Resolve external table related issues in Azure SQL Database

Last updated by | Peter Hewitt | Jan 16, 2023 at 6:55 AM PST

Contents

Resolve external table related issues in Azure SQL Database

This content is based on the how to resolve external issues self-help article that's displayed to customers in the Azure portal when creating a support ticket.

External tables are created for implementing elastic queries (preview). The elastic query feature allows you to run Transact-SQL (T-SQL) queries across multiple databases in Azure SQL Database.

Learn how to resolve external table related issues in SQL Database by reviewing the most common issues and solutions.

Common issues and solutions

Issue	Recommendation solutions
Error message 46832: "An error occurred while establishing connection to remote data source: [Microsoft][ODBC Driver 17 for SQL Server][SQL Server] ..."	Confirm the elastic query in the local database has access to the remote database. In the remote SQL Database server's firewall configuration, ensure the option *Public network access* to *Selected networks* is allowed, and the exception *Allow Azure services and resources to access this server* is enabled. If either of these options are disabled, the external table can't connect to the remote database. Private links are currently not supported with elastic query for databases that are targets of external data sources. An alternative solution is to move the SQL databases to an Azure SQL Managed Instance, which natively allows cross-database queries.
Error message 46836: "External table schema does not match actual schema from remote table: Mismatch between actual and expected type of column ..."	Occurs when there are data types mismatches in the columns of the internally generated remote query. Add appropriate type casts or change literal types. For example, specifying Unicode string literal N'abc' instead of 'abc'.
Slow performance	Avoid including NVARCHAR(max) columns in the external table. Columns of this data type will disable the advanced batching techniques that are used in the elastic query implementation. If an external table includes NVARCHAR(max) columns it will significantly affect query performance when a large amount of data needs to be transferred from the remote database to the local database. General elastic queries performance best practices: - Elastic queries perform best for queries where most of the computation can be done on the remote databases. - For elastic queries over large databases, choose a higher-level General Purpose or Business Critical service tier for improved performance. - Elastic query isn't the preferred solution for extract, transform, and load (ETL) operations that involve heavy processing with large amounts of data

Issue	Recommendation solutions
	movement. If applicable, consider migrating to a SQL Managed Instance.
A local table with the same name already exists	The local database already contains a table with the same name as the table in the remote database. The solution is to create the external table with a different table name or schema name, and reference the details of the remote database object in the SCHEMA_NAME and OBJECT_NAME parameters in the CREATE EXTERNAL TABLE command. For example: -- execute in local database CREATE EXTERNAL TABLE [dbo].[otherDBCustomer] ([CustomerID] [int] NOT NULL, [CustomerName] [varchar] (50) NOT NULL, [Company] [varchar] (50) NOT NULL) WITH (DATA_SOURCE = MyElasticQueryDataSrc, SCHEMA_NAME = 'dbo', OBJECT_NAME = 'Customer');

Public Doc Reference

Get started with cross-database queries
Code samples that show how to create an external data source and an external table.

Internal Reference

Elastic Query troubleshooting

Root Cause Classification

Cases resolved by this TSG should be coded to the following root cause:
Root Cause: Azure SQL v3\Performance\How-to/advisory

How good have you found this content?