**Netezza DDL Converter Utility**

The Netezza DDL converter utility is an application that can convert Netezza database schema DDL to SQL equivalent. This utility was written by various members of the Data Migration Jumpstart team and migrated some early work by the Azure SQL DW Customer Success Engineering team from a PowerShell script to a C# utility. Subsequently, one of the Program Managers on the Azure SQL DW team developed an ATLR based migration utility (targeting only Azure SQL DW), so the DMJ stopped all development on this utility.

Recently, DMJ had a customer wanting to migrate 1600 stored procs to Azure SQL DW, but the new ATLR based tool does not convert stored procs, so DMJ cracked open the code to attempt to add this functionality. The DMJ developer tackling this challenge has had some regrets regarding the offer to add this functionality, given the amount of work required, but the good news is that the holiday season provided some spare time to persevere and in the end the new code provides something like a 95% conversion, with only minor manual fixes required.

The key features of this utility are that it can;

* Convert Netezza MPP DDL to SQL Server MPP (Azure SQL DW) or SMP (Azure SQL DB/MI, SQL on IaaS or on premises SQL Server) equivalents
* Take you a long way toward converting stored procedures to SQL Server platforms.

The application is a Windows console-based utility and currently supports conversion of table schema, views, stored procedures, synonyms, defaults and constraints. For Azure SQL DW targets, the application with keep table distributions (moving them as appropriate) and you have the option of turning on External Table Creation.

**Prerequisites**

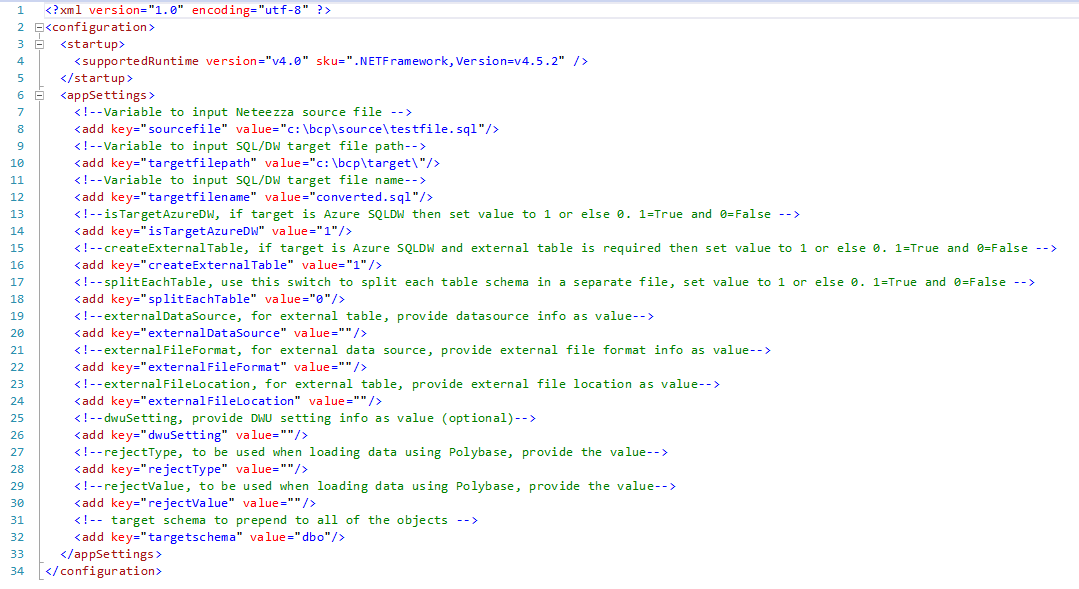
Following are the prerequisites to run this utility:

* Netezza DDL is extracted using the Netezza Tools like nzsql or Agility Workbench. DDL is extracted into a single file. This should just be the Create Tables with defaults and constraints. Views, Stored Procedures and other Netezza objects are considered out of scope for hit tool at this time.
* Azure SQL DW is configured for Polybase
* Data Source for Polybase has been created
* File Format for Polybase has been created

**Components of Utility**

This utility has two components

1. **Netezza Converter.exe.config** – This is a XML config file which stores all configuration related to conversion. One must edit this file to update the desired conversion configurations.



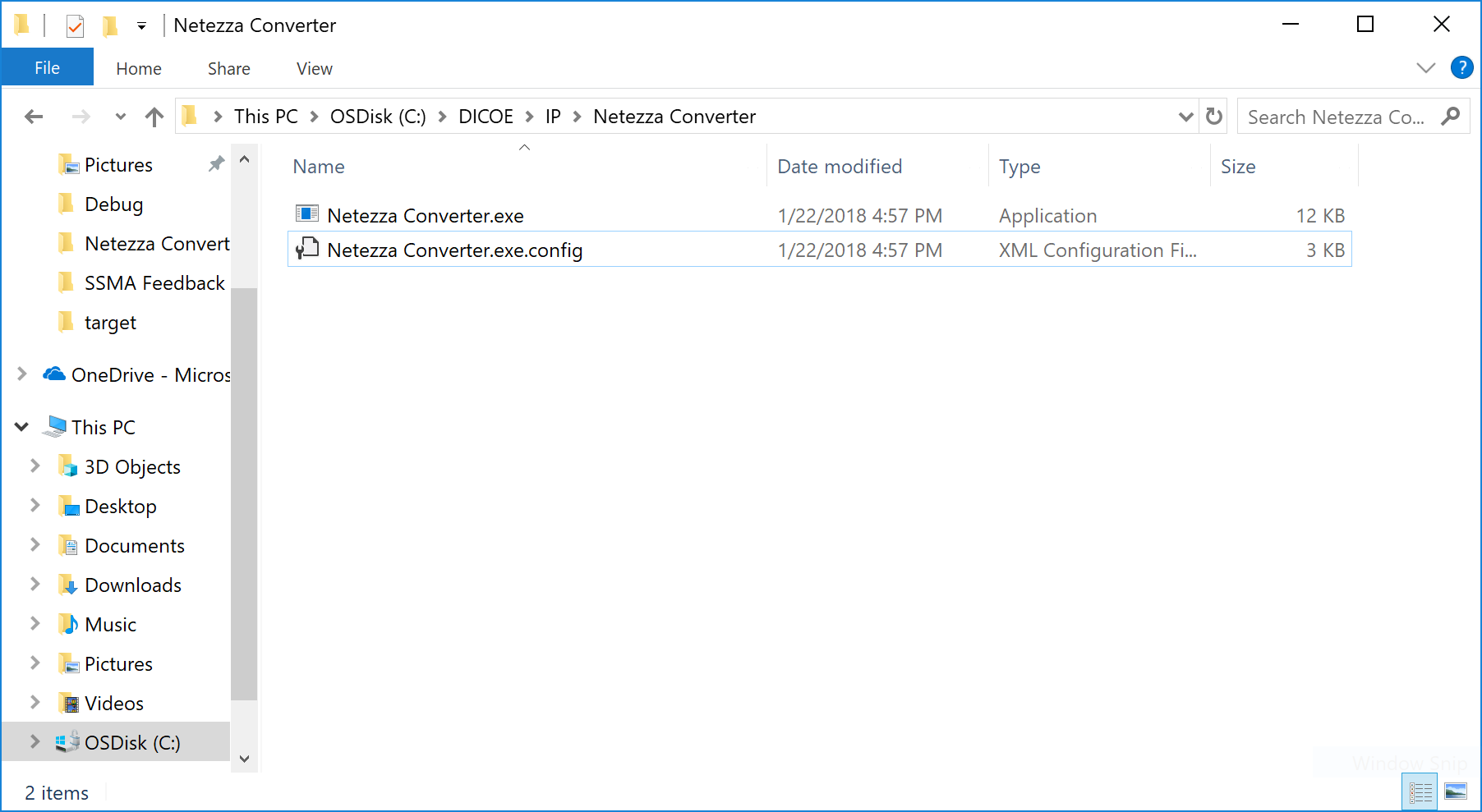
Here are the details to configurations settings to update before running the utility.

* + **sourcefile** – the full path and name of the Netezza DDL source file(s). A recent change was to allow wildcards so that multiple files in a folder can be processed.
  + **targetfilepath** – the file path for converted DDL
  + **targetfilename** – the name of converted DDL file “filename.sql”
  + **isTargetAzureDW** – set the value to 1 if conversion target is Azure SQL DW, otherwise SQL Server on premise/IaaS, Azure SQL DB or MI is assumed.
  + **createExternalTable** – Update the value to 1 if target is Azure SQL DW and you also want to create external tables
  + **splitEachTable** – Update the value to 1 if converted DDL should create one DDL file for each table. In case if Azure SQL DW with external table switch is on, this setting generates a single file including external table
  + **externalDataSource** – Update the name of Polybase data source created in Azure SQL DW
  + **externalFileFormat** – Update the name of Polybase file format created in Azure SQL DW
  + **externalFileLocation** – Update the location of WASB where data files are stored. This is the directory structure. It is assumed that each table will have its own location based on table name
  + **rejectType** -Update rejectType to use when loading data using Polybase
  + **rejectValue** -Update rejectValue to use when loading data using Polybase
  + targetschema – Update to set the schema name to use for target objects.

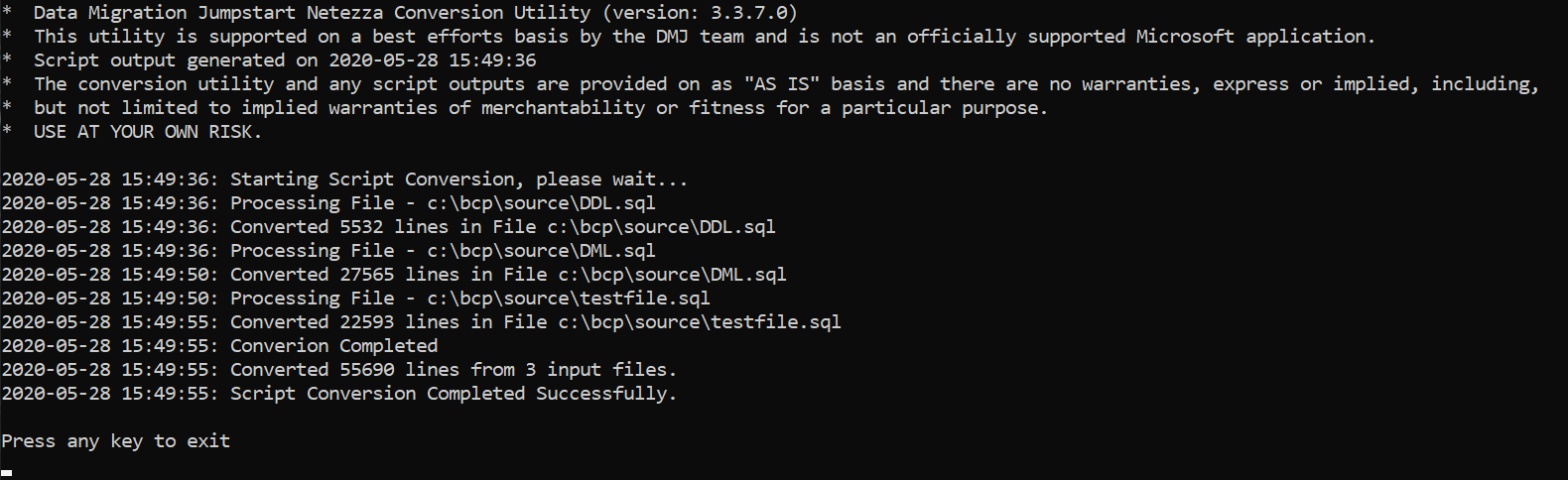
1. **NetezzaConverter.exe** – This is an executable file that does not require any parameters to run, so it can be run by double clicking the executable. The application pauses to read a line (press enter to continue) before exiting, so you have a chance to see the application output.

**Sample Run**

1. Download and switch to the folder that has the Netezza Converter exe and XML config file



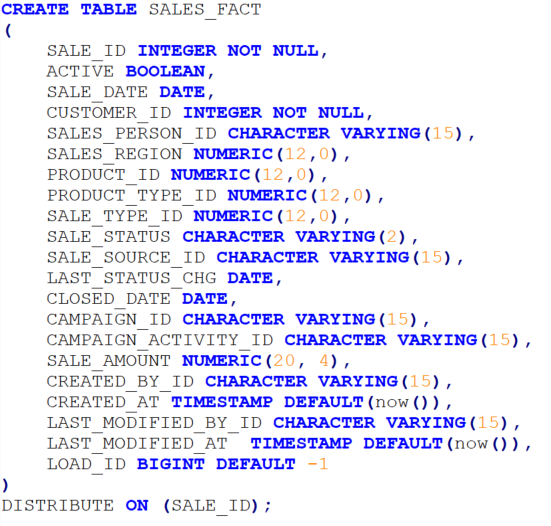
1. Edit and update the XML file with desired configuration, in the example configuration we are converting a Netezza DDL to Azure SQL DW.
2. Run the application exe.



1. Review the output in the configured output folder.

**Sample Conversions**

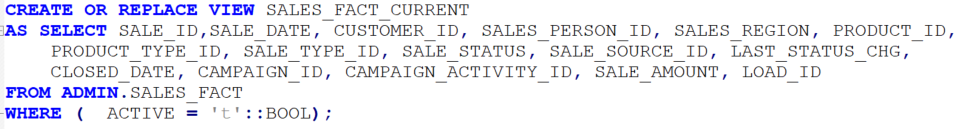
Table DDL



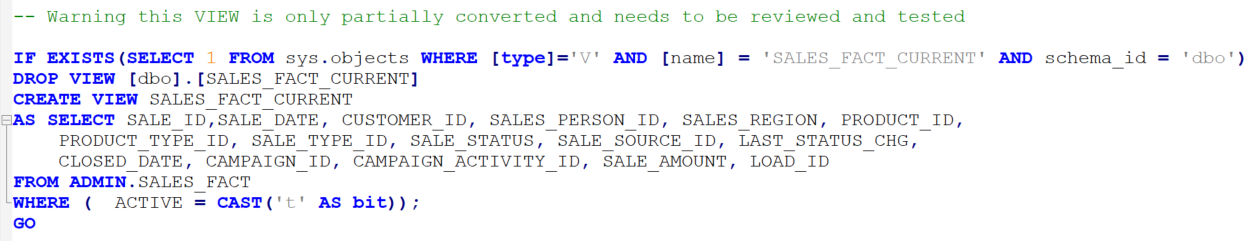
Converted Table DDL



View DDL



Converted View DDL

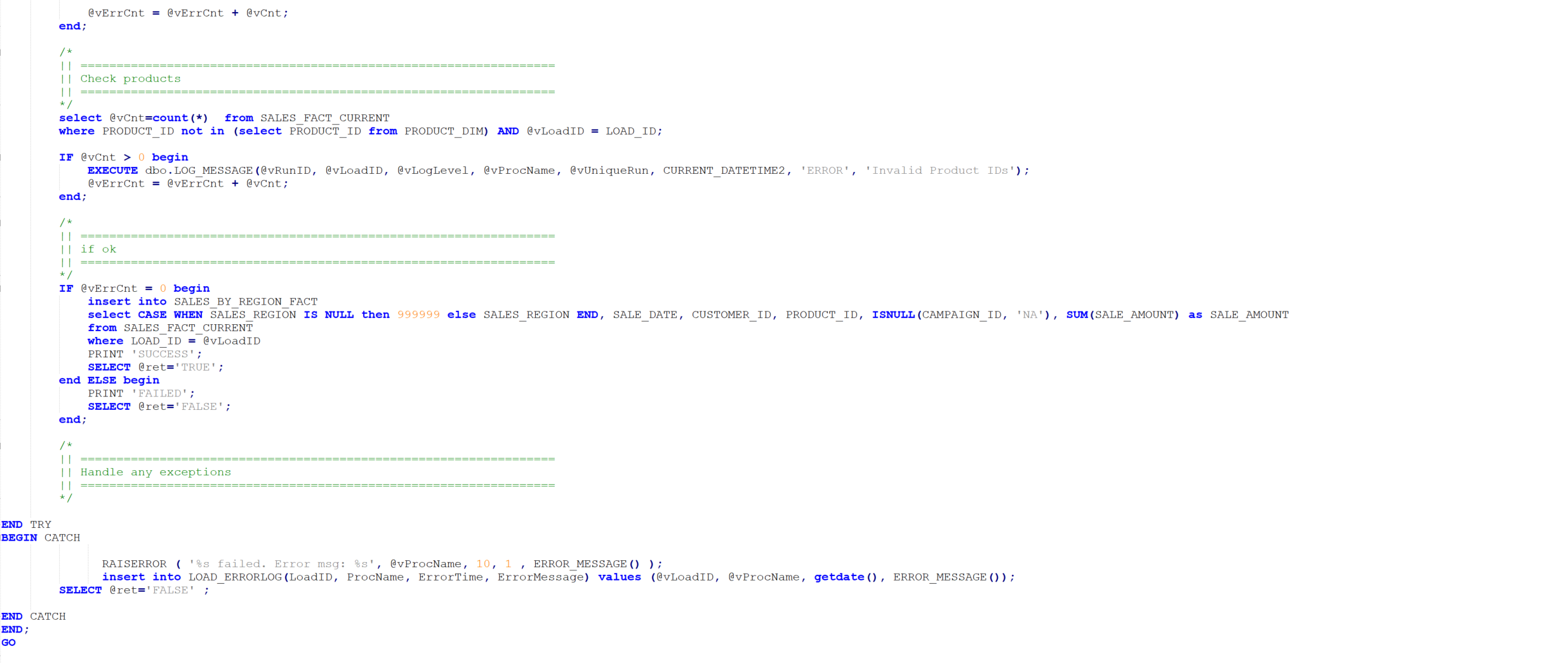


Stored Proc Sample



Converted Stored Proc





Similarly, we can use this conversion utility to generate target schema based on different settings, for an example –

* Converting Schema to Azure SQL DB
* Split tables in separate DDL files
* Etc.

**Known Issues**

There are a few issues that we don’t have work arounds for or haven’t gotten to handling yet.

* Create Table As Select (CTAS) – this is support on Synaps but these statements don’t work on SQL Server, they need to be refactored to select … into
* Statements like Create Table that have a WITH clause will not work. They can likely be easily refactored to WITH () SELECT … INTO …
* MIN/MAX/GREATEST/LEAST type functions with multiple parameters do not easily translate to TSQL – there are some work arounds, but it isn’t an easy substitution.

For any questions or comments, please reach out to Mitch.van.Huuksloot @ Microsoft.com and/or Kuldeep.chauhan @ Microsoft.com