



A to Z Cosmos DB

HASAN SAVRAN

Hasan Savran

BI MANAGER



About Me

- MS Data Platform MVP
- From Cleveland, USA
- BI Manager at Progressive Insurance
- 15 years Web Development
- 7 years Business Intelligence



<https://h-savran.blogspot.com/>



hasansavran



SavranWeb

TOOLS



Azure Cosmos DB Emulator



Azure Cosmos Data Migration Tool



Azure Portal



SQL Server 2019 & SSMS



Azure Cosmos DB ODBC Driver

A TO Z AZURE COSMOS DB AGENDA

BASICS OF NOSQL DATABASES
HISTORY OF COSMOS DB
RELATIONAL DB VS NOSQL
COSMOS DB INTERNALS
INTRODUCTION TO EMULATOR
COSMOS DB SECURITY
PARTITIONING



MULTI-MODEL API
CONSISTENCY LEVELS
INDEXING POLICIES
UNIQUE KEYS
REQUEST UNITS
QUERY PROCESSOR
BACKUP OPTIONS
TIME TO LIVE FUNCTION



STORED PROCEDURES
DATA MIGRATION TOOL
SQL API LANGUAGE
CHALLENGES WITH JSON
COSMOS DB REST API



SPATIAL DATA IN COSMOS DB
DATA MODELLING
JUPYTER NOTEBOOKS
USING VS CODE EXTENSION
USING ODBC DRIVER
SQL SERVER 2019 POLYBASE
HOW TO SETUP CHANGE FEED



COSMOS DB HISTORY

2010

Project
Florence



Internal project
for large scale
OLTP problems

2015

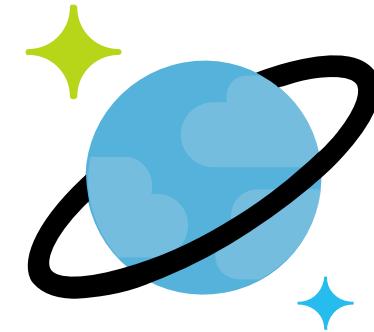
Azure
Document DB



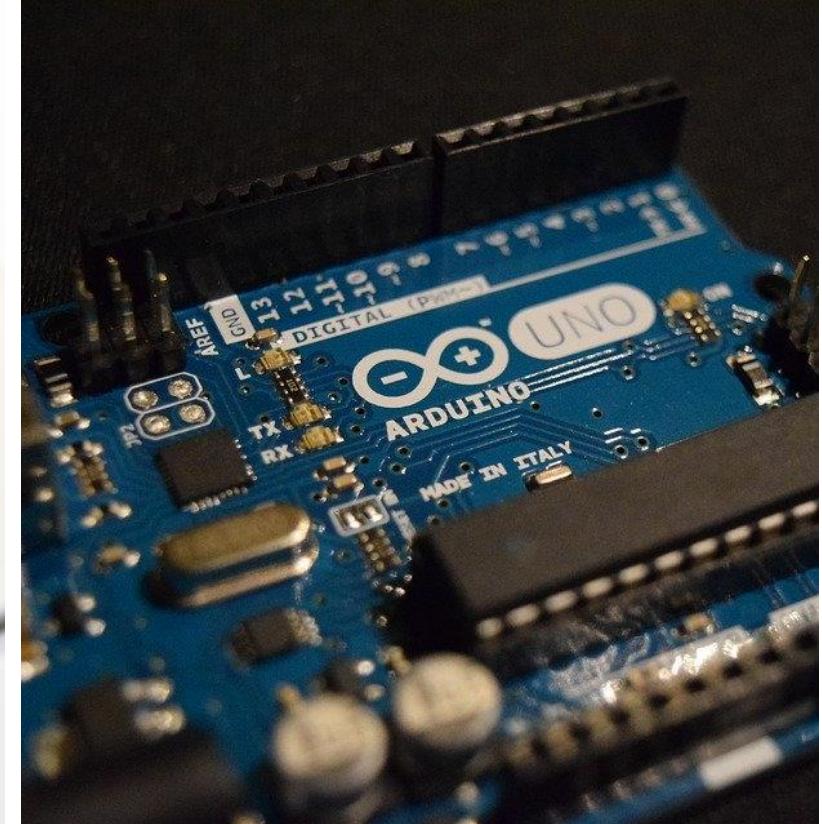
NoSQL database
for devs to store
JSON

2017

Azure
Cosmos DB



Global Dist.
Multi-Model



Use Cases



NoSQL Term
1998

Open-Source RDB with NO SQL
(Structured Query Language)

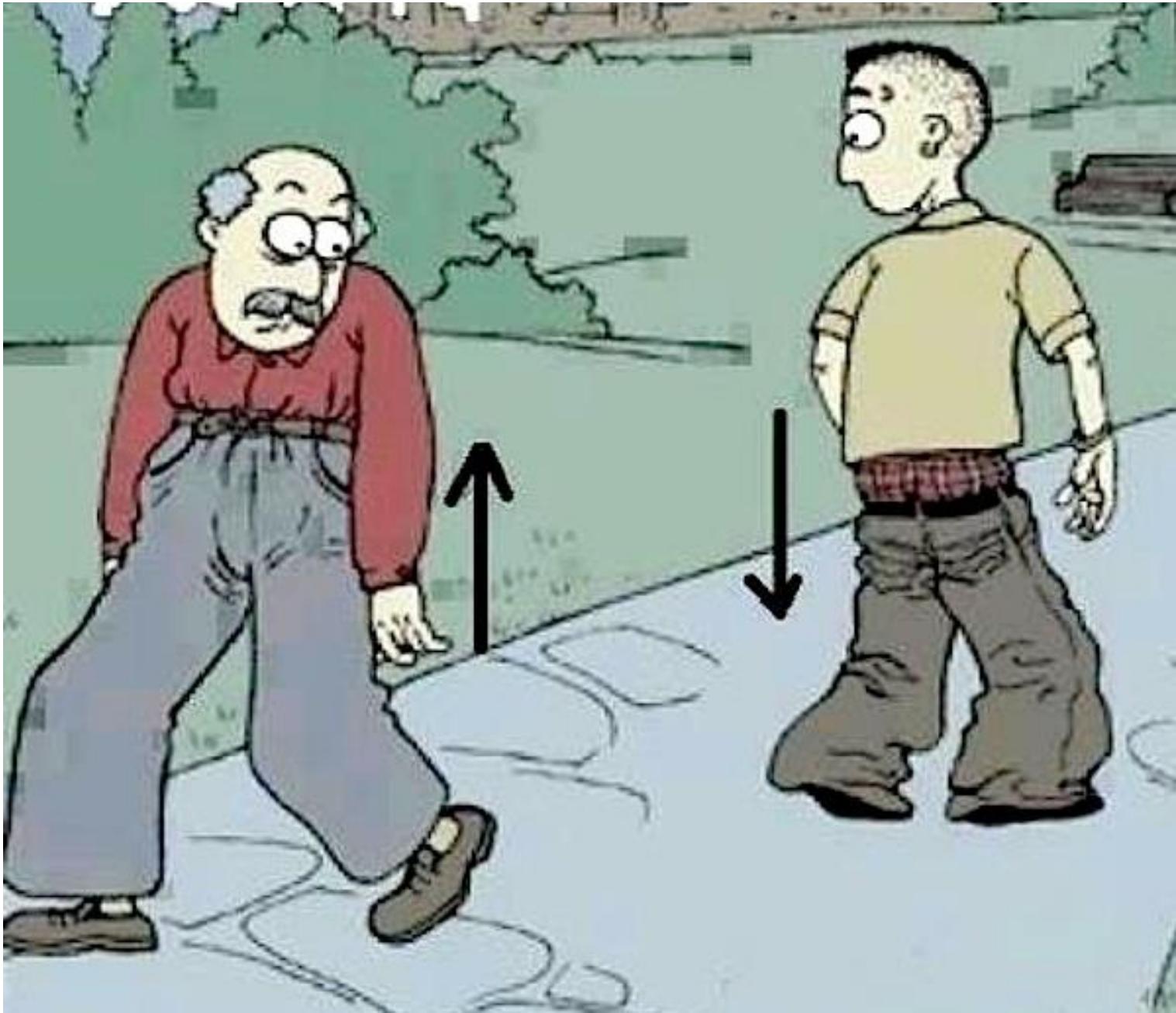
Johan Oskarsson reintroduced it in
early 2009 for Non relational
databases.

The name attempted to label the
emergence of an increasing number of
non-relational, distributed data stores

Carlo Strozzi

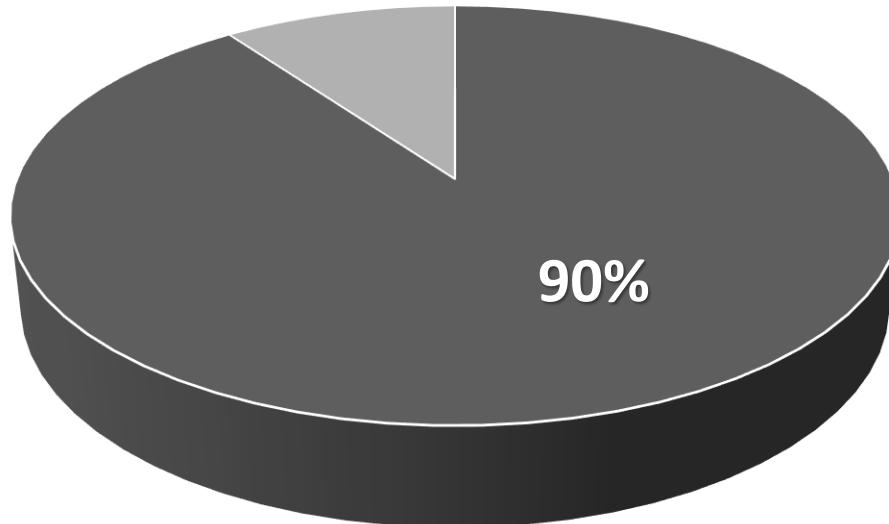
RDBMS VS NOSQL

- Joins
- Relation Attributes
- Centralized Data
- Expensive Always Available
- No Schema
- Aggregation
- Encryption
- Cloud
- Rapid Development



WHY DO WE NEED ANOTHER DATABASE SYSTEM?

DATA IN THE WORLD



500 million tweets

294 billion emails

4 PB of data by Facebook

4 TB of data created from each connected car

5 billion searches made

65 billion messages are sent on WhatsApp

WHY DO WE NEED ANOTHER DATABASE SYSTEM?



Different device types

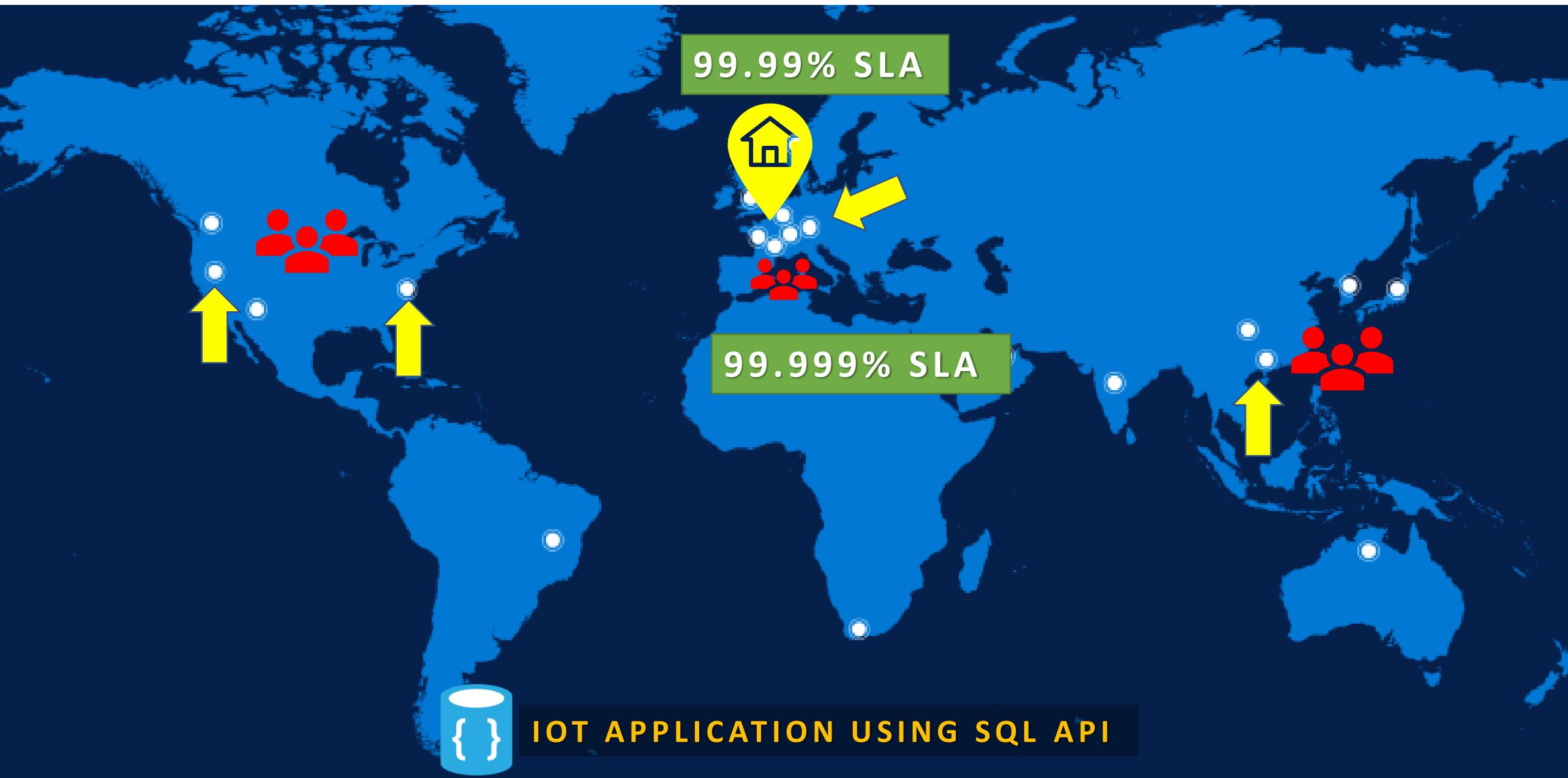
Customers expects more

Applications consume more data

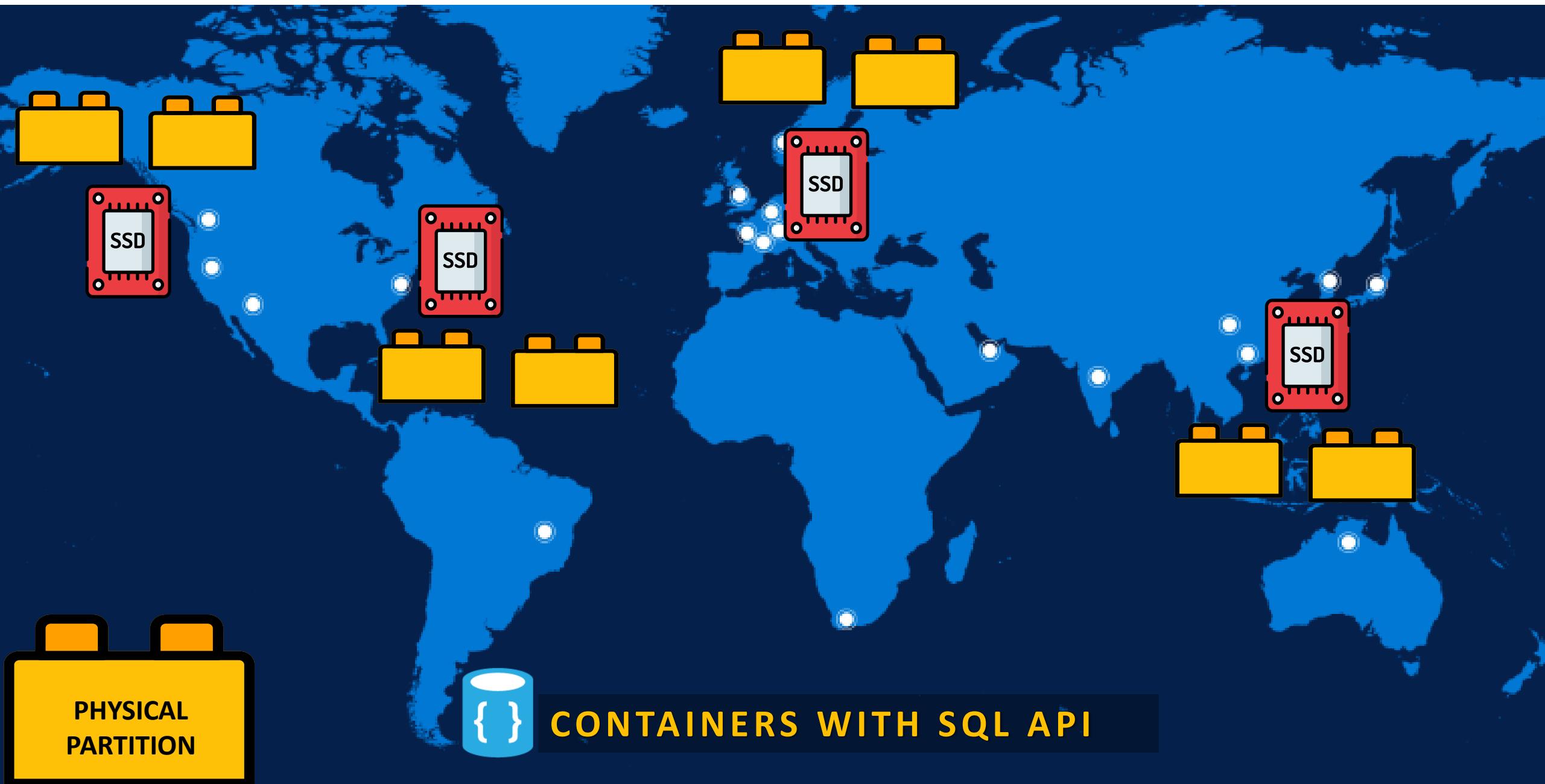
Applications generate more data

Globalization

Cosmos DB Internals



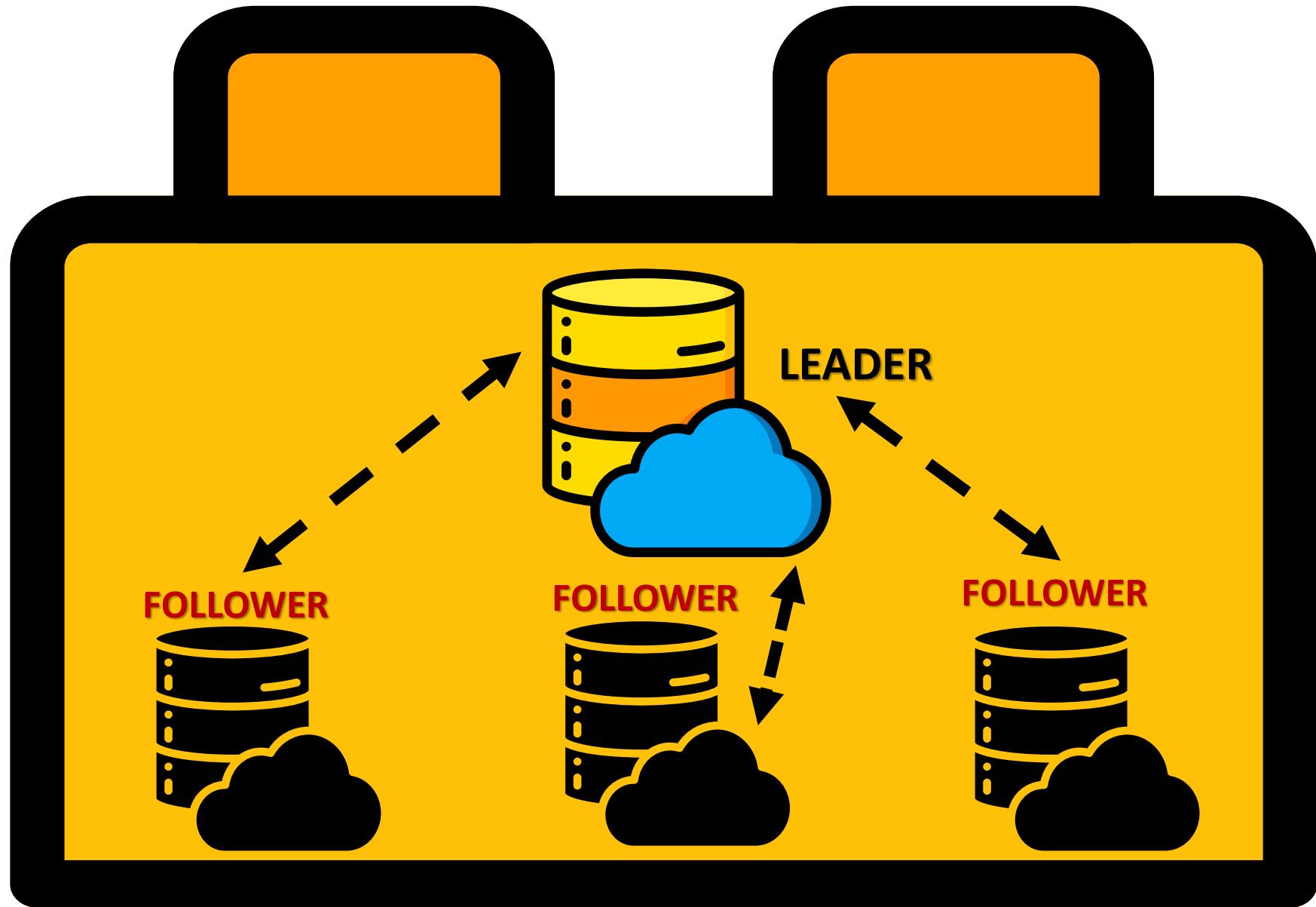
Cosmos DB Internals



Cosmos DB Internals

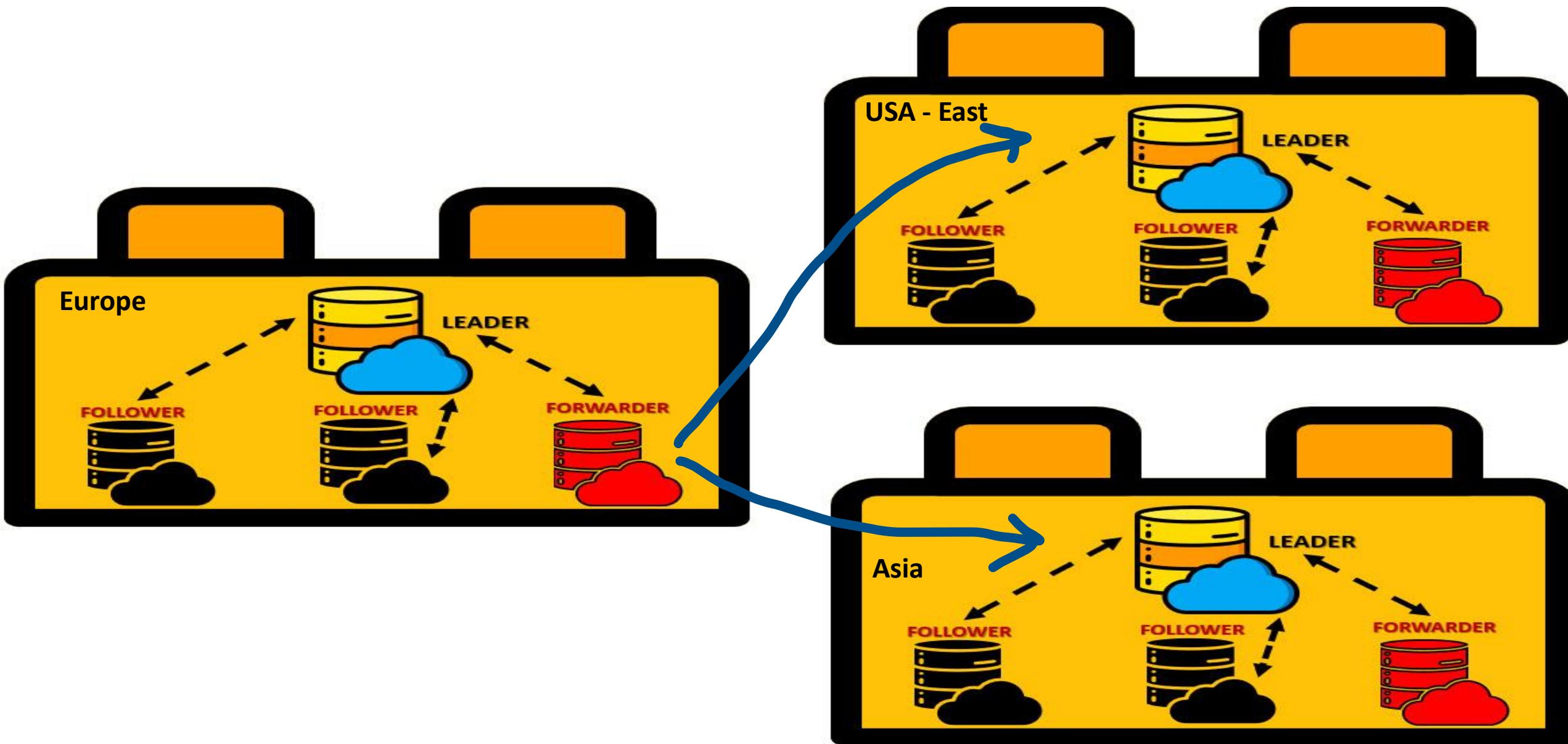
50 GB

10,000
RU



COSMOS DB PHYSICAL PARTITION WITHOUT GLOBAL DISTRIBUTION

Cosmos DB Internals



COSMOS DB PHYSICAL PARTITION WITH GLOBAL DISTRIBUTION

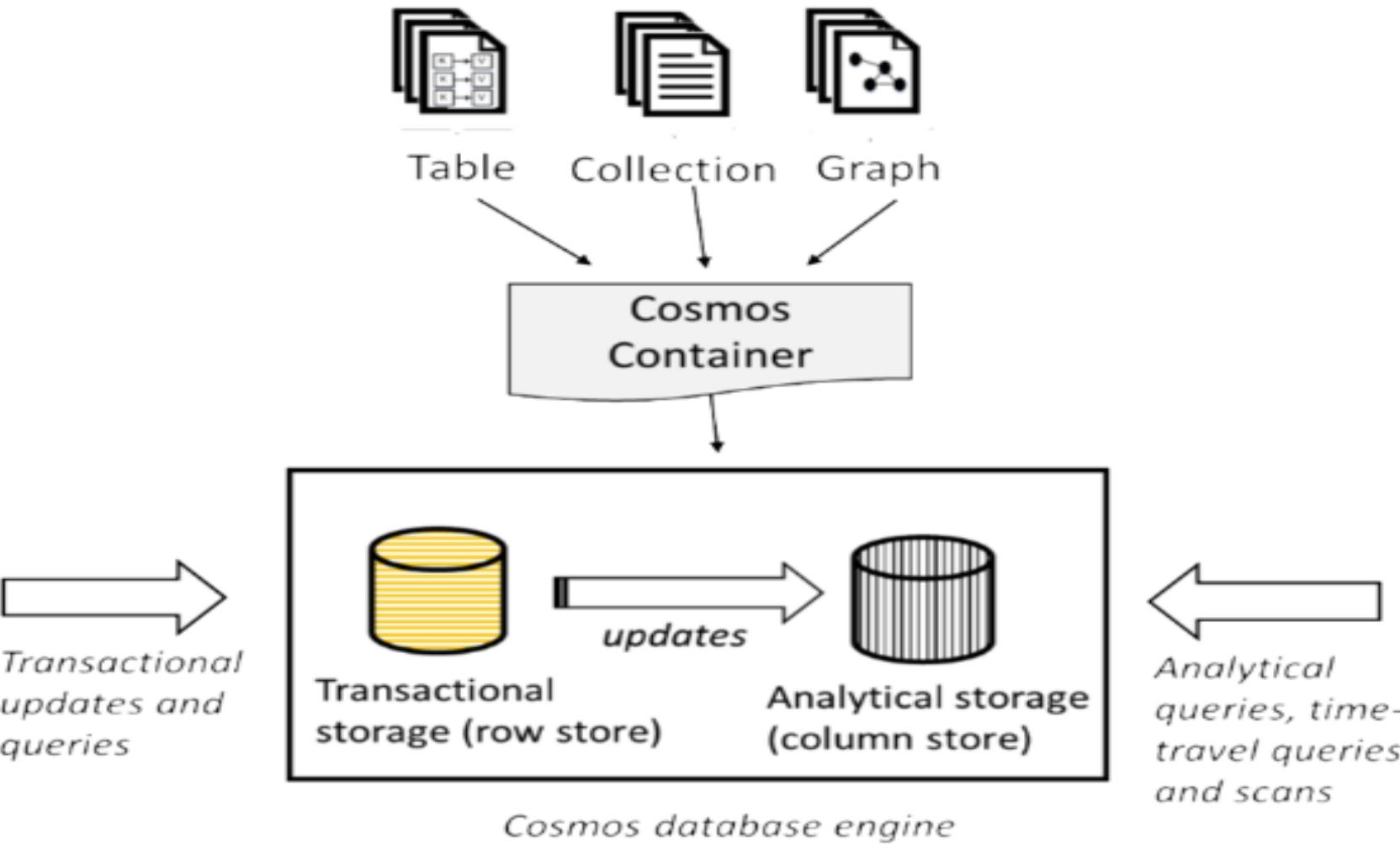
Cosmos DB Internals <CONFLICTS>



LAST WRITE WINS
CUSTOM

INSERT CONFLICTS
REPLACE CONFLICTS
DELETE CONFLICTS

Cosmos DB Internals <STORAGE>



Cassandra

SQL

SQL



API for MongoDB



Gremlin



ETCD



Table

Transactional workloads



Spark

Analytical workloads

Cosmos DB Internals <STORAGE>

ANALYTICS STORAGE



No ETL
Operations



Guaranteed to be
visible in 30 sec.



Globally
Distributed



Isolation between transactional
and analytical env.

TRANSACTIONS

PRICE

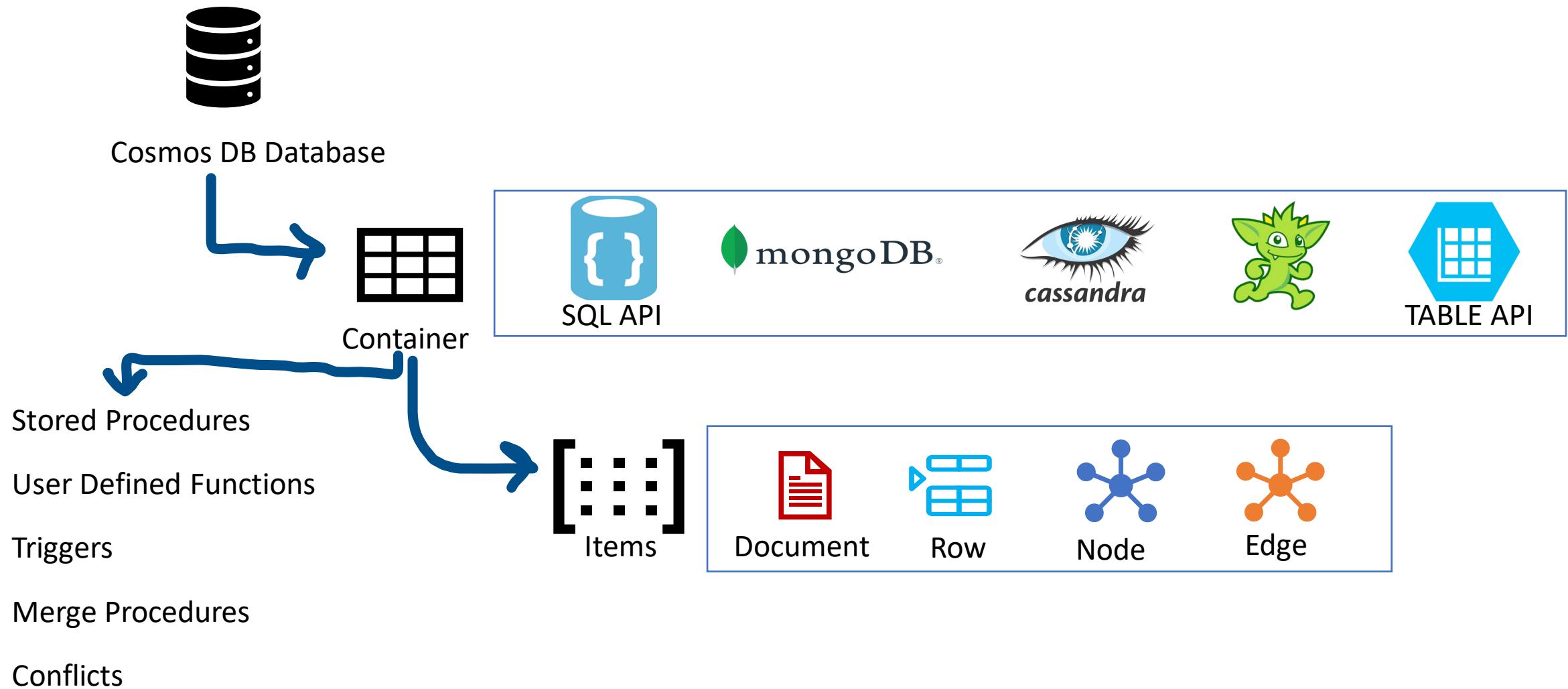
Write Operations (per 10,000 operations)	\$0.05
Read Operations (per 10,000 operations)	\$0.005

Cosmos DB Internals <STORAGE>

ANALYTICS STORAGE

Transactional Storage	Analytical Storage
20 GB limit for Logical Partitions	Unlimited storage limit for Logical Partitions
Row-oriented storage encoding	Column-oriented storage encoding (Apache Parquet format)
SQL, MongoDB, Cassandra, Gremlin, Tables, Etcd API	Apache Spark API
DefaultTimeToLive property controls retention	ColumnStoreTimeToLive controls retention

Cosmos DB Internals

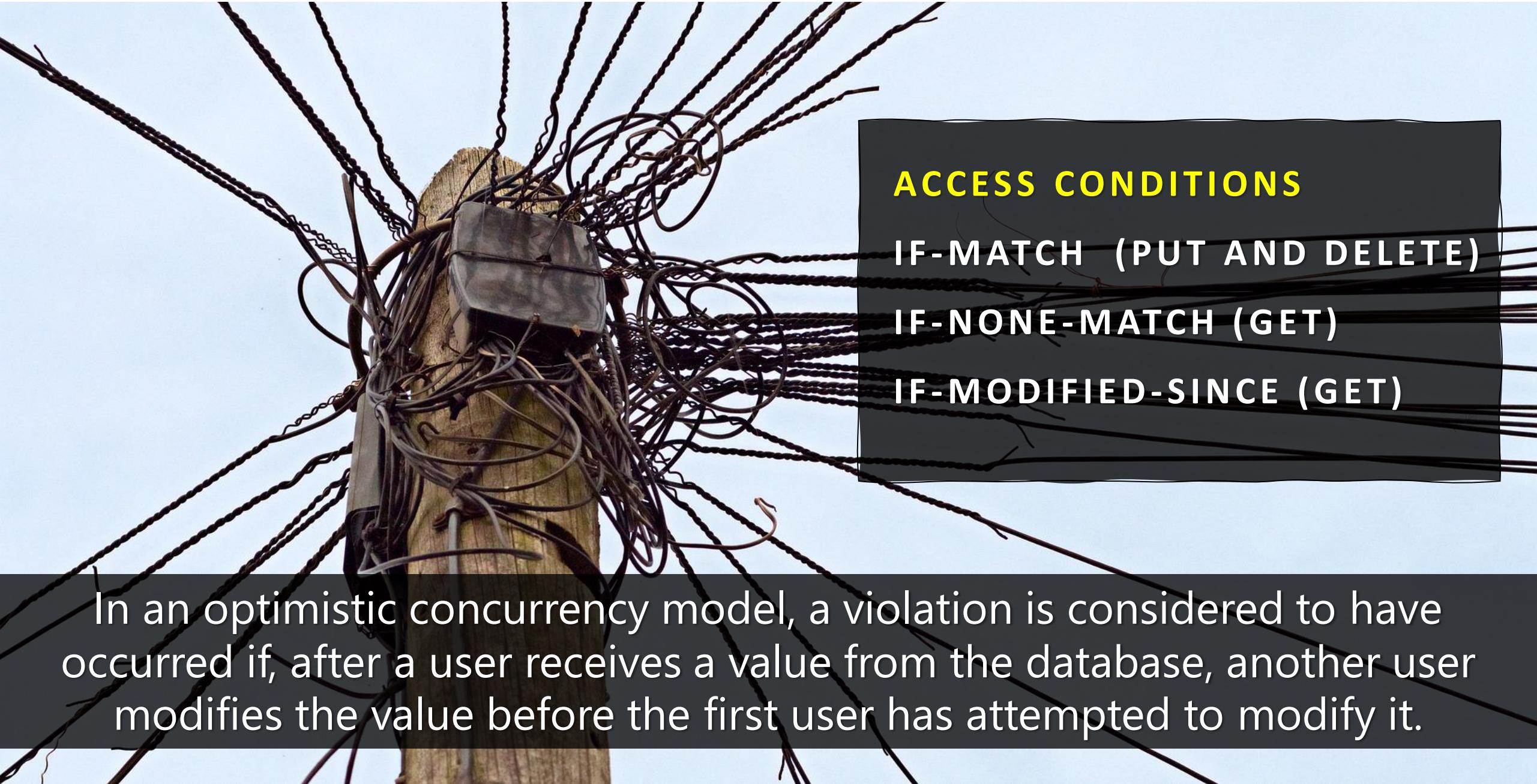


Cosmos DB Internals

SYSTEM DEFINED PROPERTY	PURPOSE
_rid	Unique identifier of the item
_etag	Entity Tag used for optimistic concurrency control
_ts	Last updated timestamp of the item
_self	Addressable Url of the item
id	User defined unique name in a logical partition

```
{"_rid": "3NU+ANujVHcBAAAAAAA==",  
"_self": "dbs/3NU+AA==/colls/3NU+ANujVHc=/docs/3NU+ANujVHcBAAAAAAAA==/",  
"_etag": "\"0000000-0000-0000-b2c5-dcef4c3d01d5\"",  
"_attachments": "attachments/",  
"_ts": 1576359096}
```

Cosmos DB Internals <_ETAG>



ACCESS CONDITIONS

IF-MATCH (PUT AND DELETE)

IF-NONE-MATCH (GET)

IF-MODIFIED-SINCE (GET)

In an optimistic concurrency model, a violation is considered to have occurred if, after a user receives a value from the database, another user modifies the value before the first user has attempted to modify it.

COSMOS DB COMPLIANCE

Globally Applicable	US Government	Industry Specific
CSA STAR Certification	CJIS	HIPAA BAA
CSA STAR Attestation	DoD SRG Level 2	HITRUST
ISO 20000-1:2011	DoD SRG Level 4	PCI DSS
ISO 22301:2012	DoD SRG Level 5	
ISO 27001:2013	FedRAMP Moderate	
ISO 27017:2015	IRS 1075	
ISO 27018:2014	NIST CSF	
ISO 9001:2015	NIST SP 800-171	
SOC 1, 2, 3		

COSMOS DB EMULATOR



Develop & Test
Without Azure
Cloud



Supports Local
Network



Multi-Model

- SQL API
- TABLE API
- MONGO API
- CASSANDRA API
- GREMLIN API
- REST API



Docker version

<https://aka.ms/cosmosdb-emulator>

<https://hub.docker.com/r/microsoft/azure-cosmosdb-emulator/>

COSMOS DB EMULATOR

Emulator	
Account	Fixed Account & 1 Master Key
Connection Keys	No Key Re-generations
Service	Not Scalable, Limited number of Containers
Consistency Levels	One
Request Units	Might not be up-to-date
Data Explorer	SQL API

COSMOS DB EMULATOR

Parameter

/Port=<port> Emulator gateway service. Default is **8081**

/EnableCassandraEndpoint Enables Cassandra API

/CassandraPort=<port> Default is **10350**

/EnableGremlinEndpoint Enables Gremlin API

/GremlinPort=<port> Default is **8901**

/EnableMongoDbEndpoint Enables MongoDB API

/MongoPort=<port> Default is **10250**

/EnableMongoDbEndPoint=<ver> 3.2 or 3.6

/EnableTableEndpoint Enables Azure Table API

/TablePort=<port> Default is **8902**.

/DataPath=<path> Path to store data files.

Defaults to %LOCALAPPDATA%\CosmosDbEmulator

COSMOS DB EMULATOR

Parameter

/EnableRateLimiting	Enable request rate limiting simulation
/Consistency=<value>	Set the default consistency level
/ResetDataPath=<path>	Removes all data (recursive)
/GenKeyFile=<filename>	Generate a new authorization key
/Key=<key_string>	Authorization key for the emulator.
/KeyFile=<file_name>	Read authorization key from the specified file
/AllowNetworkAccess	Allow access to the Emulator over the network. Note you must also pass /Key=<key_string> or /KeyFile=<file_name> to enable network access



EMULATOR DEMO

SECURITY <USER TYPES>



ACCOUNT USERS

No Username
Connection Strings
No Active Directory
Account Level Access



DATABASE USERS

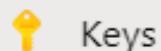
Has Users
Permissions
Active Directory
Database Level Access

SECURITY <ACCOUNT USERS>

securityiot - Keys

Azure Cosmos DB account

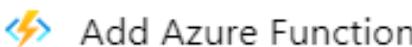
Search (Ctrl+ /)



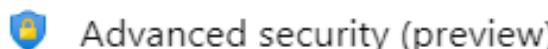
Keys



Add Azure Search



Add Azure Function



Advanced security (preview)



Preview Features



Locks

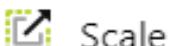


Export template

Containers



Browse



Scale



Settings

Read-write Keys

Read-only Keys

URI

<https://securityiot.documents.azure.com:443/>



PRIMARY KEY

0gRVrGqHwVyP1YMQ8Uue4orzdmrbL2wS8NMryxUZsS3YwA7V7SP1HuPZwOBmVrrdV4...



...

SECONDARY KEY

9TFpZv2wFzMfoDZQsVKx63suNR6kks0zGaee8L1UnIovUaYKMXfcIRdzQlCHA3EX9i7S7g...



...

PRIMARY CONNECTION STRING

AccountEndpoint=https://securityiot.documents.azure.com:443/;AccountKey=0gRVrGq...

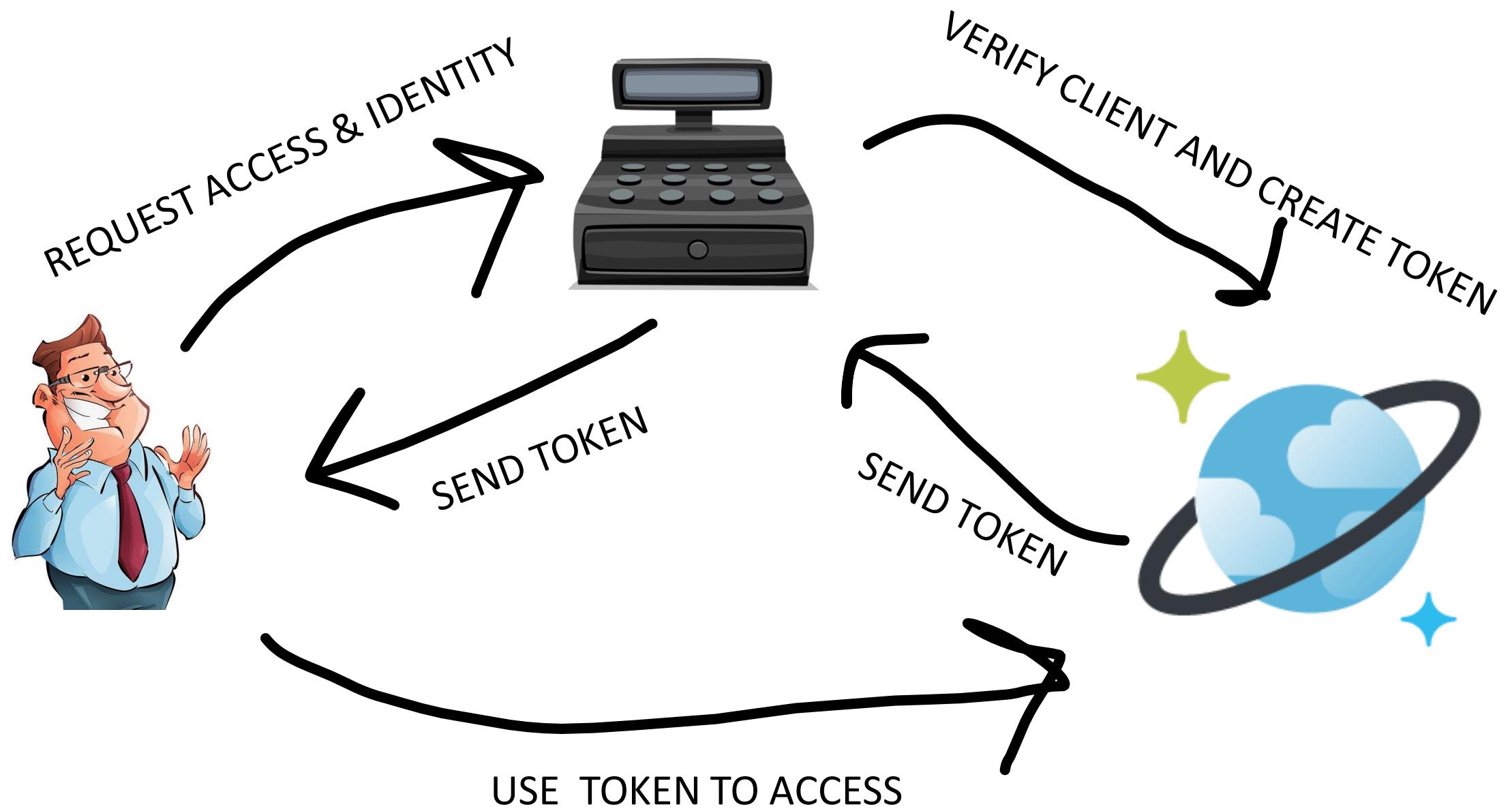


SECONDARY CONNECTION STRING

AccountEndpoint=https://securityiot.documents.azure.com:443/;AccountKey=9TFpZv2...



SECURITY <DATABASE USERS>



 Overview
 Activity log
 Access control (IAM)
 Tags
 Diagnose and solve problems
 Quick start
 Notifications
 Data Explorer
Settings
 Replicate data globally
 Default consistency
 Firewall and virtual networks
 Private Endpoint Connections
 CORS
 Keys
 Add Azure Search
 Add Azure Function
 Advanced security (preview)

Check access | Role assignment | Permissions | Classic administration | Roles

SECURITY <ROLES>

A role definition is a collection of permissions. You can use the built-in roles or you can create your own custom roles. [Learn more](#)

Name 	Type 					
Name	Type	Users	Groups	Service Principals		
 Owner 	BuiltInRole	0	0	0		
 Contributor 	BuiltInRole	0	0	0		
 Reader 	BuiltInRole	0	0	1		
 Azure Service Deploy Release 	CustomRole	0	0	0		
 Cosmos DB Account Reader 	BuiltInRole	0	0	0		
 Cosmos DB Operator 	BuiltInRole	0	0	0		
 CosmosBackupOperator 	BuiltInRole	0	0	0		
 DocumentDB Account Contrib. 	BuiltInRole	0	0	0		
 Log Analytics Contributor 	BuiltInRole	0	0	0		
 Log Analytics Reader 	BuiltInRole	0	0	0		
 Managed Application Contrib. 	BuiltInRole	0	0	0		
 Managed Application Operat. 	BuiltInRole	0	0	0		
 Managed Applications Reade 	BuiltInRole	0	0	0		

SECURITY <ROLE ASSIGNMENT>

 securityiot - Access control (IAM)
Azure Cosmos DB account

Search (Ctrl+ /) Add Edit columns Refresh Delete

Check access Role assignments Deny assignments

Check access
Review the level of access a user, group, service principal, or managed identity has to this resource. [Learn more](#)

Find Azure AD user, group, or service principal

Add role assignment

Role Select a role

Assign access to Azure AD user, group, or service principal

You are a guest user in this directory. You can search for users by their exact sign-in name, but you cannot search for groups or applications, and you cannot browse.

Search by name or email address

Selected members:
No members selected. Search for and add one or more members you want to assign to the role for this resource.

[Learn more about RBAC](#)

Save Discard

PARTITIONING

Does it work like Primary Key?

How about Table Partitioning?

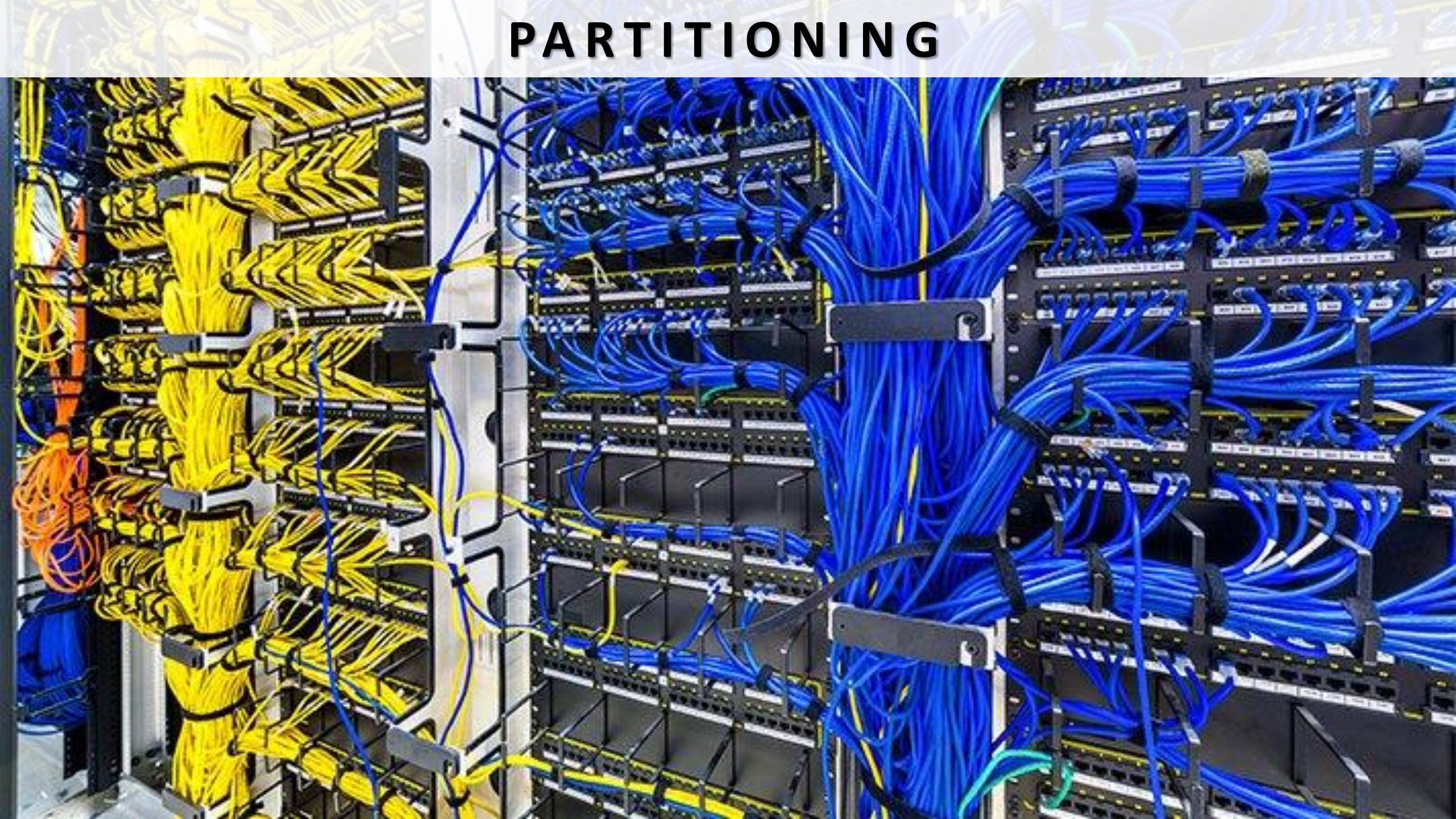
PARTITIONING



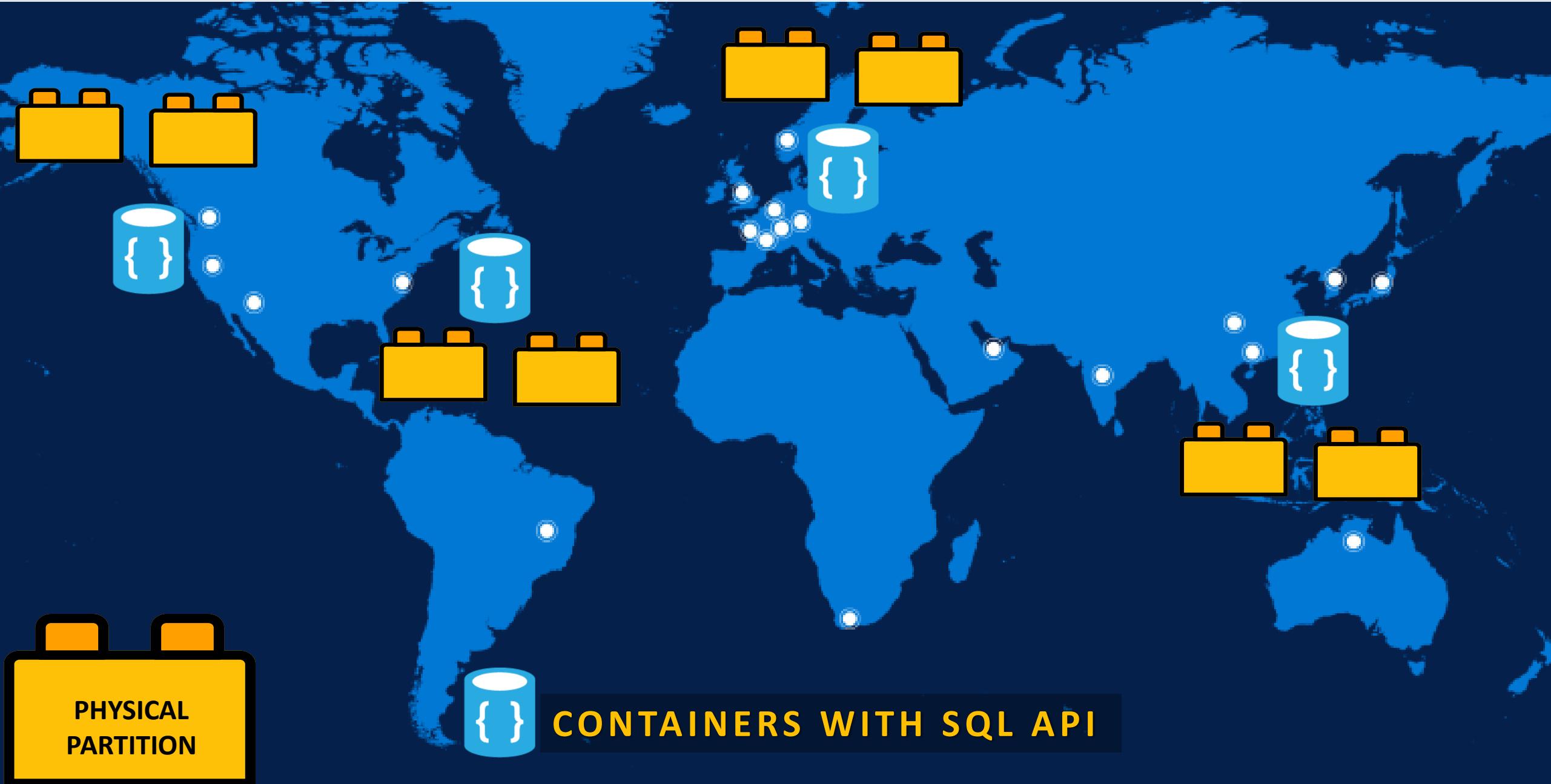
PARTITIONING



PARTITIONING



PARTITIONING <PHYSICAL>



PARTITIONING <PHYSICAL>



PARTITIONING <LOGICAL>



PARTITIONING <LOGICAL>





Users

UserId

LocationId

Orders

OrderId

ProductId

OrderDate

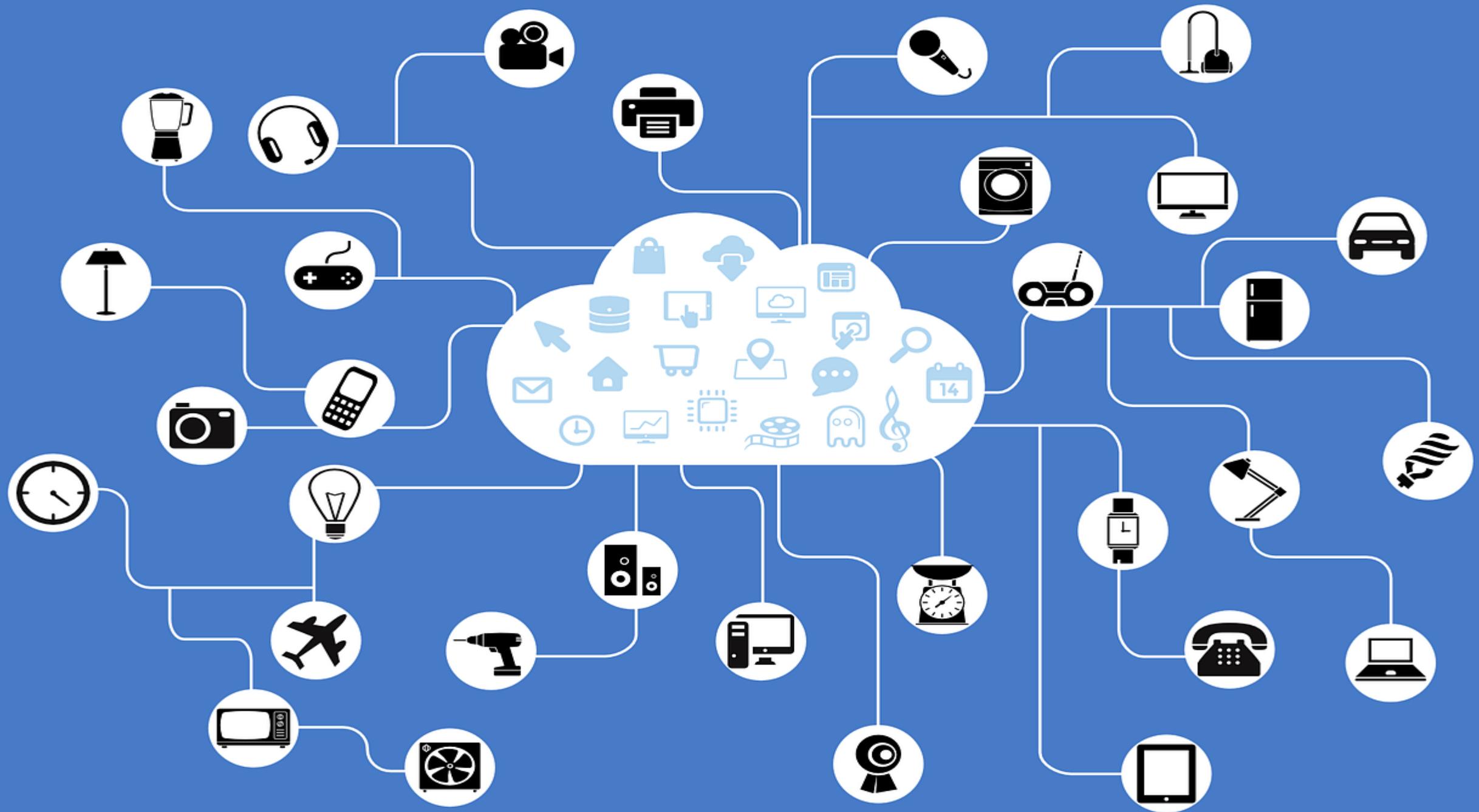
Products

ProductId

ProdCatId

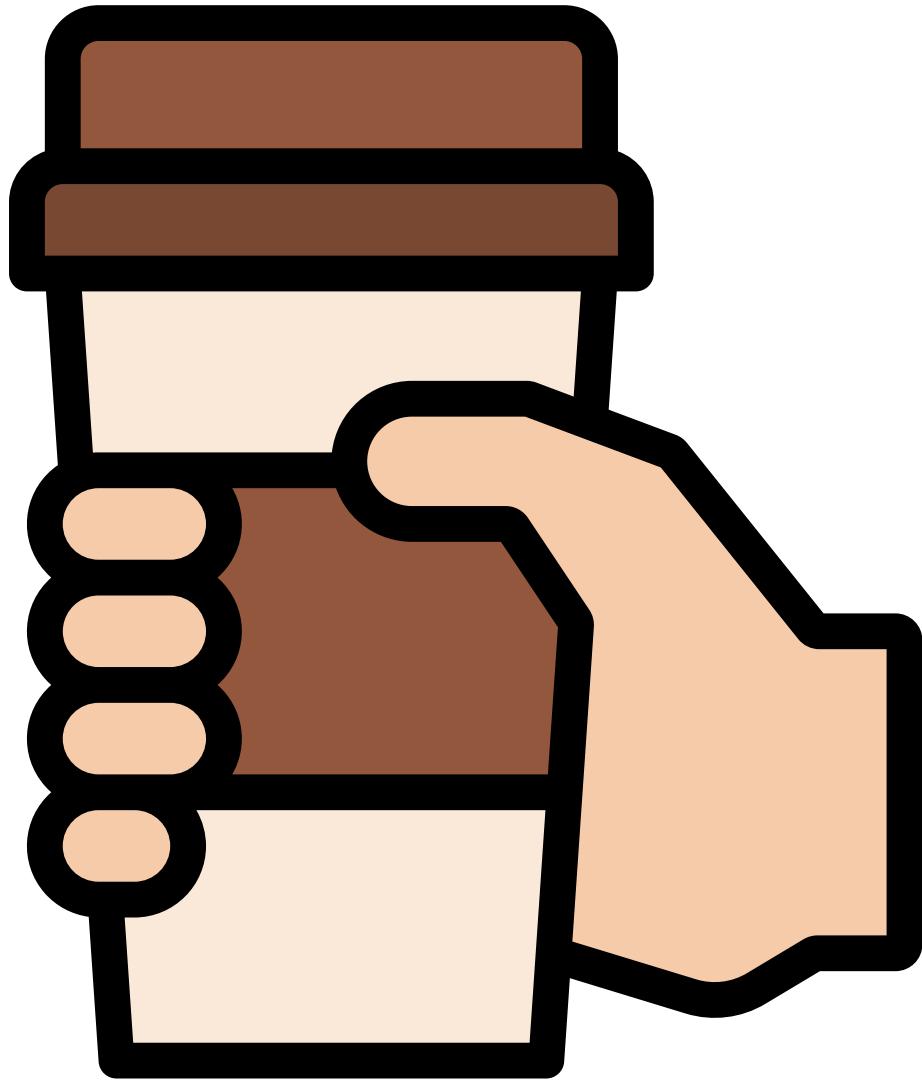
WarehouseId



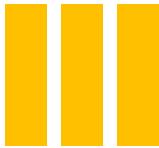


PARTITIONING <SYNTHETIC KEYS>





BREAK



MULTI-MODEL



WHY DO WE NEED MULTI-MODEL?

MULTI-MODEL



1

2

3

4

5

6

7

8

9

10

11

12

13

14

15

16

17

18

19

20

21

22

23

24

25

26

27

28

29

30

31

32

33

34

35

36

37

38

39

40

41

42

43

44

45

46

47

48

49

50

51

52

53

54

55

56

57

58

59

60

61



1

2

3

4

5

6

7

8

9

10

11

12

13

14

15

16

17

18

19

20

21

22

23

24

25

26

27

28

29

30

31

32

33

34

35

36

37

38

39

40

41

42

43

44

45

46

47

48

49

50

51

52

53

54

55

56

57

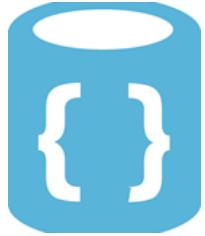
58

59

60



MULTI-MODEL



SQL API

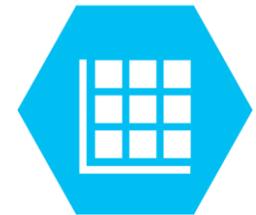


TABLE API



etcd

SQL API



Recommended API for new projects



Javascript Programming Model as the Foundation



Supports ACID Transactions by Sprocs, Triggers and SDK



Limited Aggregation

MONGO API

EASY MIGRATION

GLOBAL
DISTRIBUTION

COSMOS DB SLAs

MULTI MASTER



MongoDB wire protocol
Version 3.6



Azure Cosmos DB:
API for MongoDB

TTL

UNIQUE KEYS

PARTITION

TABLE API

AZURE TABLE

Fast, Lower Throughput limit

Single Region, No Failover

Only Primary Index, No secondary Indexes

2 Consistency Levels



COSMOS DB TABLE API

Faster, Higher Throughput limit

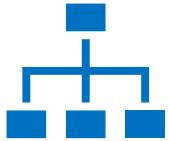
Full Global Distribution with failover & multi master

Indexing in all/any properties

5 Consistency Levels

CASSANDRA API

Cosmos DB is compatible with CQL v4



Cosmos DB removes the overhead of managing and monitoring settings



Global Distribution, Availability



Consistency Levels

GREMLIN API

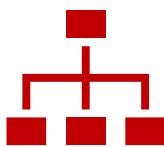
Gremlin API is based on the Apache TinkerPop graph database standard and uses the Gremlin query language.



Scalable throughput and storage



Multi Region replication



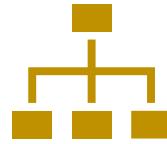
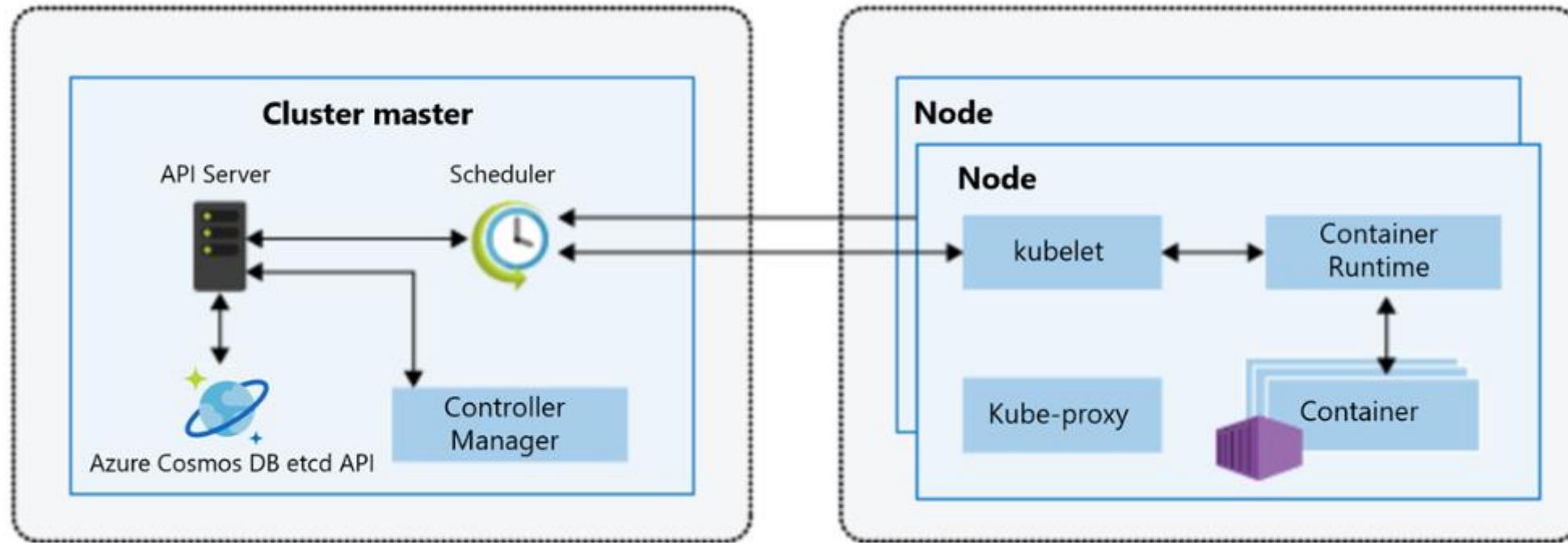
Fully managed Graph Database



Automatic Indexing

ETCD API

Azure Cosmos DB implements the wire-protocol of etcd version 3



No ETCD operations
management



Global Distribution



Elastic Scalability

MIGRATION TIPS

ESTIMATE THE
THROUGHPUT

ADJUST R/U
FOR
TRANSFER

CHOOSE
PARTITION
KEY

ADJUST
INDEXING
POLICY

TRANSFER
DATA TO
COSMOS DB



MULTI MODEL DEMO

AVAILABLE CONSISTENCY LEVELS

L
A
T
E
N
C
Y

RDBMS
High Latency
Lowest Throughput

STRONG



Specify LAG
-Operations
-Seconds

BOUNDED-STALENESS

DEFAULT



SESSION

Bounded Session without LAG
Accurate but not Current

CONSISTENT PREFIX

Out Of Order Read
Highest Scalability



EVENTUAL

AVAILABILITY AND THROUGHPUT



STRONG CONSISTENCY LEVEL



BOUNDED-STALENESS CONSISTENCY LEVEL

A close-up, low-angle shot of a person's hand and arm pushing a shopping cart. The cart has a red handle and a green frame. The background is blurred, showing shelves stocked with various grocery items like cereal boxes and bags of chips.

SESSION CONSISTENCY LEVEL

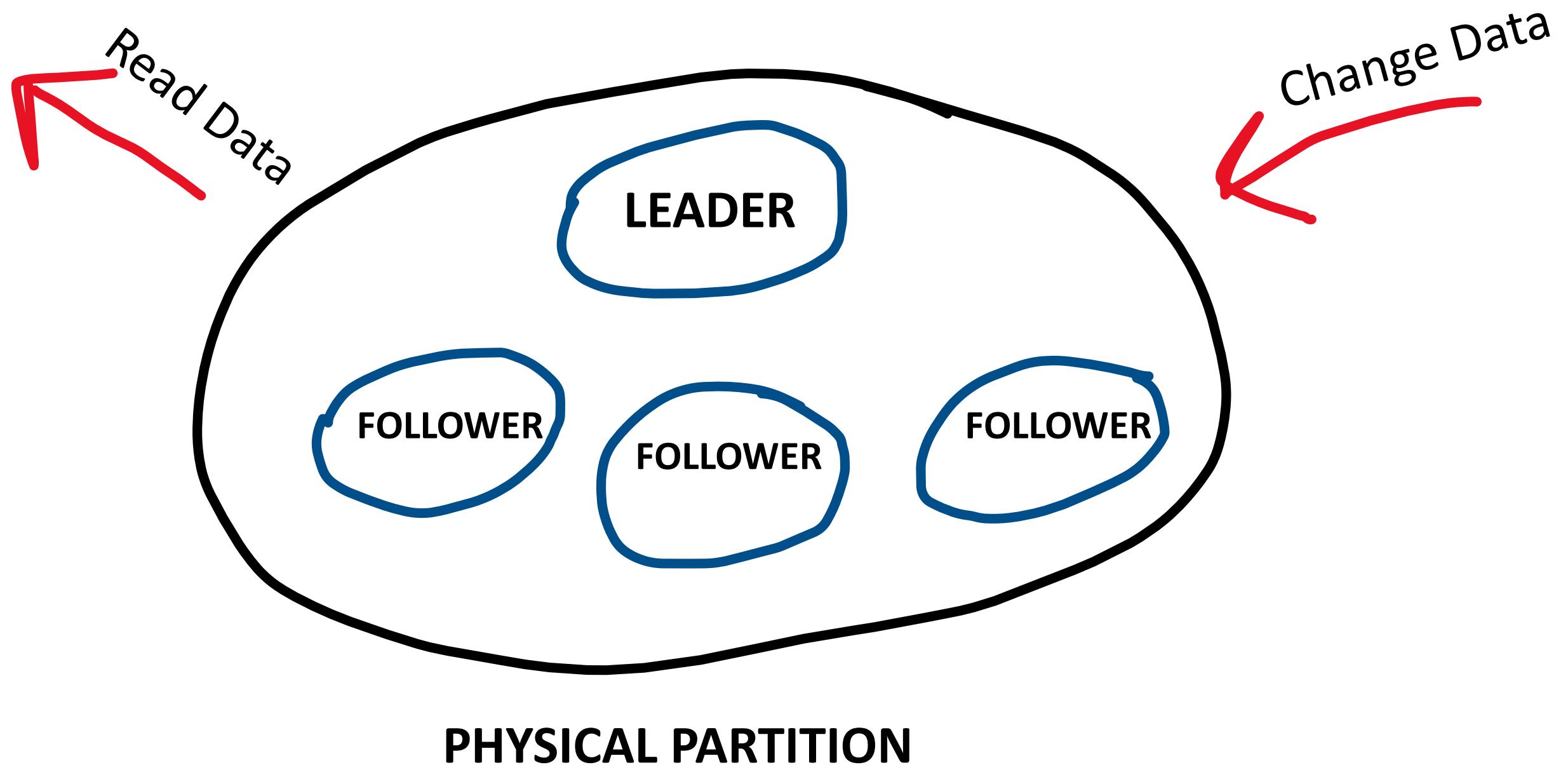


CONSISTENT PREFIX CONSISTENCY LEVEL



EVENTUAL CONSISTENCY LEVEL

DATABASE CONSISTENCY



INDEXING



Everything is
Indexed



Indexes has
their own files.



- Indexing Modes
- Consistent
 - Lazy
 - None



TTL requires
Indexing

INDEXING

RANGE

WHERE colname = 'value'

WHERE colname IN ('val1', 'val2')

WHERE ARRAY_CONTAINS(col.arr, 'val1')

WHERE colname > val

SPATIAL

ST_DISTANCE()

ST_WITHIN()

ST_INTERSECTS()

COMPOSITE

ORDER BY prop1, prop2

INDEXING (COMPOSITE INDEX)

```
SELECT * FROM Users ORDER BY State ASC, HiredOn ASC
```

```
// I add the following code to my Index Policy
```

```
"compositeIndexes": [
```

```
[
```

```
{
```

```
    "path": "/State",
```

```
    "order": "ascending"
```

```
},
```

```
{
```

```
    "path": "/HiredOn",
```

```
    "order": "ascending"
```

```
}
```

```
]
```

```
]
```

INDEXING (DEFINED PATHS)

Indexing Policy

```
1  {
2      "indexingMode": "consistent",
3      "automatic": true,
4      "includedPaths": [
5          {
6              "path": "/*"
7          }
8      ],
9      "excludedPaths": [
10         {
11             "path": "\\"_etag\\"/?"
12         }
13     ].
```

ANY ELEMENTS

SCALAR VALUE

Arrays are represented together with `/[]`

INDEXING (TIPS)

-  Lazy Indexing is cheaper and dangerous
-  None is better than Lazy
-  Queries may not return all matching results because of Index Transformation
-  Running index transformation can be cancelled by setting mode to NONE



INDEXING DEMO

UNIQUE KEYS

COSMOS DB UNIQUE KEYS VS SQL SERVER'S

- PRIMARY KEY
- UNIQUE INDEX
- UNIQUE CONSTRAINT

Be unique!

UNIQUE KEYS

* Database id ⓘ
 Create new Use existing

Provision database throughput ⓘ

* Container id ⓘ

Where did 'fixed' containers go? ⓘ

* Partition key ⓘ

My partition key is larger than 100 bytes

* Throughput (400 - 100,000 RU/s) ⓘ
 - +
Estimated spend (USD): **\$0.032 hourly / \$0.77 daily** (1 region, 400RU/s, \$0.00008/RU)

Unique keys ⓘ
✖

+ Add unique key

Items are unique within a logical partition.

Key Names are case-sensitive

Does not support Sparse

Must create with Containers.

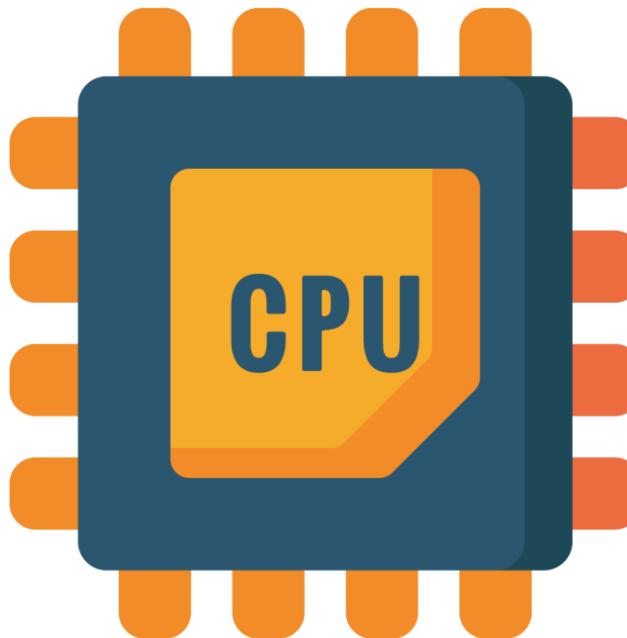
Higher Request Units (~20%)

Maximum 16 path values



UNIQUE KEYS DEMO

REQUEST UNITS



100 RU per sec. is ~\$5.84 per month

OPERATION (1KB)	ESTIMATE COST
READ	1 REQUEST UNIT
INSERT	3 REQUEST UNITS
UPSERT	3 REQUEST UNITS
DELETE	2 REQUEST UNITS
QUERY	4 REQUEST UNITS

REQUEST UNITS

RU COST DEPENDS ON

- DOCUMENT SIZE
- NUMBER OF PROPERTIES
- CONSISTENCY TYPE
- INDEXING POLICIES
- GLOBAL DISTRIBUTION
- ANY SCRIPTS & TRIGGERS
- UNIQUE KEYS



PARTITIONING <PHYSICAL>

Request Unit : 15,000



Request Unit : 10000



Request Unit : 5000

REQUEST UNITS <AUTOSCALE>



Infrequently used apps



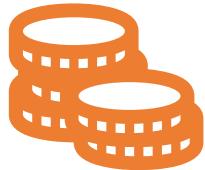
New applications



Unpredictable Workloads

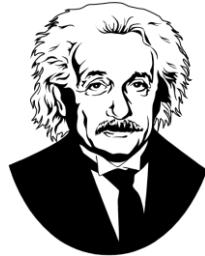


Dev and Tests



Costs %50 more

\$0.008/hour for Provisioned
\$0.012/hour for Autoscale



$0.01 * T_{max} \text{ GB}$

Max RU/s will be
automatically increased

Original 50,000 RU
 $0.01 * 50000 = 500 \text{ GB}$

Data increases to 600 GB
 $0.01 * 60000 = 600 \text{ GB}$
Max RU/s will increase to
60,000 RU

REQUEST UNITS <SERVERLESS>

! In Preview



Light Traffic



Moderate Scaling
Up to **5000 RU**



Moderate
Performance

DEVELOPMENT

TESTING

PROTOTYPING

POC

NON-CRITICAL APPS

SQL API ONLY
No Global Distribution



REQUEST UNITS DEMO

COSMOSDB QUERY PROCESSOR



PARSE THE QUERY

- Any syntax problems?
- All objects exist?

IS PARTITION KEY IN FILTER?

- YES: route to that partition
- NO: execute query in all partitions & merge the results in client side.

EXECUTE QUERY

- Execute query in series or parallel.
- Might make one or more round trips

SUMMARIZATION

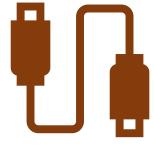
- SDK handles aggregation if it needs to.



BACKUP



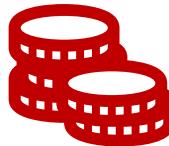
No Manual Backups



Automatic Backups every **4** hours
Only 2 latest backups are stored!



Retains snapshots for **30** days.



No extra charges



Contact Azure Support to restore data

You want to manage your own backups?
• Azure Data Factory
• Change Feed



Time To Live (TTL)

Time To Live (TTL)



Container or
Item level

Value is
configured in
seconds

Deletes happen
automatically

REST API
support

Emulator
support

```
1  {
2      "id": "10d3a105-5d8e-4752-8dd2-04503b21333f",
3      "DeviceId": "AG-10001",
4      "SensorId": 2,
5      "SensorCode": "BrokenGlass",
6      "FiredOn": "2019-08-29T11:19:51.726-04:00",
7      "_rid": "Ooc7ANjKz20CAAAAAAAA==",
8      "_self": "dbs/Ooc7AA==/colls/Ooc7ANjKz20=/docs/Ooc7ANjKz20C",
9      "_etag": "\"7200b707-0000-0200-0000-5d93ca7f0000\"",
10     "_attachments": "attachments/",
11     "_ts": 1569966719
12 }
```

Time To Live (TTL)

Search (Ctrl+ /) < | SQL API > Settings x

Advanced security (preview) | Save | Discard

Preview Features

Locks

Export template

Containers

Browse

Scale

Settings

Document Explorer

Query Explorer

Script Explorer

Monitoring

Alerts

Metrics

Diagnostic settings

Logs

Support + troubleshooting

SQL API

Settings

! Changing the TTL or Indexing Policy impacts query results while the operation completes. For more information see, [TTL and index interaction](#)

OFF IS DEFAULT
SET -1 FOR NO EXPIRATION

Time to Live

Off On (no default) **On**

1 second(s)

Partition key

/CustomerId

Indexing Policy

```
1 {  
2   "indexingMode": "consistent",  
3   "automatic": true,  
4   "includedPaths": [  
5     {  
6       "path": "/"  
7     }  
8   ],  
9   "excludedPaths": [  
10    {  
11      "path": "/$t  
12    }  
13  ]  
14}
```

Time To Live (TTL)

```
public class Order {  
    public string id { get; set; }  
    public string OrderId { get; set; }  
    public string CustomerId { get; set; }  
    public DateTime OrderDate { get; set; }  
    public DateTime ExpectedDeliveryDate { get; set; }  
    public Customer CustomerInfo { get; set; }  
    public SalesPerson SalesPersonInfo { get; set; }  
    [JsonProperty(PropertyName="ttl", NullValueHandling = NullValueHandling.Ignore)]  
    public int? TimeToLive { get;set; }  
}
```

```
//Container TTL will be applied  
currentOrder.TimeToLive = null;  
// use -1 to prevent this item from expiring  
currentOrder.TimeToLive = -1;
```

Time To Live (TTL)



Left-Over R/U is
used to delete



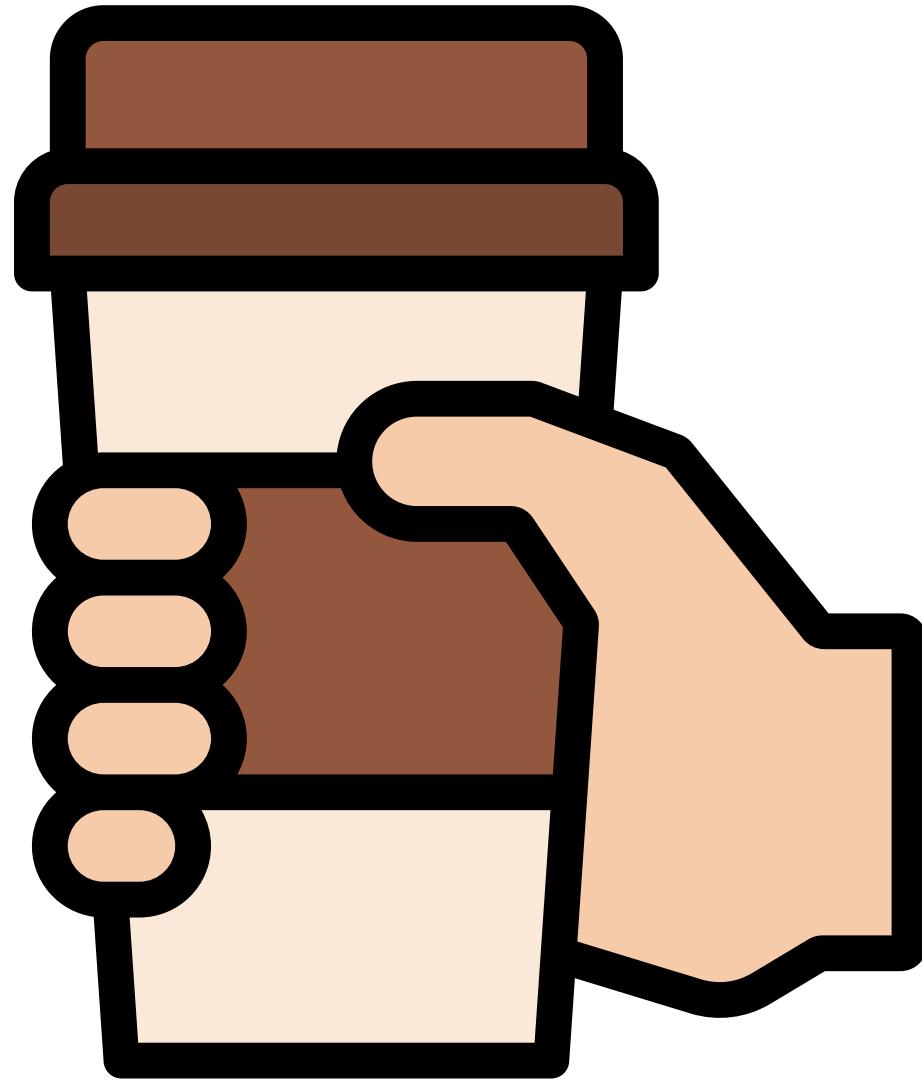
Data deletion
might be
delayed



Indexing is
Required

Maximum value

2147483647



BREAK

STORED PROCEDURES



Written in Javascript



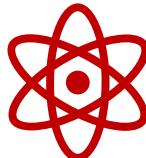
Registered & Executed against **A** collection



Supports Create, Read, Update, Delete, Query operation



You can perform complex operations



Atomic Transactions



Best suited for write-heavy applications

STORED PROCEDURES <READ>

SQL API

DATA

- RetailDemo
- Stackoverflow
- Posts

Items

Scale & Settings

Stored Procedures

```
function readpost(postid) {  
    var collection = getContext().getCollection();  
    collection.chain().filter(function(doc){  
        return doc.PostId = postid;  
    }).map(function(doc){  
        return {  
            id: doc.PostId,  
            body: doc.PostBody,  
            Owner: doc.OwnerUserId  
        };  
    }).value();  
}
```

STORED PROCEDURES <INSERT>

SQL API

DATA

► RetailDemo

▼ Stackoverflow

► Posts

Items

Scale & Settings

► Stored Procedures



```
function Createpost(itemToCreate) {  
    var context = getContext();  
    var container = context.getCollection();  
    var accepted =  
        container.createDocument(container.getSelfLink()  
,  
        itemToCreate, function (err, itemCreated) {  
            if (err) throw new Error('Error' +  
                err.message);  
            context.getResponse().setBody(itemCreated.id)  
        });  
    if (!accepted) return;  
}
```

STORED PROCEDURES <INPUT PARAMETERS>



Input Parameters are always sent as a string

```
function sample(arr) {  
    if (typeof arr === "string") arr = JSON.parse(arr);  
  
    arr.forEach(function(a) {  
        // do something here  
        console.log(a);  
    });  
}
```

STORED PROCEDURES <.NET SDK 3>

CREATE NEW SPROC

```
.StoredProcedureResponse storedProcedureResponse = await  
client.GetContainer("database", "container")  
    .Scripts.CreateStoredProcedureAsync(new StoredProcedureProperties {  
        Id = "SprocName",  
        Body = File.ReadAllText(@"..\js\sproc.js")  
    });
```

EXECUTE SPROC

```
var result = await client.GetContainer("database",  
"container").Scripts.ExecuteStoredProcedureAsync<string> ("SprocName", new  
PartitionKey("PartitionId"), newItem);
```



SPROCS DEMO

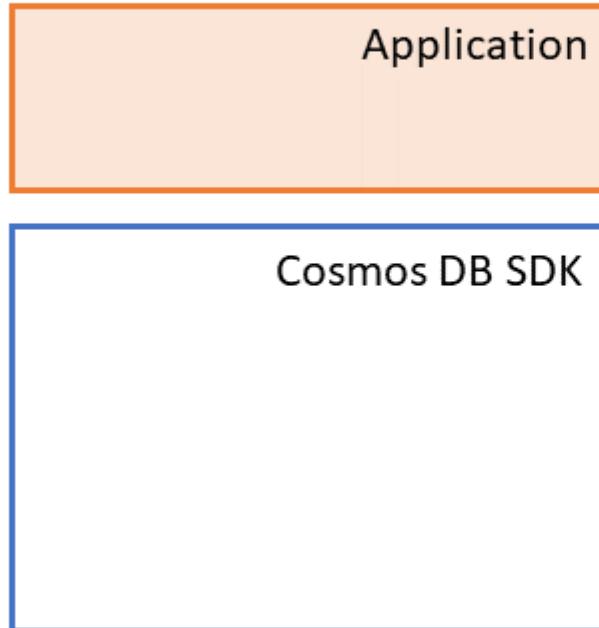
BULK DATA

BULK EXECUTOR LIBRARY

.NET or JAVA
SQL API & GRAPH API
Supports Import & Update

.NET SDK 3.4.0 or later

.NET
SQL API
Supports Import & Update



BULK DATA

```
static async Task<bool> BulkUpdate(List<StackOverflowPost> posts)
{
    try
    {
        var options = new CosmosClientOptions() { AllowBulkExecution = true };
        var cosmosClient = new CosmosClient(connectionString, options);
        Container container = cosmosClient.GetContainer("Stackoverflow", "Posts");
        posts.ForEach(t => t.Tags = "Update tags");
        List<Task> concurrentTasks = new List<Task>();
        double total = 0;
        foreach (var itemToUpdate in posts)
        {
            var tsk = container.UpsertItemAsync(itemToUpdate, new PartitionKey(itemToUpdate.PostId));
            concurrentTasks.Add(tsk);
        }
        Console.WriteLine("Started : " + DateTime.Now.ToString("yyyy-MM-dd HH:mm:ss"));
        await Task.WhenAll(concurrentTasks);
        Console.WriteLine("Ended : " + DateTime.Now.ToString("yyyy-MM-dd HH:mm:ss"));
        return true;
    }
    catch (Exception ex)
    {
        Console.WriteLine(ex.Message.ToString());
        return false;
    }
}
```

BULK DATA <Optimizing Bandwidth>

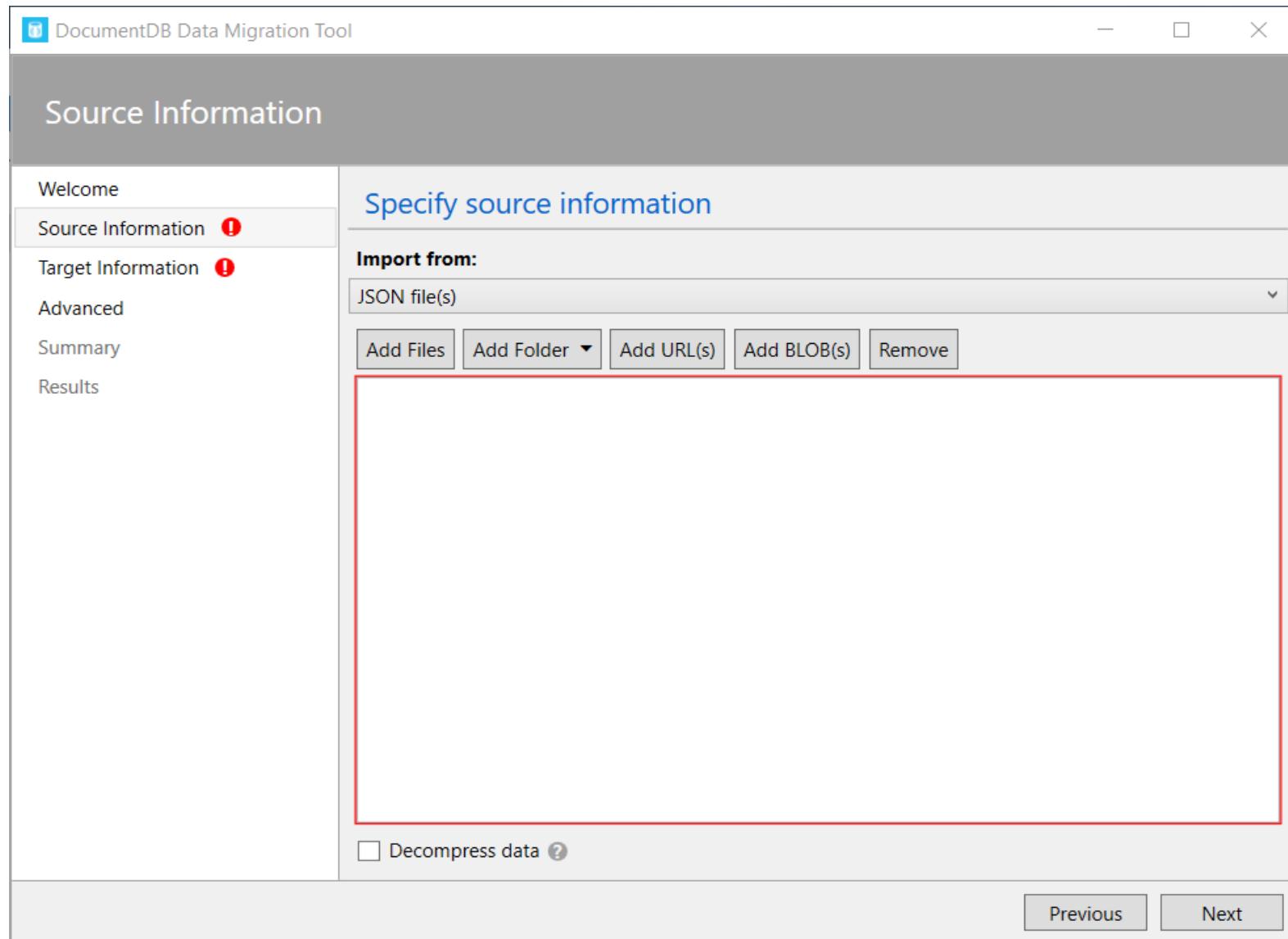
Response contains the original request in **WRITE** operations. Do you need this info?

```
1 ToDoActivity item = new ToDoActivity() { ...};  
2 ItemResponse<ToDoActivity> itemResponse = await this.container.CreateItemAsync(  
3     item,  
4     requestOptions: new ItemRequestOptions() { EnableContentResponseOnWrite = false });  
5 ToDoActivity responseContent = itemResponse.Resource; // This is really the same as "item"  
6 double consumedRUs = itemResponse.Headers.RequestCharge;  
7 string etag = itemResponse.Headers.ETag;  
8 string session = itemResponse.Headers.Session;
```

enablecontentresponseonwrite-sample-write-disable.cs hosted with ❤ by GitHub

[view raw](#)

DATA MIGRATION TOOL



AVAILABLE SOURCE OPTIONS

- JSON files
- MongoDB
- SQL Server
- CSV files
- Azure Table storage
- Amazon DynamoDB
- HBase
- Azure Cosmos DB collections

IMPORTING DATA FROM SQL SERVER

DocumentDB Data Migration Tool

Source Information

Welcome

Source Information

Target Information !

Advanced

Summary

Results

Specify source information

Import from:

SQL

Connection String

Server=DESKTOP-BL9A454;Database=WideWorldImporters;Trusted_Connection=True;

Enter Query Select Query File

select top 2 sales.orders.OrderId as id, sales.orders.OrderId,sales.orders.OrderDate, sales.orders.ExpectedDeliveryDate, Sales.Customers.CustomerID, Sales.Customers.CustomerName as [Customer.CustomerName], Sales.Customers.CreditLimit as [Customer.CreditLimit], Sales.Customers.AccountOpenedDate as [Customer.AccountOpenedDate], Sales.Customers.IsOnCreditHold as [Customer.IsOnCreditHold], Sales.Customers.DeliveryAddressLine1 as [Customer.Address1], Sales.Customers.DeliveryAddressLine2 as [Customer.Address2], Sales.Customers.DeliveryPostalCode as [Customer.Zipcode], Application.People.PersonID as [SalesPerson.PersonId], Application.People.FullName as [SalesPerson.FullName], Application.People.PhoneNumber as [SalesPerson.PhoneNumber], Application.People.IsExternalLogonProvider as [SalesPerson.IsExternalProvider], Application.People.EmailAddress as [SalesPerson.Email] from sales.orders join Sales.Customers on Sales.Orders.CustomerID = Sales.Customers.CustomerID join Application.People on sales.Orders.SalespersonPersonID = Application.People.PersonID

Nesting Separator ?

Previous Next

TRANSFORMING DATA

Data Migration Tool
Stored Procedures
Using Javascript

Specify target information

Export to:

- DocumentDB - Bulk import (single partition collections)
- DocumentDB - Sequential record import (partitioned collection)
- DocumentDB - Bulk import (single partition collection)
- JSON file

Advanced Options

Bulk Insert Stored Procedure [?](#)

C:\Users\hasan\OneDrive\Desktop\Cosmos\Import 1.7\BulkTransformationInsert.js

Batch Size

50

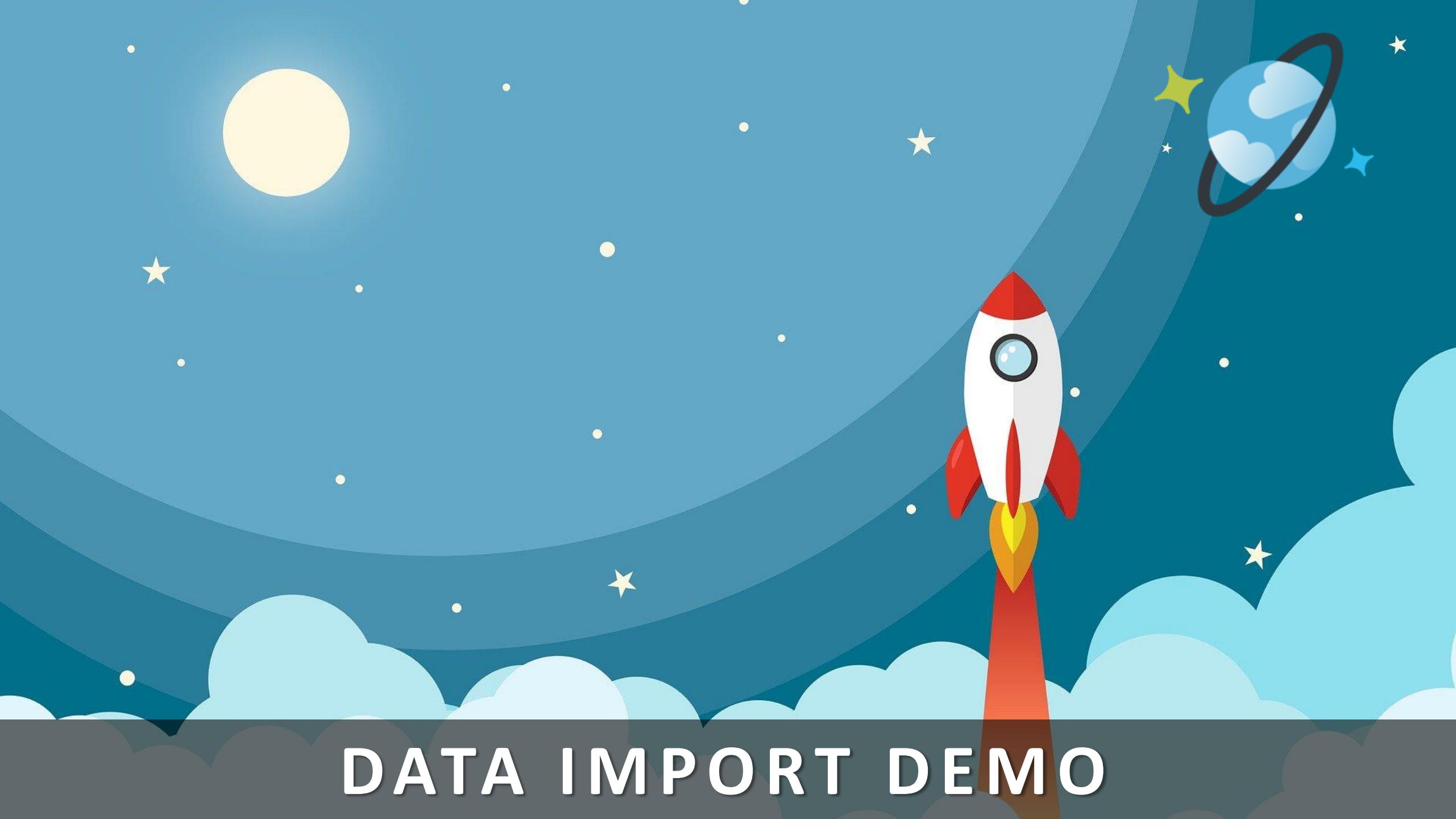
Max Script Size (bytes)

524278

Disable Automatic Id Generation [?](#)

Update Existing Documents [?](#)

Persist Date and Time as



DATA IMPORT DEMO



SQL API

AGGREGATES	MATHEMATICAL		TYPE CHECKINGS	STRING	
COUNT()	ABS()	CEILING()	IS_ARRAY	CONCAT	CONTAINS
MIN()	EXP()	FLOOR()	IS_BOOL	ENDSWITH	INDEX_OF
MAX()	LOG()	LOG10()	IS_NULL	LEFT	LENGTH
SUM()	POWER()	ROUND()	IS_NUMBER	LOWER	LTRIM
AVG()	SIGN()	SQRT()	IS_OBJECT	REPLACE	REPLICATE
ARRAY	SQUARE()	TRUNC()	IS_STRING	REVERSE	RIGHT
ARRAY_CONCAT	ACOS()	ASIN()	IS_DEFINED	RTRIM	STARTSWITH
ARRAY_CONTAINS	ATAN()	ATN2()	IS_PRIMITIVE	SUBSTRING	UPPER
ARRAY_LENGTH	COS()	COT()	GEOSPATIAL		
ARRAY_SLICE	DEGREES()	PI()	ST_WITHIN	ST_DISTANCE	
	RADIANS()	SIN()	ST_INTERSECTS	ST_ISVALID	
	TAN()		ST_ISVALIDDETAILED		

SQL API <SELECT>

```
SELECT prop1, prop2, prop...
FROM container (alias)
WHERE filter
```

```
SELECT container.prop1
FROM container
WHERE table1.tid = 5
```

```
SELECT t.prop1
FROM container t
WHERE t.tid = 5
```

```
SELECT t["prop1"]
FROM container t
WHERE t.tid = 5
```

```
SELECT t["prop1"] as Name
FROM container t
WHERE t.tid = 5
```

SQL API <SELECT>

```
SELECT DISTINCT prop1  
FROM container t  
WHERE t.tid = 5
```

```
SELECT VALUE(t.prop1)  
FROM container t  
WHERE t.tid = 5
```

```
SELECT DISTINCT VALUE(t.prop1)  
FROM table t  
WHERE t.tid = 5
```

```
SELECT {Id: t.tid, Version: 5, Name :'N/A'} as CustomObj,  
[1,2,3,4,5] as Numbers  
FROM container t  
WHERE t.tid = 5
```

SQL API <JOIN>



You can not join to other collections or documents

```
SELECT *
FROM container t
JOIN alias in t.array
WHERE t.tid = 5
```

SQL API <ARRAYS>

```
SELECT [t.prop1, t.prop2] as CustomArray  
FROM container t  
WHERE t.tid = 5
```

```
SELECT t.prop1, ARRAY(SubQuery) as CustomArray  
FROM container t  
WHERE t.tid = 5
```

```
SELECT *  
FROM sub IN container t.children  
WHERE t.tid = 5
```

SQL API <TYPE CHECKING FUNCTIONS>

SELECT IS_ARRAY(<expr>)

SELECT IS_DEFINED(<expr>)

SELECT IS_BOOL(<expr>)

SELECT IS_NULL(<expr>)

SELECT IS_NUMBER(<expr>)

SELECT IS_OBJECT(<expr>)

SELECT IS_STRING(<expr>)

SELECT IS_PRIMITIVE(<expr>)

SQL API <Keywords>

```
SELECT *
FROM container t
WHERE t.col1 BETWEEN x AND y
```



```
SELECT *
FROM container t
WHERE t.col1 IN ('val1','val2',...)
```

```
SELECT TOP 1 *
FROM container t
```

SQL API <Operators>

```
SELECT (t.prop < 10) ? "val for true" : "val for false" AS alias  
FROM container t
```

```
SELECT t.prop ?? "value of col1 is not present "  
FROM container t
```

ALL OPERATOR

SELECT * FROM Table t	VALID
SELECT VALUE * FROM Table t	NOT VALID
SELECT *, t.Id FROM Table t	NOT VALID

SQL API <Aggregates>

```
SELECT COUNT(1)  
FROM container t
```

```
SELECT VALUE COUNT(1)  
FROM container t
```



SUM()



MIN()



MAX()



AVG()

SQL API <Group BY>

```
SELECT *
FROM container t
GROUP BY t.prop
```

```
SELECT *
FROM container t
GROUP BY UPPER(t.prop)
```



Cannot work with **ORDER BY**
.NET SDK **3.3** or above

SQL API <ORDER BY>

```
SELECT *  
FROM container t  
ORDER BY t.prop
```

```
SELECT *  
FROM container t  
ORDER BY t.prop1 ASC, t.prop2 DESC
```



Property must be indexed to use in Order BY
Composite Index might be required

SQL API <OFFSET LIMIT>

```
SELECT t.prop1,t.prop2  
FROM container t  
ORDER BY t.prop1  
OFFSET 1 LIMIT 1
```

...

SKIP THE FIRST
VALUE RETURN
THE SECOND



SELECT
FROM
WHERE
ORDER BY
GROUP BY
OFFSET LIMIT

SQL API <FUNCTIONS>

```
SELECT ARRAY_CONCAT([1,2],[3]) AS Custom  
[{"Custom": [1,2,3]}]
```

```
SELECT ARRAY_CONTAINS([1,2],[3]) AS Custom  
[{"Custom": false}]
```

```
SELECT ARRAY_LENGTH([1,2]) AS Custom  
[{"Custom": 2}]
```

```
SELECT ARRAY_SLICE(["Bob","George","Tom"], 1) AS Custom  
[{"Custom": ("George","Tom")}]
```

SQL API <FUNCTIONS>

```
SELECT GetCurrentDateTime() as now  
[{"now": "2020-01-30T19:40:26.681Z"}]
```

```
SELECT GetCurrentTimeStamp() as now  
[{"now": 1580413374958}]
```



GetCurrentDateTime()

UTC date and time ISO 8601 format

GetCurrentTimeStamp()

of ms that have elapsed since 1 Jan 1970

SQL API <FUNCTIONS>

```
SELECT StringToArray("[1,2,3]") as a1  
[{"a1": [1,2,3]}]
```

```
SELECT StringToBoolean(" true") as a1  
[{"a1": true}]
```



Case Sensitive

```
SELECT StringToNumber("3.38") as a1  
[{"a1": 3.38}]
```

```
SELECT StringToObject('{"Age": 40}') as a1  
[{"a1": { "Age": 40 } }] 
```

SQL API <CASE SENSITIVE SEARCH>

CONTAINS(Source, searchfor, ignore case)

STARTSWITH(Source, searchfor, ignore case)

```
SELECT *
FROM p
WHERE CONTAINS(p.Title, 'Sql' , true)
```

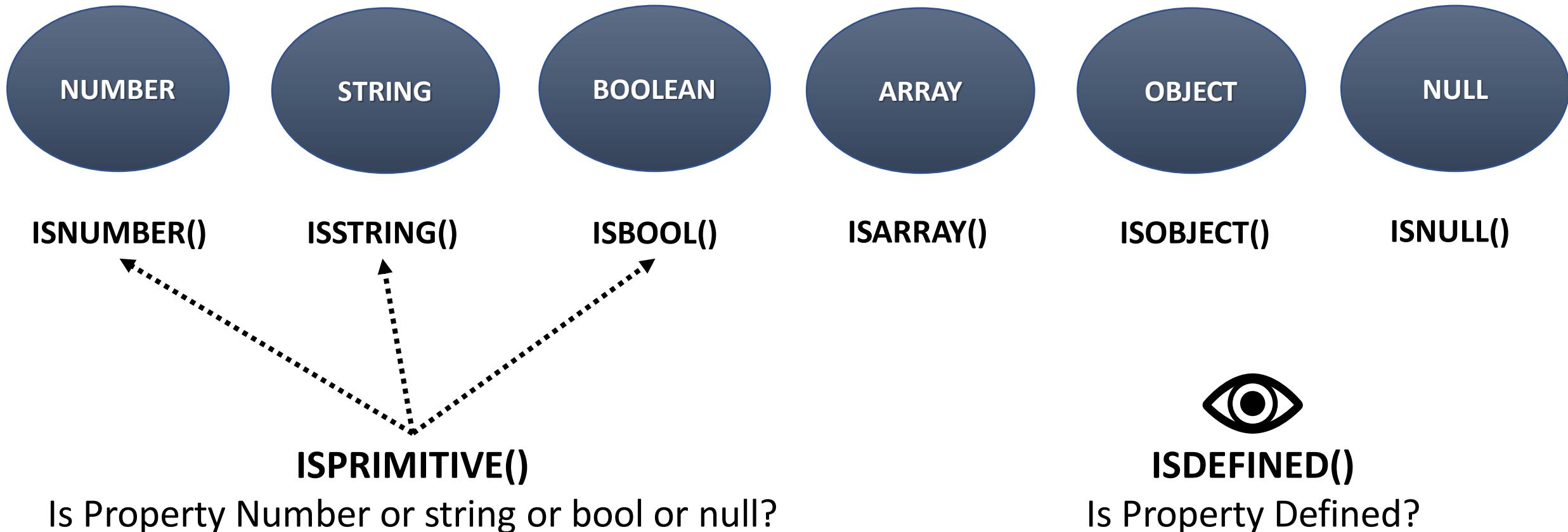
SQL API <FUNCTIONS>

OTHER STRING FUNCTIONS	
CONCAT	CONTAINS
ENDSWITH	INDEX_OF
LEFT	LENGTH
LOWER	LTRIM
REPLACE	REPLICATE
REVERSE	RIGHT
RTRIM	STARTSWITH
SUBSTRING	ToString
TRIM	UPPER



SQL API DEMO

HANDLING JSON



HANDLING JSON

```
{  
    "PostId": 9685,  
    "PostBody": "<p>You could possibly use a LINQ  
    "Title": null,  
    "ViewCount": 0,  
    "AnswerCount": 0,  
    "CommentCount": 2,  
    "FavoriteCount": 0,  
    "AcceptedAnswerId": 0,  
    "CreatedOn": "2008-08-13T12:03:05.9400000",  
    "ClosedDate": null,  
    "OwnerUserId": 1,  
    "OwnerDisplayName": "Jeff Atwood",  
    "PostType": "Answer",  
    "Score": 364,  
    "IsClosed": false,  
    "Tags": null,  
    "TagArray": [  
        "c#",  
        "web"  
    ],  
    "Details": {  
        "DetailId": 1,  
        "Value": "Terminated Account"  
    },  
    "id": "bf676822-178a-410d-a5e8-890886e26a16",  
}
```

```
1  SELECT IS_ARRAY(c.Tags) as IsTags_Array,  
2  IS_ARRAY(c.TagArray) as IsTagArray_Array  
3  FROM c  
  
1  SELECT IS_NUMBER(c.Score) as IsScoreNumber,  
2  IS_NUMBER(c.Title) as IsTitleNumber  
3  FROM c  
  
1  SELECT IS_PRIMITIVE(c.Title) as IsTitleExists,  
2  IS_PRIMITIVE(c.TagArray) as IsTagArrayPrimitive  
3  FROM c  
4  WHERE c.PostId = 9685
```

Results Query Stats

1 - 1

```
[  
  {  
    "IsTitleExists": true,  
    "IsTagArrayPrimitive": false  
}]
```

HANDLING JSON <DateTime>

.NET JavaScriptSerializer.

"\"\\Date(1573844334736)\""

.NETDataContractJsonSerializer.

"\"\\Date(1573844334736-0500)\""

Built-in JSON object of JavaScript.

"2019-11-15T13:58:00.511Z"

ISO 8601 Format

"2019-11-15T13:58:00-05:00"

Unix Time stamp
of Seconds elapsed since Jan 1, 1970

```
[{"id": "598000000000000000",  
 "_rid": "iPV7AMG-W1kRJwA",  
 "_self": "dbs/iPV7AA==/c/  
 _etag": "\"3b00c9b1-000  
 _attachments": "attachm  
 "_ts": 1578718347}
```



HANDLING JSON <DateTime>

```
{  
    "Id": 4,  
    "AcceptedAnswerId": 7,  
    "AnswerCount": 13,  
    "Body": "<p>I want to use a track-bar to change a form's  
    "ClosedDate": null,  
    "CommentCount": 1,  
    "CommunityOwnedDate": "2012-10-31T16:42:47.2130000",  
    "CreationDate": "2008-07-31T21:42:52.6670000",  
    "FavoriteCount": 41,  
    "LastActivityDate": "2018-07-02T17:55:27.2470000",  
    "LastEditDate": "2018-07-02T17:55:27.2470000",  
    "LastEditorDisplayName": "Rich B",  
    "LastEditorUserId": 6786713,  
    "OwnerId": 8,  
    "ParentId": 0,  
    "PostTypeId": 1,  
    "Score": 573,  
    "Tags": "<c#><floating-point><type-conversion><double><d>  
    "Title": "Convert Decimal to Double?",  
    "ViewCount": 37080,  
    "CreationUnixDt": 1217540572,  
    "id": "17a0a499-c18e-4c1d-a56e-aad819782914",  
    "_rid": "UR1lAPoNMCIBAAAAAAA==",  
    "_self": "dbs/UR1lAA==/colls/UR1lAPoNMCI=/docs/UR1lAPoNM  
    "_etag": "\\"00000000-0000-0000-9cde-3b3b5e5c01d5\\\"",  
    "_attachments": "attachments/",  
    "_ts": 1573950636
```

```
{  
    "Id": 4,  
    "AcceptedAnswerId": 7,  
    "AnswerCount": 13,  
    "Body": "<p>I want to use a track-bar to change a form's  
    "ClosedDate": null,  
    "CommentCount": 1,  
    "CommunityOwnedDate": "2012-10-31T16:42:47.2130000",  
    "CreationDate": "2008-07-31T21:42:52.6670000",  
    "Creationdt": {  
        "Year": 2008,  
        "Month": 7,  
        "Day": 31,  
        "Hour": 21,  
        "Minute": 42,  
        "Second": 52,  
        "TimeZone": -5  
    },  
    "FavoriteCount": 41,  
    "LastActivityDate": "2018-07-02T17:55:27.2470000",  
    "LastEditDate": "2018-07-02T17:55:27.2470000",  
},
```

HANDLING JSON <DateTime>

```
1  SELECT * FROM c  
2  WHERE c.OwnerUserId = 9 and c.CreationDate > '2008-07-1T00:00:00' |
```

Results Query Stats

METRIC	VALUE
Request Charge	7.54 RUs
Showing Results	1 - 19
Retrieved document count ⓘ	19
Retrieved document size ⓘ	20084 bytes
Output document count ⓘ	19
Output document size ⓘ	20152 bytes
Index hit document count ⓘ	19
Index lookup time ⓘ	0.7000000000000001 ms
Document load time ⓘ	0.22 ms
Query engine execution time ⓘ	0.0600000000000005 ms

HANDLING JSON <DateTime>

```
1  SELECT * FROM c  
2  WHERE c.OwnerUserId = 9 and c.CreationUnixDt > 1214870400
```

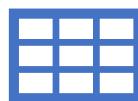
Results	Query Stats
METRIC	VALUE
Request Charge	7.64 RUs
Showing Results	1 - 19
Retrieved document count ⓘ	19
Retrieved document size ⓘ	20084 bytes
Output document count ⓘ	19
Output document size ⓘ	20152 bytes
Index hit document count ⓘ	19
Index lookup time ⓘ	0.45 ms
Document load time ⓘ	0.19 ms
Query engine execution time ⓘ	0.1 ms

AZURE COSMOSDB REST API



DATABASES

`https://{{dbaccount}}.documents.azure.com/dbs/{{db}}`



COLLECTIONS

`https://{{dbaccount}}.documents.azure.com/dbs/{{db}}/colls/{{coll}}`



DOCUMENTS

`https://{{dbaccount}}.documents.azure.com/dbs/{{db}}/colls/{{coll}}/docs/{{doc}}`



USERS

`https://{{dbaccount}}.documents.azure.com/dbs/{{db}}/users/{{user}}`

REST API (HEADERS)

REQUEST

Authorization
Content-Type
x-ms-date
x-ms-version
x-ms-consistency-level
x-ms-session-token
x-ms-continuation
x-ms-max-item-count
x-ms-documentdb-query-enablecrosspartition

RESPONSE

Content-Type
Date
etag
x-ms-continuation
x-ms-item-count
x-ms-request-charge
x-ms-resource-usage
x-ms-session-token
x-ms-retry-after-ms

REST API (CONDITIONAL OPERATIONS)

Request HEADER	TYPE	VALUE
If-Match	PUT, DELETE	Etag
If-None-Match	GET	Etag
If-Modified-Since	GET	Date

```
52     "Location": "Basement"
53   }
54 ],
55   "id": "8a631e15-646d-42db-8834-8ae4f950946b",
56   "_rid": "ebEYAP40XpsBAAAAAAA==",
57   "_self": "dbs/ebEYAA==/colls/ebEYAP40Xps=/docs/ebEYAP40XpsBAAAAAAA==/",
58   "_etag": "\\"3700274f-0000-0100-0000-5d389aeb0000\\\"",
59   "_attachments": "attachments/",
60   "_ts": 1563990763
61 }
```

AZURE COSMOSDB REST API

AUTHORIZATION

- **HTTP Verb**
- **Resource Type**
- **Resource Link**
- **Date**
- **\n**

HASHED TOKEN

- BASE64 encoded
- MIME RFC2045
- CosmosDB Key
- Token Type
- Token Version

HttpVerb + “\n” + ResourceType + “\n” + ResourceLink + “\n” + Date

type={typeoftoken}&ver={tokenversion}&sig={hashedsignature}

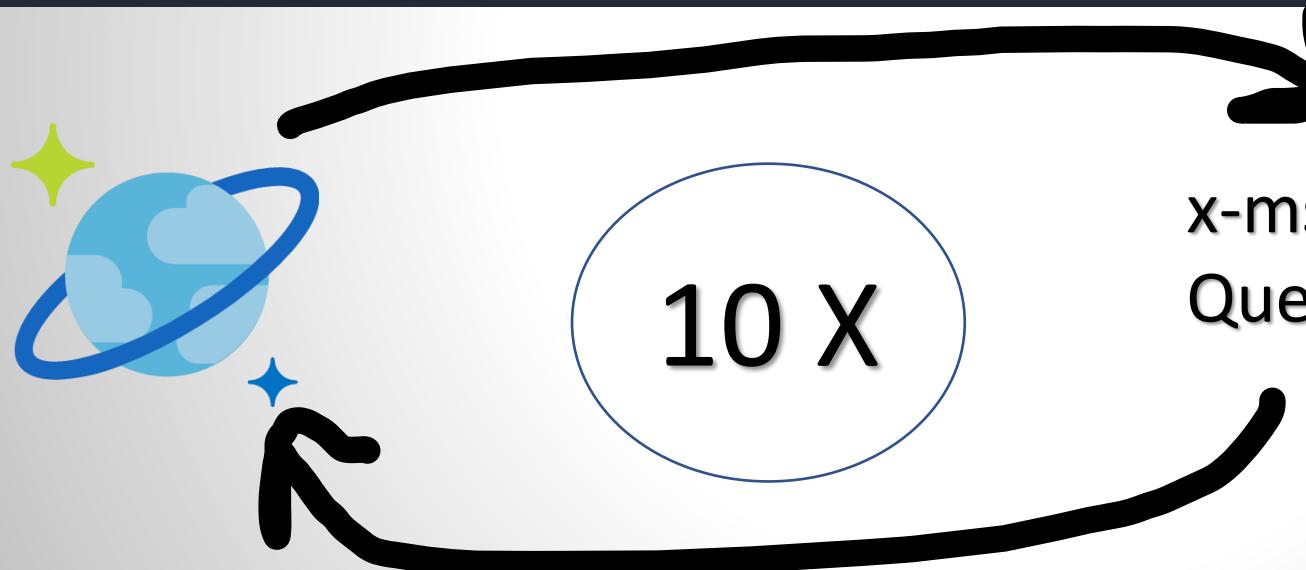
CONTINUATION TOKENS

x-ms-max-item-count in request header

x-ms-continuation in response & request headers

RequestTimeOut specifies timeout of continuation tokens

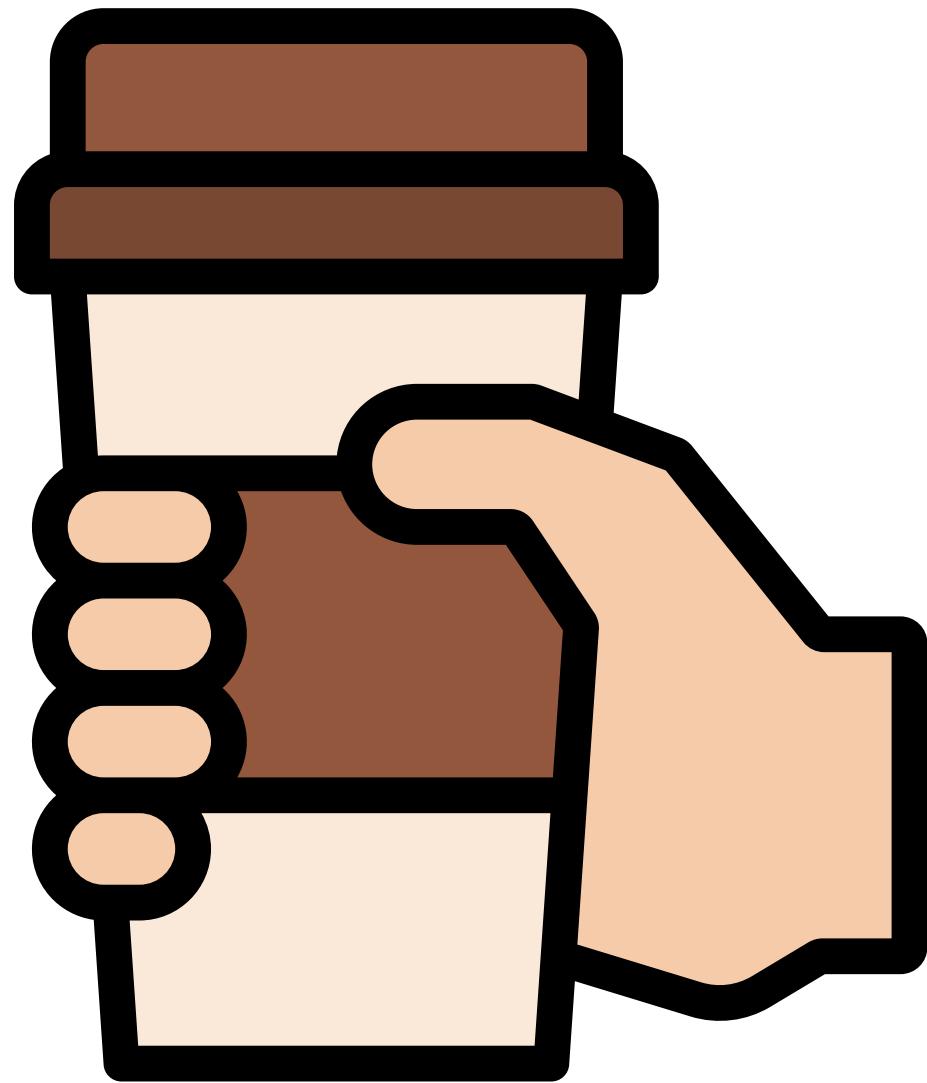
SDK handles continuation tokens automatically.



x-ms-max-item-count = 100
Query returns 1000 documents.



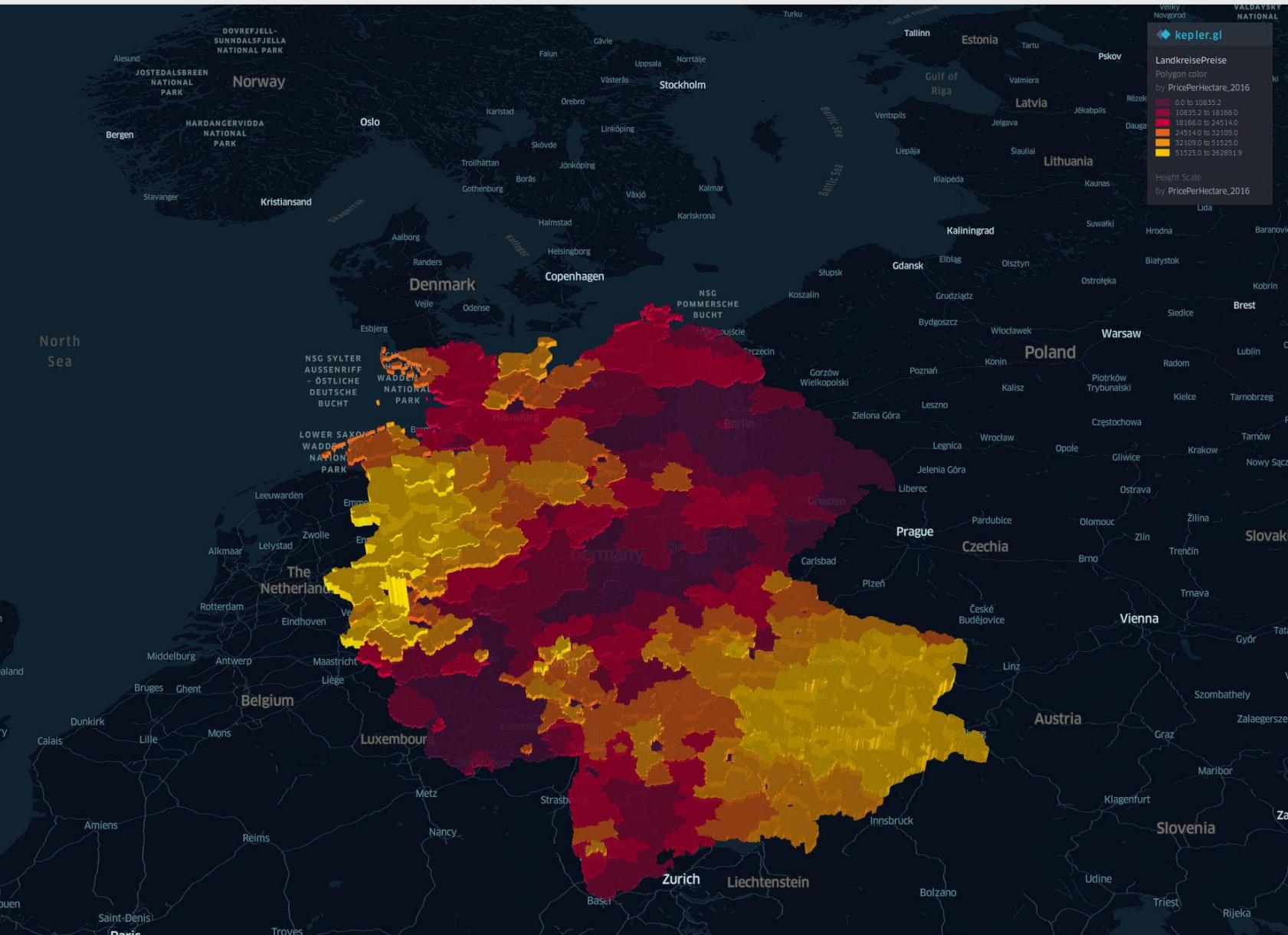
REST API DEMO



BREAK

SPATIAL DATA

- Every object can be represented by Spatial Data.
- Represents the physical location of an object.
- Represents the shape of an object.
- Helps you to make geometric range searches.
- Helps you to see high level data by using maps.
- SSRS, Power BI, Excel support

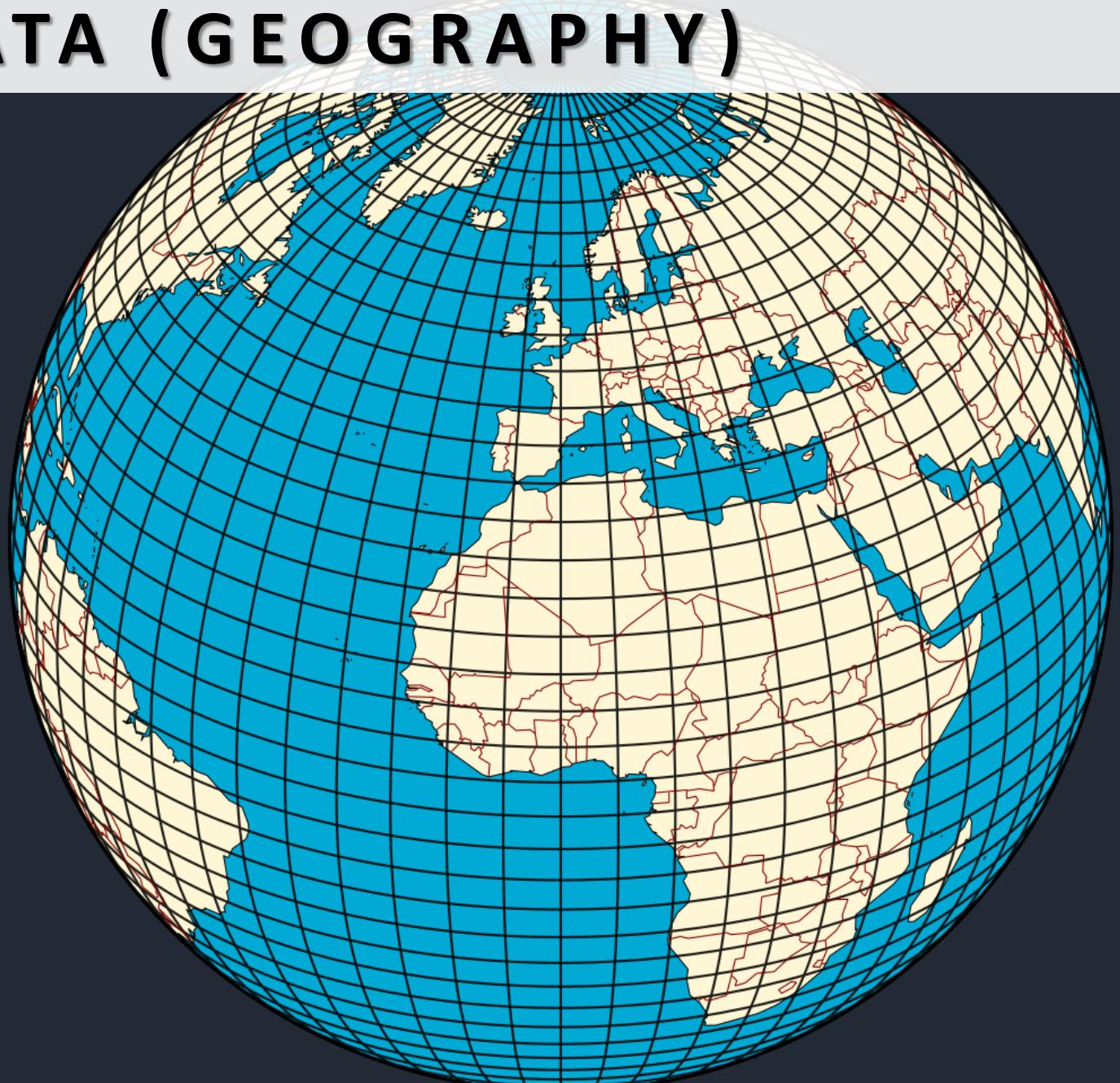


SPATIAL DATA (GEOGRAPHY)

Data Types

- Point
- LineString
- Polygon
- MultiPolygon

SQL API ONLY



SPATIAL DATA

COORDINATE REFERENCE SYSTEM

World Geodetic System-84

GPS Devices

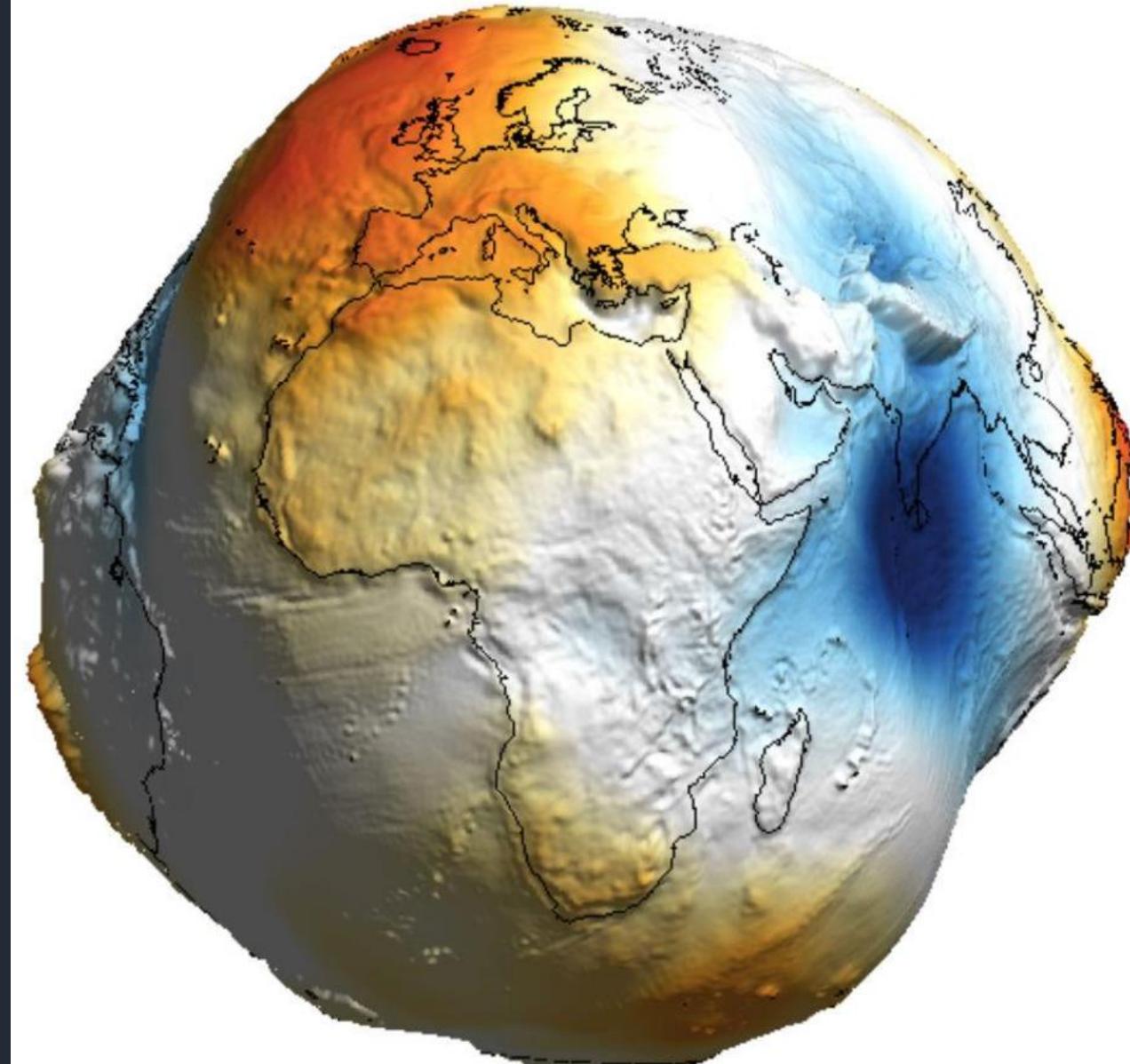
Bing Maps

Google Maps

Longitude : -180 degrees to 180 degrees

Latitude : -90 degrees to 90 degrees

Longitude, Latitude



SPATIAL DATA

ST_DISTANCE(*spatial expression, spatial expression*)

Returns the distance between two points, polygons or linestring

ST_WITHIN(*spatial obj 1, spatial obj 2*)

Checks if Spatial Obj 1 is within the second spatial object

ST_INTERSECTS(*spatial obj, spatial obj*)

Checks if given spatial objects intersects

ST_ISVALID(*spatial obj*) and **ST_ISVALIDDETAILED**(*spatial obj*)

Checks if given spatial object is valid

SPATIAL DATA



GeoJSON is a format for encoding a variety of geographic data structures using JSON

SUPPORTS

POINT LINestring POLYGON MULTIPOINT MULTILINESTRING MULTIPOLYGON COLLECTION

```
{  
  "id": "cosmosdb-profile",  
  "screen_name": "@CosmosDB",  
  "city": "Redmond",  
  "topics": [ "global", "distributed" ],  
  "location": {  
    "type": "Point",  
    "coordinates": [ 31.9, -4.8 ]  
  }  
}
```

```
{  
  "type": "Polygon",  
  "coordinates": [ [  
    [ 31.8, -5 ],  
    [ 31.8, -4.7 ],  
    [ 32, -4.7 ],  
    [ 32, -5 ],  
    [ 31.8, -5 ]  
  ] ]  
}
```

SPATIAL DATA

Inconsistency in Mapping Software and Formats

Formats

Longitude, Latitude

Latitude, Longitude

GeoJSON

geoRSS

KML

Encoded Polyline

ShapeFile

WKT

WKB

geobuf

APIs

Longitude, Latitude

Latitude, Longitude

OpenLayers

LeafLet

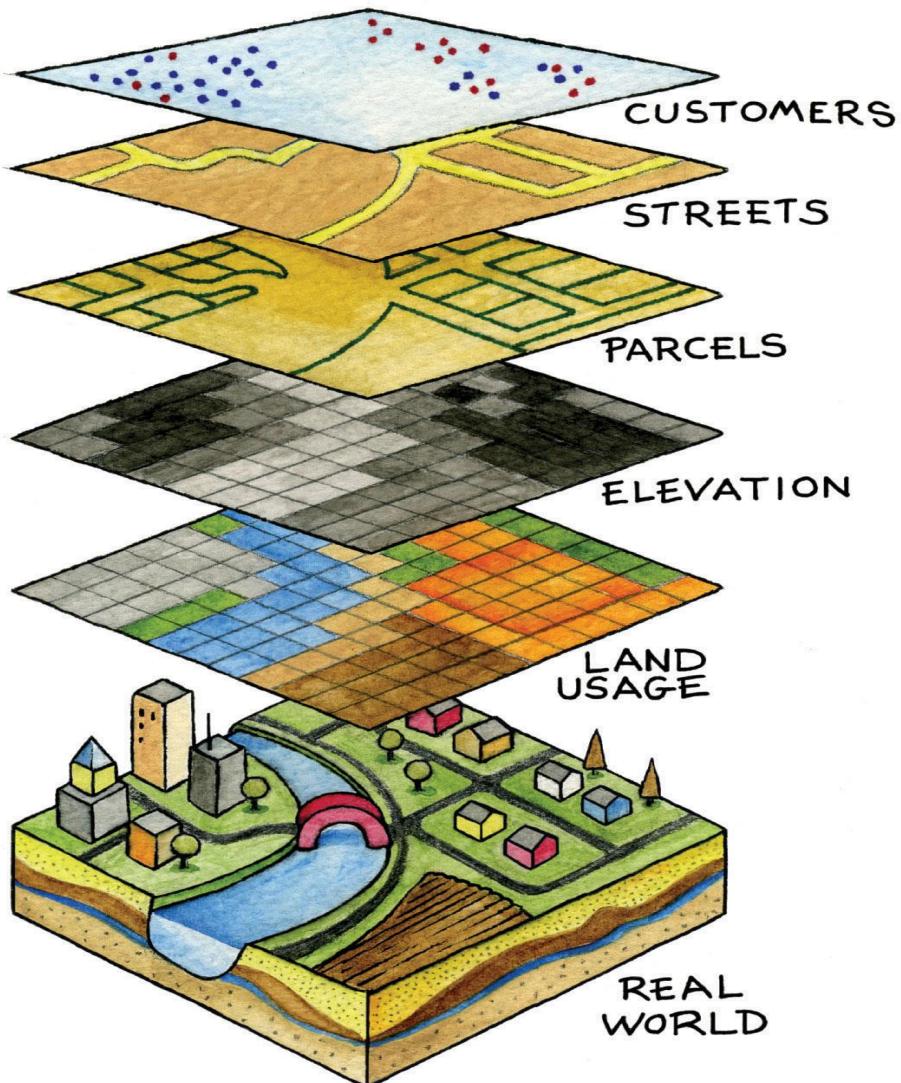
D3

Google Maps

ArcGIS

MapBox

SPATIAL DATA



Different problems need different data types

Calculating Distance

Find Data by drawing polygons



SPATIAL DATA DEMO

DATA MODELLING



RELATIONAL DB



COSMOS DB

DATA MODELLING

WHEN TO EMBED

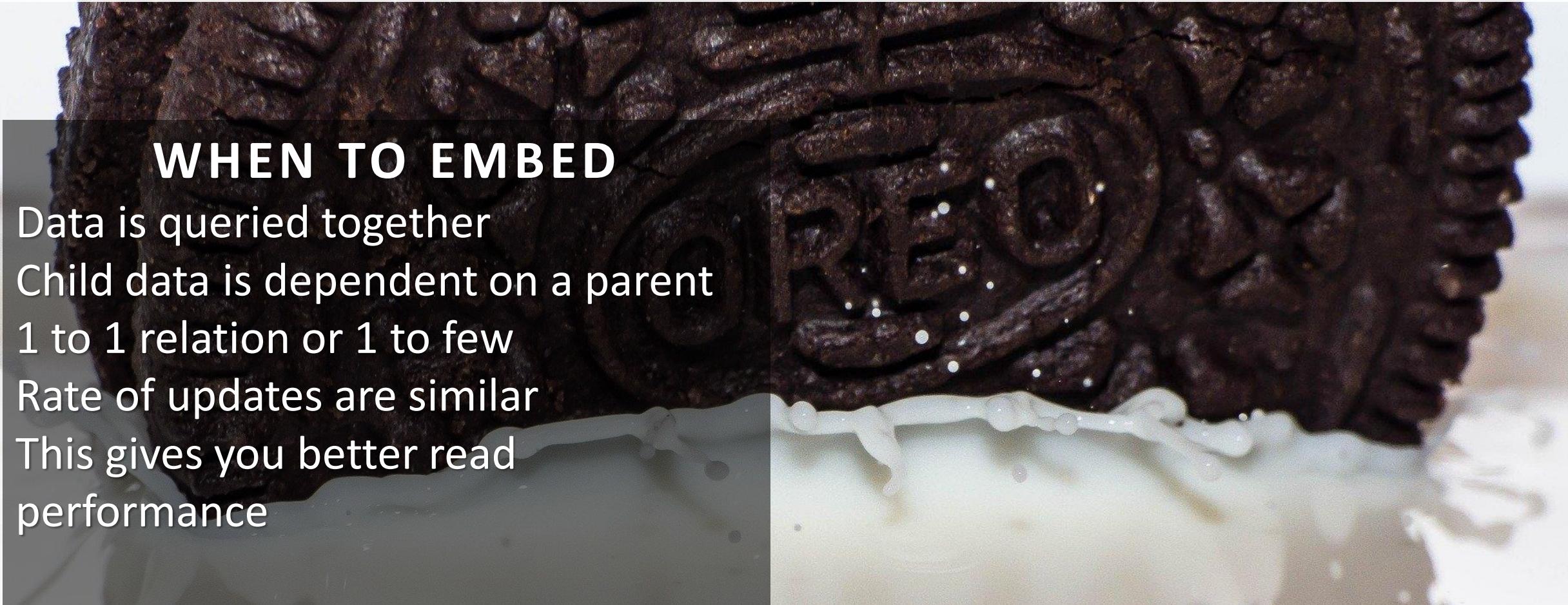
Data is queried together

Child data is dependent on a parent

1 to 1 relation or 1 to few

Rate of updates are similar

This gives you better read performance



DATA MODELLING

WHEN TO REFERENCE

1 to many

Data changes at different rates

Item is referenced by many others

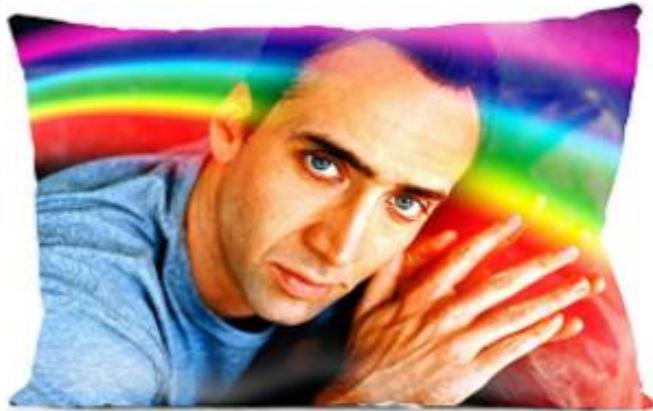
Better write performance

Might require more network calls
for read



[Back to results](#)

COSMOS DB DATA MODELLING



Roll over image to zoom in



DoubleUSA Nicolas Cage Pillowcases
Both Sides Print Zipper Pillow Covers
20"x30"

by DoubleUSA

21 ratings

Amazon's Choice for "nicholas cage pillow case"

Price: **\$9.99**

FREE One-Day Pickup. Details & FREE Returns

Pay **\$9.99** \$0.00 after using available Amazon Rewards Visa Card Points.

- HIGH DEFINITION – The Most Advanced Printing Technology Ensures The Best Definition Image Of The Pillowcase.
- ZIPPER – Hidden Zipper Closure At The Side To Keep Pillow From Sliding Out, And For Style, Comfort And Better Fit.
- 100% Polyester – Hypoallergenic, Long Staple Yarns Woven in a Sateen Weave Make These Pillowcases Incredibly Soft and Luxurious.
- 100% SATISFACTION GUARANTEE – 100% MONEY BACK GUARANTEE If Not Completely Satisfied, Return With No-Questions Asked.
- DoubleUSA is the brand of our products. The product has only one pillowcase.

[Compare with similar items](#)

\$9.99

prime

FREE One-Day Pickup. Details & FREE Returns

FREE delivery: **Friday** Details

Deliver to Hasan - Uniontown 44685

In Stock.

Qty: 1



Add to Cart



Buy Now

Sold by DoubleUSA and Fulfilled by Amazon.

Add gift options

[Add to List](#)

[Add to Wedding Registry](#)

DATA MODELLING

Relational Database

Products

ProductId

Product Details

DetailId

ProductId

ProdCatId

Product Media

MediaId

ProductId

Reviews

ReviewId

ProductId

NoSQL

Product

Details

Media

Reviews

Categories

ProductCategories

Product

Type

JUPYTER NOTEBOOKS

New Container ▾

New Notebook ▾

Open Terminal

Reset Workspace



SQL API



▼ DATA

► RetailDemo

► Stackoverflow

▼ NOTEBOOKS

▼ Sample Notebooks (View Only)

CassandraIntroduction.ipynb

GettingStarted.ipynb

GlobalDistribution.ipynb

GremlinIntroduction.ipynb

Indexing.ipynb

RequestUnits.ipynb

Visualization.ipynb

Common Tasks



Start with Sample

Get started with a sample provided by
Cosmos DB



New Container

Create a new container for storage and
throughput



New Notebook

Create a notebook to start querying,
visualizing, and modeling your data

Recents

Tips

JUPYTER NOTEBOOKS

Built-in Commands



%%SQL

Run a SQL
Query



%%UPLOAD

Upload
Data



%database

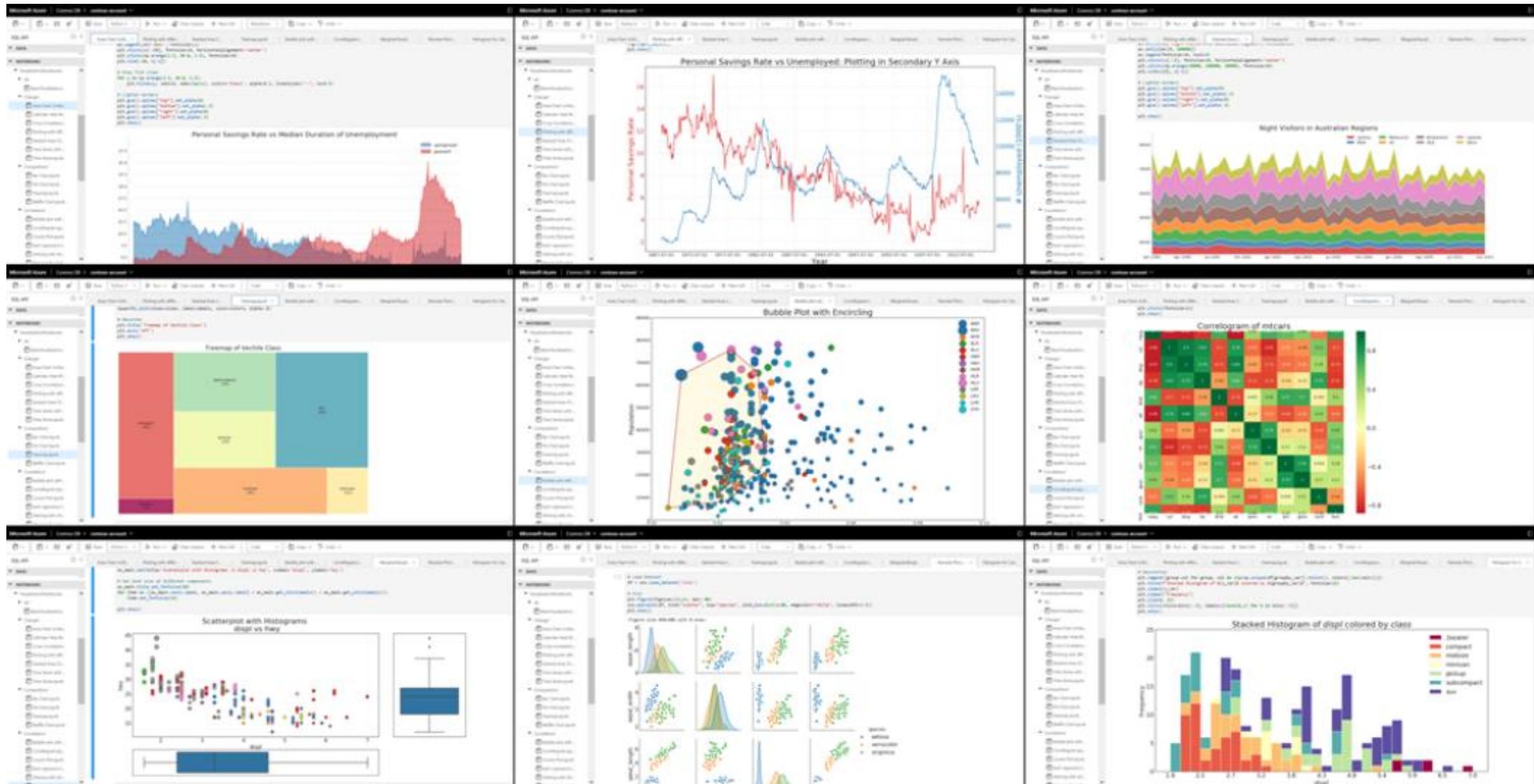
Defaults
Database



%container

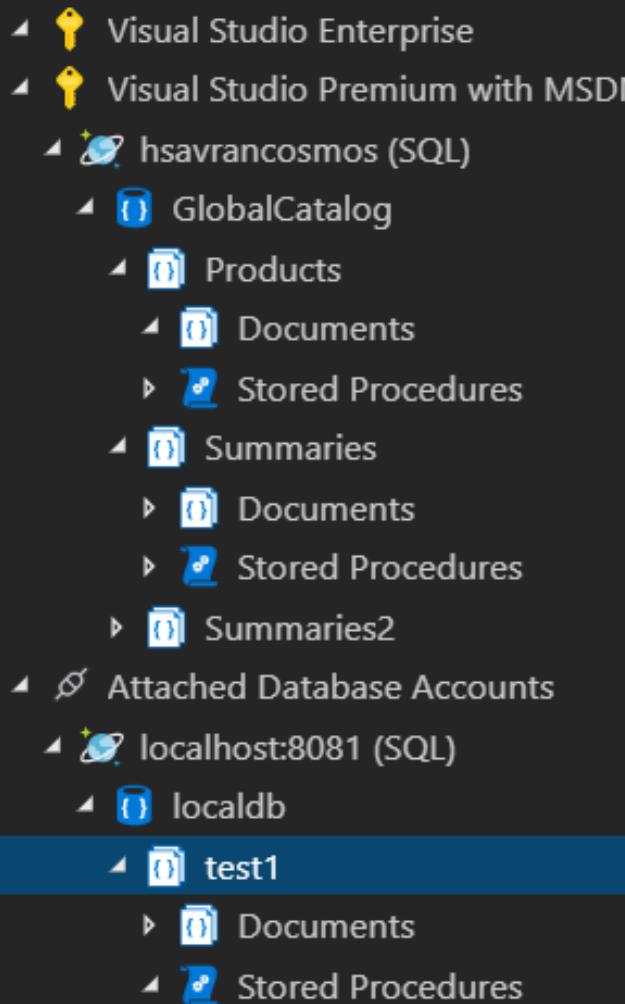
Defaults
Container

JUPYTER NOTEBOOKS





JUPYTER NOTEBOOKS DEMO



VS CODE Azure Database Extension

- Create a Cosmos DB account by clicking the + button in the title
- View Azure Cosmos DB accounts and open directly in the portal
- View/Create/Delete databases, collections, graphs, stored procedures, and documents
- Click on a document or stored procedure to open in the editor
- Click on a graph to visualize data
- Query graph using Gremlin
- Edit a document and persist changes to the cloud
- Attach a Mongo server by clicking the plug icon in the title

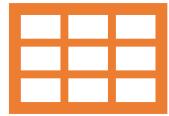
ODBC DRIVER



ODBC DRIVER



SQL API Only



Represent data in Azure Cosmos DB as tables and views

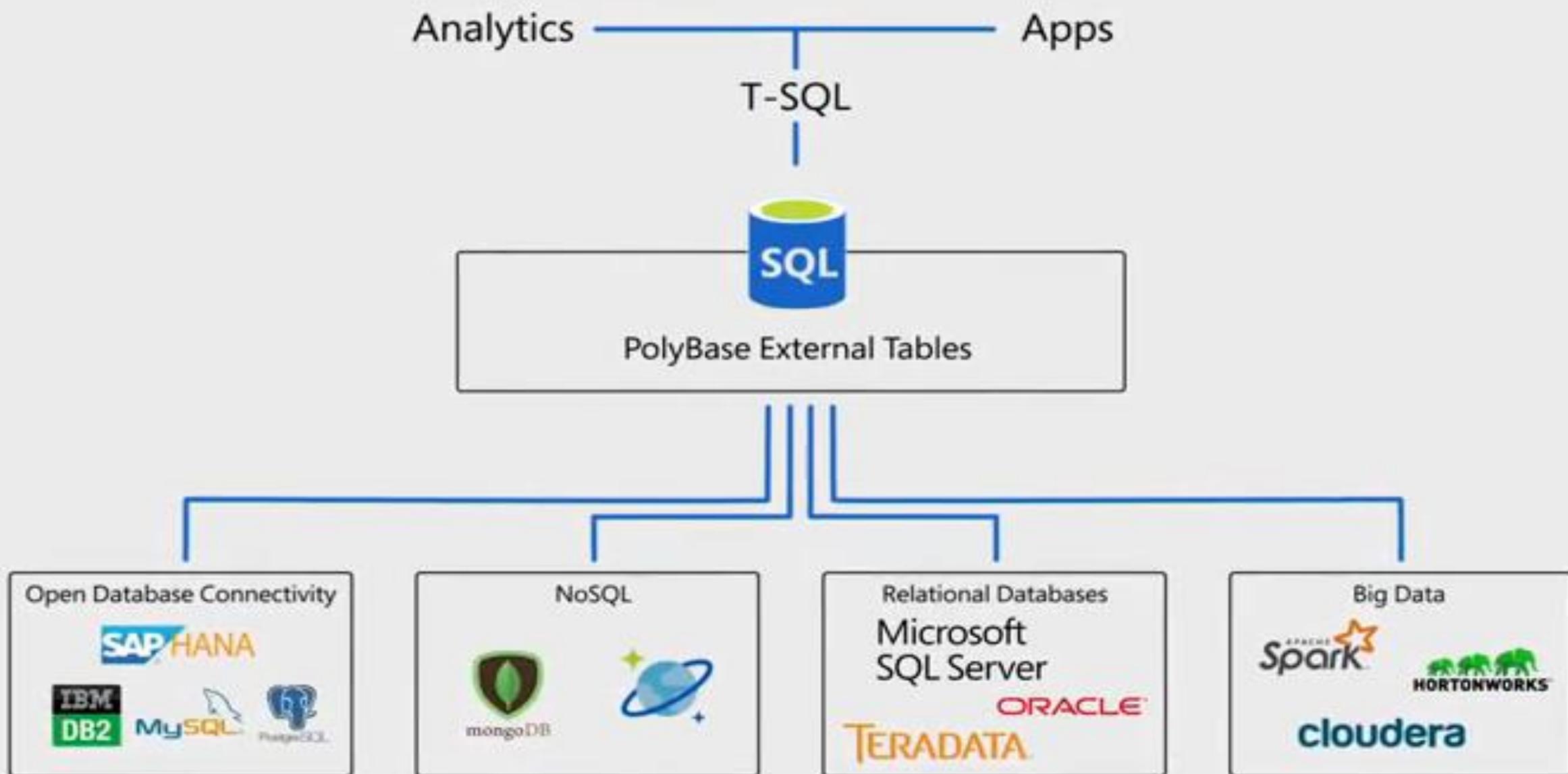


Enables SQL operations against the tables and views including group by queries, inserts, updates, and deletes.



ODBC DEMO

SQL SERVER DATA VIRTUALIZATION

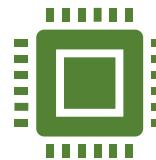


SQL SERVER DATA VIRTUALIZATION

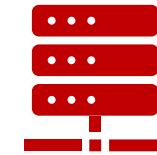
POLYBASE Requirements



.NET 4.5



16 GB RAM



TCP/IP

Enable Polybase

```
exec sp_configure @configname = 'polybase enabled', @configvalue = 1;  
RECONFIGURE WITH OVERRIDE;
```

Confirm Installation

```
SELECT SERVERPROPERTY ('IsPolybaseInstalled') AS IsPolybaseInstalled;
```

SQL SERVER DATA VIRTUALIZATION

1

CREATE MASTER KEY ENCRYPTION BY PASSWORD = 'password'

2

CREATE DATABASE SCOPED CREDENTIAL CosmosCredential
WITH IDENTITY = 'Username'
SECRET = 'MASTER KEY FROM COSMOS DB'

3

CREATE EXTERNAL DATA SOURCE CustomName
WITH (LOCATION = 'mongodb://url:port', CREDENTIAL = CosmosCredential)

4

CREATE SCHEMA cosmosdb

5

CREATE EXTERNAL TABLE TableName (Columns) WITH LOCATION = 'LocalDB.Table',
DATA_SOURCE = CosmosDB

6

CREATE STATISTICS

SQL SERVER DATA VIRTUALIZATION



Case sensitive column names



Security (sysadmin or CONTROL SERVER)



SQL SERVER 2019 only



POLYBASE DEMO

CHANGE FEED



Exposes Cosmos DB logs to outside.
Notifies you immediately for INSERTS & UPDATE
Azure Functions to catch Change Feed changes.

CHANGE FEED WITH AZURE FUNCTIONS

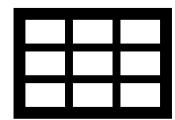
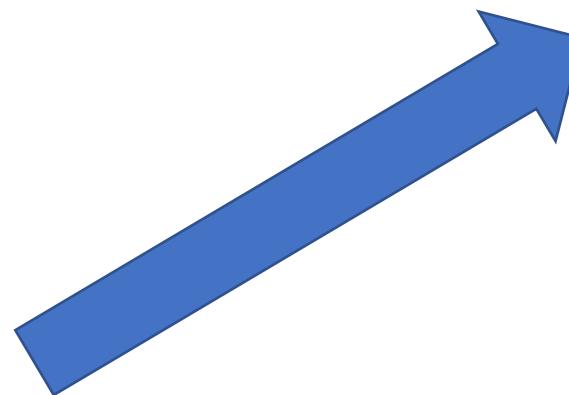
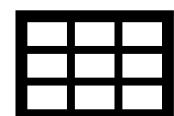


Table to
watch

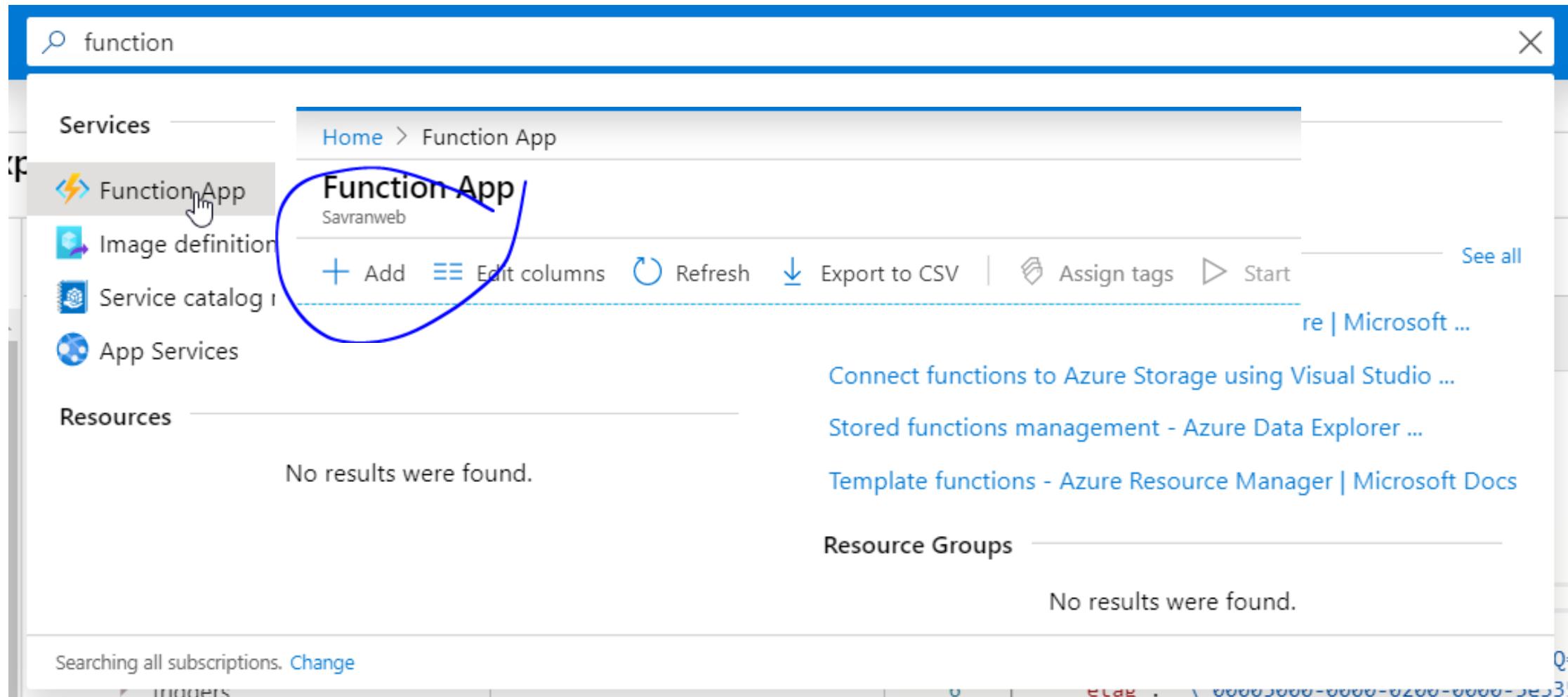


Changes
come to
this table



CHANGE FEED WITH AZURE FUNCTIONS

STEP 1 : CREATE AN AZURE FUNCTION APP



The screenshot shows the Azure portal interface with a search bar at the top containing the text "function". The main content area displays search results under the "Services" category. A blue circle highlights the "Function App" entry, which is listed under the "Savranweb" service. Below the search results, there are sections for "Resources" (with a note that no results were found) and "Resource Groups" (also with a note that no results were found). At the bottom, a status message says "Searching all subscriptions. Change".

function

Services

Function App

Savranweb

Add Edit columns Refresh Export to CSV Assign tags Start See all

Connect functions to Azure Storage using Visual Studio ...

Stored functions management - Azure Data Explorer ...

Template functions - Azure Resource Manager | Microsoft Docs

No results were found.

No results were found.

Searching all subscriptions. Change

CHANGE FEED WITH AZURE FUNCTIONS

Basics Hosting Monitoring Tags Review + create

Create a function app, which lets you group functions as a logical unit for easier management, deployment and sharing of resources. Functions lets you execute your code in a serverless environment without having to first create a VM or publish a web application.

Project Details

Select a subscription to manage deployed resources and costs. Use resource groups like folders to organize and manage all your resources.

Subscription * ⓘ

Microsoft Azure Sponsorship

Resource Group * ⓘ

test

[Create new](#)

Instance Details

Function App name *

AwesomeName

.azurewebsites.net

Publish *

Code Docker Container

Runtime stack *

.NET Core

Region *

East US 2

[Review + create](#)

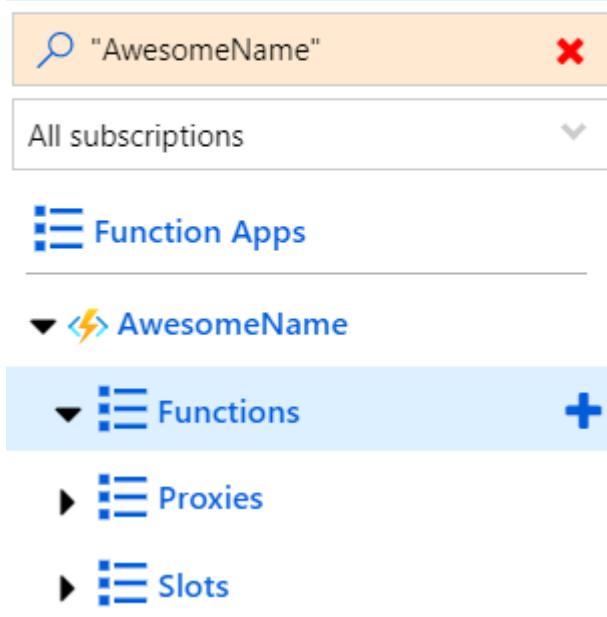
< Previous

Next : Hosting >

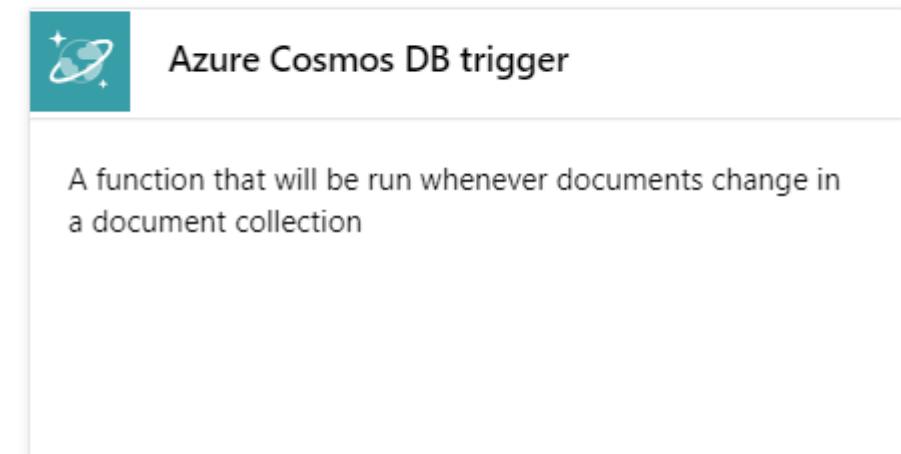
CHANGE FEED WITH AZURE FUNCTIONS

AwesomeName

Function Apps



A screenshot of the Azure Functions blade. At the top left is a search bar with the text "AwesomeName". Below it is a dropdown menu set to "All subscriptions". On the left, there's a sidebar with "Function Apps" selected. Under "AwesomeName", "Functions" is also selected. A blue circle highlights the "New function" button, which has a blue plus sign icon and the text "New function". To the right of the sidebar is a main area with a "f" icon and the word "Functions". Below that is a search bar with the placeholder "Search functions". At the bottom, there's a table header with "NAME" and "STATI" (status) columns, followed by the message "No results".



A screenshot of the "Azure Cosmos DB trigger" blade. It features a teal icon with a white planet and stars. To the right, the text "Azure Cosmos DB trigger" is displayed. Below that is a description: "A function that will be run whenever documents change in a document collection".

CHANGE FEED WITH AZURE FUNCTIONS

The screenshot shows the Azure Functions portal interface for configuring a trigger. On the left, the sidebar shows the project structure under "AwesomeName" with "Functions" selected. The main area displays the trigger configuration:

- Trigger**: Azure Cosmos DB (input)
- Inputs**: Azure Cosmos DB (inputDocument)
+ New Input
- Outputs**: + New Output

Below the trigger configuration, there are fields for setting up the trigger:

- Document collection parameter name: input
- Azure Cosmos DB account connection: stackoverflow2013_DOCUMENTDB
- Database name: Stackoverflow
- Collection name: Posts
- Collection name for leases: leases
- Create lease collection if it does not exist:

CHANGE FEED WITH AZURE FUNCTIONS

The screenshot shows the Azure Functions portal interface. On the left, a sidebar navigation bar includes a search bar ("AwesomeName"), a dropdown for "All subscriptions", and sections for "Function Apps", "AwesomeName", "Functions" (selected), "MyAwesomeTrigger" (under Functions), "Integrate", "Manage", "Monitor", "Proxies", and "Slots". The main workspace is titled "Trigger" and contains an "Azure Cosmos DB (input)" trigger configuration. The "Inputs" section is highlighted with a yellow box and shows an "Azure Cosmos DB (inputDocument)" binding with a "New Input" button. The "Outputs" section shows a "+ New Output" button. The "MyAwesomeTrigger" configuration includes fields for "Document parameter name" (inputDocument), "Collection Name" (posts), "Document ID (optional)", "SQL Query (optional)", "Database name" (stackoverflow), "Azure Cosmos DB account connection" (stackoverflow2013_DOCUMENTDB), and "Partition key (optional)".

Trigger *Trigger*

Azure Cosmos DB (input)

Inputs *Inputs*

Azure Cosmos DB (inputDocument)

+ New Input

Outputs *Outputs*

+ New Output

Azure Cosmos DB input *Azure Cosmos DB input* *delete*

Document parameter name *i*

inputDocument

Collection Name *i*

posts

Document ID (optional) *i*

Document ID (optional)

SQL Query (optional) *i*

SQL Query (optional)

Database name *i*

stackoverflow

Azure Cosmos DB account connection *i* [show value](#)

stackoverflow2013_DOCUMENTDB *v* [new](#)

Partition key (optional) *i*

Partition key (optional)

CHANGE FEED WITH AZURE FUNCTIONS

The screenshot shows the Azure Functions developer portal interface. On the left, the sidebar includes a search bar ("AwesomeName"), a dropdown for "All subscriptions", and sections for "Function Apps", "AwesomeName" (selected), "Functions" (selected), "MyAwesomeTrigger" (selected), "Integrate", "Manage", "Monitor", "Proxies", and "Slots". The main area shows the code editor for "run.csx" with the "Run" button highlighted by a blue oval. The code uses the Microsoft.Azure.DocumentDB.Core package to log information about modified documents. Below the code editor is the log viewer, which displays the following output:

```
2020-01-31T00:02:56 Welcome, you are now connected to log-streaming service. The default timeout is 2 hours.  
Change the timeout with the App Setting SCM_LOGSTREAM_TIMEOUT (in seconds).  
2020-01-31T00:03:02.424 [Error] Executed 'Functions.MyAwesomeTrigger' (Failed, Id=0f944255-2c25-486f-a531-59a807236832)  
Value cannot be null. (Parameter 'o')  
2020-01-31T00:03:02.229 [Information] Executing 'Functions.MyAwesomeTrigger' (Reason='This function was programmatically called via the host APIs.', Id=0f944255-2c25-486f-a531-59a807236832)
```



CHANGE FEED DEMO

Hasan Savran

BI MANAGER



Thank you!



<https://h-savran.blogspot.com/>



hasansavran



SavranWeb