

XpertBI BI builders

Xpert BI 3.0.0.0
USER MANUAL

Table of Contents

| | | |
|----------|--|----|
| 1. | Installation, Setup and Getting started..... | 8 |
| 1.1. | Installation requirements..... | 8 |
| 1.2. | License key | 8 |
| 1.3. | Management database | 8 |
| 1.4. | Software Updates | 8 |
| 1.5. | Multi-stack and multi-user..... | 8 |
| 1.6. | Supported Database Technologies | 9 |
| 1.7. | Database-based Vs Schema-based Solution | 9 |
| 2. | Xpert BI Extraction | 10 |
| 2.1. | Xpert BI Extraction Background | 10 |
| 2.2. | Brief Introduction to Xpert BI Extraction | 10 |
| 2.3. | Using Xpert BI Extraction– Quick guide | 11 |
| 2.3.1. | Getting started | 11 |
| 2.3.2. | Add Data Source Wizard | 11 |
| 2.3.2.1. | Page 1: Data Source | 12 |
| 2.3.2.2. | Page 2: Tables | 13 |
| 2.3.2.3. | Page 3: Names..... | 14 |
| 2.3.2.4. | Page 4: Dimensional Model | 14 |
| 2.3.2.5. | Page 5: Tabular Model | 15 |
| 2.3.2.6. | Page 6: Data Load..... | 16 |
| 2.3.2.7. | Page 7: Summary | 17 |
| 2.3.3. | Add Data Source Step-By-Step..... | 18 |
| 2.3.3.1. | Data source definition..... | 18 |
| 2.3.3.2. | Configuration | 21 |
| 2.3.3.3. | Operation | 23 |
| 2.4. | Using Xpert BI Extraction – Advanced guide..... | 24 |
| 2.4.1. | License, Installation and Setup | 24 |
| 2.4.1.1. | Installation | 24 |
| 2.4.1.2. | License..... | 24 |
| 2.4.1.3. | Configure Xpert BI – Solution Setup..... | 24 |
| 2.4.1.4. | Configure Xpert BI – remote processing (SQL Server only) | 25 |
| 2.4.2. | Home and Global Buttons..... | 25 |
| 2.4.2.1. | Help | 25 |
| 2.4.2.2. | Options..... | 26 |
| 2.4.2.3. | Global Buttons..... | 27 |

| | | |
|-----------|---|----|
| 2.4.2.4. | Data Source Explorer Filter/Search | 27 |
| 2.4.3. | Source Definition..... | 28 |
| 2.4.3.1. | Add data source | 28 |
| 2.4.3.2. | Delete data source | 28 |
| 2.4.3.3. | Add version | 29 |
| 2.4.3.4. | Delete version | 29 |
| 2.4.3.5. | Import source metadata | 30 |
| 2.4.3.6. | Export metadata database..... | 32 |
| 2.4.3.7. | Configure Relations..... | 33 |
| 2.4.3.8. | Import UDFs | 34 |
| 2.4.3.9. | List UDFs..... | 34 |
| 2.4.3.10. | Edit destination metadata | 34 |
| 2.4.3.11. | Define Global Filters..... | 37 |
| 2.4.3.12. | Properties..... | 37 |
| 2.4.4. | Configuration | 38 |
| 2.4.4.1. | Add Instance..... | 38 |
| 2.4.4.2. | Delete instance | 38 |
| 2.4.4.3. | Table Filters..... | 38 |
| 2.4.4.4. | Global filters..... | 39 |
| 2.4.4.5. | Edit Connection (instance and table level) | 39 |
| 2.4.4.6. | Set Destination..... | 41 |
| 2.4.4.7. | Check Source Metadata | 42 |
| 2.4.4.8. | Grant ODS permission..... | 42 |
| 2.4.4.9. | Copy exe parameters (instance and table level)..... | 43 |
| 2.4.4.10. | Edit Data Load Settings | 44 |
| 2.4.4.11. | Instance and Instance Table Properties..... | 45 |
| 2.4.4.12. | Tracking table values or attribute changes..... | 46 |
| 2.4.5. | Operation | 48 |
| 2.4.5.1. | Process data source | 48 |
| 2.4.5.2. | View Table relations..... | 51 |
| 2.4.5.3. | Check Consistency..... | 52 |
| 2.4.5.4. | Query Source Data | 53 |
| 2.4.5.5. | Remove Foreign Keys..... | 53 |
| 2.4.5.6. | Instance and Instance Table Properties..... | 53 |
| 2.4.5.7. | Analyse Data Source..... | 54 |
| 2.4.6. | Change Management..... | 56 |

| | | |
|-----------|--|----|
| 2.4.6.1. | Change table and column name | 56 |
| 2.4.6.2. | Add or remove column | 56 |
| 2.4.6.3. | Add or remove table | 56 |
| 2.4.6.4. | Edit relationships and/or keys | 56 |
| 2.4.6.5. | Change data type and/or length | 57 |
| 2.4.7. | Administration | 57 |
| 2.4.7.1. | Reload | 57 |
| 2.4.7.2. | Event Log..... | 57 |
| 2.4.7.3. | Configure Xpert BI dB..... | 58 |
| 2.4.7.4. | Manage Modules | 58 |
| 2.4.7.5. | License..... | 58 |
| 2.4.7.6. | Connection Manager..... | 59 |
| 2.4.7.7. | Secret Manager | 61 |
| 2.4.7.8. | Reports | 63 |
| 2.4.7.9. | Migrate Environment..... | 64 |
| 2.4.7.10. | Manage Tags | 75 |
| 2.4.8. | XBI_batch.exe..... | 77 |
| 2.4.8.1. | Installation and licensing..... | 77 |
| 2.4.8.2. | Parameters | 77 |
| 2.4.8.3. | Error codes | 78 |
| 2.4.8.4. | Scheduling | 79 |
| 2.4.8.5. | Error message handling..... | 81 |
| 2.4.8.6. | Command Window Information | 81 |
| 2.5. | ODS Structure (SQL Server and Azure SQL Database) | 82 |
| 2.5.1. | Tables | 82 |
| 2.5.2. | Primary Keys..... | 82 |
| 2.5.3. | Foreign Keys and Foreign Keys Roles | 82 |
| 2.6. | ODS Structure (Azure Synapse SQL and Snowflake) | 82 |
| 2.7. | ODS Structure (Azure Data Lake) | 83 |
| 3. | Xpert BI Transformation..... | 84 |
| 3.1. | Xpert BI Transformation Background | 84 |
| 3.1.1. | ITS – A new approach to ELT/ETL..... | 84 |
| 3.1.2. | Structure and Naming conventions in Xpert BI Transformation..... | 84 |
| 3.2. | Using Xpert BI Transformation..... | 85 |
| 3.2.1. | Solution Explorer and Object Dependency Graph | 85 |
| 3.2.1.1. | Graph menus and icons | 86 |

| | | |
|----------|---|-----|
| 3.2.1.2. | Graph Legend | 90 |
| 3.2.1.3. | Solution Explorer menus | 92 |
| 3.2.1.4. | Reload Solution Explorer | 92 |
| 3.2.1.5. | Solution Explorer Filter/Search | 93 |
| 3.2.2. | Column Lineage | 93 |
| 3.2.2.1. | Visible Levels | 94 |
| 3.2.2.2. | Stored Procedures | 95 |
| 3.2.3. | Object dependency configuration background | 96 |
| 3.2.3.1. | How Xpert BI achieves Object Dependency control | 96 |
| 3.2.3.2. | Using data source prefixes | 96 |
| 3.2.4. | Transformation and data processing design | 96 |
| 3.2.4.1. | Create ITS table (from views and stored procedures) | 96 |
| 3.2.4.2. | Specify object derived dependencies | 99 |
| 3.2.4.3. | Specify process groups | 100 |
| 3.2.4.4. | Executable Settings | 102 |
| 3.2.4.5. | Adding databases to the Xpert BI solution | 103 |
| 3.2.4.6. | Incremental data load | 104 |
| 3.2.4.7. | Dependency Loops | 104 |
| 3.2.4.8. | Create History Tables | 105 |
| 3.2.5. | Data load options | 107 |
| 3.2.5.1. | Execute object | 107 |
| 3.2.5.2. | Execute object tree | 107 |
| 3.2.5.3. | Execute process group | 107 |
| 3.2.5.4. | Execute history table | 107 |
| 3.2.6. | Smart Processing | 108 |
| 3.2.7. | Custom Object | 109 |
| 3.2.7.1. | Create Custom Object | 109 |
| 3.2.7.2. | Delete Custom Object | 109 |
| 3.2.7.3. | Edit Custom Object | 109 |
| 3.2.7.4. | Custom Object Window | 109 |
| 3.2.7.5. | Execution | 110 |
| 3.2.7.6. | Script Type: Powershell | 111 |
| 3.2.7.7. | Script Type: Python | 112 |
| 3.2.8. | Database Schema | 112 |
| 3.3. | Data Interfaces | 113 |
| 3.3.1. | OLAP | 113 |

| | | |
|----------|--|-----|
| 3.3.1.1. | Import | 113 |
| 3.3.1.2. | Refresh | 113 |
| 3.3.1.3. | Copy ELM Descriptions..... | 114 |
| 3.3.1.4. | Delete | 114 |
| 3.3.1.5. | Edit | 114 |
| 3.3.2. | Tabular | 115 |
| 3.3.2.1. | Import | 115 |
| 3.3.2.2. | Refresh | 115 |
| 3.3.2.3. | Copy ELM Descriptions..... | 116 |
| 3.3.2.4. | Delete | 116 |
| 3.3.2.5. | Edit | 116 |
| 3.3.3. | Power BI | 117 |
| 3.3.3.1. | Configure..... | 117 |
| 3.3.3.2. | Sync PowerBI.com..... | 118 |
| 3.3.3.3. | Document Power BI Objects | 119 |
| 3.3.3.4. | Copy Report/Dataset Descriptions | 119 |
| 3.3.3.5. | Import Local Power BI Report | 120 |
| 4. | Xpert BI Collection..... | 122 |
| 4.1. | Creating a Collection | 122 |
| 4.1.1. | Editing a Collection..... | 123 |
| 4.1.2. | Copying a Collection..... | 123 |
| 4.1.3. | Drag and Drop | 123 |
| 4.2. | Xpert BI Collection - Flat File | 124 |
| 4.3. | Xpert BI Collection – XML/JSON..... | 127 |
| 4.4. | Xpert BI Collection – SOAP | 130 |
| 4.4.1. | Bloomberg..... | 131 |
| 4.4.2. | Bureau Veritas REC75 | 132 |
| 4.5. | Xpert BI Collection – REST | 133 |
| 4.5.1. | NEMS Panorama | 136 |
| 4.5.2. | WELS Operator | 137 |
| 4.5.3. | Box..... | 138 |
| 4.5.4. | Google API..... | 138 |
| 4.5.4.1. | Google API Setup Guide | 139 |
| 4.5.5. | Azure Cosmos DB | 140 |
| 4.5.6. | Proactima | 141 |
| 4.6. | Xpert BI Collection – SAP RFC | 142 |

| | | |
|----------|---|-----|
| 4.7. | Xpert BI Collection – SSAS | 143 |
| 4.8. | Scheduling Xpert BI Collection data loads | 144 |
| 4.8.1. | XBI_collection.exe | 144 |
| 4.8.1.1. | Installation and licensing..... | 144 |
| 4.8.1.2. | Parameters | 144 |
| 4.8.1.3. | Command Window Information | 144 |
| 4.8.2. | Data loading | 145 |
| 4.9. | Analyse Data | 145 |
| 5. | Xpert BI Publication..... | 146 |
| 5.1. | REST API | 146 |
| 5.1.1. | Installation | 146 |
| 5.1.2. | Creating and Managing an API..... | 148 |
| 5.1.3. | Managing Security..... | 149 |
| 5.1.4. | API Functionality | 150 |
| 5.1.5. | API Documentation | 152 |
| 5.2. | Create analysis models | 153 |
| 5.2.1. | SSAS Multidimensional | 153 |
| 5.2.2. | SSAS Tabular..... | 153 |
| 5.2.3. | Excel Power Pivot..... | 154 |
| 5.3. | Export Groups | 155 |
| 6. | Master Data | 157 |
| 6.1. | Configuration | 157 |
| 6.2. | Master Data Explorer | 157 |
| 6.3. | Entity | 158 |
| 6.3.1. | Create Entity..... | 158 |
| 6.3.2. | Edit Entity | 159 |
| 6.3.3. | Delete Entity..... | 159 |
| 6.4. | Data Load | 159 |
| 6.4.1. | Create Data Load..... | 159 |
| 6.5. | Subscription View..... | 160 |
| 6.5.1. | Create Subscription View | 160 |
| 6.5.2. | Edit Subscription View | 161 |
| 6.5.3. | Delete Subscription View | 161 |
| 7. | DataOps..... | 162 |
| 7.1. | Automated Data Quality | 162 |
| 7.1.1. | Data Quality Item | 162 |

| | | |
|----------|--|-----|
| 7.1.1.1. | Add New Data Quality Item | 162 |
| 7.1.1.2. | Edit Data Quality Item..... | 163 |
| 7.1.1.3. | Delete Data Quality Item | 163 |
| 7.1.1.4. | Execute Test | 163 |
| 7.1.1.5. | Schedule a data quality item..... | 164 |
| 7.1.2. | DataOps Folder | 164 |
| 7.1.2.1. | Add New Folder..... | 164 |
| 7.1.2.2. | Rename Folder | 165 |
| 7.1.2.3. | Delete Folder..... | 165 |
| 7.1.2.4. | Execute All Tests | 166 |
| 7.1.2.5. | Schedule multiple data quality tests inside a folder | 166 |
| 7.1.3. | Test Types | 166 |
| 7.1.3.1. | Duplicates..... | 166 |
| 7.1.3.2. | Fact Table Relations | 169 |
| 7.2. | Unit Test..... | 171 |
| 7.2.1. | Add New Unit Test Item..... | 171 |
| 7.2.2. | Edit Unit Test Item | 171 |
| 7.2.3. | Delete Unit Test Item | 172 |
| 7.2.4. | Execute Unit Test Item..... | 173 |
| 7.2.5. | Setup/Teardown | 174 |
| 7.3. | Test Log | 175 |
| 8. | Appendix I | 177 |
| 8.1.1. | Xpert BI Solution Catalog | 177 |

1. Installation, Setup and Getting started

1.1. Installation requirements

Xpert BI is a licensed application that may only be used with a valid license.

The application install file is available from www.bi-builders.com/software/

The application is installed per-machine. The person responsible for installing Xpert BI must have administrator rights on the relevant machine. In some environments depending on UAC settings the following might be necessary:

When running the setup file use Run as administrator. This is so that Xpert BI is launched with administrator privileges so that it can create the C:\Program Files\Xpert BI folder. The setup installer adds Everyone Full Control NTFS permissions to the install folder.

Minimum server requirements:

- SQL Server 2008 R2, Azure SQL Database or Azure Managed Instance
- .NET 4.7.2 framework
- Internet connection (preferably)

1.2. License key

Upon purchase a license key will be distributed. The Xpert BI application will ask for the license file when starting the application for the first time.

1.3. Management database

Xpert BI uses a configuration database called the “management database”, which is where all application configurations and metadata are stored. This database is installed with the installation of the application and is updated with every product update.

The database name has 2 parts: Prefix and Base. The Prefix is determined by the user and will be the prefix of all Xpert BI databases in the solution. The Base is _XBI_MGMT_DEV which cannot be changed and is the management database tag.

The prefix that you choose for your management database will also be the prefix for the rest of your Xpert BI solution, such as the ODS and STG databases, and with Transformation the ETL and ELM databases.

1.4. Software Updates

The application is NOT updated automatically when a new version is available. The new version must be downloaded and installed. When installing a new version, it is important that none of the files in the C:\Program Files\Xpert BI\ is open or in use as the installation will update all files.

1.5. Multi-stack and multi-user

It is possible to have several stacks and switch between these (see Configure Xpert BI dB). During start-up, XpertBI will start with the previous database stack that was opened by the current user. If one user changes stack, it will not be changed for any other user. The exception is when a new user that has not run XpertBI yet starts XpertBI for the first time, the database stack will be the same as the last user that started XpertBI.

1.6. Supported Database Technologies

Xpert BI supports several database/storage technologies throughout the application. Below is an overview of where the various technologies are supported:

| | SQL Server | Azure SQL Database | Azure Synapse SQL | Azure Managed Instance | Azure Data Lake | Snowflake |
|---|------------|--------------------|-------------------|------------------------|-----------------|-----------|
| Application Database | X | X | | X | | |
| Stack Location | X | X | X | X | | |
| Collection Destination* | X | X | X | X | | |
| Extraction Destination** | X | X | X | X | X | |
| Extraction Destination, via Set Destination | X | X | X | X | X | X |
| Transformation | X | X | X | X | | |
| Publication | X | X | X | X | *** | |
| DataOps | X | X | X | X | | |
| Master Data | X | X | X | X | | |

* Applies to the destination tables. Files can be stored in local (or network) folders or in Azure Data Lake. Sources can also be Web APIs and SAP RFC.

** Applies to destination tables. Extraction can load from many data sources.

*** Supported as Export Group Destination

For more details on supported/unsupported features, see details in each section.

1.7. Database-based Vs Schema-based Solution

If a stack is created on SQL Server or Azure Managed Instance, the solution is database-based. What this means is that the various parts of the stack (ODS, ETL etc.) are created as separate databases (for example MyPrefix_ODS). If the stack is created on Azure SQL Database or Azure Synapse SQL, the solution is schema-based, meaning that the parts of the stack are created as schemas inside the MyPrefix_STACK database.

2. Xpert BI Extraction

2.1. Xpert BI Extraction Background

Xpert BI Extraction is an application used in the ETL process of a data warehouse, BI solution or as a part of an information management architecture. After simple configuration and specification of a data source, the application will load data from a local or external data source into a destination of your choice.

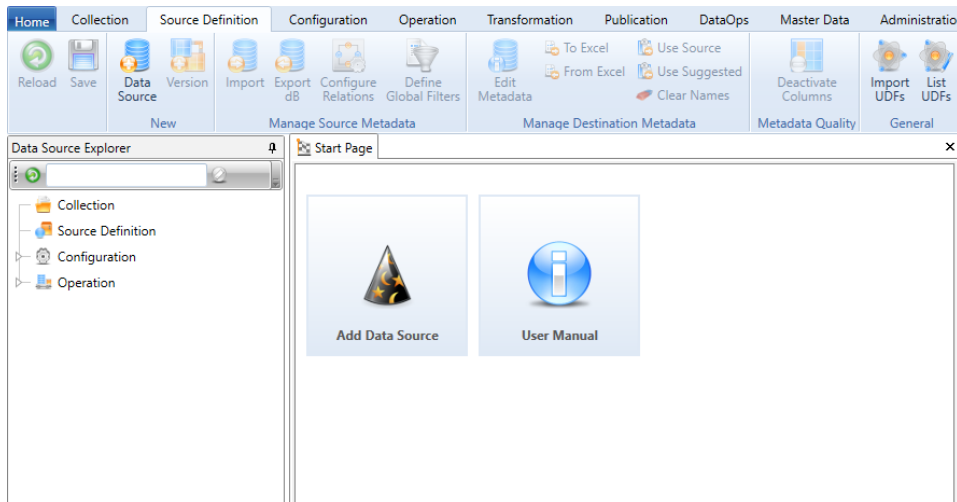
Xpert BI Extraction ensures data integrity and quality and provides you with complete control of your data imports.

High performance data extractions and loads are done through multi-threaded processing using bulk copy from the data source via a staging area and to the data warehouse destination.

The product is designed to dynamically reflect and detect changes in the data model. It also enables analysis and troubleshooting of data consistency.

2.2. Brief Introduction to Xpert BI Extraction

Xpert BI Extraction is structured into three main functional areas; Source Definition, Configuration, and Operation. These are represented by three top nodes in the Data Source Explorer and as separate ribbon tabs.



Source Definition

This is where you start to define the tables, relationships and columns of your data source. This information is imported from a SQL server, Oracle or DB2 source database or through an application adapter with extended metadata support (or a SQL server or Oracle metadata database, which must be created if your data source is unsupported). Source definition is also where you can give new user-friendly names to your tables and columns and add functions to your columns.

Configuration

This is where you set filters to your data source tables that will be included in the data source query. Configuration is also where you set the connection string to the source database and other data load configurations.

Operation

This is where you run the data load. You can also do other data analysis in Operation, such as viewing relationship and query source data.

2.3. Using Xpert BI Extraction– Quick guide

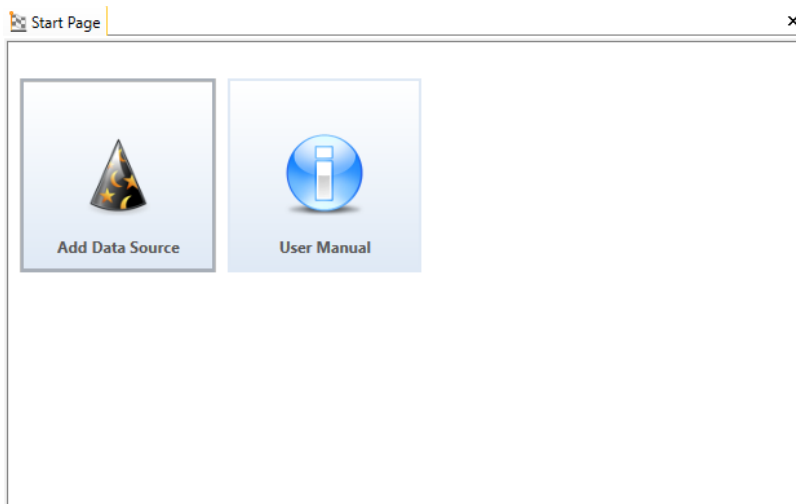
2.3.1. Getting started

When starting the application for the first time you must specify the Xpert BI database prefix, i.e. a naming prefix that will be used in all Xpert BI databases. See section 1.3 for more information.

The next step is to import data. The easiest way to do this is to follow either the Add Data Source Wizard in section 2.3.2 or the Add Data Source Step-By-Step guide in section 0.

2.3.2. Add Data Source Wizard

The easiest way to get started with importing data is to use the add data source wizard. The wizard can be opened from the start page by clicking on the 'Add Data Source' tile, or from the Home menu.



The following sections will explain each wizard page. However, it is not necessary to complete all pages to import data. First time users can skip page 4 to 6 by clicking on the finish button on page 3.

2.3.2.1. Page 1: Data Source

Select 'Create New Data Source' to import data from a new data source. Select 'Use Existing Data Source' to use an already imported data source. This will skip the next two pages and go directly to page 4.

1 Data Source

2 Tables

3 Names

4 Dimensional Model

5 Tabular Model

6 Data Load

7 Summary

Connect To New Data Source

Enter a name and description for the new data source and select a connection.
Or create a model for existing data source.

☒ Create New Data Source

☐ Use Existing Data Source

Name

MyDataSource

Description

Connection

Schema ⓘ

< Back Next > Finish Cancel

Name is the name of the data source and is required. It must be unique and cannot start with 'XBI_' as it is a reserved prefix.

Description is the data source description. It is optional, but strongly recommended.

Connection button sets up the connection to the data source and is required.

Schema is used to indicate what the database schema in the source is. It is optional. If it is not set, it defaults to the 'dbo' schema.

Use Existing Data Source

To select an existing data source, choose any data source listed in the drop-down box under **Name**. Only data sources that has been imported and configured are visible. That is, at least one instance must be available (see section 2.4.4.1 for more information). When using existing data source, the wizard will skip directly to page 4 since metadata has already been imported.

New Data Source

1 Data Source

2 Tables

3 Names

4 Dimensional Model

5 Tabular Model

6 Data Load

7 Summary

Connect To New Data Source
Enter a name and description for the new data source and select a connection.
Or create a model for existing data source.

☐ Create New Data Source

☒ Use Existing Data Source

Name

MyExistingDataSource (1/1)

< Back Next > Finish Cancel

2.3.2.2. Page 2: Tables

On page 2 it is possible to select which tables are to be imported. By default, all tables are included. It is possible to click on finish at this stage, if finish is clicked the wizard will skip to the last page.

New Data Source

1 Data Source

2 Tables

3 Names

4 Dimensional Model

5 Tabular Model

6 Data Load

7 Summary

Select Tables to Import
Import all tables, or choose to import only some tables.

☐ Include all tables (5)

☒ Select tables

| Table Name | |
|------------|-------------------------------------|
| Company | <input checked="" type="checkbox"/> |
| Person | <input checked="" type="checkbox"/> |
| Project | <input checked="" type="checkbox"/> |
| TimeReg | <input checked="" type="checkbox"/> |
| Un Related | <input type="checkbox"/> |

< Back Next > Finish Cancel

Include all tables will try to import all tables. Some tables, if they do not contain a primary key, cannot be imported. If any table could not be imported, it will be shown on page 3.

Select tables allows for selecting and deselecting tables. At least one table must be selected in order to advance the wizard.

2.3.2.3. Page 3: Names

On this page it is possible to change the names of the tables and columns. If any table names are empty or duplicate, it is not possible to advance the wizard. Same applies for columns. If any tables could not be imported, it will be shown on this page with an explanation. It is possible to click finish and skip directly to the last page.

Edit Names
Edit names of table and columns.

| | Table Name Source | Table Name |
|-------------------------------------|-------------------|------------|
| <input checked="" type="checkbox"/> | Company | Company |
| <input checked="" type="checkbox"/> | Person | Person |
| <input checked="" type="checkbox"/> | Project | Project |

| Include | Column Name Source | Column Name |
|-------------------------------------|--------------------|---------------|
| <input checked="" type="checkbox"/> | Project | Project |
| <input checked="" type="checkbox"/> | ProjectNo | ProjectNumber |
| <input checked="" type="checkbox"/> | ProjectType | ProjectType |

| | | |
|-------------------------------------|------------|-------------------------------------|
| <input checked="" type="checkbox"/> | TimeReg | TimeReg |
| <input checked="" type="checkbox"/> | Un Related | (Table is disabled. No primary key) |

< Back Next > Finish Cancel

2.3.2.4. Page 4: Dimensional Model

This page configures the type of tables imported. A table is either a dimension or fact table. The checkbox on the left side of each table indicates if the table is going to be included in the model. If it is not selected, it will only be added to ODS. The wizard will attempt to automatically determine whether a table is dim or fact. This can be changed by clicking on the Fact and Dim buttons in the Type column. **Show Model** checkbox will toggle the visibility of the graph. Boxes with orange border are fact, and boxes with blue border are dim. Relations between fact and dim are automatically created and cannot be altered in the wizard. See section 3.2.4.2 on how to customize relations. **Reset Selection** will reset the fact and dim selection back to the automatically selected values.

Dimensional Model
Select which tables are fact tables and which tables are dimensions.
Deselect tables to stop automatic generation of ETL and ELM views.

☒ Show Model Reset Selection

| <input checked="" type="checkbox"/> | Table Name | Type |
|-------------------------------------|------------|----------|
| <input checked="" type="checkbox"/> | Company | Fact Dim |
| <input checked="" type="checkbox"/> | Person | Fact Dim |
| <input checked="" type="checkbox"/> | Project | Fact Dim |
| <input checked="" type="checkbox"/> | TimeReg | Fact Dim |

< Back Next > Finish Cancel


2.3.2.5. Page 5: Tabular Model



On this page it is possible to enable generation of a tabular model. This step is optional.

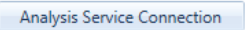
New Data Source

Tabular Model
Choose whether to automatically create a Tabular Model.

Generate Tabular Model ☒

Database Name 
MyTabular

Project Folder 
C:\TabularProjects\ 



< Back Next > Finish Cancel

Generate Tabular Model enables or disables the generation of a tabular model.

Database Name is the name used for the database, tabular cube and the project name. Since the name is used as database name, it must be unique on the analysis service server. This field is not optional.

Project Folder is the path where the tabular project is generated. This is the root folder where the project file is created. This field is not optional.

Analysis Service Connection will open a connection dialog and a tabular type analysis service server must be selected. This field is not optional.

2.3.2.6. Page 6: Data Load

Enable or disable what data to load when finishing the wizard. This step is optional.

New Data Source

1 Data Source
2 Tables
3 Names
4 Dimensional Model
5 Tabular Model
6 **Data Load**
7 Summary

Load Data From Source
Select what data to load. Tabular Model processing requires data source load.

Load Source Data ☒

Process Tabular Model ☒

< Back Next > Finish Cancel

Load Source Data will load data from the source defined on the first page. This data will be loaded into ODS.

Process Tabular Model will execute the tabular model. The data loaded into ODS will then be loaded into the tabular cube. 'Load Source Data' must be selected for this to be enabled.

2.3.2.7. Page 7: Summary

The summary page shows all the steps that has been configured. When finish is clicked, all the steps will run in sequence.

If any step fails, the wizard will stop and not execute any other step. The message column will show what went wrong. If it failed, it is not possible to go back and change the configuration. It is only possible to close the wizard. When closing after failure, a prompt will ask to try and revert any changes made.

| Task | Message |
|------------------------------|---------|
| Create Data Source | |
| Save Metadata | |
| Create Instance | |
| Create ODS Tables | |
| Create ETL and ELM Views | |
| Create Process Group | |
| Create Table Dependencies | |
| Create Tabular Project Files | |
| Deploy Tabular Cube | |
| Process Data | |
| Process Tabular Model | |

If finish was clicked before page 6, the summary page will have the option of opening the data processing tab after wizard is closed.

☐ Open Data Processing After Wizard

Click Finish to save.

< Back Finish Cancel

2.3.3. Add Data Source Step-By-Step

This step-by-step guide will show each step needed to import metadata from a source and load the data.

2.3.3.1. Data source definition

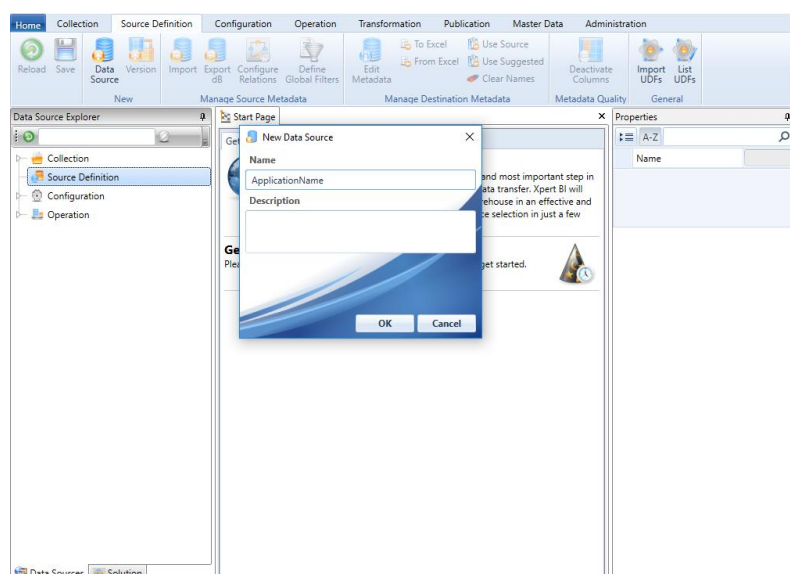
Source definition is where the data source metadata is retrieved and selected. See section 0 for more detailed information.

Naming your data source and version

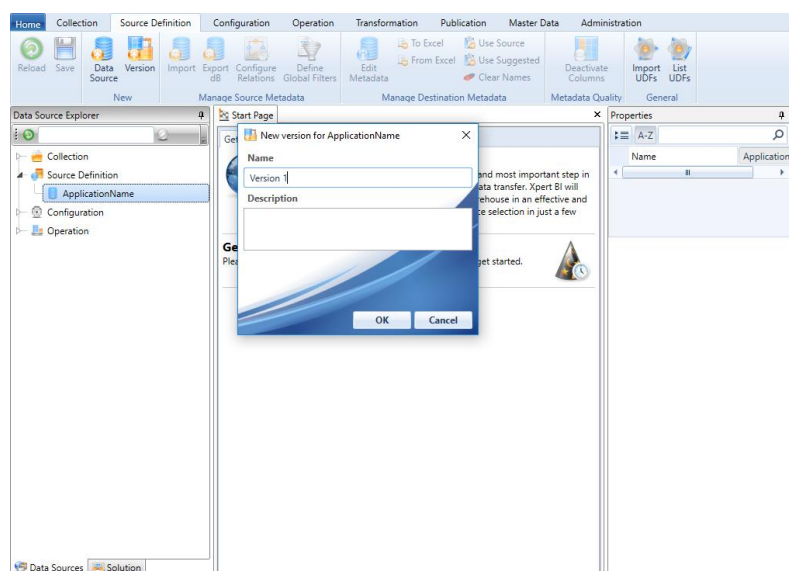
Click on Source Definition and choose Add Data Source from the ribbon or from a right-click.

A window appears. Type in the name of your data source (required) and a description (optional).

You will now find your data source listed in the solution explorer under Source Definition.



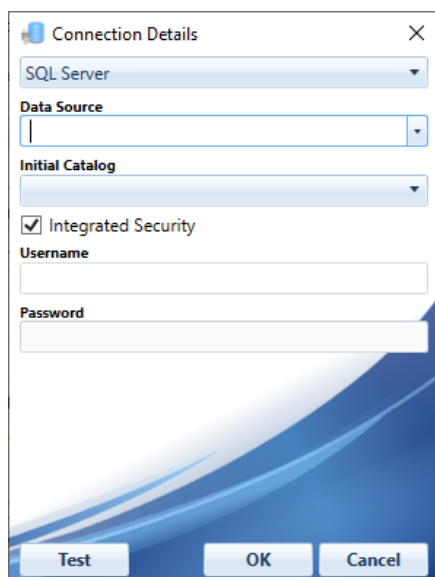
Next, you need to specify your data source version. Click on your data source and choose New Version from the Ribbon or from a right-click. Specify the name or number of your data source version.



Note: You may not name your data source with the "XBI_" prefix as this is a reserved prefix in the Xpert BI application.

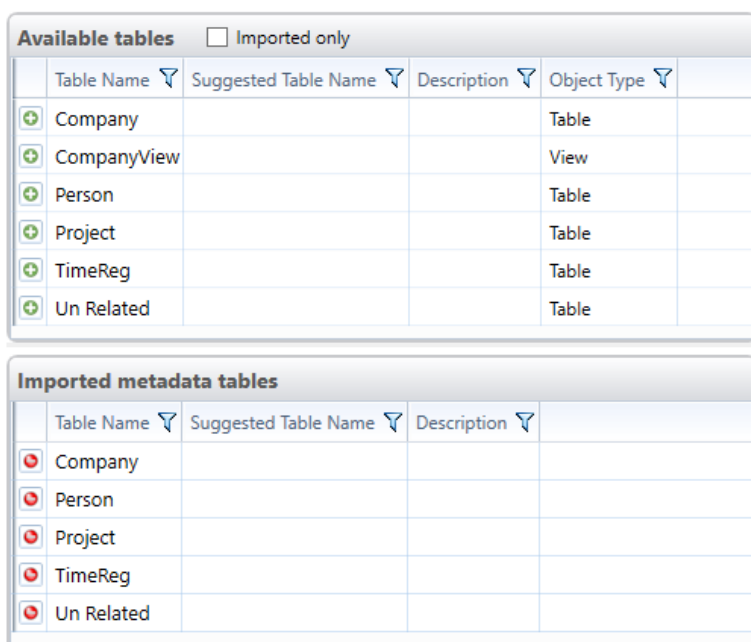
Defining the data model of the data source

Right-click on the version you want to define and choose Import Source Metadata or choose Import from the Manage Source Metadata group in the ribbon. A new tab opens. Click the Connection button to open the connection dialog box. Select the connection provider and fill out the connection string parameters. Click Test to test the connection, and OK to use the connection.



The 'Connection Details' dialog box is shown. It has a title bar with a close button (X). The 'Provider' dropdown is set to 'SQL Server'. Below it is a 'Data Source' dropdown. The 'Initial Catalog' dropdown is empty. The 'Integrated Security' checkbox is checked. There are text boxes for 'Username' and 'Password'. At the bottom are 'Test', 'OK', and 'Cancel' buttons.

You import/remove table metadata by using drag-and-drop between Available tables and Imported metadata tables, or by clicking the import or remove buttons. (You can ctrl-select and shift-select tables). Click Save to save the metadata.



The interface shows two panels. The top panel is titled 'Available tables' and has a checkbox 'Imported only' which is unchecked. It contains a table with columns: Table Name, Suggested Table Name, Description, and Object Type. The bottom panel is titled 'Imported metadata tables' and contains a table with columns: Table Name, Suggested Table Name, and Description. Both tables have a green plus icon in the first column, indicating they can be imported.

| Table Name | Suggested Table Name | Description | Object Type |
|-------------|----------------------|-------------|-------------|
| Company | | | Table |
| CompanyView | | | View |
| Person | | | Table |
| Project | | | Table |
| TimeReg | | | Table |
| Un Related | | | Table |

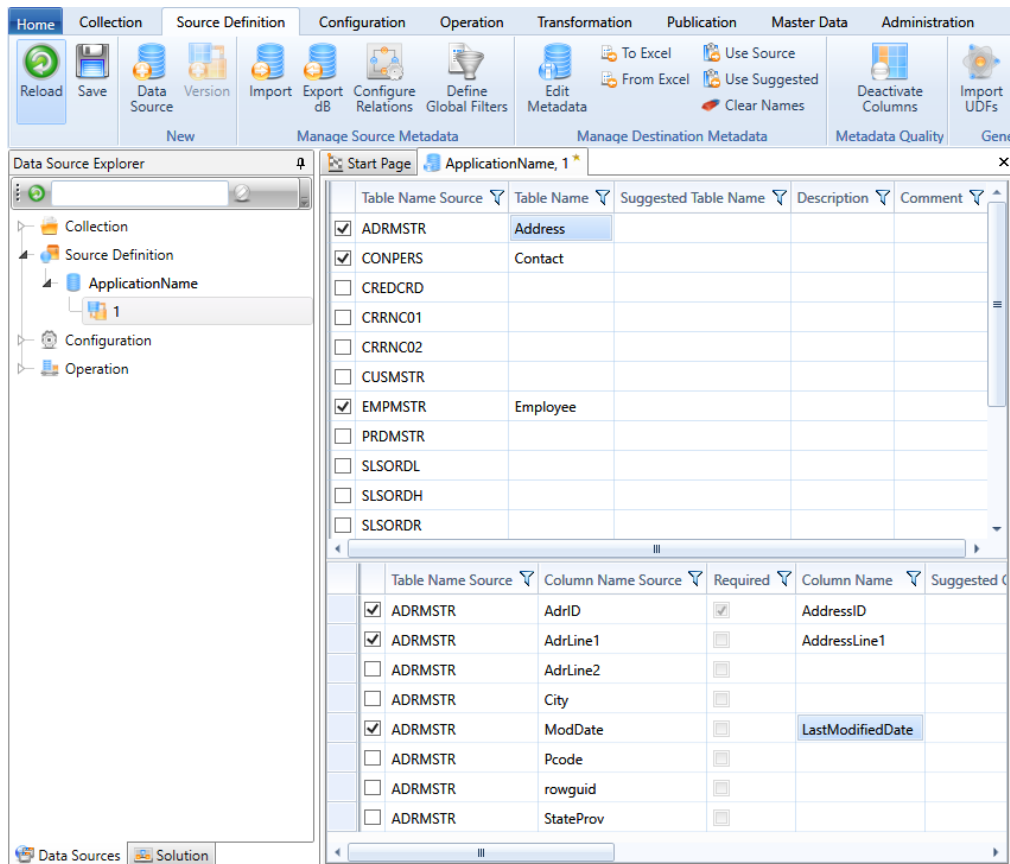
| Table Name | Suggested Table Name | Description |
|------------|----------------------|-------------|
| Company | | |
| Person | | |
| Project | | |
| TimeReg | | |
| Un Related | | |

Next, the metadata can be exported to a metadata database in your data warehouse for further refinement of metadata. You can create relationships in this database and then import it to Xpert BI again. Please see chapter Export metadata database for further details.

Note: To fully utilize the functionality of Xpert BI, it is recommended that you spend some time to ensure a complete definition of your data source table relationships.

Configure destination metadata

Xpert BI allows you to map the table and column names of your data source to more user-friendly names. This can be done directly in Xpert BI or in Excel.



If you do not want to rename table names or column names, you can choose Use Source from the Manage Destination Metadata tool group in the ribbon.

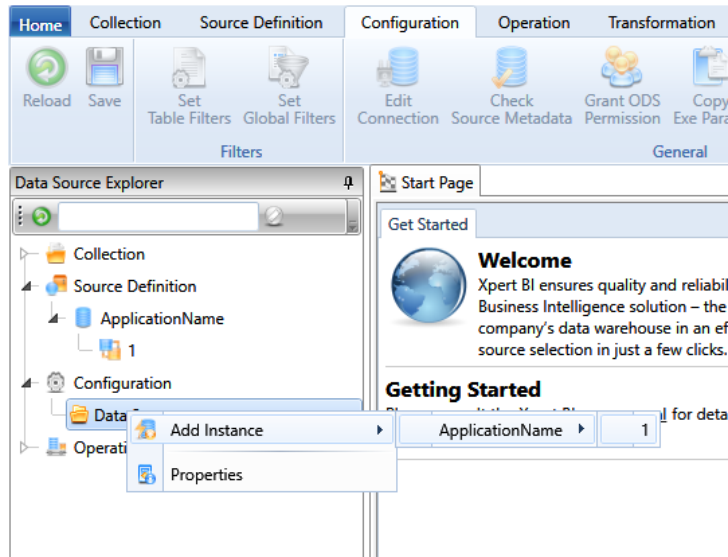
2.3.3.2. Configuration

In the configuration part, it is possible to configure multiple settings to allow for more advanced ways of loading data. See section 2.4.4 for more information.

In this quick guide, however, only the connection to the source data is set up.

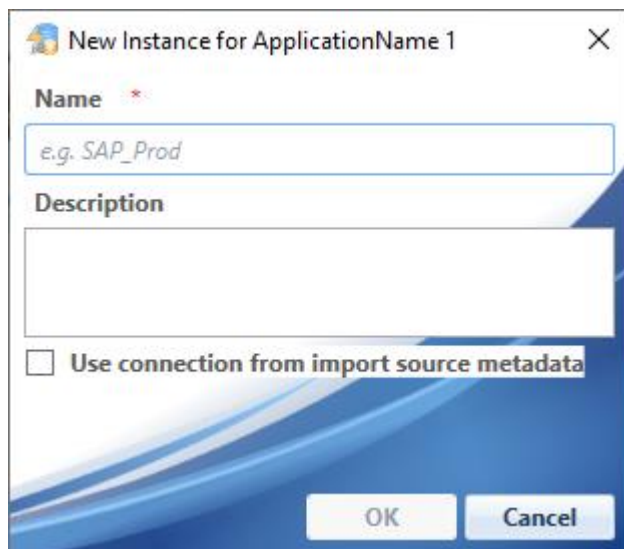
Create instance

Expand Configuration and right-click on Data Sources. Choose Add Instance and choose the desired version from the suggested data source versions. It will list out data source versions that are defined in Source Definition.



A data source must have destination metadata defined, i.e. table and column names for it to be configured and included, in order to create an instance.

When creating an instance, this window appears



Name: The instance name.

Description: Description of the instance.

Use connection from import source metadata: Enabling this will use the connection used to get the metadata, to load data from.

Connection string

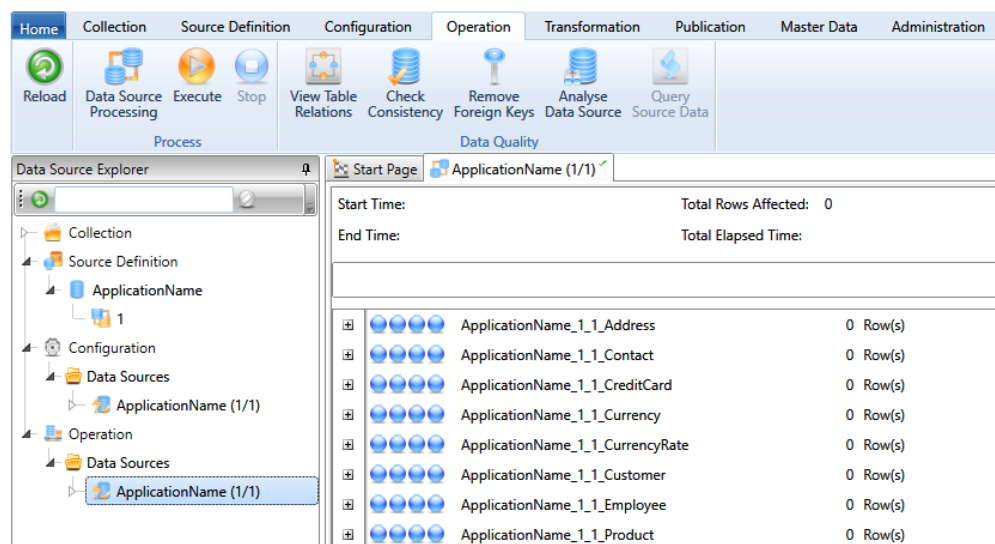
Right-click on your instance and choose Edit Connection or choose Edit Connection from the General tool group in the ribbon, or click on the connection icon in the properties window to configure the connection to your data source. This will bring up a dialog box where you can configure the connection to the source data. The connection set on the instance level will be inherited to all tables in that instance unless a specific table connection is set. A table connection is set by clicking on the table and edit connection.

2.3.3.3. Operation

Operation handles the execution of the actual data load. See section 2.4.5 for more details.

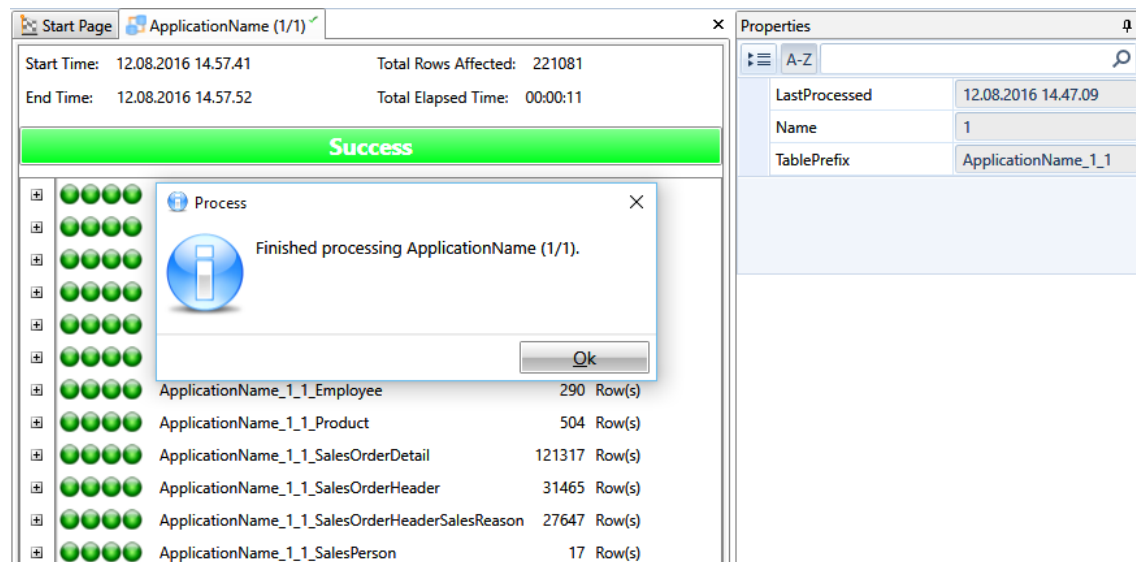
Process data source

Expand Operation and Data Sources to view the instances that are available for data processing. Choose Data Source Processing from a right-click on the instance or from the Process tool group in the ribbon. Click yes on the confirmation dialog to validate the data model quality and start loading data, or no to just perform validation.



Blue signals indicate that the data source is ready.

If you clicked no in the confirmation dialog, click Execute from the Process tool group in the ribbon to start loading data.



All green signals indicates success and the data has been loaded into the ODS database. The properties window shows the last successful process for that instance.

This completes the quick guide to set up a data source.

To schedule a data source import, please see section XBI_batch.exe for how to use and run the batch file.

2.4. Using Xpert BI Extraction – Advanced guide

2.4.1. License, Installation and Setup

2.4.1.1. Installation

The application install file is available from www.bi-builders.com/software/

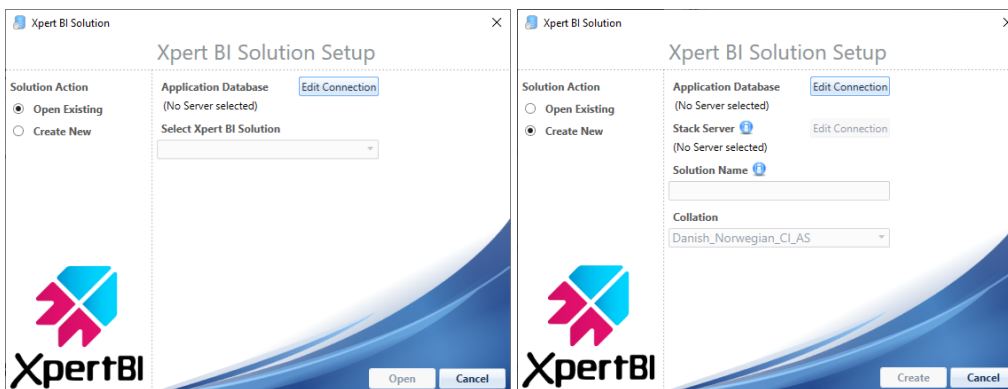
A password is needed when downloading the install file.

2.4.1.2. License

To obtain a valid license please contact BI builders (www.bi-builders.com for contact information)

2.4.1.3. Configure Xpert BI – Solution Setup

At first time start-up of Xpert BI you have to configure the connection installation details:



Solution Action: Choose whether to open an existing solution or create a new solution.

Open Existing

Application Database: The connection to the server where the Xpert BI management database is installed.

Select Xpert BI Solution: A list of available solutions found on the selected server.

Click the 'Open' button to open the selected solution. If the solution requires an upgrade, the button will have the text 'Upgrade' instead of 'Open'.

Create New

Application Database: The connection to the server where the Xpert BI management database is going to be installed. The management server contains all the configurations for the Xpert BI application.

Stack Server: The connection to the server where the database stack is going to be installed (ETL, ELM, ODS). If not changed, this server will be the same as the application database server.

Solution Name: This is the name given to the solution. It will become a prefix to the management database and the data warehouse stack databases.

Collation: The collation for all the solution databases. Xpert BI will suggest the server default collation. Only CI collations are supported.

Click the 'Create' button in the lower right corner to create the new Xpert BI Solution.

Note: It is recommended to set the collation to the same as the database server default.

Note: If you get a 'login error' when Xpert BI tries to create the databases, please view the Application Database/Stack Server connections and set the correct security.

Note: When creating a new solution and if the stack database/databases have to be created outside of Xpert BI, make sure to follow these rules:

Creating a stack on SQL Server, Managed Instance or similar:

- Database names must be :
 - {Solution Name}_ODS
 - {Solution Name}_ETL
 - {Solution Name}_ELM
 - {Solution Name}_STG
- Collation for all databases should match the one chosen in the setup.

Creating stack on Azure Synapse SQL, Azure SQL Database or similar:

- Database name must be "{Solution Name}_STACK".
- Collation should match the one chosen in the setup.
- Azure SQL Database should have additional setting: CATALOG_COLLATION = DATABASE_DEFAULT.

2.4.1.4. Configure Xpert BI – remote processing (SQL Server only)

The Xpert BI management database and the Xpert BI client can be located on two separate servers.

Both servers have the following prerequisites:

- Microsoft .NET Framework 4.7.2

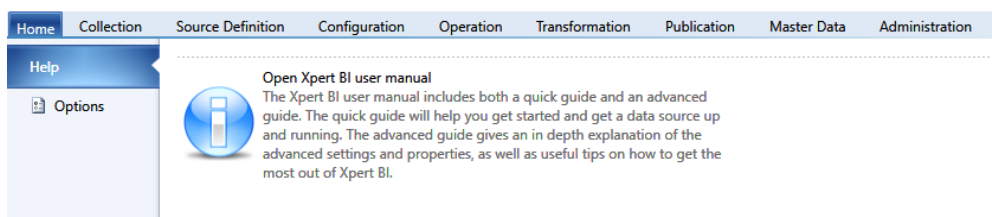
After Xpert BI installation on the client server/machine, copy the folder C:\Program Files\Xpert BI\ from the client server/machine to C:\Program Files\Xpert BI\ on the database server. This can be done manually or automatically through Xpert BI → Home → Options → Miscellaneous → Specify path, then Copy Batch Files.

Every time the Xpert BI client is updated to a new version, the files in C:\Program Files\Xpert BI\ **must** be copied to the database server to ensure the latest version of the software binaries.

2.4.2. Home and Global Buttons

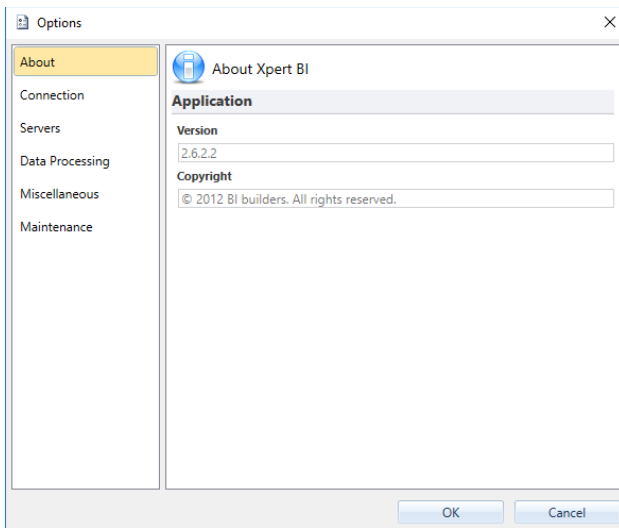
2.4.2.1. Help

The Home tab displays the Help section where you can find a link to the Xpert BI user manual.



2.4.2.2. Options

Xpert BI application settings are stored and configured in the Options window.



About

Displays the installed software version and copyright information.

Connection

Displays the server name and management database which Xpert BI is connected to. Please see Management database under Management Database for further information.

Servers

Displays the server name of the different databases in the solution.

Data Processing

- **Connection Timeout (in seconds):** Timeout when connecting to the data source database. The default is 30 seconds.
- **Command Timeout (in seconds):** Timeout when executing queries against the data source database. The default is 300 seconds.
- **Max number of threads when processing data:** The default maximum number of threads when processing data is 32 which means that 32 concurrent processes/threads can start if and when possible. This default can be overrun by the software as the number of threads or the number of I/O completion threads will not be a smaller than the number of processors in the computer.

Miscellaneous

- **Other Command Timeout (in seconds):** Timeout when executing commands not related to data processing. E.g. Check Consistency, Repair Consistency, View Table Relations. The default is 300 seconds.
- **Time Dimension Range:** Sets the **start year**, **end year** and **first day in week** for the time dimension generated by Xpert BI. The default settings are
 - Start: <Current Year> -7
 - End: <Current Year> +3.
 - First day in week: Mon

To change the time range for XBI_Time located in ODS, change the start/end date and click 'Update Time Dimension'

- **Copy Batch Files:** Copy the files required for running the Batch processing into a folder. UNC/Network folders can be used. The default is C:\Program Files\Xpert BI (install directory).
- **Today as initial date:** Change the initial date behaviour for tracking/changing. If enabled, use today's date as initial date on new rows when running tracking/changing. The default is disabled, initial date is set to start date of time dimension.
- **Auto Refresh Solution Explorer:** Enables automatic refreshing of views. When disabled the Solution Explorer will only refresh the structure. This applies only to startup and first run. When explicitly refreshing from the Solution Explorer and when migrating, a complete refresh will be performed regardless of this setting.

Note: It is recommended to disable this when in production environment and enabled in development environment.

Maintenance

Displays settings for logging history and displays the solution database prefix and collation configurations.

- **Keep logs in number of days:** Delete log entries older than the number of days. Will delete entries from the following logs: Event log, Performance log, Performance runs. The default is 10 days.

2.4.2.3. Global Buttons

The **Reload** and **Save** buttons are displayed in the upper left area of the ribbon on all tabs where relevant. These two buttons are global and does not have a tool group name. **Reload** will retrieve the latest saved information from the management database. **Save** will save your changes to the management database.

2.4.2.4. Data Source Explorer Filter/Search

The Data Source Explorer toolbar has a text field for filtering the items shown in the tree structure. The filter is case insensitive and is implemented as 'contains'. This filter can, for example be used to view all items in a source, or to search for a specific item by entering a part of the name.

To the right of the text field there is a Clear-button used to clear the filter text (pressing "Escape" does the same). If an item in the tree is selected when this happens, it will stay selected and the tree will be expanded down to it. If no item is selected (or the selected item is a top-level) all items will be collapsed.

2.4.3. Source Definition

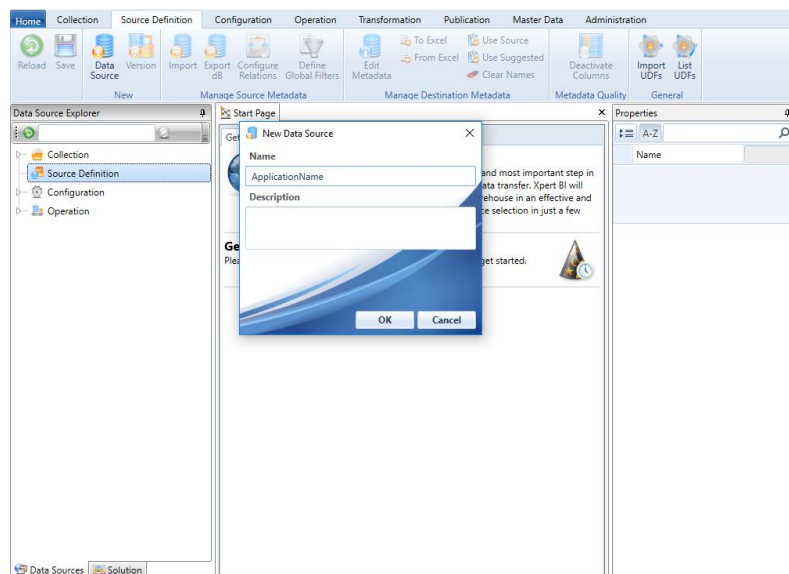
2.4.3.1. Add data source

Click on Source Definition and choose Add Data Source from the ribbon or from a right-click.

A window appears. Type in the name of your data source (required) and a description (optional)

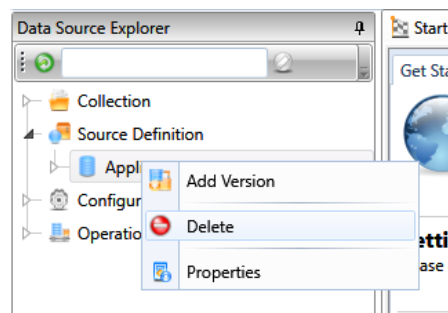
You will now find your data source listed in the solution explorer under Source Definition.

Note: You may not name your data source with the “XBI_” prefix as this is a reserved prefix in the Xpert BI application.



2.4.3.2. Delete data source

To delete a data source, right-click on the data source you wish to delete and choose Delete.

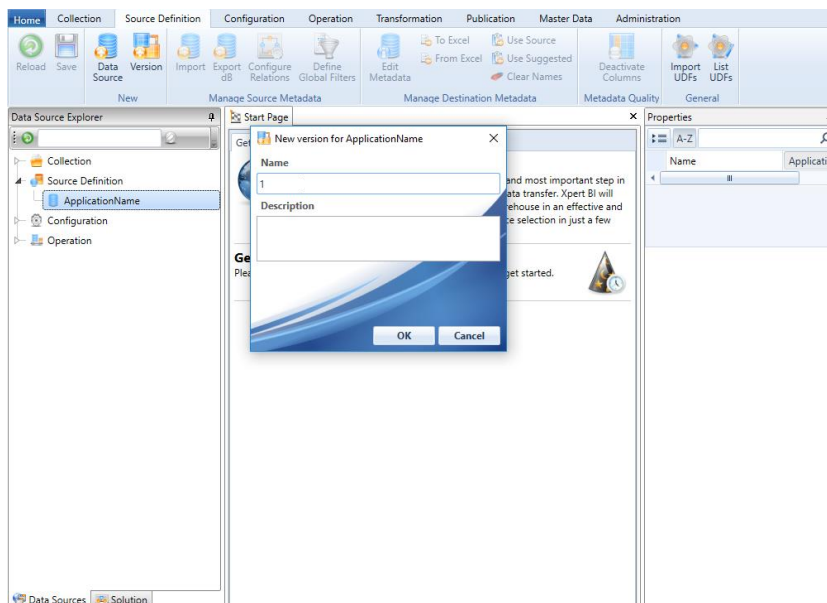


The delete will prompt for a confirmation of the deletion. Remember that deleting a data source is **irreversible** and will include deletion of all versions, instances, filters and other settings related to this data source.

Note: Tables in the ODS database are not deleted

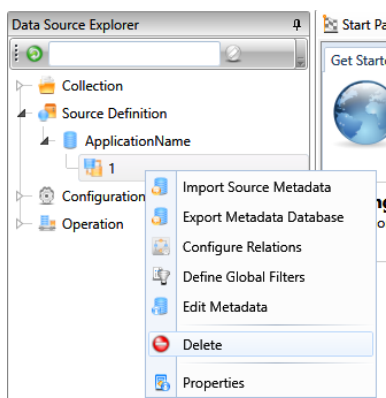
2.4.3.3. Add version

Click on your data source and choose New Version from the ribbon or from a right-click.



2.4.3.4. Delete version

Right-Click on the data source version you wish to delete and choose Delete to delete a data source version.

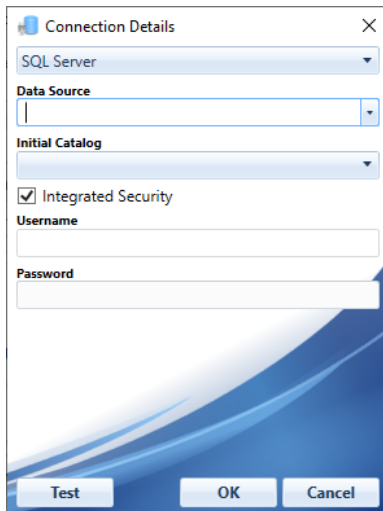


Remember that deleting a version is **irreversible** and will include deletion of all instances, filters and other settings related to this version.

Note: Tables in the ODS database are not deleted.

2.4.3.5. Import source metadata

Right-click on the version you want to define and choose Import Source Metadata, or choose Import from the Manage Source Metadata group in the ribbon. Click the Connection button to open the connection dialog box. Select the connection provider and fill out the connection string parameters. Click Test to test the connection, and OK to use the connection.



If your data source is SQL Server, Azure SQL Database, Azure Synapse SQL, Azure Managed Instance, Azure Data Lake, Oracle or DB2 you can connect directly to your data source and import metadata (table and column names and relationships). If your data source is another database type you must create a metadata database, i.e. table and column names, table primary key and relationships, in a supported structure (e.g. SQL Server) and import this metadata database as your source metadata. In addition to the native databases, several applications (such as SAP) are supported through the application adapters (license controlled).

Uncheck the Include Relations checkbox if the relations should not be a part of the metadata import.

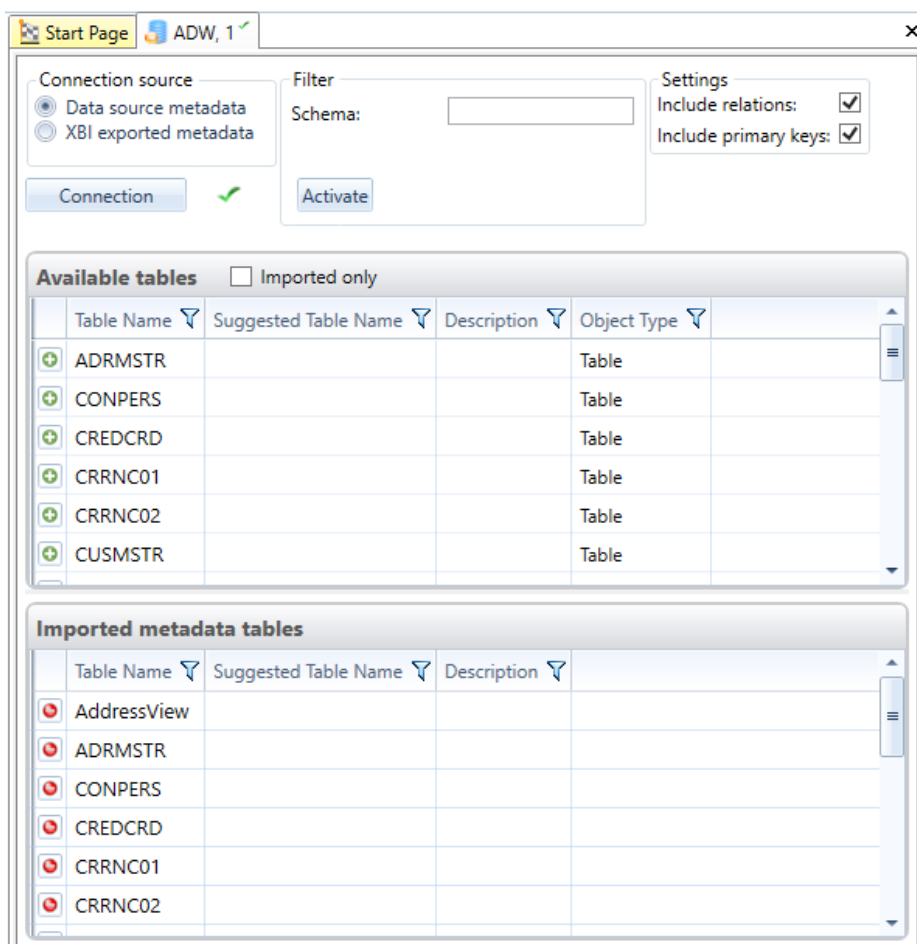
Uncheck the Include Primary Keys checkbox if the primary keys should not be a part of the metadata import.

If your data source has a specific schema, you can type the schema in the text box for Schema filter and click Activate.

If your application data source has a specific metadata version, you can type the version in the text box for Metadata Version filter and click Activate. E.g., SAP uses Metadata Version: 0.

For some application data sources it is possible to filter the Available tables according to available Kits. Use the Kit filter dropdown box to activate a filter.

Selecting XBI exported metadata (under Connection source) will change the connection to a database containing previously exported metadata and load from there. The Connection button and Filter section are disabled when XBI exported metadata is selected. The connection defined when Data source metadata is selected is saved when XBI exported metadata is selected and restored when Data source metadata is selected again. Enabling Data source metadata will enable the connection button as well as the filtering options. The Connection source settings are auto-saved.



You import/remove tables by using drag-and-drop between Available tables and Imported metadata tables, or by clicking the import or remove buttons. (You can ctrl-select and shift-select tables). Click Save to save the metadata.

The checkbox "Imported Only" will only list Available tables which are already imported. This can be useful on re-imports.

The Object Type column is used to filter between tables and views. This column is only available for SQL Server and Oracle data sources.

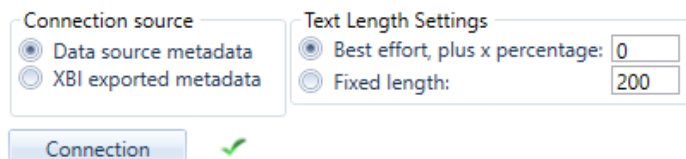
Note: Please see Export Metadata Database and Configure Relations for how to analyse the metadata.

Azure Data Lake

Xpert BI can import metadata from CSV or Parquet files stored in Azure Data Lake. It is expected that all the folders under the root folder represent one table each, and that each table has a consistent schema. The supported destinations when loading data from Azure Data Lake is Azure Synapse SQL or Snowflake.

Data Lengths (text) and Data Types in Azure Data Lake

When determining data types and lengths the first 1000 rows in the first file for CSV is read. For Parquet the first row- group from the first file is read. To override the default data length behaviour, choose to add a set percentage to the found data length, using the **Best effort, plus x percentage** option. This will read the first rows and then add x percentage to the maximum data length found. The other option is to set all data lengths to a fixed length.



| Available tables <input type="checkbox"/> Imported only | | | |
|---|------------|----------------------|-------------|
| | Table Name | Suggested Table Name | Description |
| | Address | | |
| | Contact | | |

Primary Keys

When the metadata is imported from Azure Data Lake, you can define your primary keys simply by checking the primary key column for the desired columns in **Edit Metadata** (after the metadata is imported).

2.4.3.6. Export metadata database

The metadata can be exported to a metadata database in your data warehouse for further refinement of metadata. The exported database will get a default name which you cannot change. The format of the name is: XBI_<prefix>_MD_<DatasourceName>_<InstanceName>. You can edit relationships in this database and then import it to Xpert BI again as described in Import Source Metadata.

Right-click on the version you want to export and choose Export Metadata Database, or choose Export dB from the Manage Source Metadata group in the ribbon. This will bring up a window displaying the name of the created metadata database.

Note: To fully utilise the functionality of Xpert BI, it is recommended that you spend some time to ensure a complete definition of your data source table relationships.

Note: Renaming a previously exported metadata database through SSMS, will result in a failure when exporting metadata database. This is due to the file and log names of the database. To back up previously exported metadata, do a regular 'backup' or 'script as' in SSMS.

Note: The exported metadata database will be created on the application database server.

2.4.3.7. Configure Relations

Some applications contain more relationship definitions than you require or relationship definitions that are not actually implemented or in use. Xpert BI allows you to deactivate relationships that are found through the source metadata import.

To deactivate relationships, click on your data source version and choose Configure Relations from ribbon or from the right-click menu. The relationships are listed both as PK i.e. “To” the table and FK i.e. “From” the selected table. Thus, all relationships are listed in both tables, and if deactivated in one of them, both tables are modified. You deactivate the relationship by deselecting the Is Active flag on the relationship in the bottom grid.

A comment column allows you to write comments on the relations.

Start Page

ADW, 1

Disable Unused Relations

| Table Name Source | Table Name | Modified |
|-------------------|-----------------------------|----------|
| > ADRMSTR | Address | |
| CONPERS | Contact | |
| CREDCRD | CreditCard | |
| CRRNC01 | Currency | |
| CRRNC02 | CurrencyRate | |
| CUSMSTR | Customer | |
| EMPMSTR | Employee | |
| PRDMSTR | Product | |
| SLSORDL | SalesOrderDetail | |
| SLSORDH | SalesOrderHeader | |
| SLSORDR | SalesOrderHeaderSalesReason | |
| SLSPPERS | SalesPerson | |

| Is Active | Direction | Current Table | Current Columns | Relationship | Related Table | Related Columns | Comment |
|---------------------------------------|-----------|----------------------------|--------------------|--------------|---------------------------------------|-----------------|-------------------|
| > <input checked="" type="checkbox"/> | From | SLSORDH (SalesOrderHeader) | FK_'ShipToAddr' | To | ADRMSTR (Address) | PK_'AddrID' | This is a comment |
| <input checked="" type="checkbox"/> | From | SLSORDH (SalesOrderHeader) | FK_'BillToAddr' | To | ADRMSTR (Address) | PK_'AddrID' | |
| <input checked="" type="checkbox"/> | From | SLSORDH (SalesOrderHeader) | FK_'Contact' | To | CONPERS (Contact) | PK_'ConID' | |
| <input checked="" type="checkbox"/> | From | SLSORDH (SalesOrderHeader) | FK_'CredCrdID' | To | CREDCRD (CreditCard) | PK_'CardID' | |
| <input checked="" type="checkbox"/> | From | SLSORDH (SalesOrderHeader) | FK_'CurrencyRate' | To | CRRNC02 (CurrencyRate) | PK_'CrrRateID' | |
| <input checked="" type="checkbox"/> | From | SLSORDH (SalesOrderHeader) | FK_'CustID' | To | CUSMSTR (Customer) | PK_'CustomerID' | |
| <input checked="" type="checkbox"/> | From | SLSORDH (SalesOrderHeader) | FK_'SalesPersonID' | To | SLSPPERS (SalesPerson) | PK_'SlsPersID' | |
| <input checked="" type="checkbox"/> | From | SLSORDH (SalesOrderHeader) | FK_'Territory' | To | SLSTERR (SalesTerritory) | PK_'SlsTerrID' | |
| <input checked="" type="checkbox"/> | To | SLSORDH (SalesOrderHeader) | PK_'SalesOrderID' | From | SLSORDL (SalesOrderDetail) | FK_'SlsOrdHID' | |
| <input checked="" type="checkbox"/> | To | SLSORDH (SalesOrderHeader) | PK_'SalesOrderID' | From | SLSORDR (SalesOrderHeaderSalesReason) | FK_'SlsOrdHID' | |

Note: A reason why you would want to deactivate relationships instead of exporting a metadata database and deleting them is to avoid importing them again later if you decide to import metadata from your data source again later.

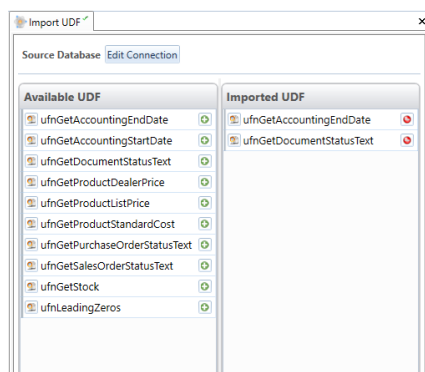
Clicking **Disable Unused Relations** will find and disable any unused relations. This function uses data in ODS, so the data source must be processed before using it. If there are multiple instances of a source, all instances are searched (and therefore needs to be processed beforehand). A relationship must be unused in all instances to be disabled by this function.

2.4.3.8. Import UDFs

User Defined Functions (UDFs) are used if you want to change the destination data type (e.g. “IntToDate”) or if you want to apply a function to a column to change the content of the data. To apply a function, it must first be imported into Xpert BI from an available SQL server database. Note that UDFs are not supported when extraction destination is Azure Synapse SQL, Azure Data Lake or Snowflake.

To import User Defined Functions that you want to apply to your data source, click on any node in the Source Definition tree and choose Import UDFs from the General tool group in the ribbon.

Start by selecting the database to import from by clicking ‘Edit Connection’. Import/remove UDFs by using drag-and-drop between ‘Available UDFs’ and ‘Imported UDFs’, or by clicking the import or remove buttons. (You can ctrl-select and shift-select tables). Click Save to save the UDFs.



Note: You may not name your UDFs with the “XBI_” prefix as this is a reserved prefix in the Xpert BI application.

2.4.3.9. List UDFs

To display the list of User defined functions you have imported, click on List UDFs from the General tool group in the ribbon. Click on the Code icon to view the code. Click on the Description icon to view or edit the description.

Note: You will not be able to edit the XBI functions.

2.4.3.10. Edit destination metadata

Xpert BI allows you to rename your tables and columns to more user-friendly names that will be the table and column names in your data warehouse. The Manage Destination Metadata tool group in the ribbon contains various ways of editing destination metadata.

- **Edit Metadata (available from ribbon, right-click menu, or single left-click on version)**

You can type in new table and column names in the grid in Xpert BI. Click on your version and choose a table in the grid to display the columns.

- **Import/Export to Excel**

You can edit the table and column names in excel. The To Excel and From Excel buttons in the Manage Destination Metadata group in the ribbon will export and import the metadata to and from Excel. You should not add or delete rows or columns in the Excel sheets. To add or remove a source table or column, use Import source metadata.

- **Use Source**

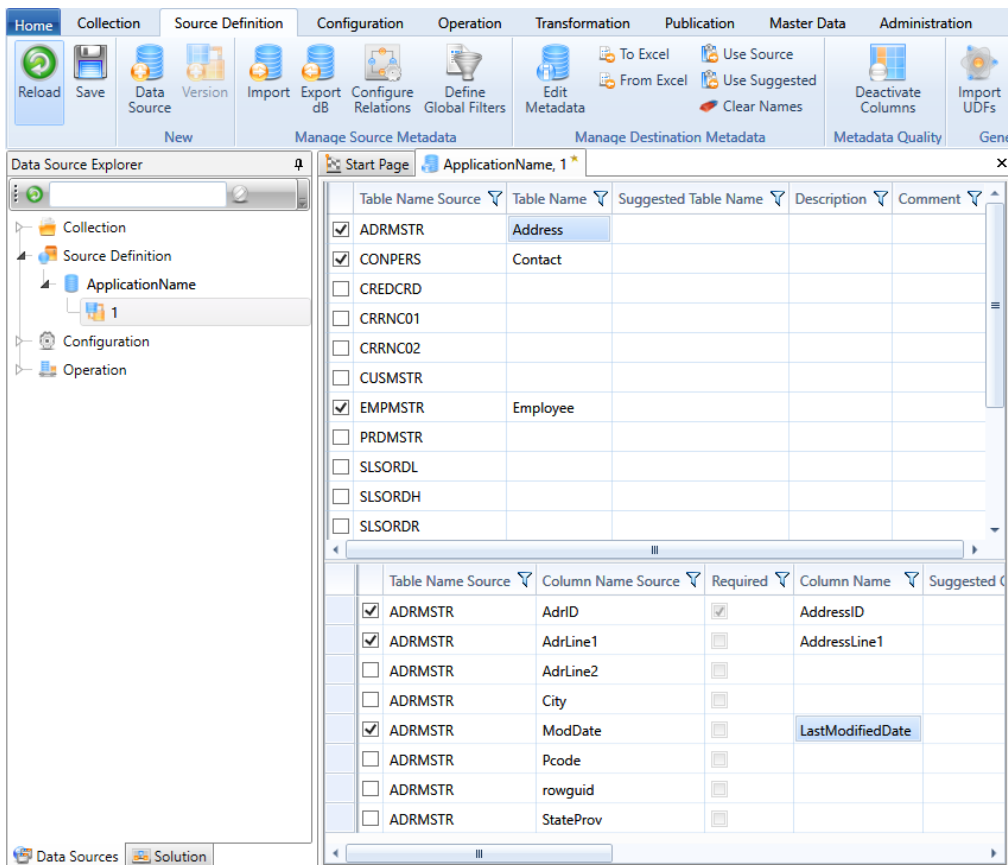
This option is used when you want to keep your database table and column names. Applies to all tables and columns

- **Use Suggested**

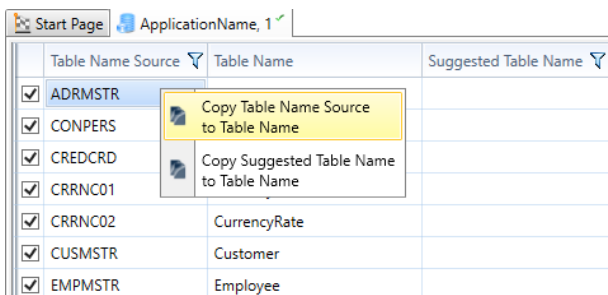
This option is available to pre-defined data sources (Solution Modules). Applies to all tables and columns

- **Clear Names**

This option clears all the table and column names



In addition to the options from the ribbon, you can also select and then Right-Click on a source name, both in the table and column area. The right-click options lets you copy source or suggested on table and/or column level.



Click Save to save your destination metadata.

Note: The left checkboxes are the selection criteria for which tables and columns to include in the data load from the data source. Tables and/or columns that are unchecked will not be processed.

Note: Exporting the destination metadata to excel can be useful not only to do edits but to have a backup/documentation of the mapping.

Note: If the suggested column names for a table contains duplicates, they will automatically be made unique by appending the Column Name Source to it.

Fields in the table grid

| Column | Description | Editable |
|----------------------|--|----------|
| Left checkbox column | Decides if this table is active (will be processed) | Yes |
| Table Name Source | The original table name | No |
| Table Name | The new table name | Yes |
| Suggested Table Name | Available on pre-defined data sources | Yes |
| Description | A column for describing the table content (optional) | Yes |
| Comment | A column for comments regarding the table (optional) | Yes |

Fields in the column grid

| Column | Description | Editable |
|-----------------------|---|----------|
| Left checkbox column | Decides if this column is active (will be processed) | Yes |
| Table Name Source | The original table name | No |
| Column Name Source | The original column name | No |
| Required | Will be checked when the field is either a primary or a foreign key and therefore a required field to include | No |
| Column Name | The new column name | Yes |
| Suggested Column Name | Available on pre-defined data sources | Yes |
| Function Name | A column for adding a function from a drop-down list to a column. Please see Import UDFs for how to import functions (optional) | Yes |
| Destination Data Def | The data type of the destination. Apply UDF to change destination data type | No |
| Source Data Def | The data type of the source | No |
| Description | A column for describing the column content (optional) | Yes |
| Primary Key | Is checked if the field is a primary key | No* |
| Foreign Key | Is checked if the field is a foreign key | No |
| Comment | A field for comments regarding the column (optional) | Yes |

* Is editable if data source is Azure Data Lake.

Note: Foreign key columns only require a new name when the related table is included in the data load.

Deactivate Single-Valued Columns

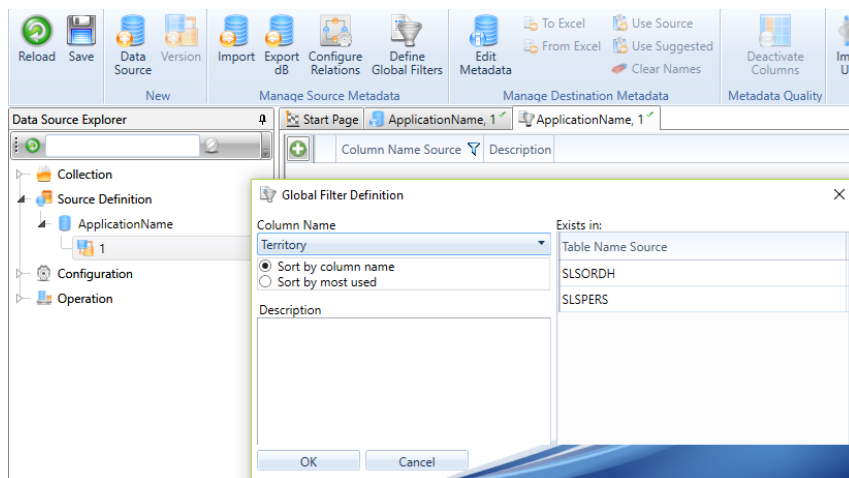
When clicking this button, Xpert BI will find all columns containing only a single value. Null is considered a value, but the first row in each table (the null-row) is skipped. This function uses data in ODS, so the data source has to be processed before using it. If there are multiple instances of a source, all instances are searched (and therefore needs to be processed beforehand). A column has to be single-valued in all instances to be a true single-valued column.

If a column has a SQL datatype that is not comparable (for example geometry), it will be skipped in the search for single-valued columns. The search runs table by table, so if it for some reason fails on one table, it will skip it but continue with the rest of the tables.

The columns found are displayed in a window, where you can deselect any columns that should not be deactivated. Clicking OK will deactivate the selected columns. Click *Save* to save the changes.

2.4.3.11. Define Global Filters

Right-click on your version and choose Define Global Filters or choose Define Global Filters from the Manage Source Metadata tool group from the ribbon to define the columns on which you wish to set a global filter. Click on the Add Filter button (green “+” button) to define a filter column. A definition window appears where you can choose a column in your data source from a dropdown list. The dropdown list can be sorted either ascending by column name or descending by most used. When you choose a column, the tables in which the column name exist will be displayed in “Exists in” list.



The Global filter columns are defined in Source Definition and set (i.e. given a value) in Configuration.

Note: A global filter will normally be a column that is included in all or most tables and is a part of the primary and foreign keys. If you are only interested in one specific value of this column you can use the Global Filter functionality in Xpert BI to set one value in every table, and the column does no longer need to be a part of the primary and/or foreign key definition which makes for a simpler data model.

Note: Global Filters are not supported when the source is Azure Data Lake.

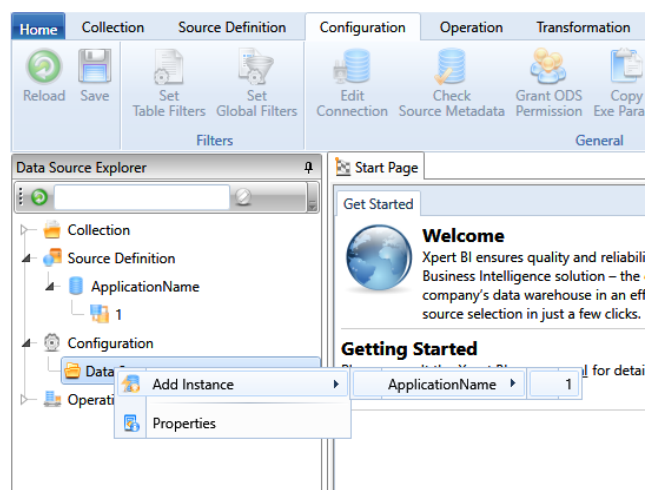
2.4.3.12. Properties

By clicking on the object, properties for the object will show in the properties panel on the right.

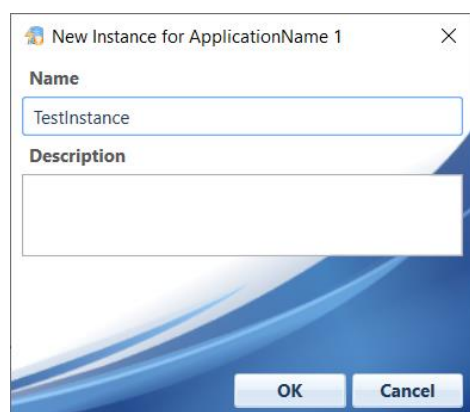
2.4.4. Configuration

2.4.4.1. Add Instance

Expand Configuration and right-click on Data Sources. Choose Add Instance and choose the desired version from the suggested data source versions. It will list out data source versions that are defined in Source Definition. A data source must have destination metadata defined for it to be configured.



Please name your instance in the windows that appears.



2.4.4.2. Delete instance

Right-Click on the instance you want to delete and choose Delete. You have to confirm your deletion in a dialog box before your instance is permanently deleted.

Remember that deleting an instance is **irreversible** and will include deletion of filters and other settings related to this instance.

Note: Tables in ODS are not deleted.

2.4.4.3. Table Filters

Right-click on your instance and choose Set Table Filters or click on the Set Table Filters icon from the Instance tool group from the ribbon. You can now set data selections per table.

There are two types of rules; Incremental and Source Selection.

Note: Table filters are not supported when the source is Azure Data Lake.

Incremental Column

Choose the column from the dropdown list on which to define an incremental rule. Data will now be loaded based on the '>' sign and the value of this column on last imported row. If you want to import overlap, you can specify how much overlap you want.

Source Selection

Click on the 'Edit Source Selection' icon to set other data selections to limit the data extraction. This will bring up an expression builder where you can choose columns from the selected table and set the selection. When double-clicking on a column the column will appear in the Expression with a default of equal sign (=) next to it. From there you can edit the expression to create the desired table selection.

| | Table Name Source | Table Name | Incremental Column | Last Value | Overlap | Source Selection | Consolidated Selection |
|---|-------------------|-----------------------------|--------------------|------------|---------|------------------|------------------------|
| > | ADRMSTR | Address | > | | 0 | | |
| | CONPERS | Contact | > | | 0 | | |
| | CREDCRD | CreditCard | > | | 0 | | |
| | CRRNC01 | Currency | > | | 0 | | |
| | CRRNC02 | CurrencyRate | > | | 0 | | |
| | CUSMSTR | Customer | > | | 0 | | |
| | EMPMSTR | Employee | > | | 0 | | |
| | PRDMSTR | Product | > | | 0 | | |
| | SHIPMET | ShipMethod | > | | 0 | | |
| | SLSORDH | SalesOrderHeader | > | | 0 | | |
| | SLSORDL | SalesOrderDetail | ModifiedDate | > | | | |
| | SLSORDR | SalesOrderHeaderSalesReason | > | | 0 | | |
| | SLSPERS | SalesPerson | > | | 0 | | |
| | SLSREAS | SalesReason | > | | 0 | | |
| | SLSTERR | SalesTerritory | > | | 0 | | |
| | SPECOFF | SpecialOffer | > | | 0 | | |
| | SPECOFP | SpecialOfferProduct | > | | 0 | | |

You can reset the Last Value by clicking the Reset button or by choosing None from the incremental column. To clear the incremental and source selection filters from a table, click the red clear button at the start of the row.

Consolidated Selection

The consolidated selection will show the entire 'where' clause that will be used for each table, including the global filters if any.

Note: Table filters are used to load only a selection of the data in a table, or to set incremental update on the data. Incremental rules and table selections can only be set on columns from the chosen table, not across tables.

2.4.4.4. Global filters

Right-Click on your instance in Configuration and choose Set Global Filters or choose Set Global Filters from the Instance tool group in the ribbon. A list of your defined global filters will appear and you can set the desired value in the Filter Value column.

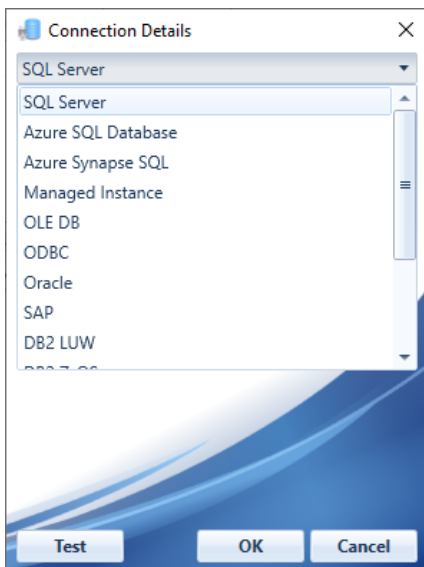
The Comparator dropdown lets you select the comparator used when comparing with the value in Filter Value.

You have the option of excluding tables from the global filter by clicking Exclude table in the bottom grid.

2.4.4.5. Edit Connection (instance and table level)

Right-click on your instance and choose Edit Connection or click on the connection icon in the properties window. This will bring up a dialog box where you can configure the connection to the source data. The connection set on the instance level will be inherited to all tables in that instance unless a specific table connection is set. Table connection is set by right-clicking on the table and choosing Edit Connection or by clicking the connection icon from the properties window.

The connection string can be edited manually by typing directly into the property text box. Password is not shown in the property text box, and cannot be edited. A blank connection string must be edited through the connection string dialog box (Edit Connection).



Below are more details on how to connect to different database types.

SQL Server

When your data source is SQL server, choose SQL server from the dropdown menu. Fill in server name and instance in Data Source, and fill in database name in initial catalog. Choose either integrated security or fill in username and password.

Azure SQL Database, Azure Synapse SQL, Managed Instance

These work the same way as SQL Server.

OLE DB

When you want to connect using OLE DB, choose OLE DB from the dropdown menu.

ODBC

When you want to connect using ODBC, choose ODBC from the dropdown menu.

Fill in the DSN that was used when setting up the ODBC connection through the Microsoft ODBC Data Source Administrator.

Oracle

When your data source is Oracle, choose Oracle from the dropdown menu.

SAP

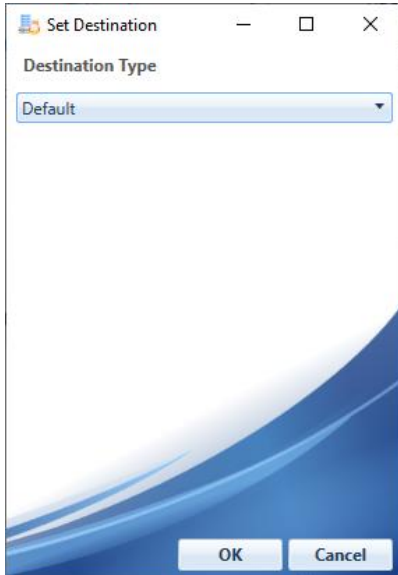
When your data source is SAP, choose SAP from the dropdown menu. Xpert BI typically uses components from the BizTalk adapter pack to connect to SAP data sources.

Azure Data Lake

When your data source is Azure Data Lake, choose Azure Data Lake from the dropdown menu. Fill in the Connection String, choose the Blob Container and optionally add a Root Path (the path to the folder which holds all the “tables”). The CSV Settings only apply to CSV files in the source.

2.4.4.6. Set Destination

The default destination for ODS (where your data is loaded to) is in your stack. If the stack is schema based the data will be loaded into the stack database in the “ods” schema, otherwise it will be loaded into the ODS database. Right-click an instance and choose **Set Destination** to open a window (as seen below) where you can change the destination. If the destination is SQL Server, Azure Synapse SQL or Azure SQL Database and the destination database name is not already added, you will get an option to add the destination database as an additional database when you click OK.



Other than the default, the following are available:

SQL Server and Azure SQL Database

Choosing this will load the data into the database configured in the connection window. Destination tables are created in the “ods” schema and the staging tables are created in the “stg” schema.

Azure Synapse SQL

Choosing Azure Synapse SQL as your destination lets you either load the data directly into Synapse using SQL Bulk Copy, or to copy it via Azure Data Lake. If you choose to copy via Data Lake, the data will first be loaded into the Data Lake as your chosen file type and then loaded into Synapse using Copy statements.

The destination tables in Synapse will not have implemented primary keys, but source tables are still required to have one. We use the primary key defined in the metadata to create a hash-based Id-column with the name <TableName>_Id. There are also no foreign keys defined on the tables, but we create a hash-based foreign key Id-column based on the foreign key metadata.

Destination tables are created in the “ods” schema and the staging tables are created in the “stg” schema. If these schemas do not exist, they are created. All tables use round-robin distribution option.

Azure Data Lake

Choosing Azure Data Lake as your destination will load data from the source into your configured Data Lake as files. The file type is configurable.

Snowflake

Choosing Snowflake works similar to Azure Synapse SQL, except the only option is to load data via Azure Data Lake. To load data from the data lake into Snowflake, we need a storage integration. This can be created using option 1 found here: <https://docs.snowflake.com/en/user-guide/data-load-azure-config.html>.

Notes on Set Destination

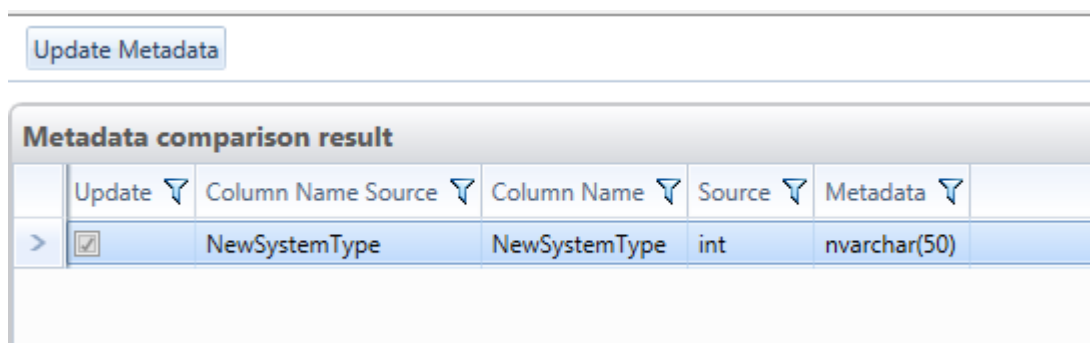
When using Analyse Data Source with destination set to Snowflake or Azure Data Lake, ODS Size (MB) in the Table Information analysis is not valid.

Note that when source is set to Azure Data Lake, it is not possible to set destination to Azure Data Lake. In addition, copy method for Azure Synapse destination is not configurable.

2.4.4.7. Check Source Metadata

Right-click on your instance table and choose Check Source Metadata or choose Check Source Metadata from the General tool group in the ribbon. If the source metadata and the metadata defined in Xpert BI are identical, “Source is identical” is displayed.

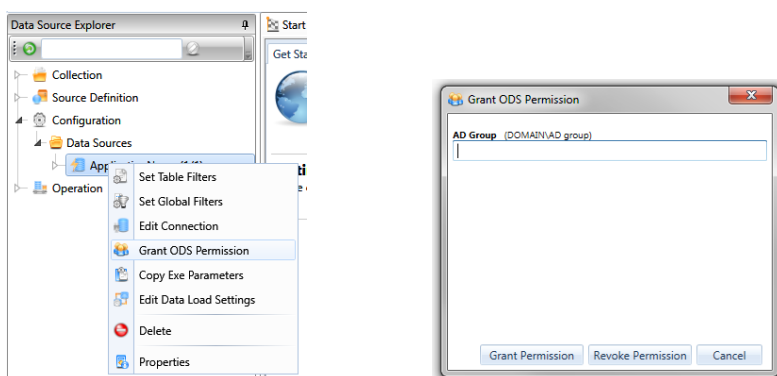
Any metadata differences are listed in the Metadata comparison result pane. Click the “Update Metadata” to update the stored metadata. Remember to run “Check Consistency” when finished to make sure the metadata change is propagated further on in the stack. Uncheck the Update checkbox for any entries which should be left out when doing a metadata update.



Note that this is not supported when the data source is Azure Data Lake.

2.4.4.8. Grant ODS permission

Right-click on your instance and choose Grant ODS Permission or choose Grant ODS Permission from the General tool group in the ribbon. Type in the domain and Active Directory group name that you want to grant read permission to the ODS tables of this instance. Only one group can have access to one instance. If you would like to grant access to more than one group, this must be handled in Active Directory by creating a new group.



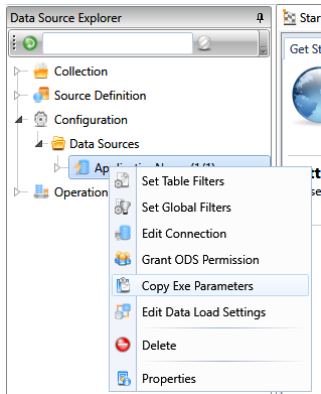
The Grant ODS permission will always show you what group (if any) has access to the tables of the instance in this dialog box. By overwriting the group name, the initial group loses permission and the new group is granted permission.

The permission privileges are implemented when “Grant Permission” is clicked. If any new tables are added to the instance the “Grant Permission” has to be executed again to give permission to the new tables. Use the “Revoke Permission” to revoke the permission for the specified AD Group.

Note that this is only supported on SQL Server.

2.4.4.9. Copy exe parameters (instance and table level)

Right-click on your instance in Configuration and choose Copy Exe Parameters or choose Copy Exe Param from the Instance tool group in the ribbon to copy the parameters that will be used in execution of your data source instance. The parameters are then copied to the clipboard.



Note: The parameters will be used when you set up regular processing of your data source either from a script, SSIS dtsx package, SQL server Agent or another application.

2.4.4.10. Edit Data Load Settings

Right-click on your instance in Configuration and choose Edit Data Load Settings or choose Edit Data Load Settings from the Instance tool group in the ribbon to manage the data load setting for each table in an instance. The available data load settings depend on the destination type for the selected instance. If the destination is Azure Data Lake, Edit Data Load Settings is not available.

ADW (1/1)

| | Table Name Source | Table Name | Update Only If Changed | Truncate Before Update | Insert Only | Skip Duplicates |
|---|-------------------|--------------|--------------------------|--------------------------|--------------------------|--------------------------|
| > | ADRMSTR | Address | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> |
| | Company | Company | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> |
| | CONPERS | Contact | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> |
| | CREDCRD | CreditCard | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> |
| | CRRNC01 | Currency | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> |
| | CRRNC02 | CurrencyRate | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> |
| | CUSMSTR | Customer | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> |
| | EMPMSTR | Employee | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> |
| | Person | Person | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> |

The settings Update Only If Changed, Truncate Before Update and Insert Only are mutually exclusive, it is only possible to check one of the checkboxes for the same table.

Update Only If Changed

Checking this setting will only update rows when data has changed. This will have impact on the UID_LastChanged timestamp as the row will not get an updated timestamp if no changes on the row values are detected.

Truncate Before Update

Checking this setting will truncate the table before it starts the Extraction. It will not truncate the surrogate table.

Insert Only

Checking this setting will ignore all update statements and only load the inserts.

Skip Duplicates

Checking this setting will ignore duplicate rows in the staging table when loading. The ignored rows are moved to a log table in the staging database, with the naming convention TableName_Deleted. The table is reloaded every time duplicate rows are detected.

Processing Priority

This is set on table level and is a property for specifying prioritization on tables i.e. setting the processing order for tables. This will not overwrite table dependencies but is used in special cases where one or more, typically large tables are more important or less important than other tables. The order range is 1-5 where 3 is default.

Table Distribution

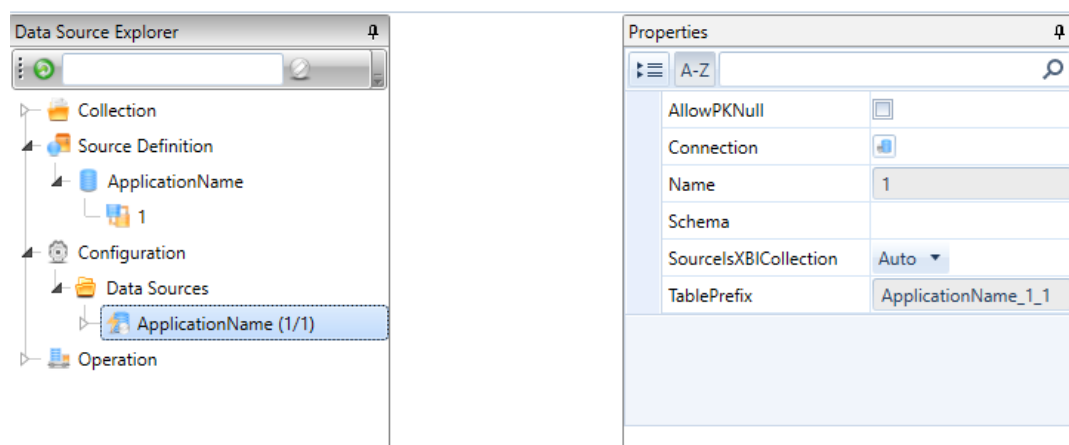
This allows you to change the distribution of a table.

Hash Distribution Column

If the table distribution is set to **Hash** then at least one column must be selected as the hash column.

2.4.4.11. Instance and Instance Table Properties

By clicking on a Data Source Instance the following properties will appear:



AllowPKNNull

This property is checked if your data source can contain a NULL value in the defined PK. Xpert BI will then replace the NULL value with either a 0 for numeric/integer fields, <blank>/' in character fields, or 19000101 for date related fields.

Connection

The connection string will be visible in this property. If no connection is visible on table level, it will inherit connection from Version level. The button links to the Edit Connection window. See section Edit Connection for how to edit connection.

Name

Displays the name of your instance or table.

Schema

If your data source instance has a specific schema to be used in connection, this can be specified in the Schema property. The schema set on the instance level will be inherited to all tables in that instance unless a specific table schema is set.

SourceIsXBICollection This property is used by Xpert BI when dependencies are generated between Extraction/Data sources and Collections. The default is Auto, in which case Xpert BI will try to match by naming. If a relation is created that should not be, you can set this to No. Setting this to Yes will in most cases lead to the same results as Auto but can be set for documentation purposes.

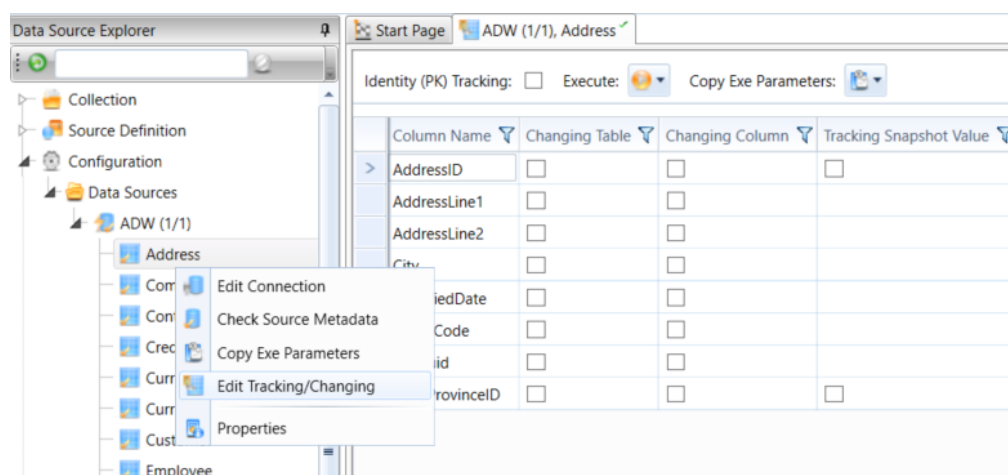
TablePrefix

Display the TablePrefix that will be used for all tables of this instance in the ODS database.

2.4.4.12. Tracking table values or attribute changes

The Tracking/Changing functionality in Xpert BI is set on table level and is found in Configuration in the Data Source Explorer. Right-click on a table to bring up the pane for the Tracking/Changing configuration. Xpert BI provides four different methods to track changes to a table.

Right-click on a table in Configuration to configure tracking/changing settings for that table.



Track Changing Table (CT)

Select the attribute(s) you want to monitor for changes. One new table, which holds all attributes, will be created in the ODS database, and new records will be added to the table when any of the selected attributes changes. This setting is generally used for dimension-type tables.

Track Changing Column (CC)

Select the attribute(s) you want to monitor for changes. One new table will be created in the ODS database for each of the selected attributes, and new records will be added to the table when that attribute changes. This setting is generally used for dimension-type tables.

Tracking Snapshot Value (TS)

This option is for tracking snapshot data. Select the value(s) you want to track and one new table will be created. Xpert BI will create transactions for each snapshot in this table by inserting both the snapshot value and a delta/diff column which calculates the difference between each snapshot. This setting is generally used for fact-type tables.

Identity (PK) Tracking (TI)

This option is for tracking number of records/primary key and creating timestamps for added and deleted occurrences. This setting is generally used for fact-type tables.

To schedule tracking/changing, click on the 'Copy Exe Parameters' button. This will open a dropdown where you can select desired tracking/changing type. After selecting a type the parameters are copied to the clipboard. Schedule the batch process from any scheduler.

To execute a tracking/changing type from the GUI, click the 'Execute' button and select the desired type from the dropdown.

Modifying or deleting a tracking/changing column must be resolved manually in the ODS tracking/changing table. This is due to the complexity of this type of functionality.

Note: The lowest time granularity for scheduling the tracking/changing updates once per day. Xpert BI does not support configuration of intra-day tracking.

Note: Tracking/Changing tables may become very large over time if you have large data sets and run the tracking/changing updates frequently. It is recommended to run these updates at the interval which is relevant for reporting (e.g. monthly/weekly/daily) and not necessarily as often as you regular data updates.

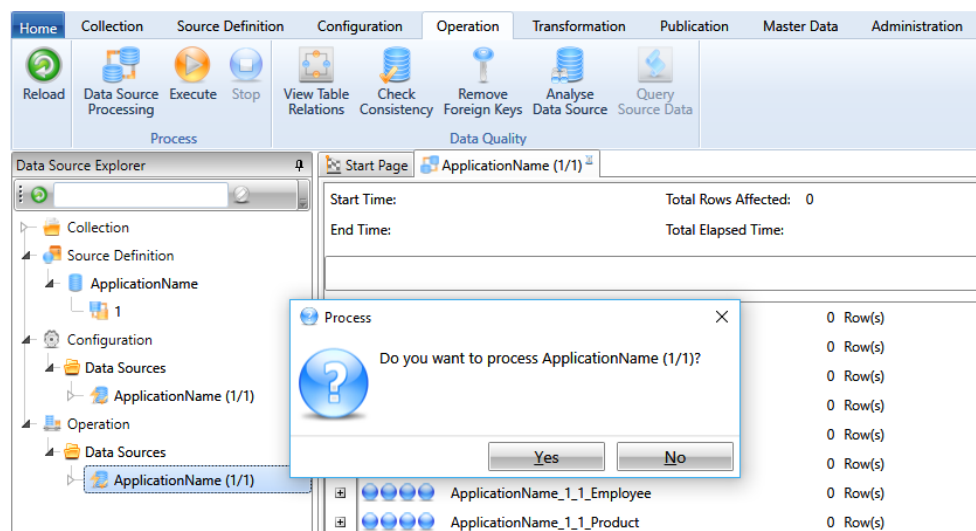
Note: This is supported on SQL Server and Azure SQL Database.

2.4.5. Operation

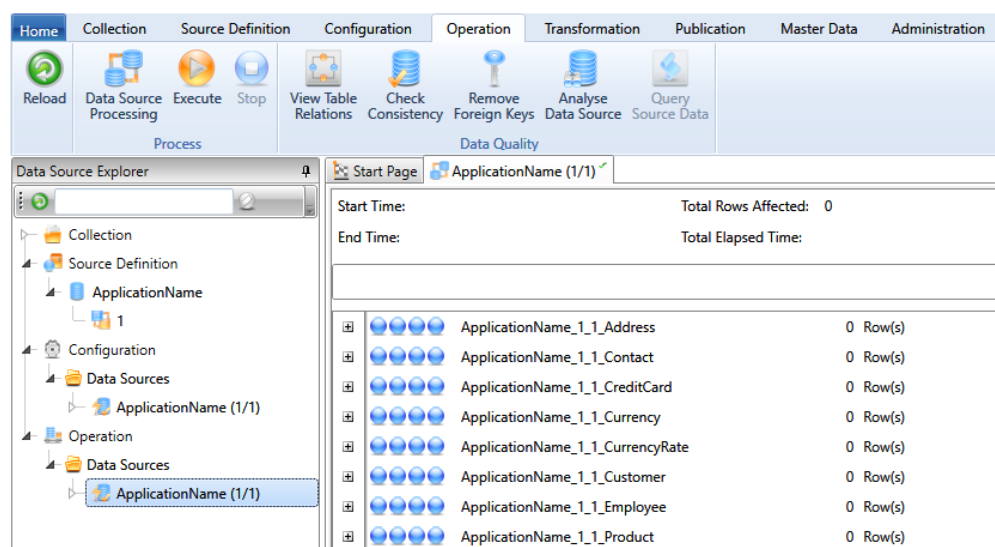
2.4.5.1. Process data source

Instance Level

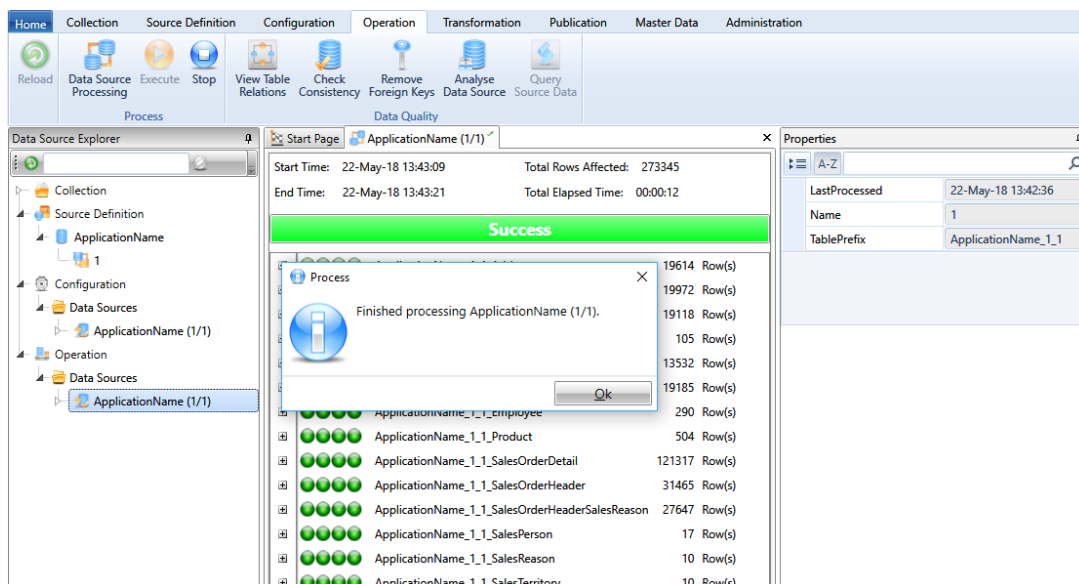
Right-click the data source instance under configuration and choose Data Source Processing or select an instance and choose Data Source Processing from the Process tool group in the ribbon to begin processing your data source. This will open a confirmation dialog asking if you want to process.



Clicking yes will start a validation process that Xpert BI does to ensure data model quality and process the data source thereafter. If you click no, only the validation process will be run. The validation includes checking for primary keys, duplicate table names, duplicate column names, and missing column names. If the validation process returns an error, you must resolve the issue(s) to process your data successfully. If your data model is successfully implemented, and you have just blue status lights the data loading can start.



If you clicked no in the initial confirmation dialog, you can click the Execute button from the Process tool group in the Ribbon to start loading data. The different process steps are illustrated as green when successfully finished or red if an error occurred.



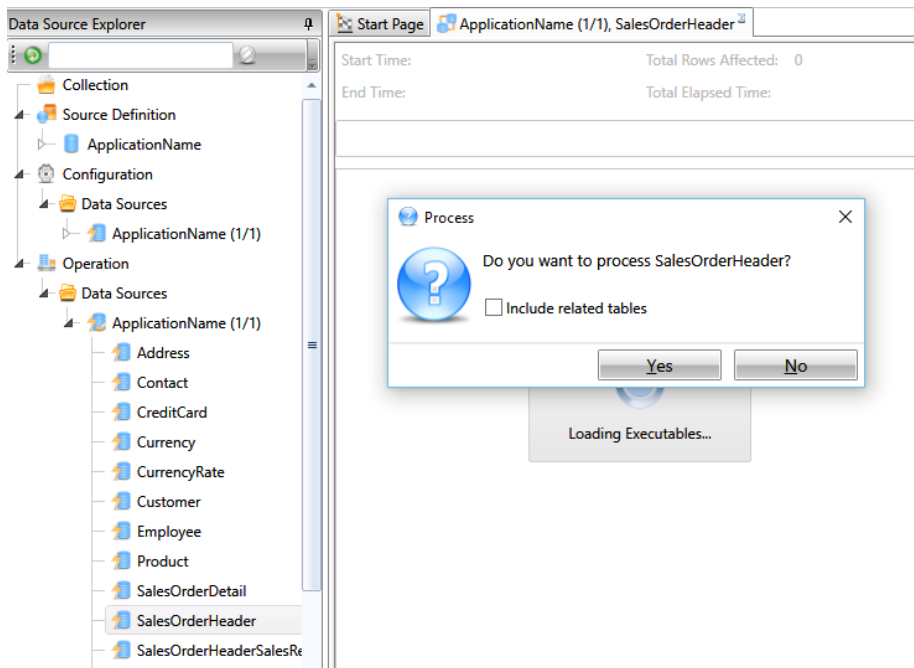
To view the details of the process step, click the '+' sign to expand the process into the process steps.

If one or more processing tasks fail, you will see red signals. If you experienced an error, you will see exactly which process failed and why it failed. You can also open the Event Log under Administration to view the error details.

Click the Stop button from the Process tool group in the Ribbon to stop loading the data. This will prevent any new task from starting and stop the already running tasks.

Table Level

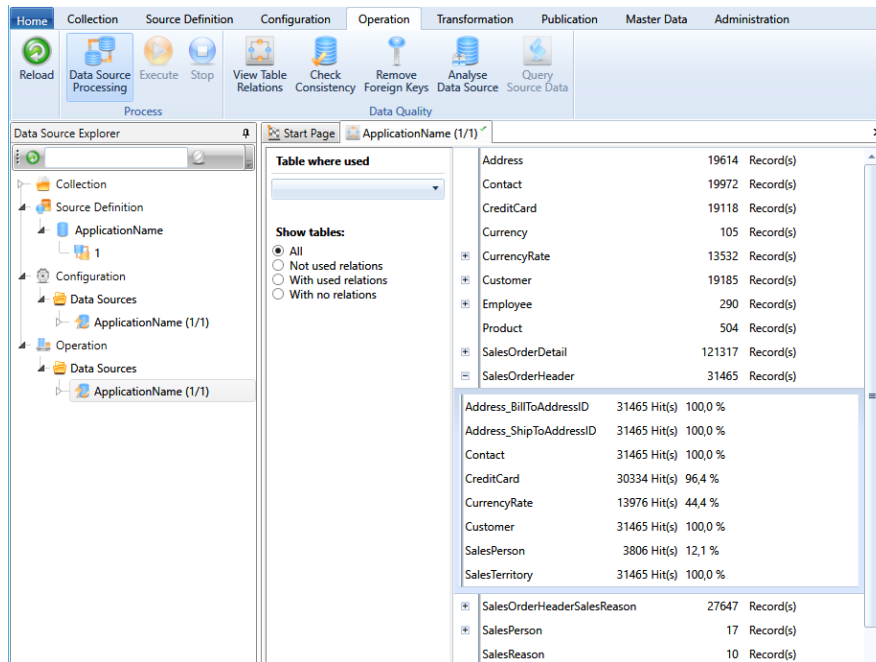
Right-click the table and choose Table Processing or select a table and choose Data Source Processing from the Process tool group in the ribbon to begin processing your table. Table processing runs the same validation checks as those run on instance level. The steps of Execute and Stop are also the same as on instance level. The confirmation dialog asking if you want to process has a "Include related tables" checkbox. Checking this (and clicking yes) will process the selected table and all its related tables. If you want to check what those tables are before processing, you check the checkbox and click "No", followed by "Execute" from ribbon if you want to process the listed table(s).



Note: When stopping a data load you should check your incremental rules. The last loaded value may or may not be updated depending on when the process was stopped.

2.4.5.2. View Table relations

Right-click the data source instance under configuration and choose View Table Relations or select an instance and choose View Table Relations from the Data Quality tool group in the ribbon to view the quality of the defined relationships and tables in your data source. The View Table Relations window will list out the tables in your data source in a tree structure. Tables which have defined relationships with other tables are indicated with a '+' sign for you to expand. Clicking the '+' sign will list the related tables with a percentage. This percentage shows the quality of the relationships and is defined as the number of records that finds a match in the related table of the total number of records. This functionality is not available when data is loaded into Azure Data Lake or Snowflake.



Note: you need to process the data before you can use the table relations functionality.

You have five view options when viewing tables and their relationship quality:

Tables where used (dropdown)

When a table is selected from the dropdown list, the related table(s) – if any – will be listed.

Note: This view option can be used if you plan to change or delete a table from your solution, and by selecting it you can view dependencies, i.e. the tables that have a defined relationship with this table.

All (radio button)

This will list all tables with the number of records in each table.

Not used relations (radio button)

This will list all tables that have no valid relationships with other tables where a relationship is defined.

With used relations (radio button)

This will list all tables that have records and valid relationships with other tables.

Note: Most of your tables should be of this type.

With no relations (radio button)

This will list all tables that do not have any relationship to any other table in your data source.

Note: None or very few of your tables should be of this type.

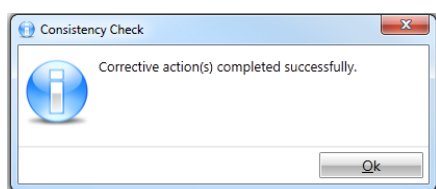
2.4.5.3. Check Consistency

Right-click the data source instance under operation and choose Check Consistency or select an instance and choose Check Consistency from the Data Quality tool group in the ribbon to do a consistency check. This will start a validation process to ensure metadata consistency. If any discrepancies are found between the metadata defined in Xpert BI and that in the destination database (ODS), these will be reported in the Consistency Check window. The inconsistencies can be changes in table or column names, data types, keys and/or foreign keys.

| Critical Inconsistencies Detected | | | | | | | | | |
|-----------------------------------|------------------|---------------------|---------------------|---------------|------------------------------|---------------|---------------|-------------------|--|
| Repair Inconsistencies | | | | | | | | | |
| Base Object | Base Object Type | Object Element Type | Object Element Name | Severity Code | Problem | Property Name | Value Current | Value New | |
| TestSource_1_1_Employee | Table | | | Fixable | | | | | |
| TestSource_1_1_Employee | Table | Column | FirstName | Fixable | | | | | |
| TestSource_1_1_Employee | Table | Column | FirstName | restruct | Column name has been changed | ColumnName | FirstName | EmployeeFirstName | |

The Severity code will tell you if Xpert BI can repair the inconsistency in metadata. You will find the exact inconsistency in the Problem column.

Xpert BI will offer to resolve the variations in metadata. This is done by clicking the Repair Inconsistencies button. Xpert BI will then update the ODS database tables with the new metadata definitions, even those that require table recreation and display the following message when complete:



If you want to investigate the consequences of repairing the inconsistencies before running the Repair Inconsistencies you are free to do so, but you will not be able to process the table before the inconsistency is repaired. Inconsistencies are generally solved in two ways, either change the metadata or run Repair Inconsistencies. When the destination tables are consistent with the defined metadata you can start processing the data.

In the case there is an error in the repair script this will be displayed together with the script so that the inconsistencies can be resolved manually. An example scenario is incompatible data type conversions.

Note: You should always perform a consistency check if you have done any changes in your metadata definitions.

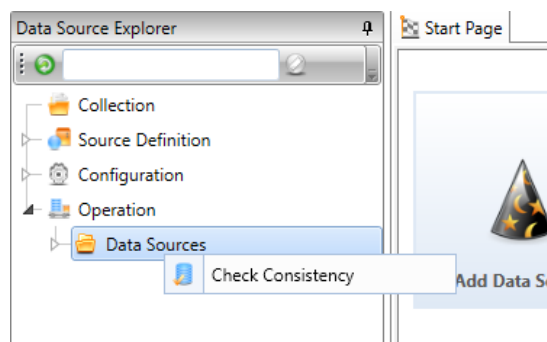
Note: Repair Inconsistencies may take a while to complete depending on the discrepancies and table size. If you experience timeout on queries, you can adjust the timeouts in Options-Miscellaneous.

Note: Check consistency is not available for Azure Data Lake and Snowflake.

Check Consistency All

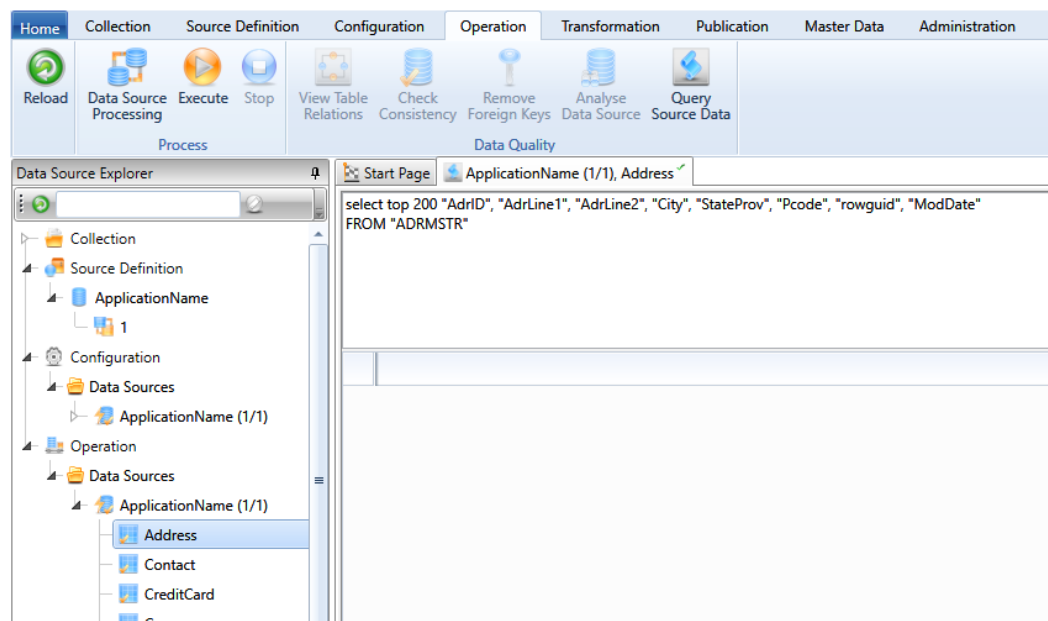
By right clicking on the Data Sources folder in Operation, it is possible to check consistency for all data sources.

This will open the same pane as a normal check consistency does.



2.4.5.4. Query Source Data

Right-Click on a table and choose Query Data or select a table and choose the Query Data icon from the Data Quality tool group in the ribbon.



This will bring up a query window with a select statement querying the data source. You may edit this query to fit your query needs (including changing to another table). The query syntax must match the syntax in the source system you are querying. To run the query, click Execute from the Process tool group in the ribbon.

Note that this is not supported if the source is Azure Data Lake.

2.4.5.5. Remove Foreign Keys

Right-click the data source instance under operation and choose Remove Foreign Keys or select an instance and choose Remove Foreign Keys from the General tool group in the ribbon to remove the foreign keys in the ODS database for a data source instance. The foreign keys will be implemented again at the next data source processing. This is only supported on SQL Server and Azure SQL Database.

2.4.5.6. Instance and Instance Table Properties

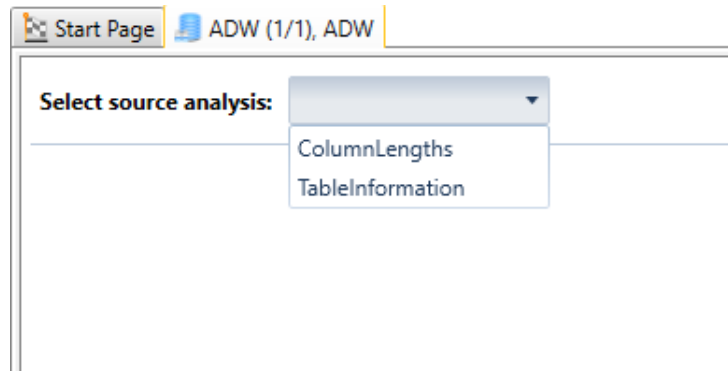
LastProcessed

This property shows when there was a successful update of the data source. If there are schedules on table level, the LastProcessed date will be updated both on instance and table level when the process is successful. However, if the entire data source is processed and only one table fails, the LastProcess date will not be updated.

2.4.5.7. Analyse Data Source

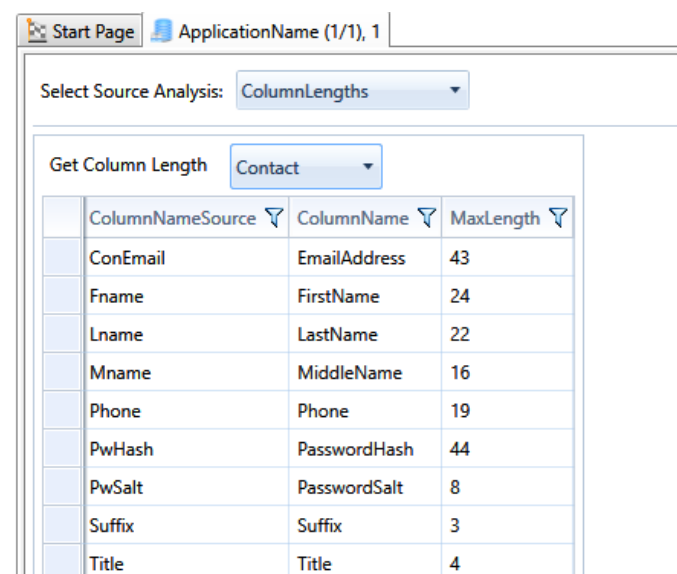
Right-click on a data source instance under operation and choose Analyse Data Source or select an instance and choose Analyse Data Source from the Data Quality tool group in the ribbon to open the Analyse Data Source pane.

The dropdown lets you choose between the different analysis types available. The source analysis will only evaluate tables and columns that are imported in Xpert BI.



ColumnLengths



The dropdown lets you choose from a list of available tables in the chosen data source instance. Selecting a table will query the data source to find the maximum column lengths for all character type columns (char, nchar, varchar and nvarchar) in the table. A popup will inform you if there are no character columns in the selected table.



Note: this query connects to and runs against the source system.

TableInformation

This will show a list of all tables in the chosen data source instance with row count and ODS size information.

 Start Page  ADW (1/1), ADW

Select source analysis: TableInformation ▼

Table row counts

| | TableNameSource ▼ | TableName ▼ | Rows ▼ | ODS Size (MB) ▼ |
|--|-------------------|-----------------------------|--------|-----------------|
| | SLSORDL | SalesOrderDetail | 121317 | 29.21 |
| | SLSORDH | SalesOrderHeader | 31465 | 15.83 |
| | SLSORDR | SalesOrderHeaderSalesReason | 27647 | 2.77 |
| | CONPERS | Contact | 19972 | 9.5 |
| | ADRMSTR | Address | 19614 | 5.14 |

Note: this query connects to and runs against the source system, except for ODS Size, which queries ODS.

Note that Analyse Data Source is not supported if the source is Azure Data Lake.

2.4.6. Change Management

2.4.6.1. Change table and column name

Changing table and/or column names are done in the Source Definition. Click on the version you want to edit and edit the new names directly in Xpert BI or export/import to excel to edit the new names.

After you have verified the change, run 'Check Consistency' to analyse and implement the changes you have made in ODS.

2.4.6.2. Add or remove column

If source metadata is imported

A column is added or removed by checking or unchecking the checkbox for the column. If it should be included it also has to have a name. Removing the name will automatically uncheck the checkbox.

After you have verified the change, run 'Check Consistency' to analyse and implement the changes you have made in ODS.

If source metadata is not imported

If the column is a new source column, the table must be re-imported from the metadata database by using Import Source Metadata. If the column does not exist in the metadata database, it must also be added there before it can be imported. After the new metadata is imported, it is added by giving it a name. This will automatically check of the left checkbox.

After you have verified the change, run 'Check Consistency' to analyse and implement the changes you have made in ODS.

2.4.6.3. Add or remove table

If source metadata is imported

A table is added or removed by inserting or removing the new name for the table and naming the columns that you want to include. If you want to deactivate a table or column but keep the name for future use, you can use the checkbox to the left of the table/column to deactivate it. It is possible to multi-select before activating/deactivating.

After you have verified the change, run 'Check Consistency' to analyse and implement the changes you have made in ODS.

If source metadata is not imported

If the table is a new table, the table must be imported from the metadata database by using Import Source Metadata. If the table does not exist in the metadata database it must also be added there before it can be imported. After the new metadata is imported, it is added by giving it a name. This will automatically check of the left checkbox.

After you have verified the change, run 'Check Consistency' to analyse and implement the changes you have made in ODS.

2.4.6.4. Edit relationships and/or keys

Table relationships are managed in the metadata database in SQL Server management studio. If you need to add, change or remove a relationship or key, edit the metadata in SQL server management studio and re-import the relevant tables.

After you have verified the change, run 'Check Consistency' to analyse and implement the changes you have made in ODS.

2.4.6.5. Change data type and/or length

Data types are managed in the metadata database in SQL Server management studio. If you need to change the data type, edit the metadata in SQL server management studio and re-import the relevant tables.

After you have verified the change, run 'Check Consistency' to analyse and implement the changes you have made in ODS.

2.4.7. Administration

The administration tab contains Xpert BI Event log and application configuration and setup.

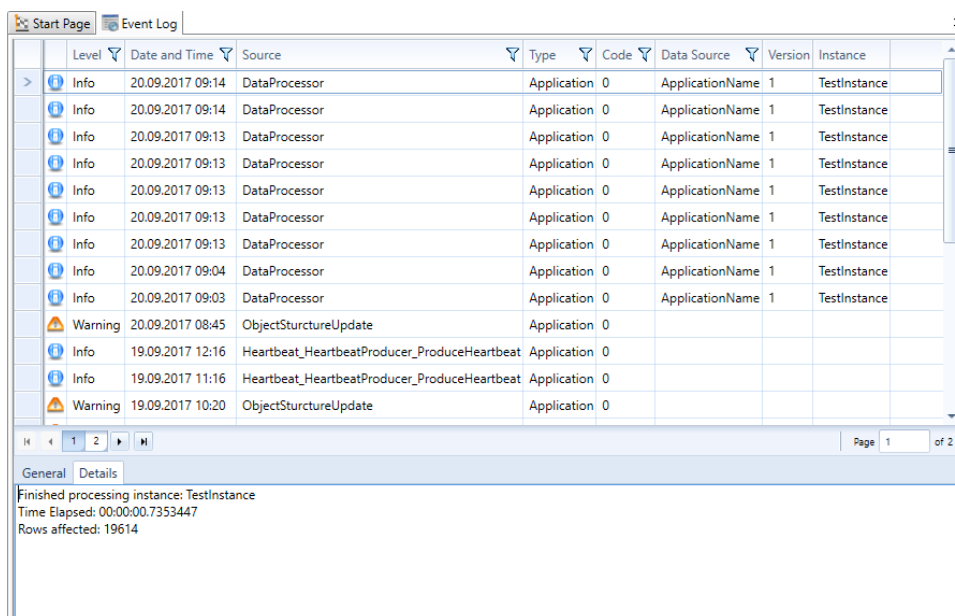
2.4.7.1. Reload

The Reload icon will reload the current administration tab you have opened, e.g. the Event Log.

2.4.7.2. Event Log

The Event Log shows all events in Xpert BI. By selecting the event, you find the details in the window under the log.

In addition to the Event log, there is also a table in the management database called PerformanceLog which is a table that holds performance related information regarding the data processing tasks.



| Level | Date and Time | Source | Type | Code | Data Source | Version | Instance |
|---------|------------------|--|-------------|------|-----------------|---------|--------------|
| Info | 20.09.2017 09:14 | DataProcessor | Application | 0 | ApplicationName | 1 | TestInstance |
| Info | 20.09.2017 09:14 | DataProcessor | Application | 0 | ApplicationName | 1 | TestInstance |
| Info | 20.09.2017 09:13 | DataProcessor | Application | 0 | ApplicationName | 1 | TestInstance |
| Info | 20.09.2017 09:13 | DataProcessor | Application | 0 | ApplicationName | 1 | TestInstance |
| Info | 20.09.2017 09:13 | DataProcessor | Application | 0 | ApplicationName | 1 | TestInstance |
| Info | 20.09.2017 09:13 | DataProcessor | Application | 0 | ApplicationName | 1 | TestInstance |
| Info | 20.09.2017 09:13 | DataProcessor | Application | 0 | ApplicationName | 1 | TestInstance |
| Info | 20.09.2017 09:04 | DataProcessor | Application | 0 | ApplicationName | 1 | TestInstance |
| Info | 20.09.2017 09:03 | DataProcessor | Application | 0 | ApplicationName | 1 | TestInstance |
| Warning | 20.09.2017 08:45 | ObjectSturctureUpdate | Application | 0 | | | |
| Info | 19.09.2017 12:16 | Heartbeat_HeartbeatProducer_ProduceHeartbeat | Application | 0 | | | |
| Info | 19.09.2017 11:16 | Heartbeat_HeartbeatProducer_ProduceHeartbeat | Application | 0 | | | |
| Warning | 19.09.2017 10:20 | ObjectSturctureUpdate | Application | 0 | | | |

General Details

Finished processing instance: TestInstance
Time Elapsed: 00:00:00.7353447
Rows affected: 19614

The Level specifies the event severity. An Info event includes events that are related to user activity, successful processing, license heartbeat etc. An Error event includes processing errors, metadata errors etc.

The Date and Time specifies when the event occurred

The Source specifies where in the application the message originates from.

The Type is the event type and specifies from where the event originated. This will in most cases be 'Application' which is the Xpert BI application.

The Code is a placeholder for a detailed error code. This will in most cases be 0.

The Data Source, version and instance specifies which data source, version and instance the event is related to, if any.







The name column will contain different things based on what type of event it is. For collection it will contain the folder and table name, for process groups it will contain the process group name and so on.

2.4.7.3. Configure Xpert BI dB

This allows you to review and/or edit the management database and BI Solution prefix. Please see Configure Xpert BI for how to set up a management database.

2.4.7.4. Manage Modules

The Manage Modules pane is divided into two main areas. The top area makes it possible to manage Xpert BI Solution modules. The Solution modules list reflects the Solution modules listed in the license file. Each Solution module can be bought as a separate module. Each entry gives the name of the module, the version of the application it belongs to, an installed/not installed checkbox, a description button, and an install button.

| Solution modules | | | | | | |
|------------------|-------------------------------------|--|---|---|---|--|
| | Installed | Application Name  | Application Version  | Description | Install Module | |
| > | <input checked="" type="checkbox"/> | SAP | Base |  |  | |
| | <input checked="" type="checkbox"/> | Siebel | Base |  |  | |

When clicking the Description button a dialog box will open. The dialog box contains a short description of the modules and the names of the included objects.

When installing Xpert BI, any licensed Solution modules will automatically be installed. It is however possible to re-install any of the listed Solution modules by clicking the Install Module button.

The bottom area lists any modules which have been updated since the last release (of Xpert BI or of a Solution module). This works as a patching mechanisms for small updates to the Xpert BI core, and/or updates to the Solution modules. A module is updated by clicking on the Update Module button.

2.4.7.5. License

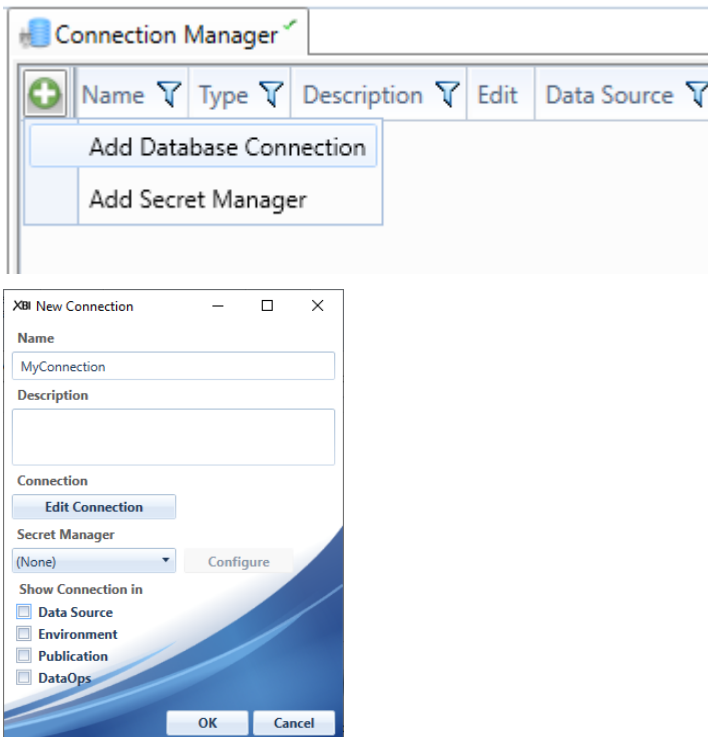
This allows you to review and renew your license details.

2.4.7.6. Connection Manager

The connection manager allows connection strings to be defined and re-used in several different aspects of Xpert BI. It is **important** to note that if the connection is updated, it will be updated for all connections that uses this managed connection.

It is recommended to create all managed connection strings for all environments (development, QA, production) in the lowest environment, ie. development. This will make it easier when migrating managed connection strings, as all the managed connections in the migration target will be overwritten.

The Connection Manager is found in the Administration ribbon menu. To create a new managed connection, click on the plus-sign and select Add Database Connection



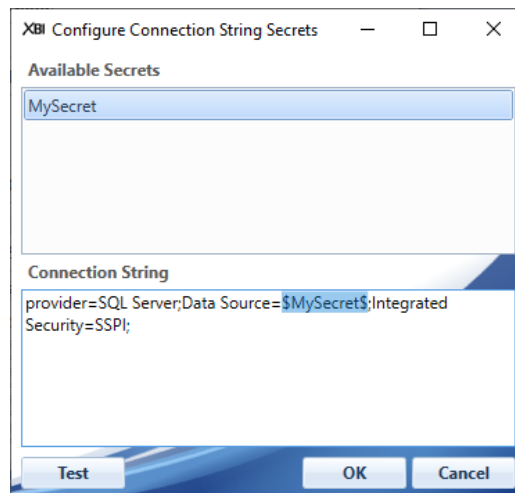
Name is the name of the connection. This must be unique.

Description is the description of the connection.

Connection is the connection string. This will open the connection dialog box.

Secret Manager enables the use of services that stores secrets securely. These secrets can then be used in the connection string. See section Secret Manager for details.

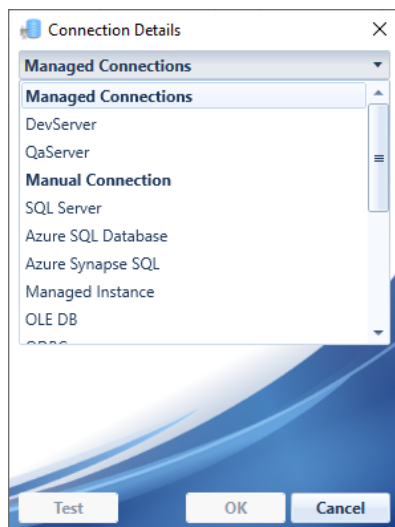
Configure is used to configure the connection string with secrets. Replace any part of the connection string with \$MySecretName\$ where MySecretName is the name of the secret. Double click on any available secret to automatically insert the secret in to the connection string at the selected location.



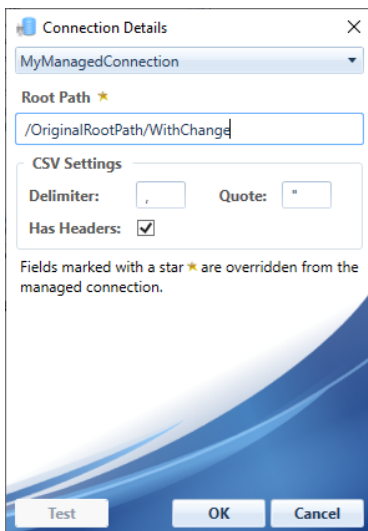
Show connection in controls where the managed connection will be visible.

- Data Source: Will be visible when selecting a connection in Extraction or Collection.
- Environment: Will be visible when selection a connection when migrating, changing stack or when adding remote additional databases.
- Publication: Will be visible when selecting a connection when exporting to Database.
- DataOps: Will be visible when selecting a connection when running a unit test in DataOps.

In order to use a managed connection, it must be selected in the drop down box when connecting to a database, as seen below.



Some connection types, when used as managed connection, can have properties that can be overridden. If the managed connection has any properties that can be overridden, they are shown in the connection details window. Azure Data Lake type is one of these types:



Connection Details

MyManagedConnection

Root Path ★
/OriginalRootPath/WithChange

CSV Settings

Delimiter: , Quote: "

Has Headers: ☒

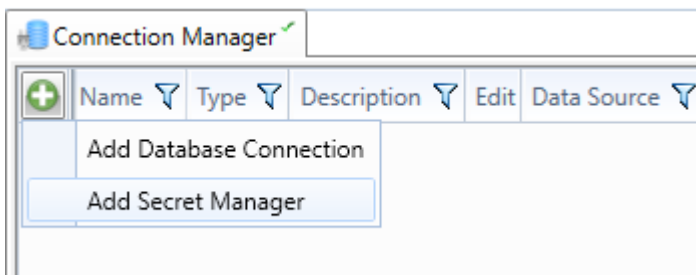
Fields marked with a star ★ are overridden from the managed connection.

Test OK Cancel

2.4.7.7. Secret Manager

The secret manager allows for interaction with services that stores sensitive information. Most commonly used to store keys and passwords.

The Secret Manager is found in the Administration ribbon menu. To create a new secret manager, click on the plus-sign and select Add Secret Manager



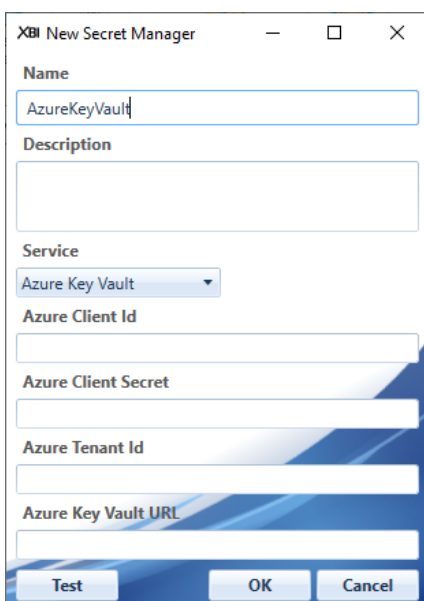
Connection Manager

+ Name Type Description Edit Data Source

Add Database Connection

Add Secret Manager

This window will show:



XBI New Secret Manager

Name
AzureKeyVault

Description

Service
Azure Key Vault

Azure Client Id

Azure Client Secret

Azure Tenant Id

Azure Key Vault URL

Test OK Cancel

Name is the name of the secret manager. This must be unique.

Description is the description of the secret manager.

Service is the type of storage the manager is going to connect to. The following input fields after service depends on the service selected. See sub section for more information.

Test button will try to connect to the service and retrieve all available secrets.

Services:

Azure Key Vault

Azure Client Id is the application or client id of the service account that is used to access the Key Vault.

Azure Client Secret is the secret key that is used to authenticate the client.

Azure Tenant Id is the tenant id that the Key Vault is created in.

Azure Key Vault URL is the URL for the Key Vault, usually on the format
'https://myKeyVaultName.vault.azure.net/'

Usage

The secret manager can be used in a managed connection or used to insert secret values into strings in Xpert BI. To insert into a string, a special syntax must be used.

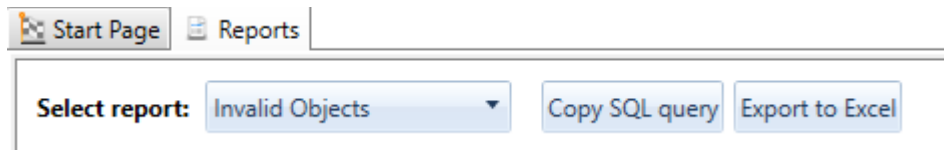
Example: `$Secret->MySecretManagerName->MySecret$`

The syntax is a text starting and ending with a dollar-sign '\$'. After the first dollar-sign, the text 'Secret' must follow to indicate usage of a managed secret. Next is the arrow '->' to indicate the next part of the syntax, which is the name given to the secret manager that is to be used (i.e. 'AzureKeyVault' shown in the example image above). The next part is the name of the secret in the secret storage.

This syntax can be currently used in some special fields in SOAP and REST collection. See the Collection chapter for specifics.

2.4.7.8. Reports

This allows you to view various reports based on information in the Xpert BI management database.



To view a report, simply select it from the dropdown menu found at the top. The reports available are:

- Invalid Objects – displays all objects that have invalid SQL code
- Executable Objects – displays all inline views and inline SPROCs
- Unused Objects – displays all objects which are NOT part of any process group structure
- Performance Log – displays the data load performance log
- Objects With Description – displays all objects that have a description
- Incremental Objects – displays all objects with incremental loading
- Objects Used By Multiple PGs – displays all objects which are used by multiple process groups. This report is default filtered to show only executable objects and objects that are not located in the ODS database.
- Objects Only In All Group – displays all object which are only used by “All” group
- MDS Source Objects – displays objects used in MDS reports
- Run Collision – displays performance runs that has overlapping execution times. The List view mode shows only collisions and the Gantt view mode shows all runs in the selected time period.
- Objects In Process Group – displays all objects included in the selected process group.
- Sources In Process Group – displays all source objects (leaf nodes) in a process group.
- Tagged Columns – displays all tagged columns and their parents.
- Undocumented Objects In Process Group – displays all SQL objects without a description.
- Used Inactive ODS Tables – displays all ODS tables that are inactive or not part of any data source, but still in use. A table is considered in use if any object reads from it. If the table is in Azure Synapse SQL, it is considered in use if either the table itself or its “_Current” view is used.

The selected report is loaded, and the result displayed below. If there are no rows in the report, it signals an empty report. For example, if no rows appear after loading the Invalid Objects report, there are no invalid objects found.

The ‘Copy SQL query’ button copies the SQL query for the selected report to the clipboard.

The ‘Export to Excel’ button allows you to export the selected report to an Excel file.

When right clicking on a list item in an object-report, a context menu to view dependency graph of the clicked object will appear.

2.4.7.9. Migrate Environment

Clicking the Migrate Environment button in the ribbon bar opens a wizard that guides you through the process of migrating your environment (for example from development to production). This chapter uses 'Source' and 'Target' as description of respective 'From Environment' and 'To Environment'. The migration process must be configured from the target environment.

Note: The management database on both source and target must be refreshed (using the Refresh Solution Explorer button) before the migration starts. The management databases must have identical names (prefix) and Xpert BI version numbers.

Note: Page 3 and 4 of the migration wizard are only available if your license includes the Transformation module.

Note: The following will not be migrated by the Xpert BI migration process and must be handled manually if applicable:

- Security/authentication on objects and databases are not migrated
- No data in the solution are migrated (except for the Xpert BI management database)

Note: If there are no SQL Server databases included in the target environment, page 1 is skipped.

Page 1 - Target Environment Database Backup

Performs a backup of any (or all) of the databases on the target environment. Simply check the databases you want to backup and then click the Backup button. The backup directory needs to be an existing directory on the target SQL Server. If the databases are split across multiple servers the folder must be available to all servers, or the backup must be performed in multiple steps.

| Database Name | Options |
|--|---------|
| <input checked="" type="checkbox"/> TC_Simple_DMH | |
| <input checked="" type="checkbox"/> TC_Simple_ELM | |
| <input checked="" type="checkbox"/> TC_Simple_ETL | |
| <input checked="" type="checkbox"/> TC_Simple_MAN | |
| <input checked="" type="checkbox"/> TC_Simple_ODS | |
| <input type="checkbox"/> TC_Simple_STG | |
| <input checked="" type="checkbox"/> TC_Simple_XBI_MGMT_DEV | |

Backup Directory: C:\Program Files\Microsoft SQL Server\MSSQL14.MSSQLSERVER\MSSQL\Backup

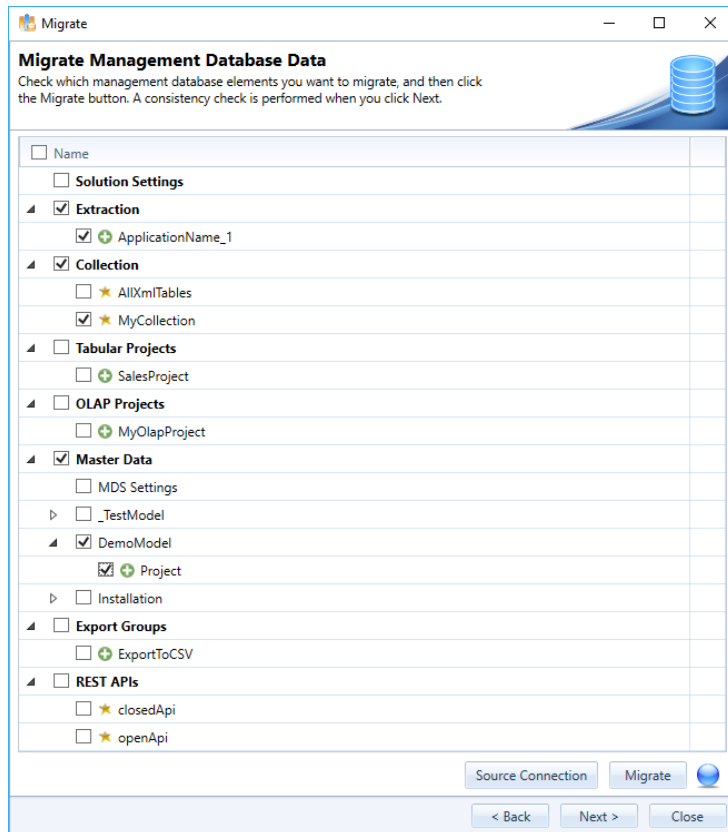
Clicking the **Options** button will open a window with backup options. The following option(s) are available:

Copy Only enables the Copy-only option for the backup(s).

Page 2 - Migrate Management Database Data

Migrates the data in the management database (Solution Settings, Extraction, Collection and Master Data). This means the configurations done in Xpert BI.

First, select the source management database by clicking the Source Connection button to get the correct element list.



Next, check of the various elements you want to migrate and click the Migrate button. After migration, click next, then a consistency check is performed automatically on all data source instances and migrated collection tables. If any critical inconsistencies are detected, a window displaying them will appear. This window also has a Repair Inconsistencies button. When this is clicked Xpert BI will try to automatically fix the inconsistencies. The consistency check is run for all data sources (and versions), not just those that are migrated. If you do not want to repair the inconsistencies during migration, you can click the Skip Repair button. Note that this is not recommended if you have migrated changes to any extraction data source and plan to migrate a process group using tables from that extraction data source. If there are any remote databases (after migration) with missing connection string in target, a window will open giving you an option to set the connection string.

If **Solution Settings** is checked, the following is migrated:

Generic Solution Settings

- All information and user-changeable settings in the Options window, except the Auto Refresh option.
- All information on any non-remote additional databases added in Xpert BI. The consistency check will create any new non-remote databases not found in target. However, deleted databases in source will not automatically be deleted in target. If there are any remote databases (after migration) with missing connection string in target, a window will open giving you an option to set the connection string.
- Information on all imported User-Defined Functions (UDF), including the function code. Note: Functions that are removed in source will be removed from target.

- All descriptions added to database schemas in the Database Schemas pane.

Managed Connections/Secrets

- All Managed Connections
- All Managed Secrets

If an **Extraction** source (data source version) is checked, the following is migrated:

- All information directly related to that source metadata (Import Source Metadata) including connection information.
- Information related to the destination metadata for the selected source includes:
 - o All information and user-changeable information in Edit Metadata, Define (and Set) Global Filters and Configure Relations.
- Information related to Configurations of all instances of the selected version
 - o Everything in Set Table Filters, except the Last Value.
 - o Everything in Edit Data Load Settings
 - o All instances of that data source, except any connection information set on an instance or table, last successful processing information (start/end time and rows affected) and Last Value for any incremental rules for this source (Table Filters).

If a data source or instance is deleted in source it will be deleted in target. Non-existing in target will be created.

Note: If a new data source version is migrated, the connection string will need to be set manually on target after the migration. This also applies if the connection string is changed in source and you want to update it in target as well. The same is true for table level connection strings in migrated data sources.

Note that data sources where Set Destination is set to something other than default, the destination type is migrated, but the connection info is not.

If a **Collection Folder** is checked (first level under the **Collection** header), the following is migrated:

- All information belonging directly to that Collection except the information in Top Level Node Path and connection strings.

This means that if a new Collection is migrated, the Top Level Node Path or connection strings have to be set manually after migration is done. For updated Collections this is left untouched.

If a **Collection Table** (leaf level under **Collection**) is checked, the following is migrated:

- All information on the table except the paths to the table files (Config, Input etc.)

This means that if a new table is migrated, the paths to the file system must be set manually after migration is done. For updated tables these paths are left untouched.

If a Collection or table definition is deleted in source, it will be deleted from Xpert BI configuration in target, but any folders on the file system will remain and the table is not deleted in the collection destination database. (this is, if applicable, the next step of migration). If a new table is migrated to target, the output table will be created in the database.

Note: Collection files (Excel, csv, etc.) located on the file system are not migrated, and any collected data (in the database) are also not migrated.

Note: If a new Collection folder is migrated, the Top Level Node Path will have to be set manually after the migration. The same applies if the path is changed in source and you want to update the target as well.

Paths to XML-collections' XSLT-files will have to be updated and any files copied. The same goes for XML-collections that use the "IncludeAllXmlTables"-feature, they need their XSD file path set and the file copied to target.

Note on first Collection migration: The collection and table IDs needs to be in sync between the source and target environment before migration. The IDs can be seen in the Properties window or in the management database.

If a **Tabular Project** is checked, the following is migrated:

- All project metadata stored by Xpert BI, including any descriptions.

If an **OLAP Project** is checked, the following is migrated:

- All project metadata stored by Xpert BI, including any descriptions.

Note: If an OLAP or Tabular project is migrated, make sure the cube is also migrated to keep the documentation in sync with the cube.

If **MDS Settings** is checked, the following is migrated:

- All information and user-changeable settings in the Configure MDS Connection window.

If an **MDS Entity** is checked, the following is migrated:

- All information stored by Xpert BI belonging to that entity.
- All information belonging to any data loads created for that entity.
- All information stored by Xpert BI belonging to any subscription views created for that entity.

If an **Export Group** is checked, the following is migrated:

- All information and user-changeable settings for the selected Export Group, including the list of included objects.

If a **REST API** is checked, the following is migrated:

- All information and user-changeable settings for the selected REST API, including the list of included objects. Security settings are not migrated, this includes OAuth values, bearer token values and roles.

If a **Power BI Workspace** is checked, the following is migrated:

- All synced Power BI objects included in this workspace. This includes all synced metadata for the workspace itself and the reports, datasets, data sources and dataflows in the workspace and all their descriptions.

If the **PowerBI.com Settings** are checked, the following is migrated:

- All information and user-changeable settings in the Configure Power BI Connection window.

If a **DataOps Folder** is checked, the following is migrated:

- The folder and all its subfolders.
- All data quality items.
- Add test items.

If a checkbox for a folder is disabled, it falls in under one of these categories:

- The parent folder has been renamed.
- The parent folder is a new folder.
- The parent folder is deleted.

A disabled checkbox will follow the selected value for its parent folder.

To see exactly which tables and columns from the management database is included in the migration, consult the below tables. In the first one there are some columns which are excluded from the migration (seen under “Excluded columns”) and in the last table all columns are included.

| Migrated Object | Tables Included in Migration | Excluded columns |
|--|---------------------------------------|--|
| Solution Settings - Generic | Application_Settings | AutoRefreshSolutionExplorer, LastFullRefreshTime, StackConnectionString, StackDestinationType |
| | Customer | DatabasePrefix, TestA, Server_Master, Server_ELM, Server_ETL, Server_INT, Server_ODS, Server_STG, Server_DEV, Server_QA, Server_PRD, UDF_Server, UDF_Database, RunNo |
| | Customer_Active_Databases_Additions | ConnectionString |
| | Userdefined_Functions | |
| | DatabaseSchema | |
| Solution Settings - Managed Connections / Secrets | ManagedConnection | |
| | SecretManager | |
| Extraction | DataSources | |
| | Datasource_Version | |
| | Datasource_Tables | |
| | Datasource_Columns | |
| | Datasource_Foreign_Keys | |
| | Datasource_Foreign_Key_Columns | |
| | DataSources_Global_Filter_Definitions | |
| | Customer_DataSources | Schema, DatabaseType, ConnectionProvider, ConnectionString, ProcessRowsAffected, ProcessStart, ProcessEnd, LastSuccessfulProcess, DataTaskXml, |

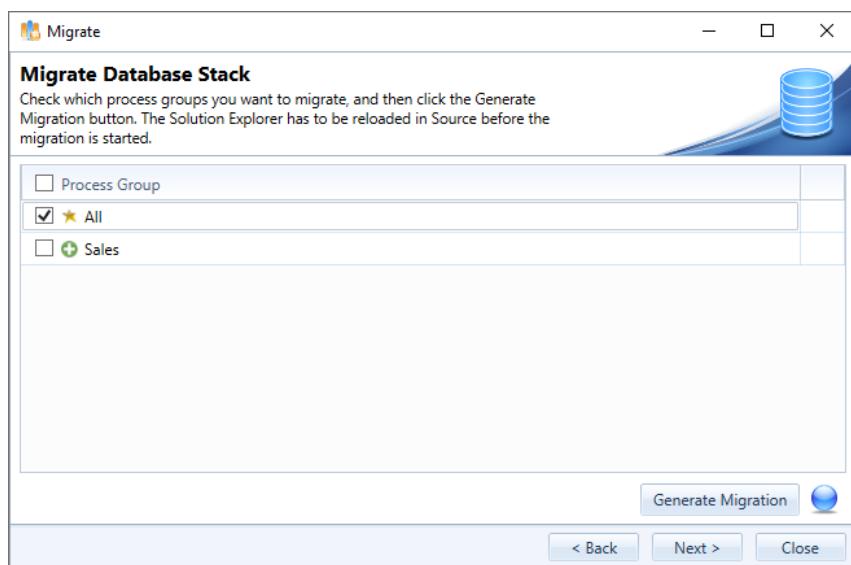
| | | |
|--------------------------|---|---|
| | | DestinationConnectionString, AzureDataLakeConnectionString, AzureDataLakeContainerName, AzureDataLakeRootPath, AzureDataLakeFileType |
| | Customer_DataSources_Rules_Table_Level | Schema, DatabaseType, ConnectionProvider, ConnectionString, LastValue_Variant, LastValue_Display, LastValue_Select, LastSuccessfulProcess |
| | Customer_DataSources_Rules_Column_Level | |
| | Customer_DataSources_Global_Filters | |
| Collection Folder | Collection_Folder | Path, ConnectionString, FileStoreConnectionString |
| Collection Table | Collection_Table | ConfigFilePath, XsltFilePath |
| | Collection_Column | |
| | Collection_WebService | |
| | Collection_WebService_InputArg | |
| | Collection_WebService_PaginationParameter | |
| REST APIs | RestApi | AzureTenant, OauthAudience, OauthIssuer, EnableRoles |
| | RestApi_Entry | |
| DataOps Folder | DataOps_DataQualityItem | |
| | DataOps_DataQualityItem_DuplicateColumn | |
| | DataOps_DataQualityItem_ExcludedFactTableRelationColumn | |
| | DataOps_DataQualityItem_ProcessGroups | |
| | DataOps_Folder | |
| | DataOps_TestFolderSetup | ConnectionString |
| | DataOps_TestItem | ActualConnectionString, ExpectedConnectionString |

| Migrated Object | Tables Included in Migration |
|-----------------|------------------------------|
|-----------------|------------------------------|

| | |
|-----------------------------|--|
| Tabular Projects | Tabular_Column, Tabular_Hierarchy, Tabular_HierarchyLevel, Tabular_Measure, Tabular_Perspective, Tabular_PerspectiveMember, Tabular_Project, Tabular_Relationship, Tabular_Role, Tabular_RoleMember, Tabular_Table, Tabular_FieldChild |
| OLAP Projects | OLAP_Cube, OLAP_CubeDimension, OLAP_Dimension, OLAP_DimensionAttribute, OLAP_DimensionUsage, OLAP_Hierarchy, OLAP_HierarchyLevel, OLAP_Measure, OLAP_MeasureGroup, OLAP_Perspective, OLAP_PerspectiveMember, OLAP_Project, OLAP_Role, OLAP_RoleMember |
| Master Data Settings | MDS_Settings |
| Master Data Entity | MDS_Entity, MDS_DataLoad, MDS_SubscriptionView |
| Export Groups | ExportGroup, ExportGroup_Entry, ExportGroup_Parameters |
| Power BI Workspace | PowerBI_App, PowerBI_DataFlow, PowerBI_DataFlow_Column, PowerBI_DataFlow_Datasource, PowerBI_DataFlow_Table, PowerBI_Dataset, PowerBI_Dataset_Column, PowerBI_Dataset_DataFlowMapping, PowerBI_Dataset_DatasourceMapping, PowerBI_Dataset_FieldChild, PowerBI_Dataset_Hierarchy, PowerBI_Dataset_HierarchyLevel, PowerBI_Dataset_Measure, PowerBI_Dataset_Relationship, PowerBI_Dataset_Role, PowerBI_Dataset_RoleMember, PowerBI_Dataset_Table, PowerBI_Dataset_Table_SourceTable, PowerBI_Datasource, PowerBI_Report, PowerBI_Report_FieldChild, PowerBI_Report_Measure, PowerBI_Report_ObjectUsage, PowerBI_Workspace, PowerBI_Report_Page, PowerBI_Report_Visual, PowerBI_Report_VisualUsage |
| PowerBI.com Settings | PowerBI_Settings |

Page 3 - Migrate Database Stack

This migrates one or more process groups. The Xpert BI migration process will not migrate objects not included in a process group (at any level). You check the process group(s) you want to migrate and then click the **Generate Migration** button. This will generate the migration steps and open the Execute Migration window.



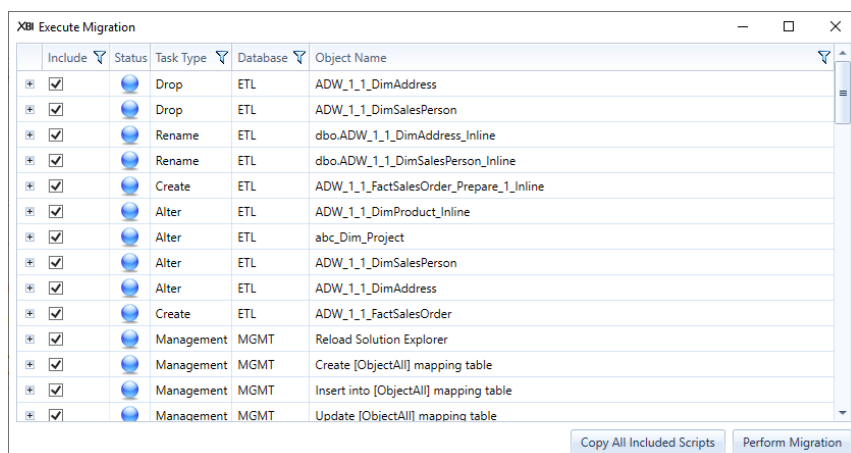
The Execute Migration window shows all generated steps for migrating the selected process group(s). The individual script for each step can be seen by expanding the rows.

There are two options to perform the migration:

- The **Copy All Included Scripts** button will copy all included SQL scripts to the clipboard. This can then be run manually to migrate the process group(s). Note that this is not supported when any of the objects included in the process group is on a remote database.
- The **Perform Migration** button will run each included step to perform the migration. If a step fails, the migration will stop and a column will appear, showing the error message. The migration can then be continued, either from the failed task or from the next one.

Note that excluding objects from the migration must be done with care, since it can easily lead to errors if other objects are dependent on the excluded object(s).

The steps with task type **Management** are related to the Xpert BI management database and should be included at all times.



For any selected process group, the following is migrated:

- Information on any Inline object and associated Incremental Filters that are included in the process group. Last Value, Rows Affected and Execution Time are not migrated for Incremental Filters.
- Information directly related to the process group, including its description and which solution objects are included in it (information that can be changed in Process Groups). If a process group is new it will be created in target and if it is deleted in source it will be deleted in target.
- All Fact Tables Dependencies information for objects/fact tables that are included in the process group.
- History Tables are potentially complicated regarding changes and must be handled with caution.
 - o If the history table destination (e.g. the <xxx>_History table) is included in a process group, all definitions of both history table source object and the history table itself will be migrated (alter table).
 - o If only the history table source object is included in the process group and not the history table destination, only the configurations and definitions of the source object will be migrated. Destination table must be handled manually.
- All fact tables, dimensions and other SQL objects that are included either directly in the process group or via a dependency, is migrated. This applies to ETL and ELM, and any additional databases added in Xpert BI. For ODS, all objects except tables and XXX_Current-views are migrated.
 - o SQL objects existing in source and not in target will be created in target.
 - o SQL objects existing both in source and target will be altered in target (if there are changes)
 - o SQL objects existing in target but not in source will be deleted in target.
- Object descriptions, column descriptions (for tables and views) and IgnoreReprocessingForMinutes for all objects included in the selected process group(s).
- Any Custom Objects used in the process group.
- Information on any Tags used by any SQL view/table column in the process group.
- Any manually defined column dependencies for objects in the selected process group(s).
- Any Executable Settings for executables included in the process group.

Note: If you have changed/added/deleted objects in a deleted process group, the objects will not be migrated.

Note: It is not possible to migrate Inline objects created in ODS, since their tables will NOT be migrated.

To see which management database tables are included in the migration of process groups, consult the below table.

| Table | Comment |
|---------------------|---|
| Object_All | Description and IgnoreReprocessingForMinutes columns are directly migrated, other columns are updated/created/deleted based on the objects included in the migrated process group(s). |
| ProcessGroups | All columns. |
| InlineObjects | All columns. |
| InlineObjects_Rules | All columns except LastTimeValue, LastIntValue, RowsAffected and ExecutionTimeSec. |
| History_Table | All columns. |

| | |
|--|---|
| Object_Columns | All columns. |
| SchemaForeignKeys | All columns, except SchemaForeignKeys_Id which is autogenerated. |
| SchemaForeignKeys_Links | All columns, except SchemaForeignKeys_Id which is taken from the SchemaForeignKeys table. |
| Object_Custom | All columns. |
| Object_Custom_Dependencies | All columns. |
| Tag | All columns. |
| Column_Tag | All columns. |
| ColumnLineage_ManualDependency | All columns, except ColumnLineageManualDependency_Id which is autogenerated. |
| ExecutableSettings | All columns. |
| ExecutableSettings_IncludeInProcessGroup | All columns. |

Page 4 - Reload Solution

Allows you to reset any incremental rules in Extraction or Transformation that might have been affected by the migration. The Extraction incremental settings are the incremental rules included in the data sources you migrated in page 2, and the Transformation incremental settings are the incremental rules for any Inline objects that were altered in page 3. If you copied the migration script and executed it manually in page 3, Xpert BI will assume that any Inline objects included in the script were in fact migrated and list their associated incremental rules here. Checking an object and clicking the Reset Incremental button will reset the incremental value, effectively forcing a full load the next time that object is processed.

Migrate

Reload Solution

Check which incremental object rules you want to reset and then press the Reset Incremental button. This will force a full load on the next processing.

☐ Name

☐ **Extraction Incremental**

☐ Address (ADW_1)

☒ **Transformation Incremental**

☒ TC_Simple_ETL.dbo.ADW_1_1_DimEmployee_Inline

Reset Incremental

< Back
Finish

Note: If a manual migration is performed (instead of using the Migration Wizard) using a database backup-restore approach, the following stored procedure has to be run in order for the Xpert BI application to detect and save the ITS objects:

```
USE [<your prefix>_XBI_MGMT_DEV]
GO
DECLARE @return_value int
EXEC @return_value = [dbo].[XBI_Tr_Create_InlineObjects]
SELECT 'Return Value' = @return_value
GO
```

Also, not all database settings are preserved in a database backup, ie. database owner is changed to the user which restored the database and trustworthy setting is set to OFF.

Run the following script on the management database:

```
USE [<your prefix>_XBI_MGMT_DEV]
GO
EXEC sp_changedbowner 'sa'
ALTER DATABASE management_database SET TRUSTWORTHY ON
GO
```







Run the following script for all restored databases that is included in the Xpert BI stack.

```
USE [<database name>]
GO
EXEC sp_changedbowner 'sa'
GO
```

Note that this is only supported on SQL Server only solutions, where the stack and management database are located on the same server.

2.4.7.10. Manage Tags

The Manage Tags module is used for creating and maintaining tags. A tag can be added to each object column, and is useful for classifying information (i.e. which columns contains sensitive GDPR-relevant data).

| Tags | | | | | | | |
|------|-----------|--------|-----------|---|-------------|---|--|
| | Name | Type | ShortName | RGB Code | Description | Delete | |
| | HR | Domain | HR |  #FF8DB3E3 | |  | |
| | R&D | Domain | R&D |  #FFFFFF00 | |  | |
| | Sensitive | GDPR | |  #FFFF0000 | |  | |

A tag must have a unique Name and Type combination. To change name or type of an existing tag, click the gear icon.

ShortName has a maximum character limit of 15 letters.

ShortName, RGB Code and Description are all optional and updated directly in the table.

To assign a tag to a column, open Solution Explorer and select an object. Click on any “Assigned Tags” cells in the “Column Tags” tab; this will show all available tags. Alternatively, the user can start typing the tag name and select desired tag from the dropdown list.

| Description | Column Description | Column Tags |
|--------------|---|-------------|
| Column Name | Assigned Tags | |
| Address_Id | <div> <div>× Domain - R&D</div> <div></div> <div>Add Tag...</div> </div> | |
| AddressID | Add Tag... | |
| AddressLine1 | <div> <div>× Domain - HR</div> <div>× GDPR - Sensitive</div> <div>Add Tag...</div> </div> | |
| AddressLine2 | <div> <div>× Domain - HR</div> <div>× GDPR - Sensitive</div> <div>Add Tag...</div> </div> | |
| City | Add Tag... | |
| ModifiedDate | Add Tag... | |
| PostalCode | Add Tag... | |

Tag Propagation

When an object is tagged, the tag information will automatically propagate based on columns upwards in the dependency tree. Exception to this are OLAP objects. Tag information will propagate up to and including OLAP. To see the changes in tag propagation after a column tag has been added or removed, the dependency graph has to be reloaded.

2.4.8. XBI_batch.exe

2.4.8.1. Installation and licensing

The XBI_batch.exe file is installed together with Xpert BI and follows the same license as the Xpert BI software. The file is placed in the folder: C:\Program Files\Xpert BI\

The file can be copied to and run from another folder, but it is advised that you run it from the default folder to ensure you always have the latest software and file updates, and that your environment is 'in sync'.

2.4.8.2. Parameters

XBI_batch.exe is called with the following parameters

- DatasourceId – application generated ID to identify the data source
- VersionId – application generated ID to identify the data source version
- InstanceId – application generated ID to identify the data source instance
- ServerName – the name of the server where the management database is installed
- DatabaseName – the name of the management database
- TableNameSource (optional) – the name of the table that will be processed. To include all related tables when extracting, append %XBI_Related% to the table name. To include multiple tables, separate the table names with a comma (example: TableNameSource="table1, table2")
- MaxThreads (optional) – the maximum number of threads that will be run in parallel when processing data. If this parameter is set it will be used instead of the value set in the Options window.
- ForceFullLoad (optional) – forces a full load when it is configured as incremental. When processing an extraction data source, this will clear the Last Value from Set Table Filters before processing. When processing an Inline view, this will clear the Last Value from Set Incremental Filters before processing.
- WindowedLoad (optional) – this option is only valid when processing a single table (using TableNameSource). It allows you to define a source selection (similar to Table Filters), that is only valid for this batch configuration. It will ignore any existing source selection on the table but will consider any active global filters and incremental filters.
- SmartProcessing (optional) – enables optimised processing for Process Groups, see Smart Processing for further details.

The parameter values are easily retrieved by right-clicking on the source or table in Configuration and choosing Copy Exe Parameters.

Usage examples:

| | |
|------------------------|---|
| TableNameSource | XBI_batch.exe DatasourceId=1 VersionId=1 InstanceId=1 ServerName=DESKTOP-1GN0QK6\INST2 DatabaseName=BI_STACK_XBI_MGMT_DEV TableNameSource=MyTable%XBI_Related% XBI_batch.exe DatasourceId=1 VersionId=1 InstanceId=1 ServerName=DESKTOP-1GN0QK6\INST2 DatabaseName=BI_STACK_XBI_MGMT_DEV TableNameSource="table1,table2" |
| ForceFullLoad | XBI_batch.exe GroupName="ProcessGroupName" ServerName=DESKTOP-1GN0QK6\INST2 DatabaseName=BI_STACK_XBI_MGMT_DEV ForceFullLoad |
| WindowedLoad | XBI_batch.exe DatasourceId=1 VersionId=1 InstanceId=1 ServerName=DESKTOP-1GN0QK6\INST2 DatabaseName=BI_STACK_XBI_MGMT_DEV WindowedLoad TableNameSource="table" |
| MaxThreads | XBI_batch.exe DatasourceId=1 VersionId=1 InstanceId=1 ServerName=DESKTOP-1GN0QK6\INST2 DatabaseName=BI_STACK_XBI_MGMT_DEV MaxThreads=4 |
| SmartProcessing | XBI_batch.exe GroupName="ProcessGroupName" ServerName=DESKTOP-1GN0QK6\INST2 DatabaseName=BI_STACK_XBI_MGMT_DEV SmartProcessing |

2.4.8.3. Error codes

The executable can return with the following codes:

0 - Success

1 - The data processing failed and the Event Log was updated

2 - The data processing failed and the Event Log was not updated because the management database was not available. The error was logged to C:\Program Files\Xpert BI\log\

3 - One or more of the arguments given are invalid.

4 -There was a mismatch between the .exe file version and the Xpert BI management database version.

5 -The heartbeat has not been answered in 5 days.

6 -The license is missing or invalid.

333 - There was an unhandled exception. The exception was logged to c:\temp\XpertBI_emergency_log.txt

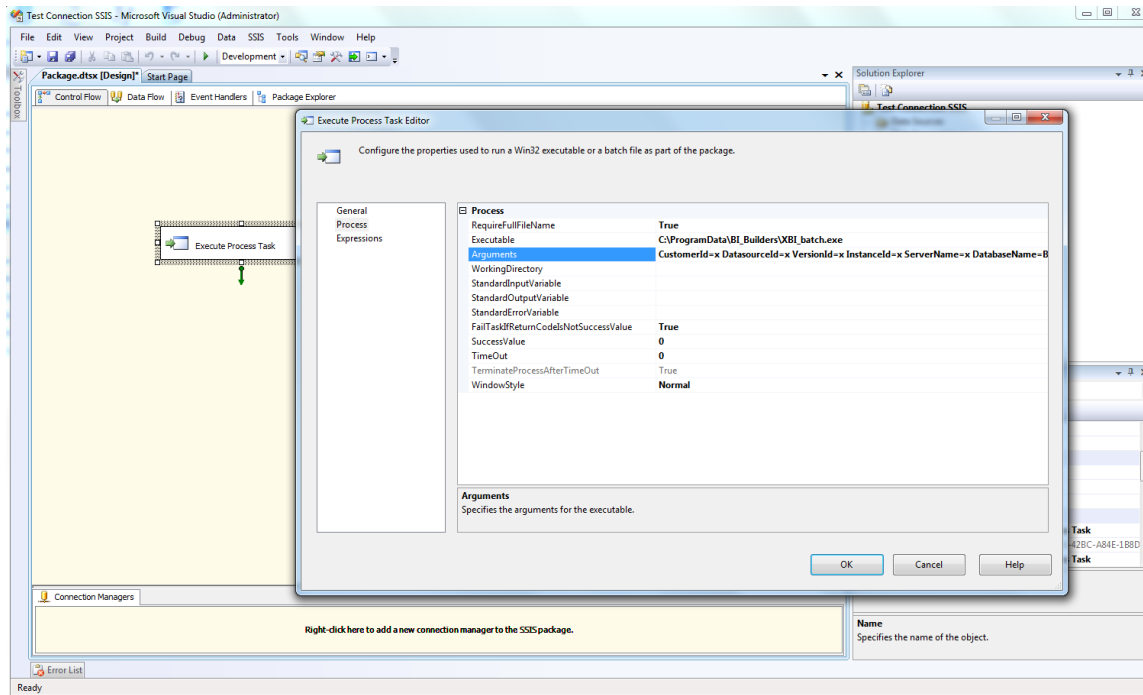
777 - There was an unhandled exception. It was not possible to log the exception to c:\temp\XpertBI_emergency_log.txt

2.4.8.4. Scheduling

The batch job can be scheduled through a script, SQL server integration services dtsx package and/or SQL Server Agent or any other method of preference.

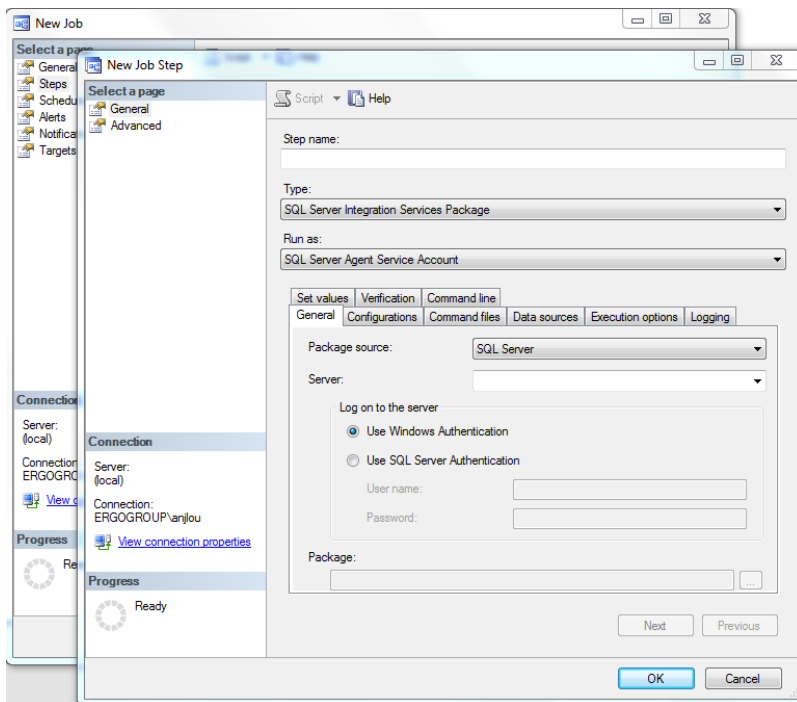
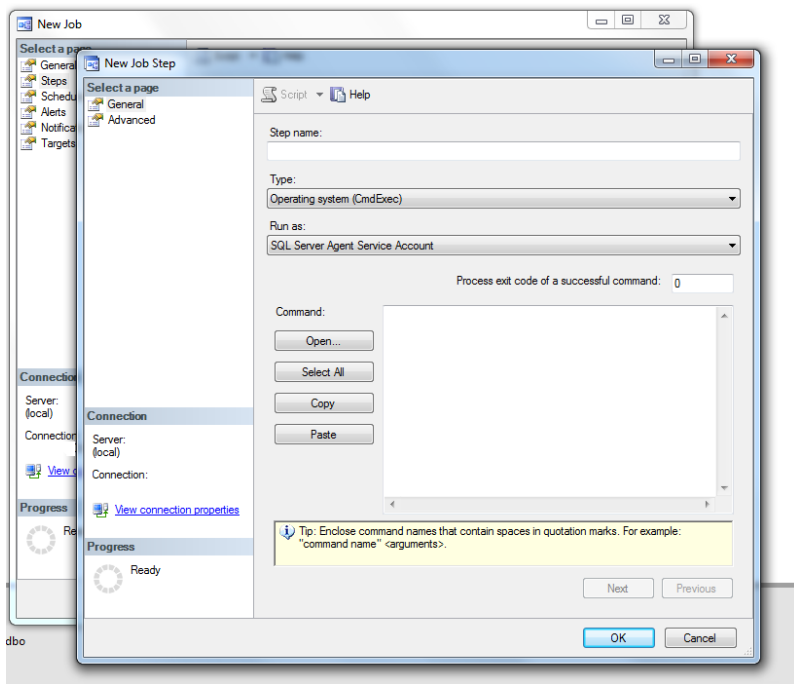
Below are screenshots of how to schedule the batch job through SSIS and through SQL Server Agent:

SSIS



Choose an Execute Process Task and set the Executable and Arguments in the Process properties. The Executable and Arguments are easily retrieved by right-clicking on the source or table in Configuration and choosing Copy Exe Parameters.

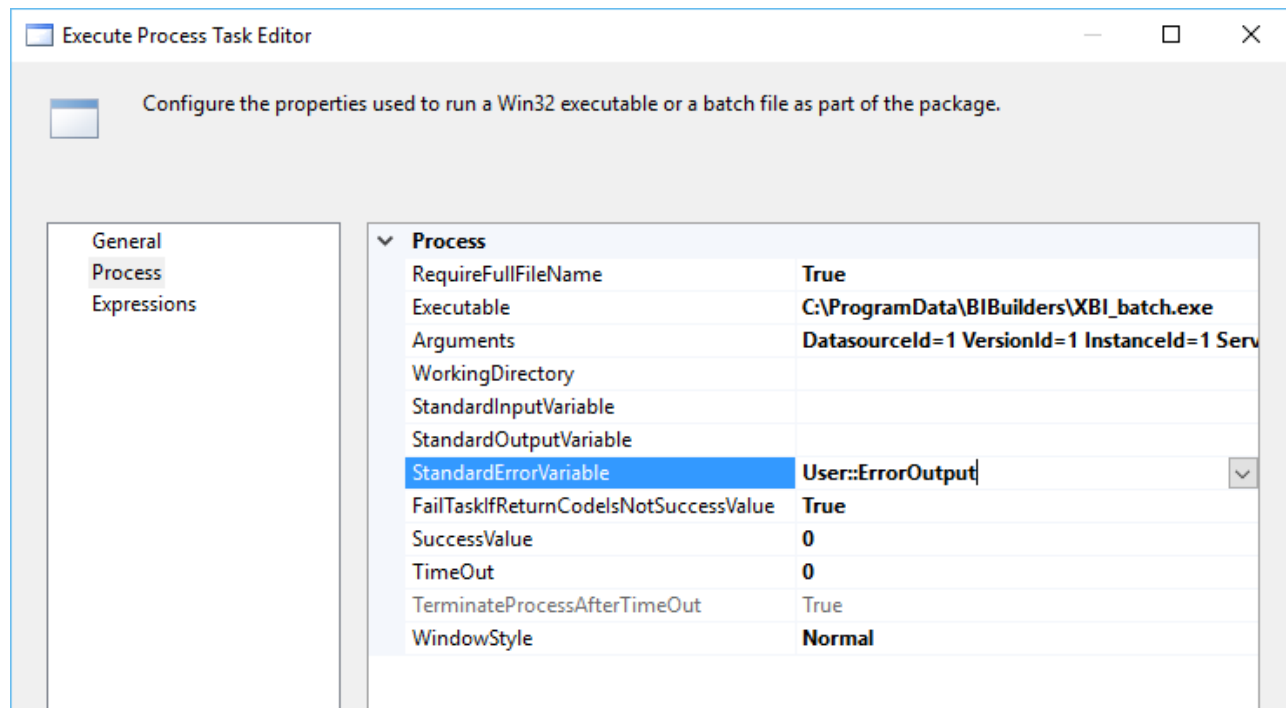
SQL Server Agent can be used to schedule either the SSIS package or to schedule the batch file directly.



Under New Job and New Step you can choose Type. Choose Operating System to schedule the batch file directly. Choose SQL Server Integration Services Package to schedule the SSIS package.

2.4.8.5. Error message handling

If the executable fails it will write an error message to the console. This error message is automatically displayed in the SQL Server Agent job history if the executable is scheduled directly from there, or it can be picked up by a SSIS package. To get a hold of the message in SSIS, you need to direct the *StandardErrorVariable* of the Execute Process Task to a user defined variable. This allows for flexibility in the error message handling; it can be sent in a mail, stored to disk, used in a script and more.



2.4.8.6. Command Window Information

When the XBI_batch.exe has started, a command window will show. Information about the current task is displayed in this command window. The following information is available:

- **Execution start** Timestamp showing when this task was started.
- **Executing as** The windows user that started the executable.
- **Command arguments** All the command line arguments given to XBI_batch.exe
- **Execution Mode** What type of task is being executed.

Based on the task being executed, there is possibly additional information related to the mode available.

2.5. ODS Structure (SQL Server and Azure SQL Database)

2.5.1. Tables

Xpert BI produces two different types of tables in ODS; data tables and surrogate key tables.

All table names are built on a robust naming convention.

Data table names have the following name structure:

`<DataSource>+'_'+<Version_Id>+'_'+<Instance_Id>+'_'+<TableName>`

Where

DataSource is data source name

Vesion_Id is the internal version id

Instance is the internal instance id

TableName is the actual table name (the new name given in the destination metadata definition)

Surrogate key tables have the following name structure:

`XBI_SGT_<DataSource>+'_'+<Version_Id>+'_'+<Instance_Id>+'_'+<TableName>`

2.5.2. Primary Keys

Xpert BI adds a surrogate primary key to all tables which is a single field primary key. The surrogate primary key is maintained in the surrogate key table and is used for both primary and foreign keys. The surrogate primary key column is of data type integer and has the following naming convention:

`<New Table Name>+'_Id'`

This naming convention makes it easy to identify both keys and foreign keys in tables.

2.5.3. Foreign Keys and Foreign Keys Roles

By using surrogate primary and foreign keys that reflects the table names and grouping all key (primary and foreign) columns as the first columns in the table, the ODS database becomes very readable for both an end user and a data warehouse expert. When you open a table you can automatically see which other tables this table is related to by the names of the surrogate foreign keys in the table.

In some cases you can have more than one relationship between two tables (e.g. InvoiceTable and AddressTable with ShippingAddress and InvoiceAddress as two different relationships). In these cases Xpert BI adds a 'role' name as an extension to the surrogate foreign key name to identify the specific relationship. Surrogate Foreign keys has the following naming convention:

`<New Table Name>+'_Id'['+'_'+<Role Name>]`

Where Role Name will be the foreign key column's new name, as defined in Xpert BI. If the foreign key is a composite key, additional role name rules apply.

2.6. ODS Structure (Azure Synapse SQL and Snowflake)

The table structure in Azure Synapse SQL and Snowflake are the same as for SQL Server with regards to naming. The biggest differences are that there are no surrogate tables and no primary keys added to the tables. A column with the same naming convention as the primary key column in SQL Server and it is the hash-value of the primary keys from the source. The same is done for the foreign keys. In addition to the ODS table, a `<New Table Name>_Current` view is also created. This view will always show the last loaded version of each row (based on primary key).

2.7. ODS Structure (Azure Data Lake)

When a data source instance is loaded into Azure Data Lake, one file is generated per table. The file type is either CSV or Parquet, depending on the settings. Each file will contain all the columns configured in Edit Metadata, with the given column names (no extra columns are added). The column ordinal position is the one imported from the source. The files are put in a folder structure: data source name / table name / year / month / day / timestamped file.

3. Xpert BI Transformation

3.1. Xpert BI Transformation Background

3.1.1. ITS – A new approach to ELT/ETL

A major challenge of BI solutions whether in large or small organizations is having control over data quality, updates and dependencies as solutions develop and expand over time.

Xpert BI Transformation introduces a new approach – Inline Transformation Services (ITS) - which gives the BI developer complete control of all database objects in the BI solution as well as automatic up to date documentation from source to destination at all times. The result is better data quality and reliability, the ability to adapt to change requests in the solution efficient and effectively and less maintenance problems over time. ITS also incorporates best practices in terms of naming conventions, data flows and object dependencies.

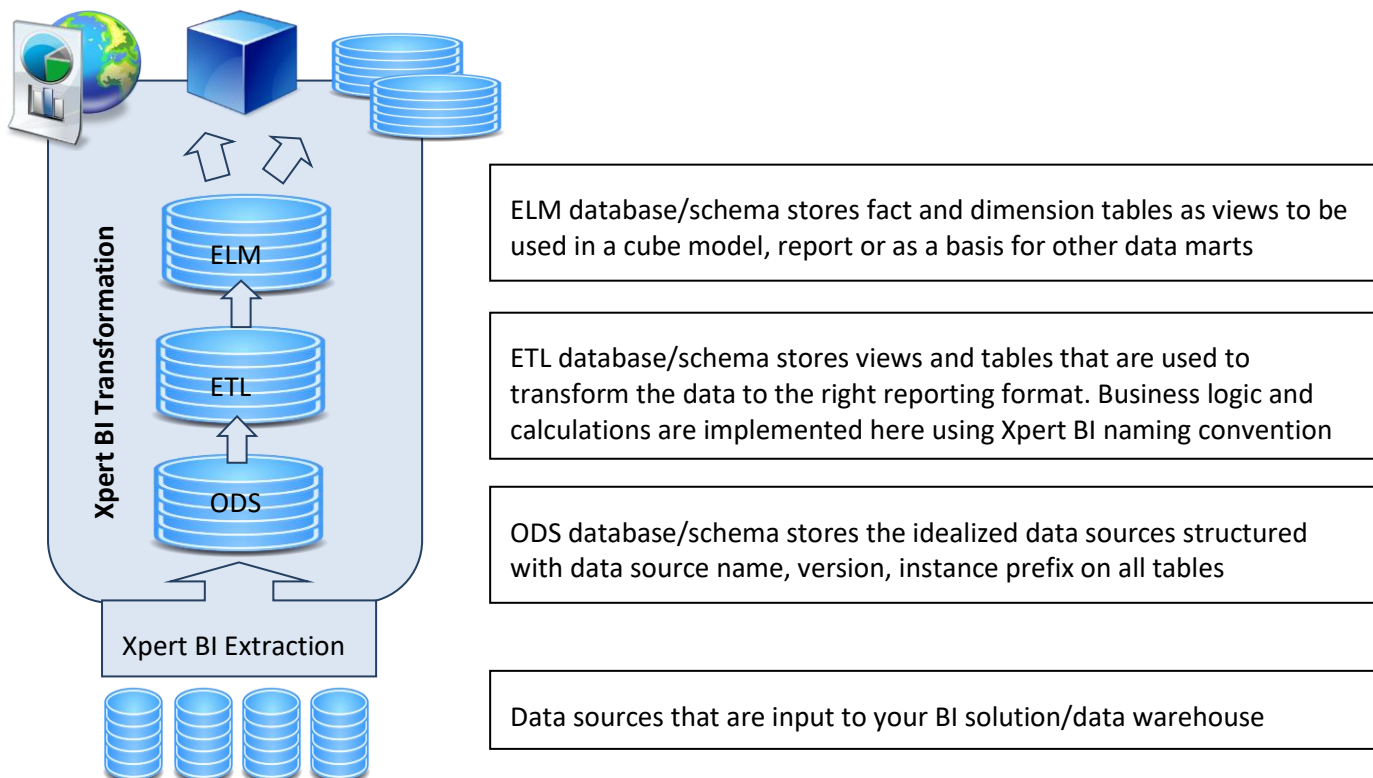
Xpert BI Transformation handles all data loads and it automatically and dynamically optimizes the data load processes. The application does this through having complete object dependency control across the databases in the data warehouse. This dependency structure gives you, as a developer, control of all objects in your entire data warehouse solution (views, tables, functions and stored procedures). Through the Xpert BI ‘Process Groups’ you have the flexibility to define and divide data update processes on an aggregated level.

Further, the dependency control can be used to perform a change consequence analysis, i.e. what happens if you change a certain table or view, does other elements need to be changed, will a modification to one reporting area affect any other areas etc. This will increase the change implementation quality and save time and resources when implementing modification requests.

3.1.2. Structure and Naming conventions in Xpert BI Transformation

To fully utilize all functionality of Xpert BI Transformation and the Xpert BI methodology of sustainable BI solutions, it is recommended that you use the naming conventions and standardized methods as described in this chapter.

Your data warehouse should be structured in the following way:



Xpert BI automatically installs these databases/schemas. In addition to the standardized Xpert BI databases you can have additional databases which you can add to the solution. See Adding databases to the Xpert BI solution for how to add a database to the solution.

Xpert BI uses the 'ELM' database/schema as a top node in the dependency chain. Xpert BI provides a number of functionalities, including

- Process groups
- Star schema relationships
- Object in use
- Error in code

which requires a specific naming convention and structure. The standard is based on BI development best practices.

The Xpert BI standard:

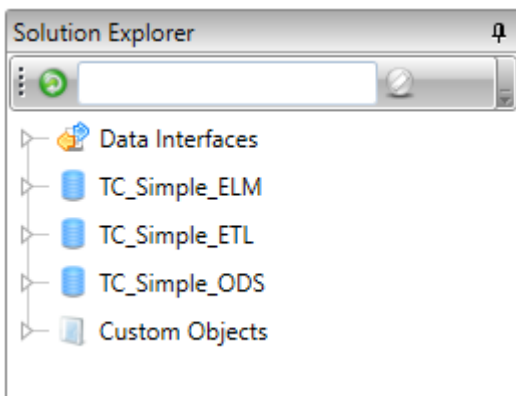
- From a process group perspective, the top node in the dependency structure is normally fact tables in ELM.
- Fact tables/views have the following naming standard: it must have a name that starts with either 'Fact' or 'MG'
- Dimensions are related to Facts by naming similarities on PK and FK columns. (see Specify object derived dependencies for how to connect derived dependencies)
- Fact and dimension tables should have the primary key column in the first column
- PK and FK columns should have the suffix '_Id' or '_Id_<role>'
- Views containing 'Union', cannot directly be converted to a '_Inline' view. (To load such a view into a table, you must create a 'dummy' view that points to your 'union' view)

3.2. Using Xpert BI Transformation

Xpert BI Transformation is both a tool for creating effective data transformations and developing structured and documented data warehouse environments as well as a tool to more efficiently perform maintenance, operation and debugging.

3.2.1. Solution Explorer and Object Dependency Graph

Xpert BI Transformation introduces a new tab in addition to the Data Source Explorer tab, which is the Solution Explorer tab. The Solution Explorer will list all the databases that are included in the BI solution in an object tree. The view is similar to the one you find in Microsoft SSMS.



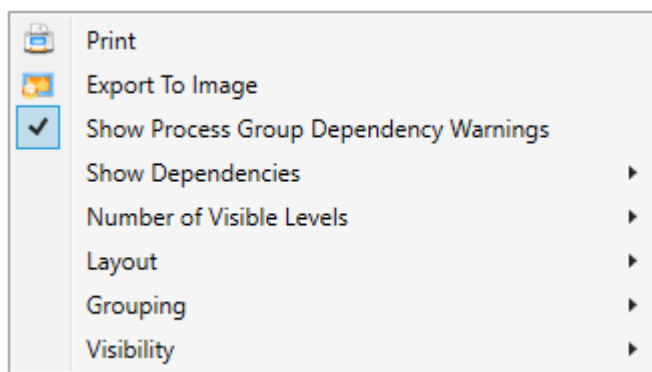
The first node is data interfaces, which contains OLAP Cubes, Tabular Projects and Power BI reports. See Data Interfaces for more details. The next nodes are databases. Expanding databases will bring up the database objects grouped into tables, views, functions and stored procedures. If there are more than one schema in use in the database, they will be shown between the database and the tables/views/etc level. The last node is Custom Objects, see section Custom Object for more details.

Clicking on an object in the solution explorer will bring up a pane in the main window section with a graph showing the object and all its dependencies upwards and downwards as two different hierarchies. Clicking on another object will refresh the graph to focus on the new object. You can also change focus by right-clicking an object in the graph and choosing Set Focus.

The objects in the Solution Explorer have different right-click menus depending on the object type. The objects in the graph also have right-click menus.

Note: The 'back' and 'forward' mouse buttons can be used to navigate between previously focused nodes.

3.2.1.1. Graph menus and icons



A right click *in the white graph background* will bring up a menu showing:

Print

Will print the graph.

Export To Image

Will export a graph snapshot to a PNG file.

Show Process Group Dependency Warnings

Process Group Dependency Warning is a warning icon that appears on the object box when the object is not linked directly or through derived dependency to an object included in any process group. Xpert BI defines an object as “in use” if it is included directly or indirectly in a process group.

Show Dependencies

- All: Show all dependencies, both derived and non-derived.
- Only Derived: Shows only the derived dependencies. Can for example be used to show a fact table and all its related dimensions, without any clutter.
- Only Non-Derived: Show only the non-derived dependencies.

Derived Dependencies are those dependencies that are saved by the user which are not found in the database system tables, but are derived based on naming similarity and user definitions. (See section Specify Object Derived Dependencies for derived dependencies).

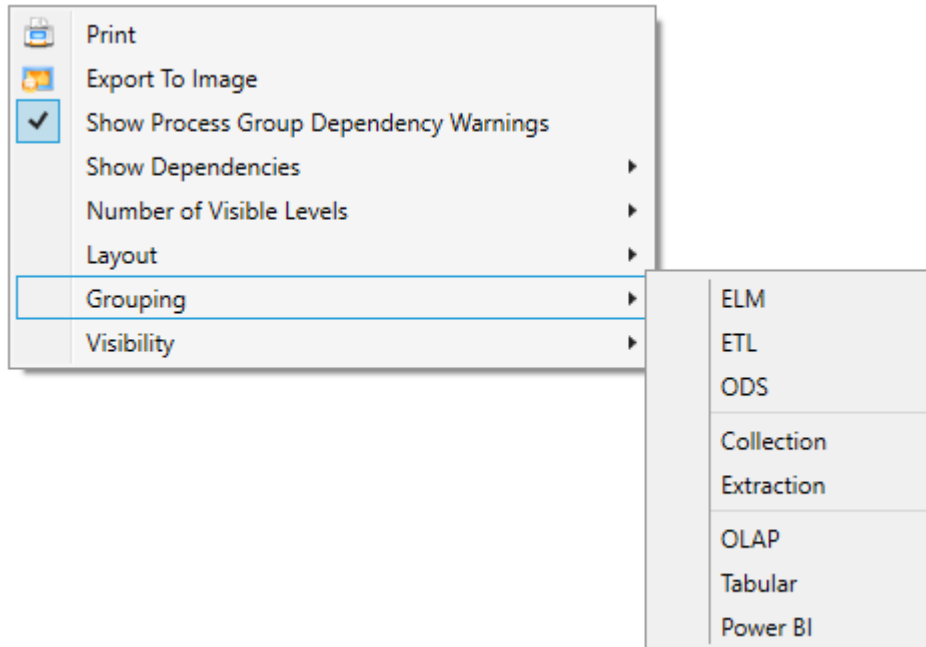
Number of Visible Levels

Sets the number of graph levels to display, where the node in focus is on level 0.

Layout

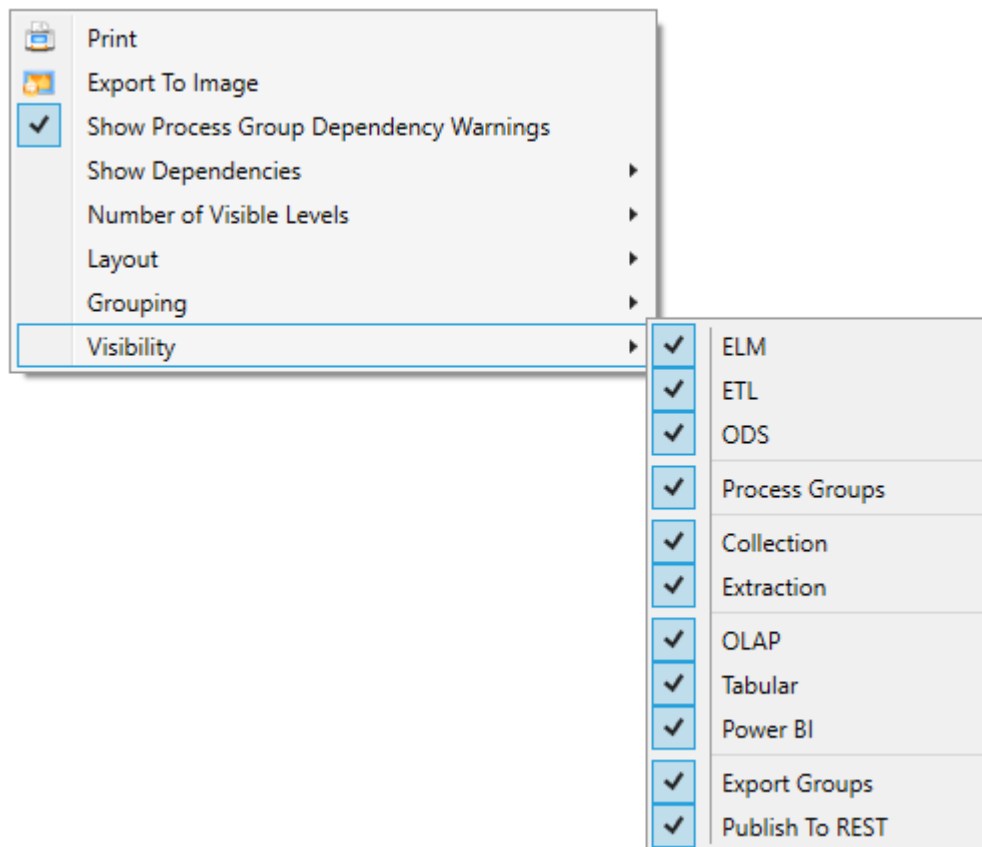
Sets the layout algorithm used to display the dependency graphs. *Tree up-down* uses tree-up for the parent graph and tree-down for the child graph (similar for *Tree right-left*). *Sugiyama* (horizontal and vertical) use the same for both graphs.

Grouping

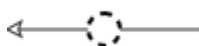


Grouping objects will enclose all objects of a specific type into a single node.

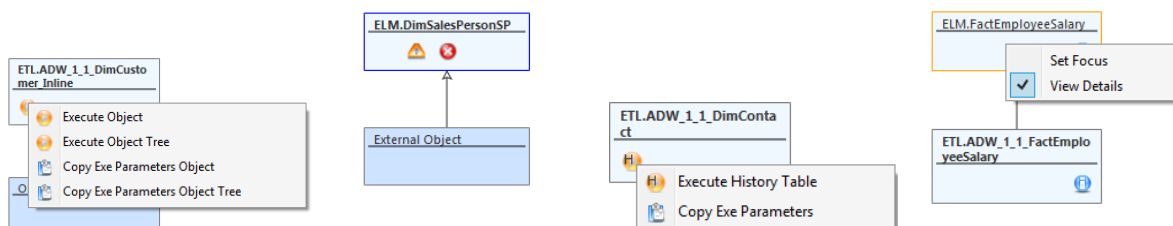
Visibility



Visibility will show or hide objects of a specific type. When there are hidden objects between nodes, the arrow connecting the nodes will be changed to the following type:



A right click *on the objects* will bring up context sensitive menus depending on if you click in the box or on the icons in the lower level of the box:



Execute and **Copy Exe Parameters** will be explained in chapter 2.2.4

An **information icon** appears when the object has a description, or a configured post process.

A **warning icon** appears when the object is not linked directly or through derived dependency to a top node fact table. Xpert BI defines an object as “in use” if it is linked directly or indirectly to a fact table object in the ELM database.

An **error icon** appears when the object is not valid, i.e. that there is something wrong in the object code. That can be that it refers to a table, view or function that does not exist.

Set Focus will change the graph focus object to this object.

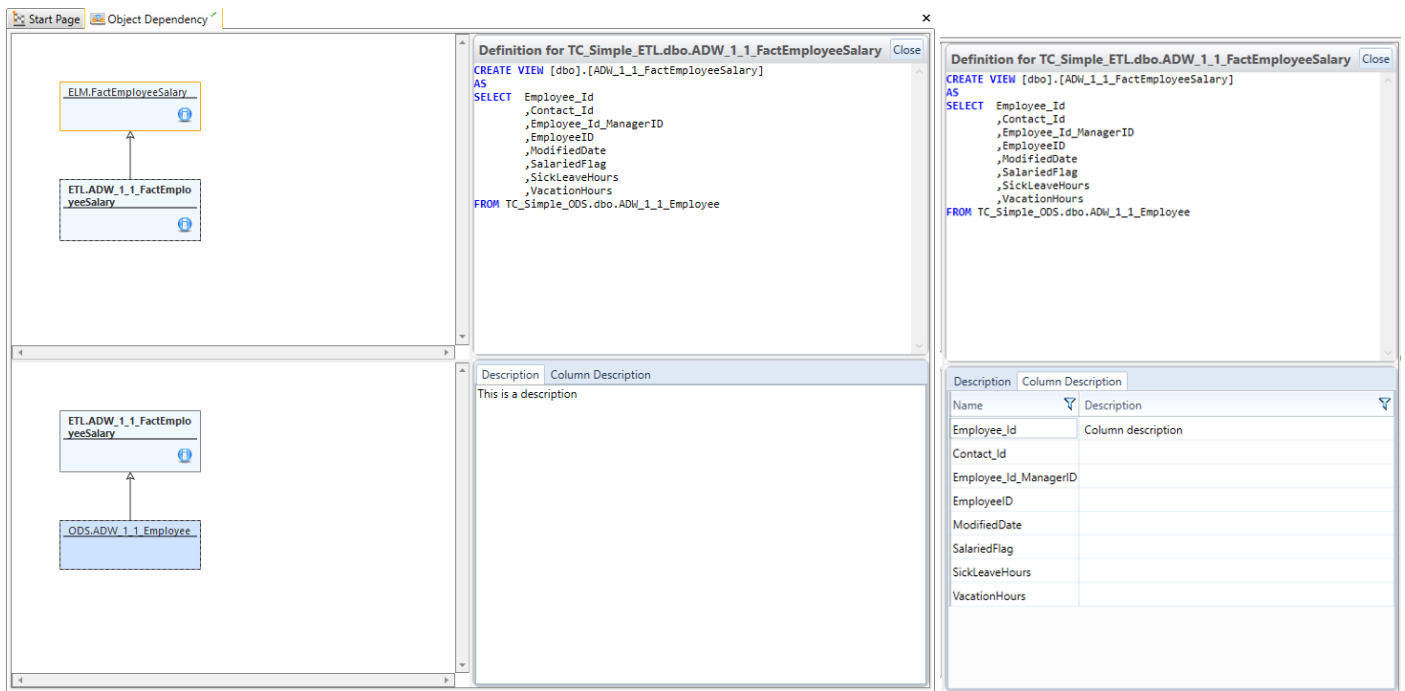
View Details

When View Details is enabled, the right side of the Dependency Graph will show information about the selected object if it has any. The type of information will depend on the type of object. For most objects the details view is split in two, where the top section shows a set of properties and the bottom section shows list-based properties. It is possible to copy the title by right clicking the title.

| Table Name Source | CUSMSTR |
|-------------------------|---|
| Table Name | Customer |
| Description | |
| Comment | |
| Database Type | SQL Server |
| Connection String | provider=SQL Server;Data Source=DESKTOP-1GN0QK6\INST2;Initial |
| Last Successful Process | 11/06/2019 10:53:29 |

| Column Name | Column Name Source | Description |
|---------------|--------------------|-------------|
| AccountNumber | AccountNumber | |
| CustomerID | CustomerID | |
| ModifiedDate | ModDate | |
| rowguid | rowguid | |
| TerritoryID | TerrID | |
| CustomerType | Type | |

For SQL types the SQL object definition is shown on the top, and the description and column descriptions on the bottom. The object description and column descriptions are editable and is saved when the user clicks outside the description box (when the description box loses focus). Note that ODS tables will show table and column descriptions that has been set in extraction (Edit Metadata) instead. These descriptions are not editable from the details view. When an object description is added, an 'i' icon will appear to indicate that the object has a description. The SQL object definition is also editable, but it is NOT saved under any circumstances.




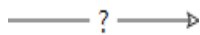
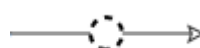
Double-clicking an object in the Dependency Graph will open View Details if not already open, but will NOT change focus to that object. Single clicking an object while View Details is open will show the details for that object.

Clicking the Close button closes the details view.

3.2.1.2. Graph Legend

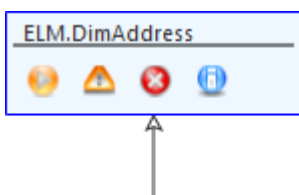
The nodes in the dependency graph can be different types with different symbols and colors, this section has information about these.


Arrow Types

-  Normal connection between two objects
-  The dependency between these nodes could not be determined uniquely. This can happen if a Power BI report is synced, but not downloaded, and it contains an Analysis Service connection where the project name exists in both Tabular and OLAP.
-  There are hidden objects between these two nodes.

Node Icons

Each node can have several types of icons and they are displayed as shown:



-  This object can be executed. Right click the icon to show execution options.



This object has a warning attached to it. Mouse over to get a tooltip text explaining the problem.



This object is not valid.







This object has a description. Mouse over to show description.

Node Color

The nodes can have different background colors and borders based on what type of object it is.

| Node Type | Color | Example Node |
|---|-------|--|
| Table | | |
| View SQL Group | | Note: Blue border for dimensions Note: Orange border for fact table |
| Stored Procedure | | |
| Function Custom Object | | |
| Mds Entity | | |
| Extraction Collection Export Group Publish To Rest | | |
| Tabular OLAP | | |
| Power BI | | Report Dataflow Dataset Dataset Table Dataflow Table |

| | | |
|---|---|---|
| Process Group |  | <div>All</div> <div></div> |
| Azure Data Lake Azure Data Lake Folder Table |  | <div>Company</div> |
| Azure Data Lake Blob |  | <div>Transactions CSV Blob</div> |

3.2.1.3. Solution Explorer menus

The right click menus displayed on objects in Solution Explorer is context sensitive, but the general menu items are:

Refresh on database or object group level will run a reload and refresh the listed object in the selected structure.

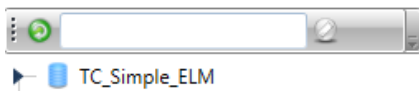
Delete will delete the object from the database. Xpert BI creates a backup script with the object definitions. This script is saved in the Xpert BI Management Database in the 'Object_Dropped' table. The description of the object is also saved.

Copy Name will copy the object name into the clipboard.

The other menu items will be explained in chapter 2.2.3

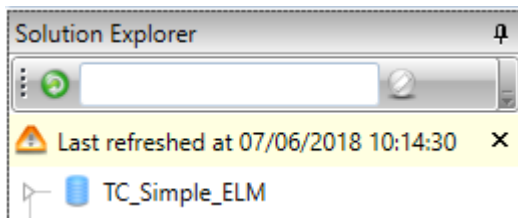
3.2.1.4. Reload Solution Explorer

When clicking the **reload** button, warning and error icons are reloaded, as well as added/removed databases, object dependencies and derived dependencies.



To reload the tabbed pane (i.e. the Object Dependency pane) you must also click reload from the Ribbon. The Ribbon reload is context sensitive to the pane in focus.

If the Solution Explorer has not been fully refreshed for 7 days, a warning will show at startup indicating how long ago last refresh was performed.

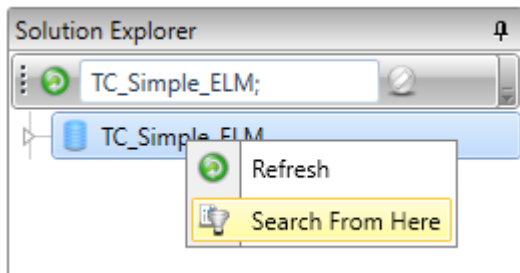


If the option for "Auto Refresh Solution Explorer" is disabled, simply press the reload button to force a refresh of everything.

3.2.1.5. Solution Explorer Filter/Search

The Solution Explorer toolbar has a text field for filtering the objects shown in the tree structure. The filter is case insensitive and is implemented as 'contains'. This filter can for example be used to view all Fact objects, or to search for a specific object by entering a part of the object name. The filter applies to all databases in the solution by default.

It is possible to restrict searching to a single database; this can be done by right-clicking desired database and selecting "Search From Here" from menu (keyboard shortcut: CTRL+F).



This will insert the database name into the search field. To perform a single database search, simply append the search phrase after the semicolon. For instance, TC_Simple_ELM;xbi will search for all words containing "xbi" inside the TC_Simple_ELM database.

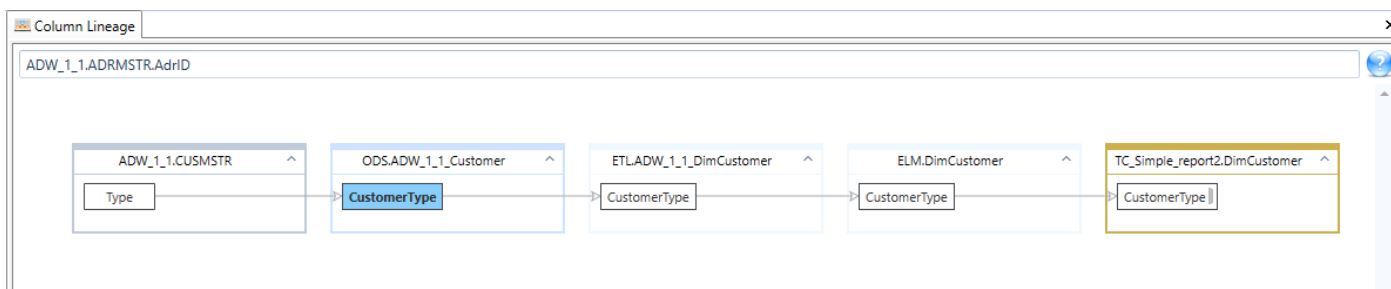
To the right of the text field there is a Clear-button used to clear the filter text (pressing "Escape" does the same). If an item in the tree is selected when this happens, it will stay selected and the tree will be expanded down to it. If no item is selected (or the selected item is a top-level) all items will be collapsed.

The search term must contain a minimum of two characters.

3.2.2. Column Lineage

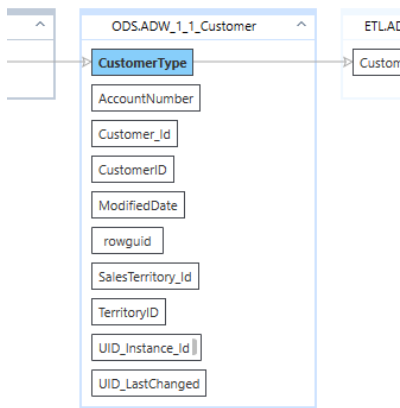
Clicking **Column Lineage** from the Transformation ribbon bar will open the column lineage graph. The search field allows you to search for columns using column name, object name and object type, or any combination of these. For more details on how the search works, hover the mouse over the blue question mark to the right of the search field.

When a column is selected, all its dependencies will be drawn, as seen in an example below. Here the column **CustomerType**, in the object ADW_1_1_Customer is in focus (as seen by the blue colouring and the bold font on the column name). The children of the column in focus are to the left, and the parents to the right. The type of the object can be seen in the mouse tooltip or by looking at the colour of the border. The border colours match those from the object dependency graph.



Right-clicking on any column opens a context menu, which allows you to **Set Focus** to a new column. Doing this will re-draw the graph with the new column in focus.

Right-clicking an object opens a context menu which allows you to **Show All Columns**. Clicking this will add all the columns in the selected object to it, as seen in the following example.



It is then easy to change focus to a different column in the object.

For SQL objects (views, tables etc) the right-click menu also contains an option to navigate to the dependency graph (**View Dependency Graph**), which will open the dependency graph with the selected object in focus.

Note the grey indicator on the right side of the UID_Instance_Id column above. This indicates that the column is **Top-Level**. Similarly, an indicator to the left indicates that the column is **Only Inbound**, meaning that the column is not available as a select column, but it is still used in the object (for example in a join or a where statement).

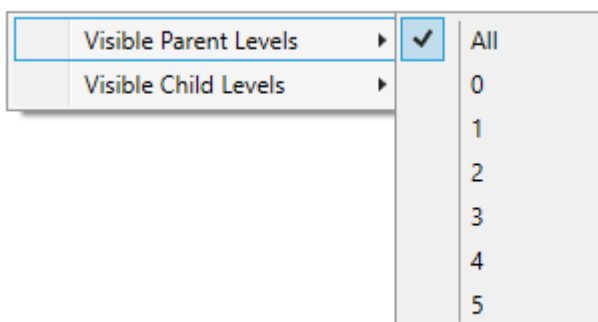
The column lineage graph can also be opened with focus directly on a column by right-clicking any column description column in the view details pane.

Note that for calculated tables in imported SSAS Tabular models (via Data Interfaces), the table expression is not interpreted, meaning that in some cases the column lineage will not be complete for columns in calculated tables.

Note that column lineage is not supported for SSAS OLAP Multidimensional cubes imported via Data Interfaces and for Custom Objects.

3.2.2.1. Visible Levels

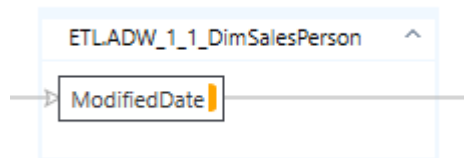
It is possible to control how many levels are visible from the node in focus. Parent levels and children levels are controlled individually. To change the level, right click on the column lineage background to bring up the following menu.



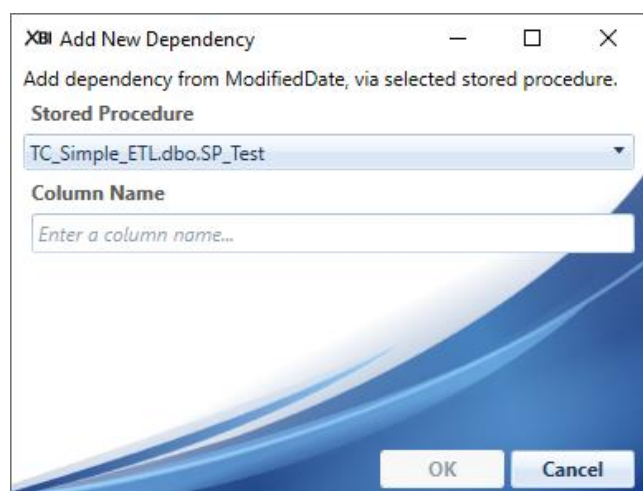
Note that the selected level is stored as a user setting. The visible levels will not be reset to All even if Xpert BI is restarted.

3.2.2.2. Stored Procedures

Column lineage is not automatic through stored procedures, because of their dynamic possibilities. The columns that are used by a stored procedure is marked with an orange indicator on the right side, as seen below. Similarly, columns that comes from a stored procedure (column in a stored procedure generated table for example) are marked with an orange indicator on the left side. The tooltip on the column will list which stored procedure(s) it is.



Xpert BI allows you to manually add the desired column dependencies through the stored procedures. You can do this by right-clicking a column with an orange indicator and clicking **Add Dependency**. Doing this will open a window as seen below. The **Column Name** text box is an auto-complete box with the available columns. Selecting one and clicking **OK** will add a dependency from the initially clicked column, through the stored procedure and to the selected **Column Name**. This dependency will then appear as a dependency in the column lineage graph.



Note that manually defined dependencies only apply to Column Lineage, they are ignored in the Object Dependency Graph.

3.2.3. Object dependency configuration background

3.2.3.1. How Xpert BI achieves Object Dependency control

This chapter explains the theory behind the Object Dependency Control. View chapter 2.2.3 on how to implement/develop this standard.

Full dependency control on views, tables, functions and stored procedures across databases in the BI solution is achieved through using the naming convention described below. This will bypass using any external data transformation/load tool to load data and thereby losing dependency control.

Loading data into a table from a view

To load data into a table in the ETL process, you should create a view that holds the data you want to load. The name of this view will be the name of the destination table. Xpert BI will rename your view to contain a suffix (`_Inline`) and create a table with the same columns and the same name as the original view. In Xpert BI you will be able to specify key column(s) and add a surrogate key if desired. The resulting naming convention is as follows. Any view that has the suffix "`_Inline`" is a basis for a table with the same name but excluding the "`_Inline`"

Note: You should not refer directly to "`xxx_Inline`" views from any other views or stored procedures; you should refer to the destination table which has the same name as the view but excluding the suffix "`_Inline`"

Loading data into a table from a stored procedure

To include a Stored Procedure in the Xpert BI object dependency, the procedure with all its arguments must be specified in a view format (through Xpert BI). In addition, the Stored Procedure can only have ONE output table and must have a parameter (specified through Xpert BI) that holds the destination table name.

There is no restriction on the number of arguments the procedure can contain or to the number of times it can be referenced.

3.2.3.2. Using data source prefixes

All your tables have a data source-, version- and instance prefix in ODS. This is to be able to group and sort the tables from different data sources within the same database. It is recommended that the table prefixes are kept also in ETL for the same purpose unless you are combining data sources. It is recommended that ELM contains only Fact and Dimension views without any data source prefix unless needed to maintain uniqueness, and that ELM does not contain any business logic, calculations or filters.

3.2.4. Transformation and data processing design

Xpert BI uses a methodology and process called "Inline Transformation Services" (ITS) to be able to create data transformation loads, create and maintain dependencies, and configure data update schedules.

3.2.4.1. Create ITS table (from views and stored procedures)

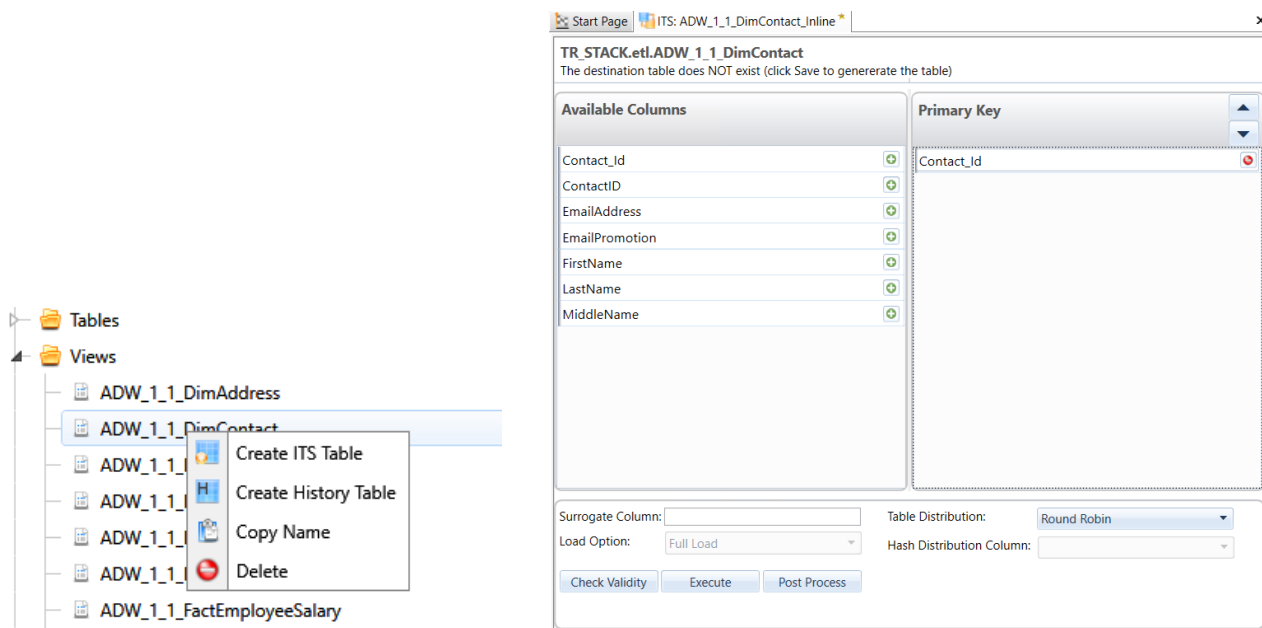
Xpert BI defines an ITS table as a table that will be updated with data from either a view or a stored procedure at scheduled intervals. The table must have a primary key and be configured using the Xpert BI structure and naming convention.

Create ITS table from a view

Any view in the ETL process can be transformed to an ITS table by using Xpert BI without losing its link in the object dependency graph as long as it fulfils the following criteria:

- No duplicates, i.e. the view must have a logical primary key column(s). Can be composite key
- Primary key column(s) cannot be NULL
- The view query cannot contain a UNION statement unless it is in a sub query.

To create an ITS table from a view simply right-click on the view in Solution Explorer and click “Create ITS Table”. This will bring up a new pane where you configure the table details.



Xpert BI will rename the view to include a suffix “_Inline” and create a table with the original view name. The table will hold the same columns as the view in the same order. The user must specify the key column(s) in Xpert BI and also the order of the key columns. Click Save to save the key configurations.

Xpert BI adds an “Order By” statement to the original view. If the view already has an “Order By” statement, the sorted columns will be listed as suggested key columns.

Note: If, for some reason no key column(s) are specified an error will be shown when saving. The ITS table will not be created (it will however be created without PK if the Inline object is executed).

You can also specify to add a Surrogate Column, i.e. a new identity column with increment=1 is added to the destination table. When a surrogate column is specified you have two load options; Full or Incremental. The Full load will give the identity column name a suffix of Id_X (Identity exchange) and the incremental load will give the identity column name a suffix of “Id_I” (Identity incremental).

When making changes to the configurations you must Save the changes for it to be implemented.

To check if the key you have specified is valid, click “Check Validity”. You need a valid key to be able to save the configurations.

You can edit the configurations at any time by right clicking the “_Inline” view and choose Edit.

Clicking "Post Process" opens a window where you can enter SQL code that will run directly after the inline is done processing. This can for example be used to add custom indexes to the inline table. If the post process fails it will log a warning but continue processing. An ‘i’ icon will appear on the object in the dependency graph to indicate that the object has a post process.

Table Distribution lets you choose the distribution of the ITS table. If you choose Hash, then the Hash Distribution Column lets you choose the hash column. This functionality is only available for Azure Synapse SQL. Note that if the inline has Load Option = Incremental Load, changing the distribution will not take effect (the table has to be re-created manually).

Note that even though the primary key is not added to the output table if the table is on Azure Synapse SQL, it is still a required parameter in Xpert BI.

Deleting ITS tables from views

Right-click on an “_Inline” view object and choose “Delete ITS Table” to delete the table and remove the “_Inline” suffix from the view. This will remove the executable object from all update processes.

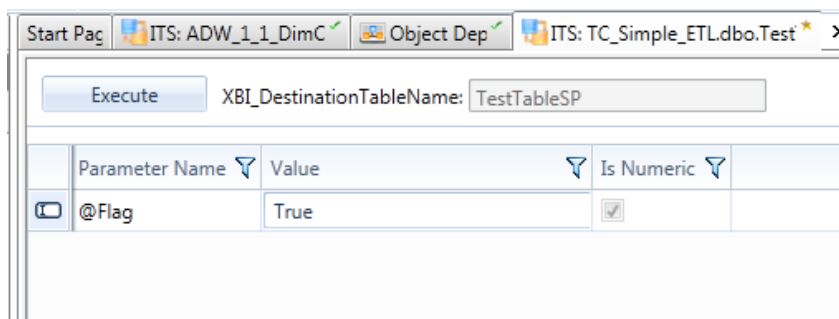
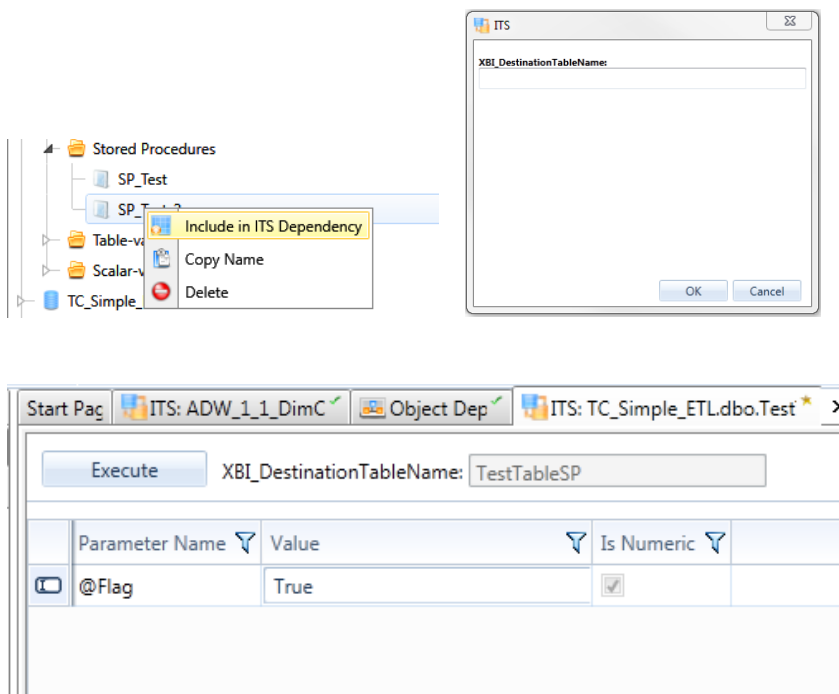
Create ITS table from a stored procedure

A stored procedure can be included in the Xpert BI ITS structure when it meets the following requirements:

- The result/output is one output table given in the required parameter name: XBI_DestinationTableName
- Input tables/view cannot use dynamic SQL

The method Xpert BI uses to include stored procedures in the automatic update structure and dependency documentation is to document the stored procedure in a view format.

To include a stored procedure in the Xpert BI structure, right-click on the stored procedure in the Solution Explorer and choose “Include in ITS Dependency”. This will bring up a window asking you to name the XBI_DestinationTableName followed by a new pane where you can configure the remaining parameters.



Fill in the name of your destination table and click OK. This will bring up a pane listing the other parameters, if any, for you to fill in. Xpert BI does not support single quotation marks in the Value field.

Click Save to save the Stored Procedure ITS configuration and then Execute to load the data. The execution will load data into the table through the Xpert BI Data Processing task (in a new pane).

For Xpert BI to log insert/update information on the execution of InlineSP objects, the Stored Procedure must include return parameters that contain this information. The return parameters that are recognized by Xpert BI, and will be included in the Event log are:

- XBI_RowsInserted
- XBI_RowsUpdated
- XBI_RowsDeleted
- XBI_ErrorMessage

(You need to include **all** or **none** of these)

Note: Editing the ITS table requires that you ‘Execute’ the ‘_InlineSP’ view, i.e. process the table for the changes to take effect.

Note: Deleting the “_InlineSP” view will not delete the destination table. This must be done manually.

3.2.4.2. Specify object derived dependencies

Xpert BI ITS uses defined structures and naming conventions to derive logical relationships between objects in the Solution databases. It also allows users to specify dependencies i.e. logical relationships in ELM to compliment or overwrite the derived dependencies. Combining the derived structure(s) with the user-defined dependencies, Xpert BI then holds a complete dependency structure that is used in the Solution Explorer graph and updating routines.

Dependencies are, in most cases, dimension table(s) that relates to fact table(s), but where either one or both of the objects are defined as views, and therefore has no “real” defined relationship in the database. Xpert BI provides a method for setting up these logical relationships. Xpert BI focuses on the fact table(s) defined in ELM, i.e. the centre object of the star in the star diagram/multidimensional model.

To define a derived dependency, click on the “Fact table dependencies” icon on the ribbon. This will bring up a new tab in the main window that lets you define dependencies.

| Start Page | | Fact Table Dependencies | | |
|------------|-----------|--|---------------------|-----------------------|
| | Database | Object Name | | |
| > | ELM | FactEmployeeSalary | | |
| | ELM | DimEmployee | | |
| | ELM | FactSalesOrder | | |
| | | | | |
| | Is Active | Foreign Key Column | Dependent Dimension | Dimension Column Name |
| > | | <input checked="" type="checkbox"/> Address_Id_BillToAddressID | DimAddress | Address_Id |
| | | <input checked="" type="checkbox"/> Address_Id_ShipToAddressID | DimAddress | Address_Id |
| | | <input checked="" type="checkbox"/> Contact_Id | DimContact | Contact_Id |
| | | <input type="checkbox"/> CreditCard_Id | | |
| | | <input type="checkbox"/> CurrencyRate_Id | | |
| | | <input checked="" type="checkbox"/> Customer_Id | DimCustomer | Customer_Id |
| | | <input checked="" type="checkbox"/> Product_Id | DimProduct | Product_Id |
| | | <input type="checkbox"/> SalesOrderDetail_Id | | |
| | | <input type="checkbox"/> SalesOrderHeader_Id | | |
| | | <input checked="" type="checkbox"/> SalesPerson_Id | DimSalesPerson | SalesPerson_Id |
| | | | DimSalesPersonSP | SalesPerson_Id |
| | | <input type="checkbox"/> SalesTerritory_Id | | |
| | | <input checked="" type="checkbox"/> Time_Id_Due | DimTime | Time_Id |

The top grid shows all Fact- and Dimension-objects that in its definition have columns with suffix “_Id” in other placements than as the first column. The warning icon that appears on the leftmost column is an indicator that the user must resolve/confirm dependencies for that object.

The bottom grid shows all columns that contains “_Id” in the selected fact table as well as suggested related/dependent dimension. This suggestion is made on the basis of name similarity.

There is also an “Is Active” check-box which is used by Xpert BI as an indicator for whether the column is in use as a foreign reference. If the column is active Xpert BI will attempt to suggest a dependent dimension and you must define a dependent dimension for the column in order to remove the warning icon on the selected object. If the column is not active Xpert BI will not make suggestion(s) for dependent dimension(s) and you can leave the dependent dimension column blank.

To delete the suggested dependencies you must first save the suggestion. Saving Fact Table Dependencies will bring up a delete icon next to the suggestions which enables the possibility of deleting dependencies. You can add/delete as many dependencies as desired for each “_Id” column.

| Start Page Fact Table Dependencies | | | | |
|------------------------------------|-------------------------------------|----------------------------|---------------------|-----------------------|
| | Object Name | | | |
| > | FactEmployeeSalary | | | |
| | DimEmployee | | | |
| | FactSalesOrder | | | |
| | | | | |
| | Is Active | Foreign Key Column | Dependent Dimension | Dimension Column Name |
| > | <input checked="" type="checkbox"/> | Address_Id_BillToAddressID | DimAddress | Address_Id |
| | <input checked="" type="checkbox"/> | Address_Id_ShipToAddressID | DimAddress | Address_Id |
| | <input checked="" type="checkbox"/> | Contact_Id | DimContact | Contact_Id |
| | <input type="checkbox"/> | CreditCard_Id | | |
| | <input type="checkbox"/> | CurrencyRate_Id | | |
| | <input checked="" type="checkbox"/> | Customer_Id | DimCustomer | Customer_Id |
| | <input checked="" type="checkbox"/> | Product_Id | DimProduct | Product_Id |
| | <input type="checkbox"/> | SalesOrderDetail_Id | | |
| | <input type="checkbox"/> | SalesOrderHeader_Id | | |
| | <input checked="" type="checkbox"/> | SalesPerson_Id | DimSalesPersonSP | SalesPerson_Id |
| | | | DimSalesPerson | SalesPerson_Id |
| | <input type="checkbox"/> | SalesTerritory_Id | | |
| | <input checked="" type="checkbox"/> | Time_Id_Due | DimTime | Time_Id |

Note: To get a correct Object Name list in the Fact Table Dependencies, make sure all fact and dimension objects has the key column as the first column and that if the object has other key columns, these are not the first column. A key column is defined as a column that contains “_Id”

Fact Table Dependencies with Many-to-Many/Bridge tables

For Xpert BI to identify Bridge tables used to resolve Many-to-Many relationships, the table or view needs to start with Dim and have “Bridge” or “bridge” in the object name. When identifying Bridge tables, Xpert BI will visualize all “_Id” columns in the list of “Foreign Key Columns” for Fact Table Dependencies. Resolving the relationships for the Bridge table requires two steps; First, click on the related Fact Table and connect this table to the Bridge table (even though this will technically be a relationship in the wrong direction), next, click on the Bridge table and connect this table to the related dimension table.

The object dependency graph will now show Fact → Bridge → Dim, which is correct in this context.

3.2.4.3. Specify process groups

Xpert BI Transformation introduces a new concept: Process Groups, which allows grouping tables (usually fact tables) that are to be updated at the same frequency.

The process group includes the **selected solution object** and **all its dependencies** (for fact tables it includes the objects defined in the Fact Table Dependencies) and **all its children** in the object dependency graph and **all children of dependent objects**. If a process group contains several objects which has overlapping depending objects, all duplicate objects are eliminated so that every object is only processed once. The object dependency model ensures that objects are updated in the right order and runs parallel processing where possible.

To define a process group, click on the “Process Group” icon in the ribbon. This will open a new tab in the main window that lets you define a process group.

Start Page

Process Groups

| <div>+</div> | Process Group Name | Description | Execute Group | Copy Exe Param | Delete Group | |
|--------------|--------------------|---------------|---------------|----------------|--------------|--|
| > | All | A group which | | | | |
| | Sales | | | | | |

Solution objects

| | Object Name | Database | Schema | Top Level |
|--------------|----------------------------------|----------|--------|-------------------------------------|
| <div>+</div> | DimCustomer | ELM | dbo | <input checked="" type="checkbox"/> |
| <div>+</div> | ADW_1_1_FactSalesOrder_Prepare_1 | ETL | dbo | <input checked="" type="checkbox"/> |
| <div>+</div> | CurrencyTesting | ETL | dbo | <input checked="" type="checkbox"/> |

Included objects

| | Object Name |
|--------------|----------------------------|
| <div>+</div> | ELM.dbo.FactEmployeeSalary |
| <div>+</div> | ELM.dbo.FactSalesOrder |

Xpert BI will always include an “All” process group which will include all fact table objects and their direct and derived dependencies. Xpert BI loops through all dependency chains and finds all executable objects to be included in the ITS update process.

To Add a new Process Group, click the Add button (Plus sign at the upper left corner), specify a name for your group, then add fact or top level objects by clicking the green plus button to the left of the object names you want to include. Click Save to save your changes to the process group. You can add or remove objects from the process group at any time. The objects included in the selected process group are seen in the included objects list. To remove objects click the red delete button to the left of the object you want to delete. Note that the solution objects list will only contain **fact** or **top level** objects. The included objects for the selected process group will not be displayed in the solution objects list.






To view the objects that will be executed at the execution of the group, click the Execute Group icon. This will bring up the standard data processing window and you can expand the group to see all the ITS objects.

To schedule a processing group, click Copy Exe Param for the specific group. This will copy the command to the batch file that can be executed from an external scheduler (SQL Server Agent, SSIS etc).

Note: Different Fact table objects can have derived dependencies to the same dimension objects, thus dimension objects may be included in several process groups. It is recommended that you spend some time to analyse the contents of your process groups and the timing of your updates.

3.2.4.4. Executable Settings



Click the **Executable Settings** button on the Transformation ribbon bar to open the Executable Settings pane. This allows you to override the default behaviour for when executable objects are executed. The default behaviour is that all executables are processed for all process groups (where they are included).

| Executable Settings ✓ | | | | |
|---|--|--|---|---|
|  | Name |  Always Include In Process Groups | Comment  | Delete |
| > | TC_Simple_ETL.dbo.ADW_1_1_DimEmployee_Inline | <input checked="" type="checkbox"/> | |  |
| | TC_Simple_ETL.dbo.ADW_1_1_DimProduct_Inline | <input type="checkbox"/> | |  |

To add a setting (override default behaviour), click the green plus icon in the top left corner of the pane. This will open a window where you can enter the name of an executable. Choose one and click ok.

To view the dependency graph of an executable, right click the executable and click “View Dependency Graph” context menu item.

If you want to skip processing for certain process groups, select the executable in the top grid, and in the bottom grid uncheck the **Include In Process Group** checkbox for the desired process groups. This means that for the unchecked process groups, the executable will be skipped during processing.

| Settings for TCSimple_ETL.dbo.ADW_1_1_DimProduct_Inline | | | |
|---|-------------------------------------|---|--|
| Process Group  | Include In Process Group | Comment  | |
| MyPG | <input checked="" type="checkbox"/> | | |
| All | <input type="checkbox"/> | | |

If you want to *only* execute the object in certain groups, uncheck the **Always Include In Process Groups** checkbox in the top grid and check the **Include In Process Group** checkbox for the desired process groups in the bottom grid. This means that the executable will ONLY be process for the checked process groups, for all others it will be skipped.

3.2.4.5. Adding databases to the Xpert BI solution

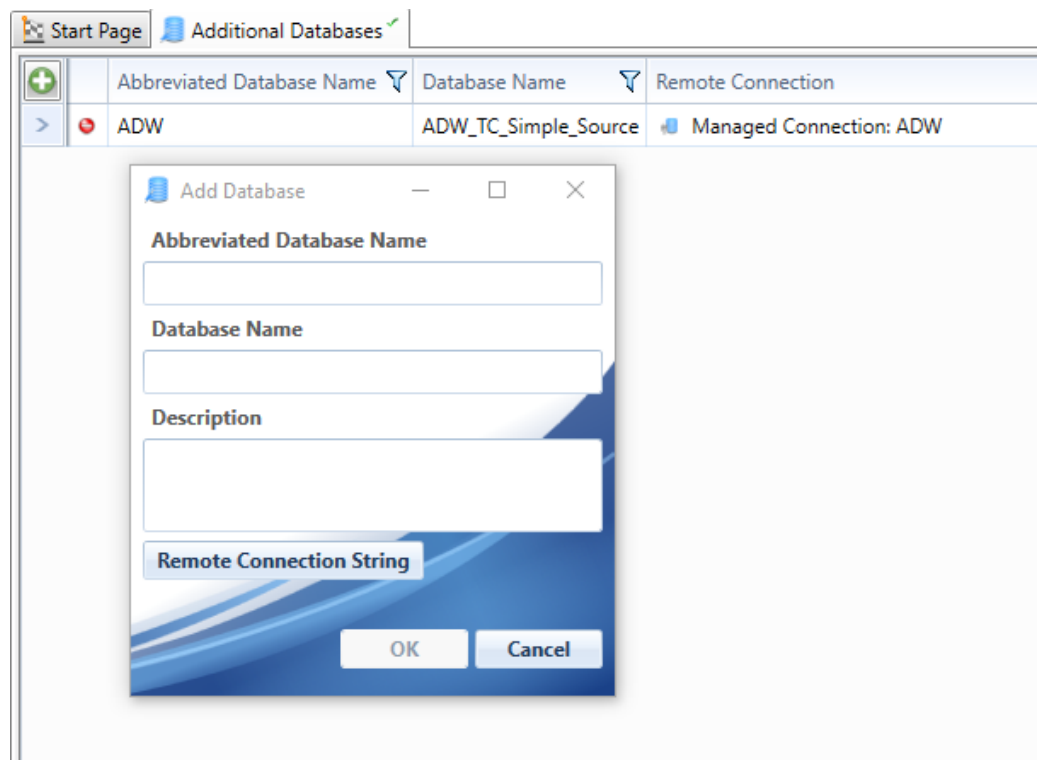
In addition to the stack database(s), you can specify your own databases to be included in the solution. Including a database means including it into the ITS structure which will make ITS functionality available for that database.

To add a database to the solution, click Additional Databases on the Transformation Ribbon menu, and click the Add icon on the top left corner. Databases in the solution have an Abbreviated Database Name and a Database Name. The Abbreviated Database Name is used in object references e.g. in the Object Dependency graph and must be unique within one Xpert BI solution.

As a default, additional databases must be on the same server instance as the rest of your solution, but it is also possible to add databases on other instances/servers. To do this, click the **Remote Connection String** button in the **Add Database** window and setup the connection to the desired database. It is possible to edit this connection later by clicking the edit connection button in the **Remote Connection** column.

Note that a remote connection string should only be used for databases that are NOT on the same server/instance as the rest of your solution.

A remote connection is supported to the following database types: SQL Server, Azure Synapse SQL, Azure SQL Database and Azure Managed Instance.



Note: All additional databases must have the same collation as the Xpert BI Solution.

3.2.4.6. Incremental data load

In the same way as you set incremental data loads in Extraction you can also set incremental data loads in Transformation on ITS tables generated from views. This is generally done on larger transaction tables where you do not wish to load the whole table every time.

To set incremental data loads click on the Set Incremental Filters from the Filters tool group in the Transformation ribbon. This will bring up a list of your ITS view objects. Next you choose the column in which to do the filter selection for the incremental load and specify the overlap, if any.

Click Save to save the changes.

| Start Page | | Set Incremental Filters | | | | | | |
|------------|---|-------------------------|-----------|---|------------|---------|---------------|----------------------|
| | Name | Incremental Column | Data Type | | Last Value | Overlap | Rows Affected | Execution Time (sec) |
| > | TC_Simple_ETL.dbo.ADW_1_1_FactSalesOrder_Inline | ModifiedDate | date | > | | | 0 | 0 |
| | TC_Simple_ETL.dbo.ADW_1_1_DimEmployee_Inline | CreditCard_Id | | > | | | 0 | 0 |
| | TC_Simple_ETL.dbo.ADW_1_1_DimCustomer_Inline | CurrencyRate_Id | | > | | | 0 | 0 |
| | | Customer_Id | | | | | | |
| | | DueDate | | | | | | |
| | | ModifiedDate | | | | | | |
| | | OrderDate | | | | | | |

Note: If you want incremental data loads from ITS stored procedures, this must be handled within the stored procedure itself

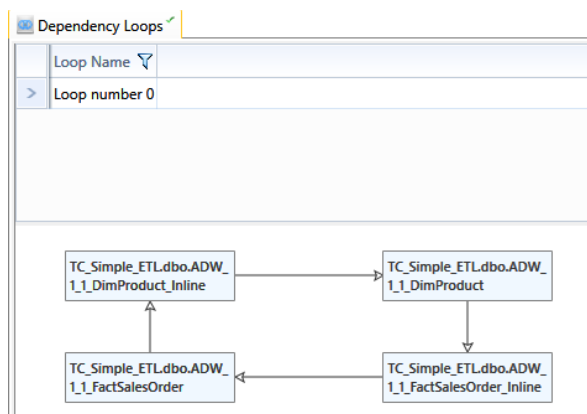
Note: If you do not have any objects on the list it is either because you have not created any ITS view objects or because the view objects are created from script and not from the application (i.e. with migrations). To include scripted ITS objects in this list please view chapter 2.2.3.3 under Notes.

Note: If the incremental column is of date type, the last value will never be set higher than the current date. This is to make sure source data input errors will not block the incremental load.

Note: Incremental loads (via incremental filters) is not support on Azure Synapse SQL.

3.2.4.7. Dependency Loops

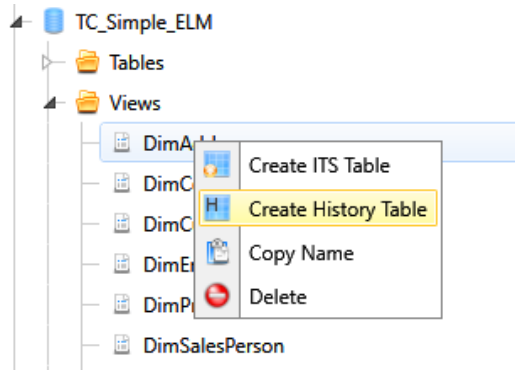
The Dependency Loops button are only enabled when one or more dependency loops are detected. Loops are detected by Xpert BI when reloading the Solution Explorer. Clicking the button will open the pane seen below. At the top a list of the loops is shown. Clicking a loop name will open a graph showing the elements that make up the loop. If a dependency in a loop is a derived dependency, the arrow between the two elements will be orange.



3.2.4.8. Create History Tables

A History Table is used to create a historical snapshot based on a table or view. The output is a table postfixed with "_History", which is created in the DHM database/schema (schema if stack or database is schema-based, database otherwise). The database/schema is created if needed. The output table has all columns from the source as well as a *XBI_HistoryDate* column. The idea behind this functionality is to create historical snapshots of important data that can then be used to later see what the data foundation for a report was at a given time.

To create a history table, right click a table or view in the Solution Explorer and click **Create History Table**.



This opens a pane where you can select which columns should be primary key in the resulting table (the *XBI_HistoryDate* column is automatically included as part of the primary key).

Clicking **Copy Exe Parameters** will copy the command line exe parameters to the clipboard. These can be used to schedule the historical snapshot.

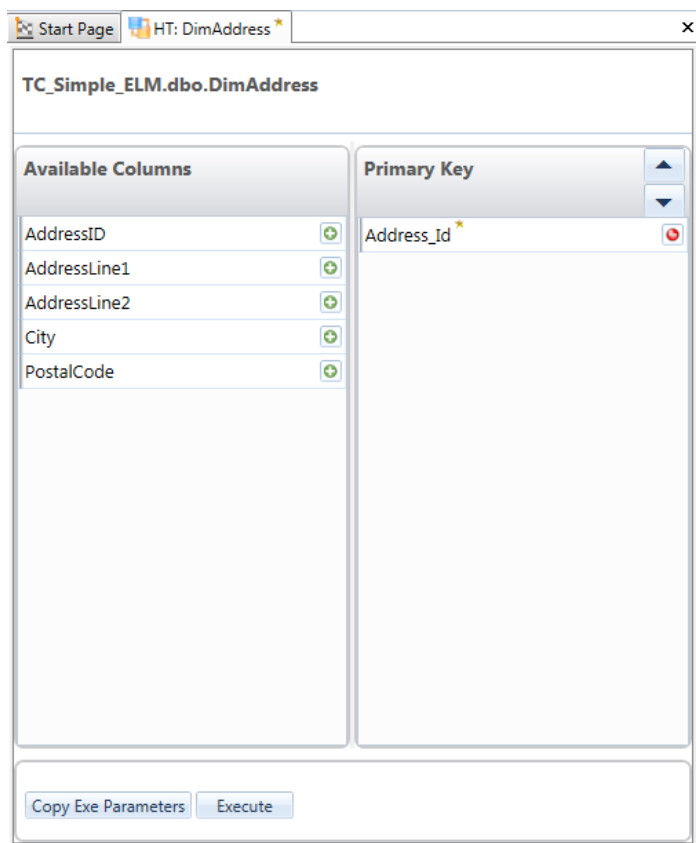
Clicking **Execute** will open a processing pane to execute the history table.

Note: Copy Exe Parameters and Execute

Clicking **Save** will create the output table and save the metadata, without processing it. If the table already exists, it will be **dropped** and re-created.

If a new column is added to the source table/view it will NOT automatically be included in the history table output. If a column is removed from the source the processing will fail - this will have to be handled manually.

Note that History tables are only available on SQL Server and Azure SQL Database.

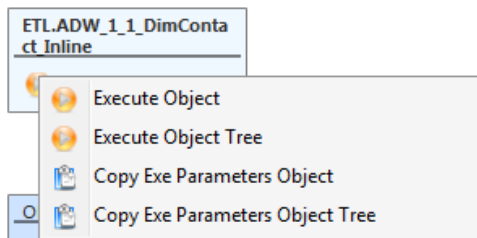


To remove a history table, right click a history table or view in the Solution Explorer and click **Remove History Table**. Removing a history table from the solution will NOT delete the output table in the DMH database.

3.2.5. Data load options

Data loads can be done through the Solution Explorer (one object), Object Dependency graph or Process Groups window during development and/or debugging, or in operation through the batch process as a scheduled task on either objects or process groups.

3.2.5.1. Execute object



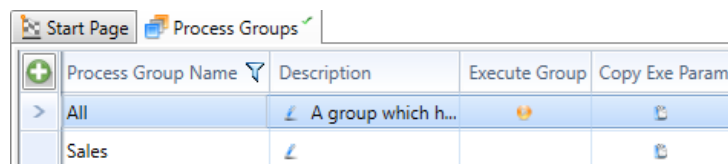
To load data from an Inline or InlineSP view into a table, right-click on the view in the Solution Explorer and choose Execute Object or right-click on the “Execute” icon in an object in the Object Dependency graph and choose Execute. This will open the standard data processing window where you click yes in the confirmation dialog or Execute from the ribbon to start the data load.

To schedule loads into one specific table choose Copy Exe Parameters Object from the Object Dependency Graph and schedule the batch process from any scheduler.

3.2.5.2. Execute object tree

Right-click on a “Execute” icon in an object in the Object Dependency graph, and choose Execute object tree to process all executable objects downwards from the selected object.

3.2.5.3. Execute process group

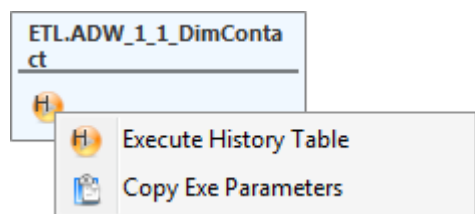


| Process Group Name | Description | Execute Group | Copy Exe Param |
|--------------------|--------------------|---------------|----------------|
| All | A group which h... | | |
| Sales | | | |

To execute a process group interactively from Xpert BI, click on the process group Icon in the General tool group and click on the Execute Group icon. This will open the Operation Data Processing window.

Xpert BI allows the user to copy exe parameters to schedule and run the data load from an external scheduler e.g. SQL server agent and/or SSIS. This is done by choosing Copy Exe Parameters on the particular group. Xpert BI automatically creates a default “All” process group, which holds all fact tables and related dimension tables in the ELM database. For how to create a process group, please view chapter 2.2.3.3

3.2.5.4. Execute history table



To load data from a view or table into a history table, right-click on the object in the Solution Explorer, choose Execute Object or right-click on the “Execute” icon on the object in the Object Dependency graph, and choose

Execute. This will open the standard data processing window where you click yes in the confirmation dialog or Execute from the ribbon to start the data load.

To schedule history table loads, choose Copy Exe Parameters from the Object Dependency Graph and schedule the batch process from any scheduler.

3.2.6. Smart Processing

"SmartProcessing" is an optional parameter that can be added when processing a Process Group using the batch file. When this flag is set, Xpert BI will optimise the processing of executable objects (Inline and InlineSP) - it will try to only process an executable if it needs to, skipping those that do not need re-processing. For each executable it will check:

1. If the executable or any of its children (down to the next executable or table) has any code changes.
2. If the executable or any of its children (down to the next executable or table) should always process. Objects that will cause an executable to always process is:
 - A table without any children that is not located in ODS.
 - View with non-deterministic code (for example GetDate()).
 - An InlineSP.
3. If any of its children (down to the next executable or table) has been processed after itself or if itself has not been processed yet.

If any of the checks are true the executable will process, otherwise it will be skipped (it will be logged as "CACHED" in the performance log).

Custom Objects will always be processed.

If any object is processed more than desired because of anything in step 2, you can set an ignore-period on the object in the Object_All table in the management database (IgnoreReprocessingForMinutes column). This will skip step 2 for this object and add the number of minutes to its own timestamp when comparing its last processing time to its children. The object will be re-processed when the ignore-period has elapsed, even if no children have been processed in the meantime.

Note: The first time any object is processed using smart process, it will always process.

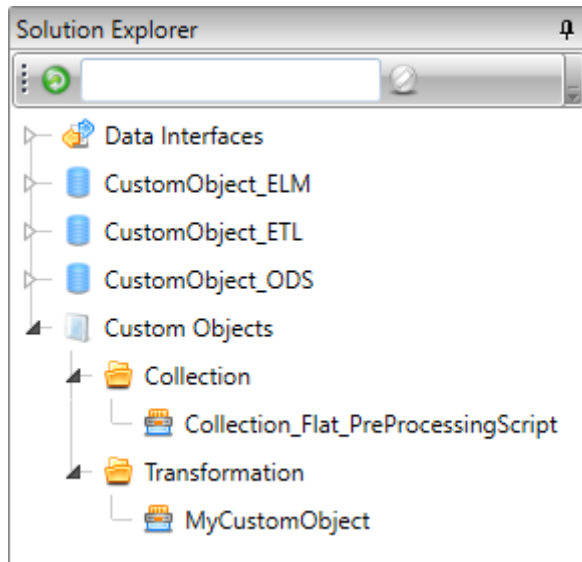
Note: Smart processing is only supported if all the objects in the process group are located on non-remote databases.

3.2.7. Custom Object

A custom object allows for executing arbitrary code. The workflows that currently support custom objects are

- Transformation
- Collection

All custom objects can be viewed in the Solution Explorer under Custom Objects. Custom objects that belong to a collection is shown in the Collection subfolder. This subfolder is only visible if there are any custom objects connected to a collection.



3.2.7.1. Create Custom Object

Creating a new custom object is different for each of the type of workflows.

- Transformation: In the solution explorer, right click either 'Custom Objects' or the 'Transformation' folder and click 'New Custom Transformation Object'.
- Collection: This is done in the collection setup pane. See collection chapter for details.

3.2.7.2. Delete Custom Object

Deleting a custom object can be done from the Solution Explorer by right clicking on the custom object that is to be deleted and clicking 'Delete Custom Object'. Collection type custom object can be deleted from the collection setup pane. See collection chapters for details. The deleted object is not automatically backed up and can therefore not be reverted (SQL objects are backed up).

3.2.7.3. Edit Custom Object

Editing an existing custom object can be done from the Solution Explorer by right clicking on the custom object and selecting 'Edit Custom Object'. Collection type custom objects can be edited from the collection setup pane. See collection chapters for details.

3.2.7.4. Custom Object Window

Object Name is a unique name given to the custom object.

Description is a detailed description of what this custom object does.

Comment can be used to convey some information about this custom object to other developers.

Script Type sets what type of script is executed. Note that the selected script type must be installed and available, no script types are installed during the installation of Xpert BI.

Code is the actual code that is going to be executed. **Insert Code Template** button will insert template code in the selected script type language. **Execute Code** will run the current code with the appropriate parameters applied.

Children is used to set what objects are children to this custom object. Objects set as child cannot be selected as parent. The custom object will be visible in the Object Dependency graph.

Parents is used to set what objects are parents to this custom object. Objects set as parent cannot be selected as child. The custom object will be visible in the Object Dependency graph.

3.2.7.5. Execution

Execution of custom objects are done automatically by the process that the custom object is connected to.

- For transformation the custom object is executed when a process group containing any of the configured parent/child is executed.
- For collection the custom object is executed before any files are processed. For REST type it is executed after files have been downloaded. Custom object is executed before XSLT transformation.

When a custom object is executed it will receive a set of parameters that contains information about the environment. The templated code for each script type will show how to access these parameters. See also the relevant script type chapter for information on how to access these parameters. The parameters will differ for each type of custom object and the following table shows what parameters are available.

| | |
|----------------|---|
| Transformation | ETL Connection String Children Parents |
| Collection | Table Name Table ID Input Path PreProcess Path Succeeded Path Rejected Path Management Connection String Destination Connection String |

3.2.7.6. Script Type: Powershell

There are two ways to access the parameters passed by XpertBI when executing the script. First method is to use the *param* command. This command must be the first line of code in the script to work. The second method is to access the *\$args[]* array, and the position of the argument is given. *\$args[]* can be accessed from anywhere in the code.

Param command for each type of custom object:

- **Transformation:** param(\$XBI_ElmConnectionString, \$XBI_Children, \$XBI_Parents);
- **Collection:** param(\$XBI_TableName, \$XBI_TableId, \$XBI_InputPath, \$XBI_PreProcessPath, \$XBI_SucceededPath, \$XBI_RejectedPath, \$XBI_ManagementConnectionString, \$XBI_DestinationConnectionString);

When accessing the *\$args[]* array, the parameters are passed in this order for the different types of custom object:

| Array Index | Transformation | Collection |
|-------------|-----------------------|-------------------------------|
| 0 | ETL Connection String | Table Name |
| 1 | List of Children | Table ID |
| 2 | List of Parents | Input Path |
| 3 | | PreProcess Path |
| 4 | | Succeeded Path |
| 5 | | Rejected Path |
| 6 | | Management Connection String |
| 7 | | Destination Connection String |

3.2.7.7. Script Type: Python

Any python version can be installed and used, but the version in use must be added to the environment variable 'PATH'. That is, python must be able to start from command line in any directory.

The parameters can be accessed as global variables in the script. Click the "Insert Code Template" button to insert placeholders for the variables available. This placeholder *must* be included, if not, XpertBI will not insert data into the script. Any data or code created between the two lines "#XBI-Reserved" and "#XBI-End-Reserved" will be lost when the script is executed.

It is recommended not to capture exceptions unless necessary. Exceptions are automatically retrieved by XpertBI from standard error.

Important: When accessing any database using pyodbc connection, it is very important to commit and close connections. To do this automatically, use auto commit when opening a connection. If auto commit is not used, it is very easy to deadlock the entire database.

3.2.8. Database Schema

Clicking the **Database Schema** button on the ribbon bar opens a pane which lets you add descriptions to the database schemas found in your solution. This button is available if there are more than one schema (with an SQL object added to it) in your solution. The default schemas (ODS, ETL, ELM) will have a default description added to it (this can be overridden).

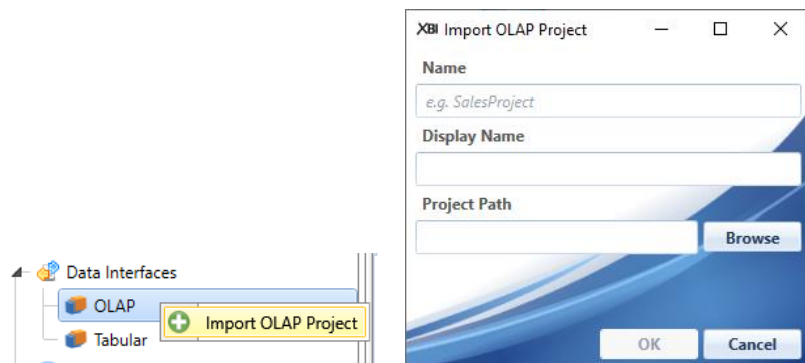
3.3. Data Interfaces

Data interfaces lets you import and manage metadata for SSAS OLAP and SSAS Tabular projects and Power BI reports (including workspaces, apps, data flows and datasets).

3.3.1. OLAP

3.3.1.1. Import

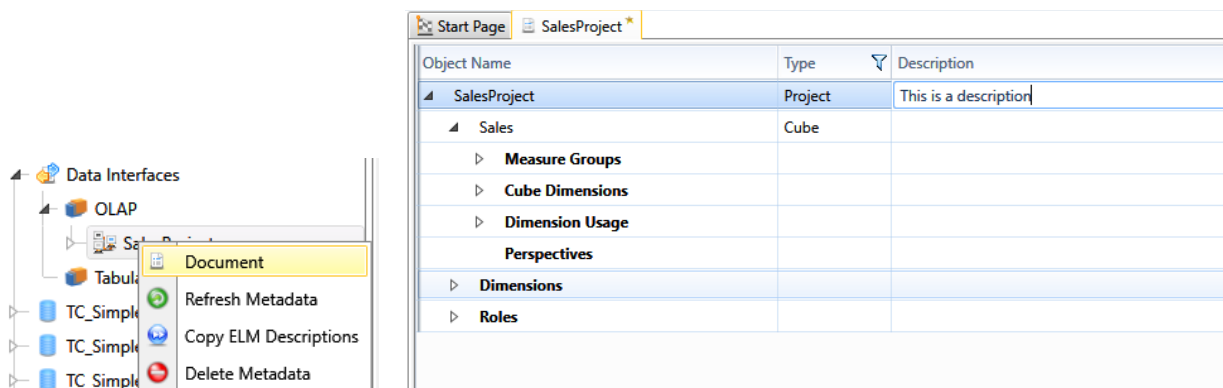
To import an OLAP project, right-click OLAP and click **Import OLAP Project**.



The **Name** of the project must be unique and the **Project Path** is the file path to the project file (dwproj-file). Click **OK** to import the metadata from the project.

To document the metadata, right-click a project and click **Document**. This will open a pane which contains a tree structure of all the metadata stored by Xpert BI. Click **Save** to save any changed descriptions.

Note: If this data is being viewed in Xpert BI Solution Catalog, certain keywords can be used for extended functionality. Please see Appendix I.



3.3.1.2. Refresh

To refresh/update the metadata, right-click the project and select **Refresh Metadata**. This will open the pane seen below. Here you can see what will be changed/added/deleted. Yellow star signals a change, green checkmark means no change, plus-sign means the object is new (added in project) and delete-icon means the object is deleted in the project and will be deleted in Xpert BI.

| Start Page SalesProject | | |
|-------------------------|---------|-----------------------|
| Object Name | Type | Description |
| ★ SalesProject | Project | This is a description |
| ▸ ★ SalesProject | Cube | |
| ▸ ✓ Dimensions | | |
| ▸ ✓ Roles | | |

Descriptions can also be added here. Click **Save** to save the refreshed metadata and any new/changed descriptions.

3.3.1.3. Copy ELM Descriptions

Right-clicking an OLAP project and selecting **Copy ELM Descriptions** will copy object descriptions from ELM to the matching dimensions and measure groups, as well as the column descriptions to the matching measures and dimension attributes.

3.3.1.4. Delete

To delete the metadata for an OLAP project from Xpert BI, right-click the project and click **Delete Metadata**. This will delete all metadata for the selected project, along with all descriptions belonging to that project.

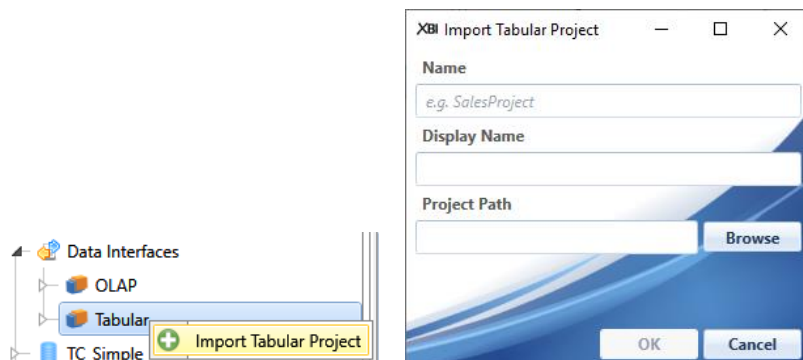
3.3.1.5. Edit

To view/edit the project path and display name for an OLAP project, right-click the project and click **Edit**. This will open an edit window. Click **OK** save.

3.3.2. Tabular

3.3.2.1. Import

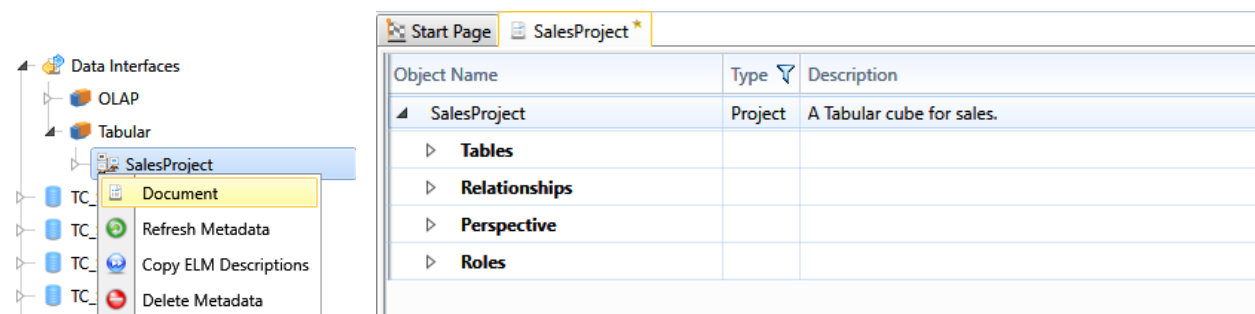
To import a Tabular project, right-click Tabular and click **Import Tabular Project**.



The **Name** of the project must be unique and the **Project Path** is the file path to the model file (bim-file). Click **OK** to import the metadata from the project.

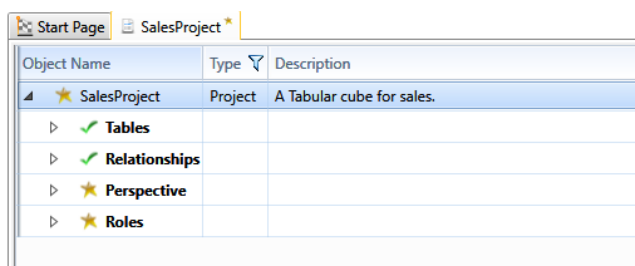
To document the metadata, right-click a project and click **Document**. This will open a pane which contains a tree structure of all the metadata stored by Xpert BI. Click **Save** to save any changed descriptions.

Note: If this data is being viewed in Xpert BI Solution Catalog, certain keywords can be used for extended functionality. Please see Appendix I.



3.3.2.2. Refresh

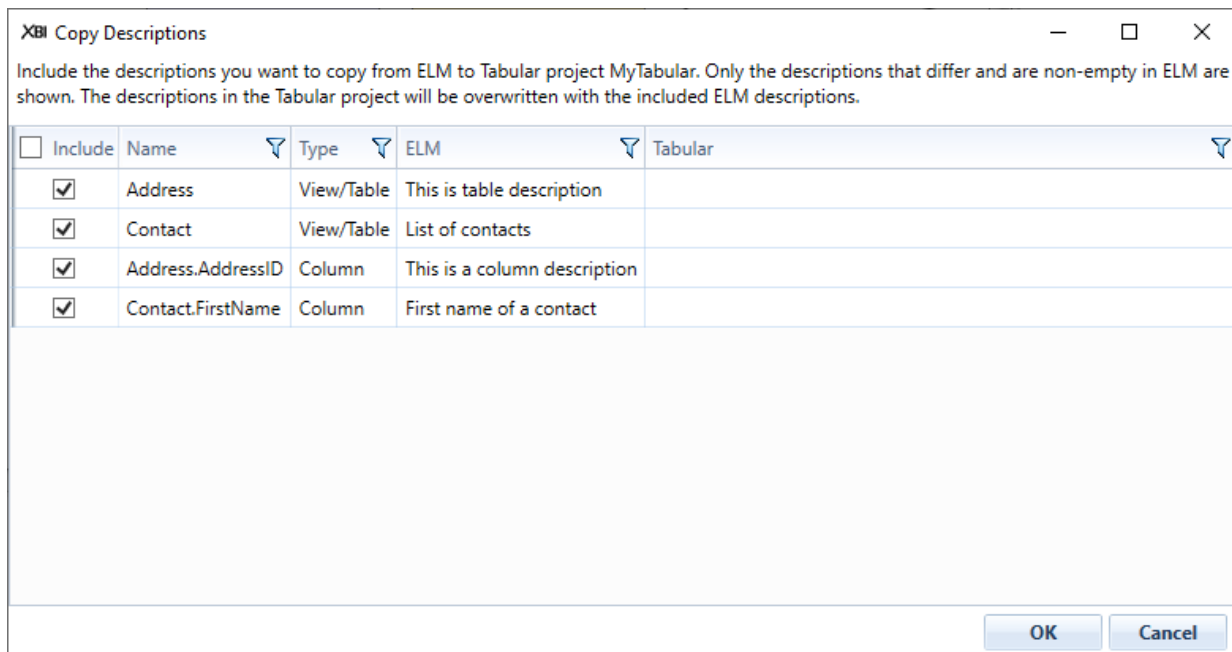
To refresh/update the metadata, right-click the project and select **Refresh Metadata**. This will open the pane seen below. Here you can see what will be changed/added/deleted. Yellow star signals a change, green checkmark means no change, plus-sign means the object is new (added in project) and delete-icon means the object is deleted in the project and will be deleted in Xpert BI.



Descriptions can also be added here. Click **Save** to save the refreshed metadata and any new/changed descriptions.

3.3.2.3. Copy ELM Descriptions

Right-clicking a Tabular project and selecting **Copy ELM Descriptions** will open a window which shows the descriptions that differs between ELM and the Tabular project (tables and columns). You can then select which descriptions you want to copy from ELM to the Tabular project and click OK.



The dialog box titled "XBI Copy Descriptions" contains a message: "Include the descriptions you want to copy from ELM to Tabular project MyTabular. Only the descriptions that differ and are non-empty in ELM are shown. The descriptions in the Tabular project will be overwritten with the included ELM descriptions." Below the message is a table with columns: Include, Name, Type, ELM, and Tabular. The table lists four items: Address (View/Table), Contact (View/Table), Address.AddressID (Column), and Contact.FirstName (Column). Each item has a checked checkbox in the 'Include' column and a description in the 'ELM' column. The 'Tabular' column is empty for all items. At the bottom right are OK and Cancel buttons.

| <input type="checkbox"/> Include | Name | Type | ELM | Tabular |
|-------------------------------------|-------------------|------------|------------------------------|---------|
| <input checked="" type="checkbox"/> | Address | View/Table | This is table description | |
| <input checked="" type="checkbox"/> | Contact | View/Table | List of contacts | |
| <input checked="" type="checkbox"/> | Address.AddressID | Column | This is a column description | |
| <input checked="" type="checkbox"/> | Contact.FirstName | Column | First name of a contact | |

3.3.2.4. Delete

To delete the metadata for a Tabular project from Xpert BI, right-click the project and click **Delete Metadata**. This will delete all metadata for the selected project, along with all descriptions belonging to that project.

3.3.2.5. Edit

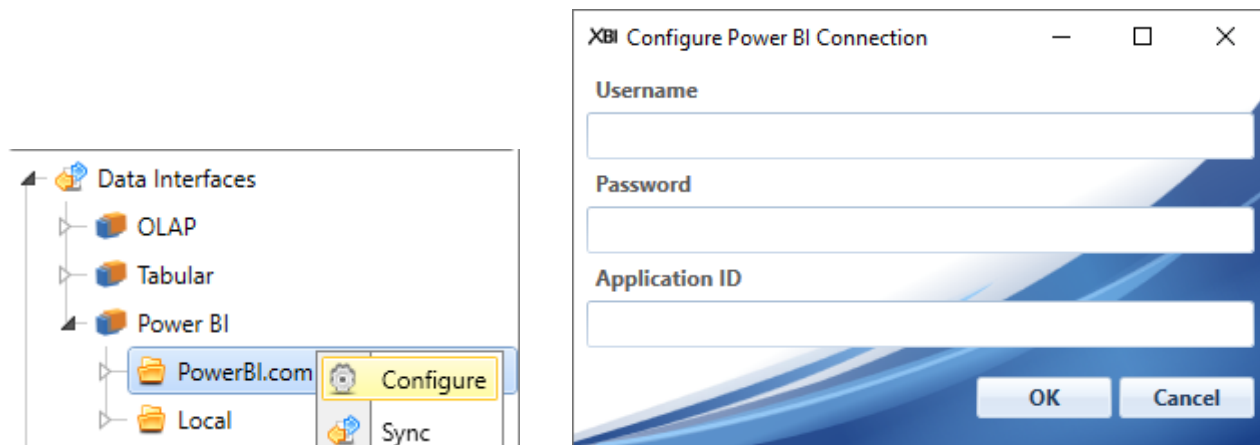
To view/edit the project path and display name for a Tabular project, right-click the project and click **Edit**. This will open an edit window. Click **OK** save.

3.3.3. Power BI

Xpert BI can import information and metadata from reports (either published to PowerBI.com or local pbix files), workspaces, apps and Power BI Data Flows.

3.3.3.1. Configure

Before you can sync anything, you must configure the connection to PowerBI.com. Right-click PowerBI.com and click **Configure**. This opens the window seen below where you can enter the **Username**, **Password** and **Application ID**.



The user must have administrator rights for PowerBI.com (for example Office 365 Global Administrator or Power BI Service Administrator). In addition, to be able to sync all report information for all reports, the user must have access to all relevant workspaces.

To get the **Application ID** you must register your PowerBI.com with Azure (as an app). Follow the instructions here: <https://docs.microsoft.com/en-us/power-bi/developer/register-app>

Use the **native** application type.

If PowerBI.com is already registered, you can find the **Application ID** in your Azure Portal, under Azure Active Directory and then App registrations.

Depending on the setup you might need to grant additional access rights afterwards. The user needs at least "Tenant.Read.All", "App.Read.All" and "Report.ReadWrite.All".

3.3.3.2. Sync PowerBI.com

To sync/download information from PowerBI.com, right-click PowerBI.com and click **Sync**. This will open the window seen below.

XBI Sync PowerBI.com

The following workspaces was found. Check which reports you want to sync. Any deleted reports will be removed from Xpert BI, including all their metadata and comments. Deletes that are not included will be soft-deleted from Xpert BI.

| <input type="checkbox"/> Include | Name | Status | <input type="checkbox"/> Include Report Details |
|-------------------------------------|---------------------------|----------|---|
| <input checked="" type="checkbox"/> | Demo | Existing | |
| <input type="checkbox"/> | LiveTabularReport | Existing | <input type="checkbox"/> |
| <input type="checkbox"/> | ImportedSqlAndExcelReport | New | <input type="checkbox"/> |
| <input type="checkbox"/> | ImportedRESTAPI | New | <input type="checkbox"/> |
| <input type="checkbox"/> | ImportedADL | New | <input type="checkbox"/> |
| <input type="checkbox"/> | WithDataFlow | New | <input type="checkbox"/> |
| <input type="checkbox"/> | Test | Existing | |

OK Cancel

Here you can see all your workspaces, expandable to reports. The status will say if this is a new, existing or deleted workspace/report. For each report who's included, the following information is synced:

- Report metadata like name, embed URL, report type etc.
- Information on data sources used by the report, like server, database etc.

For reports which has **Include Report Details** checked, the following information will also be synced:

- Metadata (including descriptions) for the dataset used by the report, including tables, columns, measures and hierarchies (for reports with imported data).
- Measures defined in the report (for reports with live connections).
- Which columns/measures that are used in the report (in visuals/filters etc).

Note: To be able to sync the report details, Power BI Desktop must be installed and set as the default application to open pbix-files with.

For each workspace with at least one report included, the following information is synced:

- Metadata for the workspace including name, state, type etc.
- Information on all the Power BI Dataflows included in the workspace, including their tables/columns and their data sources.
- Information on all the datasets in the workspace, including the name and who configured it.
- Information on the data sources for each dataset, including type, server, database etc.
- Information on any Power BI App published from the workspace, including the app name and the user who published it.

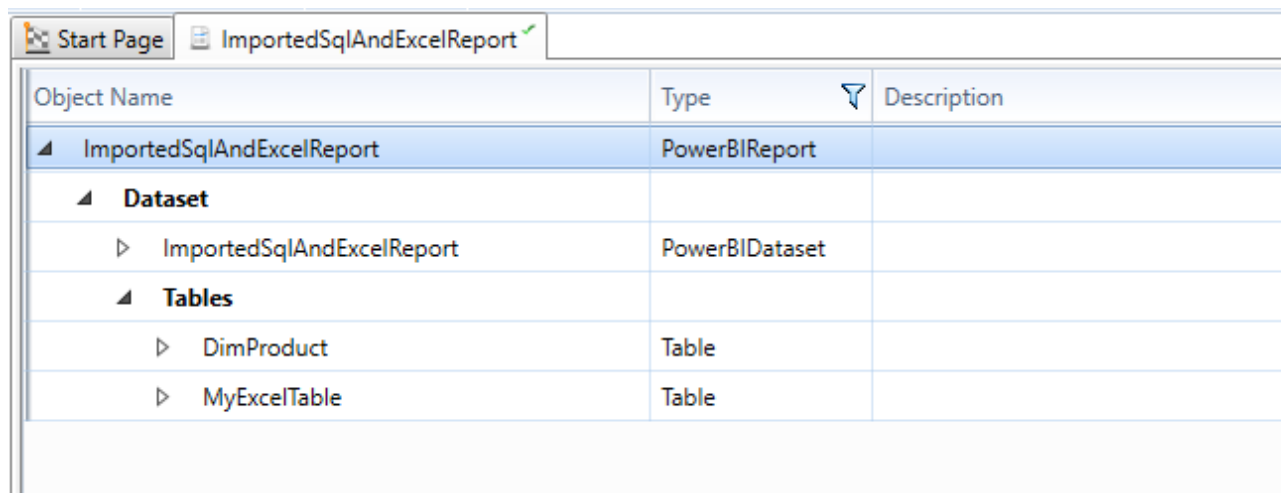
To sync the selected report/workspaces, click **OK**. After the syncing is done, you will see a window saying which reports was successfully synced.

The syncing can also be done on a single workspace (the process will be the same, just right-click on a workspace and click **Sync**) or on a single report. When syncing a single report, a pane will open showing the differences between the existing metadata (stored by Xpert BI) and the metadata found in the report. Elements in the pane will have stars if they are changed; to see what has changed hover the mouse over the changed element.

Note: To be able to sync dataset content (tables, columns etc) on a dataset, you must sync a report (including report details) which uses that dataset.

3.3.3.3. Document Power BI Objects

To document (write descriptions) on a report, dataset or dataflow, right-click it and select **Document**. This will open a pane as seen below, where you can expand the various elements of the report/dataset/dataflow and write descriptions.



| Object Name | Type | Description |
|---------------------------|----------------|-------------|
| ImportedSqlAndExcelReport | PowerBIReport | |
| Dataset | | |
| ImportedSqlAndExcelReport | PowerBIDataset | |
| Tables | | |
| DimProduct | Table | |
| MyExcelTable | Table | |

To see more information on each element/row, hover the mouse over its name. Click **Save** to save the descriptions.

Note that a dataset can be used by many reports, so descriptions on any elements included in a dataset (tables/columns etc) will be used by all reports using that dataset.

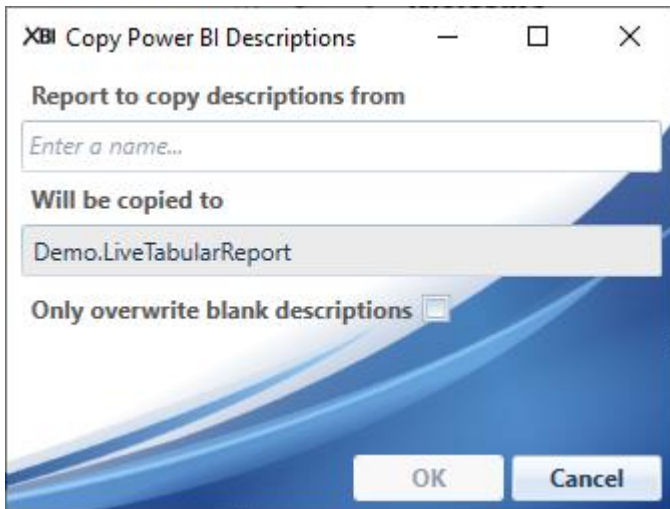
The Sort Order and Hidden field is active for Pages and Visuals in a report. This allows you to specify which pages/visuals should be visible by default in the Catalog, and how they are sorted. Pages/visuals with the same sort order is sorted by name.

The descriptions for Report Visuals are initially read from the Alt Text description on the visuals.

If any description originally read from the report is changed in Xpert BI, the new description will be used and any changes to the description in the report will not be visible in Xpert BI.

3.3.3.4. Copy Report/Dataset Descriptions

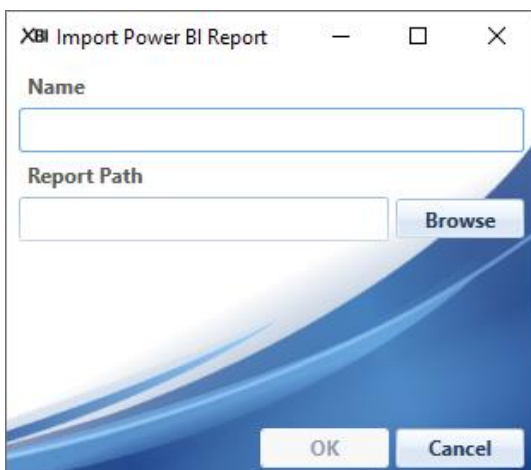
To copy descriptions from one report/dataset to another report/dataset, right-click the target report/dataset and select **Copy Descriptions**. This will open a window as seen below.



In the top input field, you choose which report/dataset you want to copy from. If you want to only overwrite the blank/empty descriptions, check the **Only overwrite blank descriptions** checkbox. Copy descriptions works by finding objects in source (**Report to copy description from**) and matching them by name to the target (**Will be copied to**) and copying/overwriting the description. Objects that don't match will not be touched.

3.3.3.5. Import Local Power BI Report

To import metadata on a local report (a report which is not published to PowerBI.com), right-click the **Local** folder and select **Import Report**. This will open the window seen below, where you can give the report a name and **Browse** to its location. Click **OK** to import the report.



This will open a pane showing the details from the report, as seen below. Click **Save** to save the metadata, including any descriptions written by you.

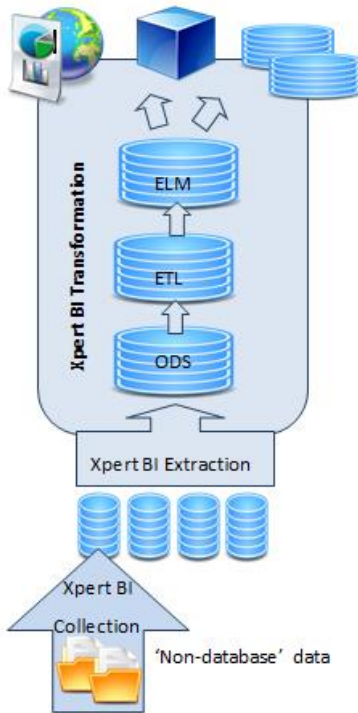
| Start Page MyReport * | | |
|-----------------------|----------------|-------------|
| Object Name | Type | Description |
| ▲ ★ MyReport | PowerBIReport | |
| ▲ ★ Dataset | | |
| ✓ MyReport | PowerBIDataset | |
| ▲ ★ Tables | | |
| ▷ + MyExcelTable | Table | |
| ▷ + DimProduct | Table | |
| ▷ ★ Object Usage | | |

If you have made changes to the report and want to re-sync it, right-click a local report and select **Refresh Metadata**. This will re-import the metadata from the report and show a pane like the one seen above, showing which elements are changed with a star. Click **Save** to save the changes.

To change the path of the report, right-click it and select **Edit Report Path**. This will open a window where you can browse to the new report location.

To delete the report, including all its metadata and stored descriptions, right-click it and select **Delete Metadata**.

4. Xpert BI Collection



Xpert BI Collection enables data collection from flat files, excel, XML files, Web services etc. and it is licensed by connector type. The objective of this module is to be able to include 'non-database' data in the data warehouse solution and to store it in the same structure as the rest of the company's data and to be able to do reporting and analysis on such data in the same way as with other application/database data. Various features of Xpert BI Extraction and/or Transformation will enable the integration of such 'non-database' data with other data from other databases.

In the data warehouse architectural structure, Xpert BI Collection will typically be a step before Xpert BI Extraction, where the database produced by Xpert BI Collection can be treated as any other source application on SQL server.

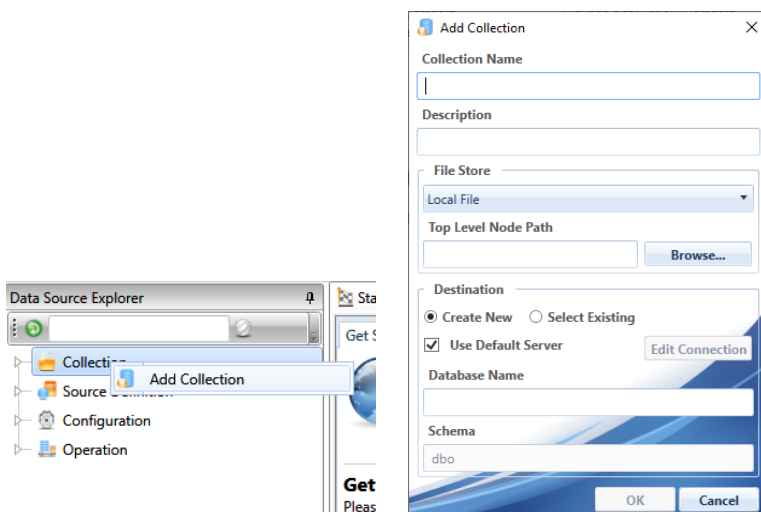
Data imported with Xpert BI Collection can also be directly integrated in the ETL database layer. Implementations will vary depending on data and data models.

With Xpert BI Collection you can use one tool for all your data transfers. All logging will be handled by the standard Xpert BI logs and the data load scheduling is very flexible.

4.1. Creating a Collection

To start configure a flat file source, right-click on Collection and choose Add Collection or choose Collection from the New task group on the ribbon. A Collection is a logical grouping of files/tables and can contain data imports from the various types of connectors (csv, excel, xml, REST, SOAP, SAP RFC). You can define many Collections in Xpert BI and one Collection can contain many tables.

The Collection Name can be the name of a data source/application or the name of the group of files you want to import. i.e. BudgetImport. You can choose to add a description to the Collection.



Next you must specify which **File Store** type to use for the Collection folder structure. The folder structure includes one folder per table/configuration, with subfolders: **Input, Succeeded, Rejected, Config, PreProcessInput** (for XML). The available places where the files can be stored are:

Local File: Saves the files to local windows file storage. A top level node path must be set.

Azure Data Lake: Saves the files to blobs in an Azure Data Lake container. A connection string must be set with a root path.

Note: It is highly recommended that Hierarchical namespace is turned on for the storage account. If this is turned off, folders will automatically disappear when they become empty. Xpert BI will not be able to create the initial folder structure as it would disappear immediately and the user would have to create them manually. If this is the case, remember that paths are case sensitive.

And lastly, you must specify **Destination** database. It is possible to either create a new database or select an existing database. When creating a new database, a database name must be specified and a schema if the target server is schema based.

The destination is determined by either using the default server, which is the same server as the Xpert BI Solution stack. Or by specifying a different server by deselecting the **Use Default Server** checkbox and clicking **Edit Connection**.

4.1.1. Editing a Collection

An existing Collection may be edited. Select the Collection Name under the Collection node and right click and choose Edit Collection or choose Edit Collection from the General task group on the ribbon. Edit the desired properties and choose OK to accept the changes or Cancel to discard them.

- If **File Store** is changed, the Collection folder structure is automatically created in the new location.
- If the **Destination** is changed, the database or schema will be created if it does not exist.

If either the **File Store** or **Destination** is changed, it is recommended to manually open each collection and check that paths are valid and force it to re-create the tables. Some scripts that are created when saving are different depending on the destination server type.

4.1.2. Copying a Collection

An existing Collection may also be copied. Select the Collection Name under the Collection node and right click and choose Copy from the menu. This creates a new Collection based on the clicked item. All parameters are copied to the new instance except the Table Name.

When adding a Table Name, a new Collection folder structure is generated in the new location.

4.1.3. Drag and Drop

A collection folder has drag and drop support for Flat Files. Drag and drop a file from the file system to automatically configure the collection table. If multiple files are dropped, one collection table will be created per file. Supported file types are the same as for Flat File. The collection folder will be created with the following settings:

- Table Name: File name without extension
- Delimiter: For non-Excel files: automatically detected, with fall back to semi-colon “;”.
- Automatic PK: On
- The file is copied into the Config and Input folder under the collection table on the file system.

4.2. Xpert BI Collection - Flat File

The Xpert BI Collection Flat File connector supports various types of text files and Microsoft Excel files (that contain data in table formats). Supported file types are: csv, tsv, tab, psv, dsv, txt, xls, xlsx, xlsb, xlsm, xlsx.

After creating a Collection folder, you are ready to start configuring the file and table mapping. Right-click on the collection and choose Add Table Source Flat File or choose Table Source Flat File from the New task group on the ribbon menu to start the configuration.

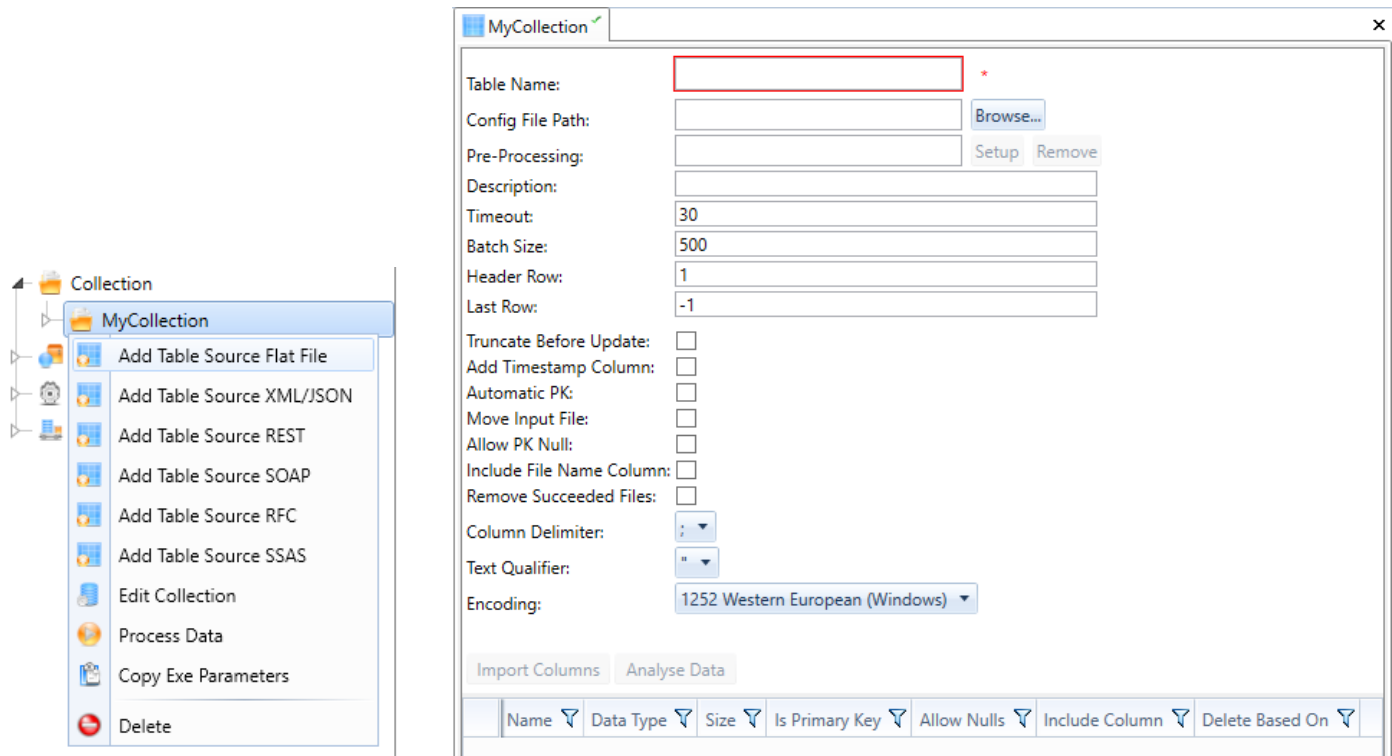


Table Name is the name of the destination table and also the name of the folder that Xpert BI will create on the file system. As soon as you click outside the Table Name text box, the folder structure is created on the file system.

Config File Path is where the file example/copy is located and you specify this to import file format and metadata, i.e. column headers and data types. It is recommended to place this file in the generated Config folder.

Pre-Processing is a script that runs before processing the files. Press Setup to add a new script or edit existing script. See chapter about Custom Objects for more details.

Description is a field where you can add a description to the table and table content.

Timeout and **Batch Size** are set to 30 and 500 respectively, but can be changed. The Timeout specifies command timeout and the Batch Size specifies number of rows per read in the bulk copy.

Note: When reading very large CSV files, increase the timeout and batch size to speed up the processing and minimize overhead while loading.

Header Row specifies the row number for the table headers. The actual values are expected to start on the successive row. The Header Row can be set to 0 to indicate that there is no header row. The first data row is then assumed to start at row number 1. If you want to use this feature you have to do the following:

- Create a CSV-file with the desired headers (matching the number of columns in the CSV-file you want to import).
- Import Columns from the CSV-file with headers (with Header Row = 1).
- Define a Primary Key (or use Automatic PK), change to Header Row = 0 and Save.

- Finally, you can Process Data to load your CSV-file into the destination table (remember to put the original CSV-file in the Input folder first).

Note: Setting the header row to 0 is only supported for CSV files, NOT Excel.

First Row specifies the first row to read. Setting the first row to 1 will read all rows as data, meaning that the file does not have a header. The default is 2.

Last Row specifies the last row to read. Setting the last row to -1 will read all rows (default).

Note: When reading very large CSV files and setting a last row (other than -1), memory consumption can be high.

Sheets specifies what sheets to read from an Excel file (not valid for csv-files). Sheets can be determined by index or by name (not both at the same time). Index use the format: 1,2,3-5 and name use the format [Sheet name 1], [some other sheet name]. It is important to include the brackets when naming the sheets. Leave this field empty (the default value) to read all pages in the Excel file.

Truncate Before Update can be used to do a truncate of the destination table before the new data is loaded.

Add Timestamp Column can be used to include a timestamp column to the destination table.

Automatic PK can be used to include a unique identifier column that will be used as primary key. Checking this will automatically check **Truncate Before Update** and remove any set primary key.

Move Input File controls whether the input file(s) are moved to the Succeeded/Rejected folder after processing, or kept in the input folder and copied to the Succeeded/Rejected folder.

Allow PK Null can be used to replace NULL values with default values for columns that are included in the Primary Key. The default values are 0 for numeric/integer fields, <blank>/' for character fields, and 19000101 for date fields.

Checking **Include File Name Column** will create an extra column in the destination table, called XBI_FileName. This column will contain the filename of the files being read.

Remove Succeeded Files can be used to clean the succeeded output folder before processing new files.

Column Delimiter specifies the delimiter that is used when parsing the flat file, both on header level and data level.

Text Qualifier specifies a character which may be used to surround fields that can include special characters.

Encoding specifies the Unicode character encoding. Default value is 1252 Western European (Windows).

When you have configured the file format, click **Import Columns** (requires a Config File Path value to be clickable) to retrieve the list of columns from the worksheet(s). For Excel files with multiple worksheets, the result will be all the distinct columns in the file. For text file types, default data type is nvarchar(200). For Excel files, Xpert BI will import the datatypes from the file and string values are set to default data type nvarchar(200).

If **Import Columns** is clicked when a previously saved configuration exists, Xpert BI will try to preserve existing column information (primary key, data length etc.) by matching on column name. Changes in ordinal position of columns are reflected in the GUI and database. Changes in a column name will be seen as a new column. Removed columns will be deleted.

Clicking **Analyse Data** will open a window where it is possible to analyse the input files. See Analyse Data 4.9 for more information.

The **Include Column** checkboxes can be unchecked for individual columns to choose which columns are imported when processing.

Delete Based On lets you delete rows from the destination table based on data in the input file(s) from the chosen column(s). This step happens after the data is loaded into staging and before staging is merged into the destination table.

Click **Save** to save the configuration. This will generate a table named <Table Name> in the SQL server database you specified as destination.

To load data into the table, place an input file in the *Input* folder and choose **Process Data** from ribbon or right-click menu. Data will then be loaded into the destination table and you can view the *Event Log* for details. If the Excel file contains several worksheets they will all be processed (unless *Sheets* are used to only read specific sheets). If a column is missing from one or more of the worksheets, it will be null-filled for the rows in that sheet. Columns are mapped based on headers, not column order.

Note: If you edit the file or table configurations at a later time, the destination table will be dropped and recreated. This means that you need to manually backup the data if necessary.

During processing, one or more temporary tables are created in the database; one per file (tableName_XXX_STG) and if **Truncate Before Update** is checked, a total table per processing (tableName_XXX_TOTAL). If an error occurs during processing, the relevant tables will NOT be deleted, to simplify debugging. The destination table will not be truncated if the processing fails (for example if the input contains duplicate keys). Any remaining temp tables will be deleted on the next processing.

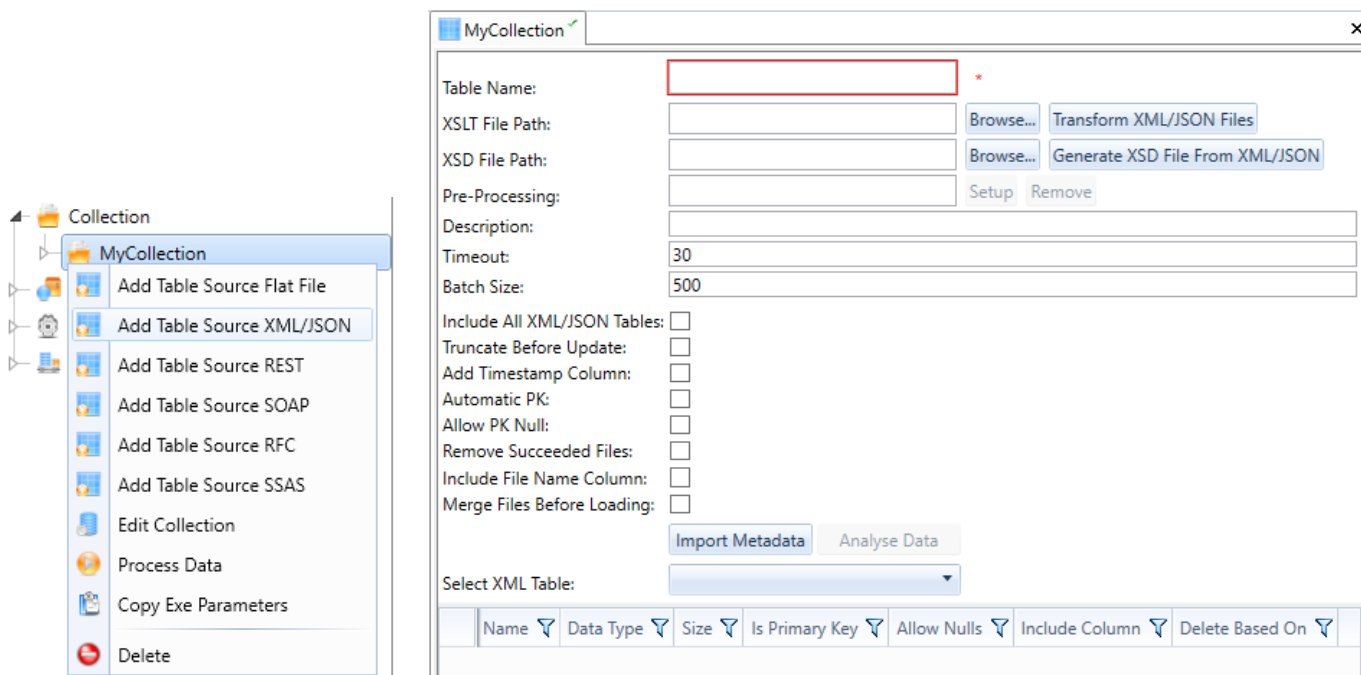
4.3. Xpert BI Collection – XML/JSON

The Xpert BI Collection XML connector supports XML documents both with and without an XSD file. It also supports to transform complex XML documents with XSLT files to cleaner or simpler structures.

JSON files are also supported. If a JSON file is detected, it will automatically be converted to XML during pre-processing and processing. XML transformation is supported with JSON files.

Supported file types are only “.xml” and “.json”.

After creating a Collection folder, you are ready to start configuring the file and table mapping. Right-click on the collection and choose Add Table Source XML/JSON or choose Table Source XML/JSON from the New task group on the ribbon menu to start the configuration.



XML structures can be complex and not necessarily automatically compatible with the table structures that you want data to be inserted into in the database. Xpert BI Collection uses the XML transforming standard XSLT to transform a complex XML file to a simpler and restructured XML file.

The XSLT file must be generated manually outside the Xpert BI environment, but when you have a XSLT file you want to apply, Xpert BI Collection can transform the XML file structure according to the XSLT file.

Table Name is the name of the destination table and also the name of the folder that Xpert BI will create on the file system. As soon as you click outside the Table Name text box, the folder structure is created on the file system. When using **Include All XML Tables**, Table Name is used as a prefix for all tables generated from the XSD file.

XSLT File Path is the path to where you have saved your XSLT file. It is optional to use an XSLT file. It is recommended to place this file in the generated *Config* folder.

To transform the XML/JSON file, copy the original XML/JSON file in the *PreProcessInput* folder, and click on **Transform XML Files**. Xpert BI will then transform the XML structure in the file and place the transformed file in the *Input* folder. The original file will be placed in the *Succeeded* folder.

XSD File Path is the path to where you have saved your XSD file. To read data from XML/JSON into a table structure, an XSD file is required in order to retrieve the metadata i.e. column names and data types.

If you do not have an XSD file, Xpert BI can create one from the XML/JSON file you want to import. Click **Generate XSD File from XML/JSON**, select the wanted XML/JSON file, then save the XSD file. The XSD File Path is auto populated with the path of the saved XSD file.

NOTE: If the XML file contains multiple namespaces, Xpert BI will generate one XSD file per namespace (the first one will have references to the others). If this happens, some unexpected behaviour might occur regarding data types (for example nillable columns not being recognised as nillable). If this happens (most relevant when reading all XML tables), the XSD can be re-generated from a XML file with just one namespace or the XSD files can be manually fixed/merged into one file.

Pre-Processing is a script that runs before processing the files. Press Setup to add a new script or edit existing script. See chapter about Custom Objects for more details.

Description is a field where you can add a description to the table and table content.

Timeout and **Batch Size** are set to 30 and 500 respectively, but can be changed. The Timeout specifies command timeout and the Batch Size specifies number of rows per read in the bulk copy.

Include All XML Tables can be used to process all XML tables defined in the XSD file. When checked, all settings except **Truncate Before Update** are disabled. To support multi-file loading, an additional column is automatically added as part of the primary key, XBI_RunTimestamp. The column value is constant for each run/file load.

Truncate Before Update can be used to do a truncate of the destination table before the new data is loaded.

Add Timestamp Column can be used to include a timestamp column to the destination table.

Automatic PK can be used to include a unique identifier column that will be used as primary key. Checking this will automatically check **Truncate Before Update** and remove any set primary key.

Allow PK Null can be used to replace null/empty values with default values for columns that are included in the Primary Key. The default values are 0 for numeric/integer fields, <blank>/' for character fields, and 19000101 for date fields.

Remove Succeeded Files can be used to clean the succeeded output folder before processing new files.

Checking **Include File Name Column** will create an extra column in the destination table, called XBI_FileName. This column will contain the filename of the files being read.

Merge Files Before Loading will load all the files into memory before inserting/merging the data into the destination table.

Delete Based On lets you delete rows from the destination table based on data in the input file(s) from the chosen column(s). This step happens after the data is loaded into staging and before staging is merged into the destination table.

When you have configured the **XSD File Path**, click **Import Metadata** and select an XML table from the dropdown list to retrieve the list of columns.

Clicking **Analyse Data** will open a window where it is possible to analyse the input files. See Analyse Data 4.9 for more information.

Click **Save** to save the configuration. This will generate a table named <Table Name> in the SQL server database you specified as destination. If **Include All XML Tables** is enabled, all tables in the XSD file will be generated with <Table Name> as prefix, instead of a single table.

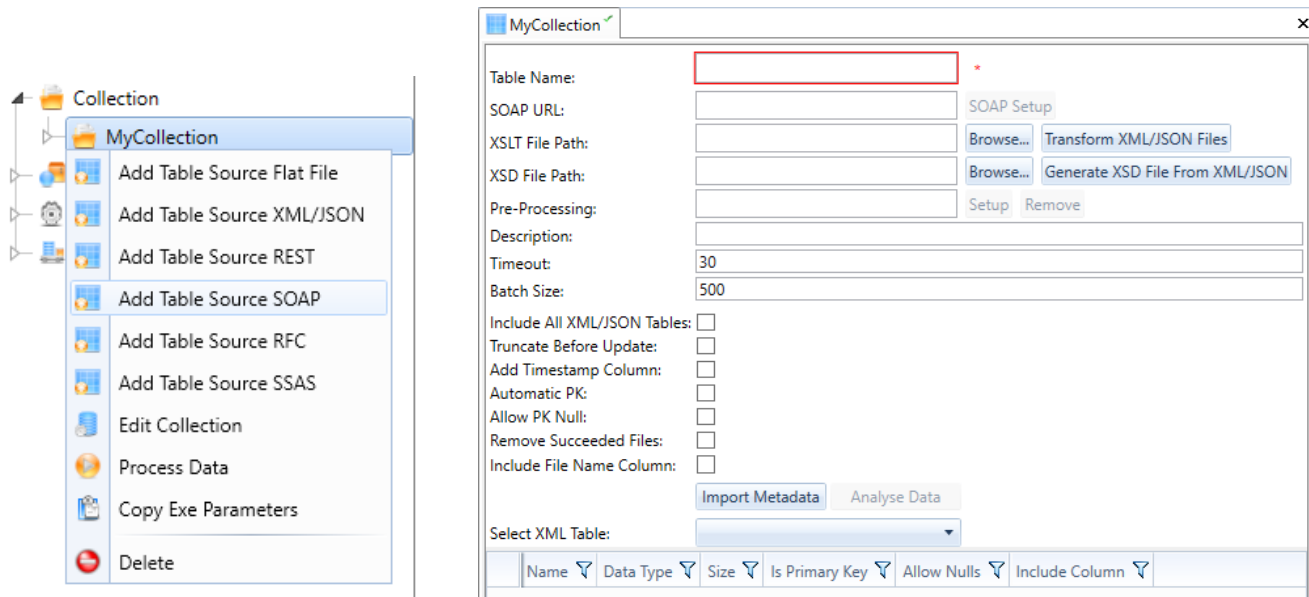
To load data into the table, place an input file in the *Input* folder (or the *PreProcess* folder if using XSLT) and choose **Process Data** from ribbon or right-click menu. Data will then be loaded into the destination table and you can view the *Event Log* for details.

During processing in single table mode, one or more temporary tables are created in the database; one per file (tableName_XXX_STG) and if **Truncate Before Update** is checked, a total table per processing (tableName_XXX_TOTAL). If an error occurs during processing, the relevant tables will NOT be deleted, to simplify

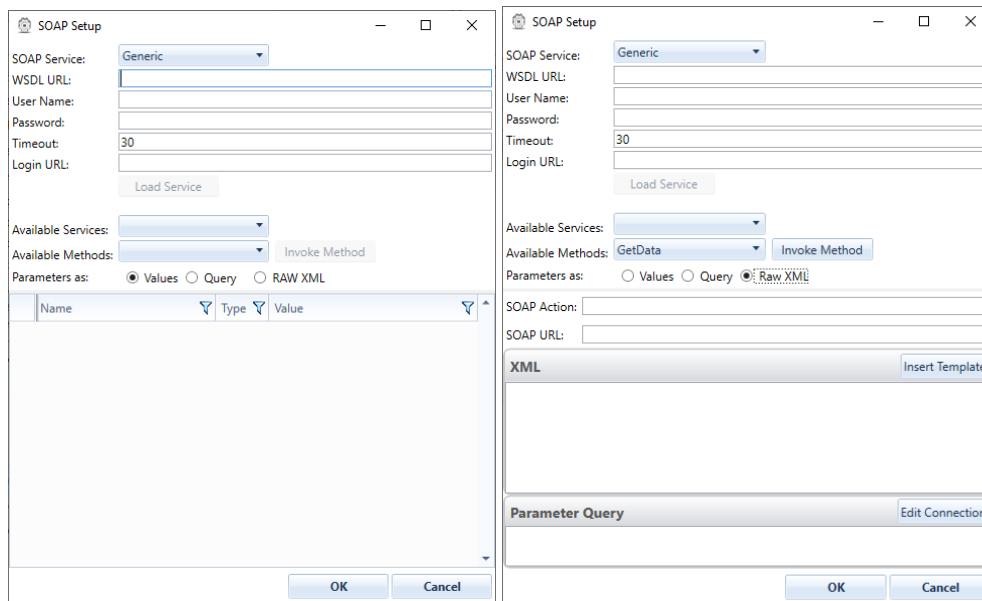
debugging. The destination table will not be truncated if the processing fails (for example if the input contains duplicate keys). Any remaining temp tables will be deleted on the next processing.

4.4. Xpert BI Collection – SOAP

The Xpert BI Collection SOAP connector allows you to connect and download XML files from various SOAP web services. To setup a SOAP connection, right-click a collection and choose Add Table Source SOAP.



Enter a **Table Name** and click **SOAP Setup** to open the SOAP web service setup window.



(Fields with * supports a managed secret. See chapter 2.4.7.7 **Secret Manager** for information)

SOAP Service specifies the the type of SOAP service you are trying to connect to.

WSDL URL* is the URL to the WSDL or SingleWSDL describing the SOAP service you want to connect to.

User Name* and **Password*** are the optional user name and password to the web service.

Timeout specifies the command timeout. Default is 30 seconds.

Login URL is used for services that has a separate login URL. If a login URL is given, Xpert BI will try to authenticate against that URL before downloading the WSDL or calling any of the service methods. If the authentication is successful, Xpert BI will use the returned credential cookie(s) for the following request(s).

Click **Load Service** to load the service. The available services (usually just one) and the available methods are shown in the respective drop down menus. If a method has any input parameters, they will be listed at the bottom of the window. Parameters can either be given as values or as a SQL query. If the parameters are given as values, the web service will be called once, using the given values. If the parameters are given as a query, the query will first be executed against the management database and then the web service will be called for each row of the returned result set.

Note: Ordinal position of the parameters must **not** be changed in the query.

Note: If the datatype of a parameter is an array, use commas to separate the values.

When using **Raw XML*** the XML text is sent verbatim to the SOAP service. It must follow SOAP design in order to work properly. A template for the XML can be inserted by clicking on **Insert Template**. **SOAP Action*** is the SOAPAction header needed when constructing a SOAP request and it must follow SOAP design specifications. It is auto inserted when selecting a method. If the SOAP action is wrong, it is possible to change it. **Parameter Query** is a SQL query which can be used as dynamic parameters in the SOAP Action field and the XML field. Columns in the query can be used by writing `$MyColumnName$` in either the SOAP Action or the XML. If the query has more than one row, it will execute the SOAP query several times and save the result to individual files.

It is possible to specify where the query is going to be executed by clicking **Edit Connection**. If no connection is specified, it will be executed on the server and database specified on the collection folder.

Click **Invoke Method** to invoke the method. This will call the selected method using the given parameters (if there are any) and save the returned result as XML files in the input folder.

Click **OK** to save the SOAP setup. When this is done, the rest of the functionality is the same as for Xpert BI Collection XML. See Xpert BI Collection – XML for further details. When a SOAP collection is processed, it will first invoke the web service to download the XML file(s) to the *Input* (or *PreProcessInput*) folder and then process those files the same way as XML Collection.

4.4.1. Bloomberg

Select Bloomberg as the **SOAP Service** if you are connecting to Bloomberg Data License Web Services.

The screenshot shows the 'SOAP Setup' dialog box with the following fields and sections:

- SOAP Service:** A dropdown menu with 'Bloomberg' selected.
- Service Endpoint:** An empty text field.
- Certificate Path:** An empty text field.
- Password:** An empty text field.
- Method:** A dropdown menu with 'GetHistory' selected, and an 'Invoke Method' button next to it.
- From:** An empty text field.
- To:** An empty text field.
- Periodicity:** A dropdown menu with 'Daily' selected.
- Bloomberg Headers:** A section with a 'Verify Headers' button and a table with two columns: 'Header' and 'Value'.
- Fields:** An empty text field.
- Instruments:** A text area containing the following SQL query:


```
SELECT InstrumentID
      ,InstrumentType
      ,YellowKey
FROM MyBloombergInstruments
```
- Buttons:** 'OK' and 'Cancel' buttons at the bottom right.

(Fields with * supports a managed secret. See chapter 2.4.7.7 **Secret Manager** for information)

Service Endpoint* is the URL to the Bloomberg service.

Certificate Path is the file path for the Bloomberg certificate file.

Password* is the certificate password.

Method lets you choose between GetHistory and GetData. **Invoke Method** is as described above.

From and **To** lets you define a date period for the request (only available for GetHistory). The dates must be formatted as yyyy-MM-dd.

Periodicity specifies the periodicity you want on the returned data (only available for GetHistory).

Bloomberg Headers lets you specify which headers to include with the request. Click the plus-button to add a new header. The **Verify Headers** button will verify the headers. Empty headers (header or value) are ignored.

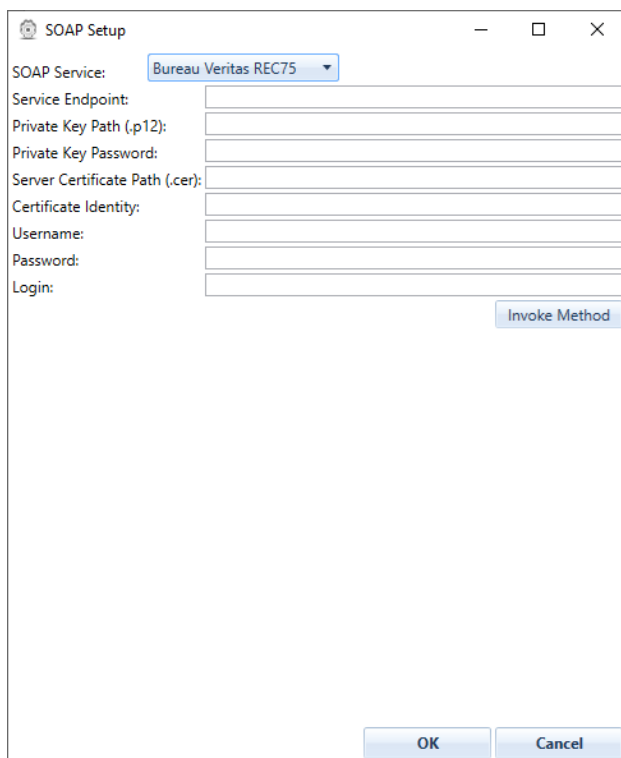
Fields is a comma-separated list of the data fields you want.

Instruments is a SQL query which must return a list of instruments you want data for. The three pre-filled columns are all expected (they are matched by ordinal position).

Note that all fields and all instruments are combined into one single request. Consult the Bloomberg Data License Web Services user manual to see the latest limits for fields and instruments per request.

4.4.2. Bureau Veritas REC75

Select Bureau Veritas REC75 as the **SOAP Service** if you are connecting to Bureau Veritas SOAP service of the type REC75.

A screenshot of a 'SOAP Setup' dialog box. The title bar says 'SOAP Setup' with standard window controls. Inside, there's a dropdown menu for 'SOAP Service' currently set to 'Bureau Veritas REC75'. Below this are several text input fields: 'Service Endpoint:', 'Private Key Path (.p12):', 'Private Key Password:', 'Server Certificate Path (.cer):', 'Certificate Identity:', 'Username:', 'Password:', and 'Login:'. To the right of the 'Login:' field is a button labeled 'Invoke Method'. At the bottom of the dialog are 'OK' and 'Cancel' buttons.

(Fields with * supports a managed secret. See chapter 2.4.7.7 **Secret Manager** for information)

Service Endpoint is the URL to the Bureau Veritas service.

Private Key Path is the path to the private key file. File ending must be '.p12'. Note that this path is not affected by the File Store path set on the collection folder. This path is only a local file in windows.

Private Key Password* is the password for the private key.

Server Certificate Path is the path to the server certificate file. File ending must be '.cer'. Note that this path is not affected by the File Store path set on the collection folder. This path is only a local file in windows.

Certificate Identity is the identity found inside the certificate. Usually 'rec75Client'.

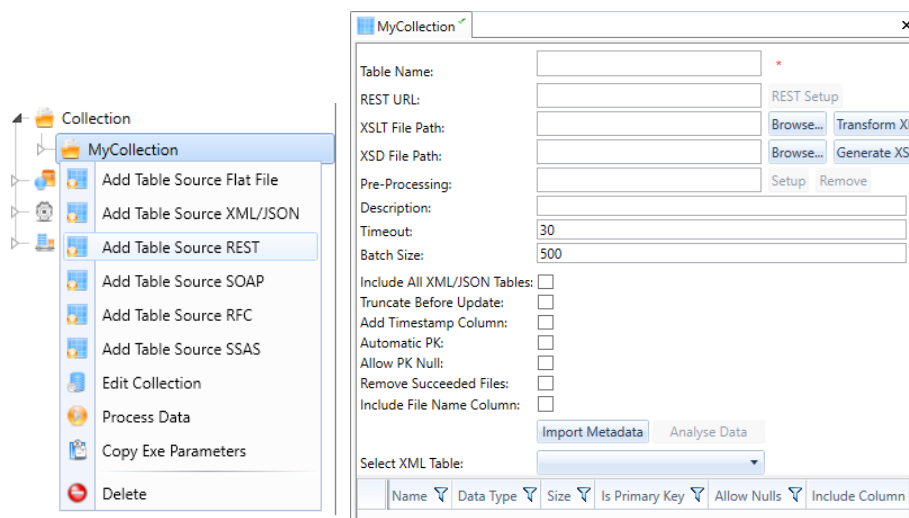
Username* is the username used to access the service.

Password* is the password used to access the service.

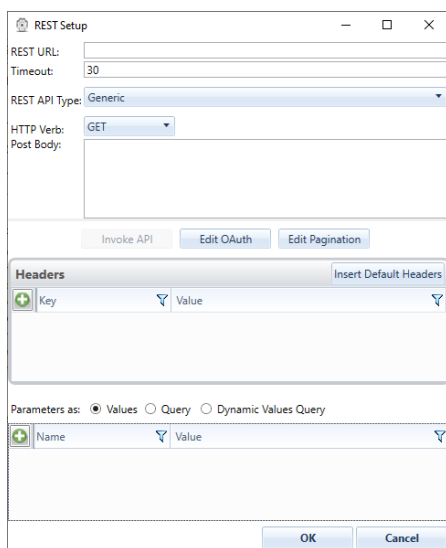
Login* is a login name used by the SOAP method. Usually the same as the username.

4.5. Xpert BI Collection – REST

The Xpert BI Collection REST connector allows you to connect and download XML or JSON files from various REST web services. To setup a REST connection, right-click a collection and choose Add Table Source REST.



Enter a **Table Name** and click **REST Setup** to open the REST web service setup window.



(Fields with * supports a managed secret. See chapter 2.4.7.7 **Secret Manager** for information)

REST URL* is the URL to the REST API endpoint. The URL path can be parameterised using curly braces. Example: If the URL is `http://example.com/myApi/{myTag}`, Xpert BI will try to replace `{myTag}` with the value of a parameter named `myTag`.

Timeout specifies the command timeout. Default is 30 seconds.

REST API Type specifies the type of the REST API you are trying to connect to.

HTTP Verb sets the type of HTTP request. Available verbs are 'GET' and 'POST'

Post Body* will be sent with the request when the HTTP verb 'POST' is selected.

Headers* will be added to the API request as key value pairs. This can for example be used to set the Authorization header in case the API uses basic authentication or an API key. Click the plus icon to add new headers. Click the 'Insert Default Headers' to insert some commonly used headers ('Accept' and 'Content-Type').

Parameters can either be given as values or as a SQL query. If the parameters are given as values, the web service will be called once, using the given values. If the parameters are given as a query, the query will first be executed and then the web service will be called for each row of the returned result set. Click the plus icon to add new parameters.

If **Dynamic Values Query** is set, the SQL query result can be used as parameters in the REST query. It will execute a REST query for each row in the SQL result. To use the values from the SQL query, every column will map to a variable. To use a variable, surround the column name with '\$' and insert it into either the REST URL, header value or the Post Body. Note that variable names are case sensitive. An example of a REST URL using a parameter:

`https://myApi.com/getDataFromMonth/$Month$`

The dynamic values query could look similar to this:

```
SELECT 'Jan' as Month
UNION ALL SELECT 'Feb'
UNION ALL SELECT 'Mar'
```

This will call the API three times with these URLs:

`https://myApi.com/getDataFromMonth/Jan`

`https://myApi.com/getDataFromMonth/Feb`

`https://myApi.com/getDataFromMonth/Mar`

It is possible to specify where the query (from **Parameters** and **Dynamic Values Query**) is going to be executed by clicking **Edit Connection**. If no connection is specified, it will be executed on the server and database specified on the collection folder.

Click **Invoke API** to invoke the API. This will call the API using the given parameters and headers (if there are any) and save the returned result as XML files in the input folder. If an API returns JSON it will automatically be converted to XML by Xpert BI.

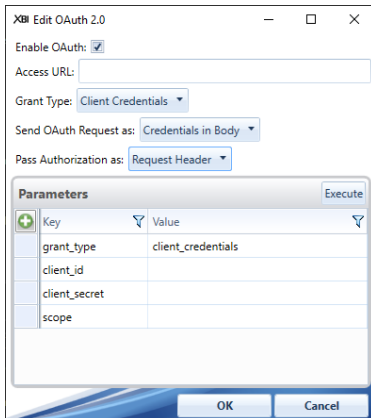
Click **Edit OAuth** to open the OAuth 2.0 settings. See explanation of these settings below.

Click **Edit Pagination** to open the pagination settings. See explanation of these settings below.

Click **OK** to save the REST setup. When this is done, the rest of the functionality is the same as for Xpert BI Collection XML. See Xpert BI Collection – XML for further details. When a REST collection is processed, it will first invoke the web service to download the XML file(s) to the *Input* (or *PreProcessInput*) folder and then process those files the same way as XML Collection.

OAuth 2.0

This window can be used to enable and configure the use of OAuth 2.0. It will retrieve an access token and use it in the main REST request.



(Fields with * supports a managed secret. See chapter 2.4.7.7 **Secret Manager** for information)

Set **Enable OAuth** to true in order to enable the use of OAuth workflow.

Access URL* is the URL to the authentication server where the access token is issued.

Grant Type describes what type of grant is used to obtain the access token. Changing this will automatically set the `grant_type` parameter.

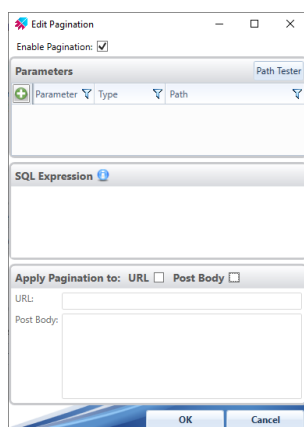
Send OAuth request as sets whether the credentials are used in the body of the request or encoded as basic authentication.

Pass Authorization as sets whether the main REST request passes the access token as an authorization header or as an URL query parameter.

Parameters* is a list of all the parameters that are included in the POST to the authentication server. When changing the grant type, commonly used parameters are automatically added to this list.

Pagination

This window can be used to enable and configure the use of pagination. It can find pagination information in the REST response and perform additional requests to get more pages.



Set **Enable Pagination** to true in order to enable the use of pagination.

Parameters are used to get values from the response data. It can get data from either the response body or the response headers. The parameters can be used either in SQL Expression or in the Apply Pagination part. To use the parameter it needs to have the following format: `$MyParamName$`

Note that these parameters cannot have the same names as any parameter in the Dynamic Values Query configured on the main REST Setup page.

Note that if any of these parameters are unable to find a value at their path, the collection will stop getting more pages and continue with the collection process. This can be used as a stop condition if no data is found in the response.

To create a parameter, click on the button with a plus and select the type of response data to get information from.

Header Path will get information from the headers. Enter the header name into the *Path* column.

XPath/JsonPath will get information from the response body if it is either XML or JSON. In the *Path* column enter the XPath to the XML element that contains the information you need. If it is JSON, enter a proprietary JsonPath.

JsonPath must follow these rules:

- Must start with `$`
- Access a named object by using `['ObjectName']`
- Access an array element by using `[index]`, 0 indexed.
- Access nested structures by concatenating path strings.

Examples of paths:

Consider this JSON:

```
{
  "key1": "value1",
  "key2": "value2",
  "array": [
    "valueA",
    "valueB",
    "valueC"
  ]
}
```

To get "value2", use this path: `$['key2']`

To get "valueB" use this path: `$['array'][1]`

SQL Expression is used to execute SQL to calculate any additional values needed to advance to the next page in the REST api. The parameter names created using the SQL Expression are the same as the column names returned from the SQL query. Only the first row will be used. The parameters can be used in the Apply Pagination part. To use the parameter, it needs to have the following format: `$MyParamName$`

Note that these parameters cannot have the same names as any parameter in the Dynamic Values Query configured on the main REST Setup page.

In the **Apply Pagination** section, the parameters defined in the previous sections can be used to alter the URL or the post body to get the next page. Add or replace any part by inserting a parameter by using this format: `$MyParameterName$`. If the original URL or post body had any Dynamic Values Query parameters, these must be included if they are needed. The parameter names used in pagination must differ from the parameter names used in the dynamic values query.

4.5.1. NEMS Panorama

Select NemsPanorama as **REST API Type** if you are connecting to NEMS PanoramaAPI. This will enable the **Access ID** and **Secret Key** input fields, which are needed for this API.

REST API Type: NemsPanorama

Access ID:

Secret Key:

Invoke API

| Headers | |
|---------------|--------------------------|
| Key | Value |
| Authorization | Autogenerated by XpertBI |
| NemsDate | Autogenerated by XpertBI |

When NemsPanorama is selected as REST API Type, the Authorization and NemsDate headers are automatically generated and added to each API request by Xpert BI.

For Nems Panorama, the following XSLT Transformation file can be used:

```
<?xml version="1.0"?>
<xsl:stylesheet xmlns:xsl="http://www.w3.org/1999/XSL/Transform" version="2.0">
<xsl:output method="xml" indent="yes"/>
<xsl:template match="/">
  <root>
    <xsl:for-each select="XBI_Autogenerated_Root/XBI_Autogenerated_Element/values">
      <element>
        <xsl:variable name="dataRow" select="."/>
        <xsl:for-each select="../columns">
          <xsl:variable name="positionCounter" select="position()"/>
          <xsl:element name="{translate(id, '@', '')}">
            <xsl:value-of select="$dataRow/values[$positionCounter]"/>
          </xsl:element>
        </xsl:for-each>
      </element>
    </xsl:for-each>
  </root>
</xsl:template>
</xsl:stylesheet>
```

4.5.2. WELS Operator

Select WelsOperator as **REST API Type** if you are connecting to WELS Operator REST API. This will enable the **User Name**, **Password** and **Login URL** fields, which are needed for this API.

REST Setup

REST URL:

Timeout:

REST API Type: WelsOperator

User Name:

Password:

Login URL:

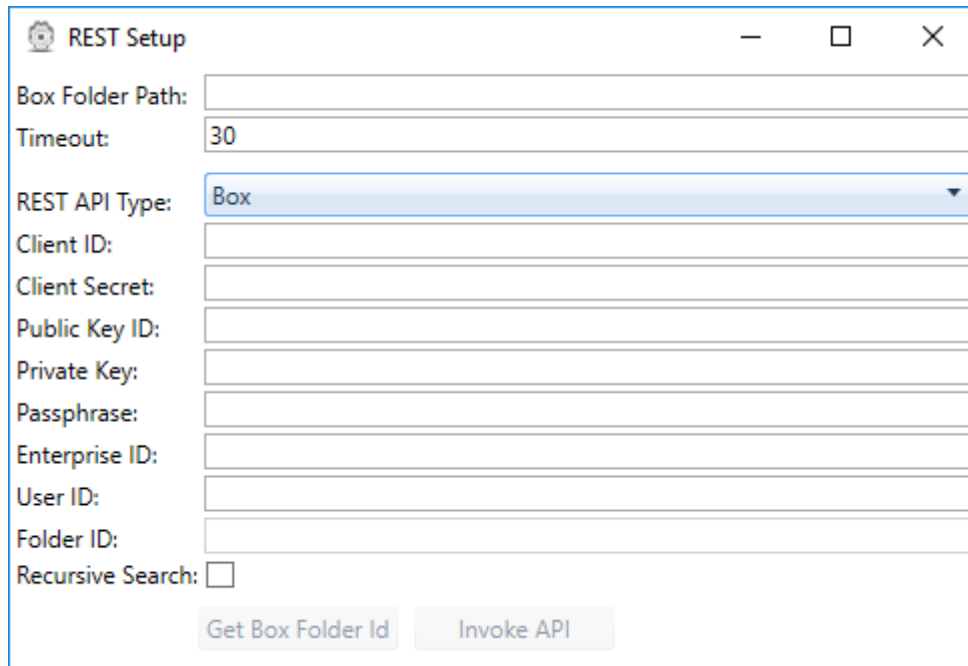
Invoke API

4.5.3. Box

Select Box as **REST API Type** if you are connecting to Box.com REST API. The Box API requires OAuth2 with JWT enabled for Xpert BI to be able to connect. Be aware that downloading files from Box.com could be quite time consuming if there are many folders and files as they have restrictions on how much traffic that can put on the API. Check the amount of API calls that can be made to the account per day and per month and limit how often the Box collection is run to avoid hitting the limit.

Xpert BI will remember what files that has already been downloaded and processed and will not download them again if they have not been updated.

There are some noticeable changes from the standard REST API setup when using Box REST API.



The screenshot shows a 'REST Setup' dialog box with the following fields and controls:

- Box Folder Path:** A text input field.
- Timeout:** A text input field containing the value '30'.
- REST API Type:** A dropdown menu with 'Box' selected.
- Client ID:** A text input field.
- Client Secret:** A text input field.
- Public Key ID:** A text input field.
- Private Key:** A text input field.
- Passphrase:** A text input field.
- Enterprise ID:** A text input field.
- User ID:** A text input field.
- Folder ID:** A text input field.
- Recursive Search:** A checkbox that is currently unchecked.
- Buttons:** Two buttons at the bottom: 'Get Box Folder Id' and 'Invoke API'.

(Fields with * supports a managed secret. See chapter 2.4.7.7 **Secret Manager** for information)

Box Folder Path specifies the named folder path to where the XML files are stored. The path must follow this format: "0\MyFolder\AnotherFolder" or "MyFolder\AnotherFolder". The "0" is the root folder at Box.com. If the files are located at root folder, the folder path must be "0". If the files are in a sub folder, the "0" can be omitted.

Client ID*/Client Secret*/Public Key ID*/Private Key*/Passphrase*/Enterprise ID* are all fields given by the JWT key pair generated in the admin console at Box.com. A note for the private key; the key must include "-----BEGIN ENCRYPTED PRIVATE KEY-----\n" in the beginning and "\n-----END ENCRYPTED PRIVATE KEY-----\n" in the end.

Folder ID* is queried from Box.com when "Get Box Folder Id" button is pressed. A valid folder id is required for the API to work.

Recursive Search will also search in sub folders of the Box Folder Path for XML files.

Get Box Folder Id checks if the Box Folder Path exists. If it exists, it saves the Folder Id.

Invoke API will try to find a single XML file and download it so it can be used for column configuration.

4.5.4. Google API

Select "GoogleApi" as **REST API Type** if you are connecting to the Google Developer API.

REST Setup

REST URL:

Timeout:

REST API Type:

Private Key:

Account:

Scope:

HTTP Verb:

Post Body:

Invoke API

Headers

| Key | Value |
|----------------------|----------------------|
| <input type="text"/> | <input type="text"/> |

Dynamic Values Query

Edit Connection

OK Cancel

(Fields with * supports a managed secret. See chapter 2.4.7.7 **Secret Manager** for information)

Private Key* to access the API. The key needs to be on the PKCS #8 format. This starts with: “-----BEGIN PRIVATE KEY-----” and ends with “-----END PRIVATE KEY-----”. It can be found in the json-file that can be downloaded from the setup-guide below. If the private key is left empty, the API call is executed without generating a security token.

Account is the access issuer. Usually a service account with an automatically generated email that has been given access to the API resource.

Scope* are the requested access types that the API call needs to have to get the data. Scopes can be found on the “OAuth 2.0 Scopes for Google API” website. Several scopes can be requested, where each scope is separated by a space.

HTTP Verb sets the request type to be either GET or POST.

Post Body is the data that is sent with the request when the HTTP Verb is set to POST.

Headers will be added to the API request as key value pairs. Click the plus icon to add new headers. Note that the Authorization header is set automatically when private key is set.

If **Dynamic Values Query** is set, the SQL query result can be used as parameters in the REST query, as described earlier. Click **Edit Connection** to specify where to execute the query. If nothing is specified the collection folder database is used.

4.5.4.1. Google API Setup Guide

This small guide contains the steps needed to perform in order to get the necessary information to use google API with OAuth 2 credentials.

- Go to the console developers dashboard for your google account
- Select a project or create a new one which is connected to your organization
- Go to the credentials page and create a service account key. Assign the key to an existing service account or create a new one. Key type should be set to JSON.
- The downloaded JSON file contains the information that is required by XpertBI. Copy the two fields “private_key” and “client_email” in to the XpertBI fields “Private Key” and “Account” respectively.

- Give the newly created service account access to a Google Service. Generally, this can be done by going to the admin page of the service and add it as a user under user management.

4.5.5. Azure Cosmos DB

Select “AzureCosmosDB” as **REST API Type** if you are connecting to Azure Cosmos DB with Core (SQL) API. The REST URL can be found in the Azure Cosmos DB account in the overview section.

(Fields with * supports a managed secret. See chapter 2.4.7.7 **Secret Manager** for information)

Master Key* is the access key needed to communicate with the Cosmos DB. It can be found in Azure under Keys in the Azure Cosmos DB account.

Resource ID* is a text identification telling the API what is going to be accessed. See the table below for usage information.

Resource Type* is the type of resource that is being accessed.

This table shows what information the API returns for a given combination of URI path, resource id and types:

| Path | Type | ID | Result |
|--|-------|---------------------------------------|---|
| /dbs | dbs | | List all databases |
| /dbs/{databaseId} | dbs | dbs/{databaseId} | Get a single database |
| /dbs/{databaseId}/colls | colls | dbs/{databaseId} | List all collections in a database |
| /dbs/{databaseId}/colls/{collectionId} | colls | dbs/{databaseId}/colls/{collectionId} | Get a single collection from a database |

| | | | |
|---|------|---------------------------------------|--|
| /dbs/{databaseId}/colls/{collectionId}/docs | docs | dbs/{databaseId}/colls/{collectionId} | List all documents in a collection in a database |
|---|------|---------------------------------------|--|

{databaseId} and {collectionId} should be replaced by the database name and collection name respectively.

To send a SQL query as input, the HTTP Verb must be changed to POST and the following headers should be present:

x-ms-documentdb-isquery : True

x-ms-documentdb-query-enablecrosspartition: True

The POST-body needs to be formatted as follows:

```
{"query": "my query"}
```

Example query to get all documents in a collection: {"query": "SELECT * FROM c"}

Note that this example uses the “List all documents in a collection in a database” row of Path/Type/ID/Result table.

When the result set is larger than what is allowed in a single request, the result is paginated. Pagination is handled automatically by XBI and no special considerations are needed.

4.5.6. Proactima

Select Proactima as **REST API Type** if you are connecting to the Proactima REST API. This will enable the **User Name**, **Password** and **Login URL** fields, which are needed for this API. Note that the Authorization header is automatically inserted and should not be added manually in the headers list.

REST Setup

REST URL:

Timeout:

REST API Type:

User Name:

Password:

Login URL:

4.6. Xpert BI Collection – SAP RFC

The Xpert BI Collection SAP RFC connector allows you to connect to SAP and execute RFCs. The result set from the RFC is stored into a table.

The screenshot shows the configuration window for the SAP RFC connector. On the left is a tree view under 'Collection' with 'MyCollection' selected. Below it is a list of actions: 'Add Table Source Flat File', 'Add Table Source XML/JSON', 'Add Table Source REST', 'Add Table Source SOAP', 'Add Table Source RFC' (highlighted), 'Add Table Source SSAS', 'Edit Collection', 'Process Data', 'Copy Exe Parameters', and 'Delete'. The main panel on the right contains the following fields and controls:

- Table Name:** GenericLongText
- Query:** exec Z_RFC_GENERIC_LONG_TEXT @CLIENT = '100', @ID = 'AVOT', @LANGUAGE = 'E', @O...
- Description:** (empty text box)
- Timeout:** 1800
- Batch Size:** 500
- Truncate Before Update:** ☒
- Add Timestamp Column:** ☐
- Import Metadata:** (button)
- Table Structure:** A table with columns: Name, Data Type, Size, Is Primary Key, Allow Nulls.

Table Name is the name of the destination table.

Query is the query that contains the exec SAP_RFC_NAME @parameter1 = XXX' ...

Description is a field where you can add a description to the table and table content.

Timeout and **Batch Size** are set to 30 and 500 respectively, but can be changed. The Timeout specifies command timeout and the Batch Size specifies number of rows per read in the bulk copy.

Truncate Before Update can be used to do a truncate of the destination table before the new data is loaded.

Add Timestamp Column can be used to include a timestamp column to the destination table.

4.7. Xpert BI Collection – SSAS

The Xpert BI Collection SSAS connector allows you to connect to analysis services and execute DAX/MDX. The result set from the SSAS is stored into a table.

The configuration window for the SSAS connector includes the following fields and options:

- Table Name:** A text input field with a red asterisk indicating it is required.
- Connection:** A dropdown menu showing 'SSAS Connection'.
- MDX/DAX Query:** A large text area for entering the query.
- Description:** A text input field.
- Timeout:** A text input field with the value '30'.
- Batch Size:** A text input field with the value '500'.
- Truncate Before Update:** A checkbox.
- Automatic PK:** A checkbox.
- Add Timestamp Column:** A checkbox.
- Import Metadata:** A button.
- Table Schema:** A table with columns: Name, Data Type, Size, Is Primary Key, and Allow Nulls.

Table Name is the name of the destination table.

Connection is the connection string to the analysis service.

Query is a query that contains either DAX or MDX.

Description is a field where you can add a description to the table and table content.

Timeout and **Batch Size** are set to 30 and 500 respectively, but can be changed. The Timeout specifies command timeout and the Batch Size specifies number of rows per read in the bulk copy.

Truncate Before Update can be used to do a truncate of the destination table before the new data is loaded.

Automatic PK can be used to include a unique identifier column that will be used as primary key. Checking this will automatically check **Truncate Before Update** and remove any set primary key.

Add Timestamp Column can be used to include a timestamp column to the destination table.

Clicking **Import Metadata** will execute the query and get the schema for the returned data.

4.8. Scheduling Xpert BI Collection data loads

To schedule data load for a table or a collection, click on the table or collection node in the Data Source Explorer and choose Copy Exe Parameters. This will copy the .exe file name with parameters of the selected item. This can be scheduled as a SSIS task or directly in SQL server as a job or through another scheduler.

4.8.1. XBI_collection.exe

Xpert BI Collection has its own executable file and does not use the same .exe file as Xpert BI Extraction and Transformation.

4.8.1.1. Installation and licensing

The XBI_collection.exe file is installed together with Xpert BI and follows the same license as the Xpert BI software. The file is placed in the folder: C:\Program Files\Xpert BI\.

The file can be copied to and run from another folder, but it is advised that you run it from the default folder to ensure you always have the latest software and file updates, and that your environment is 'in sync'.

4.8.1.2. Parameters

XBI_collection.exe is called with the following parameters

- FolderId (mutually exclusive with TableId) – application generated ID to identify the folder containing a collection of tables
- TableId (mutually exclusive with FolderId) – application generated ID to identify the table
- ServerName – the name of the server where the management database is installed
- DatabaseName – the name of the management database

The parameter values are easily retrieved by right-clicking on the collection or table and choosing Copy Exe Parameters.

For error codes and scheduling see section Error codes and section Scheduling. For error message handling see Error message handling.

4.8.1.3. Command Window Information

When the XBI_collection.exe has started, a command window will show. Information about the current task is displayed in this command window. The following information is available:

- **Execution start** Timestamp showing when this task was started.
- **Executing as** The windows user that started the executable.
- **Command arguments** All the command line arguments given to XBI_collection.exe
- **Path** Path to the collection folder
- **Executing collection folder** If executing a folder collection, this line is shown with the name of the collection folder.
- **Executing collection table** If executing a table collection, this line is shown with the name of the collection table.
- **Collection type** Type of collection (FlatFile, REST, SAP RFC, SOAP, XML).

4.8.2. Data loading

Xpert BI Collection uses the defined primary key to determine whether to update a row or insert a row in the destination table.

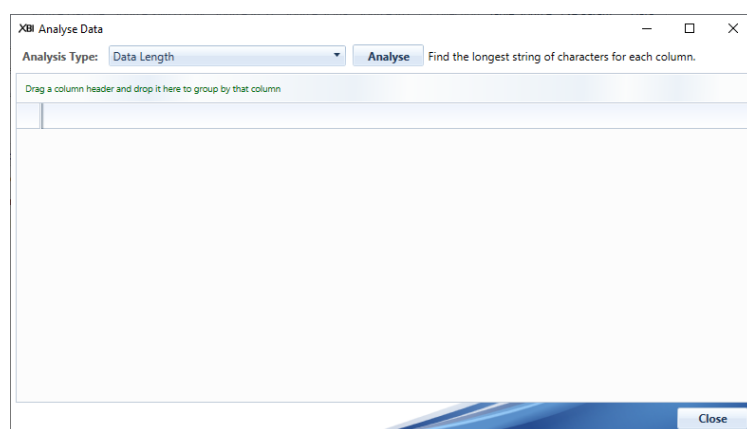
If the *Truncate Before Update* checkbox is checked, all data rows will be inserted into the table.

If the *Add Timestamp Column* checkbox is checked, all data rows which are either updated or inserted is updated with the new timestamp.

Note: To track changes of snapshot data, it is recommended to load the data through Xpert BI Extraction and run the 'Tracking/Changing' functionality to get delta values and to- and from-dates on the data. (See Tracking table values or attribute changes.)

4.9. Analyse Data

The Analyse Data can perform different analysis on the files in the input folder.



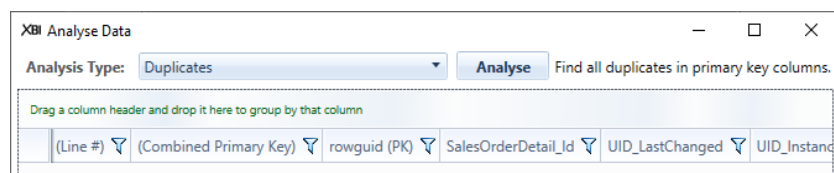
In the **Analysis Type** dropdown list, the desired analysis is selected. See below for the different types of analysis.

The **Analyse** button executes the analysis.

Analysis Types:

Data Length will find the longest string of characters for each column.

Duplicates will find all duplicates in the primary column(s). If automatic PK is enabled, this analysis cannot be run.



The first two columns are added during analysis. These show the data line that the duplicate is on (valid for Flat file collection) and the combined primary key that is the duplicate. The combined primary key is a string concatenation of all the primary keys selected. This is the value that is checked for duplicates.

Tip: When dealing with multiple duplicates, it can be beneficial to drag the combined primary key column up in the "Drag a column header and drop it here to group by that column" area. This will group on combined primary key and each group will contain all the rows that are duplicate. Making it easier to identify the different duplicates.

5. Xpert BI Publication

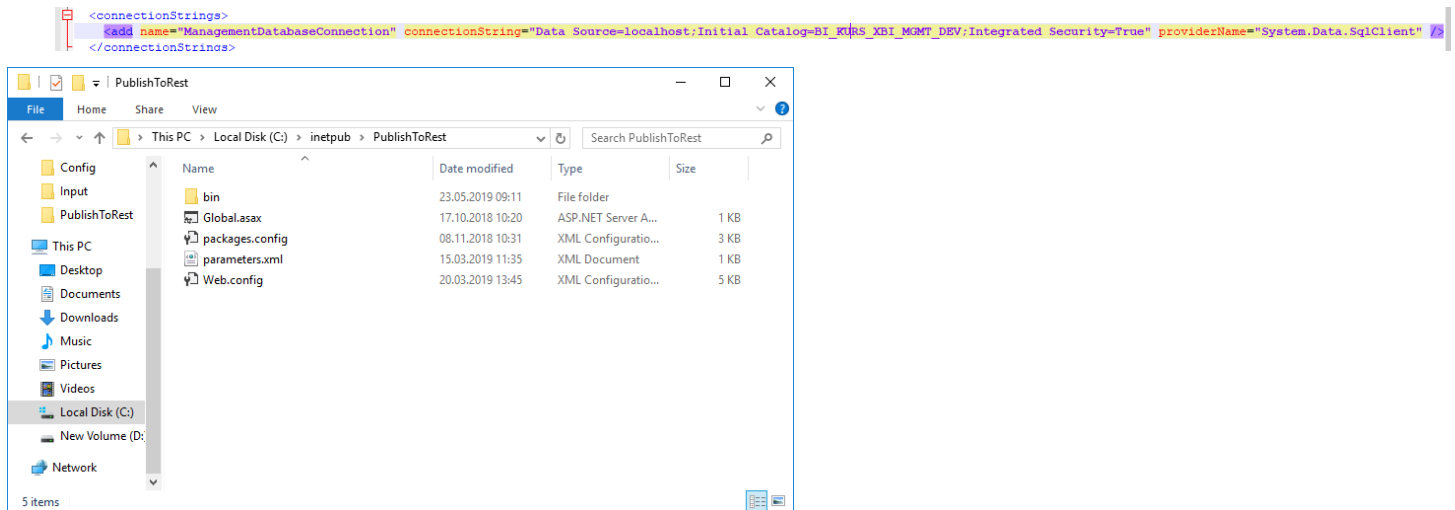
5.1. REST API

Xpert BI allows you to publish any table or view in the Xpert BI stack to a read-only REST API.

5.1.1. Installation

The APIs are managed from Xpert BI and hosted in IIS. Before the APIs can be used, it has to be installed/deployed to IIS. There are several methods to deploy the API to IIS, two of them are:

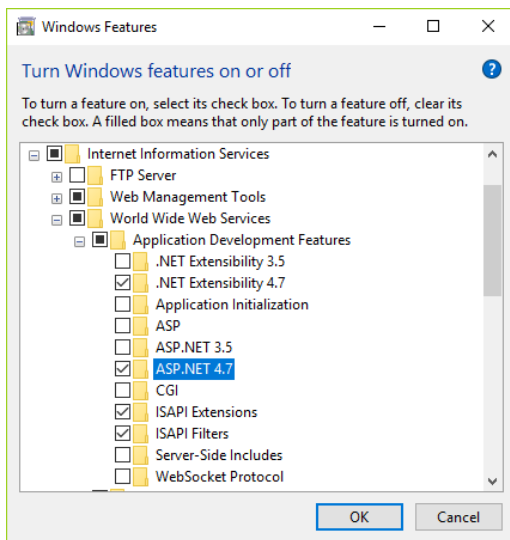
- Import the application, by right-clicking the site (in IIS Manager), Deploy -> Import Application and choosing the RestApiGenerator.zip found in the Xpert BI installation folder (Xpert BI\Include). During import you have the option to change the Management connection string. This needs to point to your management database (YourPrefix_XBI_MGMT_DEV) for the API to work.
- Manually copy the content of the '\Content\RestApi\' folder (inside the RestApiGenerator.zip) to the IIS site folder. See image below for how the IIS site folder should look like. When installing this way, you have to set the connection string to the Xpert BI Management database afterwards, either directly in the Web.config file or by changing it in IIS (in the Connection Strings settings page).



Site folder after manual installation

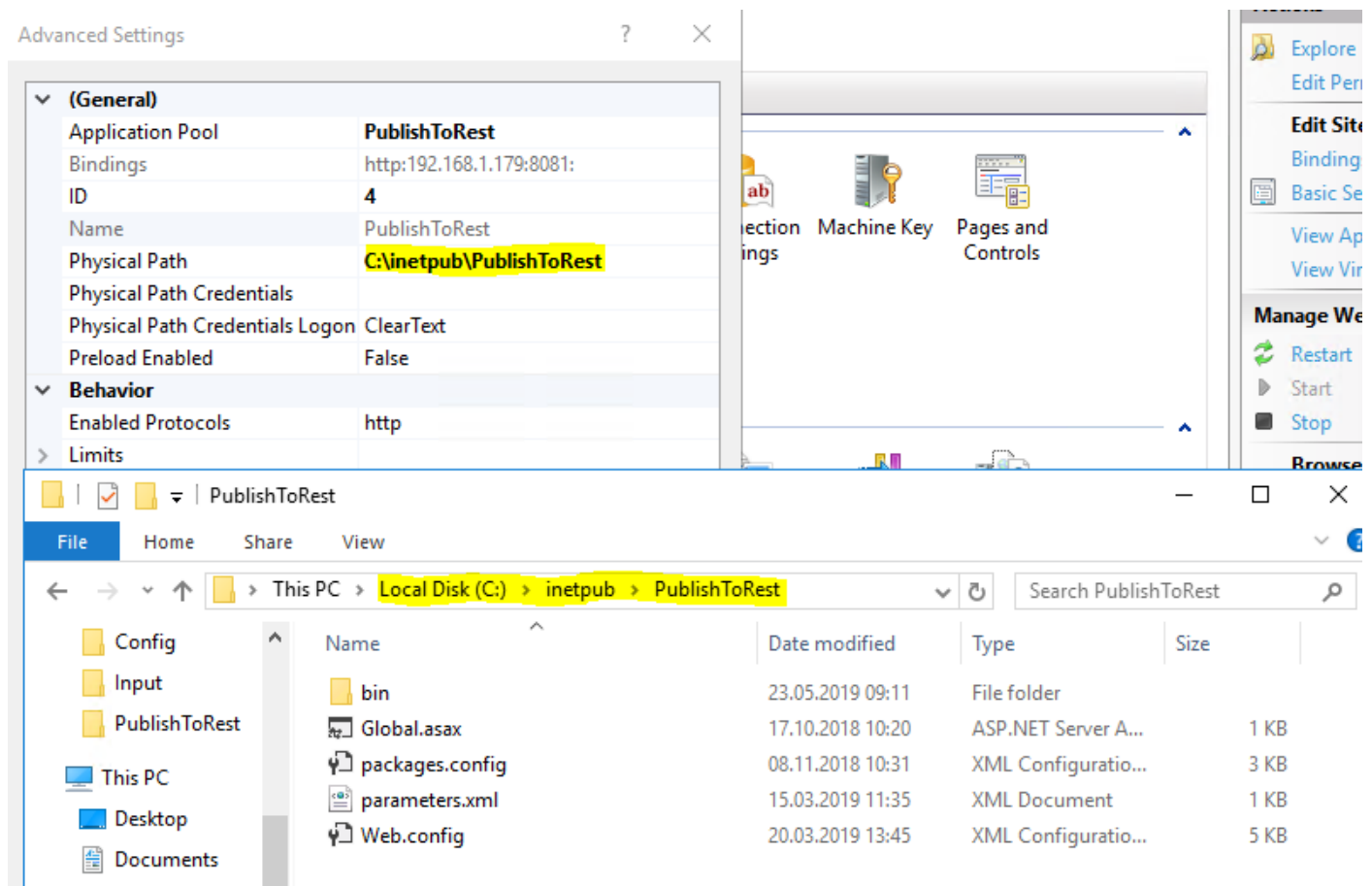
Note: If any API is set up to use Bearer Tokens as security measure, the hosting needs to be setup to use SSL.

Note: ASP.NET has to be enabled for IIS (as a Windows feature) for the API to work:



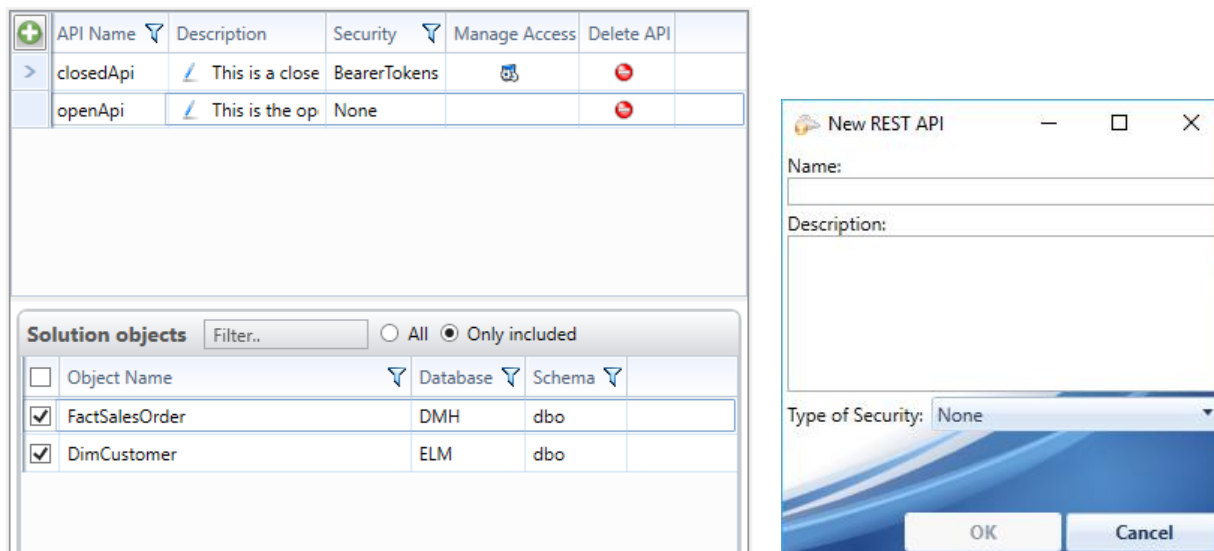
Note: The Windows/service user for the application pool that the API runs in has to be given access to the Xpert BI database stack. For security reasons it is recommended to only give this user read access. The user can be changed (in IIS) under Application Pools, Advanced Settings, Identity.

Note: When installing using the import application feature, make sure that the site physical path and actual path of the installed files are matching. Right click on the site and click on 'Advanced Settings...' under 'Manage Website'.



5.1.2. Creating and Managing an API

After the API is installed in IIS, you manage your APIs from Xpert BI. To do this, click the "REST APIs" button in the Publication ribbon tab. This will open a new tab in the main window that lets you create and manage APIs.



To create a new API, click the plus-button in the upper left corner of the tab window, give it a name, description and set the security type. Click OK to save.

The name of the API will be used in the access URL for the API. If the API is hosted at *www.yourWebsite.com* and the name of the API is *myApi*, the URL will be *www.yourWebSite.com/myApi*

The description is used in the documentation, which is open for anyone. Setting the security to None, will result in an open API (if there are no additional security configured in IIS), setting it to BearerTokens or MicrosoftOauth2 will configure the API to use bearer tokens for security.

To choose which tables/views are included in an API, you first select the API you want to configure in the upper list and then select objects to be included from the bottom list. The filter box lets you filter by object name. Clicking the **Only included** radiobutton will filter the list to only show included objects. **All** will show all available objects.

To delete an API, click the Delete button on the selected API. This will also delete any tokens for this API.

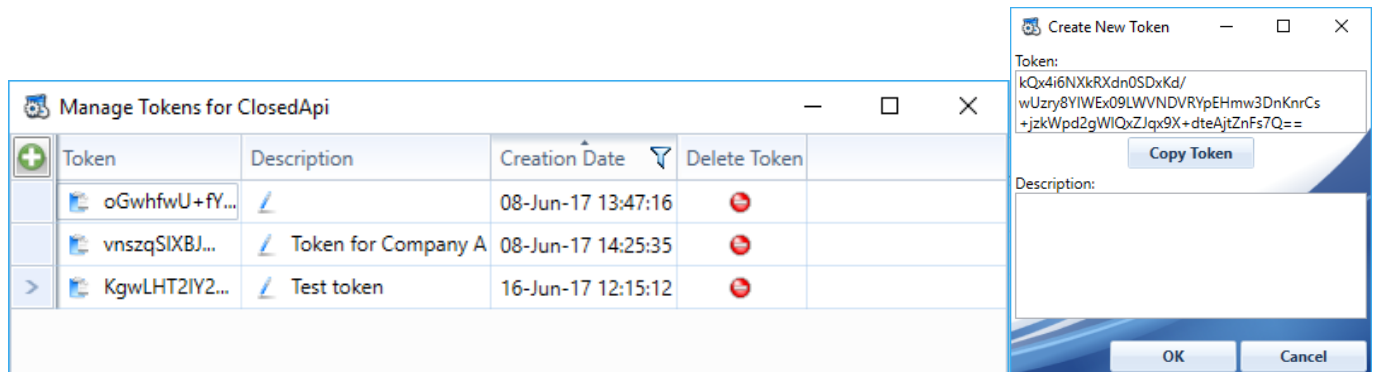
Note: Objects added to an API needs to have unique unqualified names (the object name excluding database and schema) within an API. For example, *XX_ETL.dbo.myTable* and *XX_ELM.dbo.myTable* cannot be added to the same API.

5.1.3. Managing Security

There are two types of security available. A simple bearer token and a Microsoft OAuth2 bearer token.

Static Bearer Token

If an API uses bearer tokens as security, the tokens are managed from Xpert BI. Click the Manage Access button for the API in question. This opens a new window, listing all available tokens for this API.

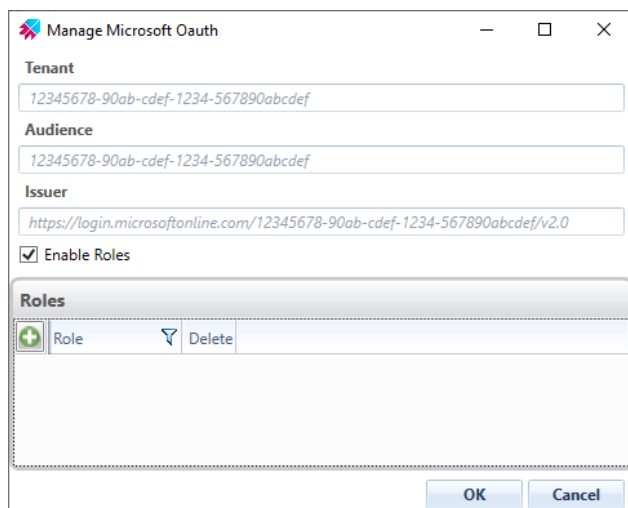


To create a new token, click the plus button in the upper left corner. The token string is automatically generated. Give it a description and click OK to save it. The description can be edited later, by clicking the Edit Description button (inside the Description column) on the selected token. To delete a token, click the red delete button in the Delete Token column. Deleting a token will revoke access to the API for anyone who was using that token.

To copy a token string to the clipboard, click the Copy Token button inside the Token column (or copy the token when it is created).

Microsoft OAuth2 Bearer Token

When using this type of security, some additional settings must be included in the API. To configure these settings, click on the Manage Access button for the API to open the configuration window.



Tenant is the tenant ID which is going to issue the token.

Audience is the application/client ID for the app registration that is managing the access to this API.

Issuer is the URL that issued the token.

Turning on **Enable Roles** will require the token to include a role in order to access the API.

To register an app, it is required to have administrator access to the Azure portal and the Azure Active Directory. In the Azure Active Directory click on App registrations and then on New Registration to register a new app. Give it a name and applicable account support types. After creating the app, go to Expose an API. Add a scope and give it a name and description. After this is done, go to the Overview where you can find the necessary values to enter into XpertBI.

Tenant is the 'Directory (tenant) ID'.

Audience is the 'Application (client) ID'.

Issuer is dependent on the token version. For version 1 the issuer is '<https://sts.windows.net/{tenantId}>' for version 2.0 the issuer is '<https://login.microsoftonline.com/{tenantId}/v2.0>'. Where {tenantId} is replaced by the tenant value.

5.1.4. API Functionality

The API has two main methods:

Table Info

This method returns a list of the tables (or views) included in the API, their description (given in Xpert BI) and a list of the available columns in each table (column name, data type and whether it can be empty or not). Example URL for this method: *www.myWebSite.com/myApi/GetTableInfo*

Where 'myApi' is the name of the API defined in XpertBI

To return information on a single table, use: *www.myWebSite.com/myApi/GetTableInfo/MyTableName*

The default return format is JSON.

To get the result in XML format, append .xml to the GetTableInfo URL.

To get the result in JSDoc (JavaScript documentation) format append ".jsdoc" to the GetTableInfo URL (*www.myWebSite.com/myApi/GetTableInfo.jsdoc*). This will return JSDoc (as plaintext) of the tables/views included in the selected API, which can be used to (among other things) achieve IntelliSense/autocomplete when working with the API in JavaScript.

A Swagger/OpenApi specification for the API can also be returned from this method, by appending ".OpenApi" to the URL (*www.myWebSite.com/myApi/GetTableInfo.OpenApi*). The return format is JSON. This specification can be used to generate API-clients in many different languages using Swagger CodeGen, Swagger Editor or directly inside Visual Studio. To create a client in Visual Studio, right-click a project, select Add -> REST API Client, point to the OpenApi specification and click OK. Note that if GetTableInfo is called with the OpenApi extension, MyTableName will be ignored; the OpenApi specification will always contain all objects included in the API.

Data Retrieval

This method is used to retrieve data from the available tables, and its URL looks like this
www.myWebSite.com/myApi/MyTableName

.xml can be appended to MyTableName to return XML, otherwise JSON will be returned. Calling the method this way will return all the rows and all the columns from the table. Several query parameters can be used to limit/filter the result:

| Parameter keys | Description | Parameter value formats |
|----------------|--|--|
| Limit | Limits the returned result set to the given number of rows | 10 (only integer values allowed) |
| Sort | Allows for sorting of the result set | mySortColumn1:desc,mySortColumn2:asc |
| Fields | Is used to limit the number of fields returned | myColumn1,myColumn2,myColumn5 |
| Where | Is used to filter the result set. | myColumn2:English,myColumn3<:4,myColumn>:2 |
| ID | Returns the element with the given ID | 17 (can be text or number) |

Note that the ID column is assumed to be the first column in the target table/view.

Note that the API is accessed using http/https GET.

Query Examples

To retrieve table information on DimContact in XML format, from myApi, use:
www.myWebSite.com/myApi/GetTableInfo.xml/DimContact

To retrieve all data from DimContact in JSON, from myApi, use:
www.myWebSite.com/myApi/DimContact

To retrieve the first 10 rows from DimContact in XML format, from myApi, use:
www.myWebSite.com/myApi/DimContact.xml?limit=10

To retrieve the first 10 rows from DimContact with only MyField1 and MyField2 returned, sorted by SortColumn1 descending and SortColumn2 ascending, in JSON and from myApi, use:
www.myWebSite.com/myApi/DimContact?limit=10&Sort=SortColumn1:desc,SortColumn2:asc&Fields=MyField1,MyField2

Query response example

The JSON response when retrieving data will be in a format similar to this:

```
[
  {
    "Contact_Id": 0,
    "EmailAddress": "example@example.com",
    "FirstName": "someName",
    "LastName": "someLastName"
  },
  {
    "Contact_Id": 1,
    "EmailAddress": "example2@example.com",
    "FirstName": "someOtherName",
    "LastName": "someOtherLastname"
  }
]
```

Token Usage

If the API is secured using bearer tokens, a valid token has to be sent with each request to the API. The token is set in the Authorization header in the format: Bearer <myToken>

(without the <>). Example Authorization header: Bearer MySuperSecretToken

5.1.5. API Documentation

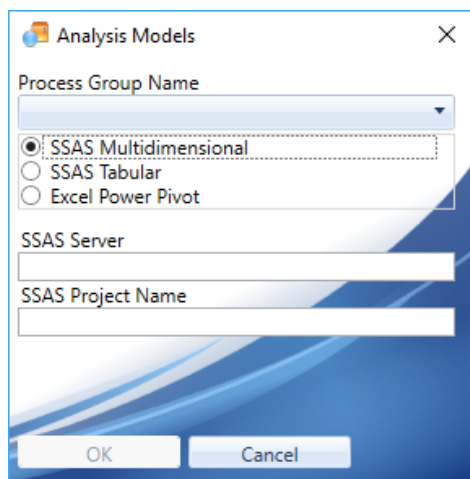
Querying www.MyWebSite.com/myApi will bring up a HTML documentation of *myApi*. This page is open, even if the API is secured using bearer tokens. That documentation contains the name and description of the API, a description of the functionality, query parameters, example queries, example responses, and information on security (if the API uses bearer tokens).

5.2. Create analysis models

With Xpert BI, you can create analysis models from the process groups defined in the Xpert BI Solution. All fact and dimension tables belonging to the selected Process Group Name is used when generating the analysis model. Relations and keys are defined in Fact Table Dependencies.

Note: For all model types, the fact tables in the chosen Process Group need to be valid (in Fact Table Dependencies).

5.2.1. SSAS Multidimensional

The screenshot shows the 'Analysis Models' dialog box. It has a title bar with a close button. Below the title bar is a 'Process Group Name' dropdown menu. Underneath is a group box containing three radio buttons: 'SSAS Multidimensional' (which is selected), 'SSAS Tabular', and 'Excel Power Pivot'. Below the radio buttons are two text input fields: 'SSAS Server' and 'SSAS Project Name'. At the bottom are 'OK' and 'Cancel' buttons.

SSAS Server: SQL Server Analysis Services instance. (Only SSAS Multidimensional)

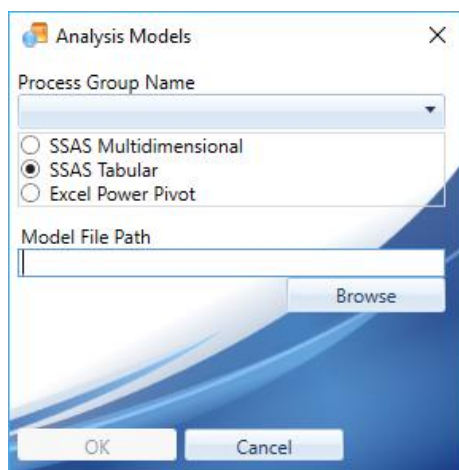
SSAS Project Name: Analysis services database name. (Only SSAS Multidimensional)

Creates a new, or overwrites an existing database on the SSAS Server and configures a data source and a data source view from the selected Process Group Name.

5.2.2. SSAS Tabular

Xpert BI can publish to an existing Tabular model (requires SSAS Tabular 2016 or newer). Select a process group, browse to the model file (*.bim) and click OK.

Note: You need to have an existing Visual Studio project, Xpert BI does not create the project for you. If you do not already have a visual studio project, you will need to create an empty project before starting the wizard.

The screenshot shows the 'Analysis Models' dialog box. It has a title bar with a close button. Below the title bar is a 'Process Group Name' dropdown menu. Underneath is a group box containing three radio buttons: 'SSAS Multidimensional', 'SSAS Tabular' (which is selected), and 'Excel Power Pivot'. Below the radio buttons is a 'Model File Path' text input field with a 'Browse' button to its right. At the bottom are 'OK' and 'Cancel' buttons.

This will open a pane like this:

| Source Name | Target Model Name |
|--------------------|-------------------|
| FactEmployeeSalary | EmployeeSalary |
| FactSalesOrder | SalesOrder |
| Columns | |
| Measures | |
| Relationships | |
| DimTime | Time |
| DimProduct | Product |
| DimCustomer | Customer |
| DimAddress | Address |
| DimEmployee | Employee |

Xpert BI will combine the list of tables from the process group with the tables already defined in the tabular model (if any) based on source table name (and source column name for columns) to create the list of all tables. The **Source Name** is the name in the source and the **Target Model Name** is the name a table/column/measure have or will receive in the target model. Tables that are equal in source and target will have a green checkmark next to them, a yellow star indicates that there is a change and a red delete icon indicates that the table is only found in the model and not in Xpert BI.

Each table can be expanded to see their columns, measures and relationships.

To update to the model file, click Save on the ribbon bar. Any changed tables will be updated, new tables/columns/measures/relationships will be created and any deleted tables/columns will be removed.

Note: Calculated columns created in the cube will get marked with a red delete icon – but these **will not be deleted** from the model. Only base columns/tables which no longer exist in Xpert BI will be deleted from the model.

Columns ending with "_Id" will be hidden and all data types will be converted from SQL to Tabular.

Columns with SQL datatype float, real, decimal or numeric will be hidden and the name prefixed with "Hidden_". A measure (sum of hidden column) will be created, with the name of the original column. If you do not want a measure to be autogenerated for a column, right-click the column and select **Do not autogenerated measure**. This will remove the "Hidden_" part from the name, make the column visible and remove the associated measure. If you want to create the measure anyway, right-click the column and select **Autogenerate measure**.

To hide/unhide a column, right-click it and select **Hide/Unhide**.

Information on measures that shall not be autogenerated and columns that are hidden/visible are stored in the model file as annotations.

To delete a table from the model (without deleting it from Xpert BI or in the database), right click it and select Delete from model.

Note: changing data type on a column from one that creates an auto generated measure to one that does not, will not remove the measure and will not "unhide" the column.

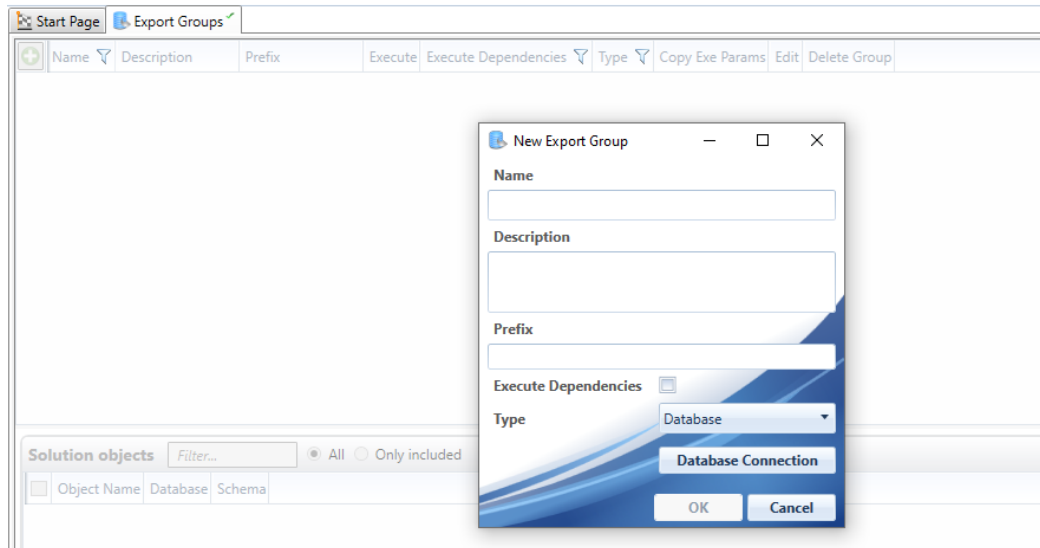
Note: deleting a relationship (in Fact Table Dependencies) will not delete the relationship in the model, unless one of the tables on either side of the relationship is also deleted.

5.2.3. Excel Power Pivot

Creates a new Excel workbook and loads a power pivot model from the selected Process Group Name.

5.3. Export Groups

An Export Group is a grouping of one or more objects which can be exported to various targets. To create an Export Group, click **Export Groups** under the Publication ribbon pane and then click the green plus-sign in the upper left corner of the pane. This will open the **New Export Group** window, as seen below.



Name is the name of the export group. This must be unique.

Description is the description of the group.

Prefix will prepend text to the exported table name. This does not apply to REST type. It is possible to use the following keywords `{DB}` and `{DB_LOGICAL}`. They are case sensitive.

`{DB}` will be replaced with the entire database name, eg. *My_Stack_ELM*.

`{DB_LOGICAL}` will be replaced with only the logical database name, eg. *ELM* or *ETL*.

Checking **Execute Dependencies** means that whenever this export group is processed, all dependencies (executable objects) will be executed before the process group is exported.

Type lets you choose what kind of export group this shall be:

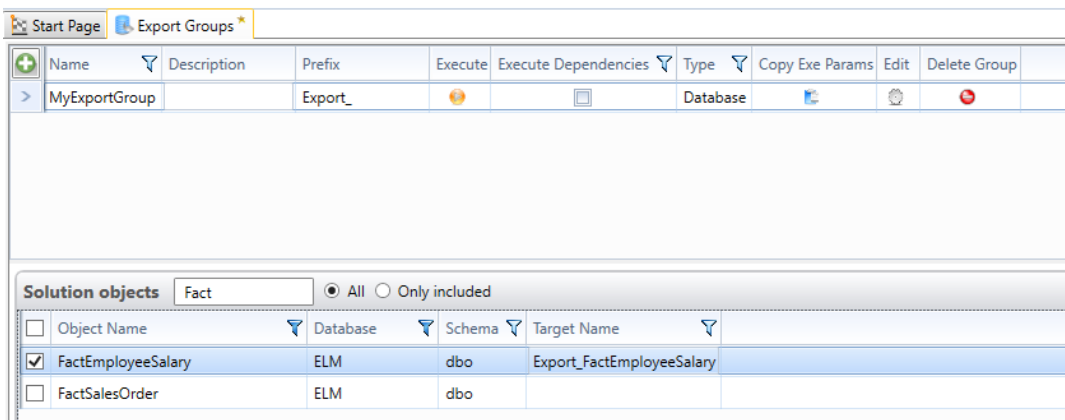
- **Database** lets you set up a database connection. An export group of this type will, for each view or table included in the group: create a temporary table in the destination database, load data into it, drop the original table if it exists and then rename the temp table. Note that if the target is an Azure SQL Data Warehouse database the objects included in the group must be only views or tables without primary key.
- **CSV** lets you browse to an output path. When this type of group is processed it will export the content of the included views/tables into CSV-files in the output folder using the selected delimiter. The files will be named the same as the views/tables, and if they already exist they will be overwritten. It is possible to use SQL query to set either a header or a footer. The rows and columns of the query will be written to the file as CSV. The keyword `{XBI_TableName}` can be used to insert the table name that is being exported into the SQL query.
- **REST** lets you configure a REST-API method that will be called for each view/table in the export group. The data from the views/tables will be included in the request body. If the view returns a single column/single

value, the value is sent raw in the body. The POST data can also be converted to JSON. Request headers can be set as key-value pairs.

- **Azure Data Lake** lets you configure a connection to Gen2 storage used by Azure Data Lake. Each table will be converted to CSV format and uploaded to the specified folder. Root folder is “\”. The files will be named the same as the views/tables, and if they already exist they will be overwritten.

If export target is a table there is a naming restriction enforced by Azure. Table names must be alphanumeric, cannot begin with a number, and must be between 3 and 63 characters long. Selecting “Append run batch to table” will append the exported data to the table where the partition key is extended with the timestamp. If this option is not selected, the table will be dropped and created before data is exported.

Note that dropping and creating a table in Azure could potentially take a minute or two.



To select which objects are included in an export group, select the group in the top grid and then check the checkbox for each object you want to include in the bottom grid. It is not possible to select two objects with the same name. Click save on the ribbon bar to save the changes.

Target Name is the consolidated name based on prefix and object name. The object name which is used can be changed by clicking in the Target Name cell.

To execute an export group, click the yellow button in the **Execute** column, or click the **Copy Exe Params** button to copy the exe parameters, which can be run directly in the command line or scheduled through any scheduler.

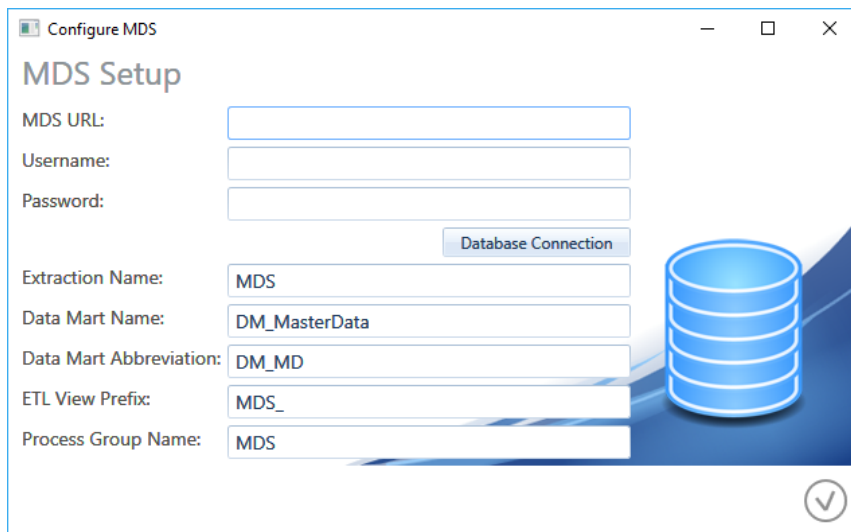
Click **Edit** to edit the settings of the export group.

6. Master Data

The Master Data module integrates with Microsoft Master Data Services (MDS), and lets you easily create and manage entities, subscription views and data loads.

6.1. Configuration

To configure the MDS connection, click the **Configure MDS Connection** in the Master Data ribbon bar. This opens the configure MDS window, as seen below.



MDS URL is the URL to the Master Data Services (including the svc ending).

Username and **Password** is a username and password with access to MDS.

Database Connection lets you configure the connection to the SQL server where the MDS database is located.

The below settings only apply if you choose to include any subscription views in the Xpert BI stack (more info on that later).

Extraction Name is used to determine the name of the data source in extraction.

Data Mart Name is the name of the database where the data subscription views will be created. If the stack is schema-based it is a schema in the stack database.

Data Mart Abbreviation is the abbreviated name of the data mart used by Xpert BI in Additional Databases. This option is not available if the stack is schema-based.

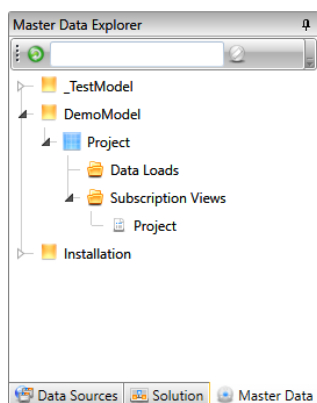
ETL View Prefix is the prefix used for any views generated in ETL.

Process Group Name is the name of the process group which all inline views in the data mart will be added to.

6.2. Master Data Explorer

Clicking the Master Data ribbon tab will open the Master Data Explorer which contains all the models, entities, data loads and subscription views found in the configured Master Data Services. The search field at the top can be used to filter the tree, and the green reload button in the top left corner is used to reload the explorer.

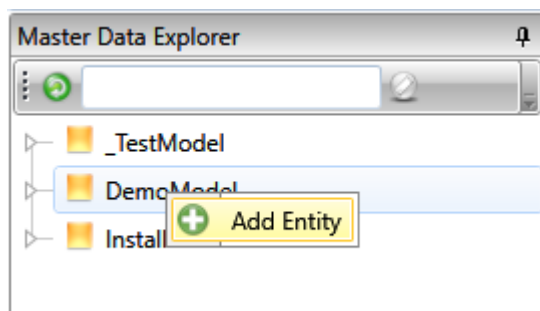
When Xpert BI is opened, the Master Data Explorer is hidden. It becomes visible when you click the Master Data ribbon tab and then it will remain visible until Xpert BI is closed.



6.3. Entity

6.3.1. Create Entity

To create a new entity, right-click the model under which you want to create an entity and select **Add Entity**.



This opens up the New Entity pane, which lets you configure the new entity.

| Source Name | Display Name | Description | Display Width | Data Type | Length | Decimals | Input Mask | Attribute Type | Domain Entity | Change Tracking | Change Tracking Group |
|-------------|--------------|-------------|---------------|-----------|--------|----------|------------|----------------|---------------|-----------------|-----------------------|
| | | | | | | | | | | | |

Source Object is the view or table in the Xpert BI stack that you want the entity to be based on. Entering a name here will autofill the **Entity Name** and the **Staging Table Name**, but these can be changed afterwards if needed. The **Attributes** list will be filled based on the columns in the selected **Source Object**. The properties on the attributes can be changed if you want to.

The **Name** and **Code** Columns has to be mapped to one of the columns in the selected source object.

Require Approval, **Data Compression** and **Autogenerate Code Values** are properties on the entity. If **Autogenerate Code Values** are checked, **Code Starts With** must be filled out (with a number).

Description is the description of the entity.

To save the new entity, click the Save button on the Master Data ribbon bar.

Note: to set the data length to max for text attributes, use 0 as **Length**.

6.3.2. Edit Entity

To edit an entity, right-click it in the Master Data Explorer and click Edit Entity. This will open a pane that looks exactly like the new entity pane, except it is filled out based on the properties and attributes of the selected entity. The attributes list are disabled until you change the source object. If the source object is changed, all of the attributes (except name and code columns) and their data will be deleted and then the new attributes will be created.

Note: Changing the entity staging name is not fully supported by MDS. A new staging table and staging stored procedure will be created, but the old will not be deleted correctly.

6.3.3. Delete Entity

To delete an entity, right-click it in the Master Data Explorer and click Delete Entity. Note that in addition to deleting the entity and all its entity members, any subscription views and data loads created for this entity will also be deleted.

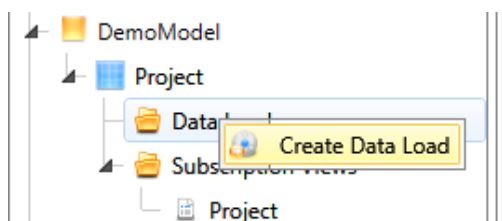
6.4. Data Load

A data load is a way of loading data from a table or view in the Xpert BI stack into an entity in MDS.

6.4.1. Create Data Load

To create a new data load, right click on an entity or its Data Loads folder and select Create Data Load. The code and name column must be defined on an entity before a data load can be created for it. If the entity was not created in Xpert BI, you must first edit it to set the name and code columns before you can create a data load.

Note: The code column cannot contain null-values.



This opens the New Data Load pane, as seen below.

A screenshot of the 'New Data Load' pane in the MDE. The pane has a title bar with 'Start Page' and 'New Data Load' tabs. It contains several input fields and buttons. The 'Name' field is set to 'Project_VERSION_2'. The 'Version' field is a dropdown menu set to 'VERSION_2'. The 'Source object' field is set to 'ETL.Project'. There are two checkboxes: 'Validate after load' and 'Automatic approval', both of which are currently unchecked. To the right of the input fields are two buttons: 'Execute' and 'Copy Exe Parameters'.

Name is the name of the data load.

Version is the version name the data will be loaded into.

Source Object is the view or table in the Xpert BI stack the data is selected from.

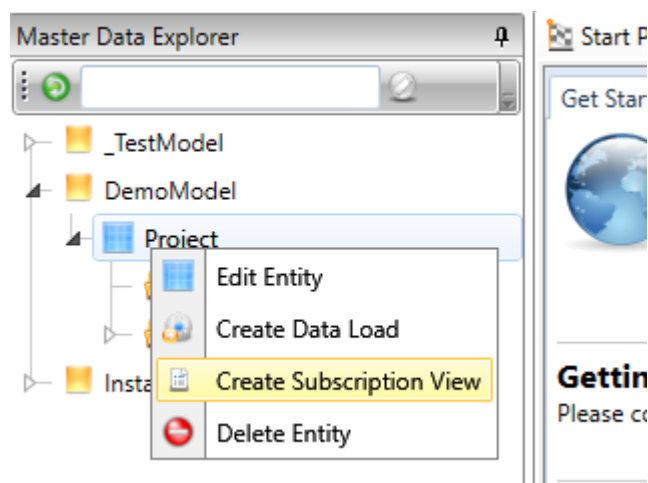
If **Validate after load** is checked, the validation process (running the business rules) will be run after loading the data. If **Automatic approval** is checked, data loads will automatically be approved, in case the entity is configured to require approval for data changes.

Click the **Execute** button to execute the data load immediately, or click the **Copy Exe Parameters** button to copy the exe parameters to the clipboard. This can then be scheduled similarly as the XBI_batch.exe or XBI_collection.exe. A data load can also be executed by right-clicking it in the Master Data Explorer and choosing Execute Data Load.

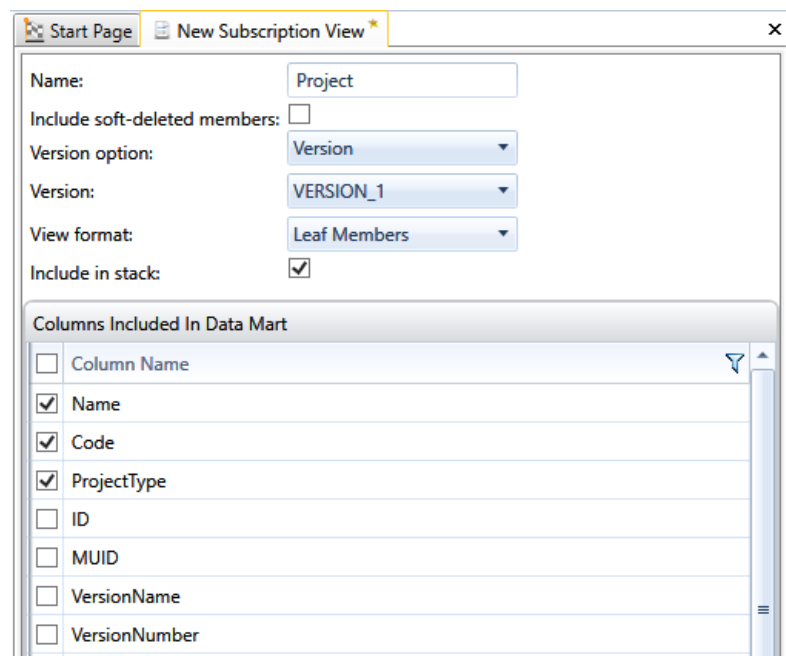
6.5. Subscription View

6.5.1. Create Subscription View

To create a new subscription view, right-click on an entity or its Subscription Views folder and select Create Subscription View.



This opens the New Subscription View pane, as seen below.



Name is the name of the subscription view.

Checking **Include soft-deleted members** will include the soft-deleted members in the subscription view. This will add a State column to the view, containing information on the state of each row (entity member).

Version option lets you select if this subscription view shall be based on a Version or a Version Flag.

Version lets you choose the version this subscription view shall be based on.

Version Flag (only visible if Version Flag is selected as **Version option**) lets you choose which version flag the subscription view shall be based on.

View format lets you choose what type of subscription view this shall be.

If **Include in stack** is checked the following will happen when the subscription view is saved:

- A data source will be created in extraction (if it does not already exist) using the extraction name and SQL connection configured in MDS Configuration settings.
- Metadata will be imported based on the subscription view.
- The table and columns will be activated in extraction (in Edit Metadata).
- Truncate before update (data load setting) will be turned on for the table.
- Check consistency will be run for the data source.
- A view will be created in ETL, based on the ODS table. The view will be prefixed with the ETL prefix from MDS configuration. A warning will be given if the view name is already taken.
- A data mart will be created (if it does not already exist) and added as an additional database to the Xpert BI stack. The name and abbreviation will be taken from the MDS configuration. If the stack is schema-based, the data mart is a schema in the stack database, and is therefore already included in the solution.
- A view will be created in the data mart, based on the ETL view.
- The data mart view will be inlined and the inline table will be added to a process group. The name of this process group is taken from the MDS configuration.

To save the new subscription view, click the Save button on the Master Data ribbon bar.

6.5.2. Edit Subscription View

To edit a subscription view, right-click it in the master data explorer and select Edit Subscription View. This will open a pane exactly like the one for create subscription view. The only difference is that when saving (if the Include in stack checkbox is checked), there will not be given a warning if the ETL or data mart view exists, they will simply be overwritten.

6.5.3. Delete Subscription View

To delete a subscription view, right-click it in the master data explorer and select Delete Subscription View. Note that it is only the subscription view in MDS that is deleted; it will not delete anything from extraction, ETL or the MDS data mart.

7. DataOps

The objective of the DataOps module is to ensure data quality.

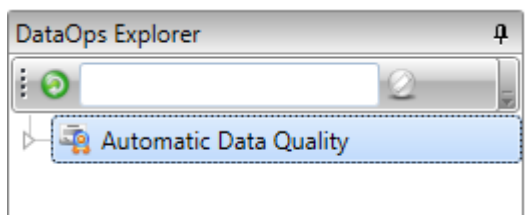
The “Automated Data Quality” section contains predefined tests. These tests can either be run directly from XpertBI or scheduled and run from a batch file. By using a folder structure for storing these tests, one can easily group and schedule multiple tests by executing the tests using the folder ID.

The “Unit Test” section contains user defined tests. These tests are user defined and can run SQL, MDX or DAX code to create datasets which are compared to an expected result.

All tests that are executed are logged in the DataOps log.

7.1. Automated Data Quality

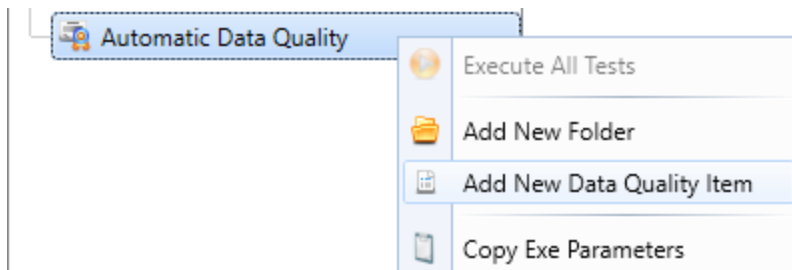
Automated Data Quality is accessed from the DataOps Explorer.



7.1.1. Data Quality Item

7.1.1.1. Add New Data Quality Item

To add a new data quality item, right click Automatic Data Quality and select “Add New Data Quality Item”.



The following window will pop up:

A screenshot of the 'New Data Quality Item' dialog box. It has a title bar with a gear icon and a close button. The form contains three fields: 'Name' with a red asterisk and a text input field containing 'e.g. SAP Duplicates'; 'Type' with a red asterisk and a dropdown menu; and 'Description' with a large text area. At the bottom are 'OK' and 'Cancel' buttons.

The **Name** must be different from the other items inside the selected folder. Select type to choose which type of test to perform. Description is optional.

Press OK button to continue.

7.1.1.2. Edit Data Quality Item

Common to all tests, the following information about the test is available:

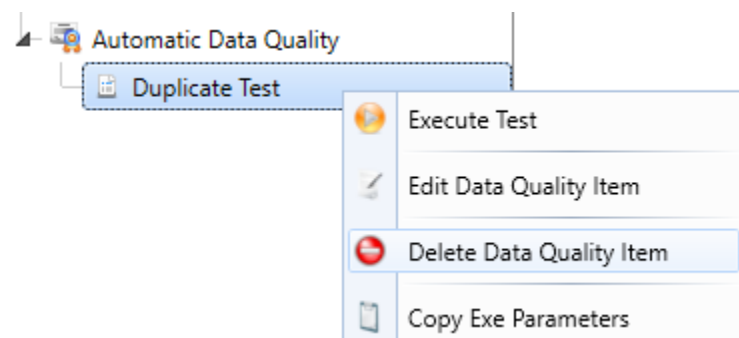
| Test Name | Explanation | Test Description | Comments |
|------------|--|----------------------|------------------|
| Duplicates | The duplicate test searches for any duplicates of the selected column(s) from the selected object(s) inside the selected process group(s). | Check for duplicates | Work in progress |
| Test Type | | | |
| Duplicates | | | |

Every element below these textboxes may be different for each test type, as each test depends on different variables.

To edit information inside the test, simply change the information and click the save button from the ribbon to save the item.

7.1.1.3. Delete Data Quality Item

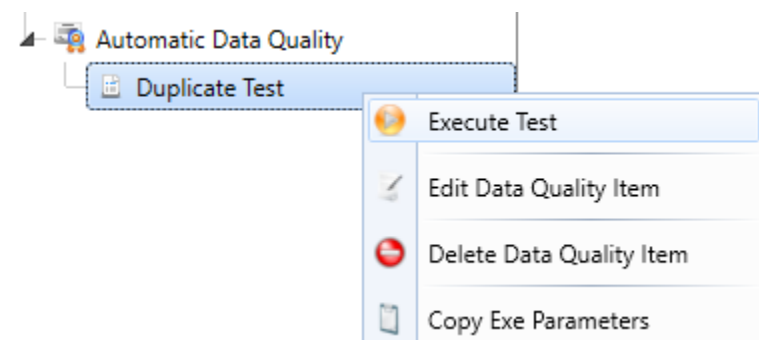
To delete an item, right click the item and select “Delete Data Quality Item”.



Alternatively, select the desired item and press the delete button on your keyboard.

7.1.1.4. Execute Test

To execute a test either open the item and click “Execute” from the ribbon bar, or right click the item and select “Execute Test”.



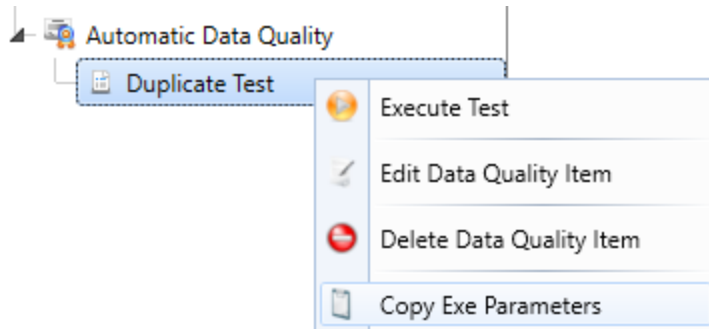
7.1.1.5. Schedule a data quality item

To schedule a data quality item, right click the item and select “Copy Exe Parameters”. Use the copied command line in i.e. SSIS to schedule the test. The command line contains the path to a batch file that will be run when executed.

The batch file will return one of the following outputs:

0 – Test is OK

1 –Test is not OK. I.e. if the test type is duplicates and the test found duplicates in the selected columns, or an exception occurred when executing the test.

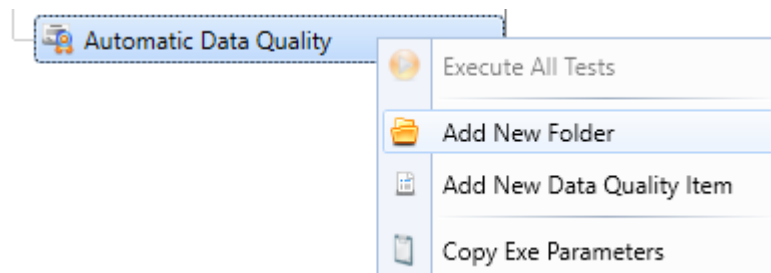


To schedule multiple items, see section Schedule multiple data quality tests inside a folder.

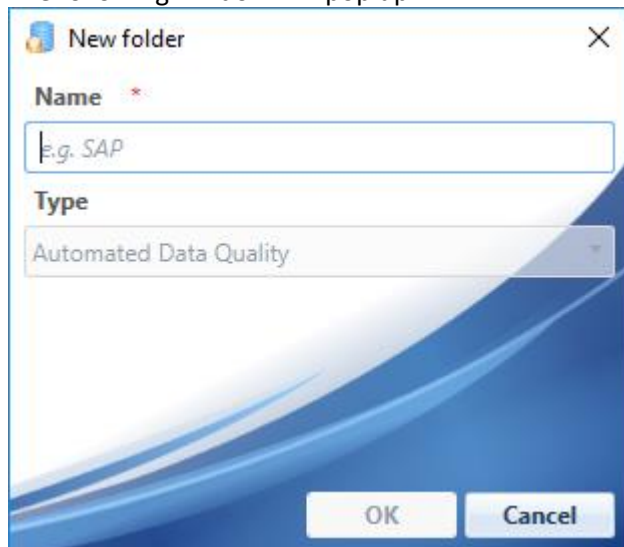
7.1.2. DataOps Folder

7.1.2.1. Add New Folder

To add a new folder right click Automated Data Quality, or a subfolder, and select “Add New Folder”.



The following window will pop up:

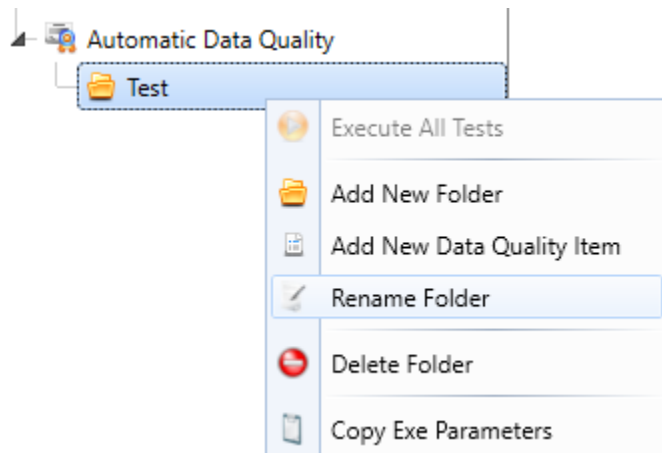


The **Name** must be different from the other folders inside the selected folder. The type “Automated Data Quality” since the folder is created inside Automated Data Quality.

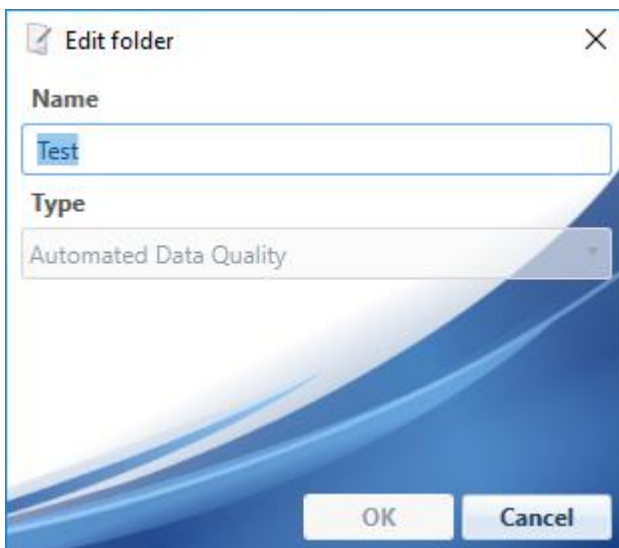
Press OK button to continue.

7.1.2.2. Rename Folder

Right click the folder you want to rename and select “Rename Folder”.



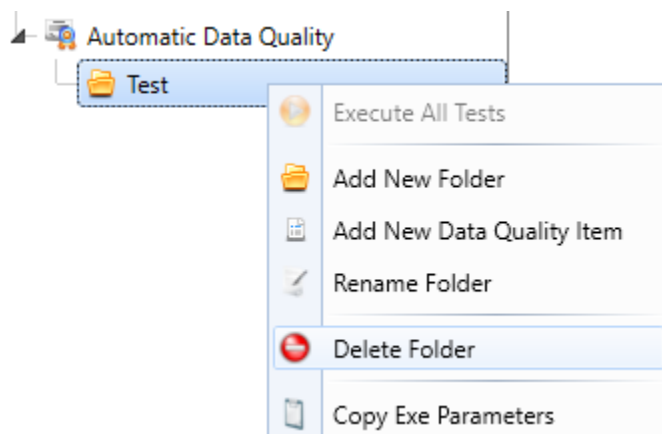
The following window will pop up:



Type in the new name and the OK button to continue.

7.1.2.3. Delete Folder

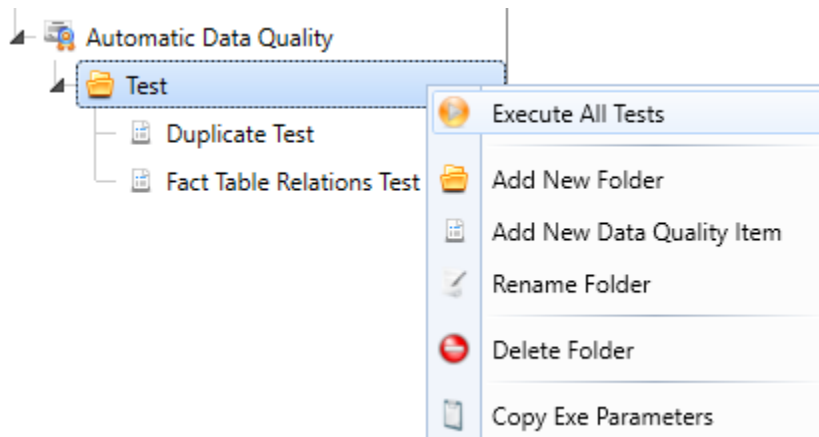
To delete a folder, right click the item and select “Delete Folder”.



Alternatively, select the desired folder and press the delete button on your keyboard.

7.1.2.4. Execute All Tests

To execute all the tests inside the folder in parallel, right click the folder and select “Execute All Tests”.



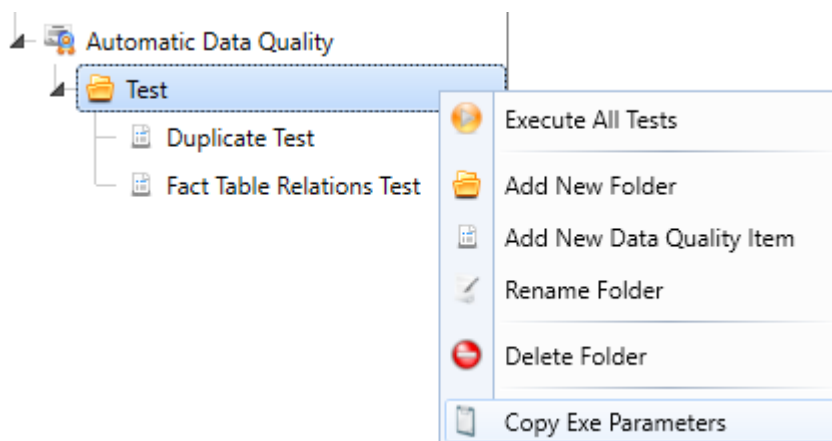
7.1.2.5. Schedule multiple data quality tests inside a folder

To schedule all the items inside a folder, right click the folder and select “Copy Exe Parameters”. Use the copied command line in i.e. SSIS to schedule the test. The command line contains the path to a batch file that will be run when executed. (Note: if the parameter ‘StopOnFoundIssue’ is set to false, the return code will always be 0, that is successful execution)

The batch file will return one of the following outputs:

0 – Tests are OK

1 – One or more tests are not OK. I.e. if one of the test types is duplicates and the test found duplicates in the selected columns, or an exception occurred when executing the test.



7.1.3. Test Types

7.1.3.1. Duplicates

The duplicate test searches for any duplicates of the selected column(s) from the selected object(s) inside the selected process group(s).

To choose a process group, click and select a group on the Process Groups textbox. You can also type in the name of the process group you want to select, and you will get suggestions for the name based on the text input.

Process Groups

Enter a process group...

All

Process Group A

Process Group B

Process Group C

You can select as many process groups as you like.

The objects inside the process group will be listed in the grid “Process Group Objects” below:

Process Groups

Enter a process group...

Process Group C ×

Process Group B ×

Process Group A ×

Process Group Objects

Filter...

☒ All ☐ Only included

| <input type="checkbox"/> | Object Name | Logic DB | Column Names | Failure Report | Status |
|--------------------------|--------------------|----------|------------------------|----------------|--------|
| <input type="checkbox"/> | FactEmployeeSalary | ELM | Enter a column name... | | |
| <input type="checkbox"/> | DimContact | ELM | Enter a column name... | | |
| <input type="checkbox"/> | DimEmployee | ELM | Enter a column name... | | |
| <input type="checkbox"/> | FactSalesOrder | ELM | Enter a column name... | | |
| <input type="checkbox"/> | DimAddress | ELM | Enter a column name... | | |
| <input type="checkbox"/> | DimCustomer | ELM | Enter a column name... | | |
| <input type="checkbox"/> | DimProduct | ELM | Enter a column name... | | |
| <input type="checkbox"/> | DimSalesPerson | ELM | Enter a column name... | | |
| <input type="checkbox"/> | DimSalesPersonSP | ELM | Enter a column name... | | |
| <input type="checkbox"/> | DimTime | ELM | Enter a column name... | | |

Click save from the ribbon bar to save the selected process groups.

For each object that you want to check for duplicates, enter one or more column names by clicking the “Column Name” cell for the corresponding row. The checkbox for the rows with new column names will automatically be checked, indicating that this row will be included in the test when executing and saving the item.

Click save from the ribbon bar to save the selected columns.

To execute the test, either schedule the test as described in section “Schedule a data quality item” or click “Execute” from the ribbon bar.

Process Groups

Enter a process group... Process Group C × Process Group B × Process Group A ×

Process Group Objects ☒ All ☐ Only included

| <input type="checkbox"/> | Object Name | Logic DB | Column Names | Failure Report | Status |
|-------------------------------------|--------------------|----------|---|----------------|--------|
| <input checked="" type="checkbox"/> | FactEmployeeSalary | ELM | Enter a column name... Employee_Id_ManagerID × | | |
| <input checked="" type="checkbox"/> | DimContact | ELM | Enter a column name... Contact_Id × | | |
| <input type="checkbox"/> | DimEmployee | ELM | Enter a column name... | | |
| <input type="checkbox"/> | FactSalesOrder | ELM | Enter a column name... | | |
| <input type="checkbox"/> | DimAddress | ELM | Enter a column name... | | |
| <input type="checkbox"/> | DimCustomer | ELM | Enter a column name... | | |
| <input type="checkbox"/> | DimProduct | ELM | Enter a column name... | | |
| <input type="checkbox"/> | DimSalesPerson | ELM | Enter a column name... | | |
| <input type="checkbox"/> | DimSalesPersonSP | ELM | Enter a column name... | | |
| <input type="checkbox"/> | DimTime | ELM | Enter a column name... | | |

Objects with duplicates are marked with a red status icon.

For objects without duplicates, the status icon is green.

For objects that has no columns or is not selected, the status icon is gray.

Blue icon indicates that the object is ready to be executed.

Click the button in the Failure Report column to view which rows that are duplicates.

ADW1_1_SalesOrderHeaderSalesReason

| SalesOrderHeaderSalesReason_Id | UID_LastChanged | UID_Instance_Id | SalesOrderHeader_Id | SalesReason_Id | ModifiedDate | SalesOrderID | SalesReasonID |
|--------------------------------|----------------------|-----------------|---------------------|----------------|----------------------|--------------|---------------|
| 0 | 8/16/2019 9:20:37 AM | 1 | 0 | 0 | | | |
| 1 | 8/16/2019 9:20:37 AM | 1 | 39 | 5 | 7/1/2001 12:00:00 AM | 43697 | 5 |
| 2 | 8/16/2019 9:20:37 AM | 1 | 39 | 9 | 7/1/2001 12:00:00 AM | 43697 | 9 |
| 3 | 8/16/2019 9:20:37 AM | 1 | 44 | 5 | 7/2/2001 12:00:00 AM | 43702 | 5 |
| 4 | 8/16/2019 9:20:37 AM | 1 | 44 | 9 | 7/2/2001 12:00:00 AM | 43702 | 9 |
| 5 | 8/16/2019 9:20:37 AM | 1 | 45 | 5 | 7/2/2001 12:00:00 AM | 43703 | 5 |
| 6 | 8/16/2019 9:20:37 AM | 1 | 45 | 9 | 7/2/2001 12:00:00 AM | 43703 | 9 |
| 7 | 8/16/2019 9:20:37 AM | 1 | 48 | 5 | 7/3/2001 12:00:00 AM | 43706 | 5 |
| 8 | 8/16/2019 9:20:37 AM | 1 | 48 | 9 | 7/3/2001 12:00:00 AM | 43706 | 9 |
| 9 | 8/16/2019 9:20:37 AM | 1 | 49 | 5 | 7/3/2001 12:00:00 AM | 43707 | 5 |
| 10 | 8/16/2019 9:20:37 AM | 1 | 49 | 9 | 7/3/2001 12:00:00 AM | 43707 | 9 |
| 11 | 8/16/2019 9:20:37 AM | 1 | 51 | 5 | 7/3/2001 12:00:00 AM | 43709 | 5 |
| 12 | 8/16/2019 9:20:37 AM | 1 | 51 | 9 | 7/3/2001 12:00:00 AM | 43709 | 9 |
| 13 | 8/16/2019 9:20:37 AM | 1 | 52 | 5 | 7/3/2001 12:00:00 AM | 43710 | 5 |

OK

7.1.3.2. Fact Table Relations

The Fact Table Relations test checks whether all referenced dimension ids exist in the corresponding dimension, for the selected fact table(s).

To choose a process group, click and select a group on the Process Groups textbox. You can also type in the name of the process group you want to select, and you will get suggestions for the name based on the text input.

Process Groups

All

Process Group A

Process Group B

Process Group C

You can select as many process groups as you like.

Process Groups

Process Group A × Process Group B × Process Group C ×

Process Group Fact Tables ☒ All ☐ Only included

| <input type="checkbox"/> | Fact Table Name | Dimensions |
|-------------------------------------|----------------------|---|
| <input checked="" type="checkbox"/> | + FactEmployeeSalary | DimContact DimCustomer DimEmployee |
| <input checked="" type="checkbox"/> | + FactSalesOrder | DimAddress DimContact DimCustomer DimProduct DimTime |

All the top-level fact tables contained inside the selected process groups will be listed in the “Process Group Fact Tables” grid. All the dimensions used by the fact tables are listed in the “Dimensions” row. Use the expand button to view more details regarding the fact table. This will bring up info of all the foreign keys inside the table and dependent dimensions and the dimensions columns.

| Process Group Fact Tables <input type="text" value="Filter..."/> <input checked="" type="radio"/> All <input type="radio"/> Only included | | | | | | |
|---|--------------------------|-------------------------|----------------------------|--------------------------|--|--------|
| <input type="checkbox"/> | Fact Table Name | Dimensions | | | | Status |
| <input checked="" type="checkbox"/> | + FactEmployeeSalary | DimContact | DimCustomer | DimEmployee | | |
| <input checked="" type="checkbox"/> | - FactSalesOrder | DimAddress | DimContact | DimCustomer | DimProduct DimTime | |
| <input checked="" type="checkbox"/> | Foreign Key | Dependent Dimension | Dependent Dimension Column | Show Results | Status | |
| <input checked="" type="checkbox"/> | Address_Id_BillToAddress | DimAddress | Address_Id | | | |
| <input checked="" type="checkbox"/> | Address_Id_ShipToAddress | DimAddress | Address_Id | | | |
| <input checked="" type="checkbox"/> | Contact_Id | DimContact | Contact_Id | | | |
| <input checked="" type="checkbox"/> | Customer_Id | DimCustomer | Customer_Id | | | |
| <input checked="" type="checkbox"/> | Product_Id | DimProduct | Product_Id | | | |
| <input checked="" type="checkbox"/> | Time_Id_Due | DimTime | Time_Id | | | |

All objects are checked by default, meaning that all the objects will be included when executing the test. Deselect columns that should not be checked.

Click save from the ribbon bar to save the selection.

To execute the test, either schedule the test as described in section “Schedule a data quality item” or click “Execute” from the ribbon bar.

Click the button in the Show Results column to view what data is missing from the column.

| Process Group Fact Tables | | | | | | <input type="text" value="Filter..."/> | <input checked="" type="radio"/> All <input type="radio"/> Only included | |
|-------------------------------------|--------------------------|--------------------|---------------------|----------------------------|--------------|--|--|--|
| <input type="checkbox"/> | | Fact Table Name | Dimensions | | | | Status | |
| <input checked="" type="checkbox"/> | - | FactEmployeeSalary | DimContact | DimCustomer | DimEmployee | | | |
| <input checked="" type="checkbox"/> | Foreign Key | | Dependent Dimension | Dependent Dimension Column | Show Results | Status | | |
| <input checked="" type="checkbox"/> | Contact_Id | | DimContact | Contact_Id | | | | |
| <input checked="" type="checkbox"/> | Customer_Id | | DimCustomer | Customer_Id | | | | |
| <input checked="" type="checkbox"/> | Employee_Id | | DimEmployee | Employee_Id | | | | |
| <input type="checkbox"/> | Employee_Id_ManagerID | | DimEmployee | Employee_Id | | | | |
| <input checked="" type="checkbox"/> | - | FactSalesOrder | DimAddress | DimContact | DimCustomer | DimProduct | DimTime | |
| <input checked="" type="checkbox"/> | Foreign Key | | Dependent Dimension | Dependent Dimension Column | Show Results | Status | | |
| <input checked="" type="checkbox"/> | Address_Id_BillToAddress | | DimAddress | Address_Id | | | | |
| <input checked="" type="checkbox"/> | Address_Id_ShipToAddress | | DimAddress | Address_Id | | | | |
| <input checked="" type="checkbox"/> | Contact_Id | | DimContact | Contact_Id | | | | |
| <input checked="" type="checkbox"/> | Customer_Id | | DimCustomer | Customer_Id | | | | |
| <input checked="" type="checkbox"/> | Product_Id | | DimProduct | Product_Id | | | | |
| <input checked="" type="checkbox"/> | Time_Id_Due | | DimTime | Time_Id | | | | |

Status Icons



Test ready. Object is ready to be executed.



Test running. Object is executing.



Test passed. Object without missing data.



Test failed. Object with missing data.

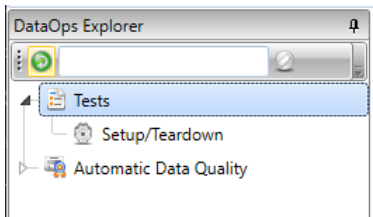


Test disabled. Objects that has no dependent dimension or is not selected.

7.2. Unit Test

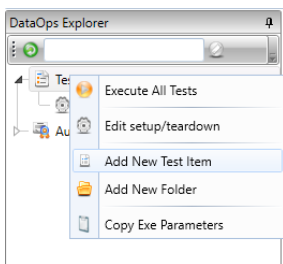
A unit test will execute user specified code and compare the result to an expected value. Before and after each test it is possible to execute a setup and a teardown.

Unit tests can be found under the 'Tests' folder in the DataOps Explorer.

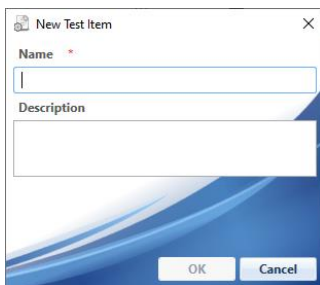


7.2.1. Add New Unit Test Item

To add a unit test, right click on either the top-level folder or any folder and select 'Add New Test Item'



Enter any name that does not already exist in the current folder. The description is optional.



7.2.2. Edit Unit Test Item

The Unit Test panel consist of these elements:

Test Description – Test description.

Comment – Test comment.

Disable test – Test will not run when executed outside of this panel.

Copy actual values to expected – If the test is executed successfully, the dataset from actual, is copied over to expected. Use hard coded values is automatically set to true.

Edit Setup/Teardown – The button opens the Setup/Teardown window for the test folder. The info text to the right of the button shows if any setup or teardown is in use.

Test Comparator – Comparator which is in use. Only single-row single-column is able to use any other comparator than '='. If the dataset has more than one row or more than one column, '=' must be used. When using any other comparator than '=' the value must be numerical or date/time.

Expected

Edit Connection – Sets the connection to use when running the script. This can be either to SQL server or Analysis Server.

Script – The script to run. Can be SQL, DAX or MDX. Type of script will depend on the target server.

Run Script – Executes the current script. **Note:** Setup and teardown will not be executed.

Use hard coded values – When checked, the script will not execute, but instead the data in the Result datatable is used for comparison.

Result – Shows the current expected dataset. When using hard coded values, right click on the header to add columns. Click the plus button in the top left corner to add new rows.

Actual

Edit Connection – Sets the connection to use when running the script. This can be either to SQL server or Analysis Server.

Script – The script to run. Can be SQL, DAX or MDX. Type of script will depend on the target server.

Run Script – Executes the current script. **Note:** Setup and teardown will not be executed.

Result – Shows the current actual dataset.

Copy to Expected – Copies the current actual dataset over to expected dataset.

My Unit Test

Test Description

Comment

Disable test: ☐

Copy actual values to expected when test succeeds: ☐

Edit Setup/Teardown In Use: Setup and Teardown

Test Comparator: =

Expected Actual

Script Edit Connection Run Script

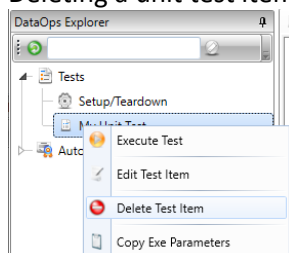
1

Result Use hard coded values: ☐

Copy to Expected

7.2.3. Delete Unit Test Item

Deleting a unit test item can be done by right clicking on the test and selecting Delete Test Item.

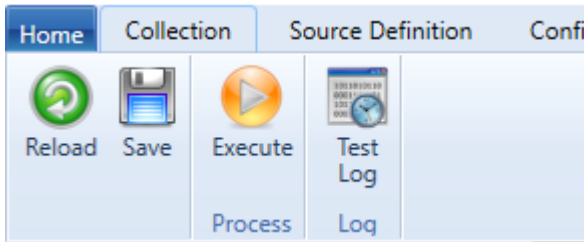


The test can also be deleted by selecting the test in the DataOps Explorer and clicking 'Delete' on the keyboard.

7.2.4. Execute Unit Test Item

The unit test can be executed in several different ways.

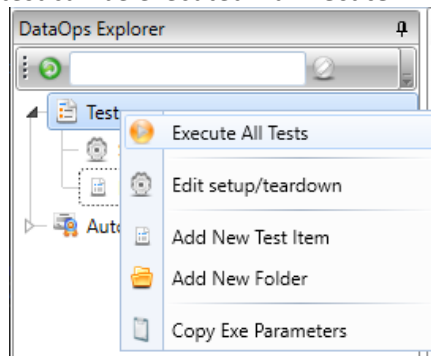
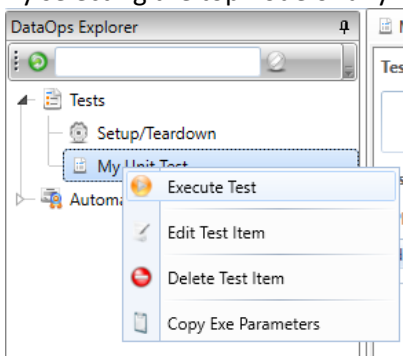
Click the **Execute** button in the ribbon bar to execute the current opened test.



Right click on the test in the DataOps explorer and select **Execute Test**.

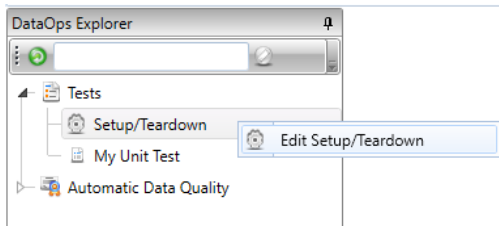
Or **Copy Exe parameters** and run it outside of Xpert BI, i.e. in SSIS. (Note: if the parameter 'StopOnFoundIssue' is set to false, the return code will always be 0, that is successful execution)

By selecting the top node or any folder, the test can be executed via **Execute All Tests**

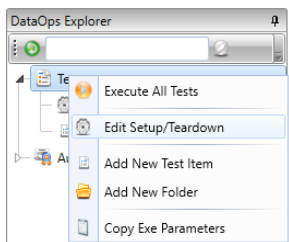


7.2.5. Setup/Teardown

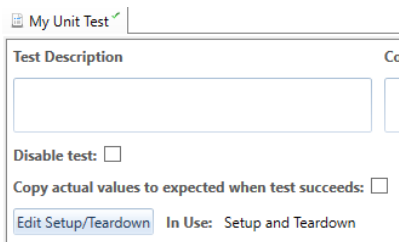
The setup/teardown are user specified code which will run before (Setup) and after (teardown) a test has been executed. Setup/teardown will only be executed for the tests in the same folder. It will not be executed for tests in any sub-folder nor any parent folder. The setup/teardown is limited to SQL code only.



The setup/teardown can be access directly under the folder it belongs by right-clicking and selecting 'Edit Setup/Teardown'.



The setup/teardown can be access by right-clicking a folder and selecting 'Edit Setup/Teardown'.



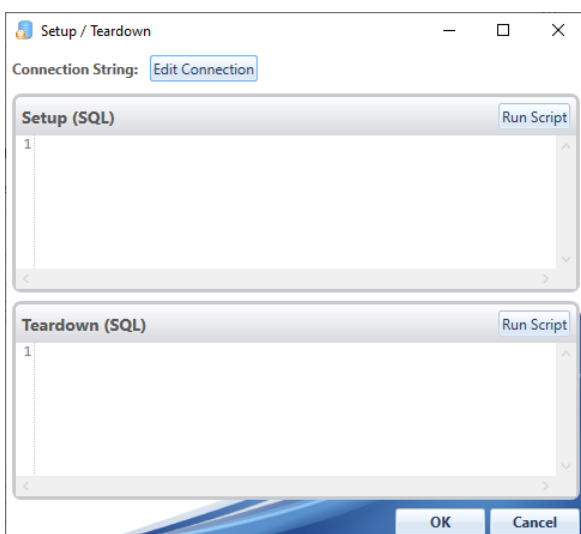
The setup/teardown can be access by clicking the 'Edit Setup/Teardown' in the edit panel of a unit test.

Connection String – Sets the connection to use when running the script. Only SQL server is available.

Setup– The SQL code for the setup part.

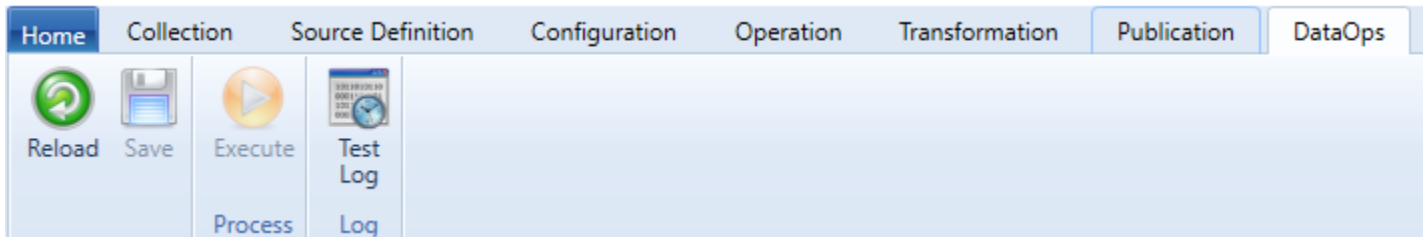
Teardown – The SQL code for the teardown part.

Run Script – Executes the script.



7.3. Test Log

The test log keeps track of every test that has been executed. To access the Test Log, navigate to DataOps -> Test Log from the ribbon bar.



Each log is kept in the database for x number of days which is user specified in the Options -> Maintenance tab as described in section “Options”.

| Drag a column header and drop it here to group by that column | | | | | | | | | |
|---|------------------------|-----------------|------------|----------------------|-----------------|--------|-----------|--|--|
| Date and Time | Type | Source | Test Name | Test Type | Completion Time | Status | Open Test | | |
| 27.04.2020 11:32 | Automated Data Quality | Xpert BI Client | FactTable | Fact Table Relations | 4.623s | | | | |
| 27.04.2020 11:32 | Automated Data Quality | Xpert BI Client | Duplicates | Duplicates | 0.32s | | | | |
| 22.04.2020 15:19 | Automated Data Quality | Xpert BI Client | FactTable | Fact Table Relations | 9.411s | | | | |
| 22.04.2020 15:18 | Automated Data Quality | Xpert BI Client | FactTable | Fact Table Relations | 9.165s | | | | |
| 22.04.2020 15:17 | Automated Data Quality | Xpert BI Client | FactTable | Fact Table Relations | 9.195s | | | | |
| 22.04.2020 15:17 | Automated Data Quality | Xpert BI Client | Duplicates | Duplicates | 5.15s | | | | |
| 22.04.2020 15:16 | Automated Data Quality | Xpert BI Client | FactTable | Fact Table Relations | 9.385s | | | | |

The test log grid consists of the following cells:

Date and Time – The date and time that the test was completed.

Type – Either “Automated Data Quality” or “Test”.

Source – Where the test was executed from. For Scheduled tests the batch file “DataOps Processor” is the source and for test executed directly in XpertBI, the source is “Xpert BI Client”.

Test Name – The name of the test.

Test Type – The type of test, i.e. “Duplicates”.

Completion Time – The time the test took to finish, shown in seconds.

Status – Status of the test shown as an icon. Green icon: Test is ok, red icon: test is not ok.

Open Test – Contains a button that opens the test.

When a row is selected, detailed information about the test is shown the “General” and “Summary” tab at the bottom of the pane.

| General | Summary |
|---------------------------------|-----------------------------|
| Type: Automated Data Quality | Logged: 27.04.2020 11:32:58 |
| Source: Xpert BI Client | Test Name: FactTable |
| Test Log ID: 3443 | Folder ID: 2 |
| Test Type: Fact Table Relations | |
| Status: Is_OK_With_Error | |

In the “General” tab you will find the following information:

Type – Either “Automated Data Quality” or “Test”.

Source – Where the test was executed from. For Scheduled tests the batch file “DataOps Processor” is the source and for test executed directly in XpertBI, the source is “Xpert BI Client”.

Test Log ID – The unique ID number of the test log.

Test Type – The type of test, i.e. “Duplicates”.

Status – Status as text.

Logged – The date and time that the test was logged.



Test Name – The name of the test.

Folder ID – The ID of the folder that the test is located in.

In the “Summary” tab there is a quick summary of the tests:

| General | Summary |
|---|---------|
| Test found 5 objects with duplicates errors in the following object(s): | |
| BI_KURS_ELM.dbo.DimSalesPerson | |
| BI_KURS_ELM.dbo.FactEmployeeSalary | |
| BI_KURS_ETL.dbo.SalesPerson | |
| BI_KURS_ODS.dbo.ADW_1_1_CurrencyRate | |
| BI_KURS_ODS.dbo.ADW_1_1_CreditCard | |

If the status is not ok (red status icon) the row can be expanded to view even more details about the objects that failed.

| Drag a column header and drop it here to group by that column | | | | |
|---|------------------|------------------------|---|------------|
| | Date and Time | Type | Source | Test Name |
| - | 27.04.2020 11:32 | Automated Data Quality | Xpert BI Client | FactTable |
| Identifier | | Completion Time | Status | |
| FactCollision.SalesPerson_Id | | 0.12s |  | |
| FactEmployeeSalary.Customer_Id | | 0.320s |  | |
| | 27.04.2020 11:32 | Automated Data Quality | Xpert BI Client | Duplicates |
| + | 22.04.2020 15:19 | Automated Data Quality | Xpert BI Client | FactTable |
| + | 22.04.2020 15:18 | Automated Data Quality | Xpert BI Client | FactTable |
| + | 22.04.2020 15:17 | Automated Data Quality | Xpert BI Client | FactTable |
| | 22.04.2020 15:17 | Automated Data Quality | Xpert BI Client | Duplicates |

This sub-grid contains an identifier, showing which of the objects that has failed, the completion time for executing this object and the status for the object.

8. Appendix I

8.1.1. Xpert BI Solution Catalog

The Xpert BI Solution Catalog is a web-based application that allows the user to view database information such as descriptions and SSAS (OLAP and Tabular) data in the browser (we recommend Google Chrome).

To allow the user more flexibility when writing documentation, it is possible to use certain “codewords” that conveys a certain meaning to the solution catalog. An example of this would be to add a hyperlink in a block of text. Below is the full list of codewords that can be used:

| Feature | Syntax | Example | Remarks |
|--------------------|--|--|-------------------------------------|
| Add hyperlink | <code>!createLink[(' <url>', '<friendly name>']</code> | <code>!createLink['http://www.bi-builders.com', 'Bi builders homepage']</code> | |
| Load data from URL | <code>!loadData['<url>']</code> | <code>!loadData['https://jsonplaceholder.typicode.com/comments']</code> | URL must point to a raw JSON source |

Currently, this feature is available for tabular and OLAP descriptions.