



Was Entwickler über SQL Server-Datenbanken wissen müssen



Thorsten Kansy (tkansy@dotnetconsulting.eu)

Meine Person- Thorsten Kansy

Freier Consultant, Software Architekt,
Entwickler, Trainer & Fachautor



Azure Cosmos DB



Mein Service- Ihr Benefit

- Individuelle Trainings
- Projektbegleitung
- Beratung
 - Problemanalyse und Lösungen
 - Technologieentscheidungen



Agenda

- Allgemeines
- Der große Abgrund
- Datentypen
- Collation
- Wie Daten organisiert werden
- Sichten, Prozeduren & Co
- Indizes-Mythen



Agenda II

- Das Multi-User-Dilemma
- Das Abfrage-Dilemma
- Das Index-Dilemma
- Snapshots
- Sicherheit
- ...



Allgemeines

The background image shows the Mausoleum of the Prophet (Nabawi Mausoleum) in Medina, Saudi Arabia. The building is constructed from light-colored stone and features a large, prominent dome. The entrance is a deep, arched doorway with intricate carvings. A large crowd of people, including men, women, and children, is gathered in the courtyard in front of the entrance. The ground is paved with a geometric pattern of light and dark tiles. The scene is set outdoors, with trees visible on the left and right sides of the building. A semi-transparent blue banner is overlaid across the middle of the image, containing the word "Allgemeines" in white text.

Allgemeines

Server und Client haben nur sporadische Verbindungen

Große Datenbanken sind nicht automatisch langsam, kleine nicht schnell

Bessere Hardware bedeutet nicht automatisch bessere Performance

Optimierung ist nicht unendlich möglich

Der Server ist sprach-neutral

Und nein, an schlechter Performance ist nicht immer der SQL Server Schuld





Der große Abgrund

Der große Abgrund

Zugriffe auf den Server sind langsam!

1.000 Milisekunden (ms)

1.000.000 Mikrosekunden (μ s)

1.000.000.000 Nanosekunden (ns)

RAM (ns) > SSD (μ s) > HDD(ms) > Server (ms)





Datentypen

Datentypen

CLR und SQL Server Datentypen
sind nicht 100% kompatibel

SQL Server	.NET
bigint	long
binary	byte[]
bit	bool
char	string
date	DateTime
datetime	DateTime
datetime2	DateTime
datetimeoffset	DateTimeOffset
decimal	decimal
filestream	byte[]
float	double
geography	Microsoft.SqlServer.Types.SqlGeography
geometry	Microsoft.SqlServer.Types.SqlGeometry
hierarchyid	Microsoft.SqlServer.Types.SqlHierarchyId
image	byte[]
int	int
money	decimal
nchar	string
ntext	string
numeric	decimal
nvarchar	string
real	Single
rowversion	byte[]
smalldatetime	DateTime
smallint	short
smallmoney	decimal
sql_variant	object
text	string
time	TimeSpan
timestamp	byte[]
tinyint	byte
uniqueidentifier	Guid
varbinary	byte[]
varchar	string
xml	string
else	object



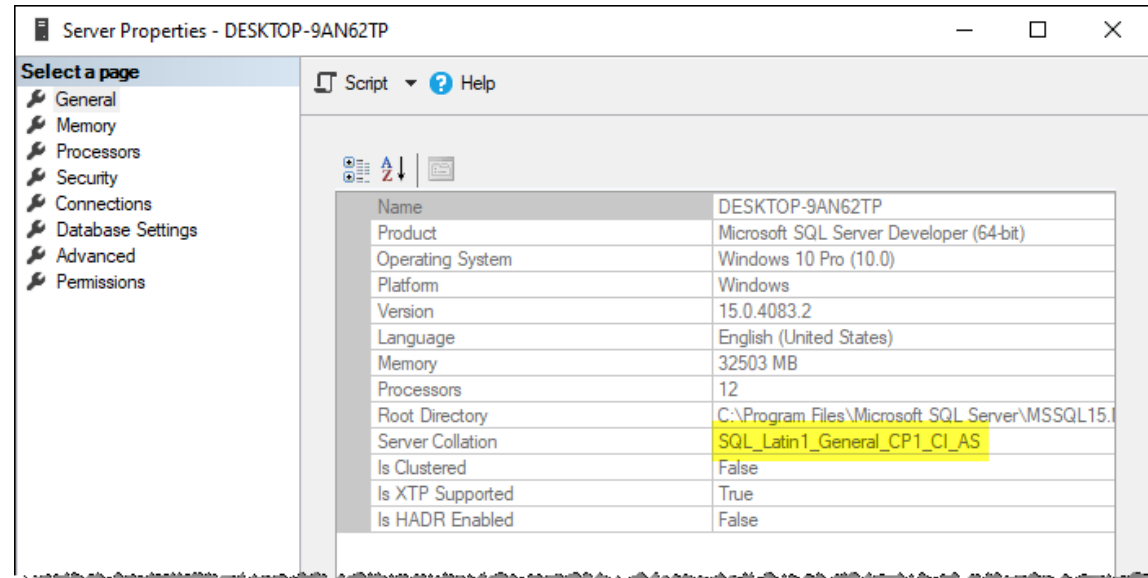


Collation



Collation

Groß-/ Kleinschreibung
Sortierung



Name	Value
Name	DESKTOP-9AN62TP
Product	Microsoft SQL Server Developer (64-bit)
Operating System	Windows 10 Pro (10.0)
Platform	Windows
Version	15.0.4083.2
Language	English (United States)
Memory	32503 MB
Processors	12
Root Directory	C:\Program Files\Microsoft SQL Server\MSSQL15.I
Server Collation	SQL_Latin1_General_CP1_CI_AS
Is Clustered	False
Is XTP Supported	True
Is HADR Enabled	False

CI Case insensitive / CS Case sensitive

AI Accent insensitive / AS Accent sensitive





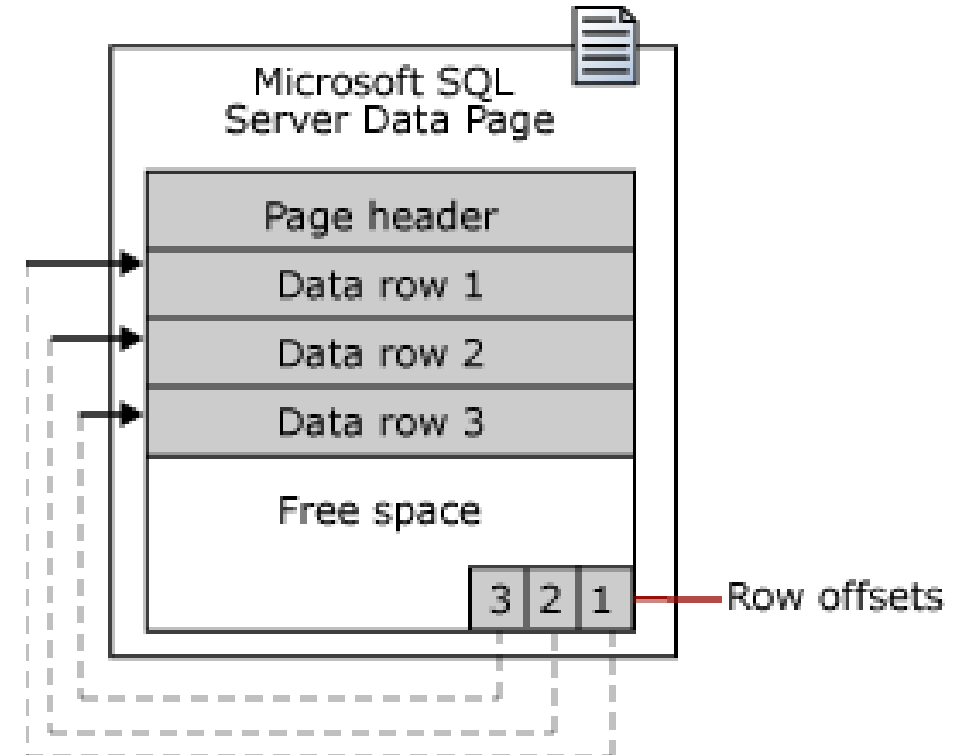
Wie Daten organisiert werden



Wie Daten organisiert werden

Pages (8k) -> Extents (64kb)

8.060 Bytes Nutzlast pro Page





Sichten, Prozeduren & Co.

Sichten, Prozeduren & Co.

SQL Server kann viel mehr als nur Tabellen

- Sichten
- Stored Procedures
- Funktionen
- Trigger
- Volltextsuche
- ...





Indizes-Mythen

Indizes-Mythen

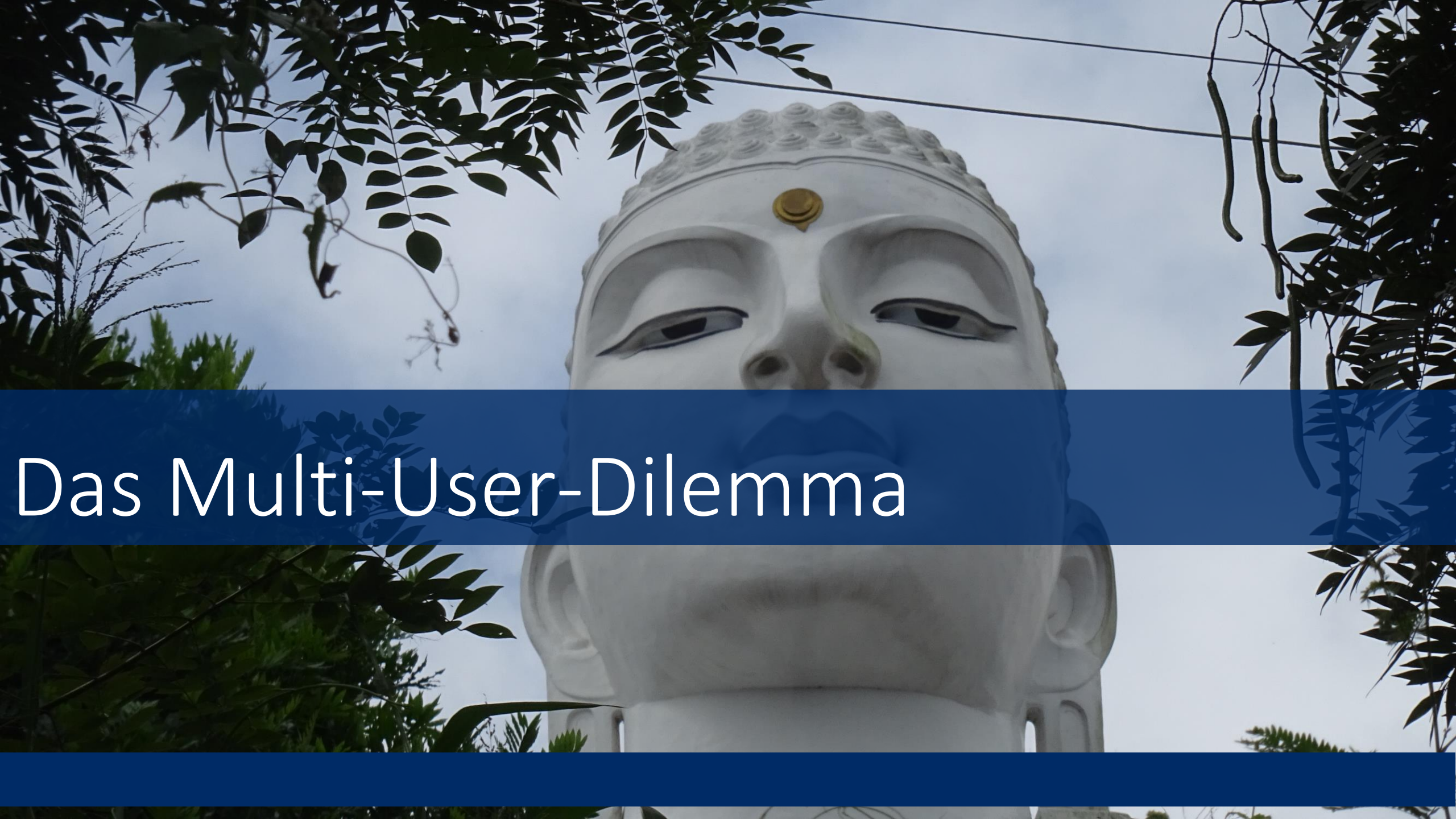
Indizes sind optional

Je mehr Indizes, desto besser (“Viel hilft viel”)

Indizes sind nicht statisch- einmal konzipieren reicht nicht!

...





Das Multi-User-Dilemma

Das Multi-User-Dilemma

Viele “chaotische” Zugriffe geschehen parallel und müssen vor den anderen Zugriffen abgegrenzt (“geschützt”) werden. Dies erfordert Sperren und verursacht längere Zugriffe.

SQL Server

- blockt Ressourcen während der Zugriffe
- Ressourcen (Row, Pages, Extends, Tabellen, Datenbanken, ...)
- Zugriffe (Exklusiv, Shared, ...)
- Zugriffe können koexistieren oder auch eben nicht (=> Locks)





Das Abfrage-Dilemma

Das Abfrage-Dilemma

“Starre”, immer wiederkehrende Abfragen sind für die Optimierung optimal. Viele Anwendungen möchte aber eher “flexible” Abfragen durchführen

Flexibel in

- Zeit
- Inhalt



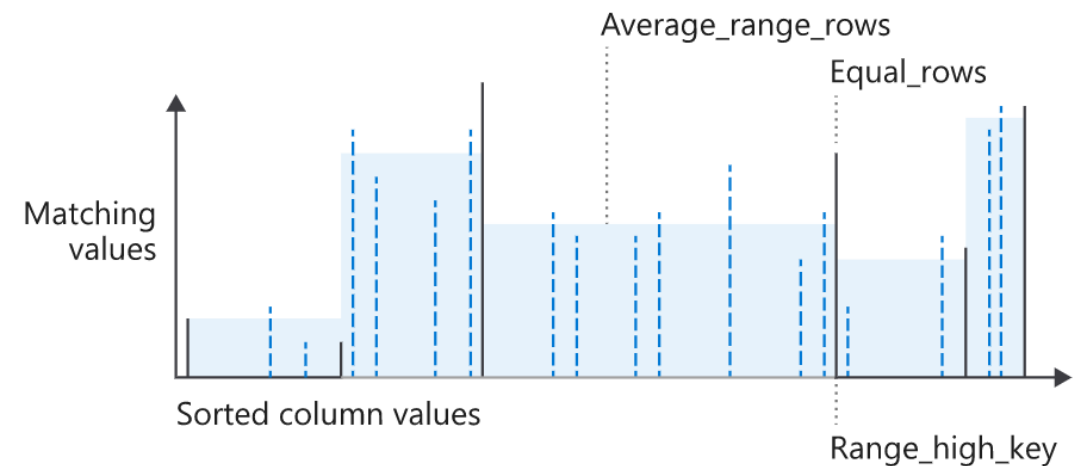


Das Index-Dilemma

Das Index-Dilemma

Der SQL Server muss möglichst schnell entscheiden, ob und welchen Index er verwendet und das kostet Zeit.

Histogramme geben Auskunft über die Verteilung der Daten





Snapshots

Snapshot

Datenbanken „einfrieren“ und wiederherstellen- schneller als Backups, aber nicht eigenständig.

```
CREATE DATABASE AdventureWorksSnapshot ON ( ... )  
AS SNAPSHOT OF AdventureWorks;
```

```
ALTER DATABASE AdventureWorks SET SINGLE_USER WITH ROLLBACK IMMEDIATE;  
RESTORE DATABASE AdventureWorks FROM DATABASE_SNAPSHOT =  
'AdventureWorksSnapshot';
```





Sicherheit

Sicherheit

Zugriffsrechte

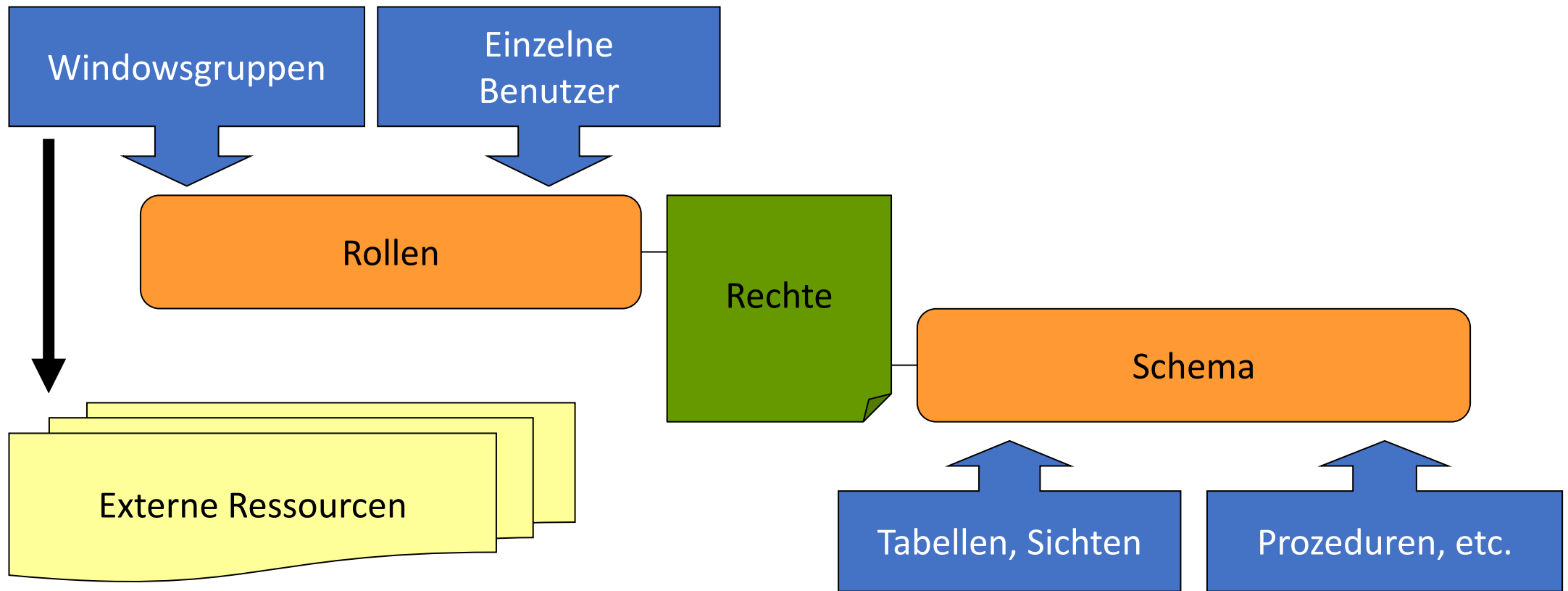
- DML
- DDL

Verschlüsselung

- Datenbank
- Daten
- Netzwerk



Best Practices



Fragen?

Links



www.dotnetconsulting.eu



[@Tkansy](https://twitter.com/Tkansy)



tkansy@dotnetconsulting.eu