*Table Dependency Diagram Generator*

|  |  |
| --- | --- |
| Bert O’Neill | |
| Department: | IT\Software Support |
| Document Title: | Auto Table Dependency Diagram - User Guide |
| Author: | Bert O’Neill |
| Date: | 01-Jan-2024 |
| Version | 1.1 |
| Email | BertONeill@yahoo.com |
| Document Goals | Provide an overview for developers, on how to quickly create an Entity Relationship Diagram for a table an it’s immediately related tables. |

# Revision History

|  |  |  |  |
| --- | --- | --- | --- |
| Date | Version | Description | Author |
| 18-Dec-2024 | 1.0 | Initial draft | Bert O’Neill |
| 01-Jan-2024 | 1.1 | Adding extended diagram (data types) | Bert O’Neill |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

Contents

[Revision History 1](#_Toc154596730)

[Introduction 3](#_Toc154596731)

[Purpose 3](#_Toc154596732)

[Scope 3](#_Toc154596733)

[Prerequisites 3](#_Toc154596734)

[SSMS Extension Installation 4](#_Toc154596735)

[Copy VSIX File to SSMS Extension Folder 4](#_Toc154596736)

[Windows Folder Permission 4](#_Toc154596737)

[Creating a Diagram 5](#_Toc154596738)

[Select Database 6](#_Toc154596739)

[Select Table 6](#_Toc154596740)

[Compact View 7](#_Toc154596741)

[Example of a Compact Layout Diagram 7](#_Toc154596742)

[Extended View 8](#_Toc154596743)

[Example of an Extended Layout Diagram (showing data-types) 8](#_Toc154596744)

[Rearrange Table Nodes 8](#_Toc154596745)

[Exporting a Diagram 9](#_Toc154596746)

[Printing a Diagram 9](#_Toc154596747)

# Introduction

This document provides an overview on how to use the SSMS (SQL Server Management Studio) extension to quickly generate a subset of tables that are related to a particular table, in a visual diagram format.

## Purpose

You may find that when you start a new project, the database schema can be overwhelming, due to the volume of tables, and you are currently working on a subset of tables. But you want to encapsulate only that subset of tables into a diagram, so that you can understand their relationship without all the other tables.

## Scope

The scope of this document is to convey the steps needed to start auto generating subset table diagrams from within SSMS.

# Prerequisites

* An understanding of SQL table relationships (1:1, 1:M, M:M. Linked List)
* An understand of SSMS
* You have the permissions to install this extension into the SQL Server Extensions folder
* You have the following SQL permissions granted:

1. Grant *VIEW ANY DEFINITION* to YourUserName;
2. Grant *CREATE TABLE* to YourUserName;
3. *SELECT and VIEW* permissions on these tables:
   1. Sys.Foreign\_Keys
   2. Sys.Tables
   3. Sys.Foreign\_Key\_Columns
   4. Sys.Columns
   5. Information\_Schema.Tables
   6. Information\_Schema.Columns

Simply perform a simple query against these tables and you will know if you need to be granted extra permissions or not, from your DBA.

* Select \* From Information\_Schema.Tables
* Select \* From Sys.Tables

# SSMS Extension Installation

There are a number of ways to install an Extension that will not be hosted or used from within Visual Studio, but rather than editing the *Regedit* database, the easiest way is to have the appropriate folder permissions and simply copy and paste the extension to the correct folder location.

## Copy VSIX File to SSMS Extension Folder

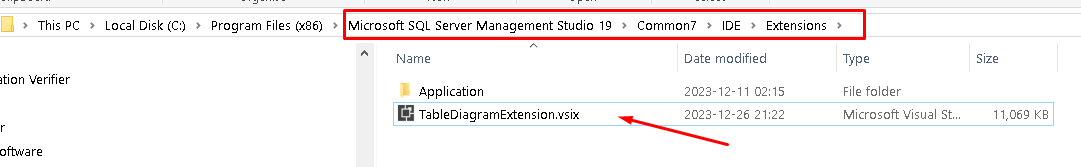
Before coping the VSIX file into your environment, close and save any work you have in SSMS.

At this stage, you will have downloaded the VSIX file to your environment:



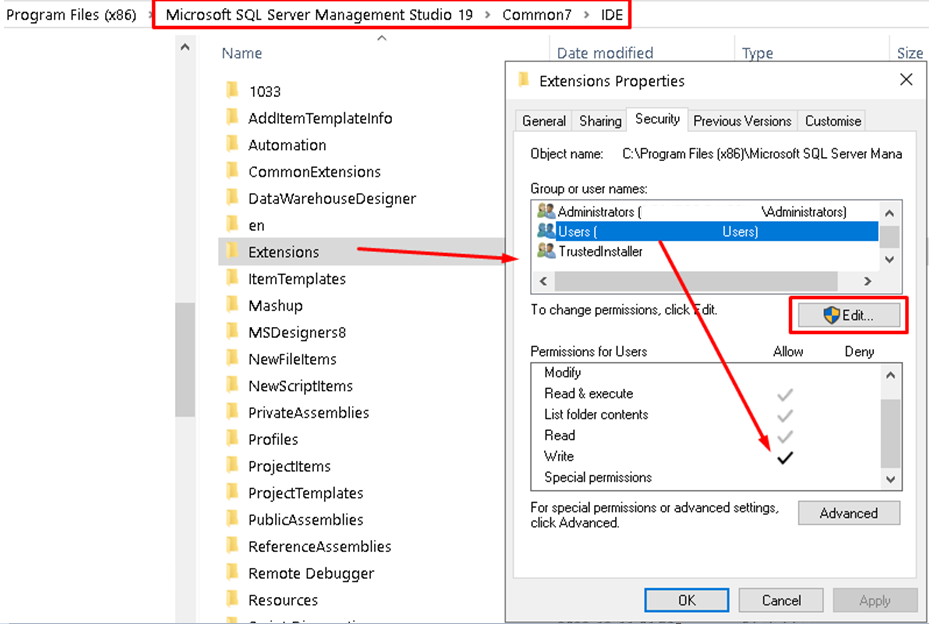
Copy the VSIX extension into your SSMS 18+ Extensions folder:

For e.g. C:\Program Files (x86)\Microsoft SQL Server Management Studio 19\Common7\IDE\**Extensions\**

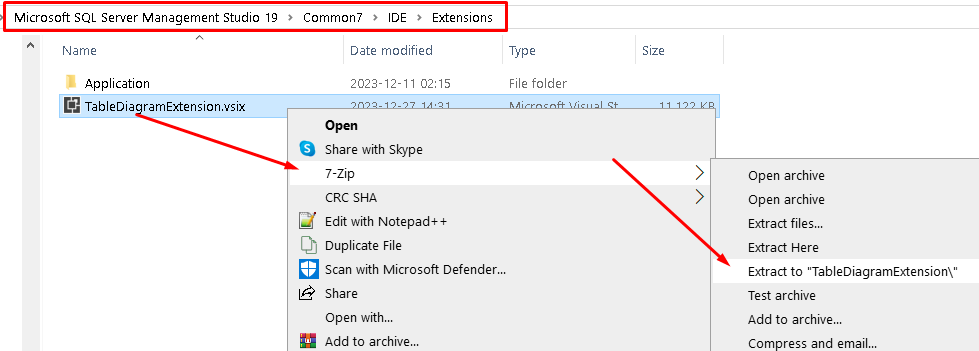


## Windows Folder Permission

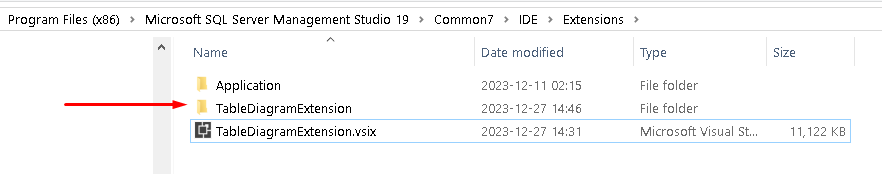
To use the extension within SSMS, you will need to unzip the VSIX file, but to do this you will need Write permissions on the Extension folder:



Unzip the VSIX file to the default folder TableDiagramExtension within the Extensions folder:



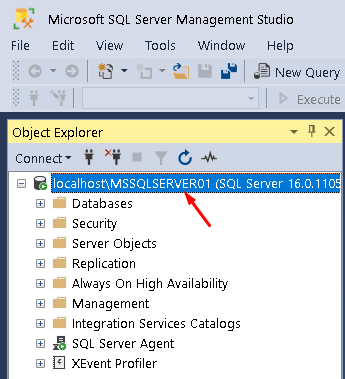
This will create the new folder, like below:



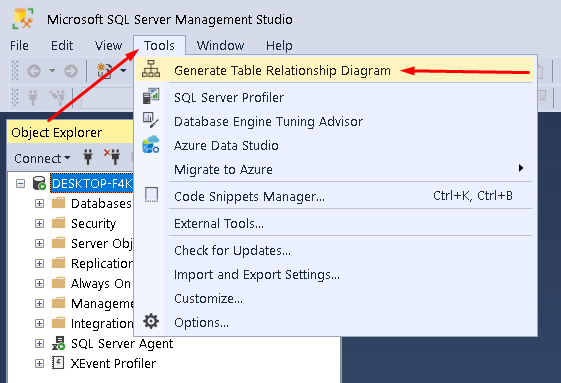
**NB:** Start SSMS to pick up the new extension.

# Creating a Diagram

Connect to the SQL Server Instance you wish to generate diagrams against. The SQL Server Instance should be selected by default, in the SQL Server Management Objects (SMO) tab.



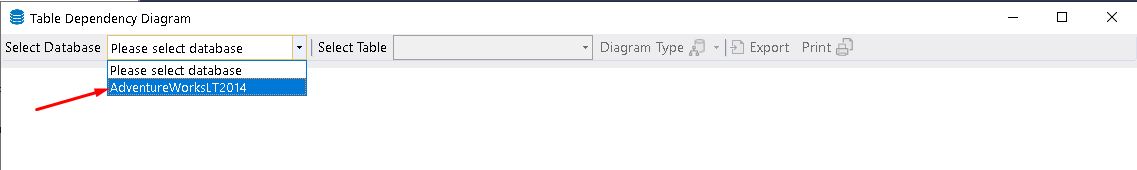
Then click on the menu option Tools →Generate Table Relationship Diagram, to bring up the diagram dialogue.



From here you can select your SQL database, which will then populate the Tables dropdown based on your database selection.

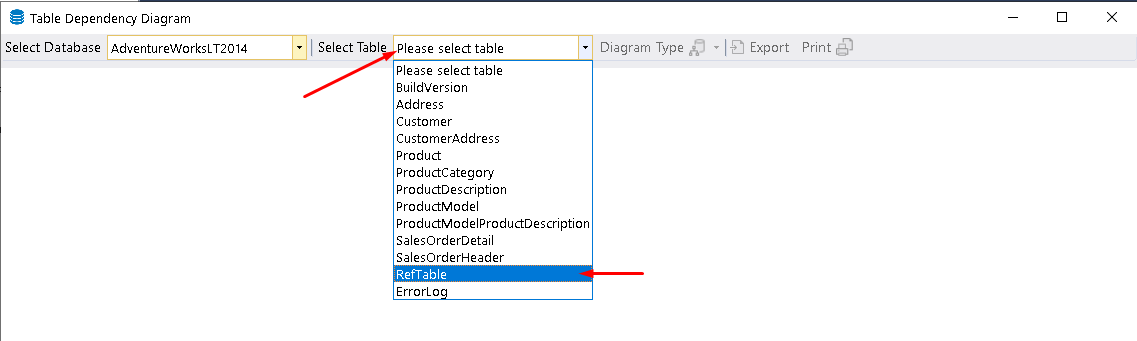
## Select Database

All the non-system related databases will be displayed.



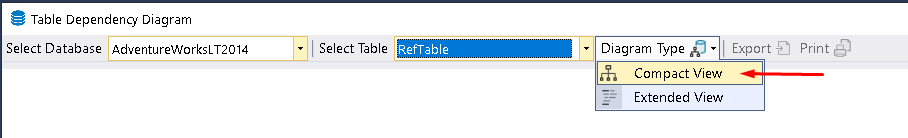
## Select Table

All the non-system related tables will be displayed.

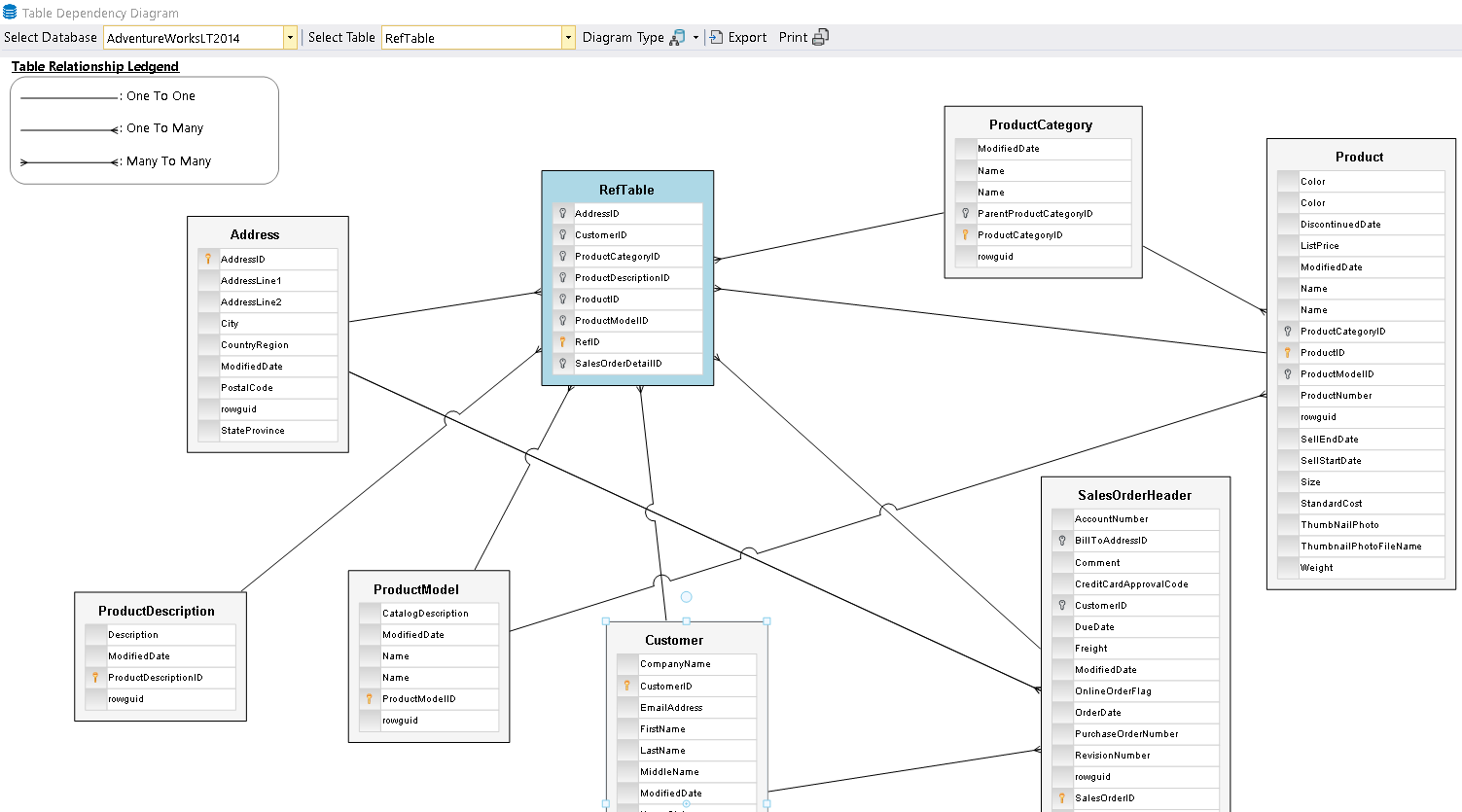


## Compact View

Once you have selected a table to base your diagram for, the button Diagram Type will be enabled. From here you can select to generate a simple (compact) layout or an extended layout (with data-types).

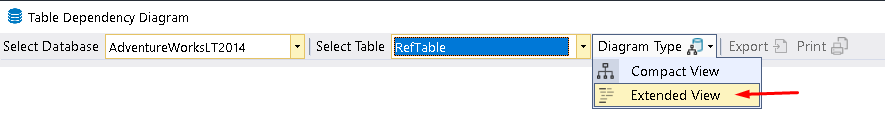


## Example of a Compact Layout Diagram

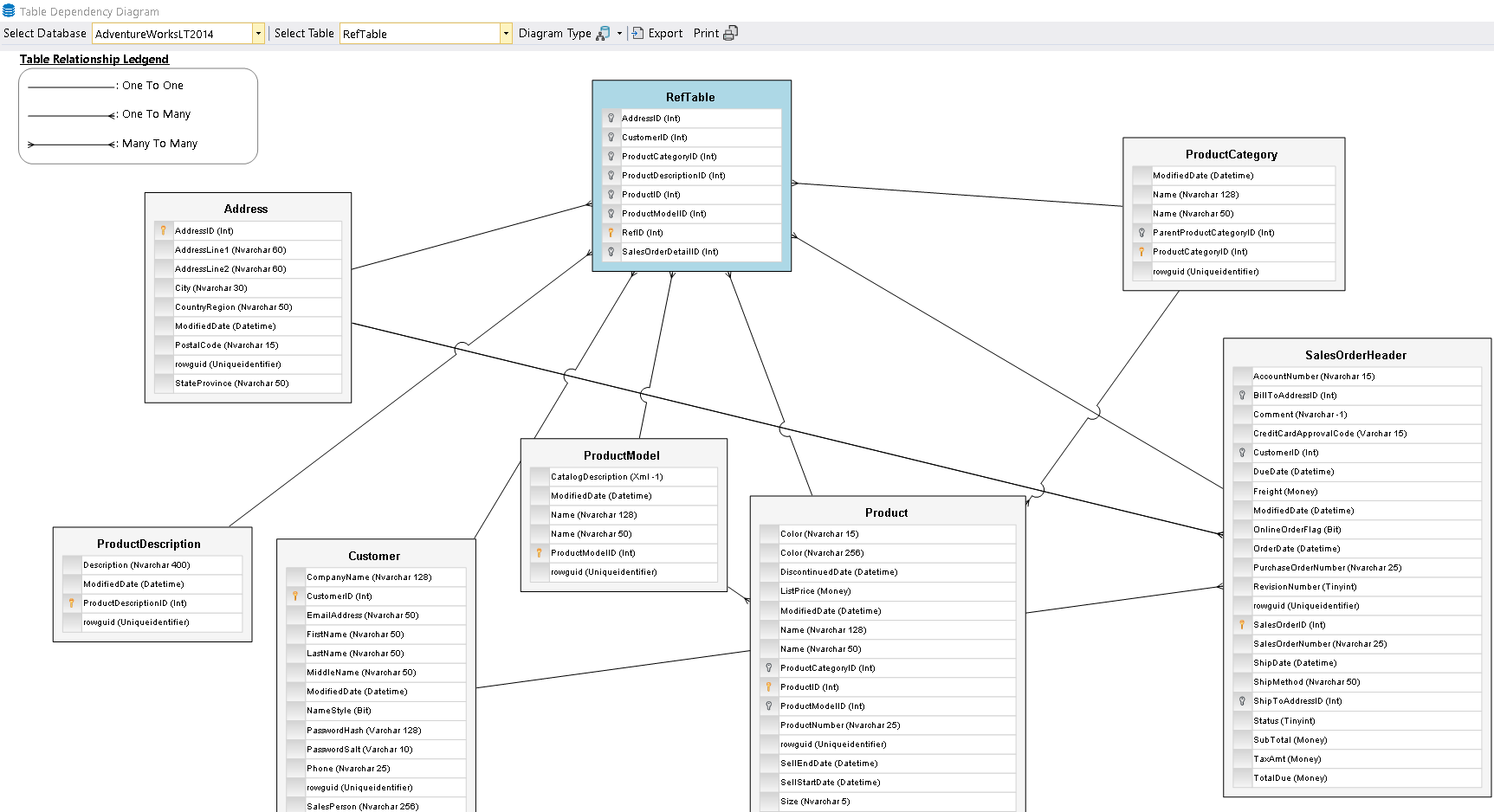


## Extended View

The user can select the extended layout from the split button.



## Example of an Extended Layout Diagram (showing data-types)

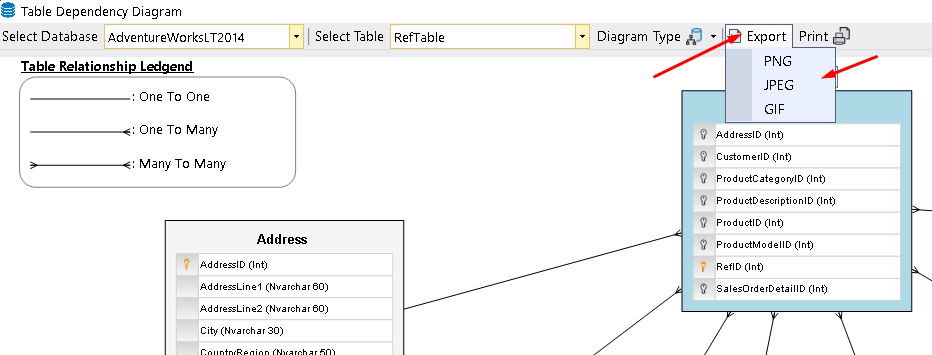


## Rearrange Table Nodes

You are able to move\drag able nodes around the diagram, to give the diagram a greater visually aesthetic lookup. Simply click on a table and hold your right mouse button down, and move using your table, the connection lines will follow your table.

# Exporting a Diagram

If you wish to save your diagram with the team or keep it for future references, simply click on the Export split button and select the format you wish to save the diagram in, you will be prompted for a location to save your file (with the extension .edd).



# Printing a Diagram

If you wish to have a physical copy of your diagram, click on the print button and a Print review dialogue will be displayed, from here you can the print as normal.

