Azure, as a cloud platform, has a lot of storage services, including various SQL databases. This document contains a comparison table of performance results for multi-model database service CosmosDB, using its SQL API, and few most popular SQL databases.



Code in the repository contains Jupiter notebooks to populate databases with test data.

PostgreSQL, SQL Server showed near the same result, however, MySQL was much faster.

Test Data	CosmosDB (SQL API)	PosgreSQL	SQL Server	MySQL (InnoDB)
Dataset: file with 48439 rows, 21 columns, 4.4 MB	Iterative insert of 1000 rows - 335.3 s	Iterative insert of 1000 rows - 525.8 s	Iterative insert of 1000 rows - 477.8 s	Iterative insert of 1000 rows -182.1 s
For benchmark purposes, dataframe with 1000 rows was created basing on the original CSV.	Read data by key - 0.3 s	Read data by key - 0.5 s	Read data by key - 0.3s	Read data by key -0.8 s
https://globaldatalab. org/assets/2019/09/S HDI%20Complete%2 03.0.csv	-fastest read by key		-fastest read by key	-fastest insert

```
CREATE TABLE
                                                                                      CREATE TABLE
                                                                                                                        CREATE TABLE
                                                        public.benchmark
                                                                                      [dbo].[Benchmark](
                                                                                                                        Benchmark(
                            "id": "0",
                                                                                                                         id` Longtext NULL,
                                                                                      [id] [varchar](max) NULL,
                           "iso_code": "AFG",
                                                         id text,
                                                                                      [iso_code] [varchar](max)
                                                                                                                         iso_code` Longtext NULL,
                           "country":
                                                         iso_code text,
                                                                                      NULL
                                                                                                                        `country` Longtext NULL,
                                                                                      [country] [varchar](max) NULL,
                                                         country text,
                                                                                                                         year` Longtext NULL,
GDLCODE` Longtext NULL,
                           "Afghanistan",
                                                                                      [year] [varchar](max) NULL,
[GDLCODE] [varchar](max)
                                                         vear text.
                           "year": "2002",
                                                         adlcode text.
                                                                                                                        `level` Longtext NULL,
                           "GDLCODE": "Ar101",
                                                         level text,
                                                                                      NULL
                                                                                                                         region` Longtext NULL,
                           "level": "Subnat",
                                                        region text,
                                                                                      [level] [varchar](max) NULL,
                                                                                                                         shdi` Longtext NULL,
                                                         shdi text,
                                                                                      [region] [varchar](max) NULL,
                                                                                                                         healthindex` Longtext NULL,
                           "region": "Central",
                                                         healthindex text,
                                                                                      [shdi] [varchar](max) NULL,
                                                                                                                        'incindex' Longtext NULL,
                           "shi": ".454",
                                                        incindex text.
                                                                                       [healthindex] [varchar](max)
                                                                                                                         edindex` Longtext NULL,
                                                                                                                        `lifexp` Longtext NULL, 
`lgnic` Longtext NULL,
                           "healthindex": "58",
                                                         edindex text.
                                                                                      NULL
                                                                                      [incindex] [varchar](max)
                                                        lifexp text
                           "incindex": ".394",
Example of
                                                                                                                        'esch' Longtext NULL,
                                                                                      NULL.
                                                        lanic text.
                           "edindex": ".409",
                                                        esch text,
                                                                                      [edindex] [varchar](max) NULL,
                                                                                                                        'msch' Longtext NULL
database structures
                           "lifexp": "57.7",
                                                                                      [lifexp] [varchar](max) NULL,
                                                        msch text
                                                                                      [lgnic] [varchar](max) NULL,
                          "lgnic": "7.213",
                                                                                      [esch] [varchar](max) NULL,
                          "esch": "9.1",
                                                                                      [msch] [varchar](max) NULL
) ON [PRIMARY]
                          "msch": "4.7",
                                                                                      TEXTIMAGE ON [PRIMARY]
                           "pop": "3443.952",
                          " rid": "",
                          "_selfAA==/",
                          " etag":
                           "\"b800871e-0000-070
                          0-0000-6128ef410000\
                               "_attachments":
                           "attachments/",
                               "_ts":
                          1630072641
                           }
```

## **SQL Dbs settings:**

To reach similar condition for all SQL Dbs, I used tables with max string column types for each database:

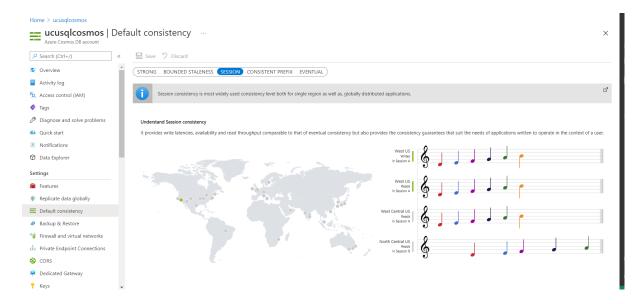
text for PosgreSQL, [varchar](max) for SQL Server, Longtext for MySQL (InnoDB)

Each result was calculated basing on 3 executions.

Each database was created using the lowest possible performance capabilities in the cloud. Data was inserted without using bufferisation, however buffesition is possible to use for MySQL and PostgreSQL.

## CosmosDB (SQL API) settings:

1) Cosmos has default consistency level named as session -



Check Microsoft animation for more details <a href="https://docs.microsoft.com/en-us/azure/cosmos-db/consistency-levels">https://docs.microsoft.com/en-us/azure/cosmos-db/consistency-levels</a>

## 2) The lowest possible capacity was used for CosmosDB (Throughput (RU/s) - 400).

Database	Throughput (RU/s)
ucudatabase	400

## 3) No replication was enabled for benchmark

