

Platforma Azure SQL Server

Azure SQL este o gamă de produse inteligente, construită pe fundația SQL Server în cloud-ul Azure, care facilitează migrația aplicațiilor și continuarea utilizării resurselor cu care utilizatorii sunt deja familiarizați în cloud. Aceasta folosește principiul *Pay as you go* adică utilizatorii plătesc doar pentru resursele pe care le utilizează, cum se întâmplă și pentru alte servicii din cloud.

Cele trei produse din familia Azure SQL sunt:

- Azure SQL Database: Oferă suport pentru gestionarea bazelor de date pentru aplicațiile în cloud.
- Azure SQL Managed Instance: Oferă o instanță compatibilă cu SQL Server.
- Azure Virtual Machines: Transferă sarcinile de lucru SQL Server, menține o compatibilitate cu SQL Server și acces la nivel de sistem de operare.

DBaaS: Azure SQL

Baza de date Azure SQL este un serviciu de gestionare a bazelor de date relaționale (DBaaS), potrivit pentru aplicațiile din cloud care utilizează funcționalitățile SQL Server printr-un proces rapid de dezvoltare. Aceasta oferă opțiuni de scalare pentru o performanță ridicată și include caracteristici adiționale care nu sunt oferite de SQL Server (spre exemplu built-in high availability și funcționalitățile de gestionare).

Există două moduri principale de implementare:

- Single database: o singură bază de date, având propriile sale resurse, gestionate printr-un server logic.
- Elastic pool: reprezintă o grupare de baze de date care împărtășesc un set comun de resurse, fiind administrate printr-un server logic. Două avantaje ale folosirii elastic pools sunt costurile reduse pentru gestionarea multiplelor baze de date și procesul propriu-zis de gestionare, bazele de date individuale putând fi mutate în/din acestea.

Arhitectura Azure SQL

Există 4 straturi în arhitectura Azure SQL:

1. Client Layer

Acesta este stratul prin care utilizatorii interacționează cu serviciile Azure SQL. Include aplicațiile, interfețele de utilizator și alte sisteme folosite pentru serviciile Azure SQL.

2. Service Layer

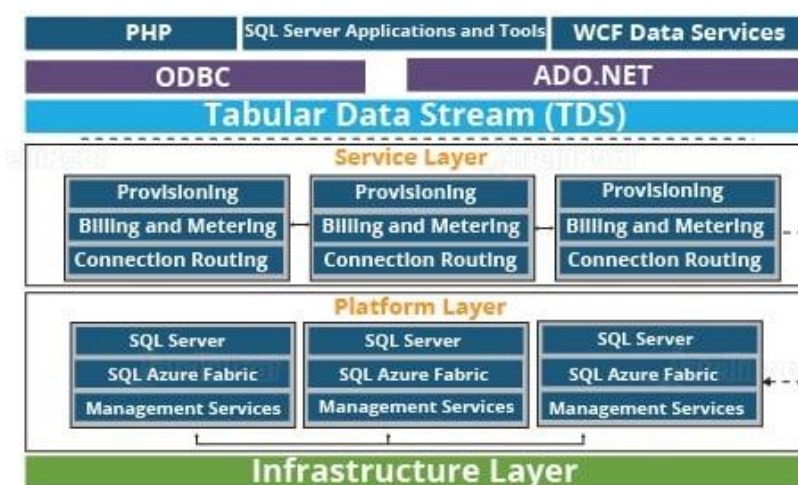
Acest strat reprezintă serviciile oferite de Azure SQL, cum ar fi gestionarea bazelor de date, securitatea și monitorizarea performanței.

3. Platform Layer

Aici se găsesc componente specifice platformei Azure SQL, inclusiv motorul de baze de date, mecanismele de optimizare, instrumentele de dezvoltare și alte funcționalități necesare pentru administrarea bazei de date.

4. Infrastructure Layer

Acesta este stratul fizic care susține și găzduiește întreaga infrastructură necesară, inclusiv serverele fizice, rețelele, stocarea datelor și alte resurse hardware și software.



Prezentarea arhitecturii Azure SQL¹

Operații CRUD cu Azure SQL

Folosind contul de student unibuc am creat o bază de date în cloud-ul Azure și am ilustrat funcționalitățile CRUD după cum urmează:

1. Configurarea bazei de date (single database) conform instrucțiunilor din [documentatia Microsoft](#).
2. Crearea tabelului Employee și popularea acestuia cu date

¹ [Akshay Shukla; Microsoft Azure SQL Database Tutorial; 30 Nov 2023](#)

The screenshot shows the Azure Data Studio interface with a query editor. The query is as follows:

```
1 CREATE TABLE Employee (  
2     employee_id INT PRIMARY KEY,  
3     last_name NVARCHAR(30),  
4     first_name NVARCHAR(70),  
5     salary INT  
6 );  
7  
8 INSERT INTO Employee (employee_id, last_name, first_name, salary) VALUES (1, 'Doe', 'John', 10000);  
9 INSERT INTO Employee (employee_id, last_name, first_name, salary) VALUES (2, 'Evans', 'Oliver', 5000);
```

The Results pane shows the message: "Query succeeded: Affected rows: 3". The Object Explorer on the left shows the database structure, including the Employee table with columns: employee_id (PK, int, not null), last_name (nvarchar, null), first_name (nvarchar, null), and salary (int, null).

3. Afișarea datelor

The query editor shows the following SQL query:

```
5     salary INT  
6 );  
7  
8 INSERT INTO Employee (employee_id, last_name, first_name, salary) VALUES (1, 'Doe', 'John', 10000);  
9 INSERT INTO Employee (employee_id, last_name, first_name, salary) VALUES (2, 'Evans', 'Oliver', 5000);  
10 INSERT INTO Employee (employee_id, last_name, first_name, salary) VALUES (3, 'Wilson', 'Eva', 13000);  
11  
12 SELECT * FROM [dbo].[Employee];  
13
```

The Results pane shows the following table:

| employee_id | last_name | first_name | salary |
|-------------|-----------|------------|--------|
| 1 | Doe | John | 10000 |
| 2 | Evans | Oliver | 5000 |
| 3 | Wilson | Eva | 13000 |

4. Schimbarea prenumelui pentru angajatul cu id = 1, urmat de afișarea datelor prin select

The query editor shows the following SQL query:

```
11  
12 SELECT * FROM [dbo].[Employee];  
13  
14 UPDATE [dbo].[Employee] SET first_name = 'Smith' WHERE employee_id = 1;  
15
```

The Results pane shows the following table:

| employee_id | last_name | first_name | salary |
|-------------|-----------|------------|--------|
| 1 | Doe | Smith | 10000 |
| 2 | Evans | Oliver | 5000 |
| 3 | Wilson | Eva | 13000 |

5. Ștergerea angajatului cu id = 2, urmată de afișarea datelor prin select

```
11
12 SELECT * FROM [dbo].[Employee];
13
14 UPDATE [dbo].[Employee] SET first_name = 'Smith' WHERE employee_id = 1;
15
16 DELETE FROM [dbo].[Employee] WHERE employee_id = 2;
17
```

Results Messages

Search to filter items...

| employee_id | last_name | first_name | salary |
|-------------|-----------|------------|--------|
| 1 | Doe | Smith | 10000 |
| 3 | Wilson | Eva | 13000 |

6. Filtrarea angajaților cu salariu mai mic ca 12000 printr-un select

```
15
16 DELETE FROM [dbo].[Employee] WHERE employee_id = 2;
17
18 SELECT last_name, salary
19 FROM [dbo].[Employee]
20 WHERE salary < 12000;
21
```

Results Messages

Search to filter items...

| last_name | salary |
|-----------|--------|
| Doe | 10000 |

Codul folosit pentru exemplele CRUD

```
CREATE TABLE Employee (
    employee_id INT PRIMARY KEY,
    last_name NVARCHAR(30),
    first_name NVARCHAR(70),
    salary INT
);

INSERT INTO Employee (employee_id, last_name, first_name, salary) VALUES (1, 'Doe', 'John', 10000);
INSERT INTO Employee (employee_id, last_name, first_name, salary) VALUES (2, 'Evans', 'Oliver', 5000);
INSERT INTO Employee (employee_id, last_name, first_name, salary) VALUES (3, 'Wilson', 'Eva', 13000);

SELECT * FROM [dbo].[Employee];

UPDATE [dbo].[Employee] SET first_name = 'Smith' WHERE employee_id = 1;

DELETE FROM [dbo].[Employee] WHERE employee_id = 2;

SELECT last_name, salary
FROM [dbo].[Employee]
WHERE salary < 12000;
```

Interoperarea cu soluțiile on-premises

Interoperabilitatea pentru Azure SQL Server este esențială pentru multe organizații care folosesc un amestec de resurse în cloud și în infrastructura lor locală. Următoarele soluții au fost create pentru a rezolva aceste tipuri de interacțiuni în funcție de nevoile utilizatorilor:

- Azure Hybrid Benefit este un program care aduce avantaje hibride pentru clienții care dețin licențe Software Assurance pentru diverse produse Microsoft. Prin intermediul acesteia, clienții pot beneficia de tarife reduse atunci când vor să migreze bazele de date existente de SQL Server către Azure SQL.
- Azure Data Sync reprezintă un instrument care permite sincronizarea și replicarea datelor între baze de date (Azure SQL și o bază de date locală/între două baze de date Azure SQL din regiuni diferite).
- Azure ExpressRoute furnizează o conexiune privată între infrastructura on-premises și resursele Azure, oferind o soluție sigură, de mare viteză. Acest lucru facilitează accesul direct la resursele din Azure, inclusiv la bazele de date Azure SQL, fără a trece prin rețeaua publică.
- Linked Servers reprezintă o funcționalitate în SQL Server care permite configurarea unui *linked server* pentru a interoga datele din Azure SQL Server folosind comenzi T-SQL dintr-un server local.
- SQL Server Integration Services (SSIS) permite migrarea datelor între Azure SQL și bazele de date locale prin intermediul pachetelor SSIS.
- Replicarea reprezintă un set de opțiuni disponibile în SQL Server pentru replicarea datelor între instanțele SQL Server on-premises și Azure SQL Server, permițând astfel sincronizarea continuă a datelor între aceste medii.

Referințe bibliografice:

[1] [What is Azure SQL? Parte din documentatia oficială Microsoft](#)

[2] [Azure SQL documentatia oficială](#)

[3] [Accelebrate: Azure SQL vs. SQL Server: How to Choose the Right Database; 25 Oct 2022 in Database and Big Data Articles;](#)

[4] [Akshay Shukla; Microsoft Azure SQL Database Tutorial; 30 Nov 2023](#)

[5] [SQL Azure CRUD operations with Azure Functions; 5 Jun 2017](#)

[6] [SQL Server Tutorial For Beginners and Professionals](#)