*SQL Server Table Dependency Diagram Generator*

|  |  |
| --- | --- |
| Bert O’Neill | |
| Department: | IT\Software Support |
| Document Title: | Generate 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](#_Toc154936573)

[Introduction 3](#_Toc154936574)

[Purpose 3](#_Toc154936575)

[Scope 3](#_Toc154936576)

[Prerequisites 3](#_Toc154936577)

[Visual Studio Extension Installation 4](#_Toc154936578)

[Integrate Through Managed Extensions 4](#_Toc154936579)

[Manual Install 4](#_Toc154936580)

[Creating a Diagram 5](#_Toc154936581)

[Connect to a SQL Server Instance 5](#_Toc154936582)

[Select Database 5](#_Toc154936583)

[Select Table 5](#_Toc154936584)

[Compact View 6](#_Toc154936585)

[Example of a Compact Layout Diagram 6](#_Toc154936586)

[Extended View 7](#_Toc154936587)

[Example of an Extended Layout Diagram (showing data-types) 7](#_Toc154936588)

[Rearranging Tables 8](#_Toc154936589)

[Exporting a Diagram 9](#_Toc154936590)

[Printing a Diagram 9](#_Toc154936591)

# Introduction

This document provides an overview on how to use the *Auto Generate Table Dependency Diagram* (Visual Studio) extension to quickly generate an Entity Relationship Diagram from a starting table, and bring together all related tables into one diagram.

## 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. You could create your own diagram, but the naming convention can be misleading and you don’t include all the related table!

## Scope

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

# Prerequisites

* An understanding of SQL table relationships (1:1, 1:M, M:M. Linked List)
* 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.

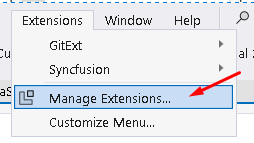
* Select \* From Information\_Schema.Tables
* Select \* From Sys.Tables

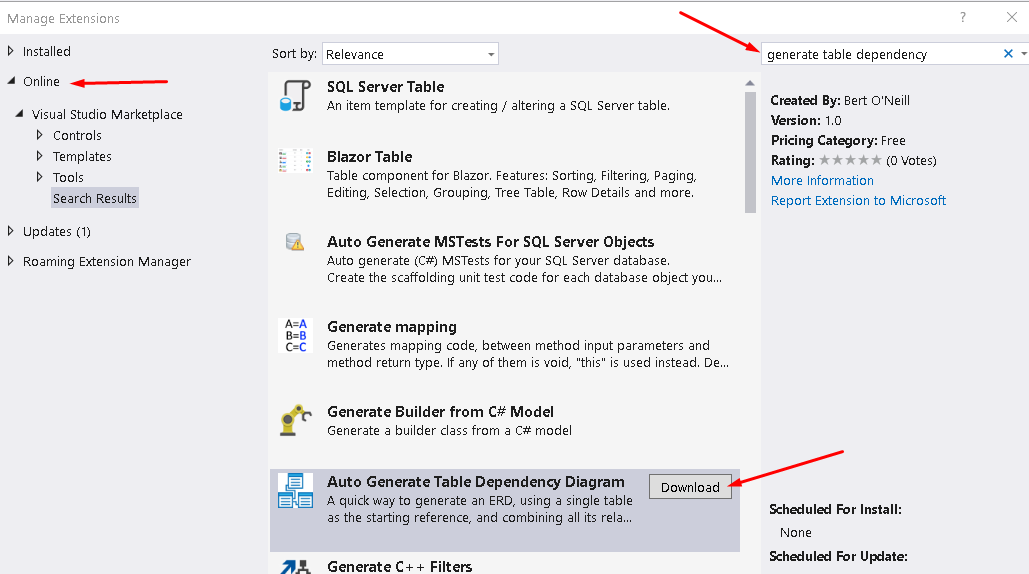
# Visual Studio Extension Installation

There are a couple of ways to integrate the extension into Visual Studio:

## Integrate Through Managed Extensions

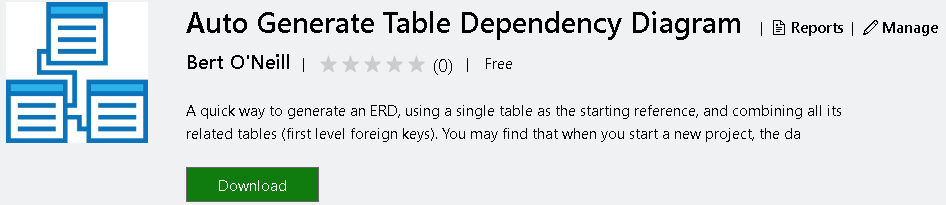
Use Visual Studio’s *Manage Extensions* option, and search for the extension there using the keywords “*generate table dependency*” and select to “*Download*” – restart Visual Studio.



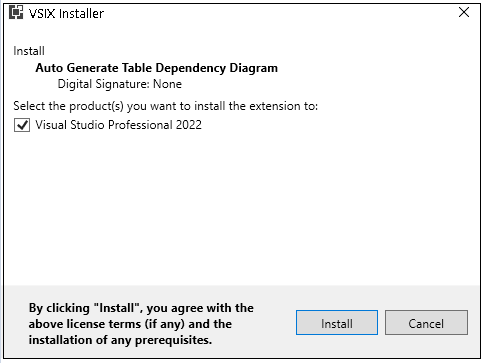


## Manual Install

Navigate to [Microsoft’s Marketplace](https://marketplace.visualstudio.com/manage/publishers/BertONeill?tracking_data=eyJTb3VyY2UiOiJFbWFpbCIsIlR5cGUiOiJOb3RpZmljYXRpb24iLCJTSUQiOiJtcy5HYWxsZXJ5Tm90aWZpY2F0aW9ucy52ZXJzaW9uLXZhbGlkYXRpb24tc3Vic2NyaXB0aW9uIiwiU1R5cGUiOiJDT04iLCJSZWNpcCI6MSwiX3hj) and download the VSIX file, close Visual Studio, then double click the VSIX file to install it (restarting Visual Studio to complete the installation).

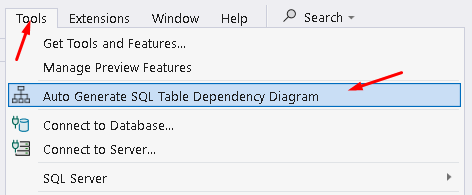


You will be prompted to install the extension for Visual Studio 22.



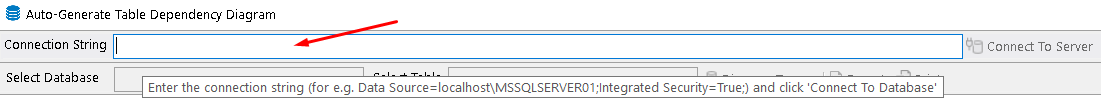
# Creating a Diagram

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



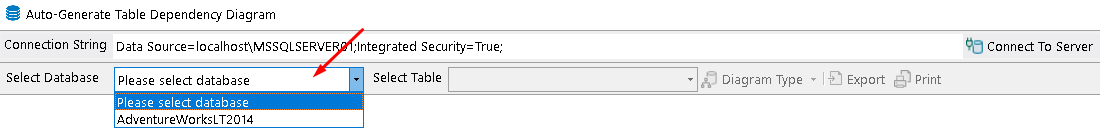
## Connect to a SQL Server Instance

Enter the connection string to your server and click on the Connect to Server button, to populate the database dropdown with the public databases.



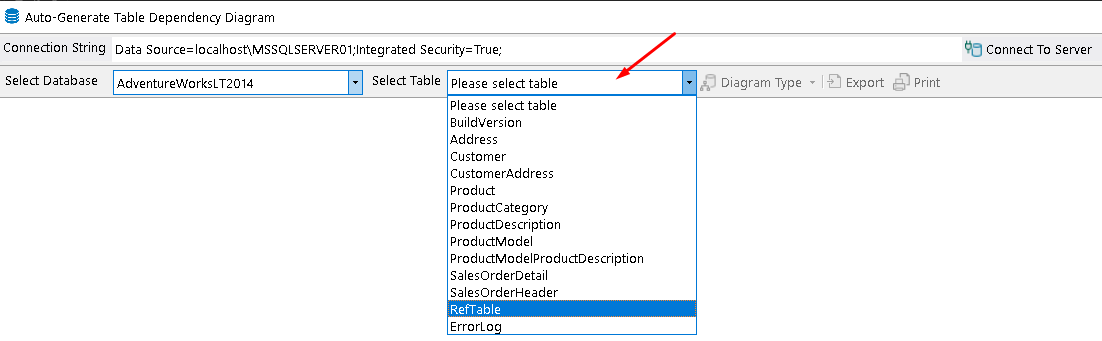
## Select Database

All the non-system related databases will be displayed.



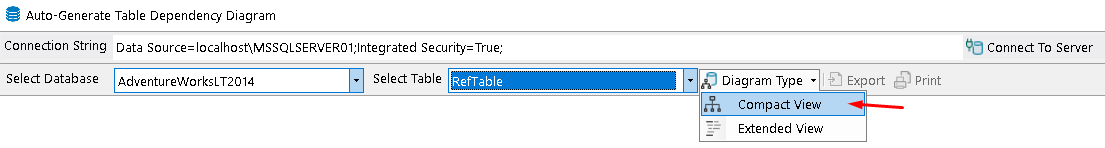
## Select Table

All the non-system related tables will be displayed.

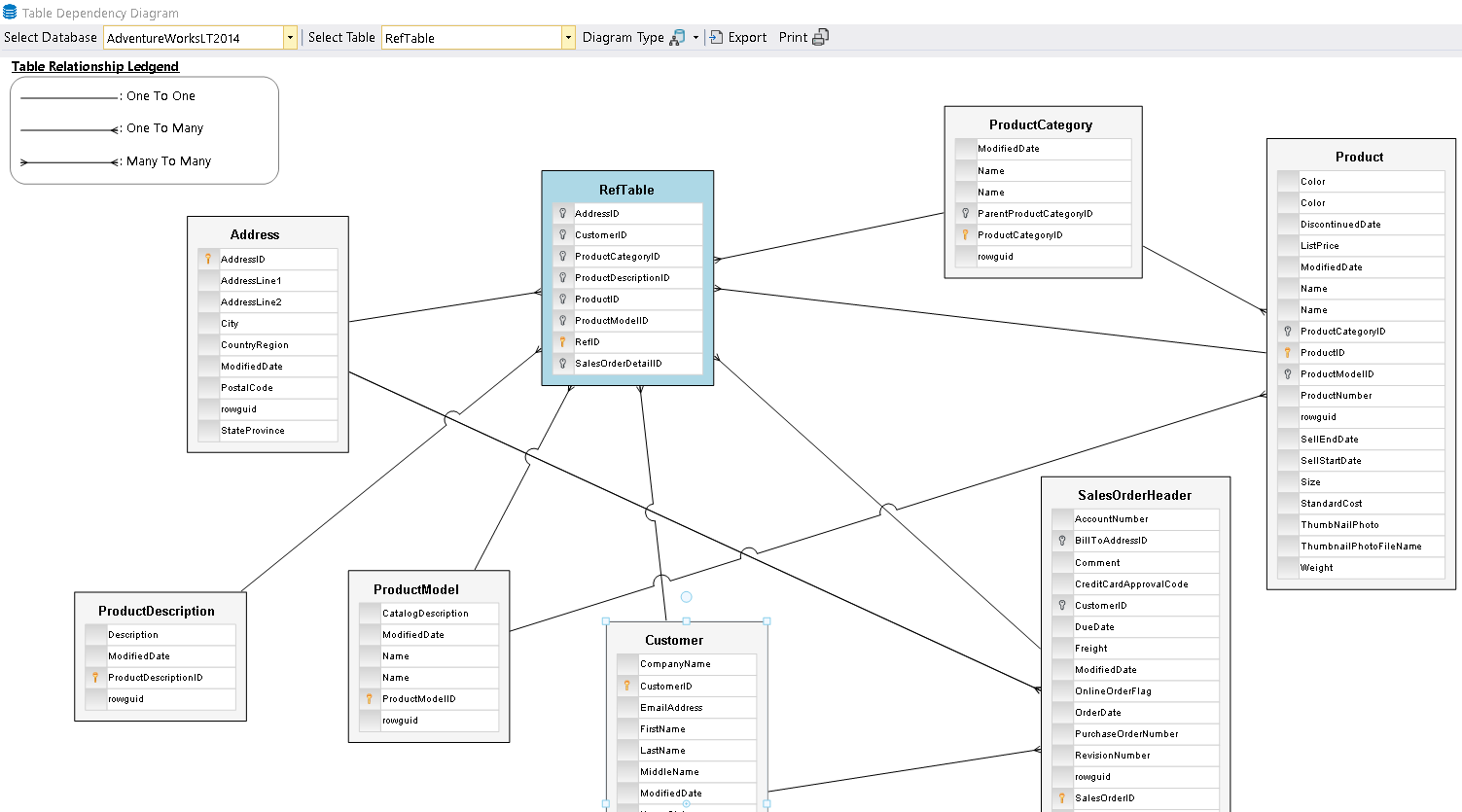


## Compact View

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).

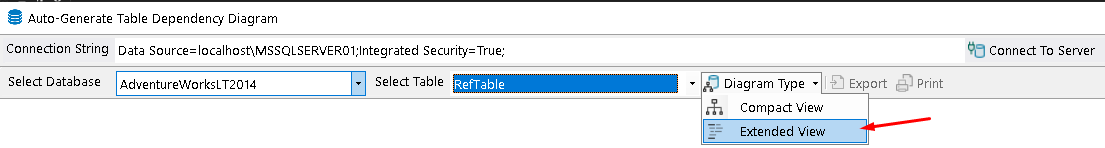


## Example of a Compact Layout Diagram

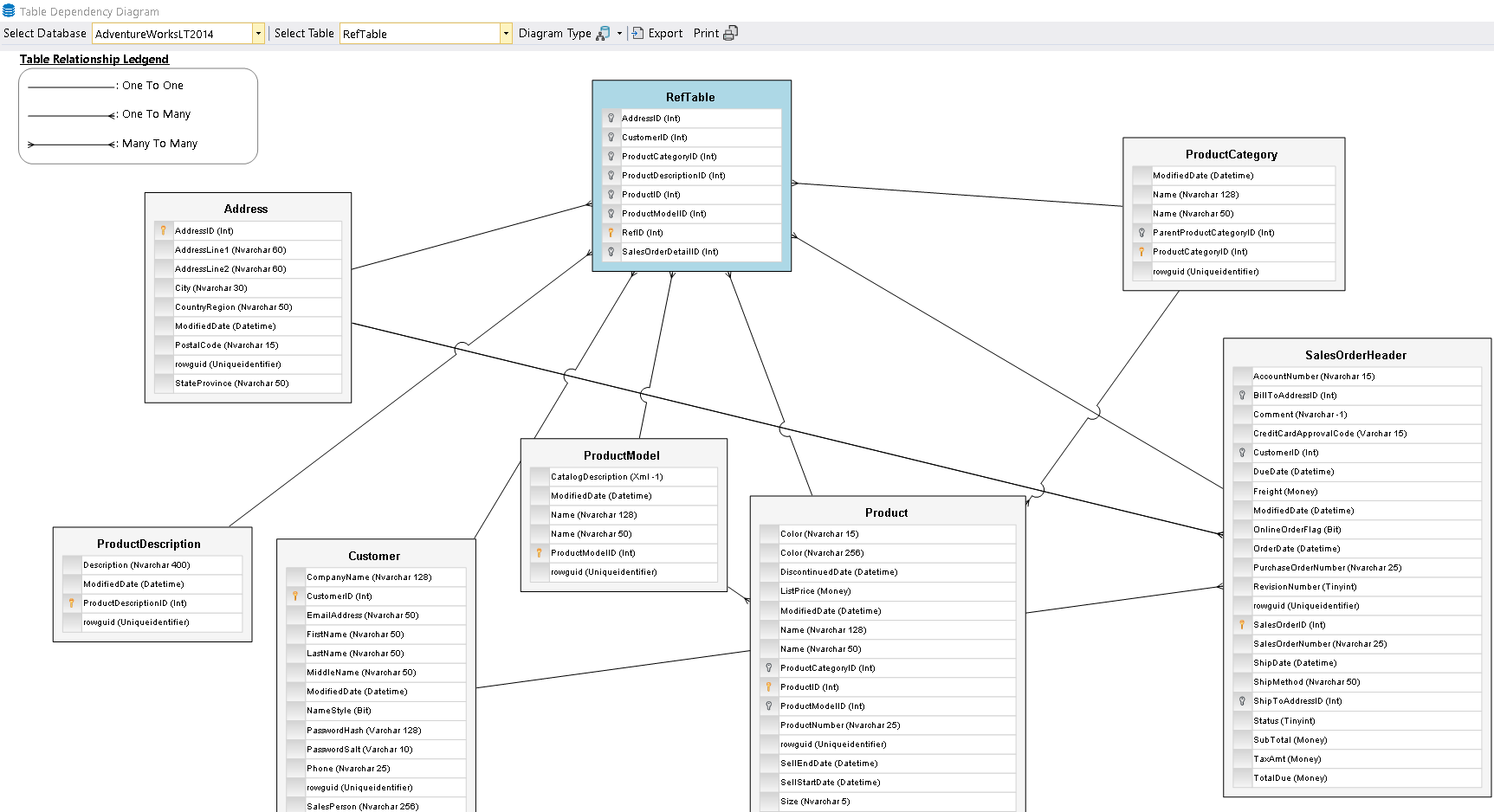


## Extended View

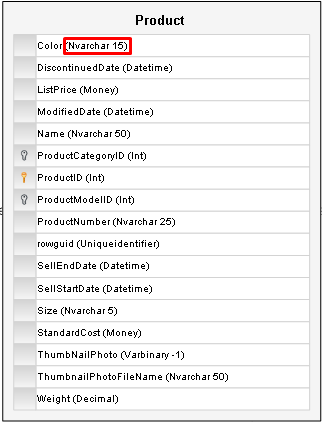
The user can select the extended layout from the split button.



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



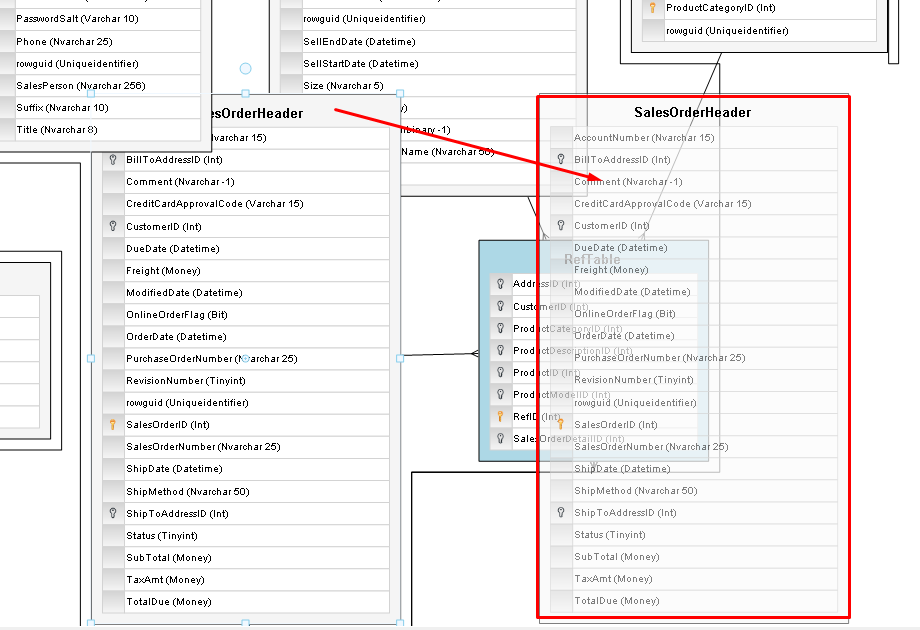
In the Extended view, you will also get the data-type (and length if appropriate) of each entity.



## Rearranging Tables

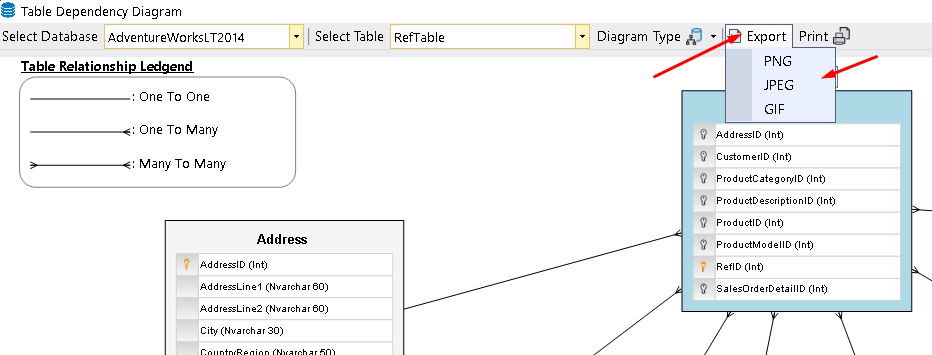
You are able to move\drag tables around the diagram, to space out the tables. Click on a table and hold your right mouse button down, and move using your table, the connection lines will follow your table.

You can also drag the whole diagram around, when the Hand icon appears (over the white space on the diagram).



# Exporting a Diagram

If you wish to share your diagram with team members or keep it for future references, you can do so by exporting the ERD as an image. 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 image.



# 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.

