




# PLATFORM-AS-A-SERVICE



```
<!DOCTYPE html>
<html xmlns="http://www.w3.org/1999/xhtml">
<head>
<title>Sample HTML Page</title>
<meta http-equiv="Content-type" content="text
<meta property="og:type" content="website" />
<meta property="og:url" content="http://www.s
<meta name="robots" content="index, follow" /
<meta name="author" content="http://www.somed
<link href="http://www.somedomain.com/" rel="
<link href="http://www.somedomain.com/" rel="
<script type="text/javascript" src="http://www
<script type="text/javascript" src="http://www
```

**Strategien, Technologien und Anbieter:**  
Ein Kompendium für IT-Entscheider

Autoren: René Büst, Dr. Carlo Velten

## Inhalt

---

<b>Vorwort</b>	<b>1</b>
<b>Danksagung</b>	<b>4</b>
<b>1. Softwareentwicklung im Cloud-Zeitalter</b>	<b>5</b>
1.1 Cloud Computing und Software-as-a-Service	5
1.2. Mobile Applikationen	7
1.3. API Economy	8
1.4. Internet der Dinge	11
1.5. Globale Skalierbarkeit	12
1.6. User Experience & Design Thinking	13
1.7. Agile Softwareentwicklung	14
1.8. Digitale Transformation	14
1.9. DevOps	14
<b>2. Grundlagen und Historie</b>	<b>15</b>
2.1. IaaS, PaaS und SaaS -Die Cloud Services-Typen	15
2.2. Platform-as-a-Service – Eine Definition	18
2.3. Die Entstehung von Platform-as-a-Service	20
2.4. Platform-as-a-Service der 1. Generation	21
2.5. Platform-as-a-Service der 2. Generation	22
2.6. Platform-as-a-Service der 3. Generation	23
<b>3. Entwicklungsvarianten und Modelle</b>	<b>25</b>
3.1. Spielarten von Platform-as-a-Service	25
3.2. Betriebskonzepte für Platform-as-a-Service	26
3.3. Entwicklungsumgebung (IDE) - lokal oder aus der Cloud	29
<b>4. Einsatzbereiche</b>	<b>30</b>
4.1. Voraussetzungen für den Einsatz von Platform-as-a-Service	30
4.2. Use Cases für den Einsatz von Platform-as-a-Service	31
4.3. Anwendungsintegration über Platform-as-a-Service	32

<b>5. Anforderungen</b>	<b>33</b>
5.1. Verfügbarkeit	33
5.2. Skalierbarkeit	33
5.3. Multiple Zugriffsmöglichkeiten	33
5.4. Ökosystem und Partner	33
5.5. Multiple Entwicklungsframeworks & -Sprachen	33
5.6. Multiple Betriebskonzepte	34
5.7. Professioneller Support und Service Level Agreements	34
5.8. Sicherheit	34
<b>6. Kriterien zur Anbietersauswahl</b>	<b>35</b>
6.1. Startups/Freelance Entwickler	35
6.2. Independent Software Vendor (ISVs)	37
6.3. Unternehmens-IT	38
<b>7. Technologie und Anbieter</b>	<b>40</b>
<b>8. Platform-as-a-Service im Praxiseinsatz</b>	<b>44</b>
8.1. Einsatzgrad und Planung	45
8.2. Gründe für und gegen Platform-as-a-Service	47
8.3. Favorisierte Betriebskonzepte für Platform-as-a-Service	49
8.4. Anforderungen an die Platform-as-a-Service-Provider	50
<b>9. Strategien für den erfolgreichen Einsatz</b>	<b>51</b>
9.1. Strategie	51
9.2. Kultur	52
9.3. Organisation	53
9.4. Prozess	54
<b>10. Ausblick</b>	<b>56</b>
<b>Über PIRONET NDH</b>	<b>60</b>
<b>Über BITMi e.V.</b>	<b>61</b>
<b>Overview Projekt PaaSport</b>	<b>62</b>
<b>Autoren</b>	<b>65</b>
<b>Über die Crisp Research AG</b>	<b>67</b>

# Vorwort

---

„Software is eating the world“ – Mit diesem Satz umschrieb der Netscape-Erfinder und berühmte Venture Capitalist Marc Andreessen in einem Beitrag für das Wallstreet Journal 2011 einen heutzutage nur allzu offensichtlichen Trend – Software wird die Grundlage für die Transformation nahezu aller Branchen, Geschäftsmodelle und Kundenbeziehungen. Software ist mehr als nur Steuerungskomponente von Hardware, sondern elementarer Wertschöpfungsbestandteil einer Vielzahl an Diensten und Produkten geworden. Von der Smartphone-App bis zum Bordcomputer in Autos. Vom Musikstreaming bis zur intelligenten Gebäudesteuerung. Vom Tracking individueller Trainingsdaten bis zur automatisierten Überwachung von Energienetzen.

60 Jahre nach Beginn der Computer-Revolution, 40 Jahre nach Erfindung des Mikroprozessors und 20 Jahre nach dem Beginn des modernen Internets haben sich die Voraussetzungen für die Entwicklung, den Betrieb und vor allem die Verbreitung von Software grundlegend verändert.

Doch nicht nur Visionäre aus dem Silicon Valley haben den strukturellen Umbruch erkannt. Auch europäische Industriekonzerne, wie beispielsweise Bosch, investieren große Teile ihrer Forschungs- und Entwicklungsbudgets mittlerweile in eine neue Software-Generation. So gab Bosch CEO Volkmar Denner die Losung aus, dass bis 2020 alle von Bosch ausgelieferten Geräte internetfähig sein werden. Denn im Internet der Dinge definiert sich der Nutzen von Produkten nur noch teilweise über deren Design und Hardware-Spezifikation. Große Teile des Nutzens – bzw. Mehrwertes – stiften die vernetzten Dienste rund um das Produkt. Und die sind software-basiert.

Die Auswirkungen dieser strukturellen Verschiebung zeichnen sich in der IT-Branche schon seit einigen Jahren deutlich ab. Große IT-Konzerne, wie IBM und HP, versuchen die Abhängigkeit vom Hardware-Business zu reduzieren und investieren intensiv im Software-Bereich.

Cloud-Pionier Salesforce zählt mit rund 4 Milliarden USD mittlerweile zu den Schwergewichten im Software-Geschäft. Startups, wie WhatsApp, erreicht mit seiner einfachen aber hochgradig nutzerfreundlichen mobilen App innerhalb von wenigen Jahren hunderte Millionen Nutzer weltweit.

Moderne Software kann heute disruptive Veränderungen hervorrufen. In Unternehmen und Märkten. In Politik und Gesellschaft. In Kultur wie im privaten Leben. Doch wie entstehen all

die neuen Applikationen? Wer schreibt all die Millionen „Lines of Code“?<sup>1</sup> Welches sind die neuen Plattformen, auf denen die abertausende Applikationen entwickelt und betrieben werden?

Mit diesen Fragen beschäftigt sich das vorliegende Kompendium. Speziell soll dargestellt werden, welche Rolle die sogenannten „Platform-as-a-Service“-Angebote für Entwickler und Unternehmen heute spielen. Denn, obwohl nach fast einer Dekade Cloud Computing die Begriffe IaaS und SaaS weitgehend bekannt sind und viele Unternehmen diese Cloud-Dienste nutzen, so haben bislang nur wenig Unternehmen Erfahrungen mit „Platform-as-a-Service“ gesammelt.

Zielsetzung ist es, IT-Entscheidern und Entwicklern einen Überblick zu den verschiedenen Spielarten und Einsatzmöglichkeiten von Platform-as-a-Service zu liefern. Denn diese sind vielfältig und reichen von der Kompositionierung von Businessprozessen (aPaaS) bis hin zur komplexen Integration verschiedener Cloud-Services (iPaaS). Zu diesem Zweck werden eingangs die Markttrends (Kapitel 1) sowie Entstehung der ersten PaaS-Plattformen geschildert und eine definitorische Grundlage für die weitere Diskussion geliefert (Kapitel 2). Danach folgt eine Beschreibung der verschiedenen Betriebsmodelle, in denen PaaS mittlerweile verfügbar ist (Kapitel 3). Relevante Einsatzszenarien bzw. Use Cases finden sich in Kapitel 4.

Bevor die verschiedenen Plattformen, Anbieter und Technologien in Kapitel 7 näher portraitiert werden, erfolgt in Kapitel 5 eine Analyse der Anforderungen an Platform-as-a-Service-Angebote sowie in Kapitel 6 die Darstellung relevanter Kriterien zur Anbietersauswahl nach Anwendergruppen. Denn die Anforderungen seitens Startups, etablierten Softwarehäusern und Großunternehmen variieren beträchtlich.

In Kapitel 8 wird der aktuelle Einsatzgrad und Planungsstand von PaaS in der Unternehmenspraxis anhand der Ergebnisse einer ersten empirischen Studie geschildert, die gemeinsam mit PIRONET NDH durchgeführt wurde. Nach dieser Bestandsaufnahme zum Thema folgt in Kapitel 9 einer Beschreibung der Strategien für den erfolgreichen Einsatz von Platform-as-a-Service, in dem auf die Veränderungen im Hinblick auf Strategie, Kultur, Organisation und Prozesse eingegangen wird.

Das Kompendium schließt mit einem Ausblick (Kapitel 10) aus Analysten-Perspektive, der einerseits die strategische Entwicklung ausgewählter Plattformen und Technologien sowie

---

<sup>1</sup> <http://www.informationisbeautiful.net/visualizations/million-lines-of-code/>

andererseits den Umgang seitens der CIOs und Entwickler mit dem Thema umfasst.

Mit dem Kompendium möchten die Autoren und Initiator PIRONET NDH einen kleinen Beitrag zum besseren Verständnis der modernen Cloud-basierten Software-Entwicklungsprozesse und -Plattformen liefern. Das Kompendium soll all diejenigen Unternehmer, Manager und IT-Experten unterstützen, die in den kommenden Jahren über die Entwicklung und den Einsatz neuer Applikationen und software-gestützter Businessprozesse entscheiden müssen. Aus Perspektive der Autoren wird Platform-as-a-Service zu einer der tragenden Säulen für die Umsetzung der digitalen Transformation in den Unternehmen. Denn die Mehrheit der neuen digitalen Anwendungen wird auf PaaS-Plattformen entwickelt und betrieben werden.

Wir wünschen Ihnen viel Spaß bei der Lektüre!

*Dr. Carlo Velten und René Büst*

# Danksagung

---

Die Hintergründe zur Entstehung dieses Kompendiums waren vielfältig. So konnten die Autoren über die letzten Jahre hinweg selbst eigene Erfahrungen im Rahmen einer Reihe von Software-Entwicklungsprojekten mit den neuen Technologien, Frameworks und PaaS-Plattformen sammeln. Zudem häufen sich die Anfragen aus den Reihen interessierter Anwenderunternehmen und Softwarehäuser.

Hinzu kommt, dass seit kurzem eine Reihe von PaaS-Technologiestacks verfügbar sind, die nun vollkommen neue Betriebs- und Hostingkonzepte für PaaS möglich machen. So kam auch die Zusammenarbeit mit PIRONET NDH zustande, die als einer der Pioniere Enterprise-PaaS-Plattformen für Unternehmenskunden in deutschen Rechenzentren betreiben. So hat PIRONET NDH als einer der ersten Managed Service Provider in Deutschland die PaaS-Technologiestacks von Microsoft (Azure Pack) und Red Hat (OpenShift) auf seiner Cloud-Infrastruktur implementiert. Dies ist vor dem Hintergrund der hohen technischen Komplexität und frühen Entwicklungsphase dieser Technologiestacks eine echte Innovationsleistung!

Durch intensive Fachgespräche mit Softwareherstellern, insbesondere aber auch mit dem Bundesverband IT-Mittelstand e.V. (BITMi), dem größten Interessenvertreter mittelständischer IT-Unternehmen in Deutschland, befasst sich das Kompendium ebenfalls mit den Fragestellungen und Anforderungen der unabhängigen Softwarehersteller. Der Leitfaden geht dabei insbesondere auf die Perspektive der mittelständischen ISVs ein.

Aus den gemeinsamen Diskussionen ging die Idee hervor, einen Überblick zu den verschiedenen Spielarten, Einsatzszenarien und Plattformen zu schaffen, um Entwicklern und IT-Entscheidern die Planung und Auswahl geeigneter PaaS-Plattformen zu erleichtern.

So ist dieses erste, deutschsprachige Kompendium zum Thema Platform-as-a-Service entstanden. Wir danken PIRONET NDH an dieser Stelle ausdrücklich für die Unterstützung bei der Realisierung des Projektes.

# 1. Softwareentwicklung im Cloud-Zeitalter

---

Die Zeiten, in den Software auf lokalen Maschinen programmiert und isoliert in abgeschirmten Firmennetzwerken betrieben wurde, sind längst vorbei. Im Cloud-Zeitalter steigt der Wert eines Software-Dienstes mit dessen Vernetzungs- und Verbreitungsgrad. Dies hat fundamentale Auswirkungen auf die Art und Weise wie Anwendungen heute entwickelt, getestet und betrieben werden. In diesem Kapitel sollen daher einleitend die wichtigsten Einflüsse und Technologietrends beschrieben werden, die die Softwareentwicklung in den kommenden Jahren prägen werden.

## 1.1 Cloud Computing und Software-as-a-Service

---

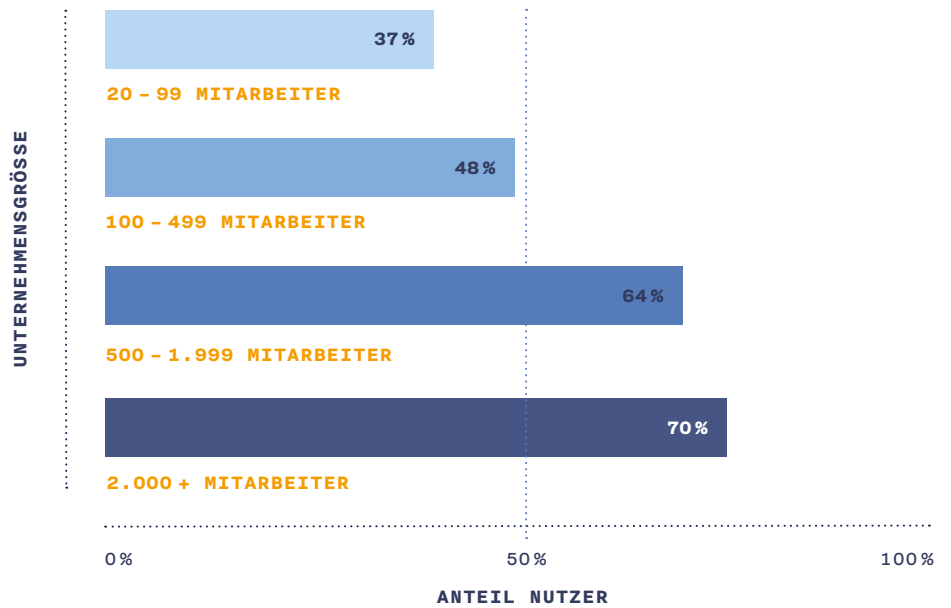
Was vor rund 10 Jahren als hipper Technologietrend begonnen hat, ist längst betrieblicher Alltag geworden. Unternehmen betreiben eine Vielzahl ihrer Applikationen nicht mehr in eigenen Rechenzentren bzw. auf eigenen IT-Infrastrukturen (“on-premise”), sondern nutzen dazu Cloud-Infrastrukturen von externen Dienstleistern (“IaaS”). Alternativ bauen Unternehmen eigene “Private Clouds” auf, um ihre Anwendungen auf einer flexiblen, virtualisierten und standardisierten Infrastruktur zu betreiben. Um das Potenzial von IaaS voll auszunutzen, müssen die Anwendungen allerdings multimandantenfähig und in modernen Programmiersprachen entwickelt worden sein. Für Legacy-Anwendungen macht der Cloud-Betrieb nur wenig Sinn.

Auch nutzen immer mehr Unternehmen standardisierte Anwendungen im “Software-as-a-Service“-Modell. Wie Studien und Prognosen belegen, zieht die Nachfrage nach SaaS über alle Firmengrenzen hinweg weiter an. Salesforce ist als einer der Pioniere mit rund 4 Milliarden USD Umsatz mittlerweile unter den 10 größten Softwarehäusern angekommen. Immer mehr Startups und ISVs (Independent Software Vendors) entwickeln und vertreiben die neue Generation ihrer Softwarelösungen, mittlerweile als “Software-as-a-Service”, im monatlichen Mietmodell. Nach Einschätzungen von Crisp Research entwickeln derzeit allein in Deutschland über 600 professionelle ISVs neue SaaS-Lösungen.

Aber auch große Anwenderunternehmen stellen auf Software-as-a-Service um und bauen interne App-Stores und -Marktplätze auf, um Software und Cloud-Dienste nach dem neuen Paradigma bereitzustellen. Das hat nicht nur Vorteile für die Endanwender, die sich über schnelle Bereitstellung (Self-Service)



### Einsatz von Cloud Computing in Unternehmen steigt Cloud Computing Nutzer nach Unternehmensgröße



QUELLE: BITKOM - KPMG, 2014

und moderne Lösungen mit guter Nutzbarkeit freuen. So muss die interne IT-Organisation keine lokalen Installationen mehr supporten, Releases können automatisiert im Hintergrund ablaufen und das Nutzerverhalten lässt sich mittels Analytics besser verstehen. Dies sind alle Gründe für einen Ausbau von Software-as-a-Service.

## 1.2. Mobile Applikationen

---

Mobility First! So heißt mittlerweile das Mantra vieler Softwarefirmen. Egal, ob groß (Microsoft, SAP et al.) oder klein, es besteht die absolute Notwendigkeit die eigenen Softwarelösungen für die Nutzung auf den mobilen Geräten anzupassen – allen voran auf die neue Generation von Tablets und Smartphones.

Hierzu ist es aber vielfach notwendig, die Gestaltung und Entwicklung komplett zu überdenken. Denn die Nutzererfahrung auf mobilen Endgeräten wird von einer Vielzahl an Faktoren beeinflusst, die auf dem Desktop im heimischen Unternehmensnetzwerk nur eine untergeordnete Rolle spielen, wie beispielsweise Benutzerfreundlichkeit, verfügbare Bandbreite oder Sicherheit in öffentlichen Funknetzen. Um mobile Nutzer wirklich zufriedenzustellen, müssen sich die mobilen Anwendungen der Situation des Anwenders (Kontext) und dem jeweiligen Gerät anpassen. Neben Grundüberlegungen zum “Responsive Web Design” spielen für Entwickler vor allem Entwicklungs-Frameworks, wie z.B. jQuery Mobile sowie PaaS-Plattformen, eine wesentliche Rolle. Denn sie vereinfachen den Entwicklungs-, Test- und Distributionsprozess über mehrere OS- und Geräteplattformen hinweg maßgeblich. Die Kombination von Entwicklungs-Framework und PaaS-Plattform ermöglicht es den Entwicklern sich vollständig auf die Anwendung zu fokussieren, während die Bereitstellung der notwendigen IT-Infrastruktur zur Verwaltung der Nutzer- und Applikationsdaten automatisiert erfolgt. Dass diese Kombination ein Erfolgsmodell ist, beweist der hohe Anteil an Entwicklern mobiler Applikationen, die schon heute auf Platform-as-a-Service setze. Laut einer aktuellen Studie von Crisp Research bereits über 45 Prozent.<sup>2</sup>

Aber auch in den Großunternehmen spielt die mobile Bereitstellung von Anwendungen seit einigen Jahren eine immer wichtigere Rolle. Unter dem Schlagwort “Enterprise Mobility” haben die Unternehmen sukzessiv begonnen ihre Anwender für den mobilen Arbeitsstil mit Tablets und Smartphones auszurüsten. Nun folgt die Anpassung der relevanten Business-Applikationen für den mobilen Einsatz.

Nach aktuellen empirischen Studien, arbeiten derzeit schon mehr als die Hälfte der Beschäftigten mobil.<sup>3</sup>

---

<sup>2</sup> <http://www.business-cloud.de/paas-trendstudie-cloud-verursacht-erdbeben-im-softwaremarkt/>

<sup>3</sup> <http://idc.de/de/ueber-idc/press-center/56517-idc-studie-deutsche-unternehmen-setzen-auf-mobile-apps-zur-verbesserung-ihrer-geschäftsprozesse>

Aus diesem Grund prüfen immer mehr mittelständische und große Unternehmen, welche PaaS-Plattformen und welche mobilen Entwicklungs-Frameworks für ihren Einsatzzweck am geeignetsten erscheinen. Denn die Anzahl der bereitgestellten mobilen Apps soll sich bei deutschen Unternehmen in den kommenden 12 Monaten von 10 auf 17 nahezu verdoppeln.

### 1.3. API Economy

---

Softwareentwicklung im Cloud-Zeitalter heißt auch, dass Anwendungen nicht nur funktional fehlerfrei sein und über eine gute Nutzbarkeit verfügen müssen (vgl. Abschnitt 3.6). Vielmehr nutzen moderne Applikationen heute meist eine Vielzahl anderer Datenquellen, die über sogenannte APIs (Application Programming Interface) angesprochen werden. Hinsichtlich der Anzahl und Vielfalt der APIs hat in den letzten Jahren ein explosives Wachstum stattgefunden. So stieg die Anzahl von rund 2.000 APIs in 2010 auf rund 11.500 in 2014.<sup>4</sup> Und wird wohl auch in den kommenden Jahren weiter wachsen.

Offene, gut dokumentierte Schnittstellen sind in der Lage ein Ökosystem zu schaffen, das nicht nur die Innovationskraft der eigenen Technologieplattform stärkt, indem neue Anwendungen die eigenen Daten und Funktionen nutzen. APIs können auch substantziell zur Verbreitung eines Dienstes beitragen und dabei neue Umsatzquellen erschließen. Daher verfolgen fast alle erfolgreichen Cloud Companies eine offene API-Strategie, um externe Entwickler als Partner an sich zu binden. Allen voran Amazon, Google, Facebook, Twitter und Salesforce. Das Beispiel Salesforce demonstriert, dass nicht nur im Consumer- sondern auch im Enterprise-IT-Umfeld eine solche Strategie erfolgreich sein kann. So haben bisher rund 800.000 Entwickler über 2,5 Millionen Anwendungen auf Basis der Force.com-API geschrieben.

Daraus resultiert, dass derzeit über 60% der Traffics auf der Salesforce-Plattform sowie ein Großteil des Umsatzes über die API generiert wird.

Für Unternehmen und Softwareentwickler ist es daher entscheidend, sich mit den APIs der großen Cloud-Plattformanbieter auszukennen, um diese bei Bedarf anzapfen bzw. einbinden zu können. Um dies in der Praxis zu bewerkstelligen, sind neben dem erforderlichen Know-How auch entsprechende API Developer und Management Tools wie z.B. Runscope<sup>5</sup> oder APIScience<sup>6</sup> notwendig. Hinzu kommt, dass Unternehmen wie

---

4 <http://www.programmableweb.com/api-research>

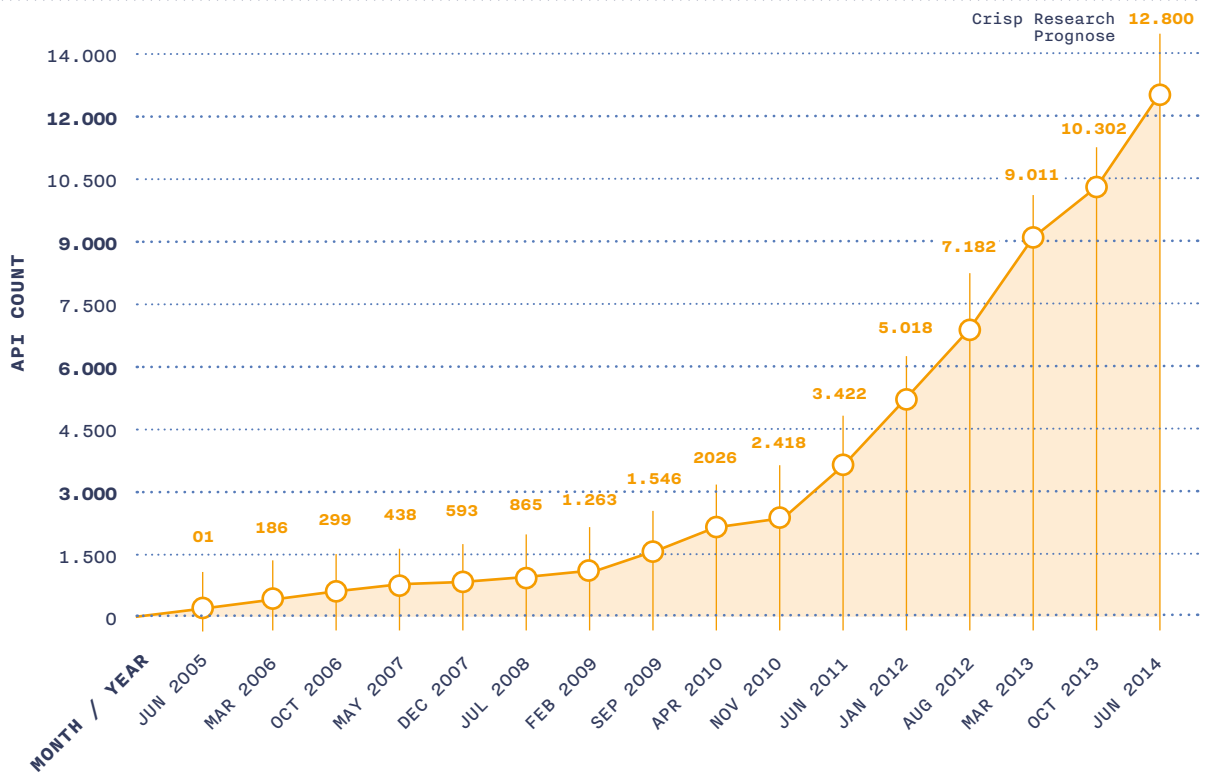
5 <https://www.runscope.com/>

6 <https://www.apiscience.com/>

Entwickler eine eigene API-Strategie definieren sollten, die den Umgang mit den eigenen Daten sowie den Daten Dritter genau beschreibt. Zwar können mittels APIs unternehmenseigene Daten erfolgreich kommerzialisiert (Geodaten, Nutzerdaten, Bewegungsdaten, Maschinendaten etc.) und Daten Dritter einfach genutzt werden, aber eine rechtlich saubere Definition sollte hier den Grundstein legen.

Absehbar ist, dass die Komplexität und Vielfalt bei der Konzeption und Entwicklung neuer Anwendungen und mobilen Apps mit den neuen APIs weiter dramatisch ansteigen wird, was den Bedarf nach Automation und Standardisierung im Rahmen der Entwicklungs- und Testprozesse weiter verschärft. PaaS-Plattformen werden hier wohl eines der Mittel zur weiteren Professionalisierung der Entwicklungsprozesse sein.

Growth in Web APIs from 2005 to 2014



QUELLE: programmableweb, Crisp Research AG, 2014

## 1.4. Internet der Dinge

Auch das langsam aufkommende "Internet der Dinge" bzw. "Industrial Internet" nimmt repräsentativen Einfluss auf die Softwareentwicklung. So bildet sich in den kommenden Jahren eine enorme Nachfrage nach zusätzlichen Developer-Ressourcen, um die Applikations- und Integrationslogik für die Vielzahl an neuen Geräten zu entwickeln und zu implementieren. Hinzu kommt ein großer Bedarf zur Entwicklung von Kontroll-, Monitoring- und Sicherheitsfunktionen für die intelligenten Geräte und Sensoren. Laut den Prognosen verschiedener Analysten- und Beratungsunternehmen soll die Anzahl der vernetzten Geräte bis 2020 zwischen 25-50 Milliarden liegen<sup>7</sup> - Autos und Telefone eingerechnet. Crisp Research schätzt das globale Marktpotenzial für die neuen Dienste im "Internet der Dinge" bzw. "Industrial Internet" auf rund 160 Milliarden Euro bis zum Ende 2016.

### Die Welt der Daten

Marktvolumen Datability weltweit (in Mio. EUR)

	2011	2012	2013	2014	2015	2016	CAGR
Sensors & Networking (Internet of Things)	990	1.330	1.830	2.560	3.635	5.020	38,4%
IT-Infrastruktur, Software & Services (Big Data)	3.158	4.055	5.462	7.554	10.205	13.542	33,8%
Analytic Services & Data Products (Data Economy)	19.538	27.742	40.924	63.378	95.349	142.046	48,7%
Gesamtvolumen (Datability)	<b>23.686</b>	<b>33.127</b>	<b>48.216</b>	<b>73.492</b>	<b>110.190</b>	<b>160.609</b>	<b>46,6%</b>

 **QUELLE:** Crisp Research AG  
 im Auftrag von PIRONET  
 NDH, 2014

Derzeit fehlen vor allem noch allgemein anerkannte Standards und Plattformen, um die unterschiedlichen Gerätetypen, Sensoren und deren Backend-Systeme miteinander verbinden und steuern zu können. Diese werden sich aber in den kommenden Jahren sukzessiv entwickeln. Erste Schritte in diese Richtung sind das Industrial Internet Consortium (ICC)<sup>8</sup>, das von Industrie- und IT-Unternehmen wie AT&T, Cisco, General Electric, IBM und Intel gegründet wurde.

Zudem bauen einzelne IT-Unternehmen wie Salesforce (Salesforce1) oder BlackBerry (Project Ion<sup>9</sup>) eigene Plattformen zum Management von vernetzten Endgeräten und Sensoren

<sup>7</sup> Gartner 2013 <http://www.gartner.com/newsroom/id/2636073>

<sup>8</sup> <http://www.iiconsortium.org/>

<sup>9</sup> <http://el.blackberry.com/projection>

auf. Auch entstehen neue Allianzen wie beispielsweise zwischen IBM und Bosch oder T-Systems und Siemens. Aber nicht die reine Anzahl an neuen Geräten und Sensoren ist aus Entwicklungsperspektive kritisch. Als eine zentrale Herausforderung werden sich Komplexitätsmanagement und Maintenance von industriellen Software-Lösungen sowie die Integration der Systeme untereinander darstellen. Schon heute weisen die in Oberklassewagen verbauten Softwaresysteme eine Code-Basis von bis zu 100 Millionen „Lines of Code“<sup>10</sup> auf. In einer Welt, in der das Auto konstant mit seiner Außenwelt (Ampeln, Straßensensoren, Objekte im Umfeld, andere Autos) und seinen Insassen in Interaktion steht, werden sich diese Code-Basis und die erforderlichen Schnittstellen weiter erhöhen.

Das Internet der Dinge, bzw. das „Industrial Internet“, wird daher ebenfalls den Einsatz von Platform-as-a-Service beschleunigen, da Unternehmen und Entwickler in diesem Kontext gezwungen sein werden in höchstem Maße standardisiert und automatisiert zu arbeiten.

## 1.5. Globale Skalierbarkeit

---

Einen weiteren wesentlichen Treiber für Platform-as-a-Service stellt die Anforderung nach globaler Skalierungsfähigkeit für neue Applikationen dar. Egal ob Startup, exportorientierter Mittelständler oder Konzern - um die Investitionen in eine neue Applikation amortisieren zu können, stellt sich meist von Beginn an die Frage nach der Möglichkeit zum globalen Roll-Out. Das gilt für den Launch eines neuen Online-Games ebenso wie für die Bereitstellung einer unternehmensweiten SaaS-Anwendung für den Vertrieb. Unternehmen wollen in der Lage sein, erfolgreiche Anwendungen und Prozesse möglichst schnell global auszurollen. Entweder um Märkte schnell zu erobern oder interne Synergien zu realisieren. Und hier bieten die PaaS-Plattformen der großen Cloud-Provider die notwendige globale Infrastruktur.

Werden Anwendungen von Beginn an multimandantenfähig und auf Basis moderner Programmiersprachen entwickelt, steht einer automatisierten und weltweiten Skalierung über PaaS-Plattformen meistens nichts im Wege.

---

<sup>10</sup> <http://www.informationisbeautiful.net/visualizations/million-lines-of-code/>

---

## 1.6. User Experience & Design Thinking

---

Neben den tendenziell technologisch geprägten Trends (APIs, Skalierungsfähigkeit, Internet der Dinge), spielt auch der Anwender mit seinen Erwartungen an neue Applikationen eine ganz zentrale Rolle. Da im Cloud-Zeitalter die alternative App bzw. Anwendung meist nur einen Klick entfernt ist, wird die User Experience in vielen Anwendungskategorien zum Erfolgsfaktor.

Mittlerweile spielt das Thema “Design Thinking” und “User Experience Management” nicht nur bei der Gestaltung von Anwendungen für Privatanwender eine Rolle, sondern hat sich auch einen festen Platz in der Produktentwicklung von Enterprise-Software erkämpft. Auch hier gilt, dass Unternehmensanwender neue Lösungen nur dann akzeptieren und effektiv nutzen, wenn deren Design, Funktionalität und Performance ihren Erwartungen entspricht.

Während bis zur Jahrtausendwende noch ein Wettrennen um Funktionen und Features geführt wurde, sind derzeit diejenigen Software-Entwickler und Softwareunternehmen erfolgreich, die in der Lage sind, die Komplexität ihrer Lösungen hinter möglichst einfachen und intuitiven Bedienoberflächen und Menüs zu verbergen. Einfachheit in der Nutzerführung wird gepaart mit hochkomplexen Backend-Prozessen und dem Fein-Tuning von Middleware und IT-Infrastruktur. Damit möchte man eine möglichst hohe Performance der Cloud-Dienste und Apps auf der Nutzer-Seite (Browser, mobile App) erreichen. Denn schon leichte Verzögerungen im Aufbau von Webseiten oder geringe Latenzen bei der Übertragung von Video oder Online-Spielen können zu signifikanten Abbruchraten, geringeren Umsätzen und einer gesunkenen Nutzerzufriedenheit führen.

Um einen Prozess zur kontinuierlichen Optimierung der User Experience zu implementieren, sind verschiedene Voraussetzungen zu erfüllen. PaaS-Plattformen erleichtern Entwicklern das Testing und Release-Management neuer Funktionen und Designs ohne großen manuellen Aufwand. So lassen sich durch vorkonfigurierte Plug-Ins oder Templates auf PaaS-Plattformen schnell neue Funktionen implementieren, wie z.B. Site-Analytics für Wordpress.

Auch die Durchführung von A/B-Tests ist ohne größeren Aufwand möglich, indem man Instanzen seiner Lösung klonet und mit veränderten Feature-Set oder Design einer Testgruppe zur Verfügung stellt.



## 1.7. Agile Softwareentwicklung

---

Der Einsatz von PaaS-Plattformen wird auch durch neue Trends bei den Softwareentwicklungsmethoden beeinflusst. Bedingt durch die steigende Komplexität und technologische Vielfalt einerseits sowie immer kürzere Innovations- und Release-Zyklen andererseits, lassen sich traditionelle “Wasserfall-Methoden” nur noch bedingt einsetzen. Agile Entwicklungsmethoden, wie beispielsweise SCRUM, gewinnen dagegen immer mehr an Gewicht. Ein iteratives Vorgehen mit häufigen Rückkopplungsprozessen ermöglicht es, neue Anforderungen während des Entwicklungsprozesses dynamisch einzubinden sowie Fehler frühzeitig zu korrigieren. Gerade bei der Entwicklung von Webanwendungen die mit APIs anderer Plattformen arbeiten, besteht die Notwendigkeit sehr flexibel zu programmieren.

## 1.8. Digitale Transformation

---

“Digital Transformation” - Unter diesem Schlagwort verstehen Unternehmen, dass nahezu jede Branche derzeit den Wandel ihrer Geschäftsprozesse und Geschäftsmodelle hin zu einer vernetzten Organisation, die ihre Wertschöpfungsabläufe und Kundenbeziehungen vollständig digital abbildet.

Um die Vision der digitalen Fabrik bzw. einer guten “Digital Customer Experience” Wirklichkeit werden zu lassen, ist neben fundiertem Branchen- und Prozess-Know-how, vor allem eine hohe Kompetenz in der Entwicklung moderner Software-Architekturen sowie der Gestaltung nutzerfreundlicher Applikationen und Cloud-Dienste gefragt.

## 1.9. DevOps

---

Das Thema DevOps zählt derzeit zu den wichtigsten IT-Trendthemen. Der Begriff leitet sich von dem Präfix “Dev” von Softwareentwickler (Developer) sowie dem Suffix “Ops” von IT-Betrieb (Operations) ab. Der zusammengesetzte Begriff DevOps symbolisiert die enge Zusammenarbeit beider Unternehmensbereiche, der Softwareentwicklung und dem IT-Betrieb. Wie auch das Cloud Computing hat sich DevOps im Laufe der Zeit entwickelt und hilft Unternehmen als Organisations- und Prozessmodell, die Software- und Produktentwicklung zu beschleunigen und zu optimieren. So lässt sich die Fehlerrate in der Software-Entwicklung deutlich reduzieren sowie die Takt-rate bei Innovationen und neuen Releases mittels DevOps-Ansätzen deutlich erhöhen. Da ein schnelleres Time-to-Market in der digitalen Welt ein entscheidender Erfolgsfaktor ist, macht sich DevOps derzeit in immer mehr Unternehmen breit.

## 2. Grundlagen und Historie

---

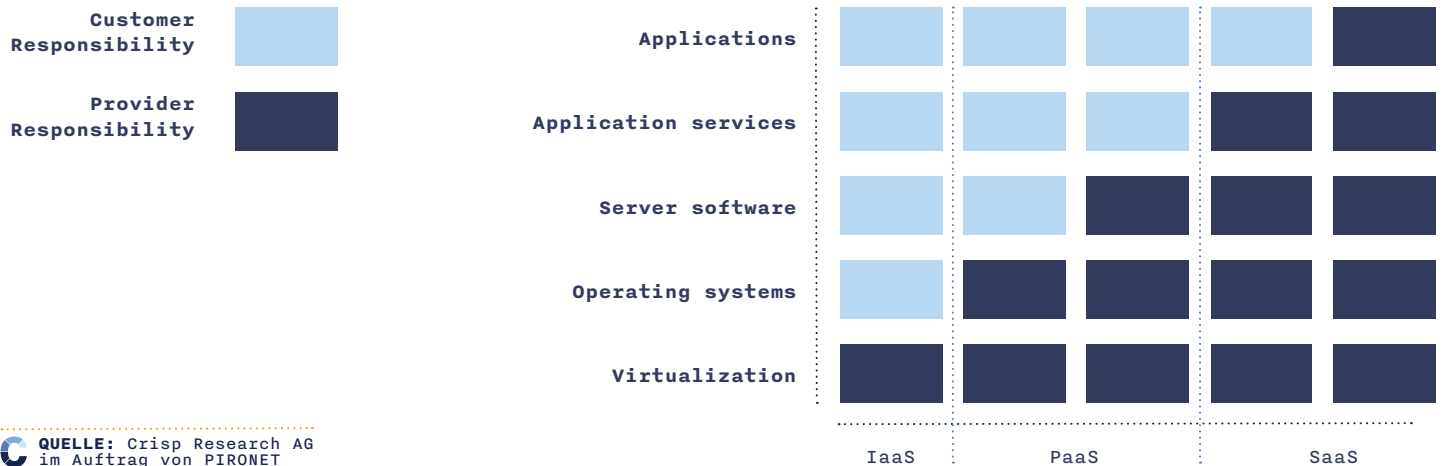
### 2.1. IaaS, PaaS und SaaS -Die Cloud Services-Typen

---

Cloud Computing hat sich innerhalb der letzten 8 Jahre von einem Hype zu einer echten Alternative für den flexiblen Bezug von IT-Ressourcen entwickelt, die zentral und weltweit verteilt aus den Rechenzentren der Cloud Anbieter bereitgestellt werden. Insgesamt gaben Unternehmen in Deutschland im Jahr 2014 rund 7 Milliarden Euro für Cloud-Dienste, -Technologien und Integrationsleistungen aus.

Bei den Cloud-Diensten lassen sich die Kategorien IaaS, PaaS und SaaS unterscheiden. Die Differenzierung lässt sich anhand der Verteilung der Aufgaben zwischen Anwender und Anbieter verdeutlichen. Während beim Infrastructure-as-a-Service vom Anbieter lediglich die Rechenleistung bzw. der Speicherplatz bereitgestellt wird, so übernimmt der Anbieter beim PaaS-Modell auch die Verantwortung für Betriebssystem, Laufzeitumgebung, Middleware und Security. In beiden Fällen kann der Anwender bzw. Entwickler noch Eingriffe vornehmen und Teile des Cloud-Stacks nach seinen Anforderungen konfigurieren und verwalten. Dies ist beim Software-as-a-Service nicht möglich. Hier wird die komplette Anwendung als fertiger Cloud-Dienst bereitgestellt und komplett durch den Anbieter betrieben und kontrolliert.

CLOUD-STACK METHODOLOGY



QUELLE: Crisp Research AG  
 im Auftrag von PIRONET  
 NDH, 2014

**Infrastructure-as-a-Service (IaaS)**

Infrastructure-as-a-Service (IaaS) stellt Rechenleistung, Speicherplatz und weitere Services bereit, mit denen sich ein eigenes virtuelles Rechenzentrum aufbauen lässt. Der Anbieter ist lediglich für die Bereitstellung und Wartung der Hardware zuständig und sorgt dafür, dass eine virtuelle Infrastruktur aufgebaut werden kann. Für alle anderen benötigten Ressourcen, wie das Betriebssystem oder Anwendungen, ist der Kunde zuständig, werden aber ebenfalls von dem Anbieter zur Verfügung gestellt.

**Software-as-a-Service (SaaS)**


Software-as-a-Service (SaaS) ermöglicht den plattformunabhängigen Zugriff auf Anwendungen über einen Standard-Webbrowser. Mögliche SaaS Applikationen sind Office und Collaboration Suites, Customer Relationship Management Systeme (CRM) oder anderweitige Business-Applikationen. Der Anbieter hat dafür zu sorgen, dass sich die Software ständig auf einem aktuellen Stand befindet. Darüber hinaus stellt er die Infrastruktur für das Hosting der Software bereit oder nutzt wiederum einen IaaS- oder PaaS-Anbieter.

## Cloud Service Typen

---

Typ	Beschreibung	Einsatz
<b>SaaS</b>	<ul style="list-style-type: none"><li>• Multimandantenfähige Applikationen</li><li>• Betriebssystem- und Browser-unabhängige Nutzung</li><li>• Schnelle Bereitstellung</li><li>• Keine Vorabinvestitionen (Capex)</li></ul>	→ Alternative zu on-Premise betriebenen oder gehosteten Lösungen.
<b>PaaS</b>	<ul style="list-style-type: none"><li>• Entwicklungs- und Laufzeitumgebungen</li><li>• Applikationsentwicklung</li><li>• Verwaltung des Applikationslebenszyklus</li><li>• Keine Vorabinvestitionen (CAPEX)</li></ul>	→ Plattformen für die Entwicklung neuer, webbasierter und skalierbarer Applikationen.
<b>IaaS</b>	<ul style="list-style-type: none"><li>• Virtuelle Infrastrukturen</li><li>• Server, Speicherplatz, Netzwerk</li><li>• Infrastruktur-Dienste</li><li>• Keine Vorabinvestitionen (CAPEX)</li></ul>	→ Für den Betrieb von bestehenden Standard-Applikationen und virtualisierten Serverlandschaften.

---

 **QUELLE:** Crisp Research AG  
im Auftrag von PIRONET  
NDH, 2014

Während in den letzten Jahren die IaaS-Plattformen im Rampenlicht standen und ein Großteil der Aufmerksamkeit seitens CIOs und Presse erhielten, werden nun auch Platform-as-a-Services (PaaS) immer wichtiger.

Denn die schnelle und flexible Entwicklung neuer Anwendungen wird derzeit zum Benchmark für viele IT-Verantwortliche. Im Kontext großangelegter Digital Transformation-Projekte sowie der langfristigen Umsetzung von Enterprise Mobility-Strategien müssen immer mehr Anwendungen mobil zur Verfügung stehen. Die Frequenz, die mit den neuen Funktionen entwickelt, getestet und ausgerollt werden, steigt, so dass PaaS-Plattformen für Softwarehäuser als auch IT-Organisationen zum Mittel der Wahl werden.

## 2.2. Platform-as-a-Service – Eine Definition

Mit Platform-as-a-Service (PaaS) werden Cloud-Dienste bzw. Cloud-Plattformen beschrieben, die den Anwendern komplette Entwicklungs- und Betriebsumgebungen zur Verfügung stellen. Diese werden meist in Form einer voll gemanagten Laufzeit- und Entwicklungsumgebung bereitgestellt. Im Gegensatz zu IaaS, lässt sich auf PaaS-Plattformen der gesamte Lebenszyklus einer Anwendung vollständig abbilden und verwalten. Über APIs wird die Plattform des jeweiligen Anbieters angesprochen. Für die Nutzung ist lediglich ein lokaler Desktop, Web-Browser, evtl. eine lokale Entwicklungsumgebung sowie eine Internetverbindung notwendig. Die restliche Entwicklungsinfrastruktur, wie Webserver, Betriebssysteme, Laufzeitumgebungen, Frameworks, Middleware und der Betrieb der zugrundeliegenden Infrastruktur befinden sich in der Verantwortung des Anbieters und werden von diesem bereitgestellt und verwaltet.

Anwender können sich dadurch auf die eigentliche Entwicklung von Anwendungen konzentrieren und müssen sich nicht mehr um Betrieb, Skalierung und Pflege der Infrastruktur- und Middleware kümmern.

PaaS erlaubt es Unternehmen und Softwareentwicklern somit den Fokus auf ihre Kernkompetenzen zu legen und IT-Betriebsaufgaben auszulagern und zu automatisieren. Im Vergleich zu herkömmlichen On-Premise Entwicklungsinfrastrukturen differenzieren sich PaaS-Umgebungen insbesondere durch die Möglichkeit zur verteilten Entwicklung von Quellcode inklusive einer dafür vorgesehenen Versionierung und der Entwicklung des Quellcodes direkt auf der PaaS-Umgebung. Weiterhin bieten viele PaaS integrierte Funktionen für das Monitoring und die Analyse der Applikation und ihres aktuellen Zustands.

### Application Lifecycle

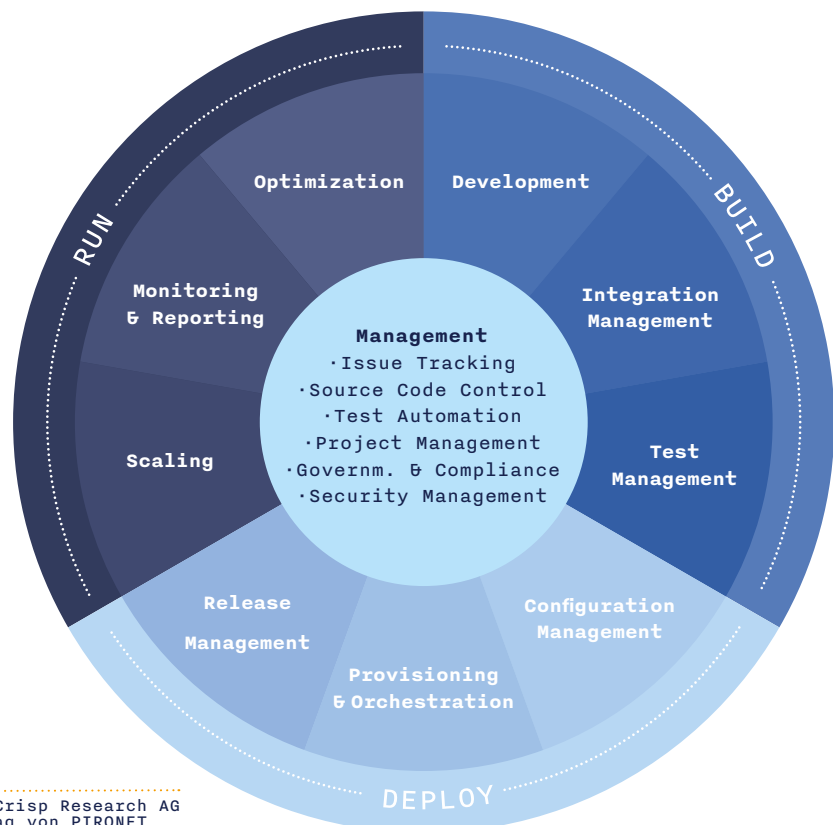
Build	Deploy	Run	Management
<ul style="list-style-type: none"><li>• Development</li></ul>	<ul style="list-style-type: none"><li>• Configuration</li></ul>	<ul style="list-style-type: none"><li>• Scaling</li></ul>	<ul style="list-style-type: none"><li>• Issue Tracking</li></ul>
<ul style="list-style-type: none"><li>• Integration</li></ul>	<ul style="list-style-type: none"><li>• Management</li></ul>	<ul style="list-style-type: none"><li>• Monitoring &amp; Reporting</li></ul>	<ul style="list-style-type: none"><li>• Source Code Control</li></ul>
<ul style="list-style-type: none"><li>• Management</li></ul>	<ul style="list-style-type: none"><li>• Provisioning &amp; Orchestration</li></ul>	<ul style="list-style-type: none"><li>• Optimization</li></ul>	<ul style="list-style-type: none"><li>• Test Automation</li></ul>
<ul style="list-style-type: none"><li>• Test Management</li></ul>	<ul style="list-style-type: none"><li>• Release Management</li></ul>		<ul style="list-style-type: none"><li>• Project &amp; Team Management</li></ul>
			<ul style="list-style-type: none"><li>• Governm. &amp; Compliance</li></ul>
			<ul style="list-style-type: none"><li>• Security Management</li></ul>

Eine PaaS-Umgebung setzt typischerweise auf einer skalierbaren Infrastruktur (z.B. Infrastructure-as-a-Service, IaaS) auf und ist somit in der Lage Rechenleistung und Speicherplatz, bei Bedarf und je nach Auslastung, zu skalieren (Auto-Scaling). Auf Basis eines PaaS lassen sich multimandantenfähige Software-as-a-Service (SaaS) Applikationen entwickeln und skalierbar bereitstellen.

Die strategischen Ziele, die Anwender und Entwickler mit dem Einsatz von PaaS verbinden, lassen sich wie folgt illustrieren:

- Time-to-Market - Beschleunigung des Entwicklungsprozesses
- Integration von Entwicklung und Betrieb (DevOps)
- Mehr Innovationen (höhere Taktrate bei Releases)
- Standardisierung der Entwicklungs- und Betriebsprozesse (weniger Fehler, Vereinfachung)
- Nutzung moderner Tools, Entwicklungsframeworks & APIs
- Bessere Kontrolle und mehr Transparenz im Entwicklungs- und Betriebsprozess

#### PLATFORM AS A SERVICE APPLICATION LIFECYCLE



QUELLE: Crisp Research AG  
im Auftrag von PIRONET  
NDH, 2014

## 2.3. Die Entstehung von Platform-as-a-Service

---

Erstmals wurden PaaS-Dienste von den damaligen Startups EngineYard (2006) und Heroku (2007) angeboten – ungefähr zeitgleich mit dem Release der Amazon Web Services. Die Gründer von EngineYard und Heroku wollten damit vornehmlich die Entwicklung und den Betrieb von RubyOnRails-Applikationen vereinfachen.<sup>11</sup> Ihre PaaS-Dienste richteten sich hauptsächlich an befreundete Entwickler, die auf die Entwicklung von Web- und Mobile-Applikationen fokussiert waren. Diese sollten sich auf die Entwicklung ihrer Anwendungen konzentrieren können, ohne sich mit dem Aufbau und der Administration der dafür notwendigen Infrastrukturen auseinanderzusetzen. Neben RubyonRails wurden zunächst keine weiteren Programmiersprachen unterstützt, was sich im Laufe der Zeit jedoch deutlich verbessert hat. Mit dem Erscheinen der Google App Engine (2008) folgten mit Python und Java weitere Sprachen in die PaaS-Welt. Der SaaS-CRM Anbieter Salesforce setzt für seine Plattform Force.com (2008) auf die proprietäre Sprache APEX. Mit Windows Azure (2010) führte Microsoft seine bekannte Programmiersprache ASP.NET in die PaaS-Welt ein.

### PaaS Anbieter

---

Anbieter	Erscheinungsjahr
EngineYard	2006
Heroku (heute Teil von Salesforce.com)	2007
Force.com (heute Salesforce1)	2008
Google App Engine	2008
cloudControl	2009
Appoleon	2009
Microsoft Azure	2010
Jelastic	2010
CloudBees	2010
Red Hat Openshift	2011
Cloud Foundry	2011
Amazon Elastic Beanstalk	2011
GBS AG	2011

Mit dem Launch einer Vielzahl weiterer PaaS-Plattformen sowie dem Einzug der in der Enterprise-IT relevanten Programmiersprachen hat sich das PaaS-Modell mit den Jahren etabliert und bietet nun nahezu allen Anwendergruppen – von Freelance Entwicklern, über ISVs bis zu Enterprise-IT-Abteilungen – interessante Möglichkeiten zur Entwicklung und dem Betrieb moderner Anwendungen. Die Evolution der PaaS-Plattformen in den letzten Jahren lässt sich dabei in oben genannte Phasen bzw. Generationen einteilen.

Hingegen war die automatische Skalierbarkeit (Autoscaling) direkt weitverbreitet, da diese typischerweise für das Cloud Computing steht. Autoscaling sorgt dafür, dass die Anzahl der Server unterhalb einer PaaS-Plattform automatisch skaliert.

Bei steigender Serverlast wird die Arbeit dann von einer größeren Anzahl an Servern übernommen. Sinkt die Serverlast, werden die nicht mehr benötigten Server automatisch heruntergefahren.

## **2.4. Platform-as-a-Service der 1. Generation**

---

„Application-Platform-as-a-Services“ (aPaaS, siehe Kapitel 3.1) waren die Plattformen der ersten Generation (2006-2010). Diese richteten sich bevorzugt an Entwickler von Mobile- und Web-Applikationen. Auf Grund ihres Reifegrads waren sie aber zunächst wenig belastbar und wurden nicht mit für den Unternehmenseinsatz tauglichen SLAs (Service Level Agreements) gestützt. Als problematisch stellte sich in den Anfangsjahren zudem heraus, dass nur wenige Programmiersprachen unterstützt wurden und auf Grund von proprietären Implementierungen die Nutzer mit Restriktionen zu kämpfen hatten.

Diese Situation hat sich mittlerweile grundlegend verbessert. Zum einen können auf den diversen Plattformen zahlreiche Programmiersprachen verwendet werden. Zum anderen hat sich der Reifegrad der Plattformen so weiterentwickelt, dass sich diese für unternehmensrelevante Projekte einsetzen lassen. Auf Grund der konstanten Verbesserungen und Innovationen in den vergangenen Jahren gehören aPaaS-Plattformen zu den populärsten PaaS-Angeboten und erfreuen sich einer steigenden Beliebtheit. Insbesondere ISVs haben das Potential erkannt, um ihre bestehenden SaaS-Implementierungen auf einen aPaaS zu migrieren bzw. ihre vorhandenen Softwarelösungen darauf zu transformieren.



## 2.5. Platform-as-a-Service der 2. Generation

---

Wurden PaaS-Angebote der ersten Generation noch über eine Public Cloud bereitgestellt, bietet die zweite Generation (2010-2015) ebenfalls Betriebskonzepte aus einer Hosted- und Managed Cloud (siehe Kapitel 5), über welche bekannte PaaS-Stacks wie z.B. OpenShift oder Microsoft Azure Pack zur Verfügung gestellt werden. Sprich die PaaS-Dienste können auch aus lokalen Rechenzentren bezogen werden.

Zusätzlich zu aPaaS haben sich neue Spielformen von PaaS etabliert. Eine neue Produktkategorie ist die „Integration Platform-as-a-Services“ (iPaaS, siehe Kapitel 3). Diese fungieren als Middleware bzw. Integrations-Layer, um verschiedene Cloud-Services und –Plattformen miteinander zu verbinden. Beispiele sind Mulesoft, Informatica oder Skyyyva.

Die PaaS-Plattformen der zweiten Generation zeichnen sich zudem durch neue Funktionen, Methoden und Technologien aus:

### Cloud-based IDE

hier werden Entwicklungsumgebungen direkt auf der PaaS-Plattform, sprich im Browser, bereitgestellt.

### Applikations-Container

Die Applikations-Container Technologie, wie die des Anbieters Docker, ermöglichen das Bereitstellen vollständiger Applikationen innerhalb eines virtuellen Containers. Dies ist möglich, indem die Applikation und alle ihre Abhängigkeiten gebündelt werden und sich somit auf jeden Linux Server beliebig verschieben und betreiben lassen. Dies erleichtert nicht nur die Deployment- und Testprozesse, sondern verringert für Entwickler und Anwender auch das Risiko eines Vendor-Lock-In.

### Ökosysteme

PaaS-Plattform Anbieter stellen anhand von „Add-ons“ ein Ökosystem/Marktplatz für die Erweiterung von PaaS Applikationen bereit. Zu Lösungen innerhalb solcher Ökosysteme zählen zum Beispiel Lasttest-as-a-Service, Database-as-a-Service, Application Performance Monitoring und Analytics oder E-Mail-as-a-Service.

Das Kontroll Level der PaaS Angebote ist weiterhin gering. Durch neue IaaS+ Ansätze (siehe Kapitel 3), wie dem Amazon Elastic Beanstalk, nähern sich IaaS-Angebote aber näher an PaaS-Plattformen an, indem der Nutzer automatisiert die notwendige Infrastruktur bereitgestellt bekommt und im Anschluss Konfigurationen an den Servern und der Software vornehmen kann.

Weiterhin haben sich viele PaaS-Anbieter auf Basis von Marktplätzen ein Ökosystem von Partnern aufgebaut. Auf den Marktplätzen stehen externe Cloud-Services und Datensätze bereit, die sich als Teilkomponenten oder Datenstream via API in die eigenen Anwendungen integrieren lassen. Dabei handelt es sich beispielsweise um Services für den E-Mail oder SMS-Versand oder Monitoring- und Datenbank-Services.

## 2.6. Platform-as-a-Service der 3. Generation

Die PaaS-Plattformen der dritten Generation (ab 2015) werden sich insbesondere durch eine höhere Affinität für Unternehmenskunden auszeichnen, indem sich die Angebote weg von reinen Plattformen für Entwickler hin zu Applikationsplattformen für Unternehmen und ISVs weiterentwickeln. Das zeigt sich speziell darin, dass Technologien wie Docker weiter in den Mittelpunkt rücken, wodurch Applikationen portabler werden und dadurch echte Hybrid- und Multi-PaaS Strategien implementiert werden können.

Da etablierte Unternehmen auch bei der Anwendungsentwicklung nicht auf der grünen Wiese starten, müssen ständig bestehende Applikationen und Services miteinander integriert werden. iPaaS-Funktionalitäten und Services können hier maßgeblich unterstützen und bestimmte Integrationsaufgaben übernehmen bzw. automatisieren.

### PLATFORM-AS-A-SERVICE ENTWICKLUNG



QUELLE: Crisp Research AG  
 im Auftrag von PIRONET  
 NDH, 2014

Die Grenzen zwischen IaaS und PaaS werden weiter verschmelzen, wie die Evolution von Microsoft Azure und Google App Engine schon gezeigt hat. Nach Einschätzungen von Crisp Research werden die Plattformen zukünftig auf der Kontrollebene granularer werden. Nutzer, die mehr individuellen Einfluss auf die darunterliegende Infrastruktur nehmen wollen, erhalten hierzu die Möglichkeit.

Die PaaS-Plattformen der dritten Generation werden sich durch folgende Merkmale auszeichnen:

- Unterstützung von hybriden Betriebskonzepten und Architekturen
- Umsetzung komplexer Governance und rollenbasierter Security
- PaaS-Plattformen werden mit Enterprise-Appstores verzahnt und bilden ein neues Service Delivery-Modell innerhalb großer Organisationen
- Unterstützung von Technologien und Workflows zur Umsetzung von DevOps-Konzepten

## 3. Entwicklungsvarianten und Modelle

---

Nicht nur in der Cloud Service-Kategorie “Infrastructure-as-a-Service” existiert eine Vielzahl unterschiedlicher Spielarten und Betriebsmodelle. Auch im PaaS Bereich haben sich im Lauf der letzten Jahre verschiedene Varianten entwickelt, um Entwicklungsplattformen schneller und agiler einzusetzen.

Zwar existieren PaaS-Plattformen schon seit geraumer Zeit (Heroku 2007, Google AppEngine 2008, Microsoft Azure 2010), allerdings wurden diese bislang nur als reine „Public Cloud Services“ überwiegend aus US-Rechenzentren zur Verfügung gestellt. Nach einem substanziellen Reifeprozess auf Technologieseite in den letzten 24 Monaten und der Möglichkeit, die PaaS-Technologien (z.B. OpenShift, Azure, CloudFoundry) auch in flexiblen, lokalen Betriebskonzepten zu nutzen (z.B. Hosting bei lokalem Provider oder On-Premise im eigenen RZ), werden diese nun auch für deutsche Unternehmen zunehmend interessanter.

### 3.1. Spielarten von Platform-as-a-Service

---

#### **Application Platform-as-a-Service (aPaaS)**

Ein Application Platform-as-a-Service (aPaaS) stellt eine Cloud Umgebung zur Verfügung, auf der sich Unternehmensanwendungen entwickeln und betreiben lassen. Diese verfügt über eine graphische Weboberfläche und diverse Schnittstellen für die Programmierung (API). Beispiele für aPaaS sind Progress Software mit Rollbase sowie Heroku, Cloud Foundry oder OpenShift.

#### **Integration Platform-as-a-Service (iPaaS)**

Ein Integration Platform-as-a-Service (iPaaS) stellt eine Umgebung für die Integration von verschiedenen Services bereit, mit denen sich mehrere Cloud-Services aber auch on-Premise Applikationen verbinden lassen. Ein iPaaS löst dabei die sonst dafür genutzten Middleware-Lösungen ab. Beispiele für einen iPaaS sind Anbieter wie MuleSoft, Dell Boomi oder Microsoft Azure BizTalk.

#### **Infrastructure-as-a-Service Plus (IaaS+)**

Ein Infrastructure-as-a-Service Plus (IaaS+) bietet im Vergleich zu einem aPaaS eine deutlich größere Kontrollebene. Auf einem IaaS+ werden Applikationen automatisch bereitgestellt. Im Hintergrund spannt der Service, je nach benötigter Performance, die entsprechende Infrastruktur auf. Im Nachhinein erhält der Nutzer einen granularen Zugriff auf diese Infrastruktur und kann die jeweiligen Server und Software nach den eigenen Bedürfnissen hin konfigurieren. Ein Beispiel für ein IaaS+ ist Amazon Elastic Beanstalk.

### Portable vs. Vertikal integrierte PaaS

Technisch/Organisatorisch lassen sich zwei Typen PaaS unterscheiden:

Portable und vertikal integrierte. Zu den portablen gehören alle Plattformen, die auf offene Standards setzen und die sich selbst sowie die darauf entwickelten Applikationen zwischen Public und Private Clouds verschieben lassen.

Vertikal integrierte PaaS sind nahtlos mit den darunterliegenden IaaS kombiniert und sind daher nicht portabel.

## 3.2. Betriebskonzepte für Platform-as-a-Service

---

In den letzten Jahren haben sich verschiedene Betriebskonzepte von Platform-as-a-Services entwickelt, welche sich überwiegend durch die Art des Bereitstellungsmodells und der Intensität von Beratungsleistungen definieren.

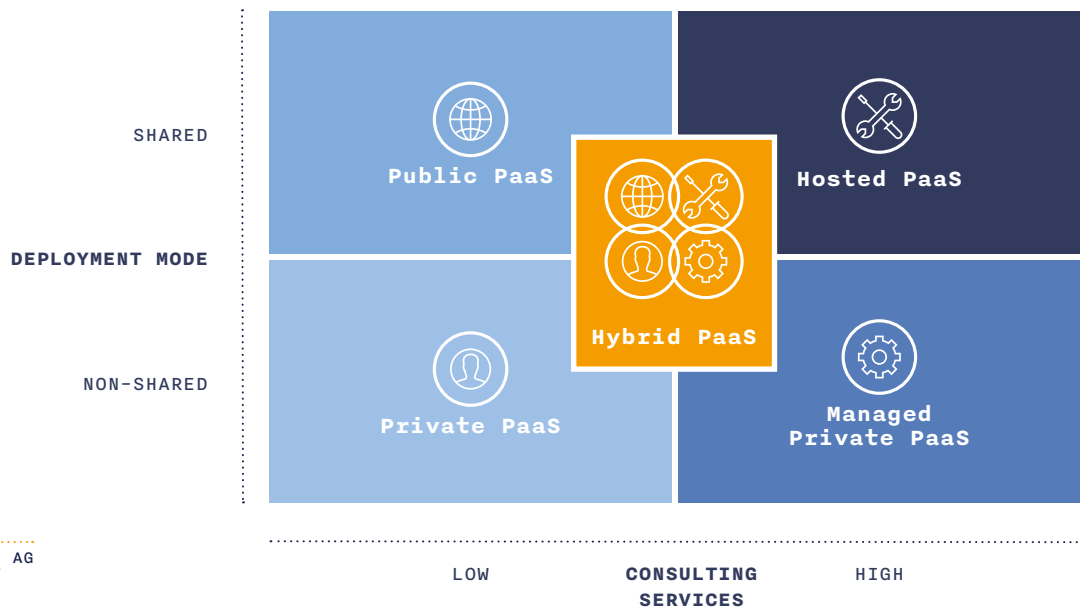
Für einen Zeitraum von etwa fünf Jahren dominierten die ersten Angebote als Public PaaS. In diesem Modell teilen sich alle Kunden dieselbe physikalische Infrastruktur und sind virtualisiert voneinander separiert. Gleichzeitig steht der Self-Service im Mittelpunkt. Das bedeutet, dass von der Anbieterseite keine weiteren Beratungsleistungen in Anspruch genommen werden können.

Private PaaS-Lösungen kapseln alle technischen Funktionen eines Public PaaS in eine Software, die sich für den Eigenbetrieb einsetzen lässt. Da die Software in erster Linie für den Einsatz in Unternehmen gedacht ist, befindet sich der PaaS in einer nicht geteilten Umgebung auf einer selbstverwalteten Infrastruktur. Der Nutzer ist in diesem Fall Selbstversorger wodurch keine Beratungsleistungen vorhanden sind.

Diese Private PaaS-Lösungen werden mittlerweile auch verstärkt von der Hosting Branche genutzt, um eigene PaaS-Lösungen in einem Hosted PaaS beziehungsweise Managed Private PaaS Modell anzubieten. In beiden Fällen nimmt die Beratungsleistung einen hohen Stellenwert ein, um den Kunden bei der Transition und Entwicklung seiner Applikationen zu unterstützen. Ein Self-Service, wie er von dem Public PaaS bekannt ist, existiert nicht. Ein Hosted PaaS unterscheidet sich von einem Managed Private PaaS in seinem Bereitstellungsmodell. Wo die Hosted Version auf eine physikalische Infrastruktur setzt, bei der die Kunden virtualisiert voneinander getrennt sind, stellt ein Managed PaaS eine exklusive physikalische Infrastruktur pro Kunde bereit, die von dem Anbieter verwaltet wird.

Das Hybrid PaaS-Modell ermöglicht die Verbindung und Integration zwischen den unterschiedlichen PaaS-Plattform-Varianten. Je nach Szenario lassen sich damit die unterschiedlichen Modelle miteinander kombinieren.

#### PLATFORM-AS-A-SERVICE METHODOLOGY



QUELLE: Crisp Research AG  
im Auftrag von PIRONET  
NDH, 2014

#### Public PaaS

Eine Public PaaS-Umgebung besteht aus einer sogenannten Shared Infrastructure. Hier teilen sich alle Kunden dieselbe physikalische Infrastruktur. Die PaaS-Plattform ist für jeden Kunden auf Basis einer virtualisierten Sicherheitsinfrastruktur logisch voneinander getrennt. Der Zugriff auf die Plattform erfolgt über das öffentliche Internet. Einige Anbieter verfügen mittlerweile über sogenannte Direct Connect bzw. VPN Verbindungen mit denen direkt und exklusiv auf die Infrastruktur zugegriffen werden kann.

#### Private PaaS

Innerhalb einer Private PaaS-Plattform betreibt der Kunde seine eigene PaaS-Plattform und ist für den Aufbau und dessen Wartung zuständig. Eine Private PaaS-Plattform wird den Designkonzepten einer Public PaaS-Plattform hinsichtlich Skalierbarkeit und Flexibilität nachgebaut. Allerdings ist auf Grund nicht vorhandener Skaleneffekte die Skalierbarkeit einer Public PaaS-Plattform nahezu unmöglich.

### Hybride PaaS-Umgebungen

Eine Hybrid PaaS-Plattform verbindet eine Private PaaS-Plattform mit den Ressourcen einer Public PaaS-Plattform. In diesem Fall betreibt ein Unternehmen seine eigene PaaS-Plattform und nutzt die Skalierbarkeit und Skaleneffekte eines Public PaaS Anbieters, um sich bei Bedarf oder auch ständig weitere Ressourcen in Form von Rechenleistung oder anderen Services zu bedienen.

### Hosted PaaS

Das Hosted PaaS-Modell überführt die Idee des Public PaaS in eine gehostete und von einem lokalen Anbieter verwaltete Variante. Hierbei befinden sich alle Kunden auf derselben physikalischen Infrastruktur und sind virtuell sicher voneinander getrennt. Der PaaS-Anbieter verfügt in diesem Fall über ein lokales Rechenzentrum, zum Beispiel in Deutschland. Die PaaS-Plattformen basieren dabei auf diversen Technologiestacks von Public PaaS-Anbietern wie dem Microsoft Azure Pack, Red Hat OpenShift oder Cloud Foundry.

### Managed Private PaaS

Ein Managed Private PaaS ist eine erweiterte Variante des Hosted PaaS. Dieser ist insbesondere für Unternehmen attraktiv, die das Public PaaS-Modell (Shared Infrastructure, Multi-Tenancy) meiden wollen, aber nicht die finanziellen Ressourcen und das Wissen besitzen, um eine PaaS-Plattform in der eigenen IT-Infrastruktur bereitzustellen. In diesem Fall stellt der Anbieter einem Kunden einen exklusiven und reservierten physikalischen Bereich auf seiner Infrastruktur bereit. Der Kunde kann den Managed Private PaaS exakt so nutzen wie einen Public PaaS, aber das auf einer nicht geteilten Infrastruktur, die sich bei einem Anbieter im Rechenzentrum befindet. Zudem steht der Anbieter mit Beratungsleistungen zur Seite, die dem Kunden dabei helfen, seine Applikationen entweder auf die Plattform zu überführen oder dort neu zu entwickeln.

### 3.3. Entwicklungsumgebung (IDE) - lokal oder aus der Cloud

---

Typischerweise wird für die Entwicklung von Applikationen für einen PaaS eine lokale Entwicklungsumgebung (IDE) verwendet. Der Quellcode wird auf dem Computer des Entwicklers geschrieben und anschließend auf die PaaS-Plattform kopiert, wo dieser ausgerollt und betrieben wird. Werden Änderungen vorgenommen, erfolgen diese ebenfalls zunächst lokal und werden dann auf die PaaS-Plattform übertragen.

Neue PaaS-Varianten gehen einen Schritt weiter und haben die IDE direkt integriert, sprich die IDE wird im Browser dargestellt. Dies ermöglicht das Schreiben, vornehmen von Änderungen und das Ausrollen direkt auf dem PaaS selbst. Dadurch wird ebenfalls die Zusammenarbeit verbessert, da mehrere Entwickler parallel an demselben Quellcode arbeiten können.



## 4. Einsatzbereiche

---

Unternehmen und Entwickler können PaaS-Plattformen in unterschiedlichen Szenarien und Use Cases einsetzen. Vorab sollten Anwender ihre Anforderungen festlegen und überlegen, ob folgende Voraussetzungen (vgl. Abschnitt 6.1) in ihrem jeweiligen Projekt vorliegen. Die relevantesten Use Cases für den Einsatz von PaaS werden in Abschnitt 6.2 skizziert und abschließend die Möglichkeit der Anwendungsintegration via PaaS (Abschnitt 6.3) beschrieben.

### 4.1. Voraussetzungen für den Einsatz von Platform-as-a-Service

---

Auch wenn PaaS derzeit als einer der heißen Trends im Cloud Computing gehandelt wird und die Plattformen in den letzten Jahren deutlich herangereift sind, müssen Unternehmen und Entwickler trotzdem kritisch fragen, ob PaaS für das jeweilige Projekt Sinn macht oder nicht. Wenn im Rahmen eines geplanten Software-Entwicklungsprojektes folgende Voraussetzungen gegeben sind, lassen sich mittels PaaS deutliche Vorteile realisieren:

- Es sind mehrere Releases pro Jahr geplant, um die Applikation funktional zu erweitern (ja/nein?)
  - Die Anwendungsentwicklung soll nach einer agilen Software-Entwicklungsmethode erfolgen (ja/nein?)
  - Die Einbindung mehrerer APIs ist geplant (ja/nein?)
  - Die Entwicklung setzt den Einsatz moderner Web-Frameworks und Entwicklungstools voraus (ja/nein?)
  - Der IT-Infrastrukturbedarf (Server-Leistung, Speicherkapazität) ist zu Beginn des Projektes nur schwer abschätzbar (ja/nein?)
  - Das erste Release soll in weniger als 6 Monaten erfolgen (ja/nein?)
  - Für den IT-Betrieb der Anwendung steht kein Vollzeit-Administrator bereit (ja/nein?)
-

Für Mittelstands- und Großunternehmen mit eigener Anwendungsentwicklung kann der Einsatz von PaaS aber auch eine strategische Frage sein. Plant das Unternehmen z.B. im Rahmen von größer angelegten Enterprise Mobility- oder Digital Transformation-Programmen die Entwicklung neuer Anwendungen ggf. auch mit externen Partnern, liefern PaaS-Plattformen ein hilfreiches Werkzeug, um die Anwendungsentwicklung und den Betrieb über die Unternehmensgrenzen hinweg zu koordinieren sowie Qualitäts- und Kontrollstandards auch für die externen Partner festzulegen und zu überwachen.

## 4.2. Use Cases für den Einsatz von Platform-as-a-Service

---

Die Bandbreite möglicher Use Cases für den Einsatz von PaaS ist sehr groß. Im Praxiseinsatz haben sich in den letzten Jahren vor allem folgende Use Cases als besonders relevant herauskristallisiert. Für diese treffen auch die meisten der in Abschnitt 6.1. beschriebenen Voraussetzungen zu.

- Web Applikationen
  - eCommerce
  - Media Streaming
  - Social Web/Social Collaboration
  - Mobile Applikationen/Apps
  - Analytics/Reporting
  - Transaktionale Anwendungen
- 

Da die Entwicklung von Enterprise Applikationen nur noch in den wenigsten Unternehmen stattfindet, sondern sich maßgeblich auf die Integration und Individualisierung von Standard-Software beschränkt, spielen PaaS-Plattformen in diesem Kontext bislang eine untergeordnete Rolle. Mit der Verfügbarkeit der ersten Enterprise-PaaS-Angebote, sprich gehosteten PaaS-Umgebungen inklusive der notwendigen Compliance- und Government-Funktionen, ändert sich das Bild sukzessive. Gerade im Hinblick auf die Verknüpfung von klassischen Enterprise- und SAP-Anwendungen mit modernen Web-Anwendungen und mobilen Frontends, werden PaaS-Plattformen auch für die Enterprise-IT relevant.

### 4.3. Anwendungsintegration über Platform-as-a-Service

---

Neben der Entwicklung und dem Betrieb neuer Anwendungen, können PaaS-Plattformen (hier speziell iPaaS) eine wichtige Rolle bei der Zusammenführung verschiedener SaaS-Anwendungen sowie der Umsetzung von hybriden Cloud-Umgebungen spielen. Überall dort, wo SaaS-Anwendungen, wie z.B. Salesforce CRM oder Social Collaboration Dienste wie Microsoft SharePoint Online, auf interne Datenbestände oder Nutzerprofile (LDAP/Active Directory) zugreifen, muss eine intelligente Integrationsschicht implementiert werden. Diese lässt sich mittels iPaaS-Plattformen meist deutlich leichter und flexibler realisieren als mit den althergebrachten Middleware-Stacks. Diese bieten zudem die Möglichkeit die Service Integration weitgehend zu standardisieren und eine Integrations-Architektur zu schaffen, die die Anbindung neuer Cloud-Dienste und APIs in der Zukunft vereinfacht.

## 5. Anforderungen

---

### 5.1. Verfügbarkeit

---

Die PaaS-Plattform muss technisch sicherstellen, dass der Zugriff auf die darauf betriebenen Applikationen als auch auf die Backend-Infrastruktur jederzeit möglich ist. Es liegt im Verantwortungsbereich des Anbieters dies sicherzustellen. Der Kunde muss, je nach Plattform des Anbieters, dies programmatisch in seiner Applikation berücksichtigen.

### 5.2. Skalierbarkeit

---

Die flexible Anpassbarkeit der Infrastrukturbasis ist ein wichtiges Merkmal einer PaaS-Plattform. Der Anbieter hat dafür zu sorgen, dass die Plattform den Anwendungen im Bedarfsfall Möglichkeiten für die automatisierte Skalierbarkeit zur Verfügung stellt. Der Kunde muss, je nach Plattform des Anbieters, dies programmatisch in seiner Applikation berücksichtigen.

### 5.3. Multiple Zugriffsmöglichkeiten

---

Die Plattform muss dem Kunden unterschiedliche Möglichkeiten bieten, um mit der Plattform zu interagieren. Hierzu gehört der Zugriff auf APIs, über eine Kommandozeile und den Webbrowser.

### 5.4. Ökosystem und Partner

---

Ein umfangreiches und konstant wachsendes Ökosystem von Partnern zeugt von einer gewissen Attraktivität des Anbieters und ermöglicht dem Kunden Zugriff auf weitere externe Ressourcen, um seine Applikationen mit Mehrwertdiensten einfacher zu erweitern.

### 5.5. Multiple Entwicklungsframeworks & -Sprachen

---

Eine große Auswahl an unterschiedlichen Frameworks und eine vielfältige Unterstützung von Programmiersprachen hilft potentiellen Kunden, dabei sich für eine Plattform zu entscheiden, um ihre vorhandenen Applikationen zu migrieren oder neue darauf zu entwickeln.

## 5.6. Multiple Betriebskonzepte

---

Je nach Einsatzszenario müssen sich unterschiedliche Betriebskonzepte umsetzen lassen, um die individuellen Bedürfnisse zu erfüllen. Moderne PaaS-Plattformen sollten in der Lage sein, Public-, Private-, Hybrid-, Managed- oder Hosted-Konzepte zu ermöglichen.

## 5.7. Professioneller Support und Service

---

### Level Agreements

---

Insbesondere für Unternehmen ist eine professionelle Unterstützung von entscheidender Bedeutung, um ihre bestehenden Applikationen entweder auf die PaaS-Plattform zu migrieren oder darauf neu zu entwickeln. In diesem Zusammenhang sind die Service Level Agreements hervorzuheben, mit welchen die Verfügbarkeit auf vertraglich-organisatorischer Ebene geregelt werden. Damit kann ein Anbieter auf die Qualität hinsichtlich der Verlässlichkeit seines Service aufmerksam machen. Dem Kunden steht damit ein Vertragswerk zur Verfügung, auf das er sich berufen kann.

## 5.8. Sicherheit

---

Der Anbieter muss für die ganzheitliche Sicherheit der Plattform sorgen. Hierzu gehören alle Maßnahmen hinsichtlich der Authentifizierung, der Zugangskontrolle sowie der Autorisierung (Authentication, Access Control und Authorization (AAA)). Weiterhin müssen den Kunden auf Applikationsebene Möglichkeiten zur Verfügung stehen, damit diese ihre Anwendung schützen können. In diesem Zusammenhang ist ebenfalls die Frage zu klären, ob sich die unterschiedlichen Kundenapplikationen in irgendeiner Form gegenseitig beeinflussen.

## 6. Kriterien zur Anbietersauswahl

---

Wie oben zu sehen spielen bei der Auswahl einer PaaS-Plattform bzw. eines PaaS-Anbieters unterschiedliche Anforderungen eine Rolle. Hierbei sind noch weitere folgende Faktoren zu differenzieren:

- Anwender Typ (Startup, ISV, Unternehmens-IT oder Digitalagentur)
  - Dev/Test-Organisations- und Prozessmodell
  - Kategorie/Charakteristika der zu entwickelnden Anwendung
- 

Im Folgenden werden die wesentlichen Charakteristika und Anforderungen nach PaaS-Anwendertypen dargestellt und erläutert.

### 6.1. Startups/Freelance Entwickler

---

Startups und Entwicklern mangelt es in den ersten Phasen ihres Projektes vor allem an einem: Kapital. Investitionen in professionelle Infrastrukturressourcen für die Entwicklung und den Test einer Applikation sind sehr kostenintensiv. In den meisten Fällen ist von vornherein zudem überhaupt nicht klar, ob diese Vorabinvestitionen langfristig einen rentablen Umsatz generieren. Das bedarfsgerechte Abrechnungsmodell auf einer PaaS-Plattform eignet sich daher sehr gut für junge Unternehmen, die einen ersten Prototypen entwickeln und der Öffentlichkeit präsentieren wollen. Sollte der notwendige Erfolg ausbleiben, lassen sich die genutzten Ressourcen auf der Plattform abschalten und verursachen keine Kosten mehr.

Ausgehend vom positiven Fall vernachlässigen viele junge Unternehmen, dass die größte Komplexität aus dem zwangsläufigen Wachstum hervorgeht. Der plötzliche Ruhm "über Nacht" hat schon mehr als ein Startup erwischt. Sind Startups und Entwickler bei steigenden Zugriffs- und Nutzerzahlen nicht in der Lage ihre Infrastruktur entsprechend auszubauen bzw. skalieren, drohen ernste Konsequenzen - unzufriedene Nutzer, weniger Umsatz und ein schlechtes Image.

Die zentralen Anforderungen seitens Startups und freiberuflichen Entwicklern an geeignete PaaS-Plattformen sind:

## Anforderungen an PaaS - Startups

---

- Typ**
- Freiberufliche Entwickler
  - Verteilte Entwicklerteams
  - Kleine ISVs, Software Startups

- Eigenschaften**
- Meist keine eigene IT-/RZ-Infrastruktur
  - Schnelle Entwicklungszyklen als Erfolgsfaktor
  - Fokus auf neue Programmiersprachen und Tools
  - Wenig/kein Budget für lokale Dev/Test-Infrastrukturen

- Kernanforderungen**
- Bandbreite von Ressourcen/Images
  - Unterstützung von mehreren Entwickler-Werkzeugen
  - Hoher Grad an Self-Service/gute API
  - Herausragende Benutzbarkeit
  - Pay per use Preismodell/Go-to-Market Support

- Typische Anwendungsfälle**
- Websites
  - eCommerce/eShops
  - Media Apps
  - Web-Applikationen
  - Content Management Systeme
  - Mobile Apps

## 6.2. Independent Software Vendor (ISVs)

---

Independent Software Vendor (ISVs) blicken auf eine lange und erfolgreiche Geschichte zurück. Allerdings kommt der Großteil durch die aufstrebende Nutzung von SaaS-Applikationen und Cloud-Services immer mehr in Bedrängnis. Die Gründe sind offensichtlich. Die Kundengruppen stellen nach und nach ihr Nutzungsverhalten von On-Premise-Software auf Cloud-basierte Applikationen um. Dadurch werden Investitionen in nicht genutzte Lizenzen eingespart, als auch Kosten für die Wartung der Software verringert. Gleichzeitig bieten SaaS und Cloud-basierte Applikationen den Mitarbeitern einen einfachen orts- und geräteunabhängigen Zugriff.

Aber auch die ISVs selbst können von der Umstellung auf ein PaaS-basiertes Entwicklungs- und Betriebsmodell profitieren. Wird die neue Produktgeneration auf Basis einer Multi-Tenancy-Architektur entwickelt und bereitgestellt, lassen sich Entwicklungsaufwand und Wartungskosten deutlich reduzieren – bei gleichzeitiger Erhöhung der Innovations-Taktrate bzw. der Releases. Um von diesen Vorteilen zu profitieren, müssen ISV aber nicht nur ihre Entwicklungs- und IT-Betriebsprozesse umstellen und in Richtung „Continuous Releases“ bzw. „Continuous Delivery“ umstellen.

In der Konsequenz ergeben sich hieraus für die ISVs allerdings auch vollkommen neue Herausforderungen beispielsweise an die Beschreibung und vertragliche Dokumentation ihrer künftigen Leistungszusagen und Cloud-Angebote (u.a. umfangreiche und rechtssichere, zugleich aber auch marktfähige Service Level Agreements) gegenüber ihren jeweiligen Endkunden. Darüber hinaus sind durch die Adressierung zusätzlicher (Online-)Zielgruppen ergänzende, primär webbasierte Vermarktungs- und Vertriebsstrategien zu entwickeln, um die neuen Cloud-Angebote tatsächlich erfolgreich im Markt einführen und positionieren zu können. Für ISVs sind mithin PaaS-Anbieter interessant, die sie auch auf diesem Transformationsprozess wirkungsvoll unterstützen und begleiten können.

ISV haben daher auch spezielle Anforderungen in Hinblick auf die Auswahl von PaaS-Plattformen, die für den produktiven Betrieb eingesetzt werden sollen: (siehe Folgeseite)



## Anforderungen an PaaS - ISVs

---

- Typ**
- Professionelle Softwareunternehmen
  - Professionelle Dienstleistungsunternehmen mit eigener Softwareentwicklung
- 

- Eigenschaften**
- Verteilte Entwicklerteams (Mix aus On-Shore / Off-Shore)
  - Wiederverwendung von Komponenten, Test-Umgebungen und Prozessen sehr wichtig
  - Intensive Nutzung meist eigener Entwicklungs- und Test-Umgebungen
  - Langfristige Transformation in Richtung SaaS/Cloud-Business Modelle
- 

- Kernanforderungen**
- Solide Administration und Sicherheitsfunktionen
  - Unterstützung für mehrere Entwickler-Werkzeuge
  - Hohe Skalierbarkeit, um Enterprise-IT Anforderungen sicherzustellen
  - Hybrid- und Private Betrieb für PaaS-Umgebung - am besten mit lokalen RZ-Standort
  - Go-to-Market Support und Unterstützung im Transformationsprozess
- 

- Anwendungsfälle**
- Entwicklung standardisierter Softwarepakete
  - Migration bestehender Softwarepakete auf ein SaaS-Modell
  - Entwicklung individueller Software
  - SAP-Implementierung und individuelle Unternehmenssoftware
- 

 **QUELLE:** Crisp Research AG  
im Auftrag von PIRONET  
NDH, 2014

## 6.3. Unternehmens-IT

---

IT-Abteilungen sollten danach streben ihren internen Kunden den besten Service zu liefern und diese dabei unterstützen die Agilität und Prozessgeschwindigkeit zu verbessern.

Damit verbunden ist der schnelle und unkomplizierte Zugriff auf IT-Ressourcen durch die Fach- und Entwicklungsabteilungen. Denn um die Wettbewerbsfähigkeit zu stärken, sind ständige Tests und die Entwicklung von Prototypen erforderlich. Dies kann durch die Bereitstellung von PaaS-Umgebungen maßgeblich unterstützt werden.

Die Anforderungen aus Perspektive der Unternehmen setzen sich aus denjenigen der Fachabteilungen (schnelles Prototyping und Testing, moderne Entwicklungswerkzeuge, flexible Nutzung & Verrechnung der PaaS-Ressourcen) und denjenigen der IT-Abteilung (Security-Level, Governance & Rechtemanagement, Integration mit bestehenden Cloud-Umgebungen) zusammen. Sollen Anwendungen auf der PaaS-Plattform nicht nur entwickelt und getestet, sondern produktiv betrieben werden, spielen die Verfügbarkeit, SLAs und der RZ-Standort eine wichtige Rolle.

Für große Unternehmen sind vor allem Private PaaS-Umgebungen interessant, um die Verantwortung und Kontrolle von Daten und Entwicklungsprozessen in der Hand zu behalten und

---

steuern zu können. Sind später mehr Ressourcen erforderlich, können die Applikationen in einem hybriden Ansatz oder vollständig auf einen Public/Hosted PaaS ausgelagert werden.

## Anforderungen an PaaS - Unternehmens-IT

---

- Typ**
- Entwicklerteams in Unternehmen
  - Interne IT-Abteilungen
  - Innovationsteams

- Eigenschaften**
- Gemischte und verteilte Entwicklerteams (On-Shore, Off-Shore, Service-Provider)
  - Entwicklungsprojekte mit hohem Integrationsaufwand
  - Wiederverwendung von Komponenten, Test-Umgebungen und Prozessen sehr wichtig
  - Meist eigene IT-/RZ-Infrastrukturen für Betrieb von Dev/Test-Umgebungen verfügbar

- Kernanforderungen**
- Hochsichere Plattform, die interne Standards erfüllt
  - Unterstützung komplexer Entwicklungs- und Testprozesse und verteilter Teams
  - Klar definierte SLAs, die den internen Prozessen entsprechen
  - PaaS-Plattformen mit vielen Schnittstellen und Integrationsmöglichkeiten
  - Hybride-Betriebskonzepte für PaaS interessant

- Typische Anwendungsfälle**
- Digitalisierung bestehender und neuer Geschäftsprozesse
  - Migration/Portierung bestehender Lösungen auf Cloud-Plattformen
  - Analytics-Anwendungen
  - SAP-Erweiterungen und Add-Ons
  - Portale und eCommerce-Lösungen
  - Mobile Lösungen

## 7. Technologie und Anbieter

Technologischer Reifegrad, Produkt- und Funktionsspektrum, Skalierbarkeit der Infrastruktur, Entwicklungshistorie sowie die SLAs und IT-Betriebskonzepte haben entscheidenden Einfluss bei der Auswahl eines PaaS-Plattform Anbieters. Auf Basis einer Vielzahl an Beratungs- und Research-Projekten werden folgend die für den deutschsprachigen Markt relevantesten Anbieter kurz portraitiert.

Die folgenden Tabellen geben einen Überblick zu den Eigenschaften und den technischen Möglichkeiten (Deployment-Varianten, unterstützte Programmiersprachen) dieser Anbieter.

Anbieter	cloudControl	Google App Engine	Heroku
<b>Erscheinungsjahr</b>	2009	2008	2007
<b>Webseite</b>	<a href="http://cloudcontrol.com">http://cloudcontrol.com</a>	<a href="https://appengine.google.com">https://appengine.google.com</a>	<a href="http://heroku.com">http://heroku.com</a>
<b>Zielgruppe</b>	Entwickler, Digitalagenturen	Freelance Softwareentwickler, Startups	Freelance Softwareentwickler, Startups, Digitalagenturen
<b>Entwicklungsmodell</b>	Public PaaS, Private PaaS	Public PaaS	Public PaaS
<b>Programmiersprachen</b>	PHP, Java, Python, Ruby, Node.js	Python, Java, Go, PHP	Java, Node.js, Scala, Clojure, PHP, Perl
<b>Besonderheiten</b>	cloudControl besitzt keine eigene Infrastruktur und setzt hierzu auf dem Public IaaS der Amazon Web Services auf. Mit der „Application Lifecycle Engine“ kann die cloudControl Technologie auch als Private PaaS betrieben werden.	Teil der Google-Cloud Plattform und somit Zugriff auf extrem skalierungsfähige Infrastruktur. Zugriff via API auf andere Google Web bzw. Cloud Services.	Einer der erfahrensten PaaS-Anbieter mit breiter Anwender-/Kunden-Basis. 2010 von Salesforce akquiriert. Betrieb der Infrastruktur dennoch auf Amazon AWS.

Anbieter	HP Helion Public Cloud aPaaS	IBM Bluemix	Microsoft Azure
<b>Erscheinungsjahr</b>	2014	2014	2010
<b>Webseite</b>	<a href="http://www.hpcloud.com/products-services/application-paas">http://www.hpcloud.com/products-services/application-paas</a>	<a href="http://bluemix.net">http://bluemix.net</a>	<a href="http://azure.microsoft.com">http://azure.microsoft.com</a>
<b>Zielgruppe</b>	Unternehmen, Entwickler	Unternehmen, Softwareentwickler	Unternehmen, Softwareentwickler, ISVs
<b>Entwicklungsmodell</b>	Public PaaS, Private PaaS	Public PaaS	Public PaaS, Private PaaS (Azure Pack)
<b>Programmiersprachen</b>	Clojure, Go, Java, Node.js, Perl, PHP, Python, Ruby	Java, Node.js, Ruby	PHP, .NET, Node.js, Python
<b>Besonderheiten</b>	Die Basis des HP Helion Cloud aPaaS bildet der Open Source PaaS Cloud Foundry. Cloud Foundry ist ebenfalls Teil der HP Helion OpenStack Distribution, womit sich ein Private PaaS realisieren lässt.	Als Basis für IBM Bluemix dient die OpenSource PaaS-Lösung CloudFoundry.	Mit dem Microsoft Azure Pack können Private-PaaS-Umgebungen auf Basis einer eigenen IT-Infrastruktur oder im Hosted Betrieb aufgebaut und genutzt werden.

Anbieter	Pironet NDH Enterprise PaaS	Pivotal Web Services	Progress Software - Rollbase
<b>Erscheinungsjahr</b>	2013	2013	2013
<b>Webseite</b>	<a href="http://www.pironet-ndh.com/Enterprise+PaaS">http://www.pironet-ndh.com/Enterprise+PaaS</a>	<a href="https://run.pivotal.io">https://run.pivotal.io</a>	<a href="http://www.progress.com/products/rollbase">http://www.progress.com/products/rollbase</a>
<b>Zielgruppe</b>	Unternehmen, ISVs, Digitalagenturen	Unternehmen, Entwickler	Unternehmen, ISVs, Entwickler, Citizen Developer
<b>Entwicklungsmodell</b>	Hosted PaaS, Managed Private PaaS	Public PaaS, Private PaaS	Hosted PaaS, Private PaaS
<b>Programmiersprachen</b>	PHP, .NET, Node.js, Python, JavaScript, Ruby, Perl, Haskell	Java, Node.js, Ruby	Java, C, JavaScript, OpenEdge ABL
<b>Besonderheiten</b>	PIRONET NDH bietet als erster Managed Service Provider in Deutschland eine gehostete Variante von Microsoft Azure (Azure Pack) sowie Red Hat OpenShift aus einem deutschen Rechenzentrum.	Pivotal ist ein Spin-off von VMware und EMC und Entwickler der Open Source PaaS-Lösung Cloud Foundry. Somit ist Cloud Foundry ebenfalls die Basis des Pivotal Public und Private PaaS.	Rollbase ist Teil der Progress Pacific Plattform und dient als Basis für das Rapid Application Development (RAD) von SaaS-Applikationen sowie deren Bereitstellung. Die Lösung kann entweder als Private PaaS oder in einem gehosteten Modell verwendet werden.

Anbieter	Red Hat OpenShift	SalesForce1	T-Systems Cloud Integration Center
<b>Erscheinungsjahr</b>	2011	2008 (ehemals Force.com)	2014
<b>Webseite</b>	<a href="https://openshift.com">https://openshift.com</a>	<a href="http://www.salesforce.com/de/platform/overview/">http://www.salesforce.com/de/platform/overview/</a>	<a href="http://testen-und-bestellen.t-systems.de/ecp-de/testen-bestellen/cloud-integration-center-applikationen/6290">http://testen-und-bestellen.t-systems.de/ecp-de/testen-bestellen/cloud-integration-center-applikationen/6290</a>
<b>Zielgruppe</b>	Unternehmen, Softwareentwickler	Salesforce Entwickler bzw. Salesforce-Anwender	Unternehmen
<b>Entwicklungsmodell</b>	Public PaaS, Private PaaS	Public PaaS	Managed Private PaaS
<b>Programmiersprachen</b>	JavaScript, Ruby, Python, PHP, Perl, Java, Haskell, .NET	Apex	JavaScript, Ruby, Python, PHP, Perl, Java, Haskell, .NET
<b>Besonderheiten</b>	OpenShift kann als Public Service genutzt werden. OpenShift Enterprise ist die Variante für die Private Cloud.	Dient als Plattform für die Entwicklung von Applikationen und Services für Salesforce.	Die Plattform basiert auf Red Hat OpenShift und setzt als Basis auf die globale VMware vCloud Infrastruktur von T-Systems.

**Anbieter Website**

---

<b>Amazon Elastic Beanstalk</b>	<a href="http://aws.amazon.com/de/elasticbeanstalk/">http://aws.amazon.com/de/elasticbeanstalk/</a>
<b>Appelon</b>	<a href="https://www.appelon.com">https://www.appelon.com</a>
<b>Apprenda</b>	<a href="http://apprenda.com">http://apprenda.com</a>
<b>AppScale</b>	<a href="http://www.appscale.com">http://www.appscale.com</a>
<b>CenturyLink</b>	<a href="http://www.centurylink.com">http://www.centurylink.com</a>
<b>CloudBees</b>	<a href="http://cloudbees.com">http://cloudbees.com</a>
<b>Dell Boomi</b>	<a href="http://www.boomi.com">http://www.boomi.com</a>
<b>Docker</b>	<a href="https://www.docker.com">https://www.docker.com</a>
<b>EngineYard</b>	<a href="https://www.engineyard.com">https://www.engineyard.com</a>
<b>GBS AG</b>	<a href="http://www.gbs.com/de">http://www.gbs.com/de</a>
<b>IBM</b>	<a href="http://www-03.ibm.com/software/products/de/cast-iron-live">http://www-03.ibm.com/software/products/de/cast-iron-live</a>
<b>Informatica</b>	<a href="http://www.informatica.com">http://www.informatica.com</a>
<b>Jelastic</b>	<a href="http://jelastic.com">http://jelastic.com</a>
<b>Jitterbit</b>	<a href="http://www.jitterbit.com">http://www.jitterbit.com</a>
<b>Mulesoft</b>	<a href="http://www.mulesoft.com">http://www.mulesoft.com</a>
<b>Nodejitsu</b>	<a href="https://www.nodejitsu.com">https://www.nodejitsu.com</a>
<b>Oracle</b>	<a href="https://www.oracle.com/cloud/paas.html">https://www.oracle.com/cloud/paas.html</a>
<b>SAP</b>	<a href="http://www.sap.com/pc/tech/cloud/software/hana-cloud-integration/index.html">http://www.sap.com/pc/tech/cloud/software/hana-cloud-integration/index.html</a>
<b>Skyvva</b>	<a href="http://skyvva.com">http://skyvva.com</a>
<b>SnapLogic</b>	<a href="http://www.snaplogic.com">http://www.snaplogic.com</a>
<b>Software AG</b>	<a href="http://www.softwareag.com/de/products/softwareag_live/integration_live/default.asp">http://www.softwareag.com/de/products/softwareag_live/integration_live/default.asp</a>
<b>TerraSky</b>	<a href="http://www.terrasky.com/skyondemand/">http://www.terrasky.com/skyondemand/</a>
<b>Tibco</b>	<a href="https://cloudbus.tibco.com">https://cloudbus.tibco.com</a>
<b>WSO2</b>	<a href="http://www.wso2.com">http://www.wso2.com</a>

## 8. Platform-as-a-Service im Praxiseinsatz

---

Mit der weltweit steigenden Digitalisierung wächst auch die Attraktivität von PaaS-Umgebungen. Für den deutschen Markt hat Crisp Research im Auftrag der PIRONET NDH 83 deutsche Softwarehäuser befragt. Im Fokus der Studie stand eine empirische Untersuchung zur Cloud-Transformation deutscher ISVs und dem Einsatz von PaaS-Plattformen.

Deutsche Softwarehäuser haben ganz bestimmte Anforderungen, wenn es um den Einsatz von PaaS im Rahmen ihrer Entwicklungs-, Test- und Betriebsprozesse geht. Besonders relevant sind für die ISVs die Konditionen des Betriebs bzw. das Betriebskonzept, sprich ob die PaaS-Dienste nach dem Public-, Private- oder im Rahmen eines Hosting-Modells betrieben werden. Und hier hat sich in den vergangenen 12 Monaten viel getan.

So waren bis Mitte 2013 keine der führenden PaaS-Plattformen als Technologie-Stack für den Eigen- oder Hosting-Betrieb verfügbar. Bis dato war Public Cloud das einzig verfügbare Betriebsmodell. Mit dem Release der CloudFoundry-Technologie unter Open Source- Lizenz machte VMware (heute bei Pivotal) den Anfang. Mittlerweile können auch die PaaS-Technologien von Microsoft (Microsoft Azure Pack), Red Hat (OpenShift), VMware (vFabric) als Software-Paket bezogen und individuell betrieben werden. Dies bietet Unternehmen, Softwareentwicklern und Hosting-Partner vollkommen neue Möglichkeiten PaaS-Dienste kundenindividuell anzubieten und hinsichtlich bestimmter Anwendungsszenarien zu optimieren.

Welche der Technologien sich mittel- und langfristig durchsetzen wird, ist derzeit noch schwer prognostizierbar, da die meisten ISVs, Hosting Provider und Unternehmensanwender sich noch nicht final festgelegt haben. Während IBM und HP auf CloudFoundry setzt, bieten eine Reihe an Hosting Providern ihren Kunden die Auswahl zwischen mehreren Plattformen.

Derzeit existiert im deutschsprachigen Markt allerdings erst eine Handvoll Hosting Provider, die PaaS in einem Hosting Modell anbieten. Dies hat verschiedene Gründe.

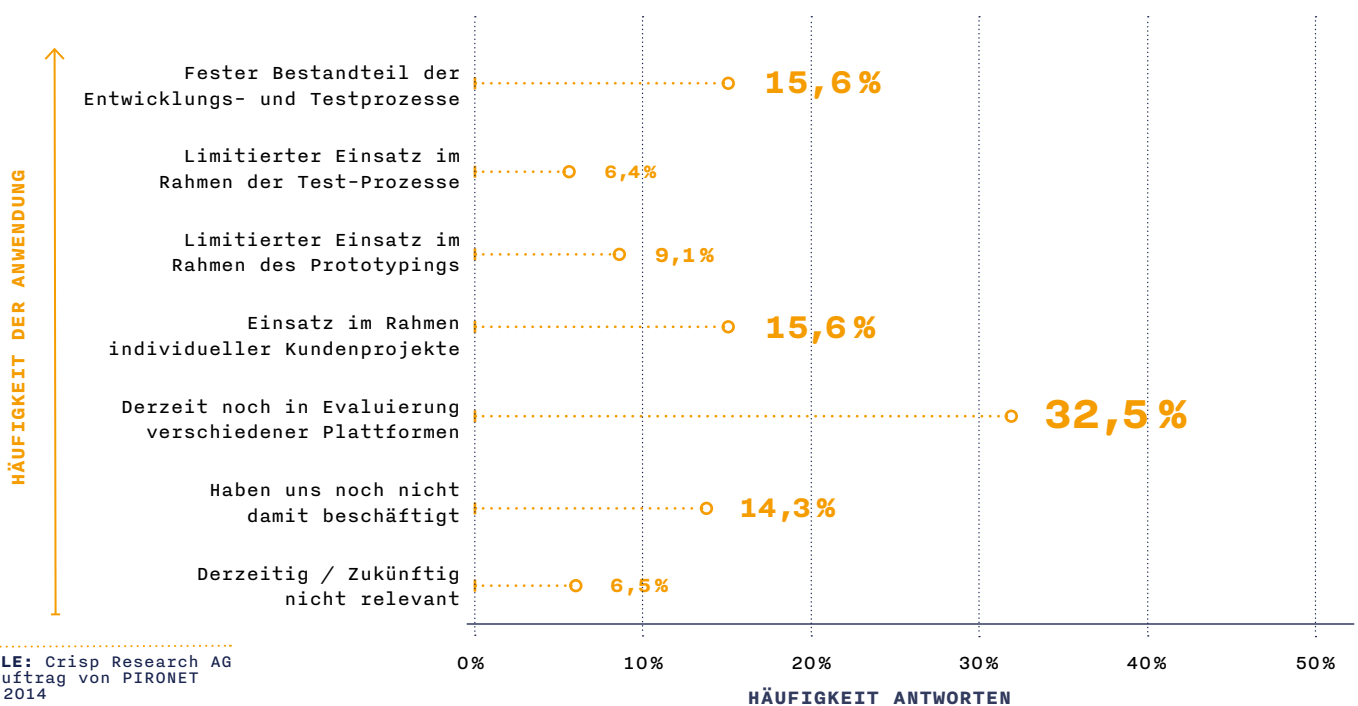
So schätzen einige Akteure wohl die Marktchancen nicht als aussichtsreich ein. Anderen fehlen schlicht die Ressourcen und Skills, um die technologisch anspruchsvollen und komplexen Plattformen aufzusetzen und zu betreiben. Auch müssen Hosting Provider mehr Verständnis und Know How in Bezug auf die Prozesse und speziellen Anforderungen der ISVs, Startups, Freelance und Unternehmensentwickler aufbauen. Denn deren Bedürfnisse unterscheiden sich deutlich von den Anforderungen

an ein klassisches Hosting oder Infrastruktur-Outsourcing. Die Anzahl erfahrener Architekten, Entwickler und Projektmanager zum Thema PaaS ist in Deutschland noch sehr übersichtlich.

## 8.1. Einsatzgrad und Planung

PaaS-Angebote sind heute bereits vielfach fester Bestandteil moderner Software-Entwicklungs-, Test- und Betriebsverfahren. Allerdings werden diese Angebote vorwiegend im anglo-amerikanischen Raum und von Entwicklern genutzt, die nicht oder nur selten im Unternehmensumfeld tätig sind. Grund dafür war die bis vor kurzem limitierte Verfügbarkeit von Alternativen zu den großen Public Cloud-Angeboten.

### Welche Rolle spielen PaaS- und Cloud-Plattformen in ihrem Entwicklungsprozess?



QUELLE: Crisp Research AG  
im Auftrag von PIRONET  
NDH, 2014

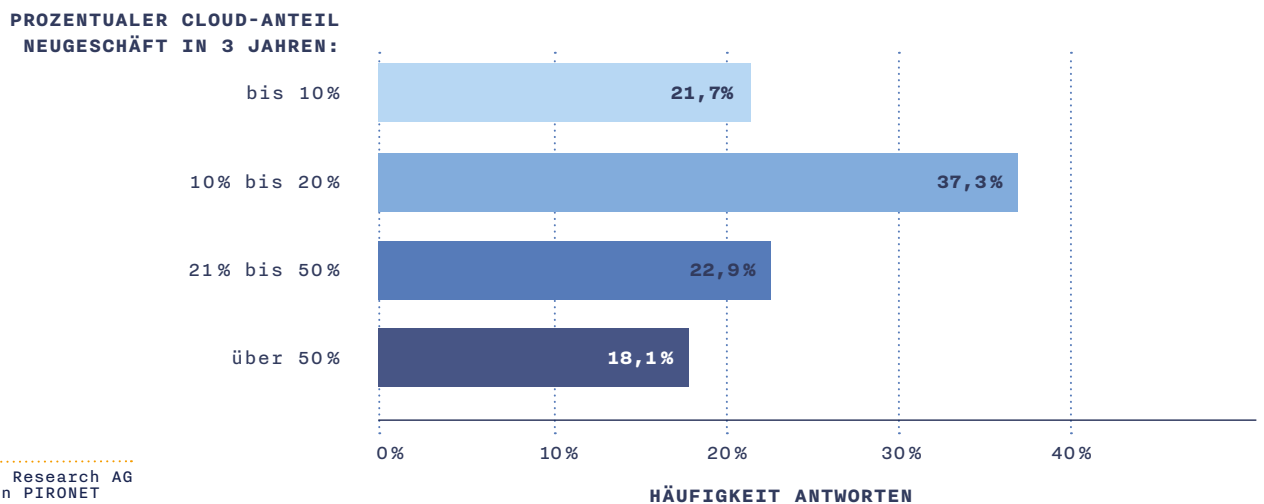
Allerdings befindet sich fast ein Drittel aller Befragten (33 Prozent) derzeit in der Evaluierung von PaaS-Plattformen. Weitere 32 Prozent nutzen solche Angebote nur punktuell und limitiert. Das heißt vor allem eines: Der Markt ist noch in großer Bewegung, da rund zwei Drittel der ISVs noch keine abschließende Entscheidung zu diesem Thema getroffen hat. Der Kampf um diese wichtige Zielgruppe hat also gerade erst begonnen und noch ist unklar wer als Sieger hervorgehen wird. Allerdings gibt diese Studie deutliche Hinweise, wo die Reise hingehen könnte, haben doch die meisten Software-Anbieter klare Vorstellungen von Deployment- Varianten und sonstigen Anforderungen.



Wie wichtig eine erfolgreiche Cloud-Transformation für die deutschen Softwarehäuser tatsächlich ist, lässt sich an dem geplanten Neugeschäft in diesem Bereich ablesen.

Fast jedes 5. Unternehmen (18 Prozent) plant in den nächsten 3 Jahren über 50 Prozent der Umsätze im Neugeschäft über cloud-basierte Modelle zu realisieren. 23 Prozent der Unternehmen planen einen Anteil am Neugeschäft in einer Spanne von 21 bis 50 Prozent über Cloud Computing zu realisieren. Dies verdeutlicht noch einmal die potenziellen Wachstumschancen, die sich aus diesem Modell ergeben, aber auch den extrem hohen

#### Wie hoch liegt der geplante Cloud-Anteil an ihrem Neugeschäft in 3 Jahren?

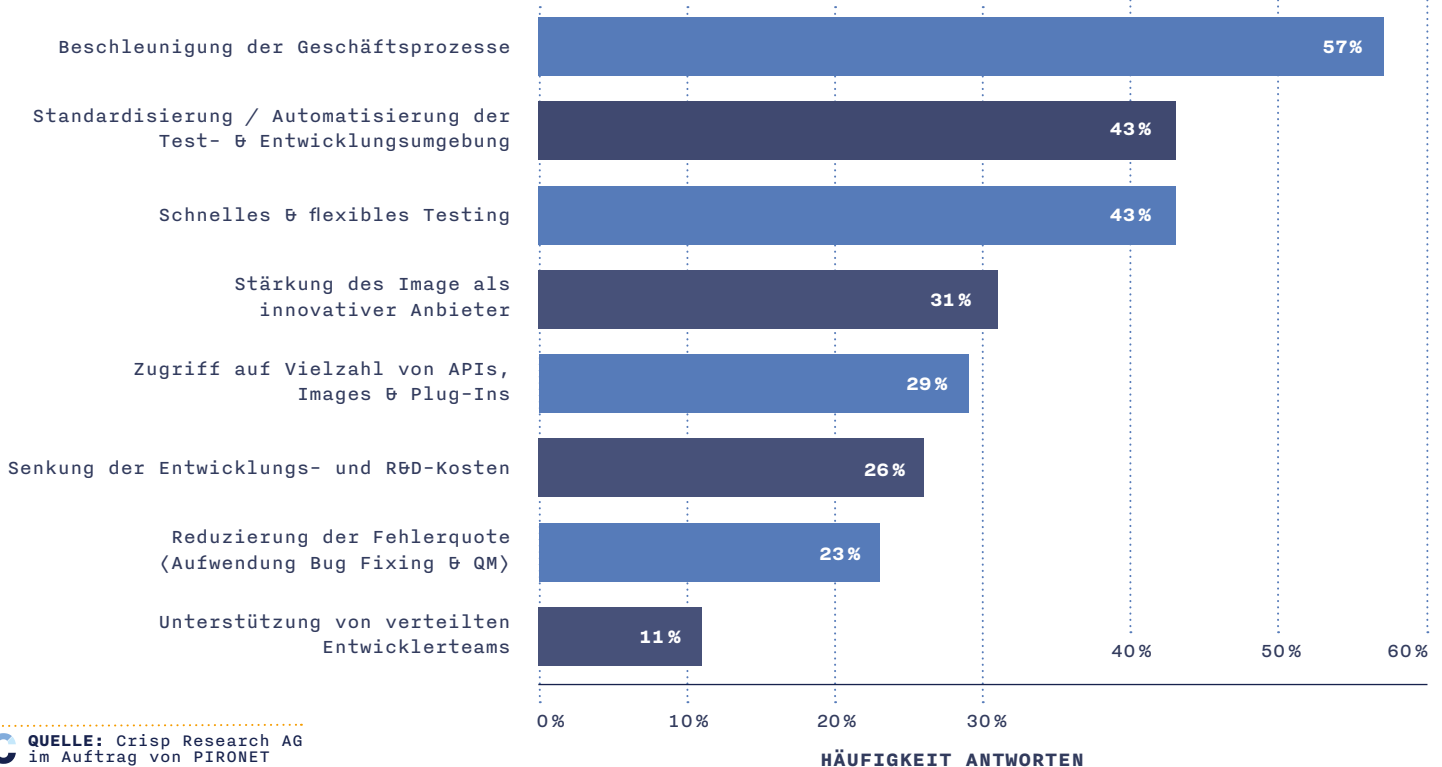


Erfolgsdruck unter dem die Softwarehäuser stehen. 37 Prozent der befragten Softwarehäuser sehen die Möglichkeit über SaaS-basierte Angebote kleinere Teile ihrer Umsätze (10 bis 20 Prozent) zu erzielen.

## 8.2. Gründe für und gegen Platform-as-a-Service

Diejenigen Softwarehäuser und ISVs, die bereits heute PaaS nutzen, sehen die Beschleunigung ihrer Geschäftsprozesse (57 Prozent) als den wichtigsten Beweggrund. Die Anforderungen an ein schnelles Going-Live, kürzere Innovationszyklen und agile Entwicklungsmethoden, wie z.B. Scrum, erhöhen kontinuierlich den Druck auf die Software-Industrie ihre Prozesse zu beschleunigen. Der Einsatz von PaaS erscheint hier ein probates Mittel.

### Welche sind die maßgeblichen Gründe für den Einsatz von PaaS- und Cloud-Plattformen?



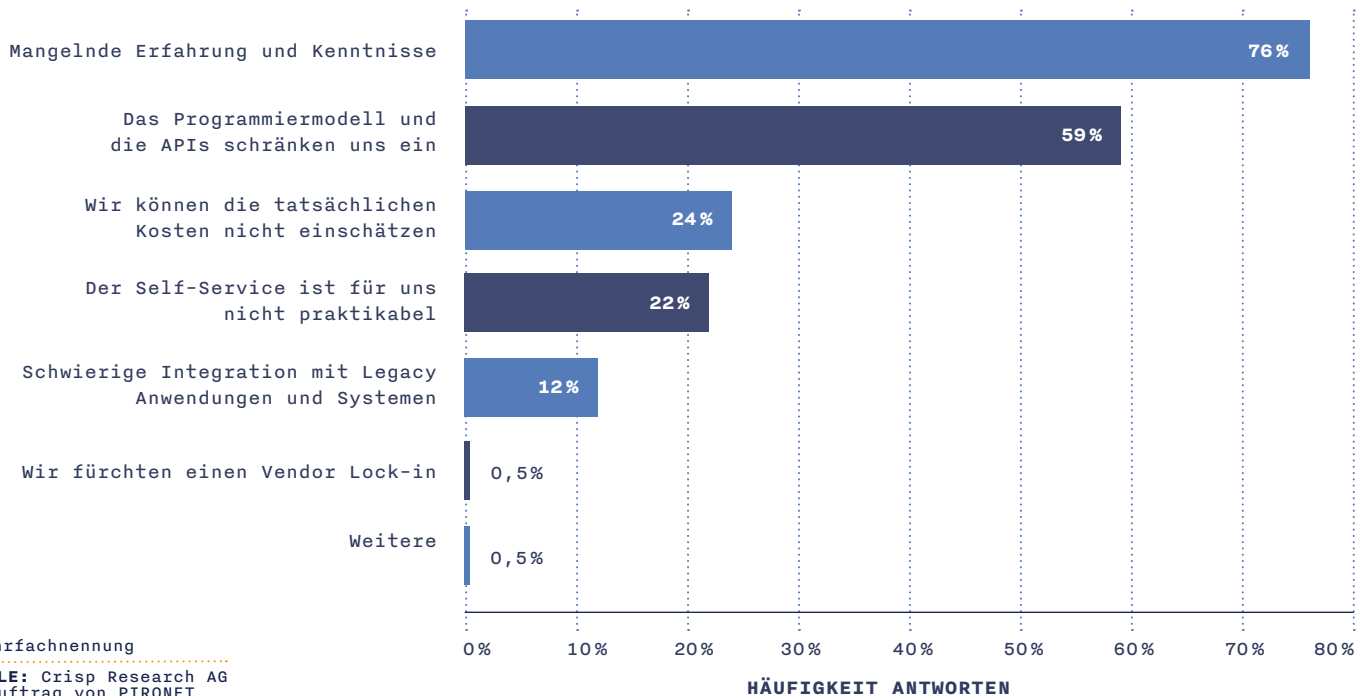
QUELLE: Crisp Research AG  
im Auftrag von PIRONET  
NDH, 2014

Dazu passt auch die Aussage von 43 Prozent der Befragten, die sich eine verbesserte Standardisierung und Automatisierung ihrer Test- und Entwicklungsprozesse erhoffen. Dies ist eine der zentralen Voraussetzungen zur Prozessbeschleunigung.

Auch zum Thema Testing gibt es hohe Erwartungen. So versprechen sich 43 Prozent der ISVs hier mehr Geschwindigkeit und Flexibilität. Ein Drittel der Befragten sieht durch den Einsatz von PaaS auch die Stärkung des Image als „innovativer Anbieter“.

Immerhin jeder 4. Befragte (26 Prozent) erhofft sich durch Platform-as-a-service eine Entlastung seiner Entwicklungs- und R&D-Budgets. Dass sich 23 Prozent der Befragten eine Reduzierung der Fehlerquoten erwarten, passt deshalb dann auch gut in das Gesamtbild, ist die Folge davon doch eine Senkung der Kosten.

Was sind die Gründe, weshalb sie keinen Platform-as-a-Service nutzen würden?



n41, Mehrfachnennung

QUELLE: Crisp Research AG  
im Auftrag von PIRONET  
NDH, 2014

Für die überwiegende Mehrheit der befragten ISVs (76 Prozent) liegt die Nichtnutzung von PaaS schlicht in mangelnder Erfahrung und nicht vorhandenen Kenntnissen begründet. Die Komplexität und Vielfalt der angebotenen Services ist groß und nicht alle Softwarehersteller verfügen über genügend Talente, die intuitiv mit solchen Plattformen umgehen können.

Die Einschränkungen durch das Programmiermodell und durch APIs sehen 58 Prozent als einen weiteren wichtigen Grund gegen den Einsatz von standardisierten PaaS-Plattformen. Aufgrund des noch jungen Themas fürchten die Softwarehersteller offenbar Fehler zu machen und sowohl einen zeitlichen als auch einen qualitativen Nachteil erleiden zu können. Die Risiken eines möglichen Vendor Lock-In-Schrecken die wenigsten. Für rund ein Fünftel (24 Prozent) sind nicht kalkulierbare Kosten ein Grund nicht auf PaaS-Angebote zu setzen.

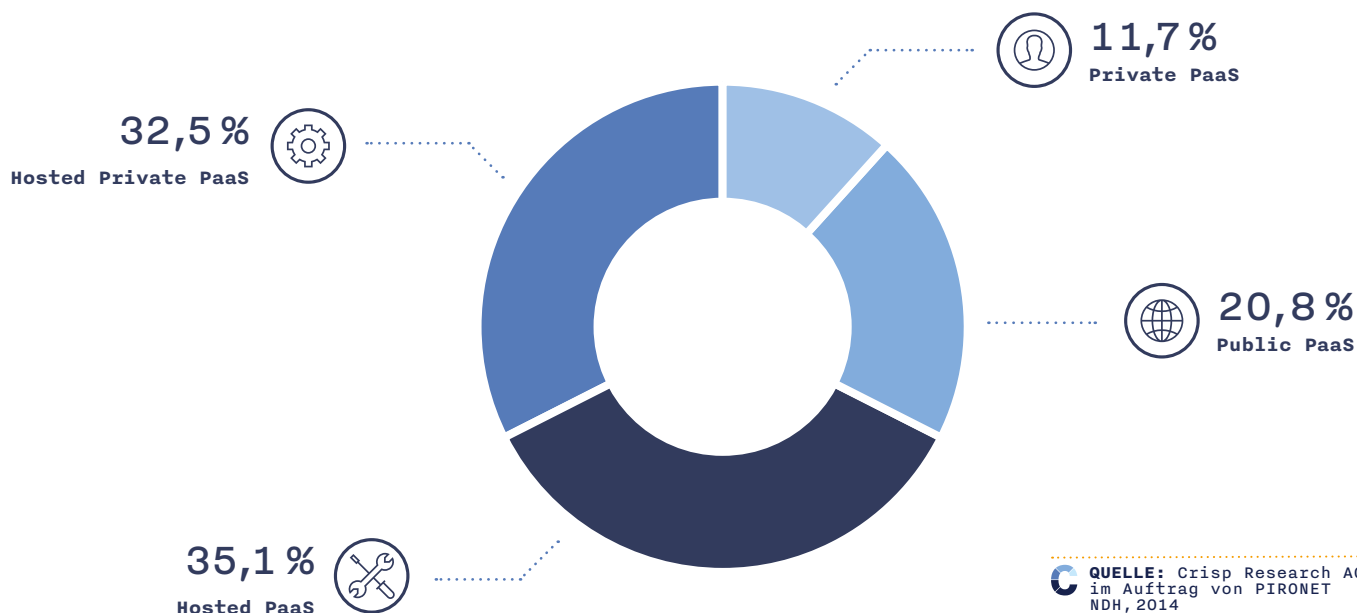
Die Studie hat auch gezeigt, dass in Zukunft ein Markt für „Managed PaaS“ entstehen wird, geben doch 22 Prozent der ISVs an, dass die Nutzung im Rahmen eines Self-Service Modells nicht praktikabel ist.

Abschließend bleibt festzustellen, dass viele der ablehnenden Gründe „heilbar“ sind, dies gilt insbesondere für die Themen Erfahrung und Kenntnisse, sowie für die Prognostizierbarkeit von Kosten.

### 8.3. Favorisierte Betriebskonzepte für Platform-as-a-Service

Die Aussagen, der im Rahmen der Studie befragten Softwarehäuser, zeichnen ein recht eindeutiges Bild. Gefragt nach dem favorisierten Betriebskonzept zur Nutzung von PaaS-Diensten im Rahmen der Entwicklungsprozesse, sprachen sich „nur“ 21 Prozent für das bisherige Public Cloud-Modell aus, während 12 Prozent sich für den internen Betrieb im Rahmen einer „Private PaaS-Plattform“ entscheiden würden. Die Mehrheit von deutlich über 60 Prozent würden PaaS-Dienste für Development & Test am ehesten im Rahmen eines Hosting-Modells beziehen.

Nach welchem Betriebskonzept würden Sie PaaS-Services am ehesten bei der Entwicklung nutzen?



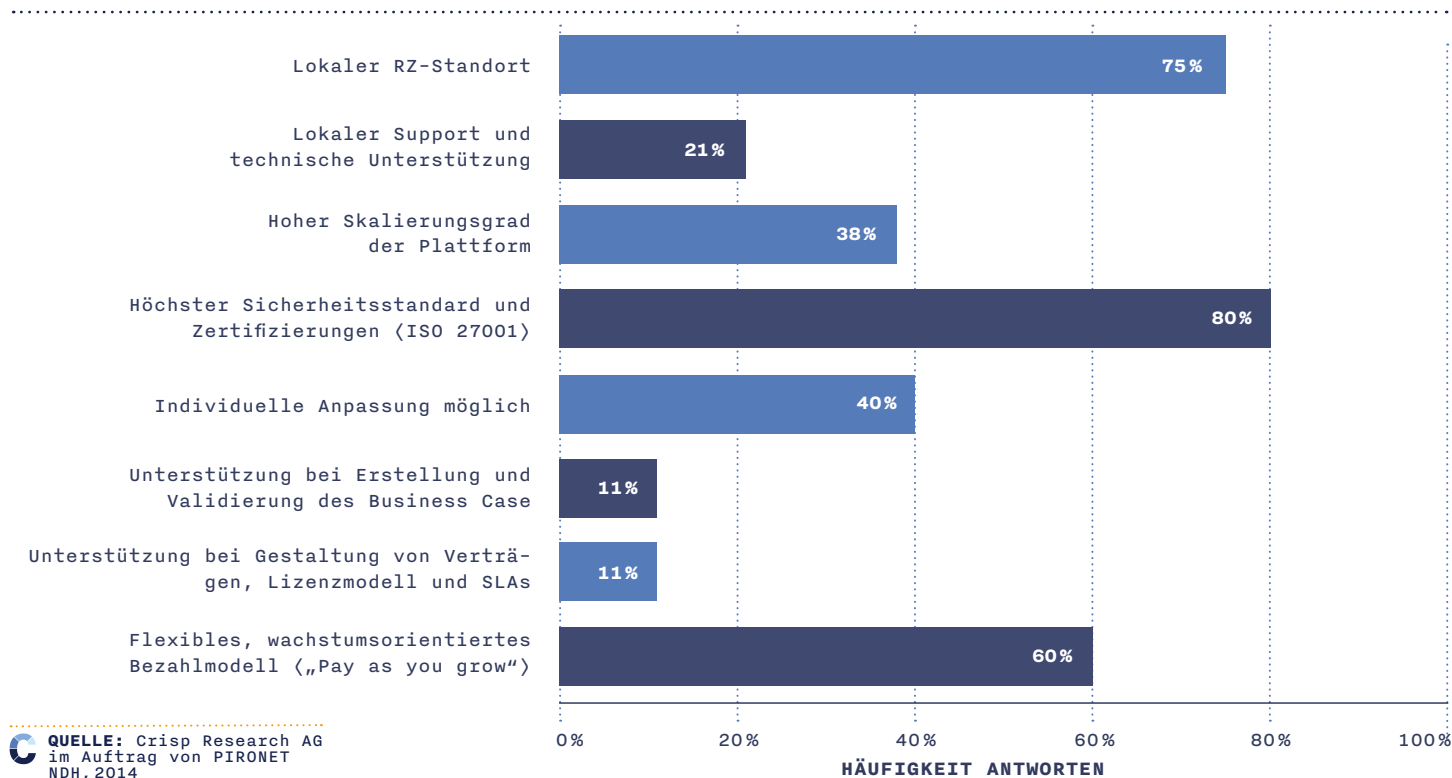
Hinsichtlich des Applikations-Betriebs sind die befragten Softwarehäuser sogar noch etwas anspruchsvoller. Hier sind es sogar nur noch 11 Prozent der befragten Unternehmen, die den Betrieb auf einer Public Cloud Umgebung befürworten würden. Die Mehrheit (38 Prozent) sieht in einem Hosted PaaS das für sie bevorzugte Betriebsmodell, um Applikationen in der Cloud zu betreiben. Weitere 30 Prozent sehen in der dedizierten Variante („Hosted Private PaaS“) ihr favorisiertes Modell. Über 1/5 der befragten Softwareunternehmen würden ihre Applikationen nur auf einer Private PaaS-Umgebung betreiben.

## 8.4. Anforderungen an die Platform-as-a-Service-Provider

Betrachtet man die Aussagen zu den zentralen Anforderungen an die Anbieter von PaaS-Diensten, so wird evident, warum die bisherigen Public Cloud-Modelle für viele professionelle Softwareentwickler nicht in Frage kamen. So stehen für rund 80 Prozent der befragten deutschen Softwarehäuser die Themen hohe Sicherheitsstandards, ISO-Zertifizierung und ein deutscher Rechenzentrumsstandort ganz oben auf der Agenda.

Ebenfalls relevant ist ein flexibles Bezahlmodell (60 Prozent). Interessanterweise ist die individuelle Anpassung der PaaS-Plattform für die deutschen ISVs genauso wichtig, wie die Skalierungsfähigkeit der Plattform – ein Argument, dass in vielen Debatten der letzten Jahre immer an erster Stelle angeführt wurde. Anscheinend ist hier die Erkenntnis eingetreten, dass die Skalierungsfähigkeit der Technologie-Plattform nur so weit von Nöten ist, wie sich das Wachstum auf Kunden- beziehungsweise Anwenderseite entwickelt.

Welche Kriterien muss ein potenzieller Cloud-Plattformanbieter erfüllen bzw. welche relevanten Services müssen bereitgestellt werden?



QUELLE: Crisp Research AG  
im Auftrag von PIRONET  
NDH, 2014

Es scheint verwunderlich, dass sich „nur“ 20 Prozent der ISVs einen lokalen Support und technische Unterstützung „vor Ort“ wünschen. Nur jeder 10. ISVs gab im Rahmen der Befragung an, Unterstützung bei der Ausgestaltung der Lizenzmodelle und SLA zu benötigen. Was vielleicht auch daran liegt, dass bislang erst wenige ISVs intensive Erfahrung mit dem Thema gesammelt haben – und daher noch etwas zu optimistisch in die Zukunft schauen.

## 9. Strategien für den erfolgreichen Einsatz

---

Die Einführung von PaaS-Plattformen in bestehende Entwicklungs- und IT-Betriebsprozesse ist eine vielschichtige Herausforderung. Denn mit dem erfolgreichen Einsatz von PaaS geht nicht nur die Nutzung von neuen Tools einher, sondern es verändern sich auch bestehende Prozesse und IT-Organisationsmodelle. Im Folgenden wird kurz skizziert, welche Aufgaben in den Bereichen Strategie, Kultur, Organisation und Prozesse notwendig sind, um PaaS in der eigenen IT-Organisation zu etablieren.

### 9.1. Strategie

---

Zu allererst sollten sich Unternehmen darüber klar werden, ob im eigenen Unternehmen der Bedarf für den Einsatz solcher neuer Plattformen sowie den damit einhergehenden Cloud-Entwicklungs- und Betriebsmodellen besteht. Unternehmen, die nur auf die Implementierung von Standardsoftware setzen und wenig Entwicklungs- und Integrationsbedarf haben, können gegebenenfalls auch auf PaaS verzichten. All diejenigen Unternehmen, deren Kunden und Partner in neue digitalisierte Prozesse eingebunden und mittels moderner Webanwendungen an das Unternehmen angebunden werden sollen, werden in den kommenden Jahren um ein Upgrade ihrer Anwendungsentwicklung und der IT-Betriebsprozesse wohl nicht herumkommen.

In der Evaluierungs- und Planungsphase können entsprechende Assessments helfen, die auf Basis des Status Quo den Transformationsbedarf und die Anforderungen an PaaS-Plattformen für das eigene Unternehmen definieren helfen.

Da sich die PaaS-Plattformen derzeit noch beträchtlich hinsichtlich Funktionalität, SLAs und Einsatzspektren unterscheiden, ist für große Unternehmen eine vorausschauende und detaillierte Evaluierung der Plattform durchaus angebracht.

Parallel empfiehlt Crisp Research die Entwickler und IT-Betriebsverantwortlichen möglichst frühzeitig in die Evaluierungs- und Planungsprozesse einzubinden und gegebenenfalls über Test- und Prototyping-Phasen aktiv zu beteiligen. Das Ausprobieren und Tüfteln kostet zwar Zeit, schützt aber davor für das Unternehmen ungeeignete Plattformen auszuwählen. Ein reiner Top-Down-Planungsansatz ist für PaaS-Projekte nicht empfehlenswert.

Neben der Business Case-Analyse, der Evaluierung der Anbieter und der Projektplanung, spielen die Konzeption des Betriebskonzeptes (Private, Public, Hosted etc.) sowie Integrationsfragen eine entscheidende Rolle.

Des Weiteren sind gegebenenfalls bereits zu diesem Zeitpunkt auch die notwendigen strategischen Überlegungen im Zusammenhang mit der späteren Bereitstellung und Vermarktung der künftigen Cloud-Lösung voranzutreiben. Gerade für kleinere Start-Ups oder für ISVs mit wenig Erfahrung in der Vermarktung Cloud-basierter Softwarelösungen und einem bislang eher traditionell geprägten Vertriebsansatz ist eine auf die neuen Anforderungen und Zielgruppen ausgerichtete und konsequent umgesetzte Go-To-Market Strategie oftmals der entscheidende Faktor für den auch kommerziellen Erfolg. Cloud -basierte Bereitstellungsmodelle erfordern darüber hinaus meist erhebliche Veränderungen in der Preisstrategie. Das vorherrschende Abrechnungsmodell in der Cloud, Kosten nutzenbezogen und monatlich wiederkehrend abzurechnen, bedeutet für Start-Ups und ISVs zudem, Investitionen, sprungfixe Kosten und laufende Kosten mit den typischerweise zunächst geringen, weil monatlichen Einnahmen, profitabel in Einklang zu bringen.

Die strategischen Aufgaben lassen sich wie folgt zusammenfassen:

- Analyse Use Case/Business Case
- Evaluierung Plattformen/Anbieter
- Proof of Concept/Testing
- Planung PaaS-Architektur und Betriebskonzept/SLAs
- Roadmap für Implementierung/Projektmanagement
- Planung Integrationsbedarf
- Verankerung PaaS in IT-Strategie
- Planung des SaaS Go-to-Market

## 9.2. Kultur

---

Ganz entscheidend für den erfolgreichen Einsatz von PaaS im Unternehmen ist der kulturelle Aspekt. Nur wenn Unternehmen die Arbeit von Software-Entwicklern wertschätzen und es schaffen, eine „Developer Culture“ bzw. eine Innovationskultur zu etablieren, lassen sich die mit PaaS verbundenen Konzepte wie DevOps und Agile Programming bzw. agiles Projektmanagement auch umsetzen. Der Trend, die interne Anwendungsentwicklung komplett an IT-Dienstleister auszulagern und sich vornehmlich auf den Einsatz von Standard-Software zu konzentrieren, hat in vielen Unternehmen die Fähigkeiten zur Inhouse-Entwicklung von Software verkümmern lassen.

Aus Perspektive von Crisp Research ist hier ein Umdenken dringend erforderlich, sofern die Unternehmen den Herausforderungen der digitalen Transformation ernsthaft entgegenzutreten wollen. Zumindest sollte sichergestellt sein, dass unternehmensintern genügend „Developer Talent“ zur Verfügung steht, um die neuen digitalen Anwendungen zu konzipieren, designen und deren Integration in die Unternehmens-IT auf Architektur-Ebene zu planen. An der finalen Entwicklung können natürlich externe Ressourcen mit beteiligt werden. Ideation, Prototyping, Konzeption und Architektur sollten allerdings in der Hand der Unternehmen selbst liegen.

Um diese „Developer Culture“ zu etablieren und im Unternehmen zu pflegen bedarf es verschiedener Initiativen. Developer Labs können (Frei-) Raum zum Testen und für das Prototyping bzw. PoCs bieten. Zudem sollte dem Thema User Experience bzw. Design Thinking mehr Aufmerksamkeit gewidmet werden.

Der Erfolg der neuen Generation an Anwendungen und Lösungen – egal ob Digital Marketing, mobile App oder Internet der Dinge – hängt elementar vom Nutzen und der Nutzerfreundlichkeit ab. Das Abhaken möglichst langer Feature-Listen kann und darf nicht mehr im Vordergrund der Anwendungsentwicklung stehen. Zusammengefasst:

- Developer Culture und Aufbau interner Entwickler-Kapazitäten
- Aktives IT-Innovation Management
- Labs und Freiräume für Prototyping/Testing schaffen
- User Experience/Design Thinking in der IT verankern

### 9.3. Organisation

---

Auch im Hinblick auf die IT-Organisation stehen Veränderungen an, sofern PaaS im Unternehmen verankert werden soll. Die teils starren Organisationsstrukturen im Infrastrukturbetrieb, die vielfach immer noch strikt nach Server, Storage und Netzwerk aufgestellt sind, werden durch die Bereitstellung von IaaS und PaaS über Cloud-Plattformen aufgebrochen bzw. müssen stärker integriert werden.

Der mit dem Stichwort „DevOps“ beschriebene Trend nimmt die stärkere Verzahnung von Anwendungsentwicklung, Testing und IT-Betrieb vorweg. In einer cloud-basierten Welt wird ständig an neuen Features gearbeitet und auch die Performance von Applikationen muss ständig verbessert und an neue Anforderungen (Endgeräte, Bandbreiten, Security-Standards etc.) angepasst werden. Fest terminierte jährliche Releases gehören der Vergangenheit an. Im Zeitalter der „Continuous Releases“



erfolgen Entwicklung, Testing, Staging und Release Management kontinuierlich und quasi im produktiven Betrieb. Hier gilt es entsprechende Organisationsmodelle und Teamstrukturen zu schaffen. Ebenso müssen Developer für die Implikationen ihrer Entscheidungen im Hinblick auf die Infrastruktur sensibilisiert werden. Sprich – wie können Anwendungen so programmiert werden, um möglichst effizient betrieben werden zu können („Lean Software Development - LSD“). Anders herum müssen die IT-Infrastruktur-Teams stärker auf die Bedürfnisse der Entwickler eingehen und genau wissen, welche Tools, Frameworks und APIs gerade gebraucht werden.

Gebraucht werden in diesem Kontext vor allem neue und gemeinsame KPIs. Solange Entwickler und IT-Betrieb auf unterschiedliche Kennzahlen incentiviert werden, kann das DevOps-Modell nicht erfolgreich sein. Hier können KPIs wie Nutzungsraten (Adoption Rate), Anwenderbewertungen und Application Performance als erste Schritte in Richtung DevOps dienen. Wichtig sind also:

- DevOps Teamstrukturen schaffen
  - Einsatz agiler Entwicklungsmethoden
  - Gemeinsame KPIs für Development und IT-Betrieb
  - Einbeziehung aller notwendigen Stakeholder
- 

## 9.4. Prozess

---

Sind auf Strategie- und Organisationsseite die Weichen richtig gestellt und erste Initiativen ergriffen, um den Mind-Set in Richtung „Developer Culture“ und IT-Innovation Management zu verändern, stehen noch einige knifflige Aufgaben an, die sich mit dem Einfluss von PaaS auf die internen Prozesse beschäftigen.

So sind beispielsweise Anpassungen im Bereich des Anforderungs- und Demand Managements von Nöten. Sind agile Entwicklungsmethoden und der Einsatz von PaaS geplant, ist es wichtig diese Prozesse stärker in Richtung „User-Centric Design“ zu entwickeln. Die Konzeption und Bewertung von Features und Anforderungen muss stärker aus der Perspektive der Endnutzer erfolgen und deren konkrete Einsatzszenarien vor Augen haben.

Auch beim Onboarding der Entwickler und den Entwicklungsprozessen ändert sich einiges. So erfolgt z.B. die Bereitstellung neuer Tools oder APIs über die PaaS-Plattform und nicht mehr wie bisher aus einem gesonderten Service Katalog oder Portal. Entwickler müssen sich mit der neuen Umgebung vertraut machen und ihre Entwicklungsprozesse der PaaS-Logik anpassen. Dementsprechend ändert sich auch

der Onboarding-Prozess sofern neue Entwickler in das jeweilige Projektteam aufgenommen werden (Rollen, Berechtigungen, Tools etc.).

Das initiale Sizing der benötigten Infrastruktur für Test- und später auch die Produktivsysteme übernehmen im PaaS-Modell in der Regel die Entwickler. Im laufenden Betrieb erfolgt das Managen von Verfügbarkeit und Kapazitäten dann über den IT-Betrieb. Um hier einen reibungslosen Übergang und eine Optimierung hinsichtlich Performance, Nutzbarkeit und Kosten im laufenden Betrieb zu erreichen, müssen entsprechende Monitoring-, Incident Management und Reporting-Prozesse aufgesetzt werden, die jeweils den IT-Infrastruktur-Teams sowie den Entwicklern gleichermaßen zur Verfügung stehen. Nur dann lässt sich ein integriertes Application Performance Management erfolgreich implementieren.

Auswirkungen hat PaaS auch auf die Prozesse des Policy und Security Managements. Einerseits stellt sich die Frage, ob die jeweilige PaaS-Plattform die vom Unternehmen geforderten Governance-Richtlinien funktional umsetzen kann. Andererseits gehen im PaaS-Modell aufgrund des Self-Service viele Verantwortlichkeiten auf die Entwickler über. Auch werden einige der ehemals manuellen Deployment- und Release Management-Aufgaben auf den Plattformen automatisiert. Daher ist es wichtig die neuen Verantwortlichkeiten zu erkennen und Regeln festzulegen, um z.B. Obergrenzen für die Bereitstellung von Infrastruktur zu definieren (max. 200 Compute-Units pro Entwickler).

Folgende ausgewählte Prozesse werden durch den PaaS-Einsatz beeinflusst:

- Requirements Engineering & Demand Management
- Development & Developer Tooling
- Testing & Test Automation
- Deployment & Release Management
- Availability & Capacity Management/IT-Operations (SLAs)
- Reporting, Metering & Chargeback
- Application Lifecycle Management
- Application Support
- Policy Management
- IT Security Management

## 10. Ausblick

---

Für die kommenden 2-3 Jahre sieht Crisp Research im Hinblick auf die Entwicklung der PaaS-Plattformen sowie dem Einsatz auf Seiten europäischer Unternehmen und ISVs folgende Trends:

### Hybrid und Private PaaS etabliert sich neben den Public Plattformen

Public PaaS-Plattformen, wie Salesforce, Heroku oder Microsoft Azure Websites haben sich in den vergangenen Jahren am Markt etabliert und bieten Entwicklern skalierbare und robuste Alternativen für die Entwicklung ihrer Web-Applikationen oder als Backend von mobilen Anwendungen. Unternehmensanwender haben allerdings meist individuelle Anforderungen hinsichtlich Skalierbarkeit, SLAs und Rechenzentrumsstandort. Dies gilt auch bei der Auswahl eines PaaS-Anbieters. Hinzu kommt, dass die großen und mittelständischen Unternehmen bevorzugt hybride bzw. Multi-Cloud-Modelle implementieren, um unterschiedliche Anwendungsszenarien umzusetzen und das Risiko zu verteilen.

### Cloud Foundry und Microsoft Azure als die Schlüsseltechnologien. OpenShift als weitere Alternative.

Von den unterschiedlichen PaaS-Stacks haben sich mittlerweile 3 führende Technologien herauskristallisiert. Hierzu gehören das von VMware/Pivotal entwickelte und unter Open Source veröffentlichte Cloud Foundry, OpenShift von Red Hat sowie das Microsoft Azure Pack, das sich sowohl für Enterprise Private PaaS als auch PaaS-Plattformen für Webhoster und Managed Service Anbieter einsetzen lässt.

### Lokale IT-Dienstleister etablieren eigene PaaS-Angebote

Dem Trend nach lokaler Datenhaltung und Kundennähe folgend, bieten nun immer mehr IT-Dienstleister PaaS-Services auf Basis der Technologiestacks (Cloud Foundry, OpenShift, Microsoft Azure Pack) aus den eigenen, lokalen Rechenzentrumsstandorten an.

Somit können Anwender von den Vorteilen einer PaaS-Plattform profitieren und gleichzeitig ihre Anforderungen hinsichtlich Datenschutz, Sicherheit und individuell definierter Performance erfüllen.

### Entwickler als neue Kernzielgruppe

Nicht nur das Zitat von Marc Andreessen „Software is eating the world“ deutet darauf hin, dass Entwickler die Zukunft der IT-Industrie maßgeblich mitbestimmen und wohl auch einen großen Einfluss auf die Ausgestaltung der digitalen Wirtschaft im Allgemeinen haben werden. Das haben auch die großen IT-Firmen erkannt. Deren traditionelle Partner (Integratoren, Systemhäuser, ISVs) haben es in den vergangenen Jahren nicht geschafft auf Basis der neuen Cloud-Plattformen genügend innovative und neue Anwendungen und Lösungen zu entwickeln. Nun sollen Startups diese Lücke füllen und sich zu einer Kernzielgruppe von Microsoft, HP, Cisco oder SAP entwickeln. Diese sind aufgrund von Unternehmensstruktur, Innovations- und Wachstumskultur sowie der Kenntnisse der neuen Programmiersprachen und Tools auch eine prädestinierte Anwendergruppe.<sup>12</sup>

### Mobile Workloads und Internet of Things als wesentliche Treiber

Was sich vor 2 Jahren noch als Phänomen für wenige ausgesuchte Anwender darstellte, ist heute Standard – die mobile Nutzung von Unternehmensanwendungen auf Tablets, Smartphones und anderen mobilen Devices. Softwareanbieter müssen seit kurzem in der Lage sein, ihre Lösungen und Services so zu entwickeln, dass sie auf einer Vielzahl von Endgeräten, Browsern und Betriebssystemen fehlerfrei lauffähig sind. Dies wirft nicht nur hinsichtlich Entwicklung und Test, sondern auch in Bezug auf den Betrieb der Anwendungen echte Herausforderungen für die Anbieter auf.

Selbiges geht von dem neuen Trend das „Internet of Things“ (IoT) aus. Die Entwicklung von Sensornetzwerken schreitet stetig voran, um Haushaltsgegenstände, Autos, Industrieanlagen, Wearables usw. untereinander zu vernetzen. Dadurch wird in den kommenden Jahren der Bedarf an weiteren Ressourcen für Entwickler steigen, um Applikationen für eine Vielzahl an neuen Geräten zu entwickeln und diese miteinander zu integrieren. Zahlen der Evans Data Corporation<sup>13</sup> zeigen, dass 17 Prozent der Entwickler bereits aktiv an IoT-Applikationen arbeiten. 23 Prozent erwarten Projektaufträge in diesem Kontext.

### Container-Technologie als Grundlage für portable Anwendungen

Anwendungen flexibel über mehrere unterschiedliche Cloud-Umgebungen bereitzustellen, sorgt für weniger Bedenken

---

<sup>12</sup> <http://www.crisp-research.com/deutschland-2014-startup-katastrophe-oder-neue-it-grunderzeit-teil-1/>

<sup>13</sup> <http://www.zdnet.com/thing-1-thing-2-internet-of-things-starts-to-gain-a-developer-following-7000032011/>

hinsichtlich der langfristigen Nutzung der Cloud. Mit Docker steht hierfür seit dem Jahr 2013 eine Technologie bzw. ein Service zur Verfügung, mit dem diese Art von Deployment-Mechanismen möglich werden. Docker hat die Idee des Container-Transports in die Softwareentwicklung überführt und stellt eine Plattform für das bereitstellen von verteilten Anwendungen zur Verfügung. Docker bietet hierzu eine standardisierte Abstraktionsschicht, mit der Applikationen und ihre Abhängigkeiten in einen virtuellen Container verpackt werden, die anschließend auf jedem beliebigen Linux Server ausgeführt werden können. Entwickler können ihre Anwendungen somit flexibel zwischen verschiedenen Umgebungen portieren.<sup>14</sup>

### **Collaborative Development durch Verbreitung Browser-basierter IDEs**

Seit jeher besteht das Ziel darin, die Qualität von Software zu verbessern. Mit der bekannten agilen Softwareentwicklungsmethode „Pair Programming“ existiert ein Konzept, mit welchem sich die Fehlerrate um bis zu 15 Prozent reduzieren lässt. Hierzu arbeiten zwei Programmierer parallel an einem Rechner.

Während einer den Quellcode schreibt, ist der andere für die Problemlösung zuständig und kontrolliert den Programmcode.

Mit der Verbreitung von Cloud-IDEs, sprich Browser-basierter Entwicklungsumgebungen, wird sich diese Entwicklungsmethode technisch einfacher umsetzen lassen. Unabhängig vom Standort können mehrere Entwickler parallel am selben Quellcode arbeiten und gleichzeitig gegenseitig bei ihrer Arbeit überprüfen. Dies führt zu einem echten „Collaborative Development“ und steigert die Produktivität, Qualität und gleichzeitig den Time-to-Market der Software.

### **Managed Services und professioneller Support als Erfolgsfaktor bei der PaaS-Transformation**

Die PaaS-Transformation ist für die Enterprise-IT als auch für die ISVs mit unterschiedlichen Herausforderungen verbunden. Grundsätzlich muss eine PaaS-Plattform als technische Grundlage verstanden werden, auf der Unternehmen und Anbieter aus der Softwareindustrie ihre digitale Transformation umsetzen können. Auf dem Weg dieser Metamorphose kann ein professioneller Support durch einen Managed PaaS-Anbieter während des Go-to-Market unterstützen und somit den Time-to-Market verkürzen.

---

<sup>14</sup> <http://www.crisp-research.com/docker-container-die-zukunft-moderner-applikationen-und-multi-cloud-deployments/>

### **Entstehung nationaler und internationaler Marktplätze für PaaS-Produkte**

Ähnlich der Entwicklung von Software-as-a-Service-Marktplätzen werden kurz- und mittelfristig auch solche für PaaS-Produkte an Bedeutung gewinnen. Als exemplarisches Beispiel mit Vorbildcharakter sei hier die Initiative „PaaSport“ genannt. Dieses EU-Projekt treibt der Bundesverband IT-Mittelstand e.V. (BITMi) als Projektkoordinator voran. Ein Projektkonsortium kombiniert europaweit Kompetenz und Ressourcen aus IT-Verbänden, Forschungspartnern sowie kleinen und mittelständischen IT-Unternehmen. Ziel ist die Erschaffung eines Marktplatzes, der es Entwicklern und Anbietern von Software-as-a-Service-Lösungen ermöglicht, PaaS-Angebote zu vergleichen. Besonderen Wert wird hierbei auf die Interoperabilität der PaaS-Lösungen gelegt. PaaSport wird sich auf den Cloud-Standard CAMP stützen. Weitere Informationen zum BITMi und zum Projekt PaaSport sind im Anhang zu finden.

# Über PIRONET NDH

---

Pironet NDH gehört zu den marktführenden deutschen Anbietern von Cloud Computing. Das börsennotierte Unternehmen mit Hauptsitz in Köln wurde 1995 gegründet und gehört seit dem Jahr 2014 mehrheitlich zum Konzernverbund der CANCOM SE in München. Im Fokus steht die Bereitstellung von Unternehmenssoftware über das Internet bis hin zu kompletten Desktop-Arbeitsplätzen und geschäftskritischen Applikationen von SAP und Microsoft.

Mit der deutschen Business Cloud® von Pironet NDH erschließen sich Unternehmen alle Optionen für die Einführung und Nutzung moderner Cloud Computing Lösungen. So stellt Pironet NDH virtuelle IT-Ressourcen und gängige Business-Anwendungen als schlüsselfertige und kostenflexible Cloud-Dienste (Software as a Service) aus seinen hochsicheren, deutschen Rechenzentren bereit. Sowohl Leistungsangebot und IT-Infrastruktur als auch die interne Organisation sind nach strengen, international anerkannten Richtlinien zertifiziert, u.a. nach DIN ISO/IEC 27001.

PIRONET NDH bietet mit einer eigenen PaaS- Entwicklungsplattform für web-basierte Applikationen eine umfangreiche und leistungsstarke, technische Basis für Ihren Cloud Computing-Erfolg. In der Zusammenarbeit mit PIRONET NDH wird künftigen PaaS-Nutzern alles Wesentliche für einen schnellen und unkomplizierten Start schlüsselfertig bereitgestellt: ein ganzheitliches Lösungsspektrum aus bereits im Rechenzentrum der PIRONET NDH implementierten PaaS-Werkzeugen sowie vielfältigen Support- und Serviceleistungen. Das bedeutet mehr Flexibilität und höhere Geschwindigkeit in der Applikationsentwicklung.

Informationen dazu unter <http://pironet-ndh.com/CloudEnabling>.



## PIRONET NDH

Ein **CANCOM** Unternehmen.  
PIRONET NDH Datacenter AG & Co. KG  
Von-der-Wettern-Straße 27  
D - 51149 Köln-Porz

**Telefon:** +49-(0)2203 / 935 30 - 0  
**Telefax:** +49-(0)2203 / 935 30 - 99  
**Email:** [info@pironet-ndh.com](mailto:info@pironet-ndh.com)

# Über BITMi e.V.

---

Der Bundesverband IT-Mittelstand e.V. (BITMi) ist der einzige IT-Fachverband, der ausschließlich mittelständische Interessen profiliert vertritt. Die Mitglieder sind mittelständische IT-Unternehmen aus allen Bereichen der IT-Branche, die überwiegend inhabergeführt sind. Im Verband kommen sowohl etablierte IT-Unternehmen als auch Start-Ups zu einer gemeinsamen Interessensfindung zusammen. Zudem haben sich neben den direkten Mitgliedern unter dem Dach des BITMi auch mit uns assoziierte Verbände zusammengeschlossen. Der Verband repräsentiert damit die Interessen von mehr als 1000 mittelständischen IT-Unternehmen in Deutschland mit einem Umsatzvolumen von mehreren Milliarden Euro.

Es ist das Ziel des BITMi durch intensive Netzwerkbildung das Unternehmenswachstum und die Produktivität der IT-KMUs zu beschleunigen und die Marktentwicklung voranzutreiben. Dabei ist es dem Verband besonders wichtig, den Standort Deutschland zu stärken. Deswegen kooperiert der BITMi mit dem Bundeswirtschaftsministerium und verschafft dem IT-Mittelstand dort eine eigene Stimme. Über die Mitgliedschaft bei PIN-SME – dem einzigen europäischen Verband, der ausschließlich IT-KMUs vertritt – findet die Meinung des BITMi auch auf europäischer Ebene Gehör. Perspektiven, Stärken, Bedeutung, Chancen und Herausforderungen des IT-Mittelstands werden in der Politik und der Öffentlichkeit kommuniziert.

Stellvertretend für seine Mitglieder ist der Bundesverband IT-Mittelstand Kontaktstelle für Endkunden, die sich auf den neuesten Stand der Informationstechnologie bringen wollen oder nach Lösungen für ihre Probleme suchen. Der Verband organisiert Veranstaltungen, kultiviert ein weitreichendes Partnernetzwerk, realisiert erfolgreiche Public-Relations-Kampagnen und organisiert den Wissenstransfer mit Universitäten und Forschungseinrichtungen.



Bundesverband  
IT-Mittelstand e.V.

**Bundesverband IT-Mittelstand e.V.**

Augustastrasse 78-80  
52070 Aachen

Telefon: +49 241 1890 558

Telefax: +49 241 1890 555

Email: [kontakt@bitmi.de](mailto:kontakt@bitmi.de)

Internet: [www.bitmi.de](http://www.bitmi.de)



# Overview Projekt PaaSport

---

## *Introduction and Motivation*

*According to market researchers the PaaS market is predicted to reach \$20.1 billion in 2014, more or less tripling its size not only in terms of revenues but also in terms vendors active in the sector. Although giant vendors occupy this emerging space, including Microsoft, Amazon, Google, and Salesforce.com, many startups try to enter the market.*

*But the prospects for new startups and SMEs entering the PaaS market are not that positive and the barriers can often be humongous. Giant vendors, such as Amazon and Google, have actually gained the lion's share of the market and have evolved into de-facto standards. Interestingly the vast majority of the Cloud providers are of non-EU origin (most of them are US-based). The battle for dominance between the big vendors on who will take the lead in the market makes them reluctant to agree on widely accepted standards promoting their own, mutually incompatible Cloud standards and formats (Gagliardi and Muscella, 2010). This dominance increases the lock-in of customers in a single Cloud platform, preventing the portability of data or software created by them. But even if in theory application and/or data portability is supported, the high complexity and in most cases the additional switching costs discourage users from doing so (McKendrick, 2010). The high effort required for exporting ones application and data from a Cloud platform also discourages start-ups and SMEs from entering and bestirring in the flourishing Cloud market.*

*Our discussion so far proves that Cloud computing solutions have not been built with interoperability in mind leading to a closed and hard-to-enter Cloud market (Rimal et al., 2011). Interoperability remains one of the greatest challenges that the longed-term adoption of Cloud computing is facing. During the next five years the leading enterprise software vendors as well as the large Cloud specialists will introduce new PaaS offerings while both large and small Cloud PaaS providers will grow through partnerships (Gartner, 2010). The formation of partnerships and federations between heterogeneous Cloud PaaS Providers involves interoperability.*

*The Cloud community and the EC have realized the significance of interoperability and has initiated several approaches to tackle this challenging issue. The first efforts to explore interoperability in PaaS are also well on track, e.g. CAMP and Cloud4SOA. A recent study conducted on behalf of the EC recommends that:*

→ The EC should promote common standards and interoperability of public cloud systems, to maximize economies of scale across the EU and create the preconditions for portability between cloud vendors.

→ The EC should create the pre-conditions so that the principle of data access and portability between cloud vendors is widely accepted and the risk of lock-in of users in proprietary systems is prevented.

*We believe that building upon and further extending such initiatives can provide us the means for delivering value to European Cloud PaaS vendors (existing and potential ones) and software SMEs and raise the barriers that prevent them from entering the dynamic and evolving but yet monopolized PaaS market.*

*Companies developing applications should be able to choose between different Cloud PaaS offerings, e.g. selecting the most reliable, the most well-reputed, the most cost-efficient or simply the one that meets their technical requirements, and should also be able to switch easily and transparently between Cloud providers whenever needed, e.g. when an SLA is breached or when the cost is too high, without setting data and applications at risk, e.g. loss of data or inability to run the application on a different platform. Moreover, they should be able to compare Cloud offerings with different characteristics, such as resource, pricing or Quality of Service (QoS) model, and to choose the one that best matches their computing needs of their services and applications (Borenstein and Blake, 2011).*

*An open market of interoperable Cloud platforms will enable the IT industry to small-medium European Cloud providers and strengthen their market position (Foster et al., 2008). They will then interoperate and cooperate creating new business models according to demand without conflicts due to interoperability problems. For example, an unexpected increase in processing power capacity could force Cloud providers to cooperate in order to overcome the problem of limited resources. Otherwise, SMEs would seem unreliable to provide the negotiated QoS, leading consumers to rely on big players for hosting their services and data (Gartner, 2010). Therefore, initiatives are needed to ensure that the IT industry does not remain dominated by monopolies of giant multinationals and to support competition in the Cloud market.*

### ***PaaSport Vision and Key Technological Concepts***

*With this problem formulation, the ultimate goal for the PaaSport project is to resolve the application and data portability issues that exist in the Cloud PaaS market through a flexible and efficient deployment and migration approach. These include, but are not limited to: image conversion to be suitable for target hypervisor, compression to aid, speed of transfer, image encryption, secure*

*protocols, QoS guarantees, trust issues and cost sharing models.*

*To this end, PaaSPort will combine Cloud PaaS technologies with lightweight semantics in order to specify and deliver a thin, non-intrusive Cloud-broker (in the form of a Cloud PaaS Marketplace), to implement the enabling tools and technologies, and to deploy fully operational prototypes and large-scale demonstrators.*

*Hence, PaaSPort will enable European Cloud vendors (in particular SMEs) to roll out semantically interoperable PaaS offerings leveraging their competitive advantage, the quality of service and value delivered to their customers, and improving their outreach to potential customers, particularly the pan-European software industry. PaaSPort will also enable European Software SMEs to deploy business applications on the best-matching Cloud PaaS and to seamlessly migrate these applications on demand. PaaSPort contributes to aligning and interconnecting heterogeneous PaaS offerings, overcoming the vendor lock-in problem and lowering switching costs.*

*In particular, the PaaSPort Marketplace will enable European Software SMEs, such as the members of the Software SME Associations participating in the consortium, to seamlessly deploy and migrate their applications to the best-matching Cloud PaaS offering. From a business perspective, PaaSPort aims at lifting the barriers that impede exporting ones applications and data from a Cloud platform, thus empowering Cloud customers and Cloud application developers (in particular, European SME Software vendors) and allowing them to choose freely the Cloud PaaS offering that best fits their needs. PaaSPort is also expected to empower the position and encourage the entrance of European SME Cloud vendors, such as Flexiant, in an emerging market which is currently dominated by American colossi, such as Google and Amazon.*

*From a scientific and technical perspective, relevant issues and discussions in regards to Cloud PaaS interoperability are becoming lively, leaving room for the PaaSPort consortium to work towards the specification and implementation of a semantically-enhanced Cloud-broker architecture and marketplace. In order to maximize the communities awareness and adoption of the PaaSPort results, the PaaSPort partners (mainly the participating Software SME Associations) intend to design and implement a focused set of European-wide dissemination activities.*

For further information on BITMi's  
Projekt PaaSPort visit  
<http://paasport-project.eu>

# Autoren

---



## René Büst

Senior Analyst & Cloud Practice Lead

[rene.buest@crisp-research.com](mailto:rene.buest@crisp-research.com)

René Büst ist Senior Analyst und Cloud Practice Lead bei Crisp Research mit dem Fokus auf Cloud Computing, IT-Infrastrukturen, Open Source und Internet of Things. Er ist Mitglied des weltweiten Gigaom Research Analyst Network, Top Cloud Computing Blogger in Deutschland und gehört weltweit zu den Top 50 Bloggern in diesem Bereich. Darüber hinaus zählt er zu den weltweiten Top Cloud Computing Influencers und den Top 100 Cloud Computing Experten auf Twitter und Google+. Seit über 16 Jahren konzentriert er sich auf den strategischen Einsatz der Informationstechnologie in Unternehmen und setzt sich zudem mit dem IT-Einfluss auf unsere Gesellschaft sowie disruptiven Technologien auseinander.

René Büst ist Autor zahlreicher Cloud Computing und Technologie Fachartikel, Referent sowie Teilnehmer in Expertenrunden. Auf [CloudUser.de](http://CloudUser.de) schreibt er über die Themen Cloud Computing, IT-Infrastrukturen, Technologien, Management und Strategien und ist auf Twitter mit dem Benutzer [@ReneBuest](https://twitter.com/ReneBuest) unterwegs. Er ist ausgebildeter Fachinformatiker Systemintegration und hat einen Abschluss als Dipl.-Informatiker (FH) in Technische Informatik von der Hochschule Bremen sowie einen M.Sc. in IT-Management und Information Systems von der FHDW Paderborn.

# Autoren

---



**Dr. Carlo Velten**

Vorstandsvorsitzender & Senior Analyst

[carlo.velten@crisp-research.com](mailto:carlo.velten@crisp-research.com)

Dr. Carlo Velten ist Managing Director des IT-Research- und Beratungsunternehmens Crisp Research AG. Seit über 15 Jahren berät Carlo Velten als IT-Analyst namhafte Technologieunternehmen in Marketing- und Strategiefragen. Seine Schwerpunktthemen sind Cloud Strategy & Economics, Data Center Innovation und Digital Business Transformation. Zuvor leitete er 8 Jahre lang gemeinsam mit Steve Janata bei der Experton Group die „Cloud Computing & Innovation Practice“ und war Initiator des „Cloud Vendor Benchmark“. Davor war Carlo Velten verantwortlicher Senior Analyst bei der TechConsult und dort für die Themen Open Source und Web Computing verantwortlich.

Dr. Carlo Velten ist Jurymitglied bei den „Best-in-Cloud-Awards“ und engagiert sich im Branchenverband BITKOM. Als Key-Note-Speaker und Moderator tritt er seit vielen Jahren auf Konferenzen und Events auf. Er schreibt als Contributing Editor auf Computerwoche.de und unterstützt als Business Angel junge Startups. Zudem ist er politisch als Vorstand des Managerkreises der Friedrich Ebert Stiftung aktiv.

# Über die Crisp Research AG

---

Crisp Research ist ein europäisches IT-Research- und Beratungsunternehmen. Mit einem Team erfahrener Analysten, Berater und Software-Entwickler bewertet Crisp Research aktuelle und kommende Technologie- und Markttrends. Crisp Research unterstützt Unternehmen bei der digitalen Transformation ihrer IT- und Geschäftsprozesse. Cloud Computing und Digital Business Transformation sind die Themenschwerpunkte von Crisp Research.

Wir verfügen in unseren Crisp Labs über ein innovatives Team an Software-Entwicklern und -Architekten, um neue Cloud-Services und Technologien unter Live-Bedingungen zu evaluieren.



Weißenburgstraße 10  
D-34117 Kassel

TEL +495612207 – 4080

FAX +495612207 – 4081

MAIL [info@crisp-research.com](mailto:info@crisp-research.com)

WEB [crisp-research.com](http://crisp-research.com)

TWITTER [twitter.com/crisp\\_research](https://twitter.com/crisp_research)

# Copyright

---

## Erstellt im Auftrag von:

PIRONET NDH Datacenter AG & Co. KG  
Von-der-Wettern-Straße 27  
D - 51149 Köln-Porz

**Telefon:** +49-(0)2203 / 935 30 - 0

**Telefax:** +49-(0)2203 / 935 30 - 99

**Email:** [info@pironet-ndh.com](mailto:info@pironet-ndh.com)

---

Alle Rechte an den vorliegenden Inhalten liegen bei der Crisp Research AG. Die Daten und Informationen bleiben Eigentum der Crisp Research AG. Vervielfältigungen, auch auszugsweise, bedürfen der schriftlichen Genehmigung der Crisp Research AG.

## Gestaltung, Layout & Infografiken:

Hellwig & Buntenbruch

MAIL [info@hellundbunt.de](mailto:info@hellundbunt.de)

WEB [hellundbunt.de](http://hellundbunt.de)

Titelmotiv: [startupstockphotos.com](http://startupstockphotos.com)

Weißenburgstraße 10

D-34117 Kassel

**TEL** +49 561 2207 – 4080

**FAX** +49 561 2207 – 4081

**MAIL** [info@crisp-research.com](mailto:info@crisp-research.com)

**WEB** [crisp-research.com](http://crisp-research.com)

**TWITTER** [twitter.com/crisp\\_research](https://twitter.com/crisp_research)

