



# Cloud Computing

Vortragender: Björn Böing  
FH Bielefeld Campus Minden - Sommersemester 2018



# Agenda

## Grundlagen & weiterführende Konzepte

- Charakteristika
- Servicemodelle
  - IaaS
  - PaaS
  - SaaS
  - FaaS
- Bereitstellungsmodelle
- Edge & Fog Computing
- Einschränkungen & Hinweise

## Amazon Web Services (AWS)

- Grundlagen
- Identity and Access Management (IAM)
- EC2 & S3
- AWS Lambda

## Cloud Design Patterns

- Grundlagen
- Cache-aside
- Sharding

## Fragen

---

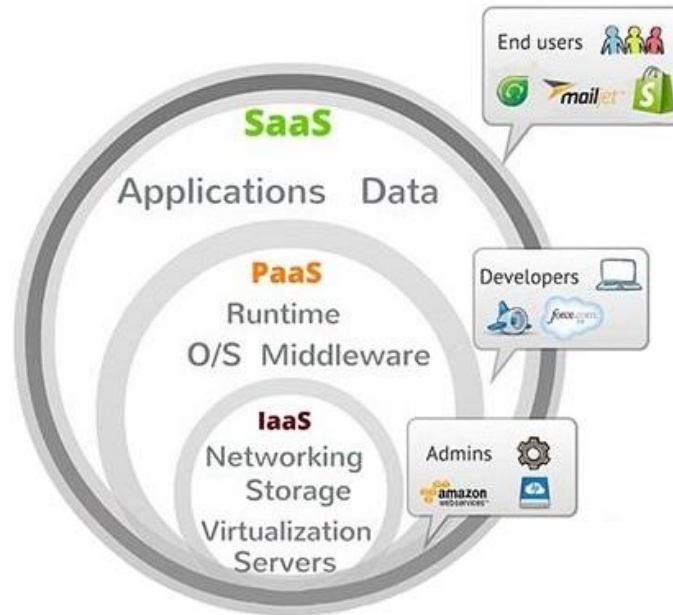
# Grundlagen & weiterführende Konzepte



# Charakteristika

- **On-demand self-service**  
Ressourcen können ohne menschliche Interaktion verwaltet werden
- **Broad network access**  
Dienste sind über Netzwerk und standardisiert erreichbar
- **Resource pooling**  
Anbieter kann mit derselben Ressource mehrere Kunden bedienen (mandantenfähigkeit)
- **Rapid elasticity**  
Funktionen können elastisch skaliert werden
- **Measured service**  
Die Nutzung von Diensten wird automatisiert überwacht und kann dementsprechend abgerechnet werden

# Service modelle





# Infrastructure as a Service (IaaS)

- Dienste zur Bereitstellung von IT-Infrastrukturen
- Typischerweise Rechenleistung und Speicherplatz
- Hypervisor werden genutzt, um mehrere VMs auf einem physischen System zu verwalten
- Cloud Computing bringt Hypervisor auf neues Level, um Rechner und Speicher automatisch zu verknüpfen
- **Beispiele**  
Amazon EC2, Microsoft Azure



# Platform as a Service (PaaS)

- IaaS + vorinstallierter Software, die für das Entwickeln und Ausführen von Software notwendig ist
- Ziel ist es die Entwicklung von Softwareanwendungen zu vereinfachen und zu beschleunigen
- Häufig:
  - Plattform wählen
  - Anwendung hochladen
  - Starten
- Fotango brachte 2006 als weltweit erster einen PaaS online
- Wurde Ende 2007 eingestellt, da Mutterkonzern es nicht als Kerngeschäft gesehen hat
- **Beispiele**  
AWS Elastic Beanstalk, Microsoft Azure Webapp



# Software as a Service (SaaS)

- IaaS + PaaS + fertige Anwendung
- Meist mittels Webbrowser als Client genutzt
- Häufig auch “On demand Software” genannt
- **Beispiele**  
Google Docs, Spreadsheets und Presentation
- Grundidee nicht neu, da bereits in den 1990er “Application Service Provider” (ASP) Anwendungen bereitgestellt haben
- ASP Kunden benötigten spezielle Clients
- SaaS setzt in der Regel Continuous Delivery um





# Function as a Service (FaaS)

- Aus dem “serverless” Bereich entstanden
- Ursprünglich nicht “ohne Server” gemeint, sondern ohne die Server-Aufgaben selbst zu erledigen
- Heutzutage häufig tatsächlich ohne den klassischen Server Gedanken
- Ressourcen fahren innerhalb von Millisekunden hoch und erledigen Aufgabe
- FaaS wird vor allem für die Entwicklung von Microservices genutzt
- **Beispiele**  
AWS Lambda, Hook.io



# Bereitstellungsmodelle

## Public

- Ressourcen von Dritten verwaltet
- Öffentlich zugänglich
- Weit verbreitet, um Arbeit auszulagern

## Private

- Ressourcen von Dritten oder selbst verwaltet
- Kein Vertrauen an Dritte notwendig
- Entspricht meist traditioneller IT-Infrastruktur

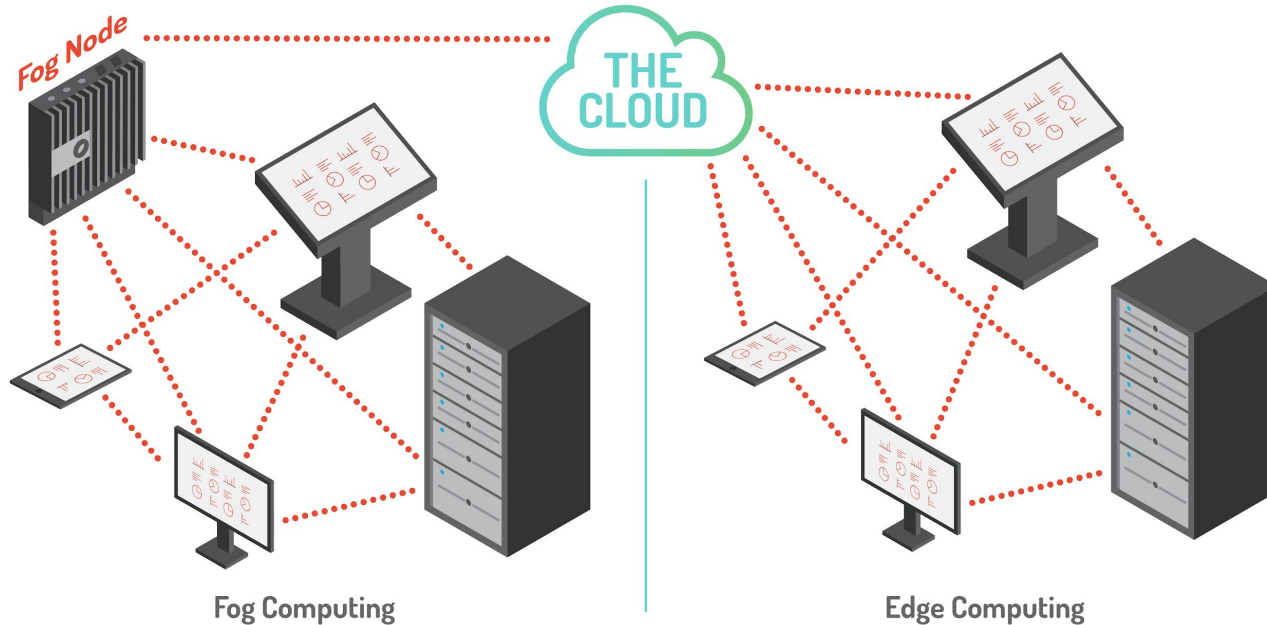
## Community

- Zwischen Public und Private
- Zusammenschluss von Unternehmen und/oder Organisationen
- Gemeinsame Ansprüche realisieren

## Hybrid

- Zwei oder mehr Clouds werden miteinander verbunden
- Ermöglicht schrittweisen Umstieg zur Public Cloud
- NASA nutzt Private + Public Cloud

# Edge & Fog Computing 1/2



Quelle: <https://forestgiant.com/articles/fog-vs-edge/>



# Edge & Fog Computing 2/2

## Gemeinsamkeiten

- Intelligenz zum Rand des Netzwerkes verlagern
- Vorteile
  - Senkung der Latenz
  - Steigerung der Sicherheit durch passendes Management
  - Senkung der benötigten Bandbreite
- Vorrangig von der Industrie getrieben, um z.B. Ausfallzeiten zu reduzieren

## Unterschiede

- Speicher, Rechenleistung und Intelligenz
  - Edge: Auf das Endgerät
  - Fog: Auf Fog-Node
- Fog Computing durch schrittweise Verarbeitung komplexer als Edge Computing



# Einschränkungen & Hinweise 1/2

- Abhängigkeit muss abgewogen werden
- **Layer 8 Fehler**
  - Verlust oder Kompromittierung kann Zugang zu gesicherten Cloud Systemen ermöglichen
  - Phishing und Social Engineering
  - Marketing Exactis im letzten Monat 340 Mil. Datensätze öffentlich zugänglich gemacht
- **Mangelhafte Backups**
  - Daten müssen adäquat synchronisiert werden
  - Ransomware als große Bedrohung, da enorme Datenmengen zentralisiert werden
- **Systemfehler**
  - Mandantenfähige Systeme müssen Daten sauber trennen
  - Code Space verlor fast alle Daten durch Angriff auf AWS Management Console



# Einschränkungen & Hinweise 2/2

## - Privatsphäre

- Immer mehr Informationen online in Clouds gespeichert (“gläserner Bürger”)
- AGBs erlauben vielen Anbieter die Weitergabe von Informationen an Dritte
- Staat kann Informationen unter Umständen einfordern
- EU-DSGVO soll mehr Transparenz und Sicherheit im Umgang mit personenbezogenen Daten schaffen

## - Inflexibilität

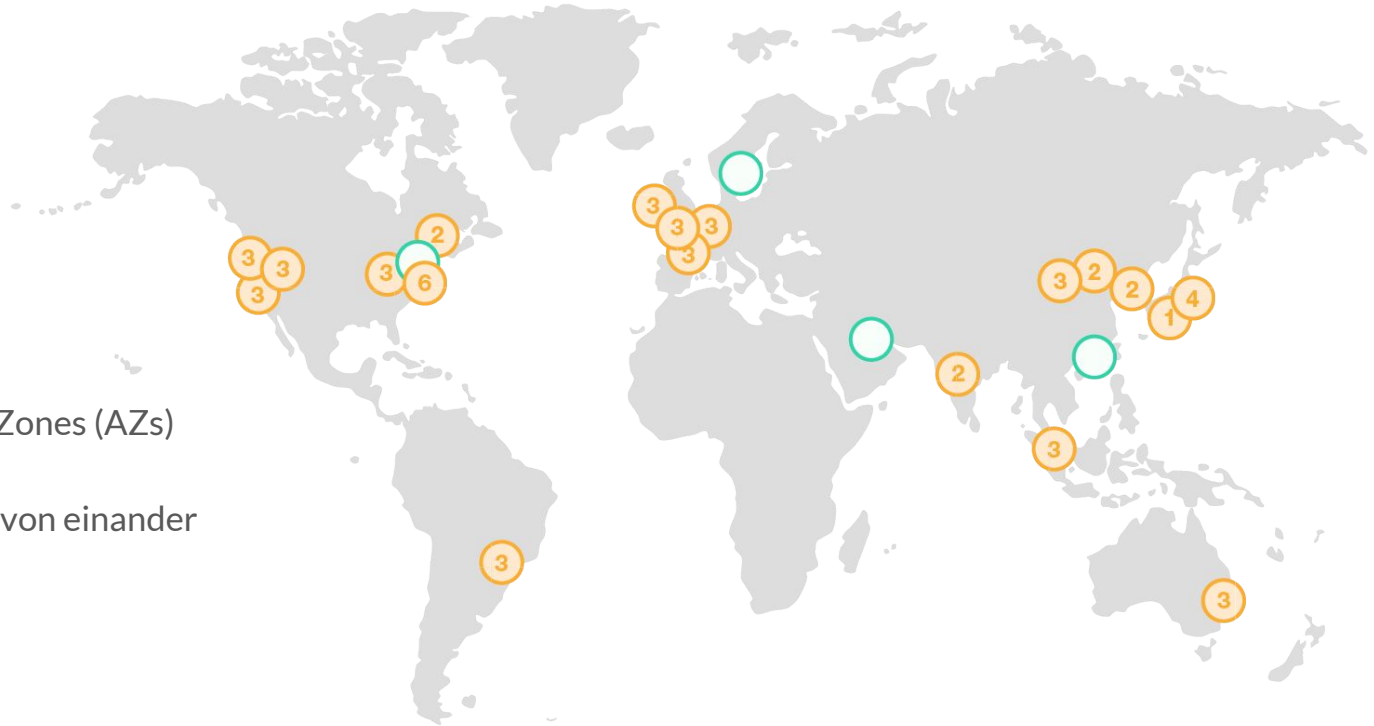
- Kontrolle zu genutzten Systemen geht stückweit verloren
- Individualisierungen geringer
- “Vendor Lock-In”, was den Wechsel zwischen Cloud-Anbietern erschwert

---

# Amazon Web Services (AWS)



- 18 Regionen
- 55 Availability Zones (AZs)
- AZs min. 10km von einander entfernt







# Grundlagen

- Im März 2006 der Öffentlichkeit bereitgestellt
- 2017 34% Marktanteil  
→ 17,4 Mil. \$ Umsatz
- Insgesamt über 90 Dienste zu über 20 verschiedenen Themenbereichen
- Pay per Use, um nur verbrauchte bzw. genutzte Ressourcen zu bezahlen
- AWS Educate als Netzwerk für Bildungsbeauftragte und Studenten
- In den USA zwei “Pop-Up Lofts”, um kostenlose Hilfestellung zu bieten



# Identity and Access Management (IAM)

- Nutzer- und Rechteverwaltung innerhalb von AWS
- Grob- und feingranulare Rechtezuweisung möglich
- Beschränkungen von Nutzern auf bestimmte Uhrzeiten und/oder Zeiträume
- Richtlinien zu Passwörtern und Multi-Faktor-Authentifizierung
- AMI-Rollen haben besondere Aufgabe
  - Dienste und Anwendungen können eine hinterlegte Rolle einnehmen, um Zugriff auf gesicherte Ressourcen zu erhalten
- Amazon Resource Names (ARNs) ermöglichen eindeutige Identifizierung



# EC2 & S3

## Elastic Compute Cloud (EC2)

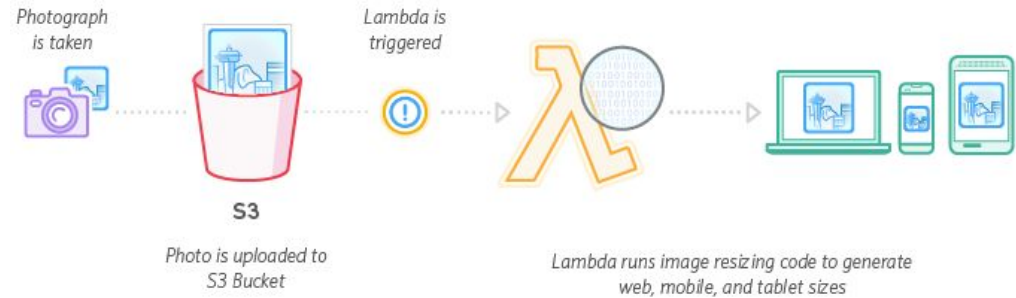
- Erstellung und Verwaltung von Rechenkapazitäten
- Amazon Machine Images (AMIs) dienen als fertige Templates
- Je nach Anwendungsfall speziell zugeschnittene Instanztypen möglich
  - 64 CPUs + 976 GB RAM + 8 NVIDIA K80-Hochleistungs-GPUs
  - bis zu 1952 GB RAM
  - 24 HDDs je 2048 GB oder 8 SSD je 1900 GB

## Simple Storage Service (S3)

- Beliebig große Datenspeicher
- “von Grund auf für eine Beständigkeit von 99,999999999% entwickelt”
- Abfragen vor Ort in Datenmengen von mehreren Exabyte
- Meist automatische Spiegelung von Daten in min. 3 AZs

# AWS Lambda

- Bildet das FaaS Modell ab
- Automatische Skalierung durch parallele Abfrageverarbeitung
- Unterstützt
  - C# (.NET Core 1.0 und 2.0)
  - Go 1.x
  - Java 8
  - Node.js (4.3, 6.10 und 8.10)
  - Python (2.7 und 3.6)



- Online-Editor für Node.js und Python
- Trigger müssen hinterlegt werden, um automatisierte Ausführungen zu ermöglichen

---

# Cloud Design Patterns

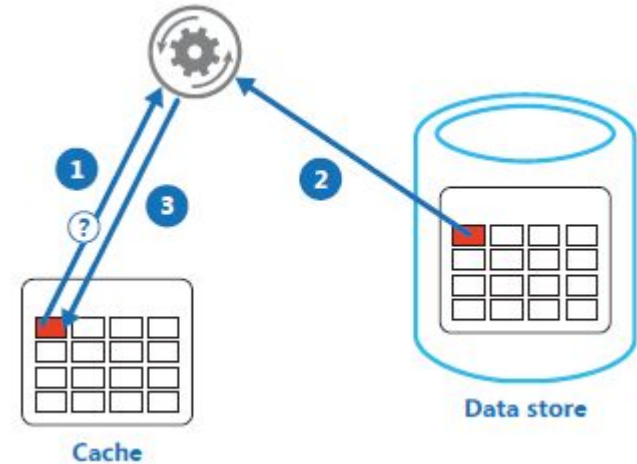


# Grundlagen

- Zu verstehen als Leitfäden und Hinweise für eine höhere Softwarequalität
- In der Regel Lösungen zu konkreten Problemstellungen
- Die Microsoft “patterns & practices group” hat ein Buch mit 24 Patterns veröffentlicht
- Problemfelder
  - Erreichbarkeit
  - Daten Management
  - Design und Implementierung
  - Benachrichtigungen
  - Management und Überwachung
  - Performanz und Skalierbarkeit
  - Stabilität
  - Sicherheit

# Cache-aside

- Cache Invalidierung/Synchronisierung stellt Problem dar
- I/O Operationen werden auf dem Cache getätigt und von dort in den Datenspeicher weitergegeben (Write-Through)
- Problematisch wenn weitere Quelle auf dem Datenspeicher arbeitet



1. Daten im Cache suchen/bearbeiten
2. Falls dort nicht vorhanden, dann aus Datenspeicher holen
3. Gleichzeitig in den Cache schreiben



# Sharding

- Ein einziger Datenserver kann bei großen Anwendungen zu Problemen führen
  - Mangelnder Speicherplatz
  - Fehlende Rechenleistung
  - Netzwerk Bandbreite
  - Geographische Anforderungen
- Datenspeicher horizontal aufteilen in sogenannte “Shards”
- Jeder Shard besitzt den selben Aufbau, speichert aber verschiedene Datensätze
- Shard-Key zur Identifizierung (meist aus Datenattributen)
- **Lookup Strategie** mappt Key zu virtuellen/physikalischen Partitionen (standardmäßiger Ansatz)
- **Range Strategie** speichert ähnliche Daten zusammen (z.B. Bestellungen im Monat X)
- **Hash Strategie** nutzt Hashverfahren, um Daten gleichmäßig auf Shards zu verteilen



# Fragen!



## Quellen 1/4

Amazon Web Services (AWS)

<https://aws.amazon.com/de/> (abgerufen am 30.06.2018)

Bianchi, Alessandra ; Inc., 01.04.2000: Say good-bye to software as we know it and hello to ASP start-up

<https://www.inc.com/magazine/20000401/18093.html> (abgerufen am 26.06.2018)

Butler, Brandon ; Network World, 21.09.2017: What is edge computing and how it's changing the network

<https://www.networkworld.com/article/3224893/internet-of-things/what-is-edge-computing-and-how-it-s-changing-the-network.html> (abgerufen am 29.06.2018)



## Quellen 2/4

Gassner, Heinz ; Smart Industry Forum, 02.12.2016: What Do We Actually Mean By: IaaS, PaaS, SaaS?  
<https://smartindustryforum.org/what-do-we-actually-mean-by-iaas-paas-saas/> (abgerufen am 21.06.2018)

Han, Bowei ; Medium, 05.11.2017: An Introduction to Serverless and FaaS (Function as a Service)  
<https://medium.com/@Boweihan/an-introduction-to-serverless-and-faaS-functions-as-a-service-fb5cec0417b2> (abgerufen am 29.06.2018)

Homer, Alex; Sharp, John; Brader, Larry; Narumoto, Masashi; Swanson, Trent ; Microsoft, 04.03.2014:  
Cloud Design Patterns  
<https://www.microsoft.com/en-us/download/details.aspx?id=42026> (abgerufen am 01.07.2018)



## Quellen 3/4

Innocent, Johnson ; DZone / Cloud Zone, 09.03.2017: Cloud Computing Deployment Models  
<https://dzone.com/articles/cloud-computing-deployment-models> (abgerufen am 28.06.2018)

Larkin, Andrew ; Cloudacademy, 26.06.2018: Disadvantages of Cloud Computing  
<https://cloudacademy.com/blog/disadvantages-of-cloud-computing/> (abgerufen am 29.06.2018)

National Institute of Standards and Technology (NIST) - The NIST Definition of Cloud Computing 2011  
<https://csrc.nist.gov/publications/detail/sp/800-145/final> (abgerufen am 03.05.2018)



## Quellen 4/4

OpenFog, 09.02.2017: OpenFog Reference Architecture for Fog Computing

[https://www.openfogconsortium.org/wp-content/uploads/OpenFog\\_Reference\\_Architecture\\_2\\_09\\_17-FINAL.pdf](https://www.openfogconsortium.org/wp-content/uploads/OpenFog_Reference_Architecture_2_09_17-FINAL.pdf) (abgerufen am 10.06.2018)

Rouse, Margaret ; TechTarget, 09.2017: Infrastructure as a Service (IaaS)

<https://searchcloudcomputing.techtarget.com/definition/Infrastructure-as-a-Service-IaaS> (abgerufen am 21.06.2018)

Utley, Gary ; The CWPS Blog, 12.05.2018: 6 Most Common Cloud Computing Security Issues

<https://www.cwps.com/blog/cloud-computing-security-issues> (abgerufen am 29.06.2018)