



# White Paper

## Export of Fabasoft Folio Objects to a Relational Database

Fabasoft Folio 2024 Update Rollup 1

Copyright © Fabasoft R&D GmbH, Linz, Austria, 2024.

All rights reserved. All hardware and software names used are registered trade names and/or registered trademarks of the respective manufacturers.

No rights to our software or our professional services, or results of our professional services, or other protected rights can be based on the handing over and presentation of these documents.

## Contents

<b>1 Introduction</b>	<b>4</b>
<b>2 Software Requirements</b>	<b>4</b>
<b>3 Prerequisites</b>	<b>4</b>
<b>4 Fabasoft Folio Database Export Example</b>	<b>5</b>
4.1 Export Definition .....	5
4.2 Full Export .....	8
4.3 Export (Changes Only).....	8

## 1 Introduction

The software components *Database Export (FSCDBEXP)* and *Database Export (Virtual Application) (FSCVDBEXP)* from the software product *Fabasoft Folio/Data* allow the export of Fabasoft Folio objects to a relational database.

There are two ways to carry out the export:

- Read objects from the standard Fabasoft Folio Kernel Interfaces  
A Fabasoft Folio query is generated and object data is fetched through standard attribute access. The values are written to the target database in chunks of 1000 objects at a time. The Fabasoft Folio security model is recognized and GET methods are used to compute properties.
- Direct access to Fabasoft Folio COO Service database  
Database export issues queries directly to the source database of the Fabasoft Folio COO Service to get the source data. Values are written in chunks of 10000 lines to the target database. The Fabasoft Folio security model is ignored and no methods are triggered to evaluate computed properties.

This document describes the export of Fabasoft Folio objects into a Microsoft SQL Server database.

## 2 Software Requirements

**System environment:** All information contained in this document implicitly assumes a Microsoft Windows environment.

**Supported platforms:** For detailed information on supported operating systems and software see the software product information on the Fabasoft distribution media.

**Descriptions in this document are based on the following software:**

- Microsoft SQL Server 2019 Enterprise (64 bit)

## 3 Prerequisites

Verify that the software components *Database Export (FSCDBEXP)* and *Database Export (Virtual Application) (FSCVDBEXP)* from the software product *Fabasoft Folio/Data* are installed in the Fabasoft Folio Domain.

In order to use the database export in a web browser environment the client user must be authenticated on the web server via Kerberos. The database export will not work using NTLM.

**Note:** You can verify the use of Kerberos turning on the "Security Audits" on the web server and then checking the logon audits in the "Security Events".

In the Microsoft Active Directory, select the *Account is trusted for delegation* option for the user in whose context the Fabasoft Folio Web Service is running.

Check the DNS settings via `NSLOOKUP`

- `<FullyQualifiedDomainName SQLServer>`
- `<IP SQLServer>`
- `<FullyQualifiedDomainName WebServer>`
- `<IP WebServer>`

Also configure a Service Principal Name in the format `HTTP/<FullyQualifiedDomainName of the web server>` for this user.

**Note:** After changing the service principal name you should restart the web server and login again on the client.

For the user, in whose context the Microsoft SQL Server is running, register a Service Principal Name using the `SetSPN` tool.

## 4 Fabasoft Folio Database Export Example

This example demonstrates how to export *Person* objects into a Microsoft SQL Server database. The Microsoft SQL Server database is located on the server where the database export is executed. The database name is "Fabasoft Folio Export".

### 4.1 Export Definition

To define a database export, an object of the *Database Export Definition* (`FSCDBEXP@1.1001:DatabaseExportDefinition`) object class has to be created in an *Administrative Tool*.

Right-click the *Database Export Definition* object and click "Properties".

In the properties of this object several settings can be defined:

- *General Settings*
  - The general settings for the database export define the structure of the database tables.
  - *Use Friendly Names*
    - Using this property it can be specified, that the language-specific names of the Fabasoft Folio properties are used instead of the references to name the single tables.
  - *Check Deleted Objects*
    - If this property is set to "Yes", an export via the "Standard Fabasoft Folio Kernel Interfaces" checks referenced objects. If a referenced object is a ghost this value will be exported as NULL.
    - **Note:** This feature can only be used when the *Export Type* is "Standard Fabasoft Folio Kernel Interfaces".
  - *Export Type*
    - The export type defines how Fabasoft Folio values are exported, via standard Fabasoft Folio Kernel Interfaces or via direct access to the Fabasoft Folio COO Service database.
  - *Object Pointer Export Type*
    - An object pointer can be mapped to the database as 64 bit integer, as low ID and high ID columns or as object address.
  - *Enumeration Export Type*
    - An enumeration property can be mapped to the database as integer value, as reference name or as language-specific name.
  - *Content Export Type*
    - A content property can be mapped to the database as content ID or as file path. When using "File" as *Content Export Type* a directory for exported contents must be defined.

- *Directory for Contents*  
This property is used to define the path to the target directory when using "File" as *Content Export Type*.
- *Additional Export Restrictions*  
In this property additional export restrictions can be specified using *Search Forms*.  
**Note:** If an additional export restriction is specified, it is recommend using the "Standard Fabasoft Folio Kernel Interface" as *Export Type* because the performance is much better.
- *Query Scope*  
In this property a query scope can be specified.
- *Prefix for Table Names*  
In this property a prefix for each created table can be specified.
- *Number of Objects per Transaction*  
In this property the maximum number of objects handled per transaction can be defined.
- *Username*  
In this property the database user can be defined.
- Password  
In this property the password of the database user can be defined.
- *Connection Settings to Target Database*  
This property defines which provider is used to connect to a relational database.

Connection Settings to Target Database				Show Details (3)
<input type="checkbox"/>	Parameter Type *	Parameter Value *	Software Component	
1	Provider	SQLOLEDB		🔍
2	Data Source	(local)		🔍
3	Catalog	ExportDB		🔍
Add Entry				

- *Parameter Type*  
In this property the type of the parameter is defined.
- *Parameter Value*  
The value of the parameter can be specified in this property.  
**Example:** To export the data in a CSV file, use "Data Source" as parameter type and "CSV" as parameter value. In the *Folder for CSVs* field (see below), you can specify a folder where the CSV will be saved.
- *Software Component*  
This property is used to define a local software component.
- *COO Services*  
Settings for the export of the single Fabasoft Folio COO Services are defined in this property.
  - *COO Service*  
The Fabasoft Folio COO Service object, which should be used for the export, is referenced in this property.
  - *Marked for Export*  
This property defines if the specified Fabasoft Folio COO Service should be considered for the export.
  - *Username*  
In this property the service user can be defined.

- Password  
In this property the password of the service user can be defined.
- *Connection Settings to COO Service Database*  
Specific connection parameters to the Fabasoft Folio COO Service database.
- *Last Export Started on/at*  
This property is used to store the date the last export started on/at.
- *Last Export Completed on/at*  
This property is used to store the date the last export completed on/at.
- *Object Classes to Export*  
In this list object classes are referenced. Objects of these object classes are exported.
  - *Object Class*  
The object class that should be exported can be specified in this property.
  - *Properties to export*  
This property contains the properties of the object class that should be exported.
  - *Include Derived Classes*  
Derived object classes can be included in the export.
  - *Include System Properties*  
This property is used to specify if system properties are exported too. Thus all properties inherited from the object class are included.
- *Table Export Definitions*  
This property contains a list of tables that should be exported (these can be taken from the system).
- *Folder for CSVs*  
You can specify a folder where the CSV will be saved, when you export the data. If no folder is specified a new one will be created, when you create a schema.
- *Logging Mode*  
The logging mode for the export can be defined in this property.
- *Folder for Logs*  
A folder where logs that are created are saved can be specified using this property.
- *Display Layout*  
A display layout for the logs can be defined in this property.

To export objects of the object class *Person* into a Microsoft SQL Server database, perform the following steps:

1. Edit the properties of the *Database Export Definition* object.  
Keep the suggested *General Settings*, define the *Connection Settings to Target Database* and the *COO Services* that should be included in the export. Specify *Person* in *Object Classes to Export*, leave *Table Export Definitions* empty and click "Next".
2. To create the tables that should be exported open the context menu of the *Database Export Definition* object and click "Create Table Export Definitions".
3. The single table export definitions can be controlled, added or deleted. Therefore right-click the *Database Export Definition* object and click "Properties".  
**Note:** If the object is opened in read mode, click "Edit".
4. Review the list *Table Export Definitions*, especially the *Column Export Definitions* column and remove those that should not be included in the export, then click "OK".

5. To create the tables in the target database right-click the *Database Export Definition* object and click "Create Schema". The tables in the target database are created, but no data is exported.

**Note:** After a system upgrade existing schemata and export definitions have to be reviewed and adjusted to changes of object classes that are used for the export.

## 4.2 Full Export

An export is performed using the command "Full Export". All objects of the defined object classes are mapped to the defined target database. To execute a full export, perform the following steps:

1. Open the context menu of the *Database Export Definition* object and click "Full Export".
2. When the export is finished, click "Next".  
All objects of the defined object classes located in the defined Fabasoft Folio COO Services are exported to the Microsoft SQL Server database.

**Note:** The export start time and the export end time are written to the *Database Export Definition* object. To review those times, right-click the *Database Export Definition* object and click "Properties".

## 4.3 Export (Changes Only)

Besides executing a full export also only changes can be exported. This export mode allows the export of objects that were changed or created after the last export. To perform an export in this export mode, perform the following steps:

1. Open the context menu of the *Database Export Definition* object and click "Export (changes only)".
2. Click "Next".  
All objects of the defined object classes located in the defined Fabasoft Folio COO Services that were changed or created after the last export were exported to the Microsoft SQL Server database.