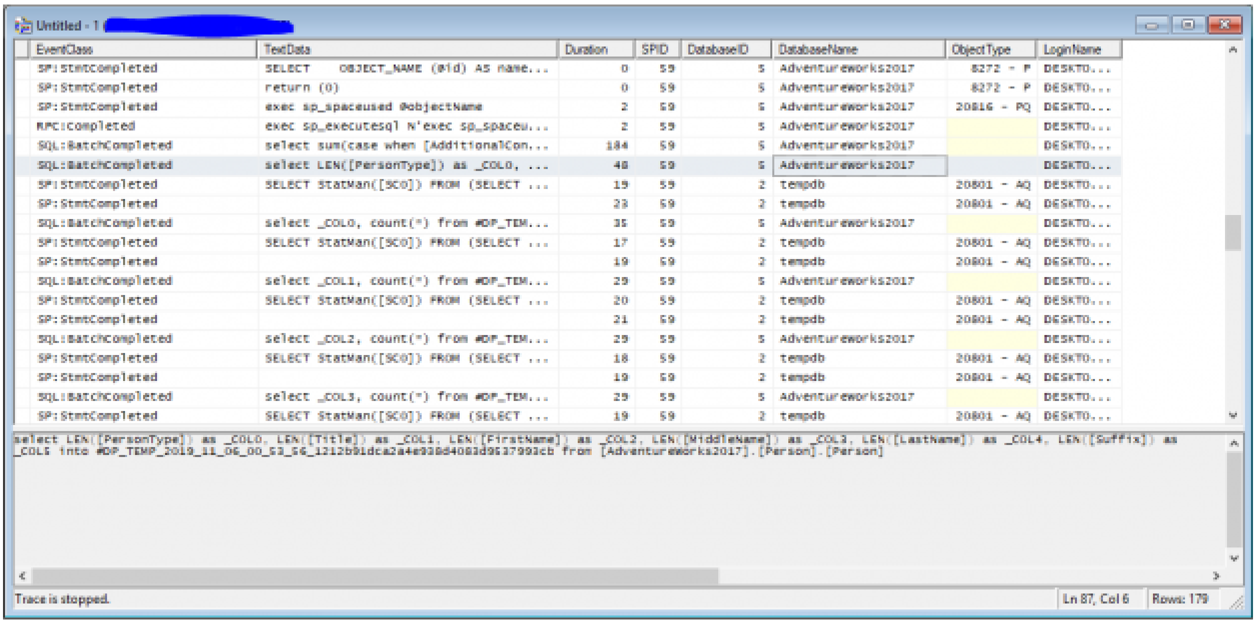
An overview of the Data Profiling task in SSIS

sqlshack.com/overview-of-data-profiling-task-in-ssis

Hadi Fadlallah

November 21, 2019



EventClass	TextData	Duration	SPID	DatabaseID	DatabaseName	ObjectType	LoginName
SP:StmtCompleted	SELECT OBJECT_NAME (@id) AS name...	0	59	5	AdventureWorks2017	8272 - P	DESKTO...
SP:StmtCompleted	return (0)	0	59	5	AdventureWorks2017	8272 - P	DESKTO...
SP:StmtCompleted	exec sp_spaceused @objectName	2	59	5	AdventureWorks2017	20816 - PQ	DESKTO...
RPC:Completed	exec sp_executesql N'exec sp_spaceu...	2	59	5	AdventureWorks2017		DESKTO...
SQL:BatchCompleted	select sum(case when [AdditionalCon...	184	59	5	AdventureWorks2017		DESKTO...
SQL:BatchCompleted	select LEN([PersonType]) as _COL0, ...	48	59	5	AdventureWorks2017		DESKTO...
SP:StmtCompleted	SELECT StatMan([SC0]) FROM (SELECT ...	19	59	2	tempdb	20801 - AQ	DESKTO...
SP:StmtCompleted		23	59	2	tempdb	20801 - AQ	DESKTO...
SQL:BatchCompleted	select _COL0, count(*) from #DP_TEM...	35	59	5	AdventureWorks2017		DESKTO...
SP:StmtCompleted	SELECT StatMan([SC0]) FROM (SELECT ...	17	59	2	tempdb	20801 - AQ	DESKTO...
SP:StmtCompleted		19	59	2	tempdb	20801 - AQ	DESKTO...
SQL:BatchCompleted	select _COL1, count(*) from #DP_TEM...	29	59	5	AdventureWorks2017		DESKTO...
SP:StmtCompleted	SELECT StatMan([SC0]) FROM (SELECT ...	20	59	2	tempdb	20801 - AQ	DESKTO...
SP:StmtCompleted		21	59	2	tempdb	20801 - AQ	DESKTO...
SQL:BatchCompleted	select _COL2, count(*) from #DP_TEM...	29	59	5	AdventureWorks2017		DESKTO...
SP:StmtCompleted	SELECT StatMan([SC0]) FROM (SELECT ...	18	59	2	tempdb	20801 - AQ	DESKTO...
SP:StmtCompleted		19	59	2	tempdb	20801 - AQ	DESKTO...
SQL:BatchCompleted	select _COL3, count(*) from #DP_TEM...	29	59	5	AdventureWorks2017		DESKTO...
SP:StmtCompleted	SELECT StatMan([SC0]) FROM (SELECT ...	19	59	2	tempdb	20801 - AQ	DESKTO...

```
select LEN([PersonType]) as _COL0, LEN([Title]) as _COL1, LEN([FirstName]) as _COL2, LEN([MiddleName]) as _COL3, LEN([LastName]) as _COL4, LEN([Suffix]) as
_COL5 into #DP_TEMP_2019_11_06_00_53_56_1212b91dca2a4e938d4083d9537993cb from [AdventureWorks2017].[Person].[Person]
```

Trace is stopped.

The Data Profiling task in SSIS is an important task that can be used to assess the quality of data sources. Unfortunately, this component is not widely used by many business intelligence developers.

In this article, we will give a brief overview of data profiling and the Data Profiling task in SSIS. Furthermore, we will mention some of its limitations and alternatives.

Why is Data Profiling important?

Before using any data source, the best practice is to assess its data quality and determine whether the data source is usable in a specific context.

There are many factors for determining data quality, such as completeness, consistency, uniqueness, timeliness, etc. Some of these factors require aggregating the data with other sources or performing some complex operations. But, the first thing to do is to analyze the data itself (NULL values ratio, values lengths, and other measurements) since this doesn't require any reference data or external aggregations.

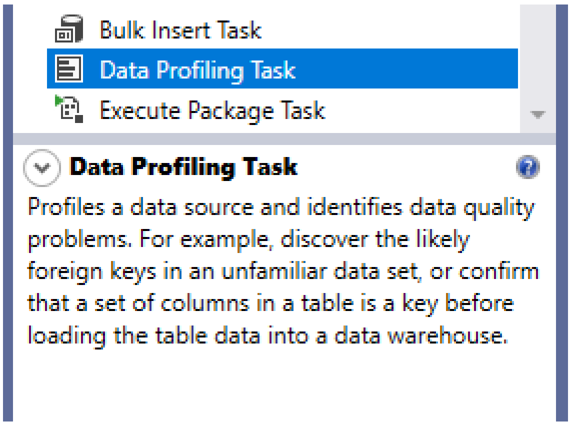
As an example, assume that you are building a reference database of world countries, and you are using an online data source. If the data source contains many missing values, it cannot be used in this context. So, it is very important to check the ratio between NULL and non-NULL values before deciding whether to use this data source.

Based on my own experience, data profiling can be used to build a knowledge base that stores quality information for each data source and specifies the contexts for the use of the data source.

This article is not intended to describe data quality and its importance, so refer to the following article for a brief introduction to data quality: An Introduction to Data Quality

Data Profiling Task in SSIS

Due to the importance of data profiling, the Data Profiling Task in SSIS was developed by Microsoft. As described in the documentation, this task is used to identify data quality problems:



Profiles a data source and identifies data quality problems. For example, discover the likely foreign keys in an unfamiliar data set, or confirm that a set of columns in a table is a key before loading the table data into a data warehouse.

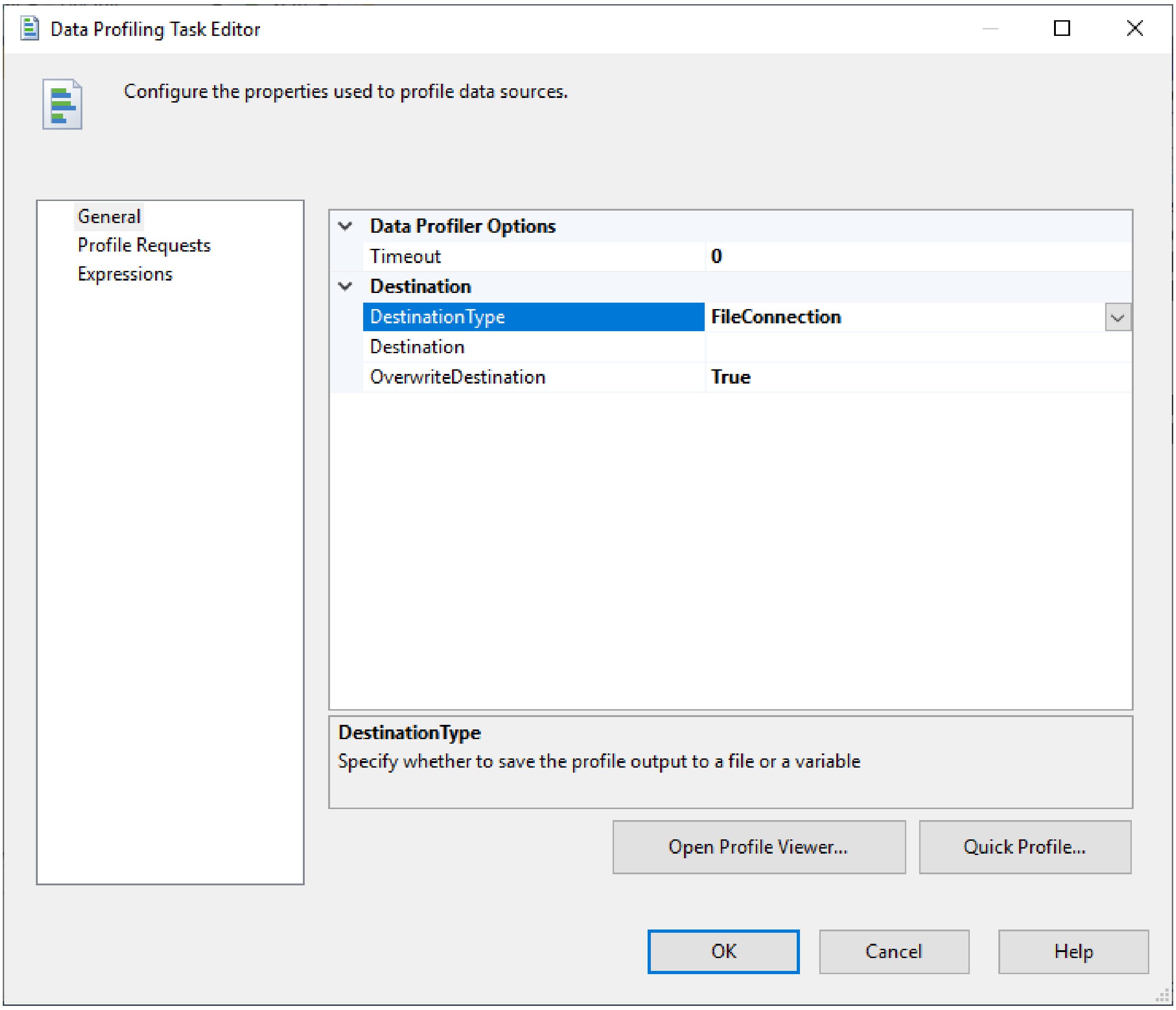
Figure 1 – SSIS Data Profiling task description from toolbox

In this section, we will describe the Data Profiling task in SSIS, profile viewer, quick data profiling and the profiles that can be created using this task.

When we open the task editor, it contains three-tab pages:

- General
- Profile Requests
- Expressions

In the General tab page, the user can specify the timeout period for each profile request, and choose to store the result in a variable (Record Set) or an external file using a File Connection since the file will be saved as XML:



Data Profiling Task Editor

Configure the properties used to profile data sources.

Data Profiler Options

Timeout: 0

Destination

DestinationType: FileConnection

Destination:

OverwriteDestination: True

DestinationType
Specify whether to save the profile output to a file or a variable

Figure 2 – SSIS Data Profiling task general tab page

Besides the general configuration, one important part of the General tab page is that you can open the Profile Viewer, which is a standalone tool that allows reading a previously exported XML file:

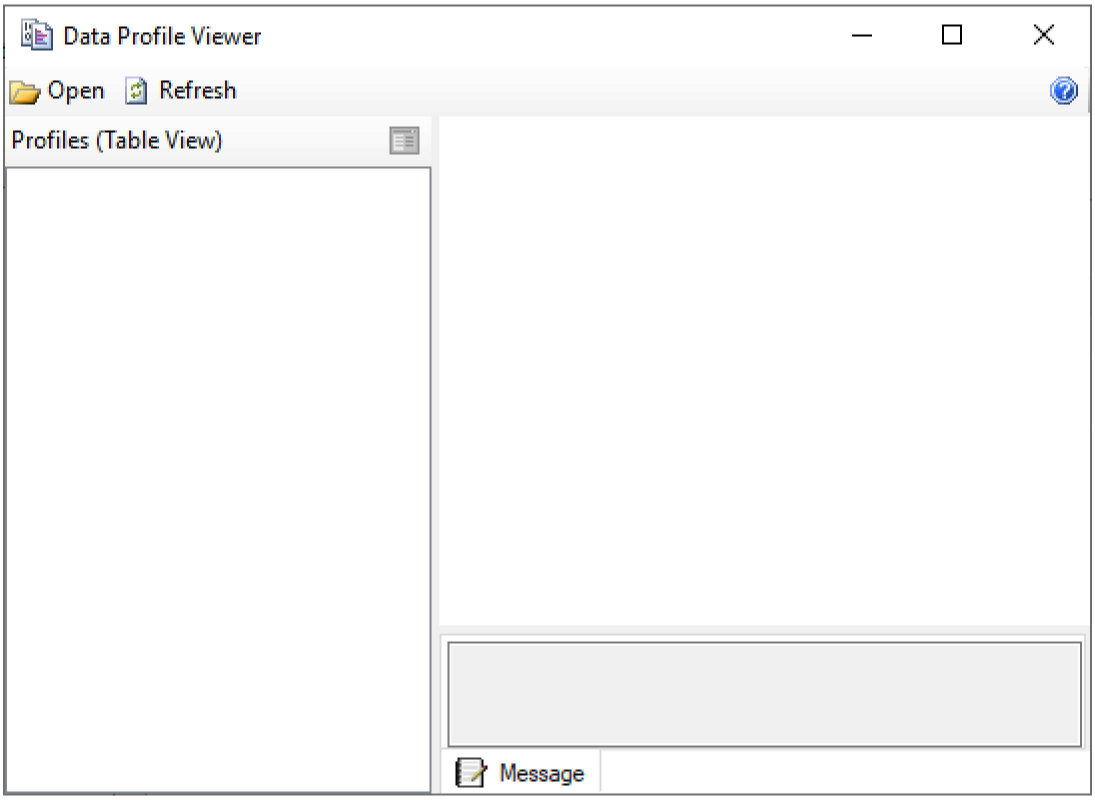
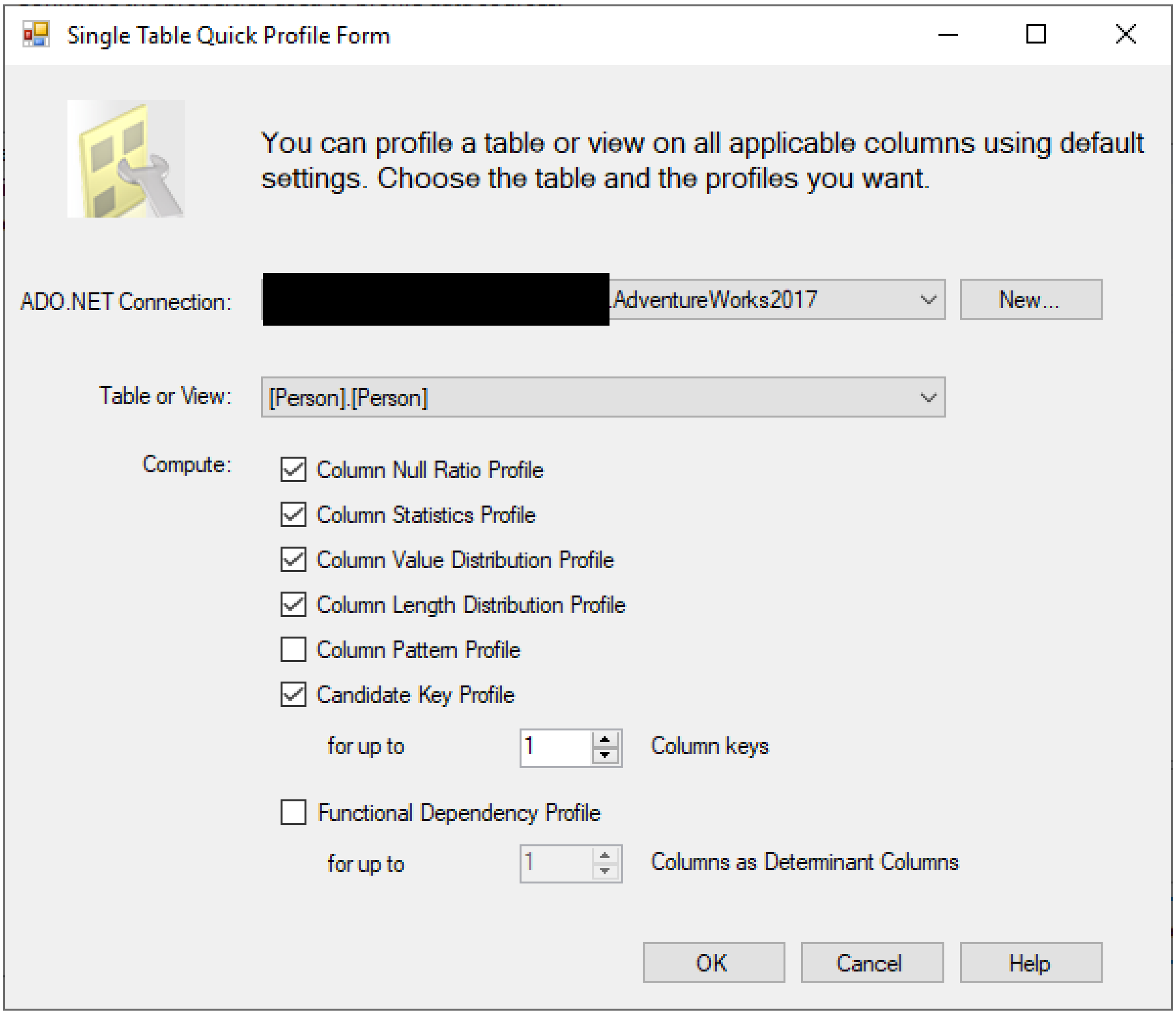


Figure 3 – Data Profile Viewer

To learn more about Data Profile Viewer, refer to the following documentation: Data Profile Viewer

In the Profile Requests tab page, you can add multiple profile requests for multiple connections (server, database, table), but you need to add them one-by-one. We'll describe this in more detail later. If you need to add multiple profile requests for a single connection/table, you can simply use the quick profile option from the General tab page as shown in the image below:



Single Table Quick Profile Form

You can profile a table or view on all applicable columns using default settings. Choose the table and the profiles you want.

ADO.NET Connection: [redacted].AdventureWorks2017

Table or View: [Person].[Person]

Compute:

☑ Column Null Ratio Profile

☑ Column Statistics Profile

☑ Column Value Distribution Profile

☑ Column Length Distribution Profile

☐ Column Pattern Profile

☑ Candidate Key Profile

for up to 1 Column keys

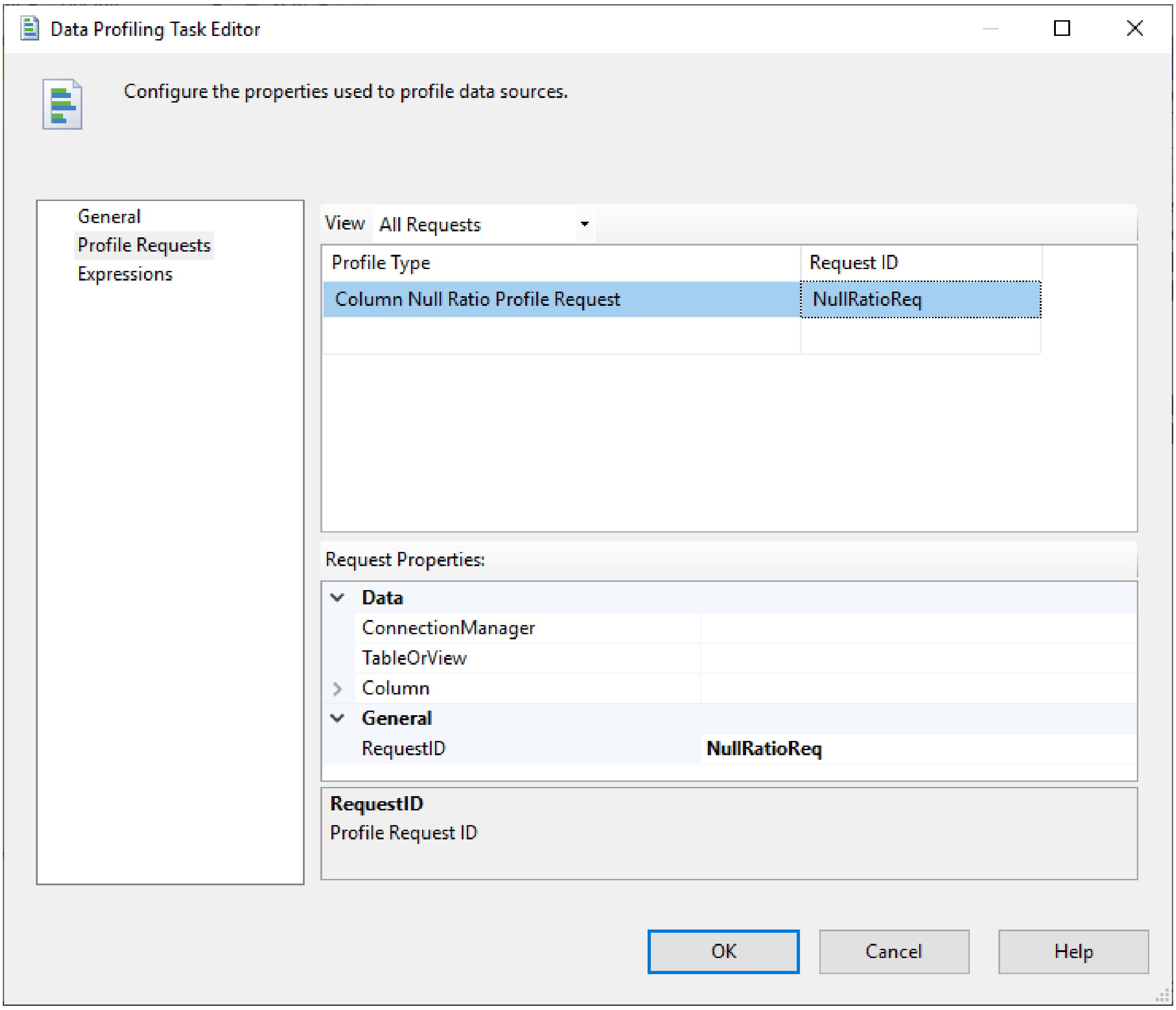
☐ Functional Dependency Profile

for up to 1 Columns as Determinant Columns

Figure 4 – Quick Profile dialog

For more information about the Quick Profile, refer to the following documentation: Single Table Quick Profile Form (Data Profiling Task)

The second tab page (Profile Requests) is where the user selects the profile requests needed. As shown in the image below, you can add a request from the grid located on the top of the form, while all configuration related to each profile request is found in the other grid:



Data Profiling Task Editor

Configure the properties used to profile data sources.

View: All Requests

Profile Type	Request ID
Column Null Ratio Profile Request	NullRatioReq

Request Properties:

RequestID: NullRatioReq

RequestID
Profile Request ID

Figure 5 – Profile Requests tab page

There are different types of profile requests. Refer to the official documentation for more information about each profile request type:

- Candidate Key Profile Request
- Column Length Distribution Profile Request
- Column Null Ratio Profile Request
- Column Pattern Profile Request
- Column Statistics Profile Request
- Column Value Distribution Profile Request
- Functional Dependency Profile Request
- Value Inclusion Profile Request

Each profile request has its own connection manager, table and column that must be specified in the configuration grid.

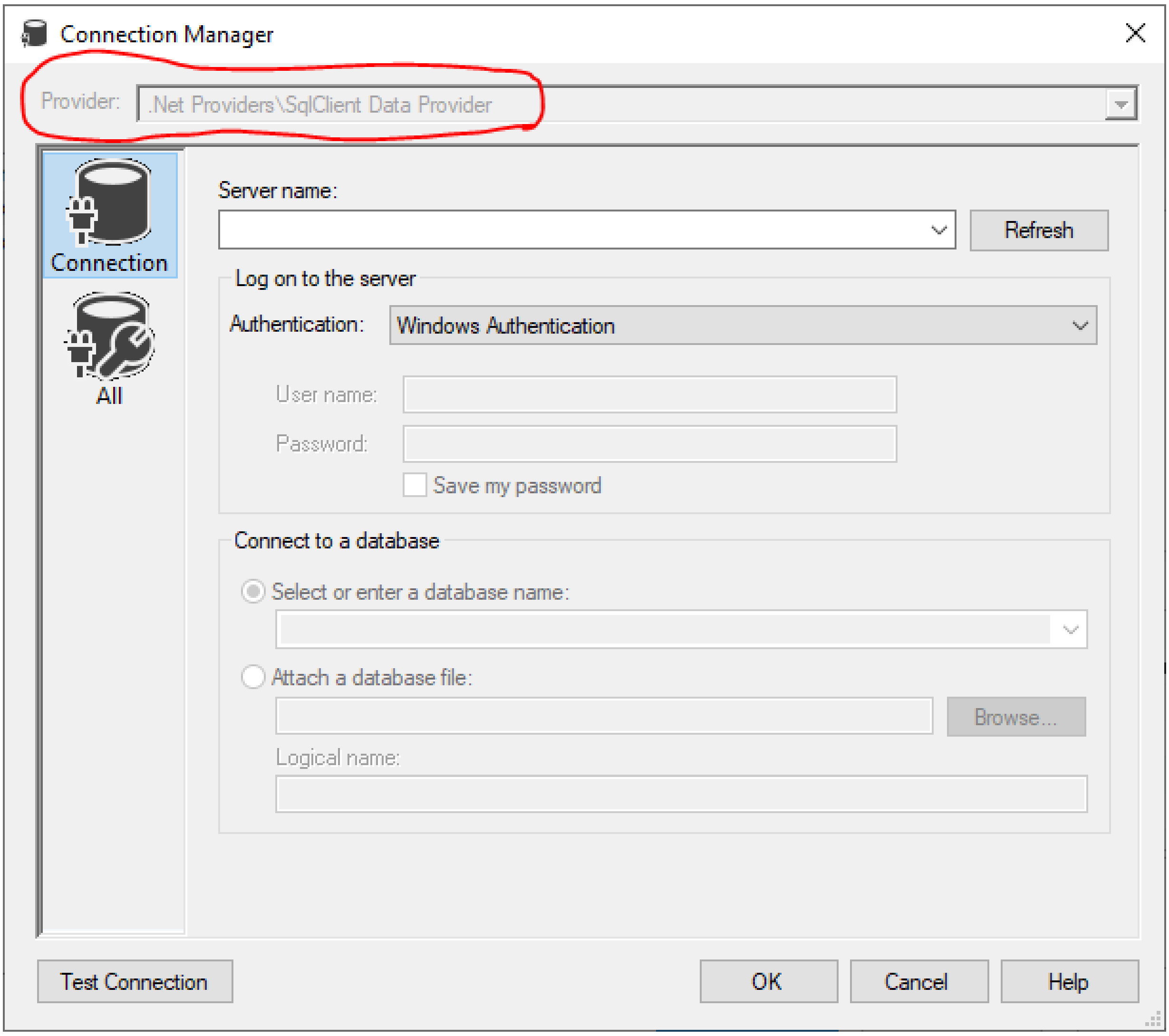
The last tab page, Expressions, is used to assign an expression to any property of the Data Profiling task in SSIS.

Limitations

There are three main limitations of the Data Profiling task in SSIS.

1. Only ADO.NET connections are allowed

The Data Profiling task in SSIS only accepts ADO.NET connection and SqlClient ADO provider, which only supports SQL Server databases. Which means you are not able to perform profile requests over other database providers such as Oracle, SQLite, MySQL, and others:



Connection Manager

Provider: .Net Providers\SqlClient Data Provider

Server name:

Log on to the server

Authentication: Windows Authentication

User name:

Password:

□ Save my password

Connect to a database

● Select or enter a database name:

○ Attach a database file:

Logical name:

Figure 6 – Only ADO.Net connection are allowed

2. No custom profile requests can be added

Even if there are many types of profile requests you can add using the Data Profiling task in SSIS, you may need to create a custom profile request based on a specific requirement, which is not allowed by this task.

3. Limited profiling destination types

While it is very important to store the result of the profile requests within a database to be used for quality assessment purposes, only the file connection manager or an object variable are available.

Alternatives

As mentioned above, no custom profile requests can be added using the Data Profiling task in SSIS, but you can write your own profile request using an Execute SQL Task and store the result in a result set. Check the following answer on Stack overflow as an example: Data profiling Task – custom Profile Request.

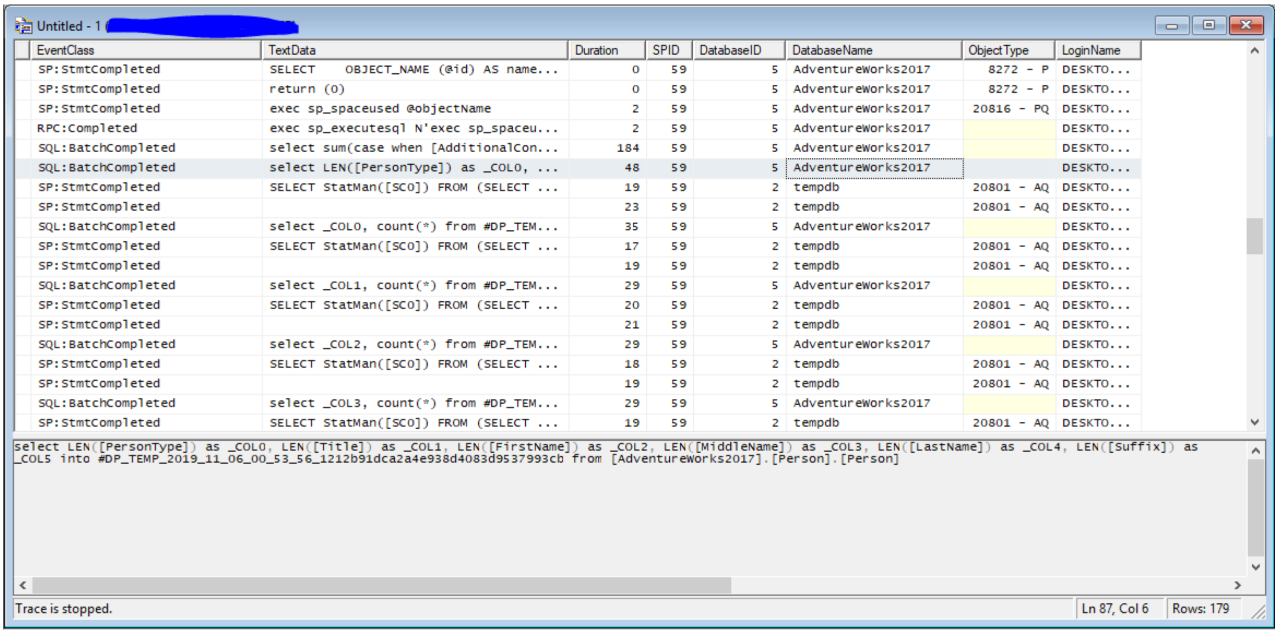
In addition, Execute SQL Task supports a wider range of connection types and can be used to perform SQL commands over different database providers such as Microsoft Access, Excel, Oracle, SQLite, MySQL and others.

How does it work?

In this section, we will illustrate how profile requests are sent to the database engine. To do that, we will use the SQL Profiler to check which commands are sent to the SQL Server instance.

To run this experiment, we created an SSIS package, and we added a data profiling task where we added 5 profile requests using the quick profile option. Then, we executed the package after creating a trace using SQL Profiler.

As shown in the image below, a batch of SQL commands are sent to the database engine in order to perform data profiling. Each profile request is not performed using only one SQL command; rather, multiple commands are executed to obtain each profile request. In addition, we can see that the target database and Tempdb are used:


Untitled - 1 ([redacted]

EventClass	TextData	Duration	SPID	DatabaseID	DatabaseName	ObjectType	LoginName
SP:StmtCompleted	SELECT OBJECT_NAME (@id) AS name...	0	59	5	AdventureWorks2017	8272 - P	DESKTO...
SP:StmtCompleted	return (0)	0	59	5	AdventureWorks2017	8272 - P	DESKTO...
SP:StmtCompleted	exec sp_spaceused @objectName	2	59	5	AdventureWorks2017	20816 - PQ	DESKTO...
RPC:Completed	exec sp_executesql N'exec sp_spaceu...	2	59	5	AdventureWorks2017		DESKTO...
SQL:BatchCompleted	select sum(case when [AdditionalCon...	184	59	5	AdventureWorks2017		DESKTO...
SQL:BatchCompleted	select LEN([PersonType]) as _COL0, ...	48	59	5	AdventureWorks2017		DESKTO...
SP:StmtCompleted	SELECT StatMan([SC0]) FROM (SELECT ...	19	59	2	tempdb	20801 - AQ	DESKTO...
SP:StmtCompleted		23	59	2	tempdb	20801 - AQ	DESKTO...
SQL:BatchCompleted	select _COL0, count(*) from #DP_TEM...	35	59	5	AdventureWorks2017		DESKTO...
SP:StmtCompleted	SELECT StatMan([SC0]) FROM (SELECT ...	17	59	2	tempdb	20801 - AQ	DESKTO...
SP:StmtCompleted		19	59	2	tempdb	20801 - AQ	DESKTO...
SQL:BatchCompleted	select _COL1, count(*) from #DP_TEM...	29	59	5	AdventureWorks2017		DESKTO...
SP:StmtCompleted	SELECT StatMan([SC0]) FROM (SELECT ...	20	59	2	tempdb	20801 - AQ	DESKTO...
SP:StmtCompleted		21	59	2	tempdb	20801 - AQ	DESKTO...
SQL:BatchCompleted	select _COL2, count(*) from #DP_TEM...	29	59	5	AdventureWorks2017		DESKTO...
SP:StmtCompleted	SELECT StatMan([SC0]) FROM (SELECT ...	18	59	2	tempdb	20801 - AQ	DESKTO...
SP:StmtCompleted		19	59	2	tempdb	20801 - AQ	DESKTO...
SQL:BatchCompleted	select _COL3, count(*) from #DP_TEM...	29	59	5	AdventureWorks2017		DESKTO...
SP:StmtCompleted	SELECT StatMan([SC0]) FROM (SELECT ...	19	59	2	tempdb	20801 - AQ	DESKTO...

```
select LEN([PersonType]) as _COL0, LEN([Title]) as _COL1, LEN([FirstName]) as _COL2, LEN([MiddleName]) as _COL3, LEN([LastName]) as _COL4, LEN([Suffix]) as
_COL5 into #DP_TEMP_2019_11_06_00_53_56_1212b91dca2a4e938d4083d9537993cb from [AdventureWorks2017].[Person].[Person]
```

Trace is stopped. Ln 87, Col 6 Rows: 179

Figure 7 – SQL Profiler screenshot

Conclusion

In this article, we gave an overview of the Data Profiling task in SSIS. We answered questions like what is the Data Profiling task in SSIS, why is data profiling important, how to use the Data Profiling task, what are the main limitations and alternatives, and how does this task perform profile requests over SQL Server database engine.

Additional information about profile requests and the use cases for each one are described in the official documentation.

See more

Check out SpotLight, a free, cloud based SQL Server monitoring tool to easily detect, monitor, and solve SQL Server performance problems on any device

Watch Video At: https://youtu.be/zchwCUSvJPs

Hadi Fadlallah

Hadi is a Lebanese Researcher, Data Engineer and Business Intelligence Developer.

He has been working with SQL Server for more than 10 years. Also, he's one of the top ETL and SQL Server Integration Services contributors at Stackoverflow.com

Hadi really enjoys learning new things everyday and sharing his knowledge.

View all posts by Hadi Fadlallah