

CC: CLOUD COMPUTING NETWORKING & HIGH AVAILABILITY

Dipl.-Medieninf. Hai Dang Le
Software Engineer
hdang.88@gmail.com

2018



AGENDA

4. NETWORKING & HIGH AVAILABILITY

- a. VNets, Loadbalancer
- b. High Availability



4. CLOUD COMPUTING NETWORKING

A. VIRTUAL NETWORKS

RECAP: NAT (LETZTE VORLESUNG)

Frage: Wie werden Computer innerhalb eines Heimnetzwerkes vor Zugriffen von außerhalb durch NAT "geschützt" ?

NETWORK ADDRESS TRANSLATION (NAT)

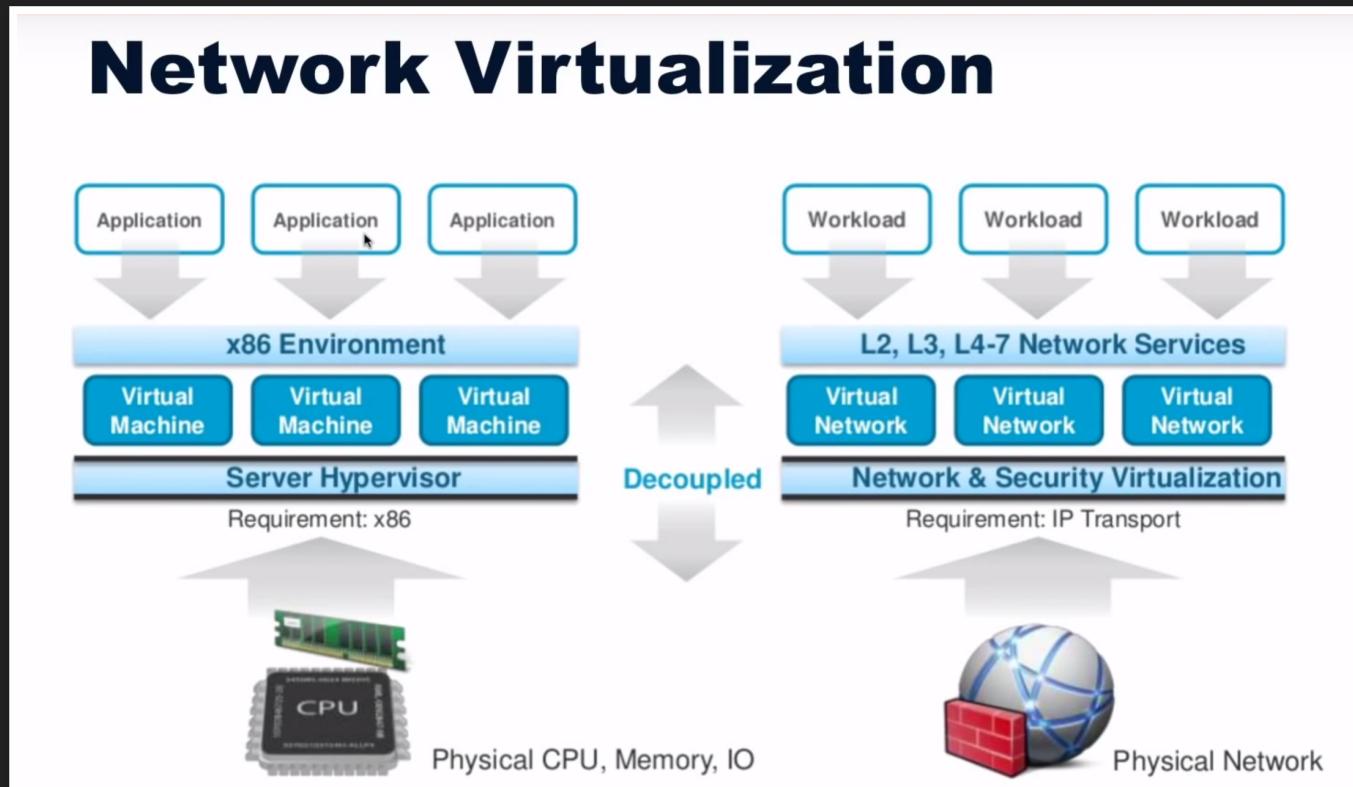
VIRTUELLE NETZWERKE (AM BEISPIEL AZURE VNET)

- Was ist ein Virtuelles Netzwerk ?
- Was ist Netzwerkvirtualisierung

VIRTUELLE NETZWERKE

- Woraus besteht ein Computer Netzwerk ?
- Firewall
- Router, Switch
- Loadbalancer
- Network Interface Cards
- etc.

NETZWERKVIRTUALISIERUNG



NETZWERKVIRTUALISIERUNG

- Abstrahierung von Netzwerk-Resourcen von Hardware-Komponenten
- Netzwerkkomponenten und Netzwerkfunktionen werden in Software repliziert
- die physikalischen Resourcen / Komponenten dienen zum Daten Transport
- die virtualisierten Komponenten steuern und definieren die Transportwege

NFV & SDN

- Techniken: Network Function Virtualization
- Entkopplung von Hardware-Resourcen von Funktionen
- Ziel: Optimierung/Effizienzsteigerung von Netzwerkfunktionen durch bessere Auslastung der Infrastruktur
- Network Functions: Firewall, Loadbalancing (L4-L7)

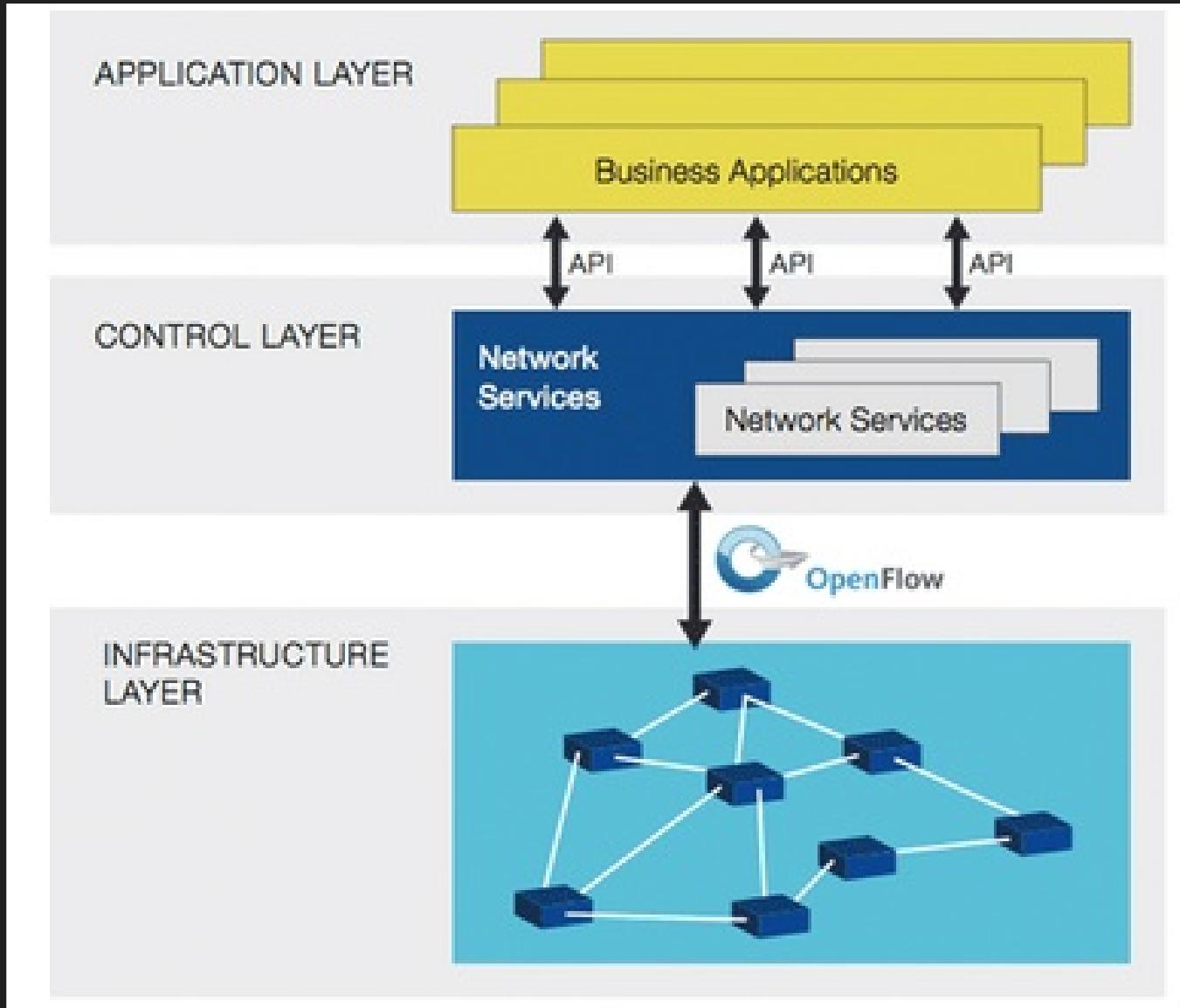
NETWORK FUNCTION VIRTUALIZATION

SDN

- Techniken: Software Defined Networking
- komplementär zu Network Function Virtualization
- dient dem Netzwerkmanagement: zentralisiert die Steuerung von Switches und Routern über APIs
- Einsatz: Cloud Computing zum steuern und bereitstellen von virtuellen Netzwerken
- Entkopplung von "Data-plane" und "Control-plane"

SDN

SDN



VIRTUELLES NETZWERK

- entsteht durch die Anwendung von Netzwerkvirtualisierung(-techniken)
- es werden logische / nicht physikalische Netzwerke gebildet
- wired oder wireless

MOTIVATION

- Welchen Nutzen bringt ein Virt. Netzwerk:
- Isolation: private network in der Cloud -> Security
- Performance: Direkte Kommunikation zwischen Netzwerk Komponenten (innerhalb des VNets)
- Organisation: Backend Servers (DBs, Storage etc.) vs Internet Facing Servers (Web Apps)
- Kontrolle: security policies, DNS routing, FW-Regeln, Subnets

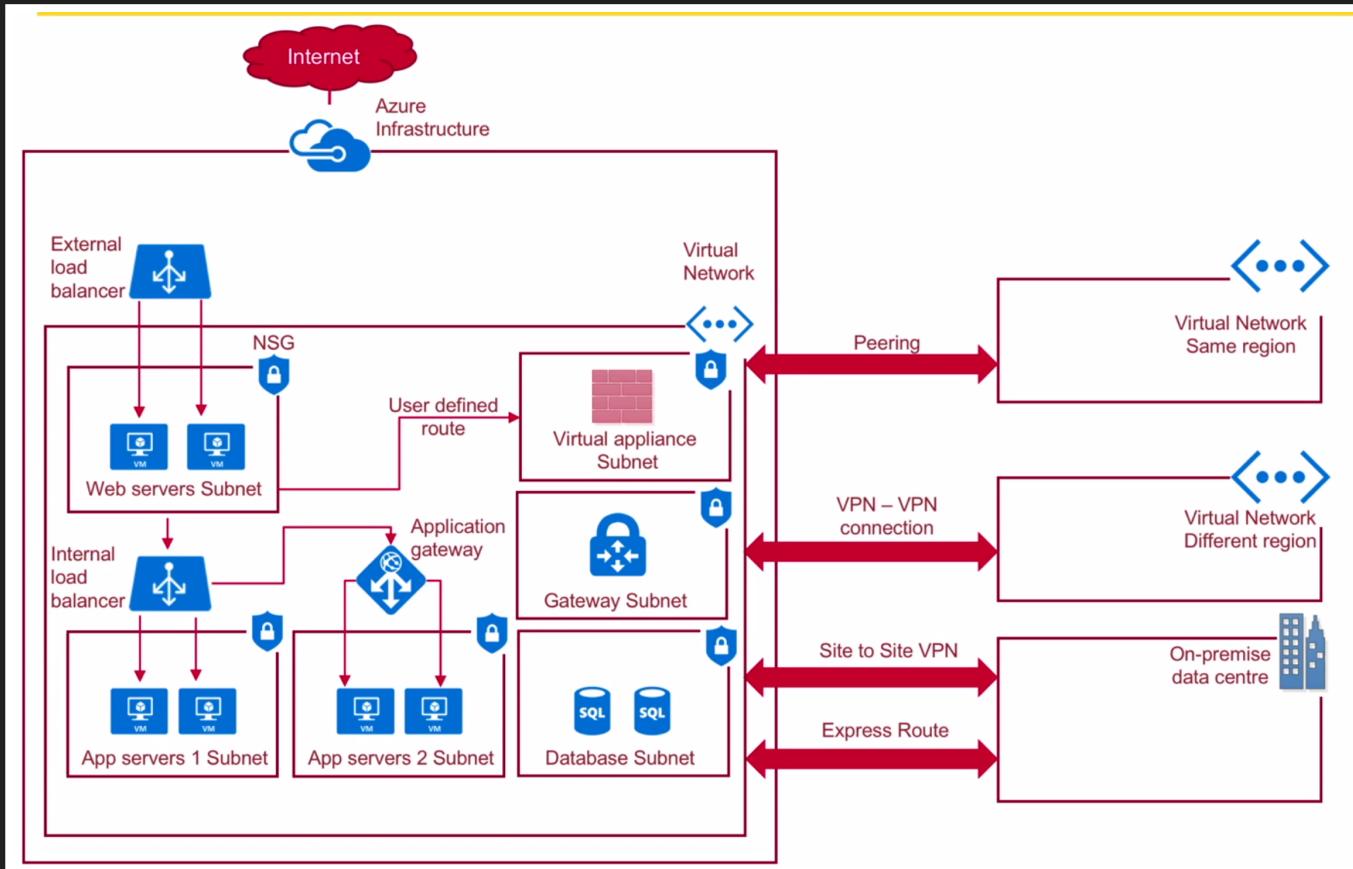
AZURE VNET

Features:

- Isolation & Segmentation
- Outbound/Internet Communication
- Network Security & Routing
- Azure Managed Services Integration
- On-Premises Communication (VPN)
- VNet Peering

Azure VNet Overview

AZURE VNET



AZURE VNET PRICING

- VNet ist kostenfrei
- pro Subscription bis zu 50 VNets über alle Regionen
- Kostenpflichtig: Public IP Adressen, VPN Gateway, API Gateway, VMs, VNet Peering

Azure VNet Pricing

AZURE VNET

Network Security Groups (NSG)

- definiert Netzwerk Routing Regeln für Netzwerk Interfaces / Subnets (Firewall)
- Inbound Regeln und Outbound Regeln
- Regel besteht aus Name, Priorität, Source, Destination, Port, Action

AZURE VNET

NSG

Inbound security rules						
PRIORITY	NAME	PORT	PROTOCOL	SOURCE	DESTINATION	ACTION
100	allow_kube_tls	443-443	TCP	Any	Any	Allow ...
101	allow_ssh	22-22	TCP	Any	Any	Allow ...
65000	AllowVnetInBound	Any	Any	VirtualNetwork	VirtualNetwork	Allow ...
65001	AllowAzureLoadBalancerInBound	Any	Any	AzureLoadBalan...	Any	Allow ...
65500	DenyAllInBound	Any	Any	Any	Any	Deny ...
Outbound security rules						
PRIORITY	NAME	PORT	PROTOCOL	SOURCE	DESTINATION	ACTION
65000	AllowVnetOutBound	Any	Any	VirtualNetwork	VirtualNetwork	Allow ...
65001	AllowInternetOutBound	Any	Any	Any	Internet	Allow ...
65500	DenyAllOutBound	Any	Any	Any	Any	Deny ...

VNET SUBNETS

- Was ist Subnetting ?
- Segmentierung des Adressraums in Teilräume
- man verwendet häufig die Classless Inter-Domain Routing (CIDR-) Notation
- Nutzen: Organisation (Sicherheit), Performance, Vereinfachung von Routing

AZURE VNET - PEERING

- VNet Peering
- Wie können 2 VMs innerhalb 2 unterschiedlicher VNets miteinander kommunizieren ? (ohne über das Internet zu gehen)

AZURE VNET - PEERING

- VMs können direkt über ihre private IP miteinander kommunizieren
- Subnetze dürfen sich nicht überlappen
(Eineindeutigkeit der privaten IPs)
- Routing geschieht über die Azure Infrastruktur
- local VNet Peering: innerhalb einer Azure Region
- global Vnet Peering: zwischen Azure Regionen

VNet Peering

AZURE VNET - PEERING

- Vorteile:
- Sicherheit - Traffic bleibt im Microsoft Backbone Network
- Performance - Traffic wird nicht indirekt über das Internet geroutet
- Datentransfer zwischen Subscriptions / Azure Services / Regionen werden ermöglicht

AZURE IP ADRESSEN

- public IPs - essentiell für Outbound Traffic
- private IPs - für traffic innerhalb eines VNets

AZURE PUBLIC IP ADRESSEN

- können folgenden Ressourcen zugewiesen werden:
- VM Network Interfaces, Internet-Loadbalancers, VPN Gateways, Application Gateways (API Gateway)
- statische public IPs - fest zugewiesen, bspw. nützlich um eigene DNS Namen zu binden
- dynamische public IP Adressen (ändern sich bei Restarts der zugeordneten Ressource)
- IPv4 und IPv6 (nur bei Loadbalancer)

VNET DEMO MIT AZURE CONTAINER SERVICES (AKS)

**FRAGE ZUM ÜBERBRÜCKEN DES
DEPLOYMENTS: WAS SIND DIE
UNTERSCHIEDE ZWISCHEN
HUB/SWITCH/ROUTER ?**

CLOUD NETWORKING - LOADBALANCER

- Was ist ein Loadbalancer ?

LOADBALANCER

- Typen: Hardware, Software Loadbalancer
- L4, L7 Loadbalancer
- Loadbalancer algorithmen: DNS round robin, gewichtetes round robin, least connection, affinity etc.

AZURE LOADBALANCER

- features:
- Loadbalancer algorithmen: source IP-hash basierte Verteilung, affinity basierte Verteilung

BONUSFRAGE

- Was ist das OSI Modell ?

OSI Modell

ZUSAMMENFASSUNG AZURE VNETS



4. CLOUD COMPUTING

B. HIGH AVAILABILITY

WAS IST HIGH AVAILABILITY (HA) ?

"... bezeichnet die Fähigkeit eines Systems, trotz Ausfalls einer seiner Komponenten mit einer hohen Wahrscheinlichkeit (oft 99,99 % oder besser) den Betrieb zu gewährleisten" - Wikipedia

BEISPIELE

- Internetdienste: Facebook, Google, E-Mails etc.
- Cloud Computing: VMs, Datenbanken, Loadbalancer, DNS Service etc.

BEISPIEL AUS DER AUTOMOBILINDUSTRIE:

- Automatisiertes Fahren: Level 0 - Level 5

VORAUSSETZUNG

Hochverfügbarkeit:

- redundante Komponenten / Funktionen
- ausfallsichere 'Connected Services'

HIGH AVAILABILITY

Die Nutzer erwarten von Systemen/Diensten/Funktionen die sie täglich / regelmäßig nutzen, dass sie immer erreichbar sind. Als hochverfügbar bezeichnet man Systeme die 99,99% und mehr "uptime" verfügen

HIGH AVAILABILITY

Was bedeutet "99,99%" (four nines)?

Level of Availability	Percent of Uptime	Downtime per Year	Downtime per Day
1 Nine	90%	36.5 days	2.4 hrs.
2 Nines	99%	3.65 days	14 min.
3 Nines	99.9%	8.76 hrs.	86 sec.
4 Nines	99.99%	52.6 min.	8.6 sec.
5 Nines	99.999%	5.25 min.	.86 sec.
6 Nines	99.9999%	31.5 sec.	8.6 msec

WIE BERECHNET MAN DIE VERFÜGBARKEIT

Availability = (uptime / uptime + downtime) * 100%

WIE BERECHNET MAN DIE VERFÜGBARKEIT

oder über

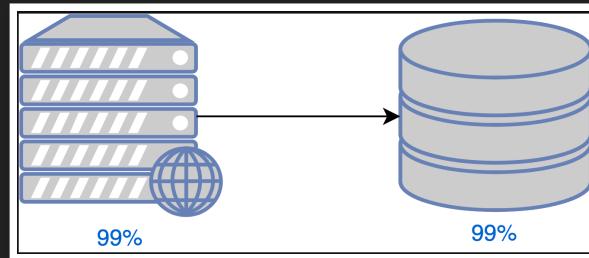
- Availability = $(MTBF / (MTBF + MTTR)) * 100\%$
- "Mean Time Between Failures"(MTBF)

$$MTBF = \frac{\sum (\text{start of downtime} - \text{start of uptime})}{\text{number of failures}}$$

- "Mean Time To Repair" (MTTR)

VERFÜGBARKEIT: KOMPONENTEN IN SERIE / GEKOPPELT

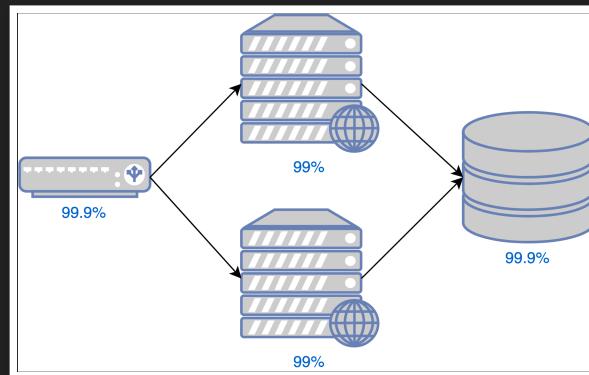
Die Systemkomponente ist erreichbar wenn beide Teilkomponenten erreichbar sind.



$$\text{Availability} = A_x * A_y$$

VERFÜGBARKEIT: KOMPONENTEN IN PARALLELBETRIEB

Die Systemkomponente ist erreichbar wenn min. eines der Teilkomponenten erreichbar ist.



Availability = ?

WIE ERREICHT MAN EINE HOHE VERFÜGBARKEIT ?

Redundanz!

- Hot Redundancy: Komponenten in Parallelbetrieb (Active/Active)
- Warm Redundancy: redundante Komponente im passiv Betrieb / Stand-By
- Cold Redundancy: redundante Komponente im "Lager", muss in Betrieb genommen werden

WIE ERZEUGT MAN REDUNDANZ ?

Welche Komponenten können ausfallen ? Welche Ebenen der Datenverarbeitung gibt es ?

DATACENTER REDUNDANZ

- Regions, Availability Zones
- redundante Strom-Generatoren, Internet Anbieter
- Kühlungssysteme
- Geo-Ip Routing, Traffic Routing um Daten zu verteilen

BEISPIEL: AZURE HIGH AVAILABILITY

Realisierung: Regions & Zones & Sets

NETZWERK REDUNDANZ

Redundante

- Router, Switches, Netzwerkkarten
- Loadbalancer, Firewalls
- Übertragungswege (Kabel - wie werden Kabelkanäle verlegt ? Wie werden Signale angenommen, Ethernet und Wifi)

SERVER REDUNDANZ

Redundante:

- Festplatten: RAID
- CPU, Motherboards, RAM Sticks
- Stromversorgung

SOFTWARE REDUNDANZ

Redundante:

- Apps- / Service-Instanzen
- Datenbank-Instanzen, Caches

"LAYER 8" REDUNDANZ

- Menschen ...
- ... werden krank, gehen in den Urlaub

WIE DESIGNT MAN EIN HA SYSTEM ?

- Redundanz zieht sich durch alle Instanzen hinweg ...
- welche Probleme treten dabei auf ?
- mein System soll HA und konsistent sein ! (geht das ?)

PROBLEME: VERTEILTE SYSTEME

- Beispiel: Online Banking, Überweisung über Web-Site
- Anforderung: das System muss immer erreichbar sein, die Transaktionen sollen konsistent sein
- Verteilte Systeme bearbeiten Daten
- manche Daten sind "stateful", manche sind "flüchtig", manche Daten sollen persistiert werden

PROBLEME: VERTEILTE SYSTEME

- stateful: Login Session, Überweisungsdaten die über mehrere Seiten hinweg verwendet werden
- flüchtig (bspw. Daten die On-Demand errechnet werden): aktuelle Uhrzeit
- persistente Daten: Kontostand, Kunden-Daten

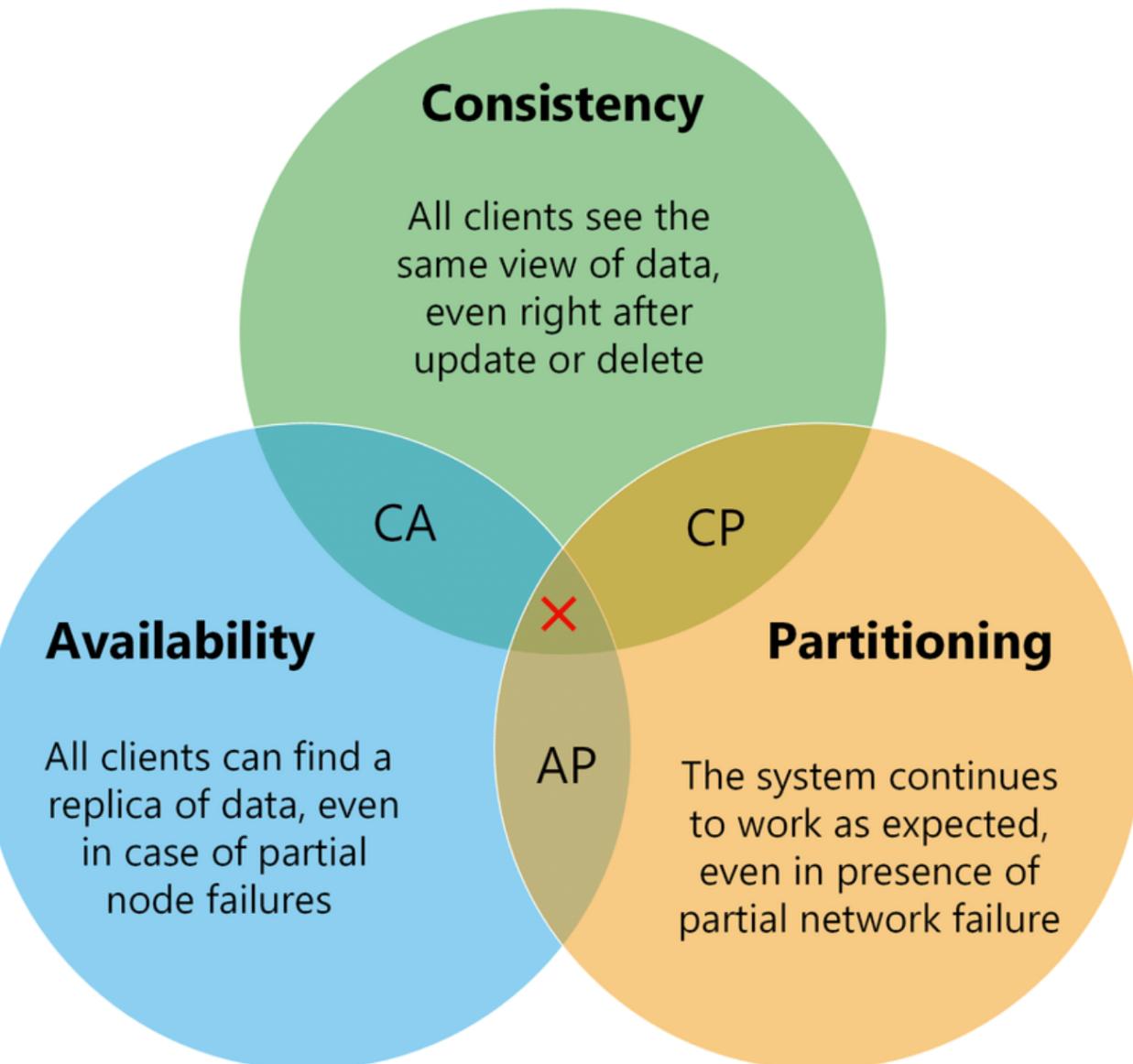
PROBLEME: VERTEILTE SYSTEME

- Um High Availability zu erreichen, betreibt man einen Cluster bzw. mehrere Instanzen einer Applikation
- Stateful-Daten müssen über alle Applikationsinstanzen hinweg verteilt (repliziert) werden
- zu persistierende Daten müssen in Datenbanken abgespeichert werden
- Auslagerung in einer Datenbank oder Cache oder Netzwerk Storage (Festplatte)

PROBLEME: VERTEILTE SYSTEME

- wie erreicht man eine hohe Verfügbarkeit bei Datenbanken ?
- Daten werden mehrfach (repliziert) und verteilt (sharding) abgespeichert
- wie erreicht man Konsistenz ? (alle Anfragen gegen die Datenbank erhalten das gleiche Ergebnis)
- Daten werden synchronisiert (abgespeichert)
- Was für ein Problem haben wir ?

CAP - THEOREM FÜR VERTEILTE SYSTEME

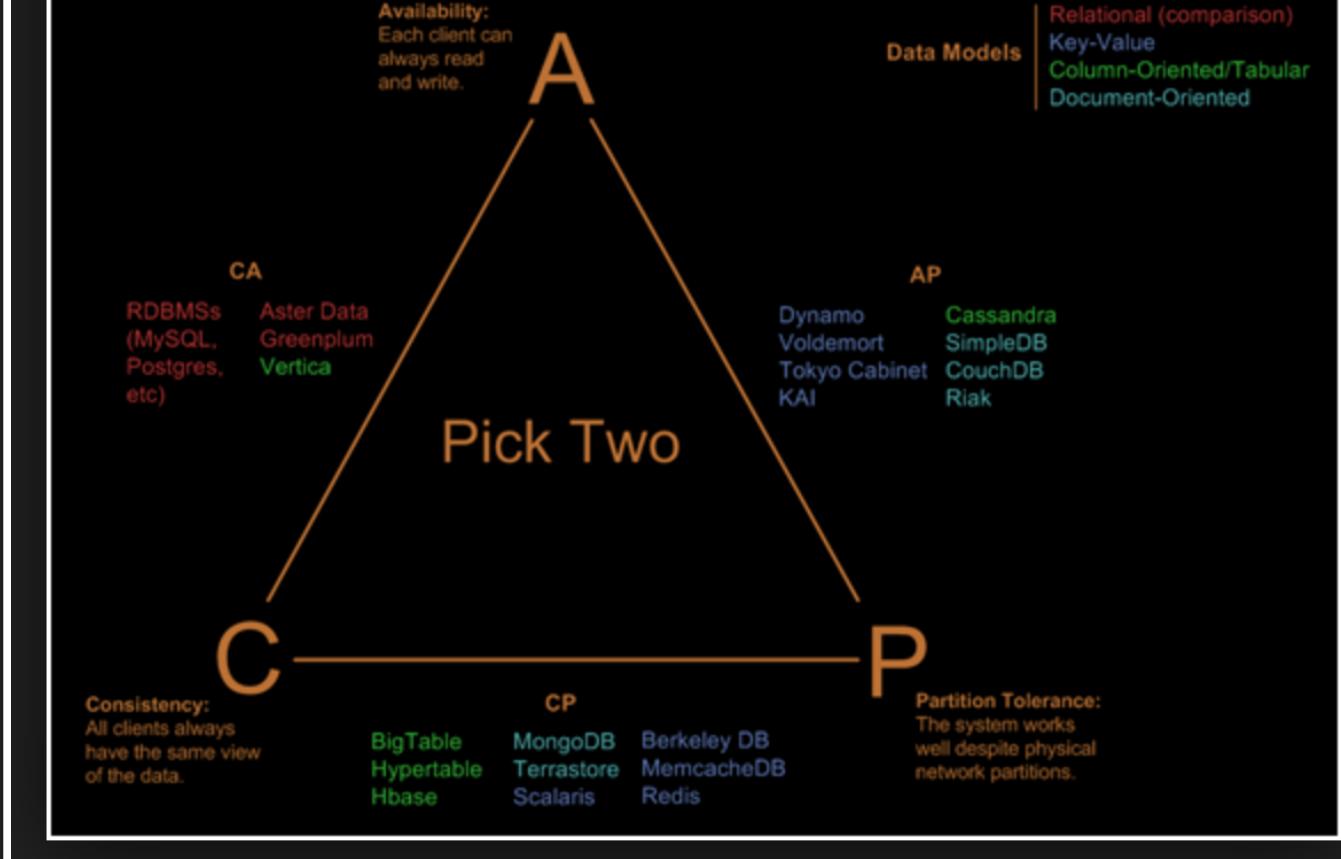


CAP - THEOREM FÜR VERTEILTE SYSTEME

CAP - FALSCH: PICK 2 OUT OF 3

Netzwerk-Ausfälle sind die Realität! Entweder C oder A!

Visual Guide to NoSQL Systems



<https://codahale.com/you-cant-sacrifice-partition-tolerance/>

FALLBEISPIEL: MONGODB

- single Master, Master-Slave DB
- Master = reads/writes, genannt "primary node"
- Slave = read only, genannt "secondary nodes"
- consistency über availability

FALLBEISPIEL: RELATIONELLE DATENBANKEN

- Bsp: MySQL, PostgreSQL etc.
- consistency über availability
- ACID (Atomicity, Consistency, Isolation, Durability)
Transactions

AVAILABILITY ODER CONSISTENCY

- SQL vs NoSQL Datenbanken
- ACID vs BASE (Basically Available, Soft state, Eventual consistency)
- Eventual consistency = keine Consistency
- irgendwann werden alle Teilnehmer konsistent
- für HA und Skalierung

ABSCHLUSSFRAGE

- Was ist Bitcoin für ein System ? CP oder AP ?

FRAGEN ZUR KLAUSUR ?