

DocumentDB

Basic concepts and queries

Agenda

Basic Concepts

Configuration

Scaling & Partitioning

Stored Procedures, UDFs & Triggers

LINQ & DocumentDB



Ing. Andreas Pollak

SpectoLogic[®] e.U.

www.spectologic.com

Basic Concepts

NoSQL?

Key-Value
Document
Wide-Column
Graph

Why?

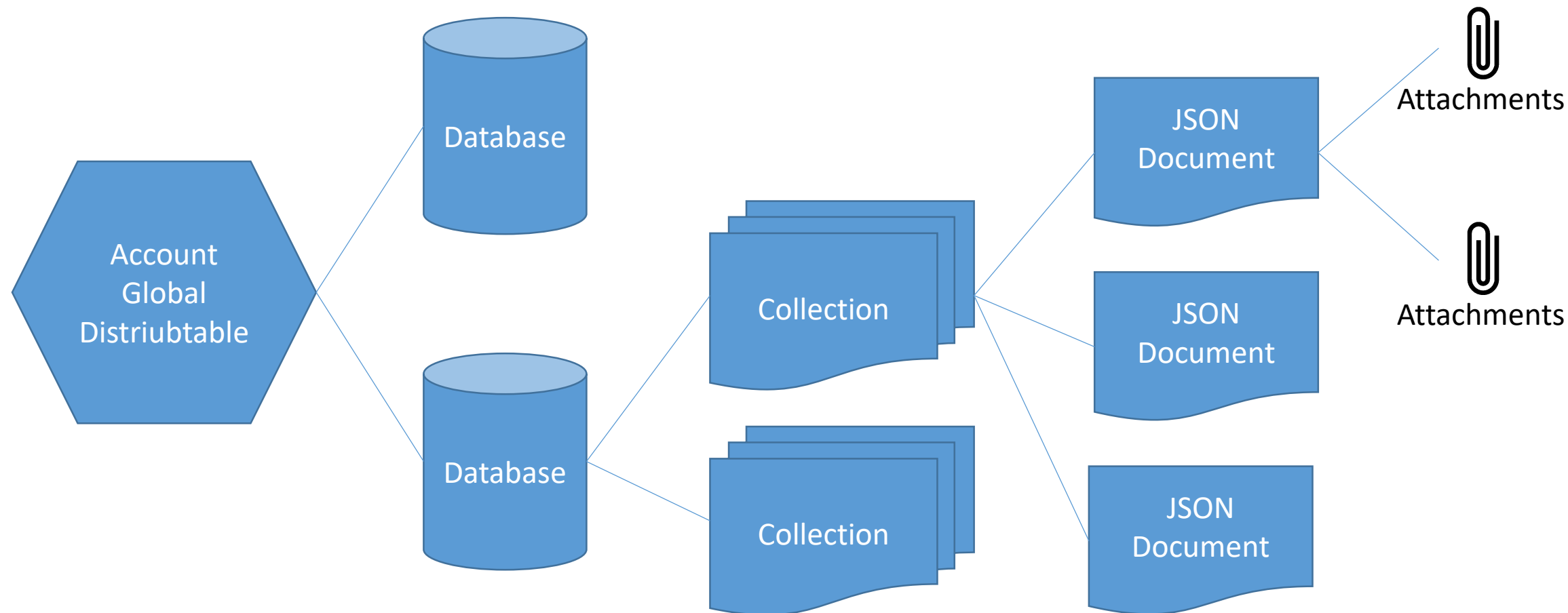
Performance
Scaling
Consistency

Document-Store

```
{
  "id": "ew12-res2-234e-544f",
  "title": "post title",
  "date": "2016-01-01",
  "body": "this is an awesome post stored on NoSQL",
  "createdBy": "User",
  "images": ["http://myfirstimage.png", "http://mysecondimage.png"],
  "videos": [
    {"url": "http://myfirstvideo.mp4", "title": "The first video"},
    {"url": "http://mysecondvideo.mp4", "title": "The second video"}
  ],
  "audios": [
    {"url": "http://myfirstaudio.mp3", "title": "The first audio"},
    {"url": "http://mysecondaudio.mp3", "title": "The second audio"}
  ]
}
```

Beispiel von <https://docs.microsoft.com/de-de/azure/documentdb/documentdb-nosql-vs-sql>

Structure



Collections

CollectionID

Storage Capacity Options

Partition Key

Throughput Capacity (RU/s)

* STORAGE CAPACITY (up to 10TB and more) ⓘ

10 GB	250 GB	Custom
-------	--------	--------

THROUGHPUT CAPACITY (RU/s) ⓘ

400 ✓

Request Units (RU)

Based on the required resources

Deterministic

Response Header

RU/s - Throttling

Highly efficient

1 Doc - 1KB Read = 1 RU

1 Doc -16KB Read = 2.19 RU

Query 100 Docs 1KB = 51 RU

Pricing

Combination of Capacity + RU/s

Only consumed capacity

Minium ~25 € / Month

10GB = ~ 2 € / Month

400 RU/s = 20 € / Month

2500 RU/s = ~125 € / Month

Scaling and Partitioning

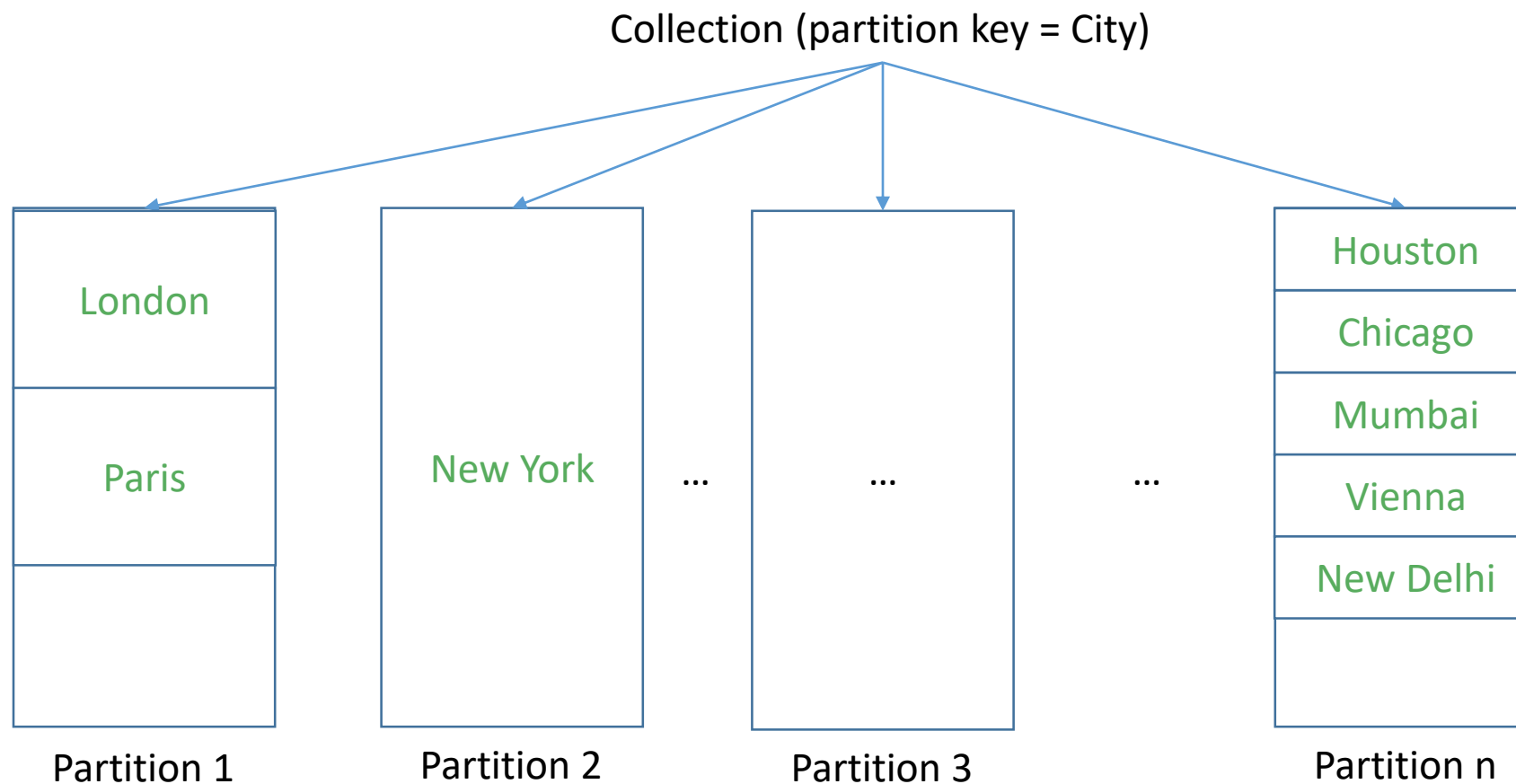
Scaling Strategies

Databases/Collection per Tenant

Collection Partitioning (e.g. Azure Stream Analytics)

Partitioned Collections

Partitioning



Finding the right partition key

Scenario based

Has wide range of values

Define as Json Path

Always indexed

/City
/City/District
/DeviceID

Increasing performance and efficiency

CRUD operations

Query

Crosspartition Query

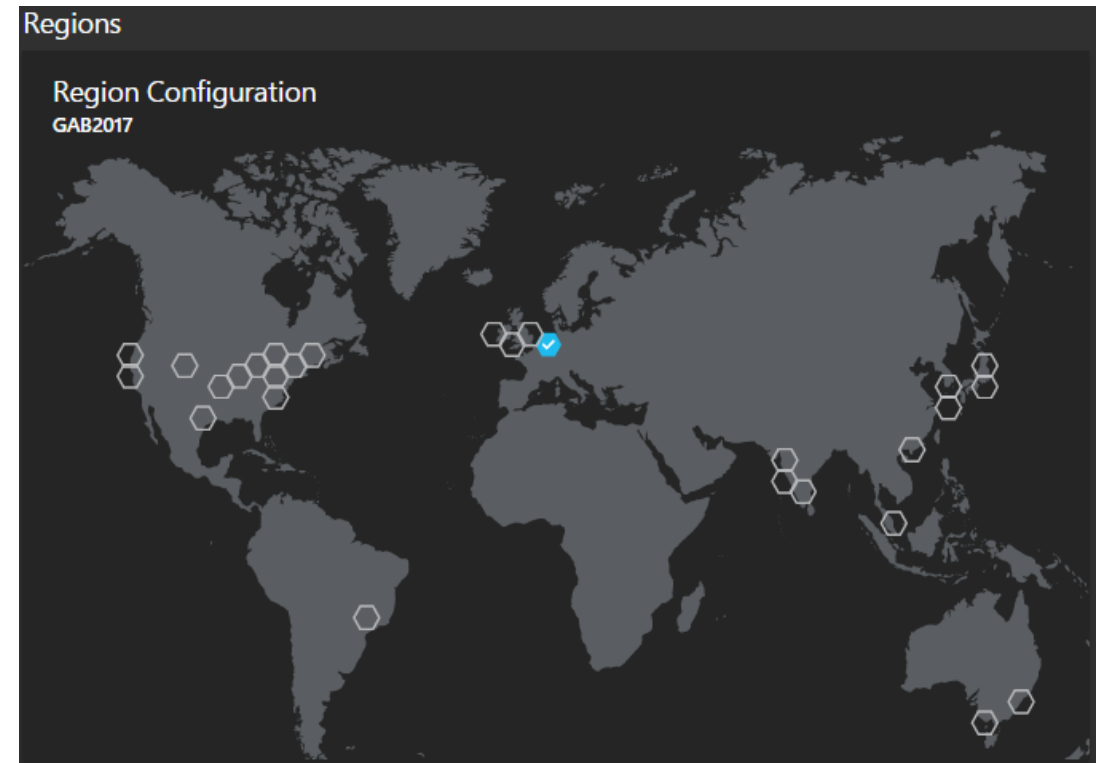
Global Distribution

Write / Read Region(s)

Read Performance

Consistency

Failovers



Consistency Levels

Strong

Bounded Staleness

Session

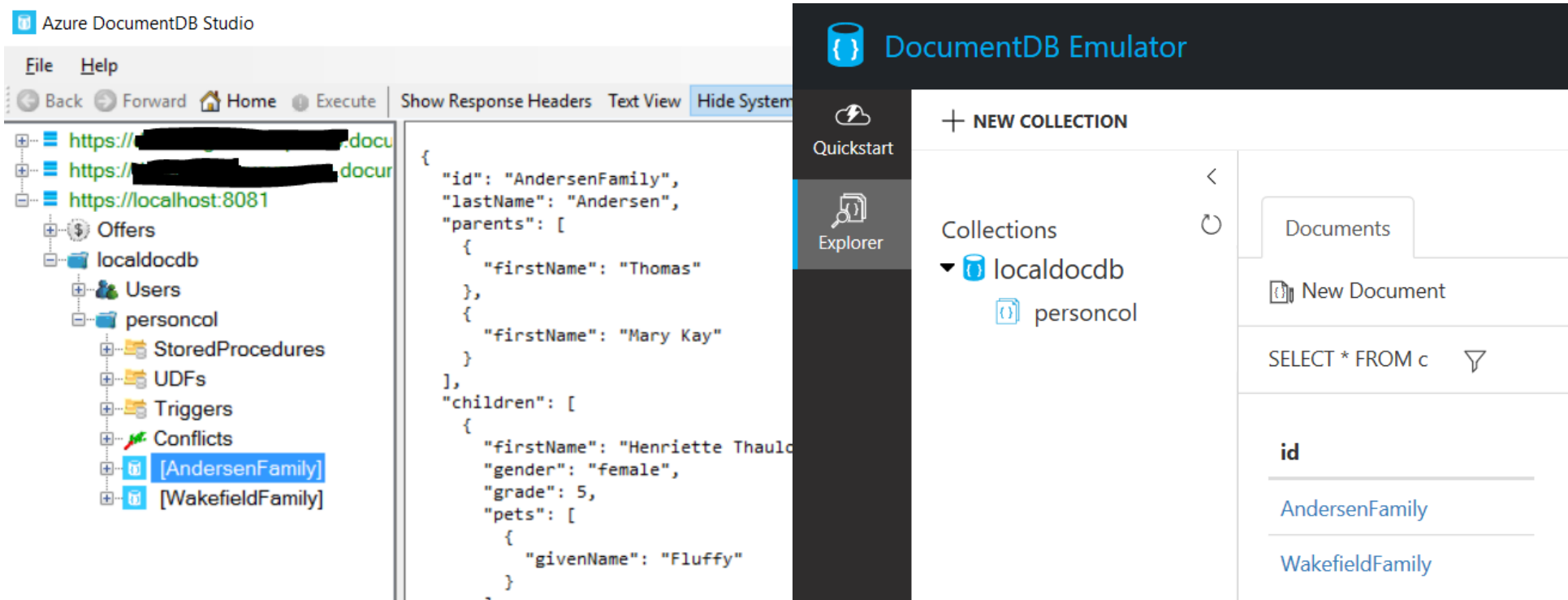
Eventual



DEMO

Creating a DocumentDB collection

Document Studio & DocumentDB Emulator



The image displays two overlapping software interfaces. On the left is the Azure DocumentDB Studio, showing a file explorer on the left with a tree view containing folders like 'Offers', 'localdocdb', 'Users', 'personcol', 'StoredProcedures', 'UDFs', 'Triggers', 'Conflicts', and documents '[AndersenFamily]' and '[WakefieldFamily]'. The main pane shows a JSON document with fields for 'id', 'lastName', 'parents', and 'children'. On the right is the DocumentDB Emulator, which has a sidebar with 'Quickstart' and 'Explorer' tabs. The main area shows a 'NEW COLLECTION' button, a 'Collections' list with 'localdocdb' expanded to show 'personcol', and a 'Documents' tab with a 'New Document' button. Below that is a query input field containing 'SELECT * FROM c' and a results table with columns 'id' and rows 'AndersenFamily' and 'WakefieldFamily'.

```
{
  "id": "AndersenFamily",
  "lastName": "Andersen",
  "parents": [
    {
      "firstName": "Thomas"
    },
    {
      "firstName": "Mary Kay"
    }
  ],
  "children": [
    {
      "firstName": "Henriette Thaulo",
      "gender": "female",
      "grade": 5,
      "pets": [
        {
          "givenName": "Fluffy"
        }
      ]
    }
  ]
}
```

Stored Procedures, Triggers

and UDF's

SQL-Queries based on T-SQL

Similar

ANSI SQL

SELECT

FROM

WHERE

AND/OR/NOT

Different

JavaScript – JSON Type System

JavaScript Expression Evaluation

JavaScript Functions (UDFs)

Hierarchical Navigation of JSON

Self-Joins

No Schema

DocumentDB query specialities

Operator Evaluation (no implicit conversions)

Parameterized SQL

SQL functions (mathematical, type-checking, string, array, spatial)

Aggregations

Availability of Aggregations

Most requested feature on user voice! No GROUP BY yet!

COUNT

MIN

MAX

SUM

AVG

DocumentDB Indexing

Autoindexing

Relational and hierarchical („paths“) queries

Tune with policies

- Automatic, manual

- Include/Exclude index paths

- Index types: **Hash** (precision), **Range** (precision), Spatial

- Index modes: Consistent, Lazy, None

DEMO

SQL Queries

JavaScript extensibility

UDFs

Stored Procedures

Triggers

DEMO

UDF (User Defined Function)

DEMO

SPROC (Stored Procedure)

DEMO

Trigger

Geospatial Data

Store & Query of GeoJson conform data

Index points

WGS-84 CRS

```
{  
  "type": "Point",  
  "coordinates": [ 31.9, -4.8 ]  
}
```

[longitude, latitude]

Geospatial Data with C#

Microsoft.Azure.Documents.Spatial

Point

Position [longitude, latitude, altitude, custom...]

LineStrings

Polygon

Supported geospatial functions

Support following functions of the OGC for querying...

ST_DISTANCE

ST_WITHIN

ST_INTERSECTS

ST_ISVALID

ST_ISVALIDDETAILED

```
SELECT f.id FROM Families f
WHERE
ST_DISTANCE(f.location,
             {'type': 'Point',
              'coordinates': [-73.8, 40.7]})
           < 30000
```

Linq to DocumentDB

DEMO

C# and the DocumentDB



THANK YOU

Specto
logic[®]

Resources

General

[Available NoSQL Databases](#)

[NoSQL vs. SQL – When do I use what?](#)

Request Units and Capacity

[Understanding Request Units](#)

[Document DB Capacity Planner](#)

Resources

Partitioning and Scaling

[Partitioning and Scaling](#)

[Consistency Levels](#)

SQL

[SQL Grammar Cheat Sheet](#)

[Document DB Query Playground](#)

DocumentDB

[.NET SDK DocumentClient Reference](#)

[DocumentDB .NET Sample Code](#)

[DocumentDB serverside JavaScript API](#)

Resources

Tools

[DocumentDB Emulator \(Free\)](#)

[Azure Document DB Studio](#)