2018 Global Azure BOOTCAMP

CloudGen
ominidi verso l'evoluzione

2018
Global Azure BOOTCAMP
VERONA

Gli sponsor Mondiali

2018 Global Azure BOOTCAMP

CloudGen
ominidi verso l'evoluzione

opsgility

JET BRAINS

SERVICEBUS360

Microsoft

myget

cerebrata

Cloudmonix

Gli sponsor del nostro evento

Platinum Sponsor

Gli sponsor del nostro evento

Gold Sponsor

Gli sponsor del nostro evento

digital joy

dgroove
Tecnologia emozionale.

Basic Sponsor

Tweet della giornata

#GlobalAzure

@cloudgen_verona

CloudGen
ominidi verso l'evoluzione

2018
Global Azure
BOOTCAMP

ARGOMENTO

Azure Cosmos DB

Un moderno database nel cloud

MARTINO BORDIN

@martinobordin

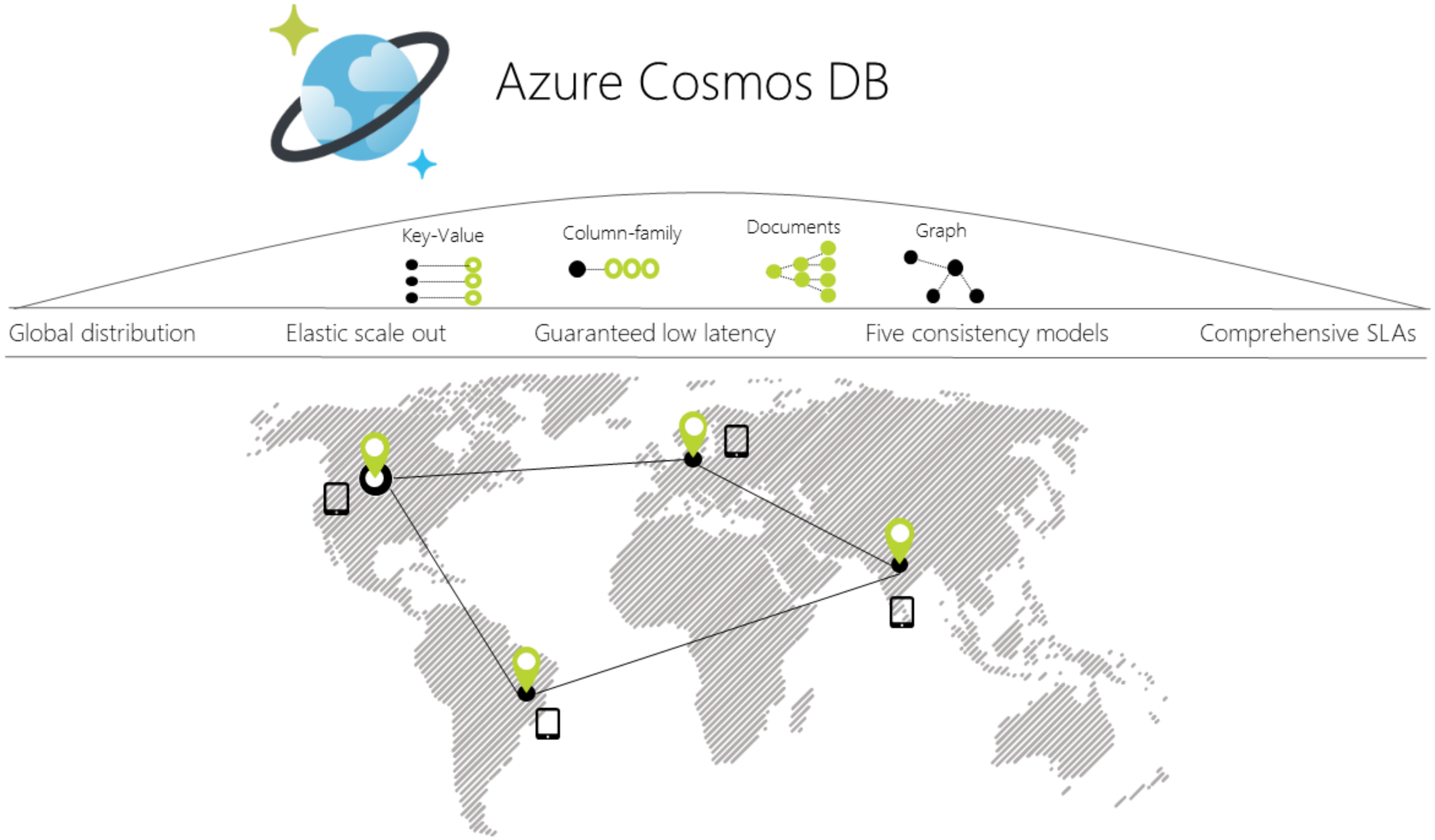
martinobordin

martinobordin

What is Azure Cosmos DB

https://azure.microsoft.com/en-us/services/cosmos-db/

«Globally distributed, multi-model database service»



To SQL or to NoSQL?

- Schemaless
- Horizontal scaling
- Consistency model -> ACID vs Eventual Consistency
- Design & Development
- Alternatives:
  - Key-value: *Redis*
  - Wide-column: *Cassandra*
  - Document: *MongoDb, RavenDb*
  - Graph: *Neo4J*

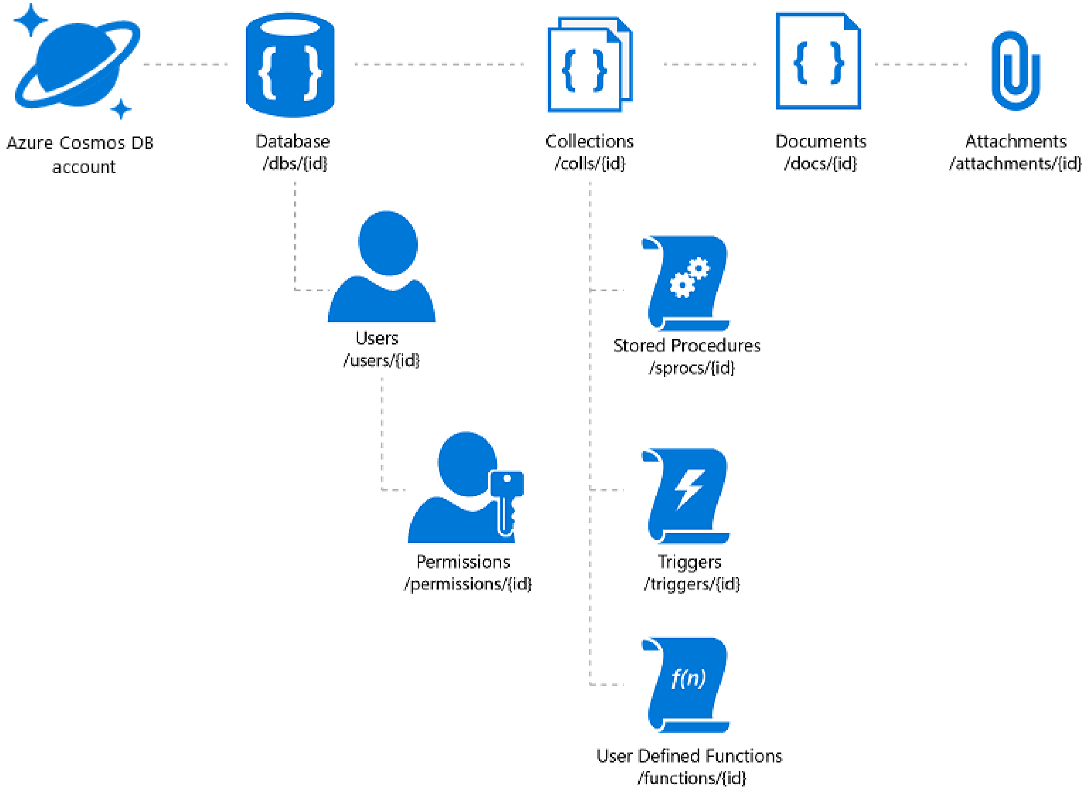
DocumentDB features

Document database on Cosmos DB

- Collections, documents..and more!
- SLA, Security & Logging
- Request Unit & Costs
- Partitions
- Different consistency levels
- Customizable indexing
- Using SQL with NoSQL
- TTL
- Global distribution
- Development
- Emulator
- Tools
- Backup/Restore
- What’s next

Collections, documents..and more!

- Databases
- Collections (Tables)
  - Stored Procedures
  - Triggers
  - User defined functions
- Documents (*Rows/Columns*)
- Attachments

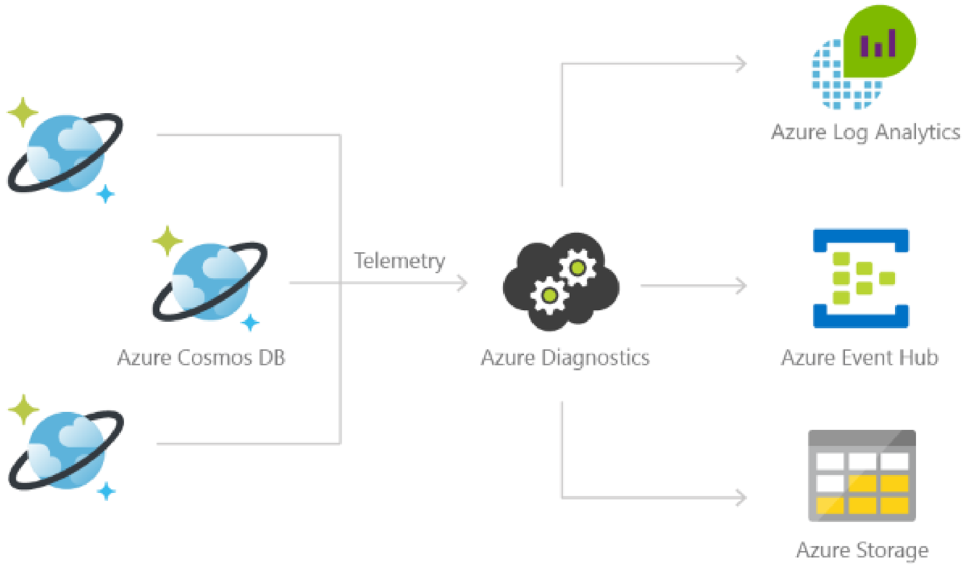


Azure Cosmos DB account
Database /dbs/{id}
Collections /colls/{id}
Documents /docs/{id}
Attachments /attachments/{id}
Users /users/{id}
Stored Procedures /sprocs/{id}
Permissions /permissions/{id}
Triggers /triggers/{id}
User Defined Functions /functions/{id}

SLA, Security & Logging

- SLA
  - 99.999% availability
  - <10 ms reads <15 ms writes
- Security
  - Key authentication:
    - *Master keys* -> Administrative resources (Databases, Users, Permissions) -> **GLOBAL**
    - *Resource tokens* -> Application resources : (Collections, Documents, Attachments, SP, TRI, UDF) -> **USER**
  - **IP** Access Control Policy
  - Data encrypted **in transit** and **at rest**
- Logging
  - CRUD, Database, Access
  - Storage Account, Event HUB, Log Analytics

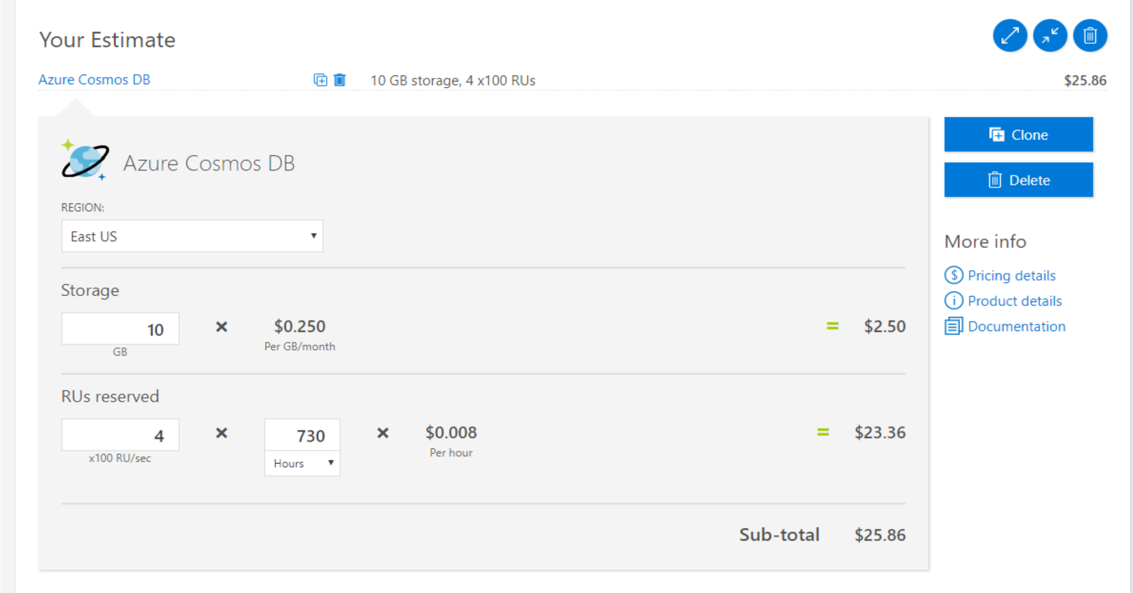
AVAILABILITY (% MONTH)	SERVICE CREDIT
< 99.99%	10%
< 99%	25%



Request Unit & Costs

- Request Unit (https://www.documentdb.com/capacityplanner)
  - Performance is based on **provisioned Request Units/sec**: 1 KB Doc -> 1 RU Read / 5 RU Write
  - Capacity planner

- Costs (https://azure.microsoft.com/en-us/pricing/calculator)
  - SSD Storage -> $0.25 GB/month
  - 100 RUs, (min **400 RUs) ->**$0.008/hour



Your Estimate

Azure Cosmos DB	10 GB storage, 4 x100 RUs	$25.86

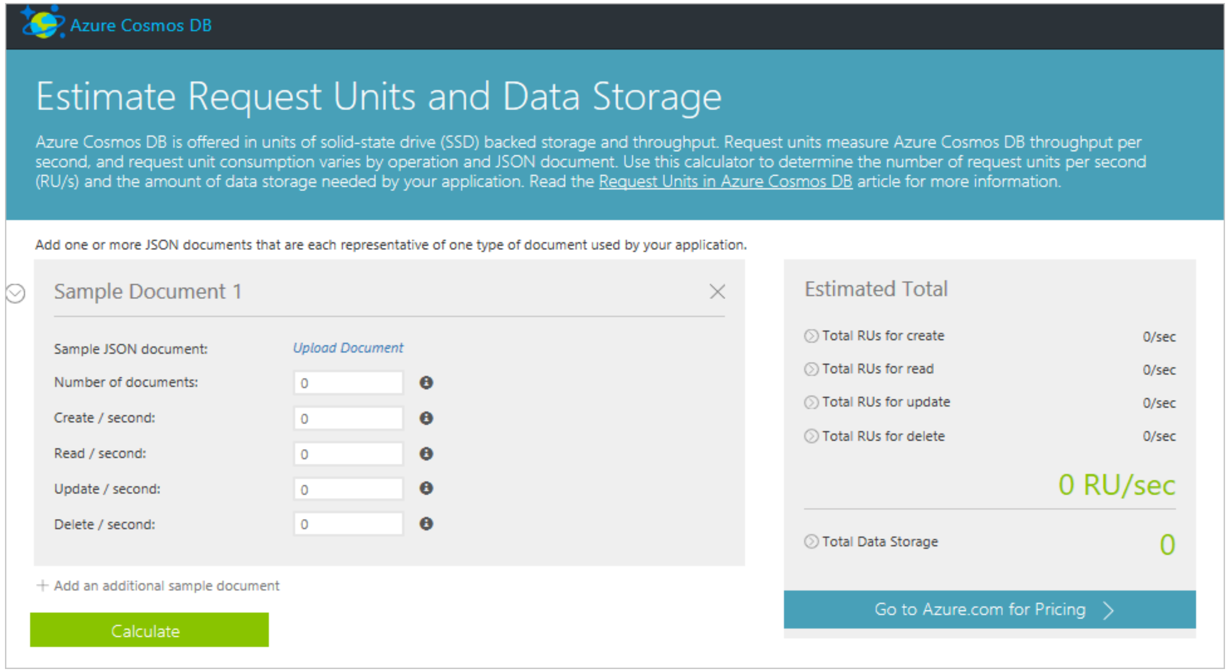
Azure Cosmos DB

REGION: East US

Storage: 10 GB × $0.250 Per GB/month = $2.50

RUs reserved: 4 x100 RU/sec × 730 Hours × $0.008 Per hour = $23.36

Sub-total $25.86



Azure Cosmos DB

Estimate Request Units and Data Storage

Azure Cosmos DB is offered in units of solid-state drive (SSD) backed storage and throughput. Request units measure Azure Cosmos DB throughput per second, and request unit consumption varies by operation and JSON document. Use this calculator to determine the number of request units per second (RU/s) and the amount of data storage needed by your application. Read the Request Units in Azure Cosmos DB article for more information.

Add one or more JSON documents that are each representative of one type of document used by your application.

Sample Document 1

Sample JSON document:	*Upload Document*
Number of documents:	0
Create / second:	0
Read / second:	0
Update / second:	0
Delete / second:	0

+ Add an additional sample document

Estimated Total

Total RUs for create	0/sec
Total RUs for read	0/sec
Total RUs for update	0/sec
Total RUs for delete	0/sec

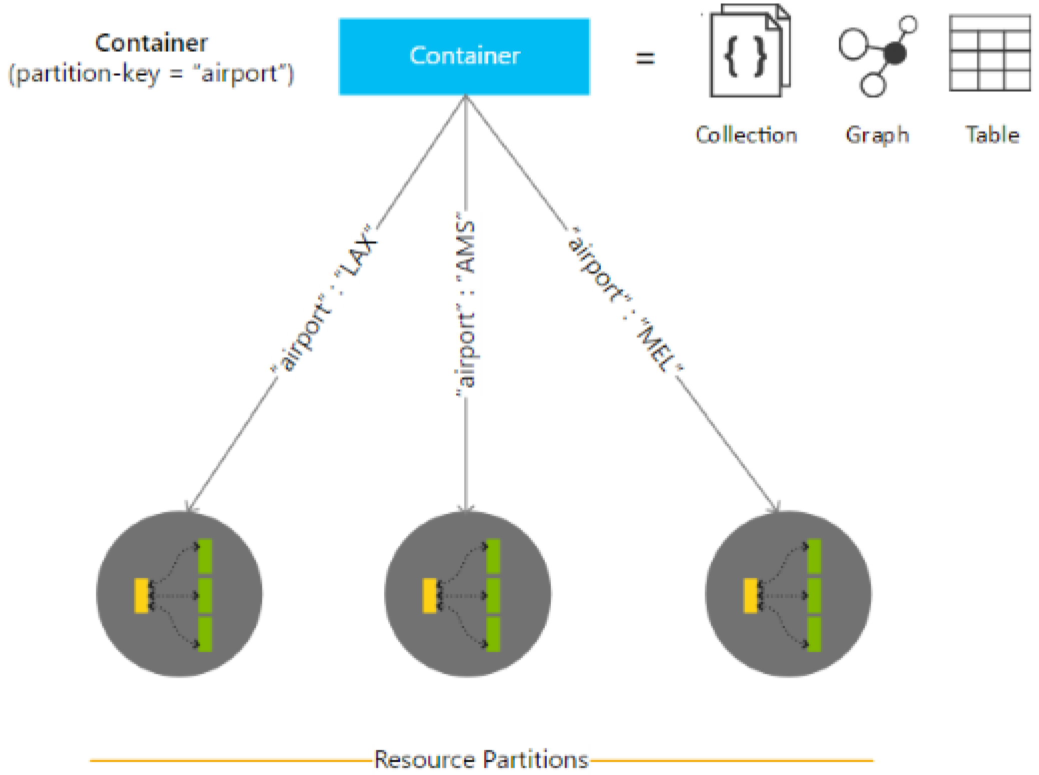
0 RU/sec

Total Data Storage	0

Partitions

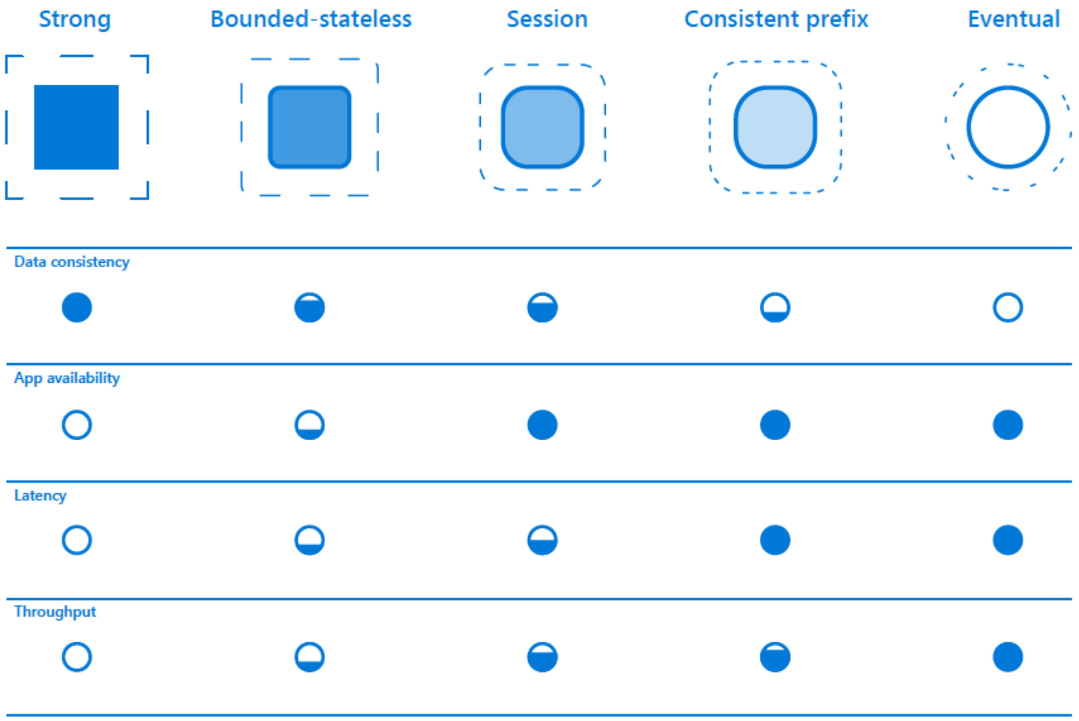
2 container (collections) types:

- Fixed:
  - Min 400 RU/S
  - Max 10,000 RU/s
  - Max 10 GB
- Unlimited
  - Min 1,000 RU/s
  - Max (?) 1,000,000 RU/s
  - **Must have a partition key**



Different consistency levels

- 5 different types:
  - **Strong** -> **RU** ☹ 1 Azure region
  - Bounded Staleness ->reads behind writes
  - Session-> read your own writes guarantees
  - Consistent Prefix->reads never see out of order writes
  - **Eventual** -> the replicas eventually converge **RU** ☺
- Per database, collection or query



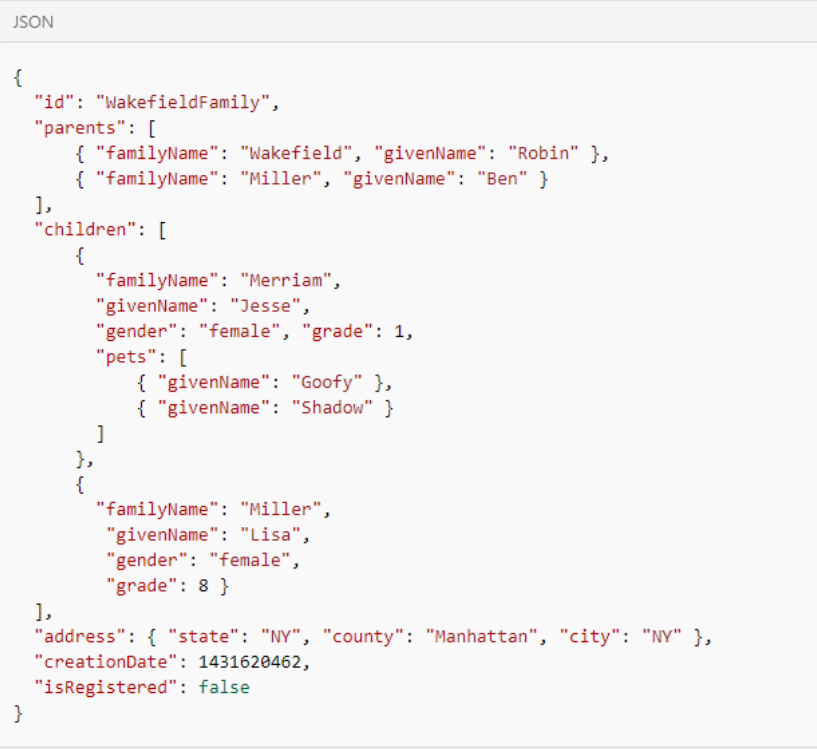
	Strong	Bounded-stateless	Session	Consistent prefix	Eventual
Data consistency	●	◕	◒	◔	○
App availability	○	◔	●	●	●
Latency	○	◔	◒	●	●
Throughput	○	◔	◒	◕	●

Customizable indexing

- All data is indexed automatically (without any assumptions or schema information)
- You can specify **custom indexing policies**
  - Include or exclude documents and paths to and from the index
  - Configure various index types
    - **Hash** -> equality
    - **Range** - > range queries
    - **Spatial** -> “distance” queries
  - Configure index update modes
    - **Consistent** -> updated synchronously as part of the document update
    - **Lazy** -> index is updated asynchronously
    - **None** -> Accessed only by their ID property

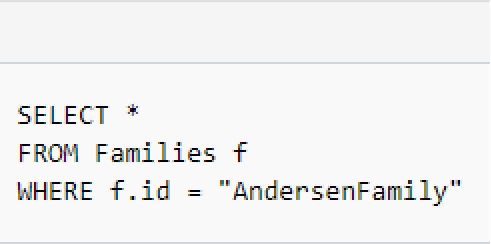
Using SQL with NoSQL

- It deals with hierarchical entities instead of rows and columns
- Schema-less data
- Relations across documents in a collection are captured by containment and not by primary key and foreign key relations

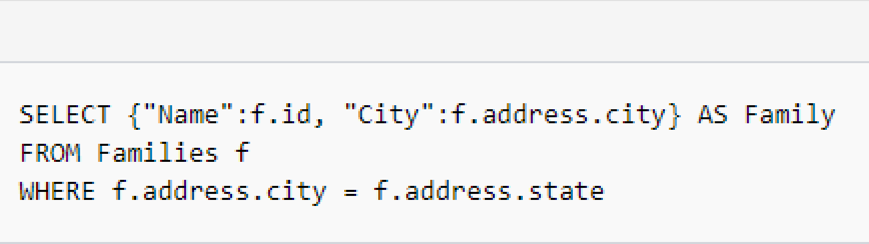


JSON

```
{
  "id": "WakefieldFamily",
  "parents": [
      { "familyName": "Wakefield", "givenName": "Robin" },
      { "familyName": "Miller", "givenName": "Ben" }
  ],
  "children": [
      {
        "familyName": "Merriam",
        "givenName": "Jesse",
        "gender": "female", "grade": 1,
        "pets": [
            { "givenName": "Goofy" },
            { "givenName": "Shadow" }
        ]
      },
      {
        "familyName": "Miller",
         "givenName": "Lisa",
         "gender": "female",
         "grade": 8 }
  ],
  "address": { "state": "NY", "county": "Manhattan", "city": "NY" },
  "creationDate": 1431620462,
  "isRegistered": false
}
```



```
SELECT *
FROM Families f
WHERE f.id = "AndersenFamily"
```



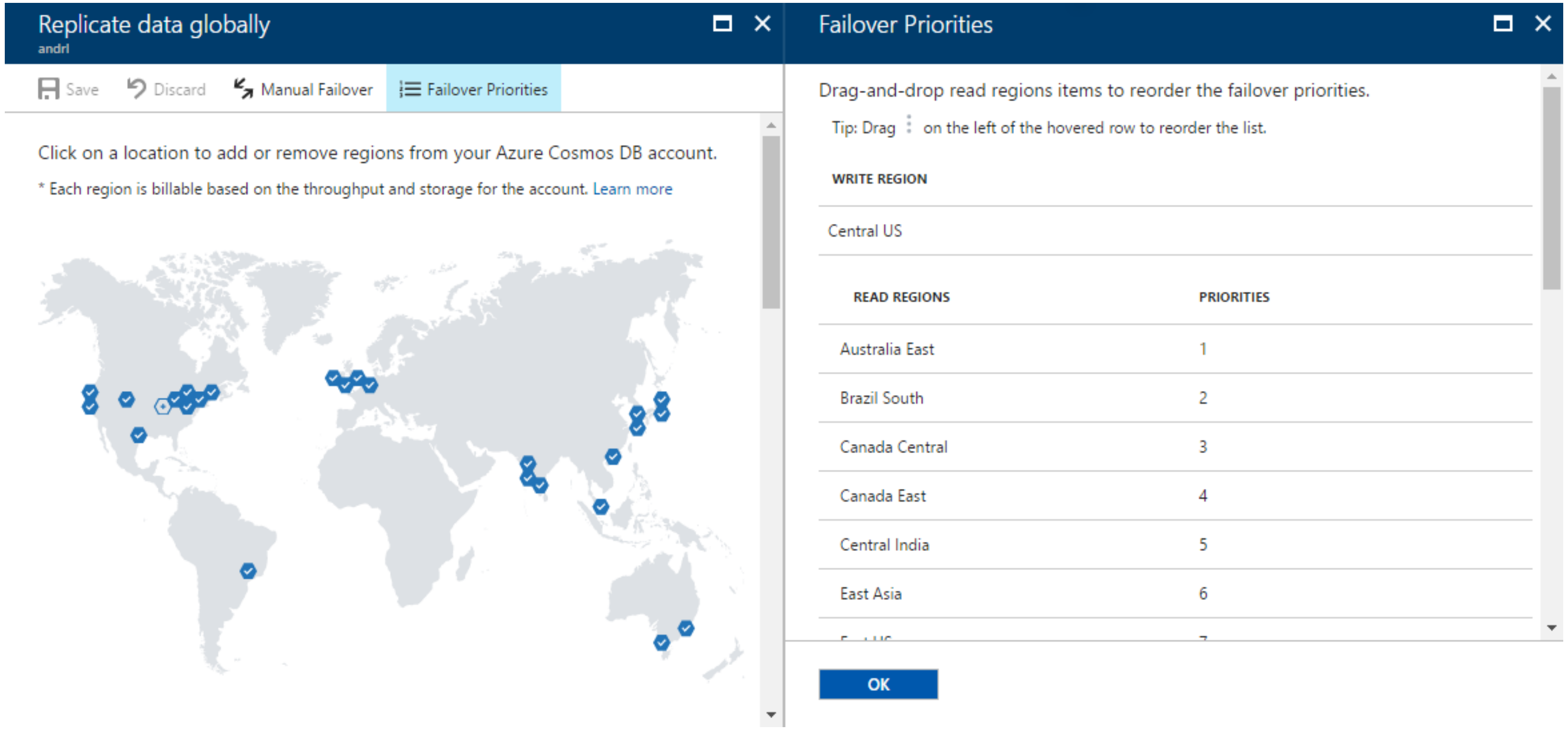
```
SELECT {"Name":f.id, "City":f.address.city} AS Family
FROM Families f
WHERE f.address.city = f.address.state
```

Time To Live Data

- Documents **automatically purged** after a period of time.
- You need to provide the period, in seconds, after the last modified timestamp of the document will be expired:
  - TTL per collection
  - TTL per document
- Expired document won’t be accessible
- No impact on RU

Global distribution

- Low-latency access to data
- Business continuity and disaster recovery
- **1** Write region
- **N** Preferred read regions
- Manual\Automatic failover
- All Azure regions available
- Service\Client configuration



Replicate data globally

andrl

Click on a location to add or remove regions from your Azure Cosmos DB account.

* Each region is billable based on the throughput and storage for the account. Learn more

Failover Priorities

Drag-and-drop read regions items to reorder the failover priorities.

Tip: Drag ⋮ on the left of the hovered row to reorder the list.

WRITE REGION

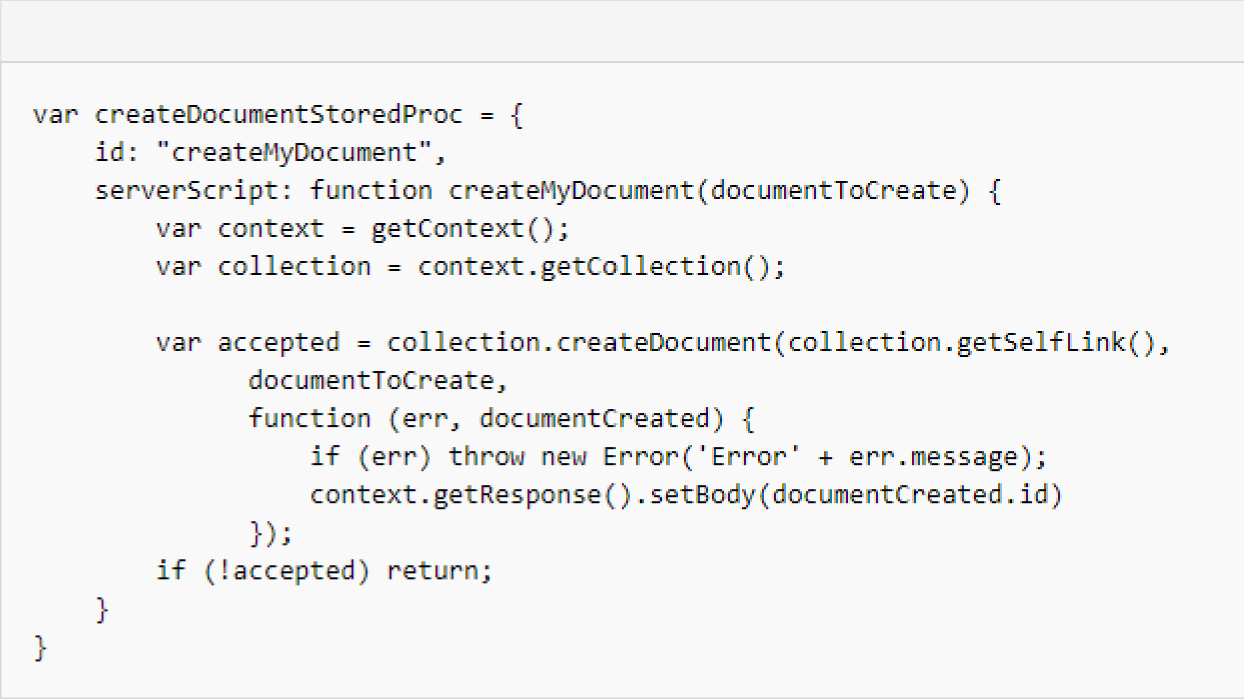
Central US

READ REGIONS	PRIORITIES
Australia East	1
Brazil South	2
Canada Central	3
Canada East	4
Central India	5
East Asia	6

Development

Client side

- .NET
- Java
- JS
- Node.JS
- Pyton



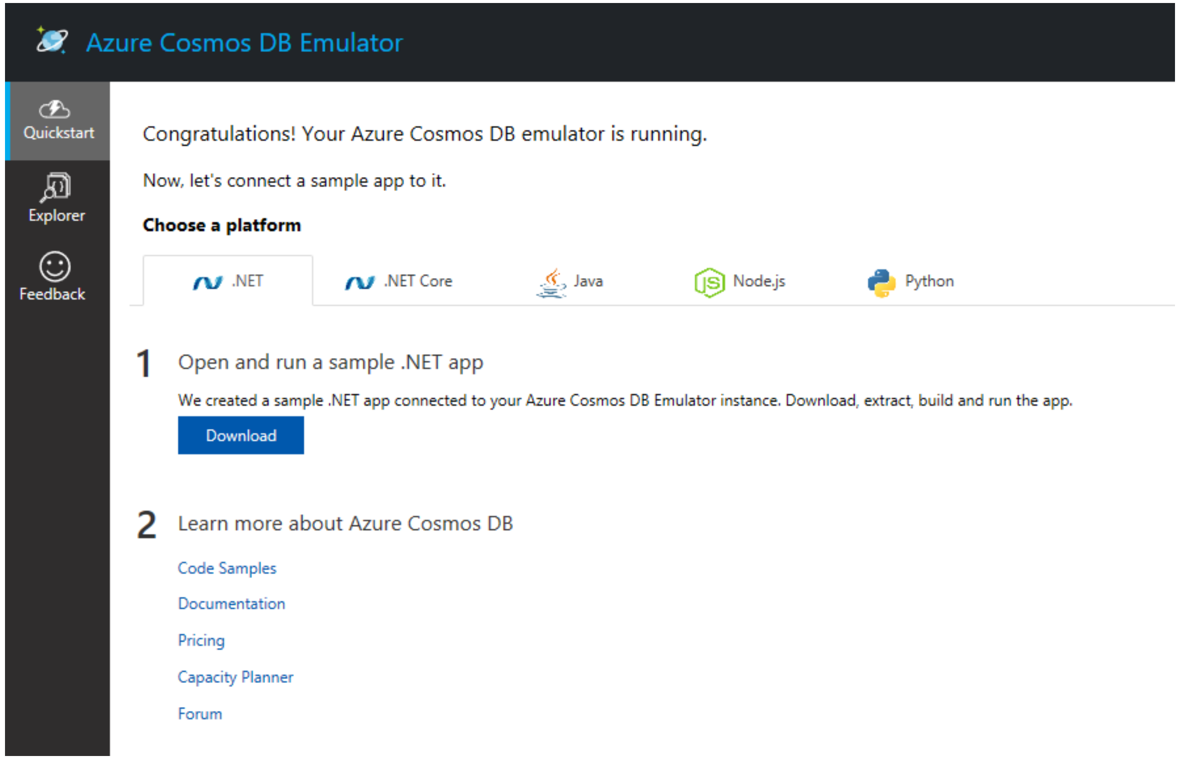
```
var createDocumentStoredProc = {
    id: "createMyDocument",
    serverScript: function createMyDocument(documentToCreate) {
        var context = getContext();
        var collection = context.getCollection();

        var accepted = collection.createDocument(collection.getSelfLink(),
              documentToCreate,
              function (err, documentCreated) {
                  if (err) throw new Error('Error' + err.message);
                  context.getResponse().setBody(documentCreated.id)
              });
        if (!accepted) return;
    }
}
```

Server side

- Javascript
  - Stored Procedure -> transactional batch operations
  - Trigger -> Pre\Post
  - User Defined Function -> extend query language

Emulator

- Local environment that emulates the Azure Cosmos DB service
- For development purposes only
  - Single fixed account and a well-known master key
  - Not scalable
  - Not replicable
  - No consistency levels
  - No encryption
- Runs on https://localhost:8081/
- Command-line tool
- Docker support



Congratulations! Your Azure Cosmos DB emulator is running.

Now, let's connect a sample app to it.

Choose a platform

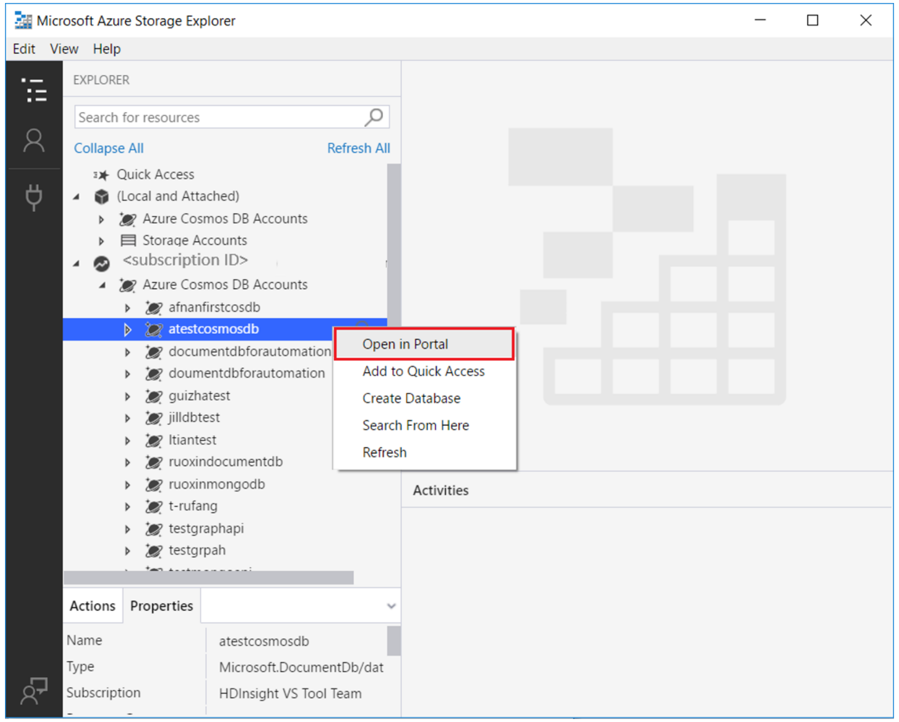
1 Open and run a sample .NET app

We created a sample .NET app connected to your Azure Cosmos DB Emulator instance. Download, extract, build and run the app.

2 Learn more about Azure Cosmos DB

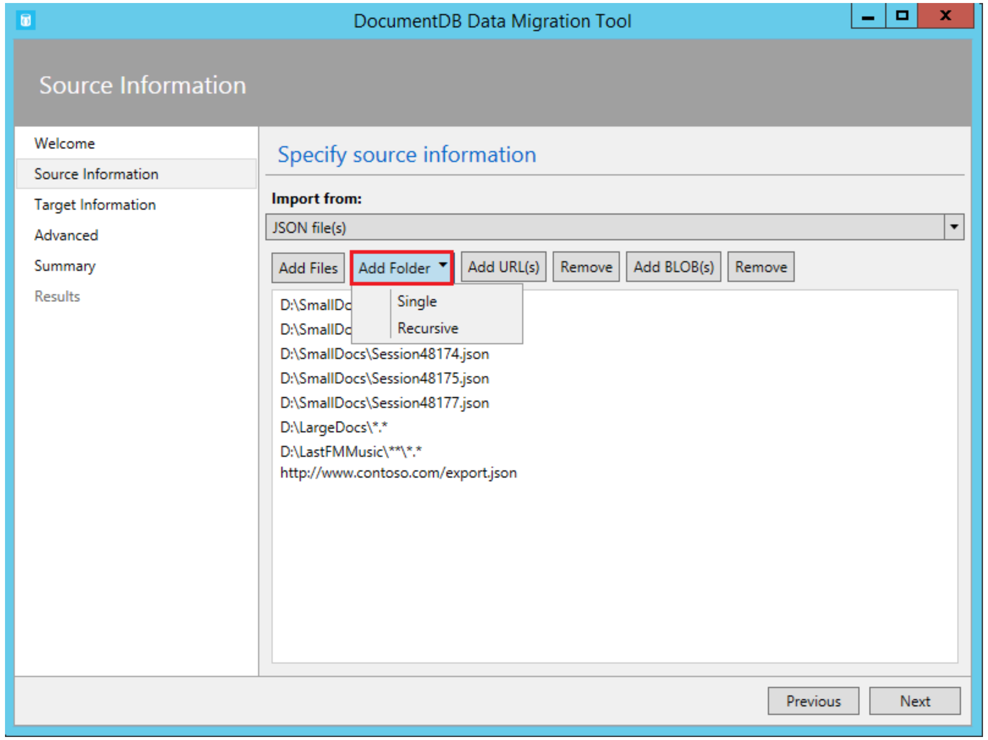
Tools

- **Azure Storage Explorer** https://azure.microsoft.com/en-us/features/storage-explorer/
- Data Migration Tool https://docs.microsoft.com/en-us/azure/cosmos-db/import-data
- DocumentDb Studio https://github.com/mingaliu/DocumentDBStudio
- Query Playground https://www.documentdb.com/sql/demo



Quick Access
(Local and Attached)
Azure Cosmos DB Accounts
Storage Accounts
<subscription ID>
Azure Cosmos DB Accounts
afnanfirstcosdb
atestcosmosdb
documentdbforautomation
doumentdbforautomation
guizhatest
jilldbtest
ltiantest
ruoxindocumentdb
ruoxinmongodb
t-rufang
testgraphapi
testgrpah

Name	atestcosmosdb
Type	Microsoft.DocumentDb/dat
Subscription	HDInsight VS Tool Team



Specify source information

Import from: JSON file(s)

D:\SmallDc
D:\SmallDc
D:\SmallDocs\Session48174.json
D:\SmallDocs\Session48175.json
D:\SmallDocs\Session48177.json
D:\LargeDocs\*.*
D:\LastFMMusic\***.*
http://www.contoso.com/export.json

Backup & Restore

- Backups not affect the performance
- **Every four hours and latest 2 backups are stored ☹**
- **Contact Azure support ☹**

What’s next

https://feedback.azure.com/forums/263030-azure-cosmos-db

- **Allow paging (skip/take)**
- **Partial update**
- GroupBy
- Distinct
- Batch operation (no SP)

Demo

https://github.com/martinobordin/CosmoGAB

Grazie

Domande?

martinobordin

@martinobordin

martinobordin