



# Cloud Computing

Das Potential von Cloud Computing für Kommunen entdecken

13. Mai 2024 | Rena Wißmeier

# Agenda

- 01** Kommunale Gemeinschaftsstelle für Verwaltungsmanagement
- 02** Aktuelle Herausforderungen in dem Kommunen
- 03** **Cloud Computing** – Was ist das eigentlich?
- 04** **Cloud Computing** – Chancen
- 05** **Cloud Computing** – Interkommunale Zusammenarbeit





# 01

## Kommunale Gemeinschaftsstelle für Verwaltungsmanagement



# Leistungsportfolio der KGSt

von Städten, Gemeinden und Kreisen getragenes **Entwicklungszentrum des kommunalen Managements** mit **mehr als 2.500 Mitgliedern in Deutschland, Österreich und der Schweiz**

## KGSt®-Berichte

durchschnittlich 12 pro Jahr

## KGSt®-Portal

Mit über 70.000 Nutzenden

## Vergleichsringe

mit Best-Practice-Austausch

## KGSt®-Kommunect

Vernetzungsplattform für Projekte

## Beratungsleistungen

ausgerichtet an den Bedarfen unserer Mitglieder

## Seminare- und Lehrgänge

Präsenz und KGSt®-Campus.digital

## Kongresse

zu unseren gutachtlichen Themenfeldern

## Stellenbewertung

nach Tarif- und Beamtenrecht

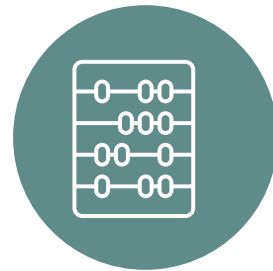


# 02 Aktuelle Herausforderungen in den Kommunen

# 02 Herausforderungen in Kommunen



Fachkräftemangel  
und demographischer  
Wandel



rapider  
(technologischer)  
Wandel



„Silos“  
auch in der  
Datenhaltung und  
-verarbeitung/  
„Monolithen“



Handlungsdruck  
von außen und  
innen (z.B. OZG)+  
Zentralisierungstendenzen

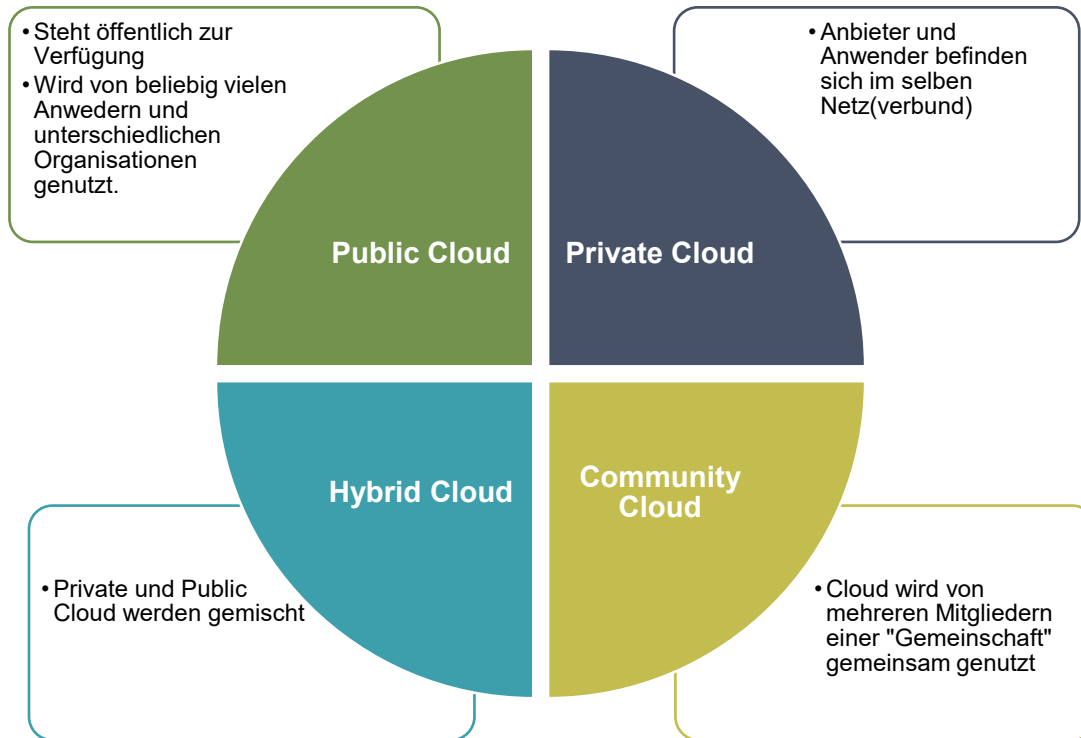


Stapelkrisen



# 03 Cloud Computing – Was ist das eigentlich?

# Cloud Computing - Bereitstellungsmodelle



- Nicht jede Cloud gleich Public Cloud.
- **Public Cloud:** beispielsweise Clouds von Microsoft (Produkt Azure), von google (Produkt Google Cloud Plattform) oder Amazon (Produkt Amazon Web Service (AWS)).
- **Community Cloud oder Private Cloud** beispielsweise gemeinsame kommunale Cloud oder auch Deutsche Verwaltungcloud.
- **Hybrid Cloud** Verknüpfung von Privat und Public Cloud und ungleich Modellen der hybriden Nutzung von on premise-Diensten und Clouddiensten zu verwechseln..



# Cloud Computing - Servicemodelle



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

vollständige Anwendungen für den Endnutzer



## **Platform as a Service (PaaS)**

Plattformen zur Entwicklung und Ausführung von Diensten



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

Hardware, auf der Plattformen oder Anwendungen laufen



# Cloud Computing: klare Kriterien (BSI)

- Zugänglichkeit und Verfügbarkeit von verschiedenen Endgeräten (Broad Network Access).
- Automatische Bereitstellung von IT-Ressourcen ohne weitere Interaktion mit dem Service Provider (On-Demand Self-Service).
- Automatisches Messen (Metering) und Optimierung der Kapazität (Measured Services).
- Schnelle und „elastische“ Verfügbarkeit der Services (Rapid Elasticity)
- Bedienen aus einem Pool des Anbieters an Ressourcen (Resource Pooling).
- Multimandantenfähigkeit/ Multi-Tenant Modell
- pay-per-use
- Service orientierte Architektur (SOA)
- (REST-)API
- **Volle Wirkung nur, wenn „wirklich Cloud“**



# Eine kleine Legende...

## – Bezos-API-Mandat (2002)



„Eines Tages erteilte Jeff Bezos also einen Auftrag.

Das macht er natürlich ständig, und die Leute regen sich wie Ameisen, [...], wenn das passiert. Aber bei einer Gelegenheit - so um 2002 herum, plus/minus ein Jahr - erließ er ein Mandat, das so gewaltig und schwerfällig war, dass es all seine anderen Mandate wie unaufgeforderte Boni [...] aussehen ließ.“

(<https://gist.github.com/chitchcock/1281611>)

# Eine kleine Legende...

## – Bezos-API-Mandat (2002)

Der Legende nach lautet der Auftrag an seine Teams wie folgt:

1. Alle Teams werden fortan ihre **Daten und Funktionen über Dienstschnittstellen (service interfaces) offenlegen**.
2. Die **Teams müssen über diese Schnittstellen miteinander kommunizieren**.
3. Es wird **keine andere Form der Kommunikation zwischen den Prozessen** erlaubt sein: keine direkte Verknüpfung, kein direktes Lesen des Datenspeichers eines anderen Teams, kein Shared-Memory-Modell, keine Hintertürchen. Die einzige Kommunikation, die erlaubt ist, sind Aufrufe von Dienstschnittstellen über das Netz.
4. Es spielt **keine Rolle, welche Technologie** sie verwenden. HTTP, Corba, Pubsub, benutzerdefinierte Protokolle - es spielt keine Rolle.
5. **Alle Dienstschnittstellen müssen ausnahmslos von Grund auf so konzipiert sein, dass sie externalisiert werden können**. Das heißt, das Team muss planen und entwerfen, um die Schnittstelle für Entwickler in der Außenwelt zugänglich zu machen. Keine Ausnahmen.
6. Jeder, der dies nicht tut, wird gefeuert.
7. Vielen Dank; einen schönen Tag noch!



# Cloud ist auch ein Konzept: SOA, API

...vorallem bei SaaS-Diensten:

## ➤ **SOA: Service Oriented Architecture**

mehrere Anwendungen als Sammlung von unabhängigen, wiederverwendbaren Diensten

-> **Baukastensystem**

## ➤ **API: Application Programming Interface**

„Beschreibung“ für andere Dienste, wie Service von außen angesprochen werden kann. Im Optimalfall ist schon die „Programmierung“ API. Kein Code im Frontend

⇒ **Interoperabilität und Modularität**



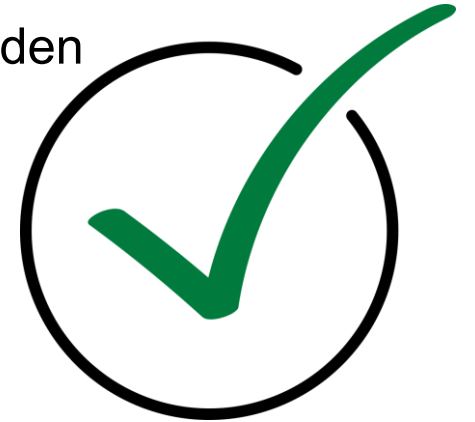
# Zusammenspiel: SOA, API

- nicht für jeden Anwendungsfall müssen neue Schnittstellen programmiert werden
- Tw. für Endnutzer anwendbar z.B. Connectoren in MSFT 365 (Workflowautomatisierung)
- keine geschlossenen monolithische Alles-Können-Anwendungen, sondern Services können „zusammengeschaltet“ werden
- Services und Module austauschbar
- bisherigen Fachverfahren zumeist nicht „cloudbereit“: Nur weil zwei bestehenden Fachverfahren in dieselben Cloud gelegt werden, löst das das „Kommunikationsproblem“ nicht
- Ende zu Ende-Umsetzung mit diesen Konzepten leichter umsetzbar.



# Chancen...

- **„Interaktion“ von Services** - Dezentralität kann technisch „harmonisiert“ werden
- **Ende zu Ende-Umsetzung**
- **Evergreening** (kontinuierliche Fortentwicklung der Dienste)
- **Automatisierung** (in vielen Diensten bereits enthaltene Anwendungen)
- 24/7 Betrieb
- **IT-Sicherheit:** Erhöhte Sicherheit durch professionelle Cloud-Anbieter mit umfassenden Sicherheitsmaßnahmen und -standards
- **Nachhaltigkeit**
- **Resourcenersparnis** durch den Verzicht auf Kauf und Wartung eigener Hardware.
- Skalierbarkeit und Flexibilität
- Zugang zu **modernster Technologie und Software** ohne hohe Investitionen in die Infrastruktur
- Innovation: **Rapid Prototyping**, Erproben und Erkenntnisgewinn im Prozess, Fail fast
- **Verbesserung der Effizienz und Agilität** durch den Zugriff auf gemeinsam genutzte Ressourcen und Services.



# Chance: Government as a Platform (2011)

## *Government as a Platform (GaaP)*

Government as a Platform wurde 2011 von O'Reilly<sup>175</sup> als Ansatz diskutiert. Im Rahmen seiner Auseinandersetzung mit Plattformen aus dem privaten Sektor wie Google, Facebook oder Twitter geht er der Frage nach, was der öffentliche Sektor lernen und wie er die Potenziale der **Plattformlösungen** nutzen kann. GaaP beschreibt ein Anwendungssystem, das auf Basis einer vorgegebenen technischen Infrastruktur aus **wiederverwendbaren Bausteinen** (Systemen, Diensten, Prozessen, Analytics usw.) eine Grundstruktur bzw. einen Kern vorgibt, in welchen die Anwendenden ihre Anwendungen und Dienstleistungen selbstständig integrieren können. Typische modulare Bausteine sind neben offenen und standardisierten Programmierschnittstellen (APIs), die die Interaktion verschiedener Softwareanwendungen sowie die Integration verschiedener Dienste und Datenquellen erst ermöglichen, auch die Festlegung von **standardisierten Technologiekomponenten** wie Hardware, Software, Netzwerke und Datenbanken. Sie bilden die Basisinfrastruktur der Plattform. Zudem bedarf es festgelegter Standards, die die Grundlagen für die Interoperabilität und die **einheitliche Kommunikation** zwischen den Systemen festlegen, sowie autorisierter Datensätze als Basis des einfachen Datenaustausches zwischen den Systemen.<sup>176</sup> All diese Komponenten können einmalig erstellt und fixiert werden.<sup>177</sup> GaaP erfordert auch die **Definition von Rollen, Verantwortlichkeiten und Leitbildern**.<sup>178</sup>

Zwei wesentliche Vorteile von GaaP sind die Steigerung der Effizienz und die Nutzerfreundlichkeit.<sup>179</sup> Die zentrale Gestaltung und Bereitstellung der Kernfunktionen erlauben es externen Plattformnutzenden, Lösungen dezentral und selbstständig zu entwickeln. Der Fokus der Entwickelnden kann hierbei auf innovativen, anpassungsfähigen und kundenorientierten Lösungen liegen, die sich infolge der Selbstorganisation in ein **Ökosystem mit Plattforminhalten** einfügen.<sup>180</sup> Das Setzen von Rahmenbedingungen bietet Plattformen den Vorteil, dass sie eine Vielzahl von dezentralen Akteuren koordinieren können, ohne dass hierarchische Strukturen vorhanden sein müssen.<sup>181</sup>

Der GaaP-Ansatz wird in vielen Ländern, u.a. im **Vereinigten Königreich, in Estland oder Italien** bereits umgesetzt.<sup>182</sup> Die strategischen Ziele zur Ausrichtung und Gestaltung der IT-Architektur wurden jeweils zentral und ganzheitlich festgelegt, bevor die technischen Komponenten hierfür dezentral entwickelt werden.<sup>183</sup> Das Vereinigte Königreich gibt Technologie- und Servicestandards, Anleitungen und Designprinzipien vor und stellt verschiedene Basiskomponenten für die Entwickelnden zentral zur Verfügung. Wenn gleich die Umsetzungen nicht zwangsläufig die Sozialleistungsverwaltung betreffen, lassen sich relevante Aspekte ableiten.

Gutachten des Normenkontrollrats:  
[https://www.normenkontrollrat.bund.de/Webs/NKR/DE/veroeffentlichungen/gutachten/\\_documents/2024-03-26-nkr-gutachten-sozialleistungen.html](https://www.normenkontrollrat.bund.de/Webs/NKR/DE/veroeffentlichungen/gutachten/_documents/2024-03-26-nkr-gutachten-sozialleistungen.html)





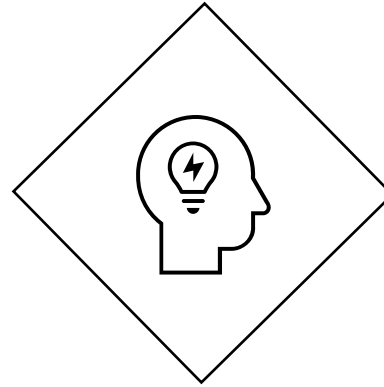


# 05 Interkommunale Zusammenarbeit

# Interkommunale Zusammenarbeit



- Einkaufskooperation: Gemeinsame Vertragsverhandlung/ -gestaltung, gemeinsame Rahmenverträge
- Outsourcing bzw. gemeinschaftliche Aufgaben -> Personalmangel/ andere Fachkräfte
- Gerade bei kleinen Kommunen, oft einzige Chance, weil nicht genug qualifiziertes Personal vor Ort
- Ansatz make vs. buy (z.B. auch in Genossenschaft + Fördermittel)
  - gemeinsame Entwicklungen auf PaaS und SaaS, die den Kommunen gehört
  - gemeinsame Automatisierung von Prozessen
  - Einsatz kommunaler Fachexperten, direkte Integration des fachlichen Expertenwissens
  - Stichwort Low Code-No Code (Der Brief gehört dem Verfassenden.)



Rückfragen? Gerne jetzt!

---



**Rena Wißmeier**

Referentin im Bereich Informations- und  
Organisationsmanagement  
Rena.Wissmeier@kgst.de  
0221 - 37 68 939