

CLOUD COMPUTING: AZURE

Hochschule Koblenz - BRICKMAKERS GmbH

ÜBERSICHT

1. Einführung
2. Azure Services
3. Workshop & Hands-On
4. Azure Services - Ausblick

CLOUD COMPUTING: AZURE
Kapitel 1: Einführung

KAPITEL 1: EINFÜHRUNG

- 1. BRICKMAKERS GmbH**
- 2. Über mich**
- 3. Was ist Azure?**

KAPITEL 1: EINFÜHRUNG

1.1 BRICKMAKERS GmbH

- Entwicklungsagentur aus Koblenz
- ca. 50 Mitarbeiter
- Services:
 - App-Entwicklung (mobile, Web, Windows)
 - UI / UX Design
 - Cloud Services
 - Online Marketing



Kommt gerne auf einen Kaffee vorbei
Oder besucht uns bei einem Meetup!

Cornelsen

Allianz 

Telefonica

 STRATO

KAPITEL 1: EINFÜHRUNG

1.2 Über mich

- Jonas Österle
- B. Eng. Elektrotechnik, HS Koblenz
- 2 Jahre bei BRICKMAKERS
- Mitglied im Team Mobile
- Tätigkeiten:
 - Full-Stack Developer
 - Azure Verwaltung
 - Weiterbildung
 - Mentoring
 - *Hochschule*

KAPITEL 1: EINFÜHRUNG

1.3 Was ist Azure

- hoch skalierbare Cloud-Computing Plattform
 - IaaS, PaaS, SaaS und weitere Services
 - seit 2010 verfügbar
 - *Hybrid Cloud*
 - Über 70+ Angebote
 - 54 globale Regionen
 - 90% der Fortune-500 Unternehmen
-
- IaaS: Infrastructure as a Service
 - PaaS: Platform as a Service
 - SaaS: Software as a Service

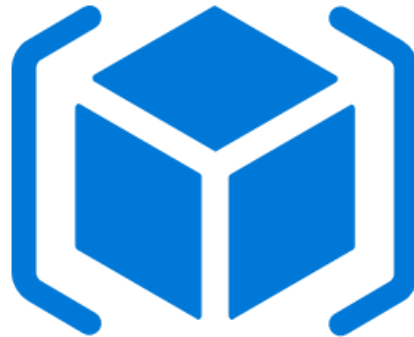
CLOUD COMPUTING: AZURE
Kapitel 2: Azure Services

KAPITEL 2: AZURE SERVICES

1. Ressource Group
2. App Service
3. App Service Plan
4. Datenbanken
5. Blob Storage
6. Deployment

2. AZURE SERVICES

2.1 RESSOURCE GROUP



- Jeder Service gehört zu einer *Ressource group*
 - dient als *logischer Ordner*
 - trennt Projekte oder Abteilungen
- Erlaubt Gruppenfunktionen (bspw. löschen)
- Kostenüberblick pro Gruppe

2. AZURE SERVICES

2.2 App Service



- Container für alle typischen Webanwendungen
 - Statische Webseite mit HTML, CSS
 - Webanwendungen (Frontend)
 - Backend Server (API)
- Betriebssystem Windows / Linux

Info:

- Name wird auch für URL verwendet
- benötigt *App Service Plan*
- Subscription -> payment
- Ressourcengruppe angeben

2. AZURE SERVICES

2.3 App Service Plan

(ASP / ASE):



- PaaS: Plattform as a Service
 - bestimmt die Hardware, die für App Services zur Verfügung steht
 - kann an spezifische Orte “gebunden” werden
 - verringert Latenzzeiten
 - ist jederzeit skalierbar (horizontal, vertikal)
-
- Lizenz Basic 1 (B1) ist kostenlos
 - Application Insights \approx Google Analytics

2. AZURE SERVICES

2.3 App Service Plan: Tipps

- Plan kann mehrere App Services beinhalten
 - die Ressourcen werden dann geteilt
 - auf *Performance-Diebe* achten
 - Anspruchsvolle Apps sollten so konfiguriert sein
 - *auto-scale based on CPU and memory*
 - *Per-Plan Scaling* oder *Per-App-Scaling*
-
- per plan: kontrolliert auf wievielen Servern die App gleichzeitig läuft
 - per app : kontrolliert **maximale** für die App zugewiesene Server

2. AZURE SERVICES

2.4 Datenbanken: SQL



- besteht aus SQL Server + SQL Datenbank
- basiert auf *Microsoft SQL Server Datenbank Engine*
- zuverlässig und sicher
- *Standardlösung*

2. AZURE SERVICES

2.4 Datenbanken: Cosmos DB

- beste Option für NoSQL
- global verteilt
- Mehrere Api´s verfügbar (ua. SQL, MongoDB)
- flexibel nutzbar, einfach skalierbar in andere Regionen
- Geschwindigkeit & Zuverlässigkeit
 - 99.99%
 - read: > 10ms
 - write: > 15ms

2. AZURE SERVICES

2.5 Blob Storage



- Objektspeicherlösung von Microsoft
 - optimiert für das Speichern großer Mengen von unstrukturierten Daten
 - Bilder oder Dokumente
 - Dateien für verteilten Zugriff
 - Audio- und Video-Streaming
 - Speichern von Daten für Sicherung / Wiederherstellung
 - Zugriff über REST-API, CLI, Bibliotheken
-
- CLI: Command Line Interface

2. AZURE SERVICES

2.6 Deployment Methoden

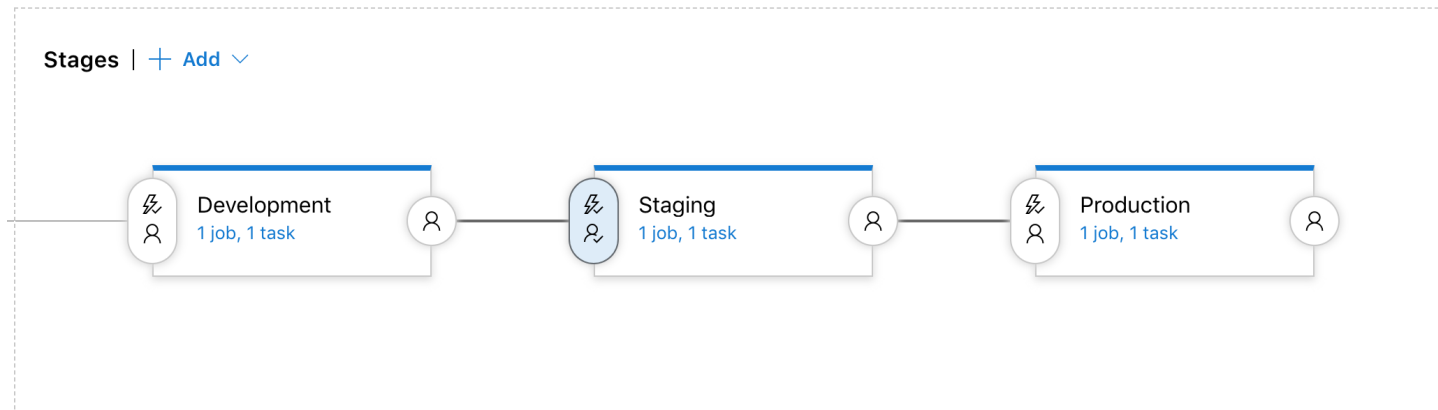
	Build	Upload
Manuell	manuell	manuell
Tools	manuell	semi-auto
CI	auto	mannuell
CI & CD	auto	auto

- für den manuellen Upload sind extra Credentials notwendig
- Tools: Einmaliger Login mit Azure Account üblich
- CI / CD : Konfiguration notwendig

2. AZURE SERVICES

2.6 Deployment : Umgebungen

- Lokale Umgebung: zum Entwickeln
- **Development:** CI & CD: *develop-Branch*
- **Staging** CI & CD: *master-Branch*
- **Production:** nach Verifizieren von Stage



2. AZURE SERVICES

2.6 Deployment : Zero-Downtime

Probleme beim Deployment:

- Server kurzzeitig nicht verfügbar
- Neuer Code könnte neue Bugs verursachen
- Tests beim Buildvorgang helfen
- manche Apps benötigen *Warmup*

Vorgehen

- App Services verfügen über Slots
- Bspw. für Production, Staging, Develop
- Deploy Slot B -> Test B -> Swap A & B
- Falls notwendig: Warmup auf Slot B vor Swap

CLOUD COMPUTING: AZURE

Kapitel 3: Workshop - App Innovation

KAPITEL 3: WORKSHOP

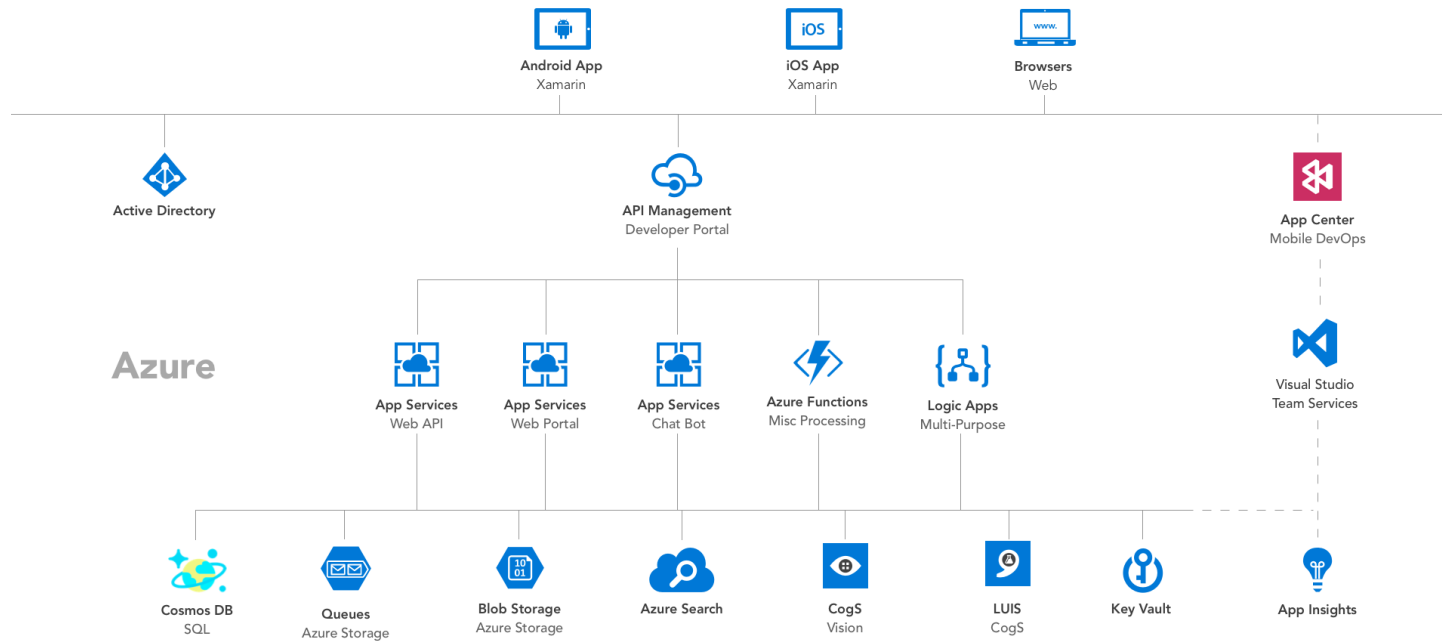
1. Systemüberblick
2. Hands-On

App Innovation Workshop

Learn Azure, Mobile & AI

3. WORKSHOP

3.1 ARCHITEKTUR & ANWENDUNG



3. WORKSHOP

Hands-On

CLOUD COMPUTING: AZURE

Kapitel 4: Azure Services - Ausblick

KAPITEL 4: AZURE SERVICES - AUSBLICK

1. Authentifizierung
2. Azure Search Engine
3. Azure Functions
4. Azure Container Service
5. azureAI
6. azureChatBots
7. azureIoT

4. AZURE SERVICES - AUSBLICK

Authentifizierung: Active Directory



- IDaaS: Identity as a Service
- Verwaltung von Benutzeridentitäten
- viele Konfigurationen für Zugriff, Sicherheit
- Multi Faktor Authentifizierung
- ermöglicht interne Geschäftsanwendungen

4. AZURE SERVICES - AUSBLICK

Authentifizierung: Active Directory B2DC

- ähnlich Azure AD
 - simple Authentifizierung
 - integration in alle Technologien
 - unterstützt Social Authentication Providers (Facebook, Twitter, etc.)
 - Login kann als *White-Label* konfiguriert werden
-
- B2C: Business to Customers

4. AZURE SERVICES - AUSBLICK

Azure Search Engine



- KI basierter Cloudsuchdienst
 - Suchfeatures integriert aus Office, Bing
 - mit Vorschlägen bei falscher Eingabe
 - Standortbezogene Suche
 - zentral skalierbar
 - stellt viele Funktionen zur Verfügung
 - kognitive Suche mit integrierten Skills
 - Zeichenerkennung in Scans
 - Entitätserkennung
 - Schlüsselbegriffserkennung
-
- Beispiel DPDHL

4. AZURE SERVICES - AUSBLICK

Azure Functions



- serverlose Architektur
- skaliert nach Anforderung
- bezahlt nach Uptime
- Unterstützt C#, JavaScript, F#, ...
- arbeitet mit Triggern (bspw. HTTPTrigger, BlobTrigger, etc.)
- besitzen Warmup Zeit
- keine Ressourcen zugeteilt
- Ressourcen sind dynamisch verwaltet

4. AZURE SERVICES - AUSBLICK

Azure Container Service



- unterstützt Kubernetes, DC / OS, Swarm
-

4. AZURE SERVICES - AUSBLICK

Artificial Intelligence

4. AZURE SERVICES - AUSBLICK

chat bots

4. AZURE SERVICES - AUSBLICK

IoT Hub

VIELEN DANK
für Ihre Aufmerksamkeit!