Working with Azure Cosmos DB – SQL (Core) API

Reza Salehi
CLOUD CONSULTANT

@zaalion linkedin.com/in/rezasalehi2008

Overview

Containers or collections?

Use Data Explorer to manage your data from the portal

Work with your data using SQL syntax

Azure Cosmos DB .NET SDK

- Version 3.x targeting .NET Standard

Demo: Working with the SQL API

Summary

Cosmos DB Containers

An Azure Cosmos container is the unit of scalability for throughput and storage

The container items and the throughput are distributed across a set of logical partitions

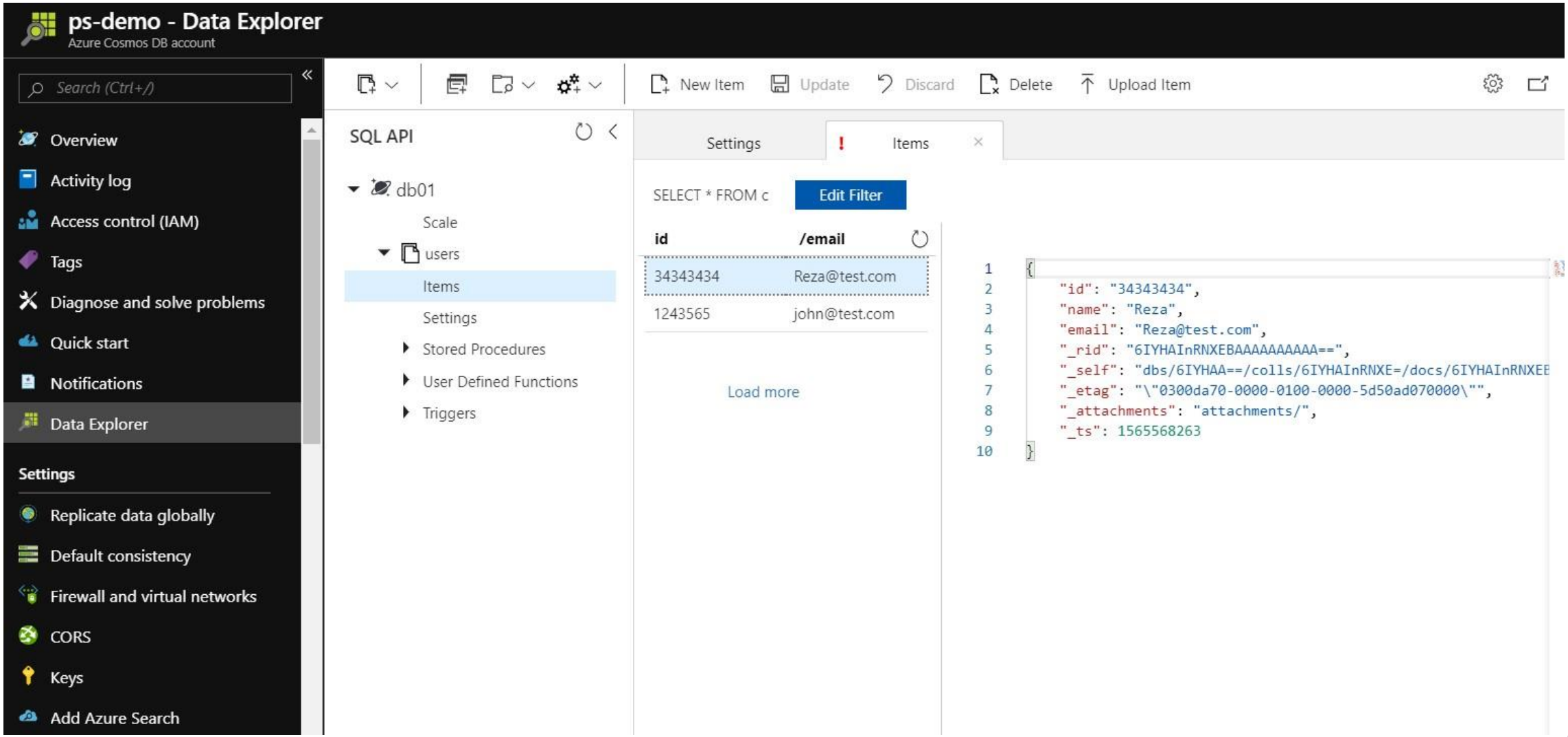
Logical partitions are created based on partition keys

An Azure Cosmos container can scale elastically

Containers in Each API

Azure Cosmos API	Specialized Entity
SQL API	**Collection**
Table API	Table
MongoDB API	Collection
Cassandra API	Table
Gremlin API	Graph

Data Explorer



Azure Cosmos DB SQL API
supports querying
JSON items using SQL.

SQL and JSON

```json
{
    "givenname": "John",
    "lastname": "Smith",
    "address": {
        "streetAddress": "1 sample street",
        "city" : "Boston",
        "country": "USA"
    },
    "email": "john@test.com",
    "phone": "+1 123 4567891"
}
```

```sql
SELECT *
FROM People p
WHERE p.givenname = "John"


SELECT *
FROM People p
WHERE p.address.city = "Boston"


SELECT {"Name":p.id, "City":p.address.city} AS Person
FROM People p
WHERE p.address.city = "Boston"
```

```
using Microsoft.Azure.Cosmos;
```

Azure Cosmos DB Client Library 3.x for .NET

Demo

Using Data Explorer to add data

- Partition keys & unique constraints

Using Cosmos DB data migration tool

Query data using the SQL syntax

Demo

Working with the Cosmos DB client library 3.x for .NET

Summary

Cosmos DB containers

- Containers or collections?

Data Explorer enables you to manage your data from the portal

Use familiar SQL syntax to interact with your Cosmos DB data

Azure Cosmos DB .NET SDK 3.x

Demo: Working with the SQL API