Salesforce über den Salesforce Marktplatz zum Kunden oder wird das Produkt für den Kunden in seiner Salesforce-Instanz aufgesetzt und angepasst?

Die Festlegung von Strategie, Marketing und Geschäftsmodell muss daher unter besonderer Beachtung der genannten Aspekte erfolgen.

5.2 Phase II: Machbarkeitsstudie

Die Machbarkeitsstudie untersucht die Vision, die ein strategisches Ziel und eine weitgehend ungeordnete Menge von Anforderungen beinhaltet. Dabei sollen die Epics, die im Sinne der Strategie ausgewählt wurden, auf ihre technische und wirtschaftliche Machbarkeit geprüft werden.

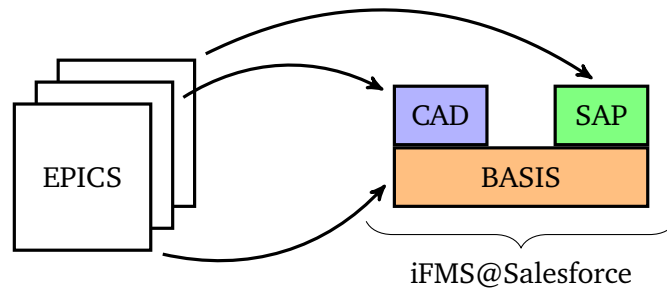


Abbildung 12: Schematische Darstellung der Zuordnung von Epics

Wie in Abbildung 12 dargestellt soll iFMS@Salesforce aus drei Versionen bestehen: der Basis-Version sowie der CAD- und SAP-Erweiterung. Aus der strategischen Ausrichtung heraus werden nun den Versionen Epics zugeordnet, sowie Funktionen die unbedingt notwendig sind um die jeweilige Version zu vermarkten. Der Preis – ob Verkaufspreis oder Lizenzierungskosten – kann aufgrund der langjährigen Erfahrung im On-Premise-Geschäft gut abgeschätzt werden. Aus dieser Erfahrung oder direkten Gesprächen heraus können die im Vertrieb tätigen Mitarbeiter auch die Bereitschaft der Kunden abschätzen, den Weg in die Cloud mitzugehen. Die Salesforce- und iFMS-Entwickler bei Reply schätzen nun in einem ersten Schritt die grundsätzliche technische Umsetzbarkeit ein.

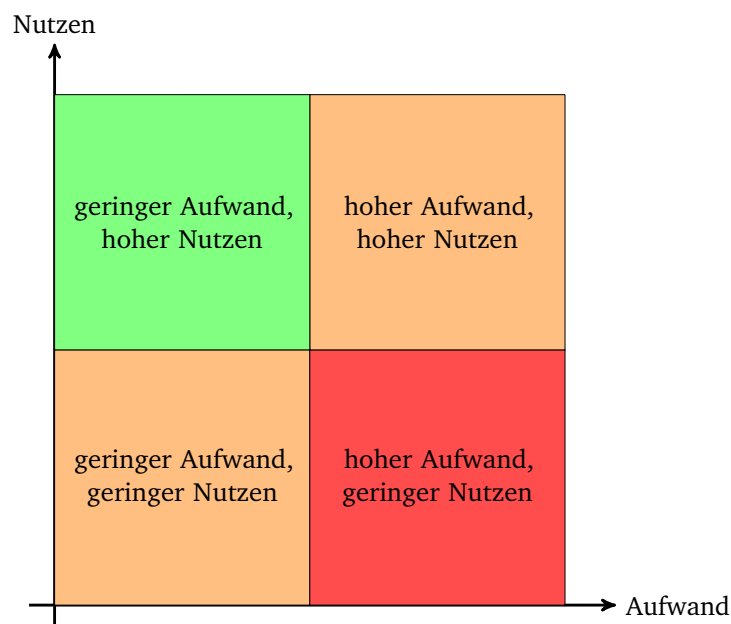


Abbildung 13: Kosten-Nutzen-Analyse der Anforderungen

Falls die bescheinigt wird, erfolgt im zweiten Schritt eine grobe Aufwandsschätzung der einzelnen Anforderungen. Mit den geschätzten Aufwänden lässt sich nun die Wirtschaftlichkeit des Projektes abschätzen. Einerseits um zu überprüfen, ob die vom Marketing erwarteten Gewinne erzielt werden können. Andererseits können Anforderungen auf ihr Kosten-Nutzen-Verhältnis hin analysiert werden. Dazu können in einer Kosten-Nutzen-Matrix wie in Abbildung 13 positioniert werden. Gerade Punkte aus dem linken, oberen Quadranten sind potentiell geeignet in der Basis-Version Kunden zu begeistern, ohne hohe Kosten zu erzeugen. Andererseits kann es vertretbar sein, dass die Basis-Version keine Gewinne abwirft, wenn mit ihr Kunden von den Erweiterungen oder Anpassungsdienstleistungen überzeugt werden sollen.

5.3 Phase III: Agiler, iterativer Entwicklungsprozess

Phase III beginnt – wie in Abbildung 9 dargestellt – mit dem Requirements Engineering. Das Entwicklungsteam sollte sich zunächst auf die Basisversion konzentrieren, um möglichst schnell ein Produkt zu schaffen, das vom Vertrieb einerseits vermarktet werden kann und die weitere Entwicklung finanziert und andererseits hilft, neue Kunden zu gewinnen. Um dem Aspekt der Kundengewinnung besser gerecht zu werden, könnte es hilfreich sein, die CAD-Erweiterung früh ins Auge zu fassen, da bei ihr – im Gegensatz zu der SAP-Erweiterung – optisch ansprechende Resultate zu erwarten sind. Neben kurzen Sprintzyklen soll eine Feedbackkultur und eine möglichst direkte, wenig formalisierte Kommunikation zwischen Entwicklern und Anwendern beim Kunden etabliert werden.

6 Diskussion

Im vorletzten Abschnitt diskutieren Sie Ihre Ergebnisse und stellen den Beitrag für die Praxis und für die Forschung dar. Gehen Sie auch auf die Einschränkungen Ihrer Arbeit ein.

7 Zusammenfassung und Ausblick

Zuletzt fassen Sie Ihre Arbeit kurz zusammen und stellen Ihre wichtigsten Schritte, Ergebnisse und Befunde dar. Geben Sie auch einen Ausblick auf mögliche anknüpfende Forschungsarbeiten. Außerdem findet sich hier Platz für eine kritische Hinterfragung einzelner Teilaspekte und auch für Ihre eigene Meinung.

7.1 Offene Forschungsfragen

- Wie kann sich ein ISV auf die strukturellen Änderungen durch die Cloud-Migration bei seinen Kunden einstellen?
- Die identifizierten Chancen und Risiken hatten Auswirkungen im Bereich Visionsentwicklung/Strategie und der Anforderungsermittlung. Wie sind die anderen Phasen betroffen.
- Einflüsse der Cloud auf Marketing, Strategie und Geschäftsmodell

7.2 Schwierigkeiten

Trennung zwischen Kundensicht und ISV-Sicht. Siehe Dreieck.

7.3 Abgabedokument

Abschlussarbeiten (Bachelor-, Master-, Diplomarbeit) sind in zweifacher Ausführung, einseitig bedruckt und gebunden abzugeben. Dazu auf CD die Abschlussarbeit in digitaler Form (z.B. Word und PDF), inkl. der Endnote-Projektdatei und der Grafiken.

8 Ideen

8.1 Einleitung

Benlian SaaS 2010: Chancen und Risiken für diesen Anwendungsfall aus Anwendersicht prüfen. In Softwareindustrie werden die einzelnen Chancen und Risiken genauer ausgeführt. S. 236 Auch Chancen und Risiken aus Anbietersicht. Softwareindustrie: S: 240 => Verweis auf Benlian 2010, S. 233

Wind: Eval. und Auswahl von Enterprise Cloud Services: Konkretisierung von Zieldimensionen (Flexibilität, Kosten, Leistungsumfang & Leistungsfähigkeit, Service & Cloud Management, IT-Sicherheit & Compliance, Ausfallsicherheit & Vertrauenswürdigkeit). Ab Seite 103. Anforderungsrahmen für die Zieldimensionen und den einzelnen XaaS-Arten. Ab S. 122. Viele Definitionen für Cloud.

Benlian Opportunities and risks of saas 2011: Sicherheit ein Hauptfaktor bei Entscheidung für oder gegen SaaS. Taxonomy of it security risks als Checkliste zur Identifikation von Risiken in bestimmten Szenarien.

Softwareindustrie: Simple Definition für Cloud aus Standard.

Ackermann, Tobias: IT Security Risk Management. Kapitel 5 enthält Empfehlungen für Risk Identification, - Quantification, - Treatment, - Review and Evaluation, - Cloud Computing Providers. S. 22-23 enthält Beschreibung der Risiken im Cloud Kontext.

Key players in the cloud computing industry (Marston et al. 2011, S. 4)

8.2 Gamechanger Cloud

„What is it about cloud that makes it a game-changer? It is reported that the business at large find cloud's affordable, flexible, on-demand, elastic delivery method, to be extremely beneficial“ – (Sarkar et al. 2012)

8.2.1 Lifecycle

Aus Sarkar et al. (2012)

Design RE und Design des Dienstes.

Interoperabilität Unter anderem Amazon, Google, Microsoft, VMWare und Salesforce bieten Cloud-Plattformen zur Anwendungsentwicklung an. Bei Nutzung einer solchen Plattform kann nachhaltige Interoperabilität, gerade bei Nutzung einer privaten oder hybriden Cloud ein Faktor sein.

Information Centric Design Bei der Nutzung durch Kunden anfallende Informationen sollten ständig ausgewertet werden und möglichst in Echtzeit in zukünftige Entscheidungen einfließen.

Engineering Entwicklung, Testen und Operationalisierung

Deployment

Usage Measurement

Support und Wartung Informationen, die im Support gesammelt werden, werden zur Weiterentwicklung genutzt.

Experience Um Kunden zu halten, muss die User Experience verbessert werden

8.2.2 Towards Modelling a Cloud Applications Life Cycle

Aus Butterfield et al. (2016)

- Bisherige Life Cycle Modelle fokussieren die Sicht der IT. Es fehlt ihnen an einer angemessenen Betrachtung wirtschaftlicher Aspekte.
- Schritte im Life-Cycle:
 - Business Case Definition
 - Decision Phase
 - Design Phase
 - Test-driven Development Phase
 - Deployment
 - Operations (monitoring, updates, resource adaptations, Decommissioning)
- Gründe in die Cloud zu gehen:
 - Finanzielle Vorteile
 - Wettbewerbsvorteile
 - Flexibilität durch On-Demand und self-service
 - Geringere Risiken
- Identifizierte Risiken
 - Sicherheit und Haftungsfragen
 - Technische Probleme

-
- Fehlende Standards
 - Literatur zu Cloud und Marketing konzentriert sich auf technische und finanzielle Faktoren im Deployment und vernachlässigt die Supply Chain des Unternehmens und die Auswirkungen auf das Unternehmen selbst
 - Die Cloud ermöglicht es einem Unternehmen unabhängig von seiner Größe, spezialisierte, hochwertige IT-Dienstleistungen in Anspruch zu nehmen.
 - Die Cloud ist eine “disruptive innovation”, ein “new Market”, auf dem IT als Dienstleistung eingekauft wird. Dieser Markt erfordert einen grundlegenden, kulturellen Wandel, mit dem die IT gesehen wird.
 - Opportunities and Impact
 - Verpasste Gelegenheiten durch Über- oder Unterprovisionierung vermeiden
 - An variable Nachfrage angepasste Dienstleistung
 - Das Verfolgen von wachsenden oder völlig neuen Märkten
 - Tätigkeiten schneller und günstiger durchführen
 - Entkoppeln des Kerngeschäfts von unterstützenden Dienstleistungen
 - Höhere Umsätze durch amortisierende Skaleneffekte
 - Höhere Umsätze durch höhere Reichweite
 - Unternehmen, die in die Cloud gehen, müssen sich den Auswirkungen auf Strategie, Business Model und Unternehmensstruktur bewusst sein.

8.2.3 Service Migration – Herausforderungen

Aus Sarkar et al. (2012)

- Was für Auswirkungen hat das neue pay-per-use-Modell?
-

8.3 Charakteristika Enterprise Applications

Aus Hajjat et al. (2010):

Bestehen in der Regel aus mehreren Layern (MVC – Datenbanken, Frontend, Logik), sind in der Regel aber viel komplexer als dieses dreischichtige Modell, da jede Schicht aus mehreren, miteinander interagierenden Komponenten aufgebaut sein kann.

Unternehmensanwendungen könnten von zwei verschiedenen Nutzergruppen nutzbar sein: Unternehmensinterne und -externe Personengruppen.

8.4 Assessment und Guidelines

Aus Scandurra et al. (2015):

8.4.1 Assessment

Gründe für die Migration Es ist wichtig die Gründe für die Migration sowie Anforderungen und Rahmenbedingungen zu kennen, um ihnen zu genügen.

Analyse der Anwendungsumgebung Alle Programme, Skripte und Interfaces listen, die auf die Anwendung zugreifen.

Analyse der neuen Umgebung Welche Ressourcen werden benötigt?

Design und Analyse der Architektur Analyse der Architektur der bestehenden Anwendung mit allen Bibliotheken, Programmen und Plattformen

Migrationstools Kleine Proof-of-Concept-Projekte um Migrationstools nach Effizienz, Genauigkeit und Optionen testen.

8.4.2 Guidelines

Interoperabilität und Zusammenstellen Nutzen von Schnittstellen wie der RESTful API.

Vermeiden des Lock-in Effektes

Modellierung des Dienstes und Cloud Deployment Artefakte Cloud Modellierungsnotationen sind nützlich.

Legacy Software verpacken Es ist unter Umständen einfacher, praktikabler, sicherer und flexibler Legacy Anwendungen virtualisiert in die Cloud zu migrieren und über eine Schnittstelle verfügbar zu machen.

Ausbalancieren von Kosten und Reifegrad Die Risiken junger, eventuell unreifer Software sollten in den Aspekt Kosten einbezogen werden.

8.5 Migrating to – or away from the public cloud

Aus Weinman (2016):

- Bei “Cloud migration” zwischen Migration der Architektur oder der Operationen.
- Migration der Architektur: Migrieren der bestehenden Anwendung zu einer skalierbaren, cloud-bereiten Architektur, häufig unter der Verwendung neuer Programmiersprachen wie Googles Go oder Apples Swift und/oder neuen Komponenten wie NoSQL-Datenbanken.
- Migration der Operationen: Verschieben einer cloud-ready Anwendung von der Private in die Public Cloud oder umgekehrt.

-
- Umdenken erforderlich: Bestehende Anwendungen besitzen häufig eine monolithische Natur, bei der das Neuschreiben einer Zeile Code die gesamte Anwendung beeinträchtigen kann. Deshalb sind nach der Änderung Tests, Neukompilieren und Deployment nötig. Hingegen bestehen Cloud-Anwendungen aus unabhängigen, verbundenen Komponenten
 - Gezogene Lehren:
 1. Jedes Unternehmen ist anders
 2. Technologien, architektonische Best-Practice Modelle und wirtschaftliche Rahmenbedingungen ändern sich. Auch aus dieser Sicht kann eine modulare Architektur positiv sein.
 3. Kosten sind der einzige Faktor, nicht mal der wichtigste.
 4. Je einfacher die Migration durch die Nutzung von spezifischen Schnittstellen wie AWS Lambda wird, desto stärker der Lock-In-Effekt.
 5. Selbst wenn die Cloud teurer erscheint, können – gerade bei Firmen im Umbruch, wie Start-ups, bei Wachstum wie bei Regression – die flexible Skalierbarkeit von Vorteil sein.
 6. Performancevorteile wiegen
 7. Die Nachfrage nach Rechenleistung schwankt etwa periodisch, während der Speicherplatzbedarf monoton wächst und daher vorhersagbarer ist. Entsprechend könnte es sich für ein Unternehmen mit einer speicherplatzlastigen Anwendung eher lohnen eigene Infrastruktur zu betreiben.

8.6 Towards an Understanding of Cloud Computing's Impact on Organizational IT Strategy

Aus Qian & Palvia (2013)

- Identifizierte Auswirkungen auf die IT Architektur:

Entwicklung einer skalierbaren, agilen IT-Architektur Implementierungen müssen mit Lastspitzen und Leerlauf umgehen können. Um das zu erreichen ist schnelles Lernen erforderlich und der Umstieg von einer lang geplanten, wasserfallartigen Entwicklung und Betrieb zur schnellen Entwicklung und Deployment

Langfristige Cloud Integrationsstrategie Cloud-Dienste sollten in das bestehende Business integriert werden. Darauf ist auf die Integration zu achten, da das System mit der steigenden Zahl genutzter Dienste bei verschiedenen Anbietern komplexer wird. Deshalb sollte die Verbindung zwischen Cloud Diensten, aber auch die Verbindung von On-Premise-Anwendungen mit der Cloud langfristig gedacht werden.

Management geht in Richtung Service Oriented Management

Auswirkungen auf die Datenhaltung Drei Fragen:

1. Was wird migriert und was nicht?
2. Welches Tool wird genutzt und wie sieht der Zeitplan aus?
3. Was das schlimmste Szenario, das auftreten kann wenn Daten verloren gehen?

Neue Strategie zur Kontrolle der Daten Zwei Formen des Kontrollverlustes über Daten: Ort und Zugriffsrechte. Erhöhte Nachfrage nach Transparenz.

IT/Business Alignment Höhere Entwicklungs- und Deploymentgeschwindigkeit mit weniger involvierten Menschen, geringeren Kosten und besserer Performance.

Sich nicht um technische Details kümmern zu müssen, hilft Unternehmen sich auf ihr Kerngeschäft zu konzentrieren. Diese Perspektive ermöglicht auch bessere IT Investitionen, auch weil Einsparungen an sinnvolleren Stellen reinvestiert werden können.

8.7 Inhaltsbeschreibungen

Sarkar et al. (2012) Viele zu bedenkende Aspekte, nicht nur für die Migration, sondern auch für den Cloudbetrieb. Eine Zusammenfassung findet sich auf Seite 9 des PDFs.

Hajjat et al. (2010) Hybride Cloud. Darstellung Charakteristike Enterprise Applications. Sonst extrem mathematisch.

Literatur

- Ahmad, Aakash & Babar, Muhammad Ali (2014): *A framework for architecture-driven migration of legacy systems to cloud-enabled software*, In: Proceedings of the WICSA 2014 Companion Volume,, S. 1–8.
- Alkhalil, Adel; Sahandi, Reza & John, David (2016): *A Review of the Current Level of Support to Aid Decisions for Migrating to Cloud Computing*, In: Proceedings of the International Conference on Internet of things and Cloud Computing,, S. 58, ISSN 1450340636.
- Azeemi, Imran Khan; Lewis, Mike & Tryfonas, Theo (2013): *Migrating To The Cloud: Lessons And Limitations Of ‘Traditional’ IS Success Models*, In: Procedia Computer Science, 16, S. 737–746 <http://www.sciencedirect.com/science/article/pii/S1877050913000781>, ISSN 1877–0509.
- Benlian, Alexander (2016): *Formatvorlage zur Anfertigung von Abschluss-, Studien- und Seminararbeiten*, http://www.ise.tu-darmstadt.de/ise/lehre_4/abschlussarbeiten_6/organisatorisches_1/organisatorisches.de.jsp.
- Benlian, Alexander; Hess, Thomas & Buxmann, Peter (2010): *Software-as-a-Service : Anbieterstrategien, Kundenbedürfnisse und Wertschöpfungsstrukturen*, Wiesbaden Gabler <http://dx.doi.org/10.1007/978-3-8349-8731-0>, ISBN 383498731X (Sekundärausgabe).
- Bohn, Robert B et al.: *NIST cloud computing reference architecture*, In: 2011 IEEE World Congress on Services, IEEE, ISBN 1457708795, S. 594–596.
- Bouvry, Pascal (2014): *Emerging paradigms and areas for expansion*, In: IEEE Cloud Computing, 1 (1), S. 58–61, ISSN 2325–6095.
- Butterfield, Reginald et al. (2016): *Towards Modelling a Cloud Application’s Life Cycle*,.
- Buxmann, Peter; Hess, Thomas & Lehmann, Sonja (2008): *Software as a Service*, In: Wirtschaftsinformatik, 50 (6), S. 500–503, ISSN 0937–6429.
- Chappell, David (2012): *How SaaS Changes an ISV’s Business, Sponsored by Microsoft Corporation*,.
- Chase, Jeff et al. (2014): *Thoughts on the State of Cloud over the Next Five Years*, In: IEEE Cloud Computing, 2 (1), S. 26–40, ISSN 2325–6095.
- Council, CSC (2016): *Migrating applications to public cloud services: roadmap for success*,.
- DaSilva, Carlos M et al. (2013): *Disruptive technologies: a business model perspective on cloud computing*, In: Technology Analysis and Strategic Management, 25 (10), S. 1161–1173, ISSN 0953–7325.
- Dufft, Nicole (2016): *Market Vision: Key Market Trends in Germany 2017*,.
- Hajjat, Mohammad et al. (2010): *Cloudward bound: planning for beneficial migration of enterprise applications to the cloud*, In: ACM SIGCOMM Computer Communication Review, 40 (4),

S. 243–254, ISSN 1450302017.

- Harms, Rolf & Yamartino, Michael (2010): *The economics of the cloud*, In: Microsoft whitepaper, Microsoft Corporation.
- Homburg, Christian (2017): *Marketingmanagement : Strategie - Instrumente - Umsetzung - Unternehmensführung*, 6. Auflage. Wiesbaden Imprint: Springer Gabler <http://dx.doi.org/10.1007/978-3-658-13656-7>, ISBN 3658136561 (Sekundärausgabe).
- Höllwarth, Tobias (2012): *Cloud Migration*, MITP-Verlags GmbH und Co. KG, ISBN 3826692241.
- Johnson, Bjorn & Qu, Yanzhen (2012): *A holistic model for making cloud migration decision: A consideration of security, architecture and business economics*, In: 2012 IEEE 10th International Symposium on Parallel and Distributed Processing with Applications,, S. 435–441, ISSN 1467316318.
- Kantabutra, Sooksan & Avery, Gayle C (2010): *The power of vision: statements that resonate*, In: Journal of Business Strategy, 31 (1), S. 37–45, ISSN 0275–6668.
- Khan, S. U. (2014): *Elements of Cloud Adoption*, In: IEEE Cloud Computing, 1 (1), S. 71–73, ISSN 2325–6095.
- Kitchenham, Barbara (2004): *Procedures for performing systematic reviews*, In: Keele, UK, Keele University, 33 (2004), S. 1–26.
- Marston, Sean et al. (2011): *Cloud computing—The business perspective*, In: Decision support systems, 51 (1), S. 176–189, ISSN 0167–9236.
- Martens, Benedikt & Teuteberg, Frank (2011): *Decision-making in cloud computing environments: A cost and risk based approach*, In: Information Systems Frontiers, 14 (4), S. 871–893, ISSN 1387–3326 1572–9419.
- Mell, Peter & Grance, Tim (2011): *The NIST definition of cloud computing*, In: National Institute of Standards and Technology.
- Olsen, Eric R: *Transitioning to software as a service: Realigning software engineering practices with the new business model*, In: Service Operations and Logistics, and Informatics, 2006. SOLI'06. IEEE International Conference on, IEEE, ISBN 1424403189, S. 266–271.
- Oredo, John Otieno & Njihia, James (2014): *Challenges of cloud computing in business: Towards new organizational competencies*, In: International Journal of Business and Social Science 5 (3), ISSN 2219–1933.
- Pahl, Claus & Xiong, Huanhuan (2013): *Migration to PaaS clouds-migration process and architectural concerns*, In: Maintenance and Evolution of Service-Oriented and Cloud-Based Systems (MESOCA), 2013 IEEE 7th International Symposium on the,, S. 86–91, ISSN 2326–6910.
- Pahl, Claus; Xiong, Huanhuan & Walshe, Ray (2013): *A Comparison of On-Premise to Cloud*

Migration Approaches, In: Service-Oriented and Cloud Computing: Second European Conference, ESOC 2013, Málaga, Spain, September 11-13, 2013. Proceedings, S. 212–226 http://dx.doi.org/10.1007/978-3-642-40651-5_18, ISSN 978–3–642–40651–5.

Pichler, Roman (2016): *10 tips for writing good user stories*, 2017 (17.02.2017) <http://www.romanpichler.com/blog/10-tips-writing-good-user-stories/>.

Qian, Ruoning & Palvia, Prashant (2013): *Towards an understanding of cloud computing's impact on organizational it strategy*, In: Journal of Information Technology Case and Application Research, 15 (4), S. 34–54, ISSN 1522–8053.

Rai, Rashmi; Sahoo, Gadadhar & Mehruz, Shabana (2012): *A five-phased approach for the cloud migration*, In: International Journal of Emerging Technology and Advanced Engineering, 2 (4), S. 286–291.

Rai, Rashmi; Sahoo, Gadadhar & Mehruz, Shabana (2015): *Exploring the factors influencing the cloud computing adoption: a systematic study on cloud migration*, In: SpringerPlus, 4 (1), S. 1, ISSN 2193–1801.

Reply (2016): *Reply Company Profile*, http://www.reply.com/InvestorsDocuments/en/Company_Profile_eng.pdf.

Reply, Syskoplan (2011): *iFMS Liegenschaftsmanager*,.

Repschläger, Jonas; Pannicke, Danny & Zarnekow, Rüdiger (2010): *Cloud Computing: Definitionen, Geschäftsmodelle und Entwicklungspotenziale*, In: HMD Praxis der Wirtschaftsinformatik, 47 (5), S. 6–15 <http://dx.doi.org/10.1007/BF03340507>, ISSN 2198–2775.

Sarkar, S. et al. (2012): *Cloud Based Next Generation Service and Key Challenges*, In: 2012 Third International Conference on Services in Emerging Markets, S. 20–29.

Scandurra, Patrizia et al. (2015): *Challenges and assessment in migrating IT legacy applications to the cloud*, In: 2015 IEEE 9th International Symposium on the Maintenance and Evolution of Service-Oriented and Cloud-Based Environments (MESOCA), S. 7–14, ISSN 2326–6910.

Schaffer, Henry E (2009): *X as a service, cloud computing, and the need for good judgment*, In: IT professional, 11 (5), S. 4–5, ISSN 1520–9202.

Schäfer, Johannes & Lichter, Horst (2016): *Changes in Requirements Engineering After Migrating to the Software as a Service Model*, In: Full-scale Software Engineering/Current Trends in Release Engineering, S. 25.

Sill, Alan (2014): *Factors in Development and Adoption of New Cloud Software and Standards*, In: IEEE Cloud Computing, 1 (4), S. 10–13, ISSN 2325–6095.

Statista (2016): *Umsatz mit Cloud Computing weltweit von 2009 bis 2016 (in Milliarden US-Dollar) erhoben durch Gartner*, In: statista, <https://de.statista.com/statistik/daten/studie/195760/umfrage/umsatz-mit-cloud-computing-weltweit-seit-2009/>.

-
-
- Tariq, Anum; Khan, Shoab Ahmed & Iftikhar, Sundas: *Requirements Engineering process for Software-as-a-Service (SaaS) cloud environment*, In: Emerging Technologies (ICET), 2014 International Conference on, IEEE, ISBN 1479960896, S. 13–18.
- Wan, Z. & Wang, P. (2014): *A Survey and Taxonomy of Cloud Migration*, In: 2014 International Conference on Service Sciences,, S. 175–180, ISSN 2165–3828.
- Weinman, Joe (2016): *Migrating to–or away from–the Public Cloud*, In: IEEE Cloud Computing, 3 (2), S. 6–10, ISSN 2325–6095.
- Williams, Byron J & Carver, Jeffrey C (2010): *Characterizing software architecture changes: A systematic review*, In: Information and Software Technology, 52 (1), S. 31–51, ISSN 0950–5849.
- Wright, Patrick M & McMahan, Gary C (1992): *Theoretical perspectives for strategic human resource management*, In: Journal of management, 18 (2), S. 295–320, ISSN 0149–2063.