
Article

[Yuri Marx](#) · Mar 7, 2022 4m read

Data migration tool - Part III: from DB2 to IRIS

This is the third part of the series of articles on migrating from the main databases on the market to InterSystems IRIS. In this part, the procedures for migrating from DB2 will be detailed. As described in previous articles, there are currently a few options to do the migration. However, the two most popular options include the usage of DBeaver (<https://openexchange.intersystems.com/package/DBeaver>) or SQLGateway. The first one will be demonstrated in this article, and the second one is presented in an excellent article by Robert Cemper, DB Migration using SQLgateway (<https://community.intersystems.com/post/db-migration-using-sqlgateway>).

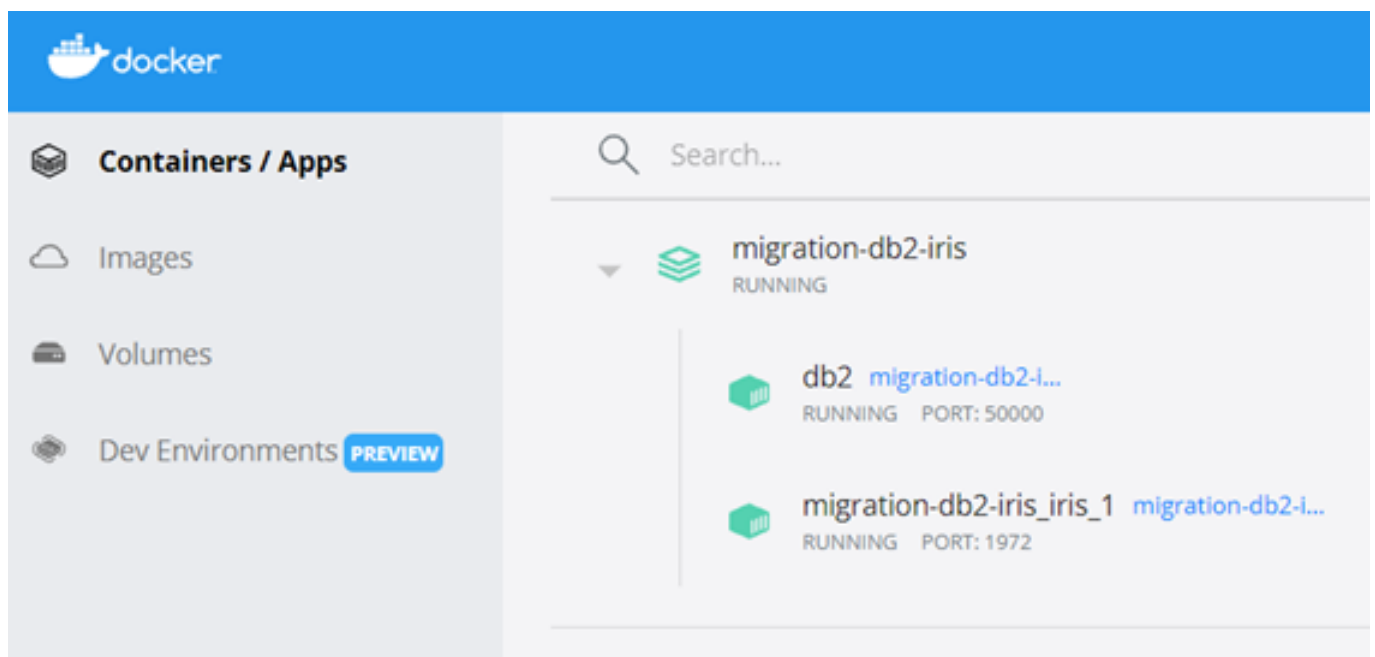
Get the sample data for the migration process

In Github it is possible to download a docker-compose project to build and run 2 databases:

- Source Database: DB2 database Docker instance with a sample database.
- Target Database: InterSystems IRIS data platform Docker instance with a ready schema to receive the source database.

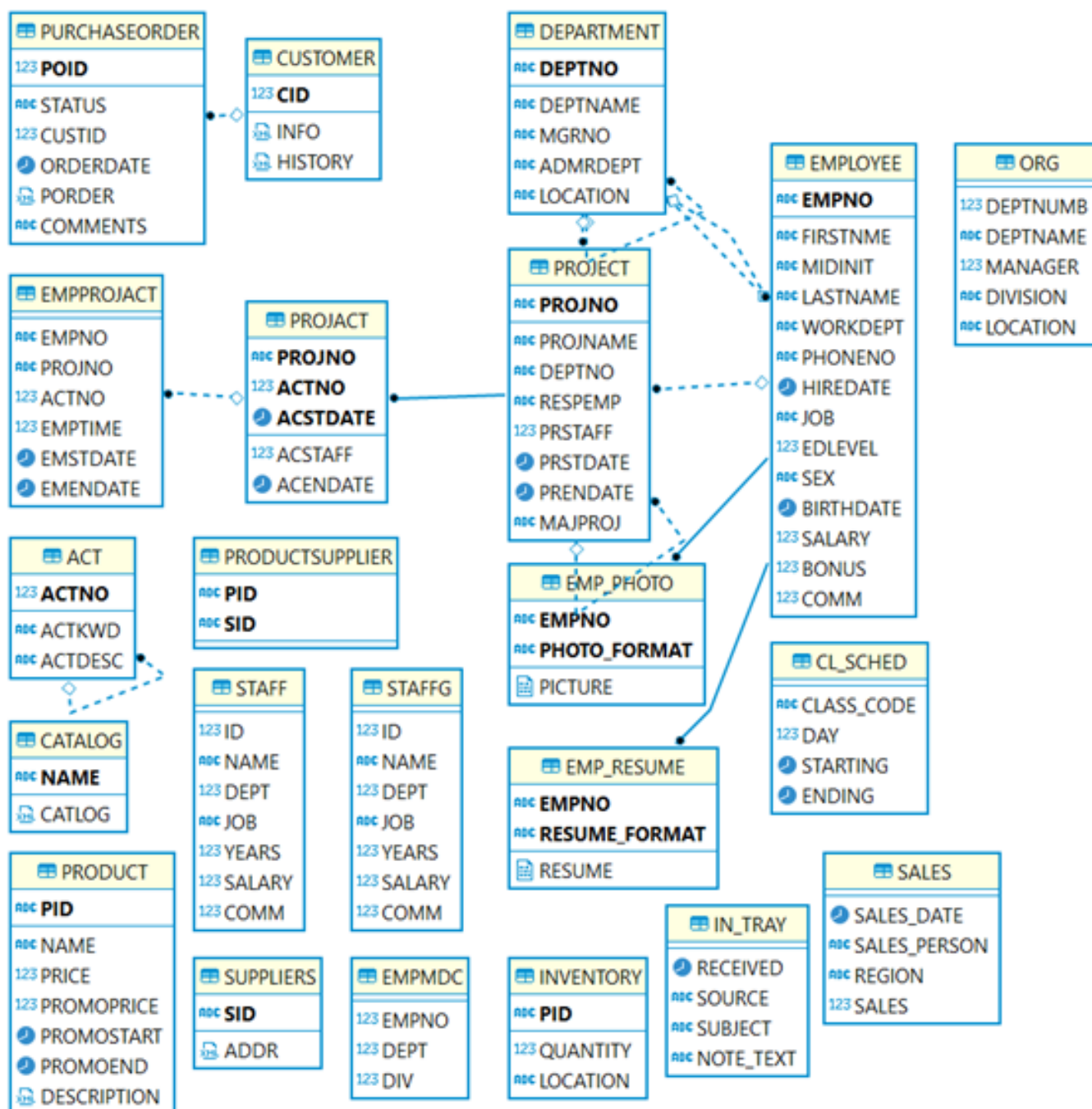
To get the sample and run it, follow these steps:

1. Go to the git repository: <https://github.com/yurimarx/migration-db2-iris>.
2. Clone the project: git clone <https://github.com/yurimarx/migration-db2-iris.git>.
3. Go to the project folder migration-db2-iris.
4. Do the build: docker-compose build.
5. Execute the containers: docker-compose up -d.
6. Check in your docker desktop with the instances if everything is ok:



About the data to be migrated

In the first two parts, we worked with a sales database. However, the DB2, when installed, goes with a sample database (it is a more complete sales database), and we will use it in this article. The data to be migrated is presented here:



So, the migration process from DB2 to IRIS will include 22 tables.

The migration destination will be `dctest` schema inside `USER` namespace in the InterSystems IRIS database.

Open-source tool to migrate from DB2 to IRIS: DBeaver

DBeaver is a database tool to connect, create, drop, select, update and delete data objects when work with the main database products in the market. Download it from:

<https://openexchange.intersystems.com/package/DBeaver>. Now follow the installation instructions to get this fantastic product into your laptop or desktop.

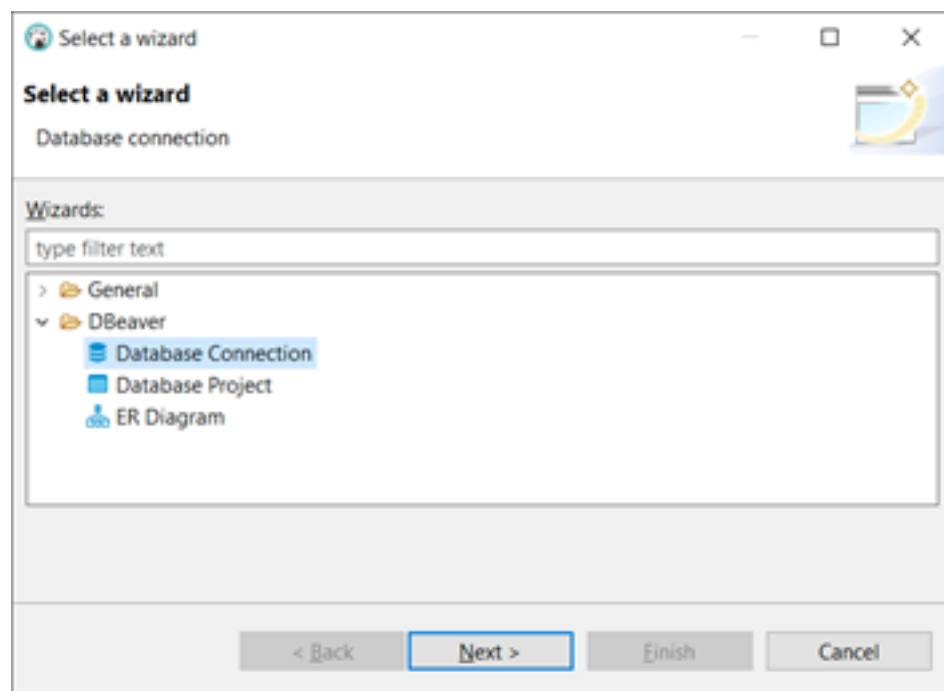
DBeaver can be used to migrate data between database connections, even if they come from different manufacturers and versions.

Connecting the Source and Target Databases using DBeaver

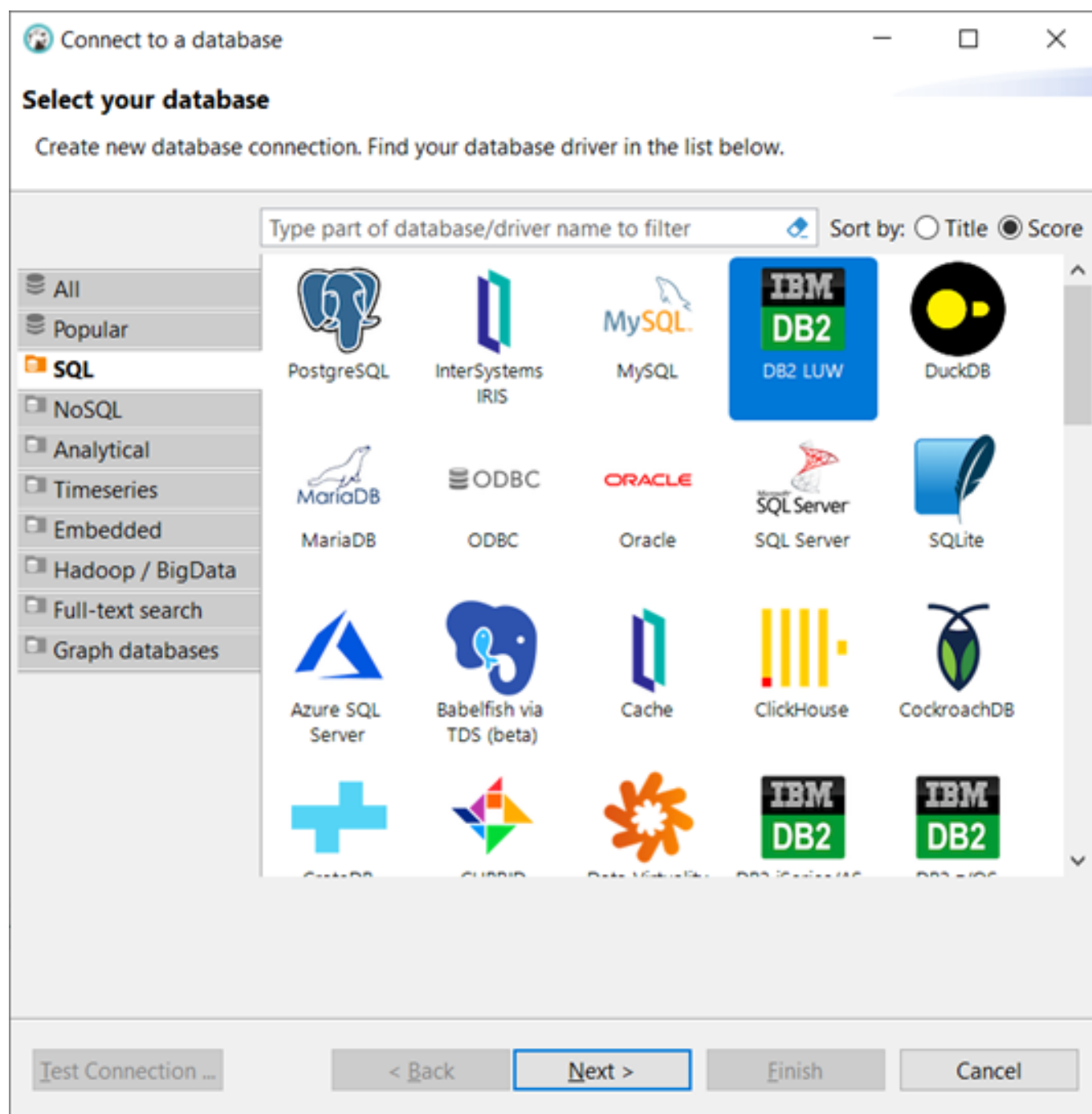
Now we will set the database connections to be migrated.

To set DB2 connection to DBeaver:

1. Before connecting DB2 for the first time, wait for 5 to 10 minutes. This is the time needed for the DB2 script to construct the sample database after the docker instance creation.
2. In DBeaver Go to File > New.
3. Select Database Connection and click Next:

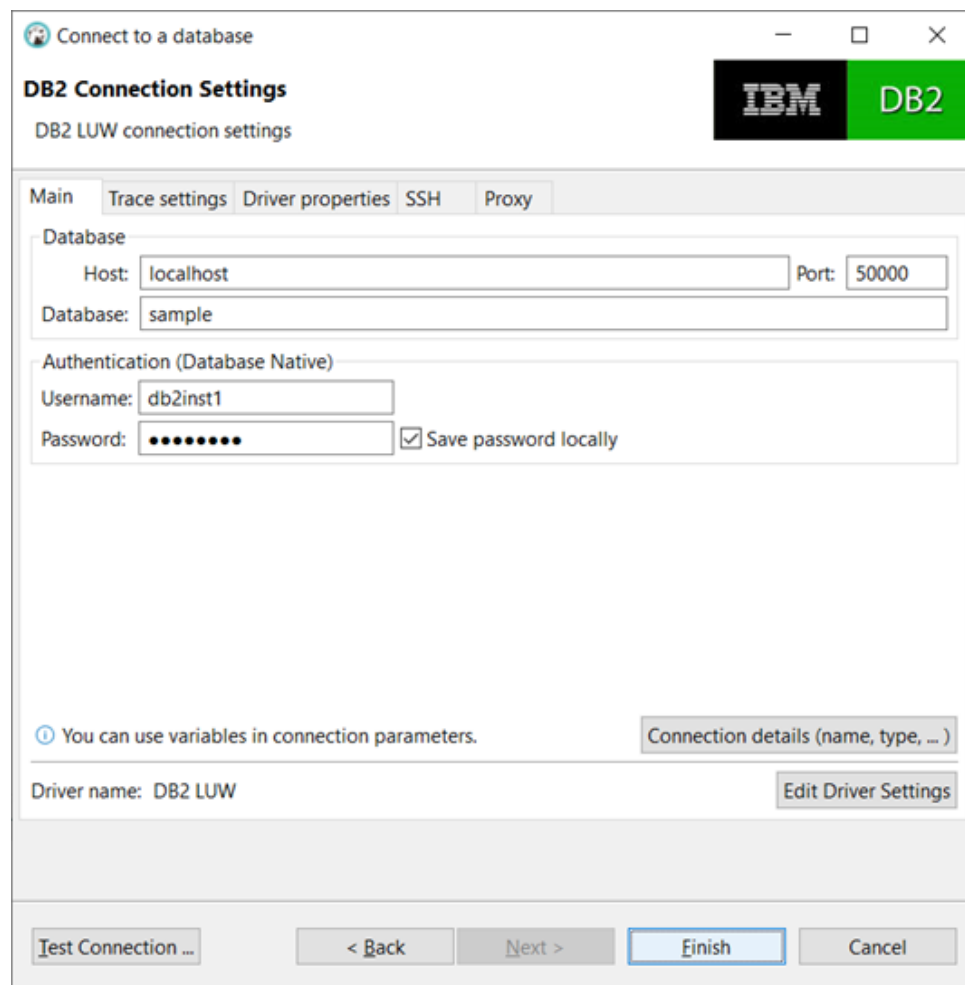


4. Choose SQL tab > DB2 LUW and click next:



5. Fill the DB2 connection fields as shown in this picture:

Host: localhost
Port: 50000
Database: sample
Username: db2inst1
Password: password
Click Finish.



The image shows the 'Connect to a database' dialog box with the 'DB2 Connection Settings' tab selected. The dialog has a title bar with a close button. The main area is divided into sections: 'Database' with fields for 'Host' (localhost) and 'Port' (50000), and 'Database' (sample). Below this is the 'Authentication (Database Native)' section with fields for 'Username' (db2inst1) and 'Password' (masked with dots), and a checkbox for 'Save password locally' which is checked. At the bottom, there is a 'Test Connection ...' button, a '< Back' button, a 'Next >' button, a 'Finish' button, and a 'Cancel' button. A 'Connection details (name, type, ...)' button is also present. The driver name is listed as 'DB2 LUW'.

Connect to a database

DB2 Connection Settings

DB2 LUW connection settings

Main | Trace settings | Driver properties | SSH | Proxy

Database

Host: localhost Port: 50000

Database: sample

Authentication (Database Native)

Username: db2inst1

Password: ☒ Save password locally

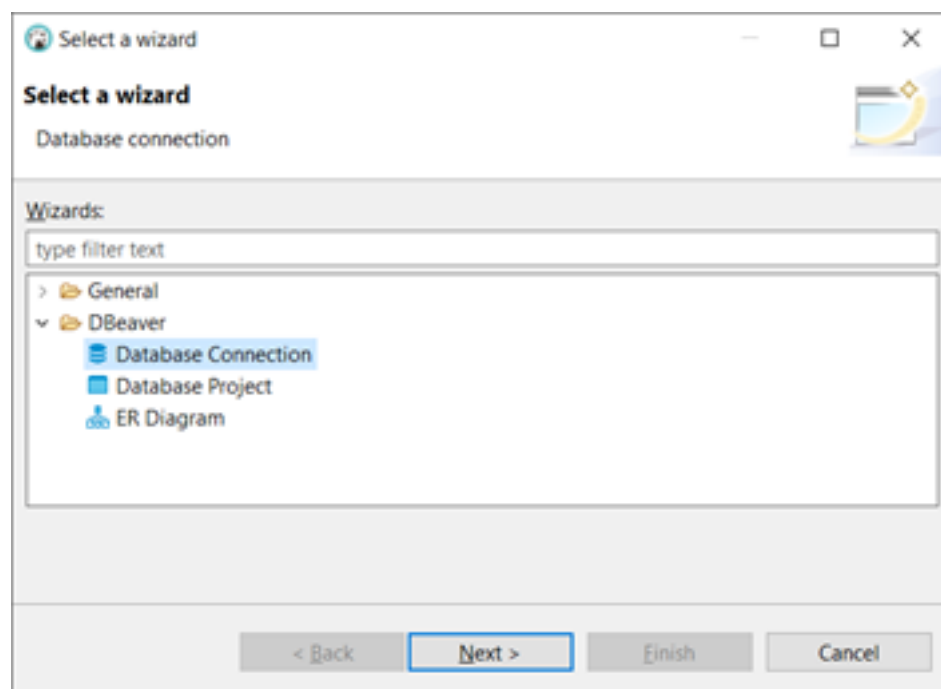
i You can use variables in connection parameters. Connection details (name, type, ...)

Driver name: DB2 LUW Edit Driver Settings

Test Connection ... < Back Next > Finish Cancel

To set InterSystems IRIS connection to DBeaver:

1. In DBeaver Go to File > New.
2. Select Database Connection and click Next:



The image shows the 'Select a wizard' dialog box. The title bar has a close button. The main area is titled 'Select a wizard' and 'Database connection'. Below this is a 'Wizards:' section with a search bar labeled 'type filter text'. A tree view shows the following structure: 'General' (expanded), 'DBeaver' (expanded), 'Database Connection' (selected), 'Database Project', and 'ER Diagram'. At the bottom, there are buttons for '< Back', 'Next >', 'Finish', and 'Cancel'.

Select a wizard

Database connection

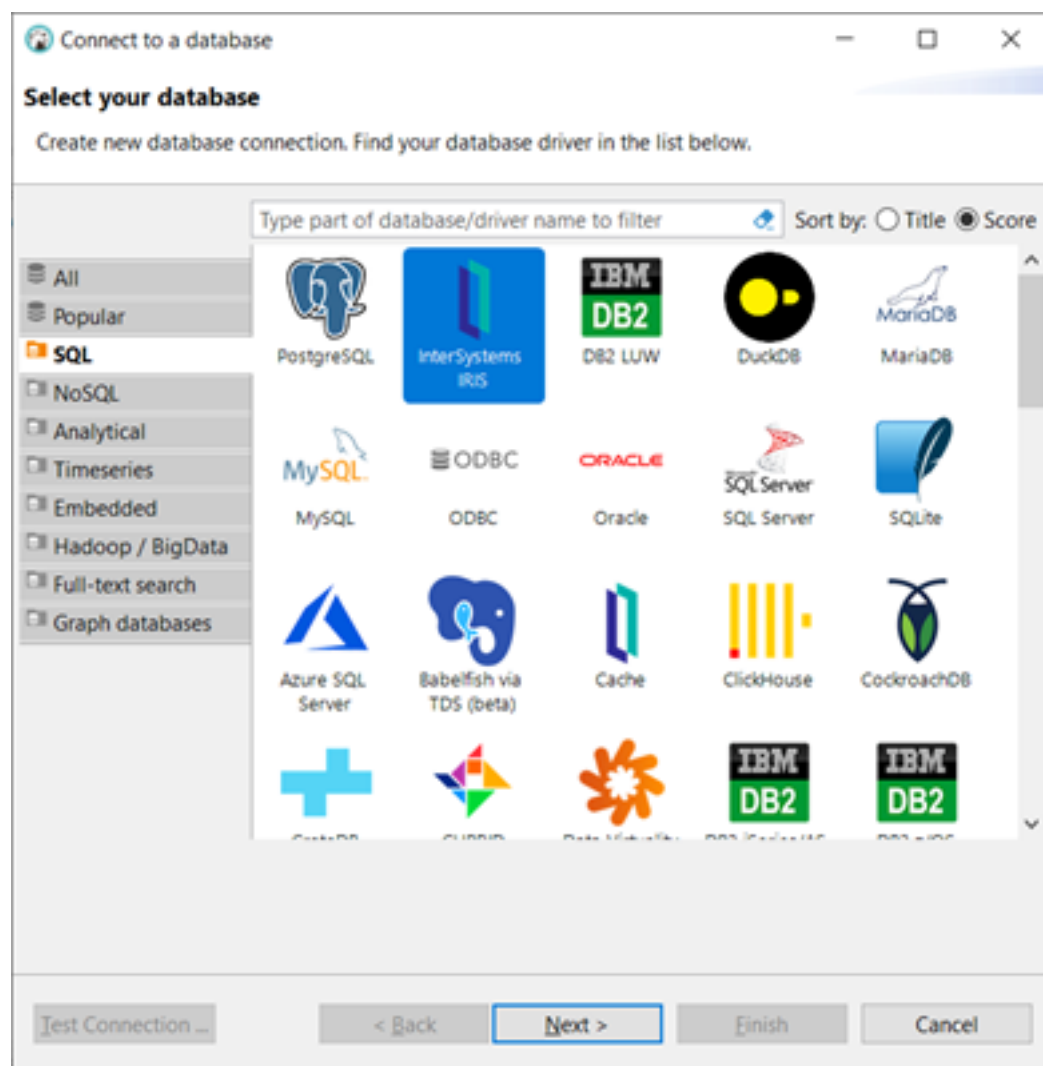
Wizards:

type filter text

- > General
- ▼ DBeaver
 - Database Connection
 - Database Project
 - ER Diagram

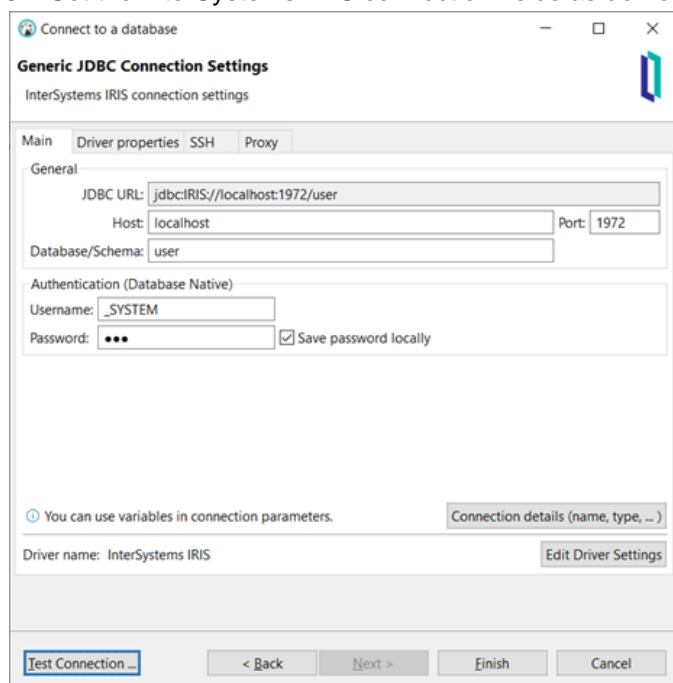
< Back Next > Finish Cancel

3. Choose SQL tab > InterSystems IRIS and click next:



4. If DBeaver requests to download the InterSystems IRIS driver, press Yes or Ok.

5. Set the InterSystems IRIS connection fields as demonstrated in this picture:



Host: localhost

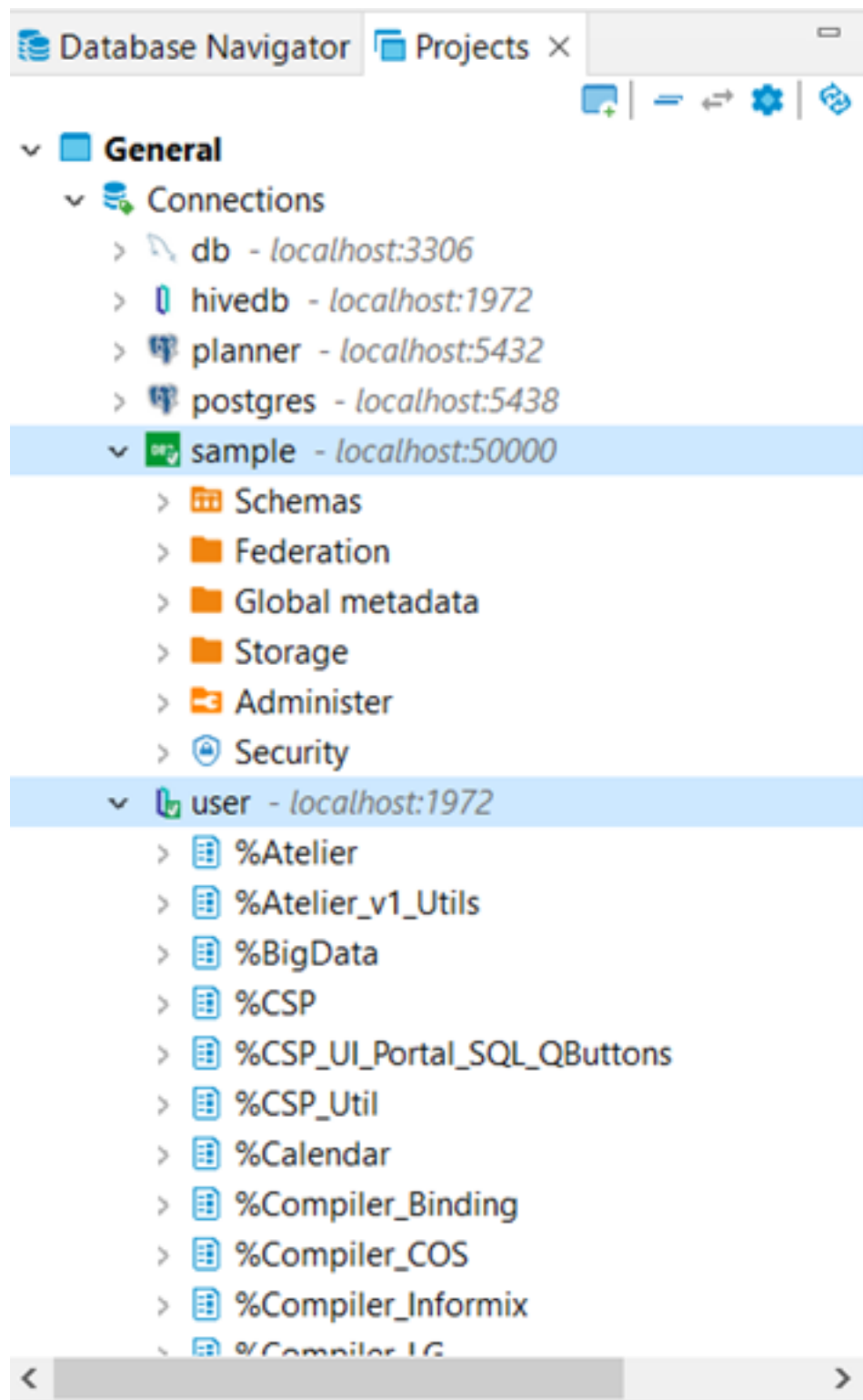
Database/Schema: user

Username: SYSTEM

Password: SYS

Click Test Connection and Finish

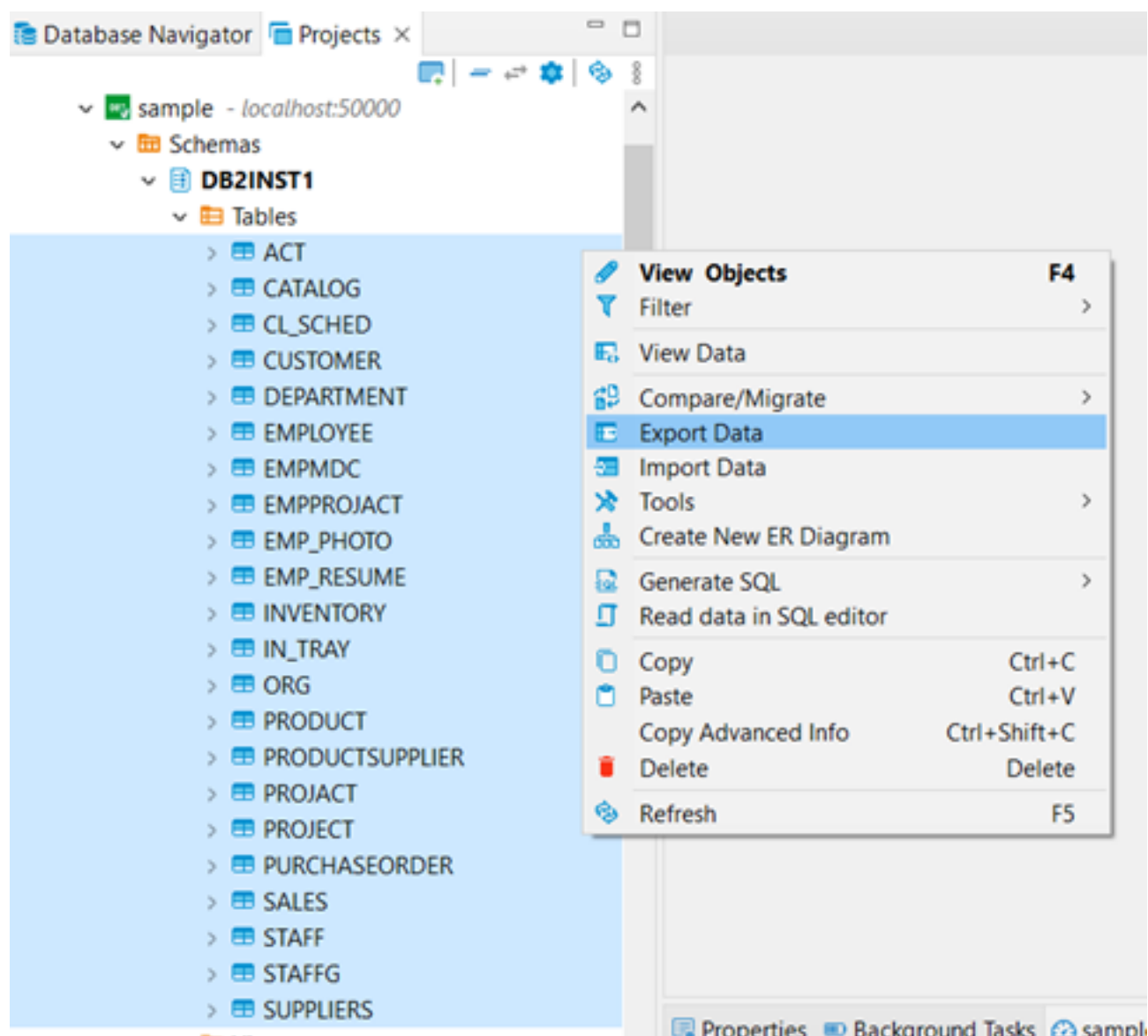
The connections (sample and user) are available in the Database Navigator:



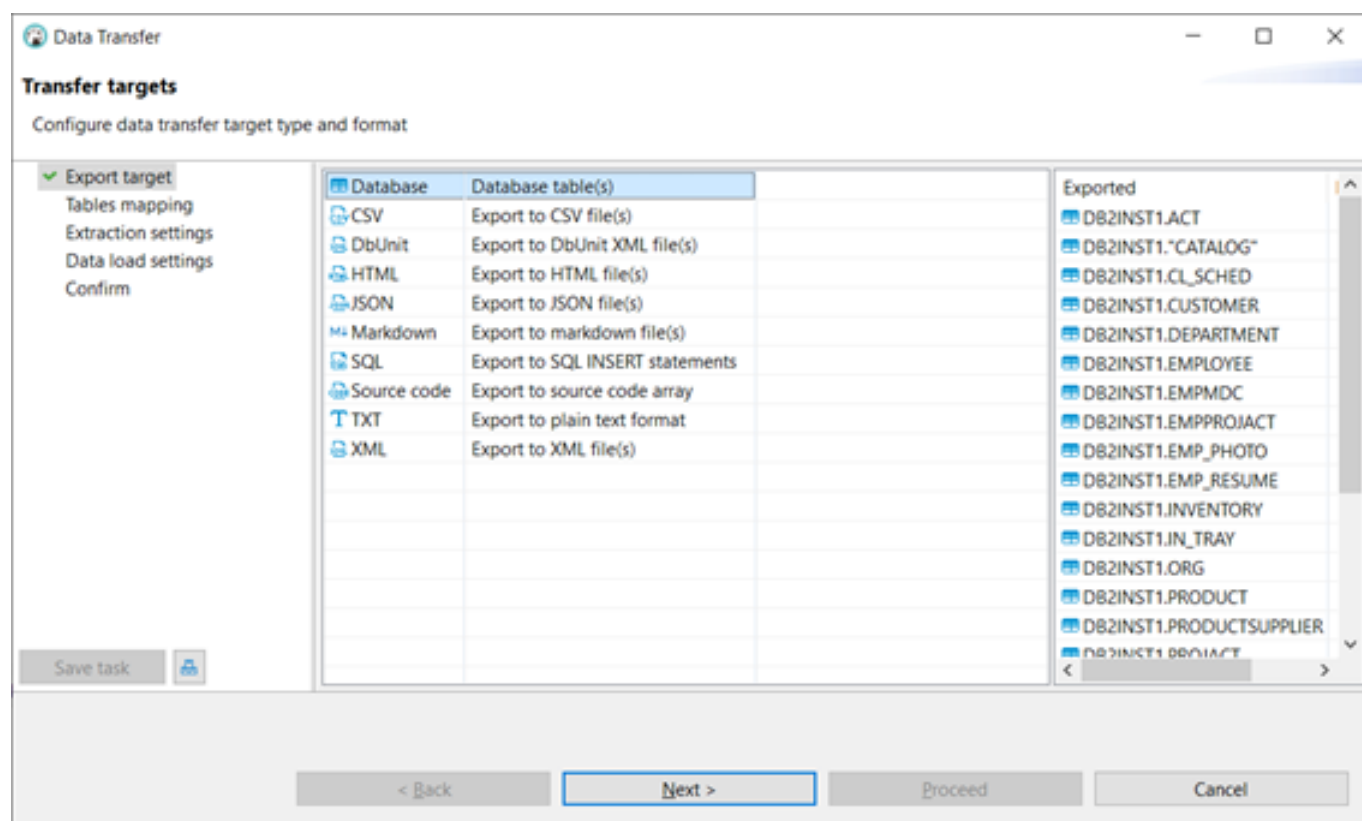
Do the migration

To do the migration, follow these steps:

1. Expand the sample connection (DB2 connection) > public and select all tables. Click on the selected tables with the right mouse button and choose Export Data, as displayed in this picture:



2. Select Database, as indicated in this picture and click Next:



3. Click the Choose button:

Data Transfer

Tables mapping

Map tables and columns transfer

☒ Export target
☒ **Tables mapping**
 Extraction settings
 Data load settings
 Confirm

Target container: **dc_test [user]** Choose ...

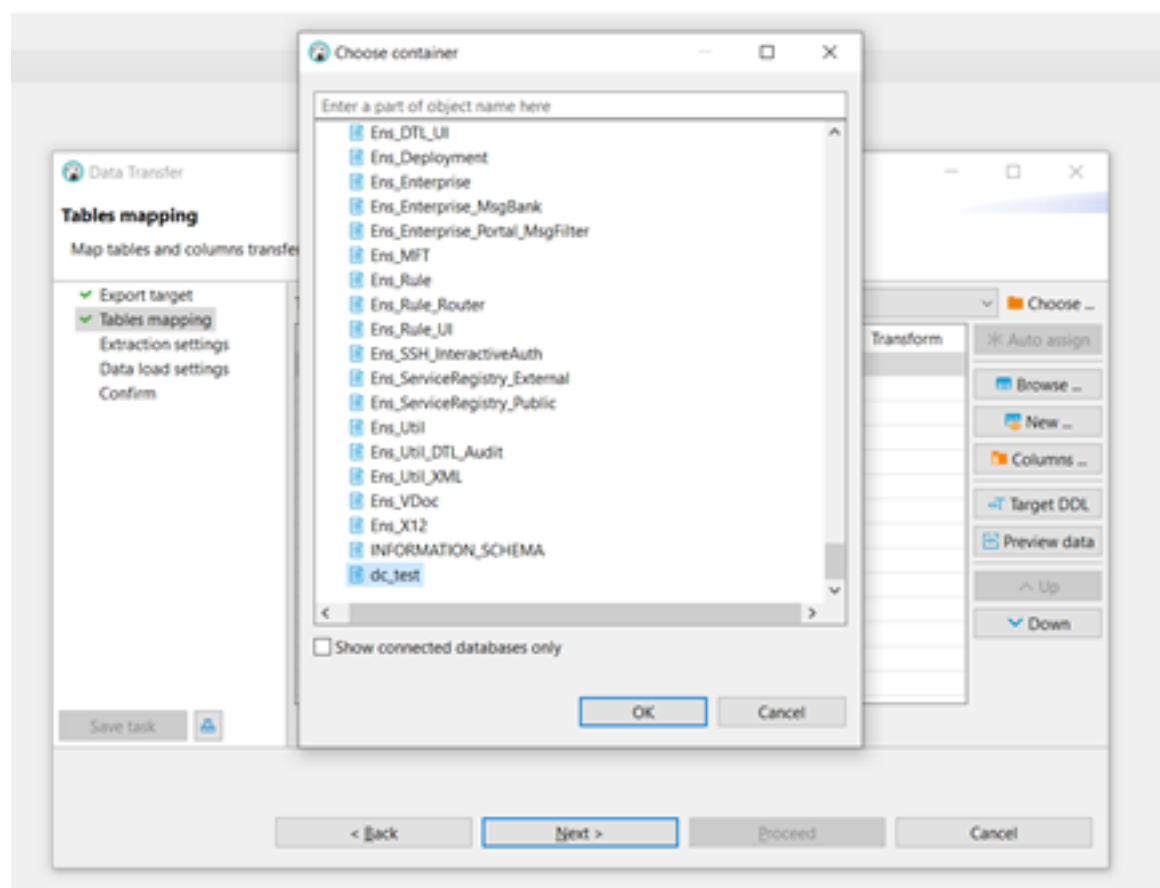
Source	Target	Mapping	Transform
> DB2INST1."CATALOG"	CATALOG	create	
> DB2INST1.CL_SCHED	CL_SCHED	create	
> DB2INST1.CUSTOMER	CUSTOMER	create	
> DB2INST1.EMPMDR	EMPMDR	create	
> DB2INST1.INVENTORY	INVENTORY	create	
> DB2INST1.IN_TRAY	IN_TRAY	create	
> DB2INST1.ORG	ORG	create	
> DB2INST1.PRODUCT	PRODUCT	create	
> DB2INST1.PRODUCTSUPPLIER	PRODUCTSUPPLIER	create	
> DB2INST1.SALES	SALES	create	
> DB2INST1.STAFF	STAFF	create	
> DB2INST1.STAFFG	STAFFG	create	
> DB2INST1.SUPPLIERS	SUPPLIERS	create	
> DB2INST1.ACT	ACT	create	
> DB2INST1.PURCHASEORDER	PURCHASEORDER	create	
> DB2INST1.DEPARTMENT	DEPARTMENT	create	
> DB2INST1.EMPLOYEE	EMPLOYEE	create	
> DB2INST1.EMPPROJECT	EMPPROJECT	create	
> DB2INST1.EMP_PHOTO	EMP_PHOTO	create	
> DB2INST1.EMP_RESUME	EMP_RESUME	create	
> DB2INST1.PROJECT	PROJECT	create	
> DB2INST1.PROJECT	PROJECT	create	

Save task Auto assign Browse ... New ... Columns ... Target DDL Preview data Up Down

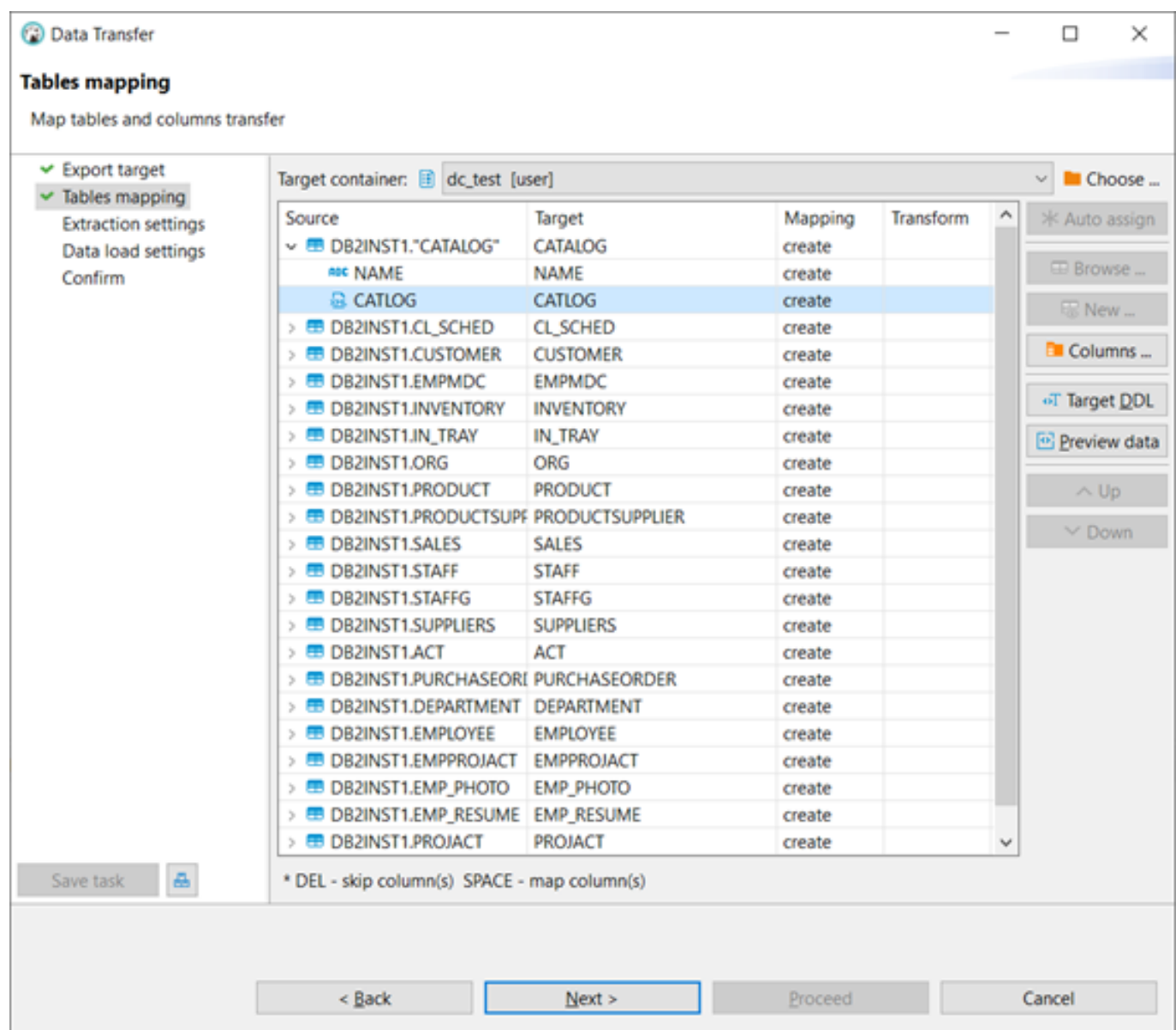
* DEL - skip column(s) SPACE - map column(s)

< Back Next > Proceed Cancel

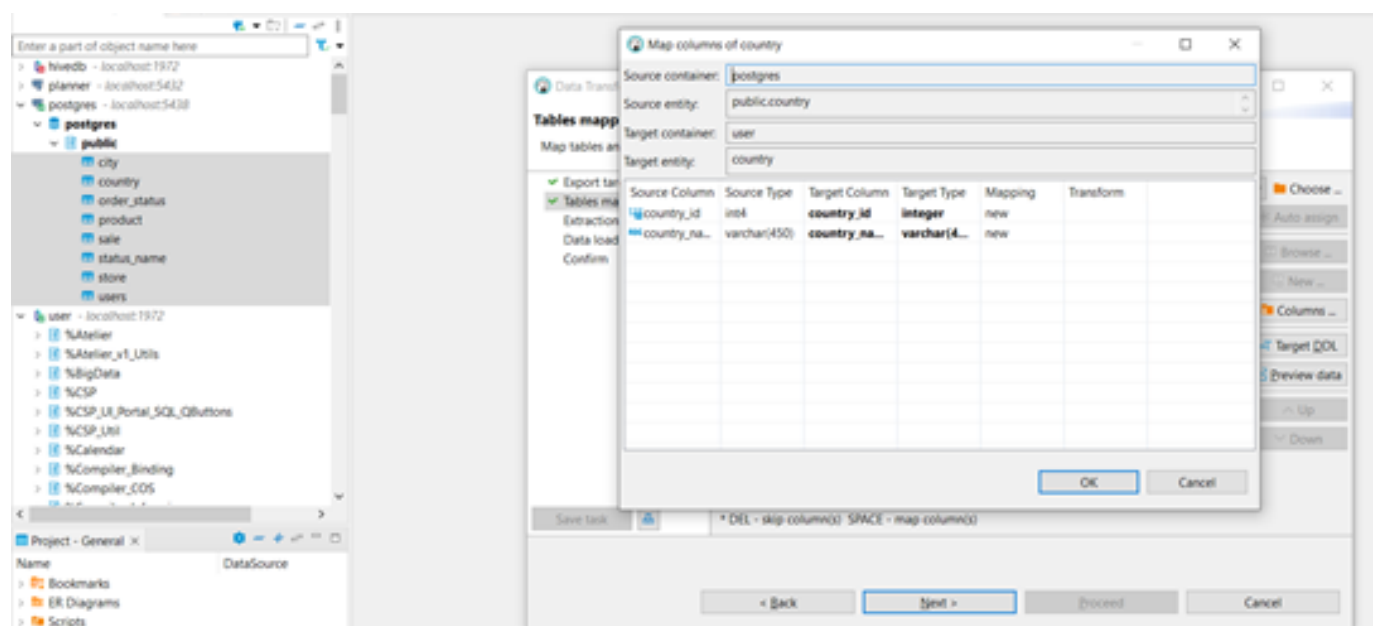
4. Select **dc_test** and click **Ok**.



5. Now it is necessary to change some data type configurations for the target database because the IRIS and DB2 use different data types to store XML values.
6. Expand DB2INST1.CATALOG table, select the CATALOG field (it is an XML field type) and click Columns...



7. Change the Target Type from LONGVARIABLE to VARCHAR(10000) and click Ok.



8. Repeat the process with the tables
 - a. CUSTOMER, fields INFO and HISTORY.
 - b. PRODUCT, field DESCRIPTION.
 - c. SUPPLIERS, field ADDR.
 - d. PURCHASEORDER, field PORDER.
9. Now with the Target Data Types changed, click Next.

Data Transfer

Tables mapping

Map tables and columns transfer

✓ Export target
✓ Tables mapping
Extraction settings
Data load settings
Confirm

Target container: **dc_test [user]** Choose ...

Source	Target	Mapping	Transform
DB2INST1."CATALOG"	CATALOG	create	
NAME	NAME	create	
CATLOG	CATLOG	create	
DB2INST1.CL_SCHED	CL_SCHED	create	
DB2INST1.CUSTOMER	CUSTOMER	create	
DB2INST1.EMPMDC	EMPMDC	create	
DB2INST1.INVENTORY	INVENTORY	create	
DB2INST1.IN_TRAY	IN_TRAY	create	
DB2INST1.ORG	ORG	create	
DB2INST1.PRODUCT	PRODUCT	create	
DB2INST1.PRODUCTSUPPLIER	PRODUCTSUPPLIER	create	
DB2INST1.SALES	SALES	create	
DB2INST1.STAFF	STAFF	create	
DB2INST1.STAFFG	STAFFG	create	
DB2INST1.SUPPLIERS	SUPPLIERS	create	
DB2INST1.ACT	ACT	create	
DB2INST1.PURCHASEORDER	PURCHASEORDER	create	
DB2INST1.DEPARTMENT	DEPARTMENT	create	
DB2INST1.EMPLOYEE	EMPLOYEE	create	
DB2INST1.EMPPROJECT	EMPPROJECT	create	
DB2INST1.EMP_PHOTO	EMP_PHOTO	create	
DB2INST1.EMP_RESUME	EMP_RESUME	create	
DB2INST1.PROJECT	PROJECT	create	

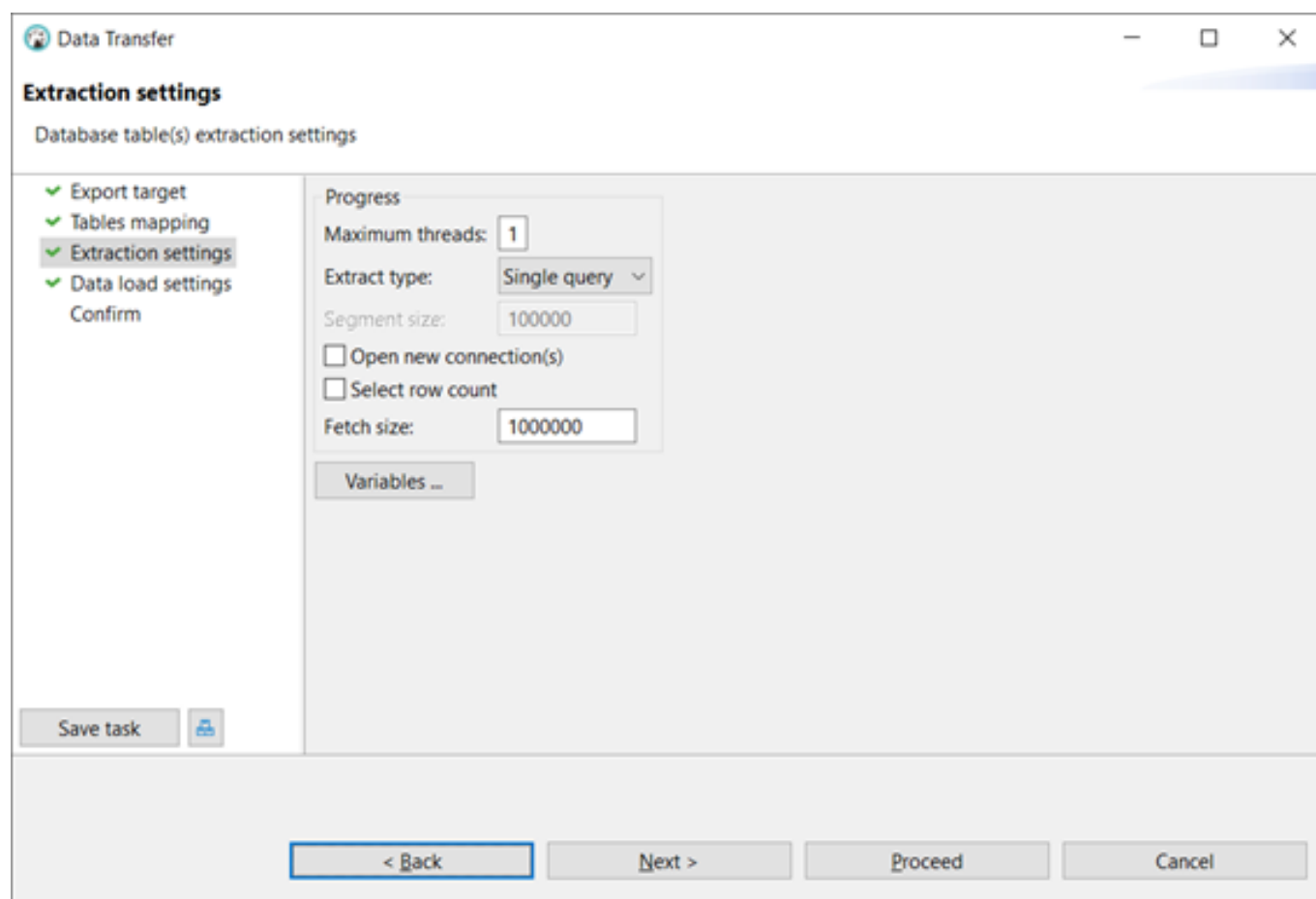
* DEL - skip column(s) SPACE - map column(s)

Save task

< Back Next > Proceed Cancel

Auto assign
Browse ...
New ...
Columns ...
Target DDL
Preview data
Up
Down

10. Set Fetch size to 1000000 and click Next.



Data Transfer

Extraction settings

Database table(s) extraction settings

- ✓ Export target
- ✓ Tables mapping
- ✓ **Extraction settings**
- ✓ Data load settings
- Confirm

Progress

Maximum threads:


Extract type:

Segment size:

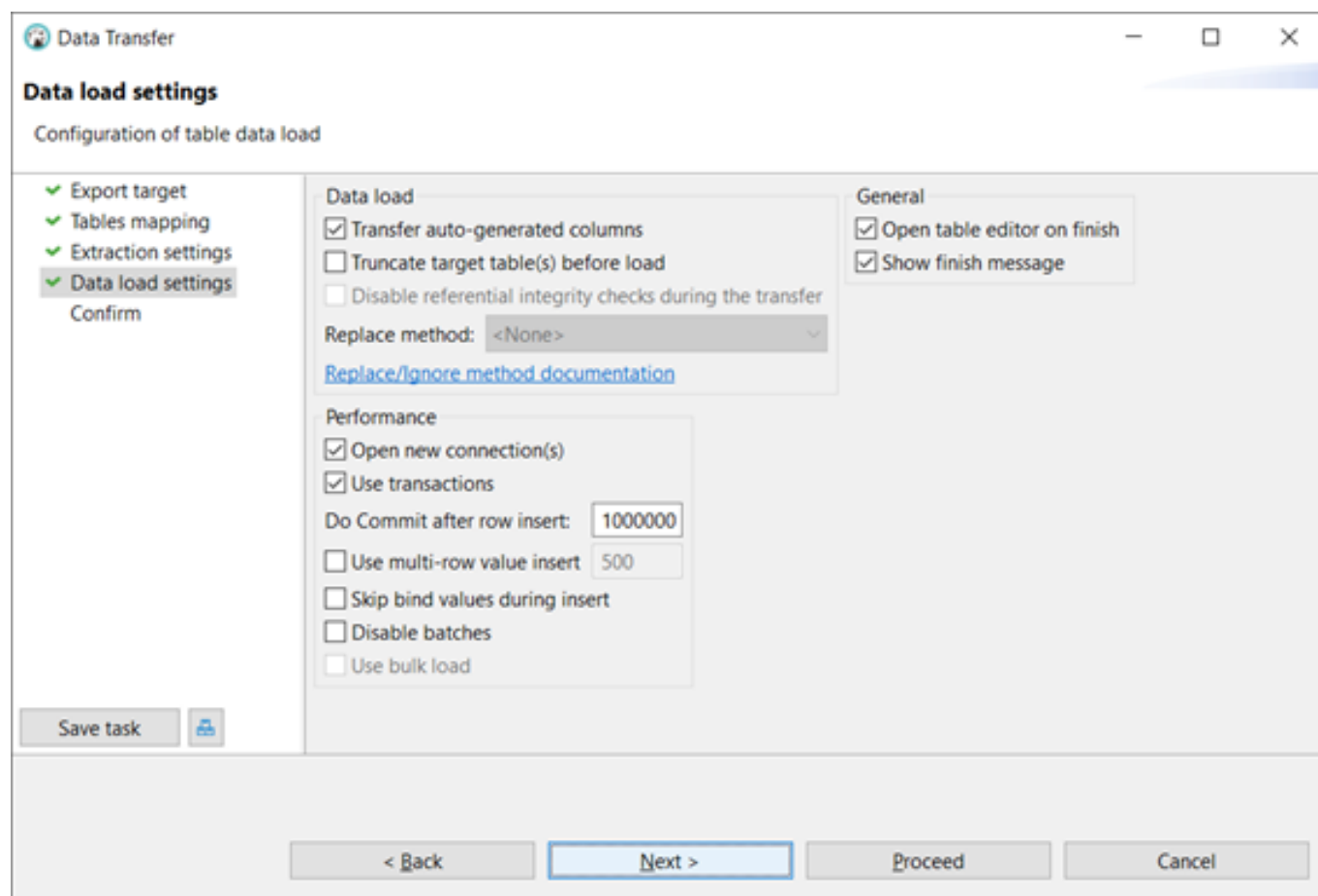
☐ Open new connection(s)

☐ Select row count

Fetch size:



11. Accept the default values in the Data load settings and click Next.



Data Transfer

Data load settings

Configuration of table data load

- ✓ Export target
- ✓ Tables mapping
- ✓ Extraction settings
- ✓ **Data load settings**
- Confirm

Data load

☒ Transfer auto-generated columns

☐ Truncate target table(s) before load

☐ Disable referential integrity checks during the transfer

Replace method:

[Replace/ignore method documentation](#)

Performance

☒ Open new connection(s)

☒ Use transactions

Do Commit after row insert:

☐ Use multi-row value insert

☐ Skip bind values during insert


☐ Disable batches

☐ Use bulk load

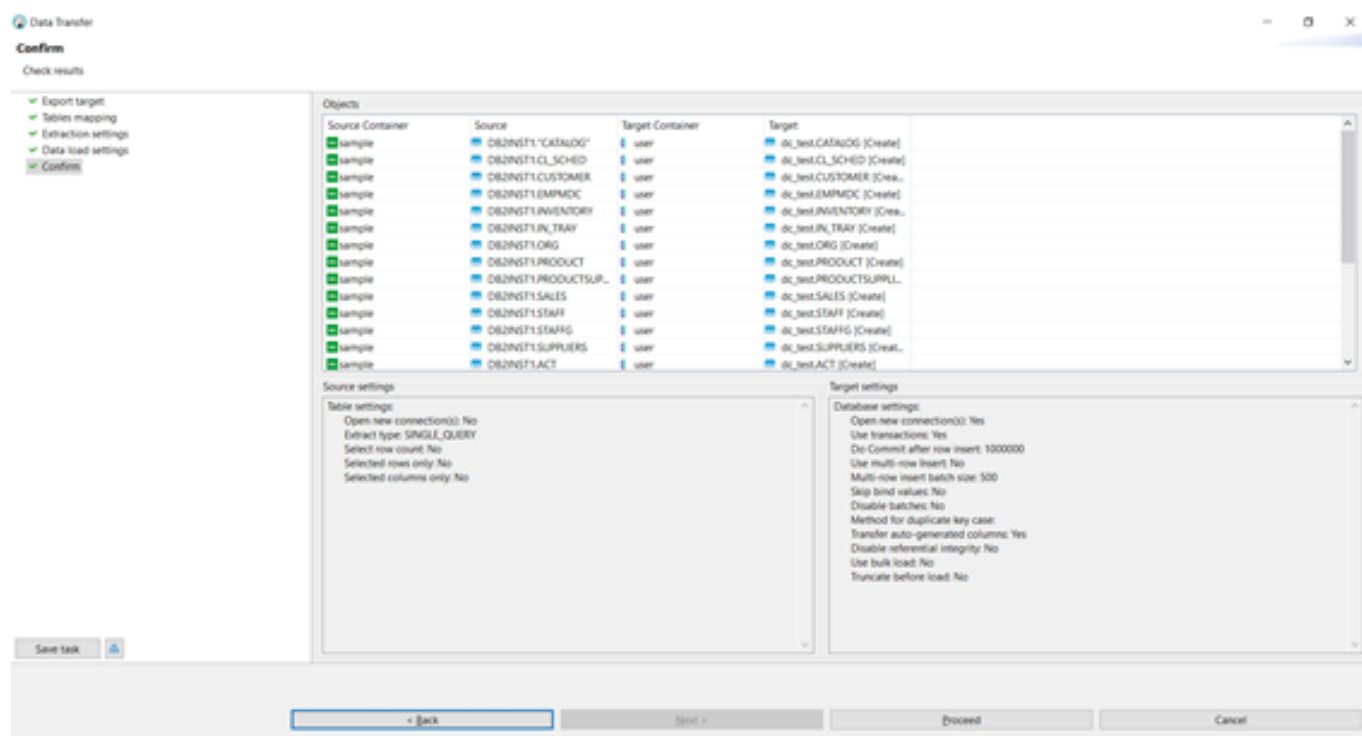
General

☒ Open table editor on finish

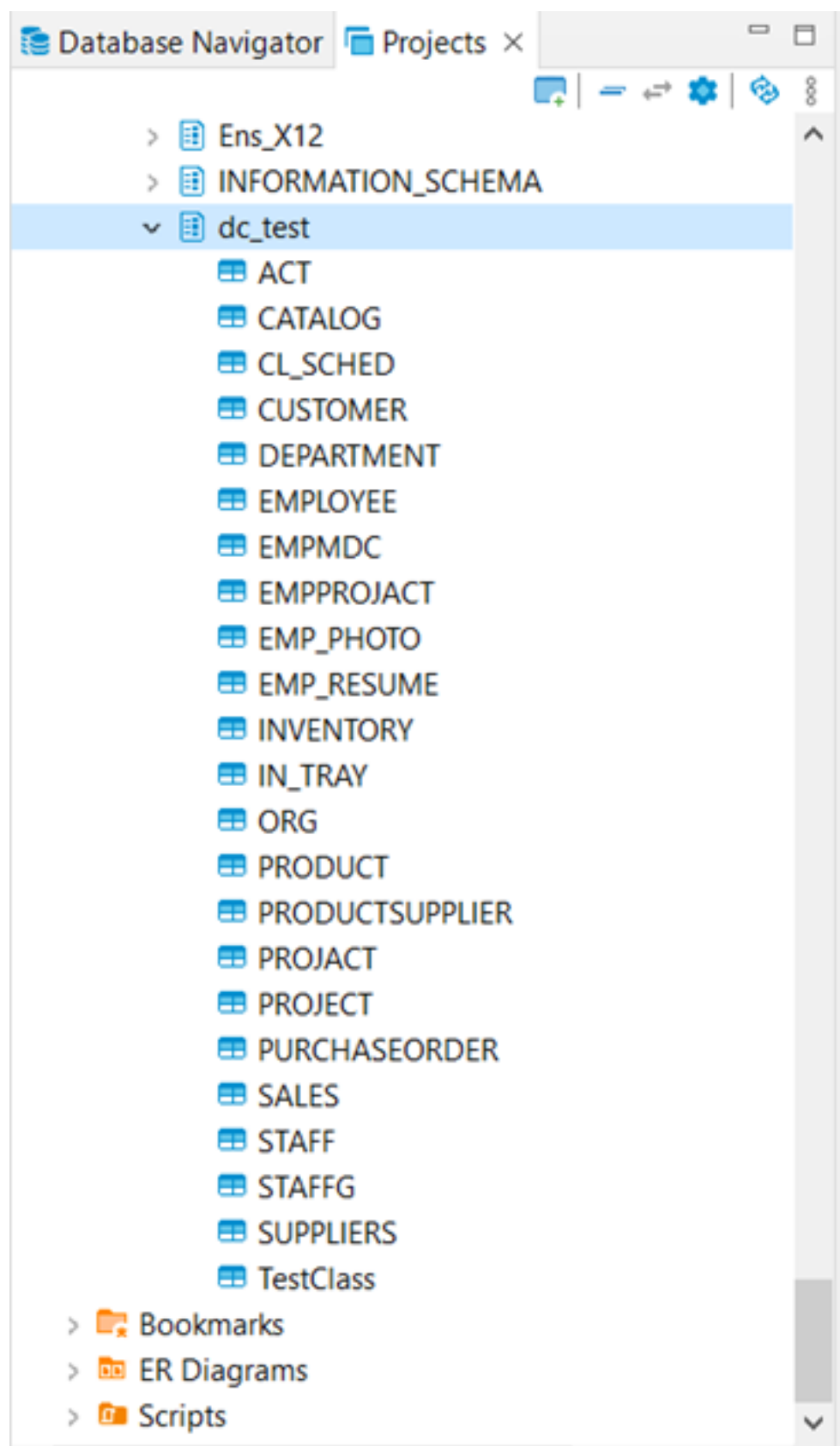
☒ Show finish message



12. In the Confirm click Proceed.



13. Now you can see all DB2 Tables inside InterSystems IRIS `dc_test` schema in the Database Navigator.



The migration process was very simple for tables. However, for views, functions, triggers and stored procedures, you will need to rewrite the SQL source code using ObjectScript or SQL.

[#Data Import and Export](#) [#InterSystems IRIS](#)

