**Migration of SQL database northwind and pubs from MS SQL Server 2016 to PostgreSQL 14.10**

Contents

[Introduction: 1](#_Toc155985187)

[Why PostgreSQL 1](#_Toc155985188)

[Tools Used: 1](#_Toc155985189)

[Details of Converting Date and Time Data Types 1](#_Toc155985190)

[Converting Data Types 2](#_Toc155985191)

[Detailed migration steps 3](#_Toc155985192)

[Northwind: 3](#_Toc155985193)

[Pubs 19](#_Toc155985194)

# Introduction:

This document provides detailed information on migrating data from Microsoft SQL server 2016 to PostgreSQL 14.10 using tool Inspirer SQL Ways 10

## Why PostgreSQL

If you're considering migrating your database away from SQL Server, PostgreSQL is a database system that should be on your radar. Here are several compelling reasons why PostgreSQL database might be the best option for your migration:

* Open Source and Cost-Effective. PostgreSQL's open-source nature means no licensing costs, making it a budget-friendly choice for organizations looking to migrate SQL Server to PostgreSQL.
* Extensive Ecosystem. PostgreSQL ecosystem offers a variety of extensions and tools to customize and tailor the database to meet specific needs, making it versatile and adaptable.
* Advanced Features. PostgreSQL boasts advanced features like JSON and XML support, full-text search, and geospatial capabilities, enabling the development of modern and feature-rich applications.
* The best option for moving to a cloud. PostgreSQL is often favored for cloud migrations due to its open-source nature, SQL standards compliance, and cost-effectiveness. Its active community support, advanced features, and compatibility with major cloud platforms, along with a mature ecosystem, make it a strong choice for scalable and reliable cloud database solutions.
* Community and Support. With an active and supportive community, PostgreSQL users have access to valuable resources and can find professional support options, ensuring they receive assistance when needed.

## Tools Used:

1. SQL Server Management Studio 19.3
2. PgAdmin4
3. Inspirer SQLWays Wizard 10 Trial edition
4. Microsoft SQL Server 2016 developer edition
5. PostgreSQL 14.10

## Details of Converting Date and Time Data Types

|  |  |  |
| --- | --- | --- |
| **SQL Server** |  | **PostgreSQL** |
| DATE | Date (year, month and day) | DATE |
| DATETIME | Date and time with fraction (milliseconds) | TIMESTAMP(3) |
| DATETIME2(p) | Date and time with fraction, 0 <= p <= 7, default is 7 | TIMESTAMP(p) |
| DATETIMEOFFSET(p) | Date and time with fraction and time zone | TIMESTAMP(p) WITH TIME ZONE |
| SMALLDATETIME | Date and time | TIMESTAMP(0) |

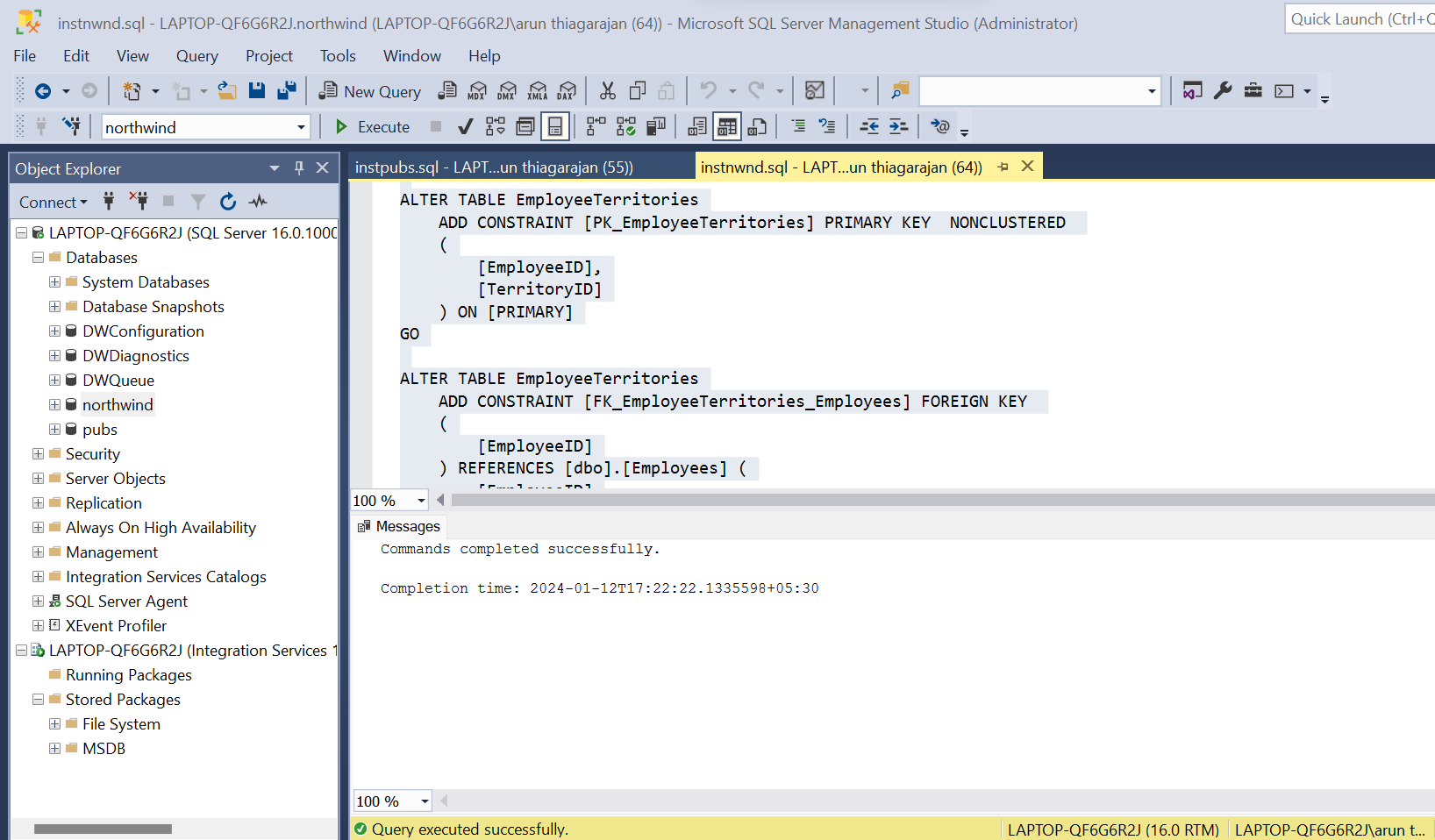
## Converting Data Types

|  |  |  |
| --- | --- | --- |
| SQL Server |  | PostgreSQL |
| BIGINT | 64-bit integer | BIGINT |
| BINARY(n) | Fixed-length byte string | BYTEA |
| BIT | 1, 0 or NULL | BOOLEAN |
| CHAR(n), CHARACTER(n) | Fixed-length character string, 1 ⇐ n ⇐ 8000 | CHAR(n), CHARACTER(n) |
| DECIMAL(p,s), DEC(p,s) | Fixed-point number | DECIMAL(p,s), DEC(p,s) |
| DOUBLE PRECISION | Double-precision floating-point number | DOUBLE PRECISION |
| FLOAT(p) | Floating-point number | DOUBLE PRECISION |
| IMAGE | Variable-length binary data, ⇐ 2G | BYTEA |
| INT, INTEGER | 32-bit integer | INT, INTEGER |
| MONEY | 64-bit currency amount | MONEY |
| NCHAR(n) | Fixed-length Unicode UCS-2 string | CHAR(n) |
| NTEXT | Variable-length Unicode UCS-2 data, ⇐ 2G | TEXT |
| NUMERIC(p,s) | Fixed-point number | NUMERIC(p,s) |
| NVARCHAR(n) | Variable-length Unicode UCS-2 string | VARCHAR(n) |
| NVARCHAR(max) | Variable-length Unicode UCS-2 data, ⇐ 2G | TEXT |
| REAL | Single-precision floating-point number | REAL |
| ROWVERSION | Automatically updated binary data | BYTEA |
| SMALLINT | 16-bit integer | SMALLINT |
| SMALLMONEY | 32-bit currency amount | MONEY |
| TEXT | Variable-length character data, ⇐ 2G | TEXT |
| TIME(p) | Time (hour, minute, second and fraction) | TIME(p) |
| TIMESTAMP | Automatically updated binary data | BYTEA |
| TINYINT | 8-bit unsigned integer, 0 to 255 | SMALLINT |
| UNIQUEIDENTIFIER | 16-byte GUID (UUID) data | CHAR(16) |
| VARBINARY(n) | Variable-length byte string, 1 ⇐ n ⇐ 8000 | BYTEA |
| VARBINARY(max) | Variable-length binary data, ⇐ 2G | BYTEA |
| VARCHAR(n) | Variable-length character string, 1 ⇐ n ⇐ 8000 | VARCHAR(n) |
| VARCHAR(max) | Variable-length character data, ⇐ 2G | TEXT |
| XML | XML data | XML |

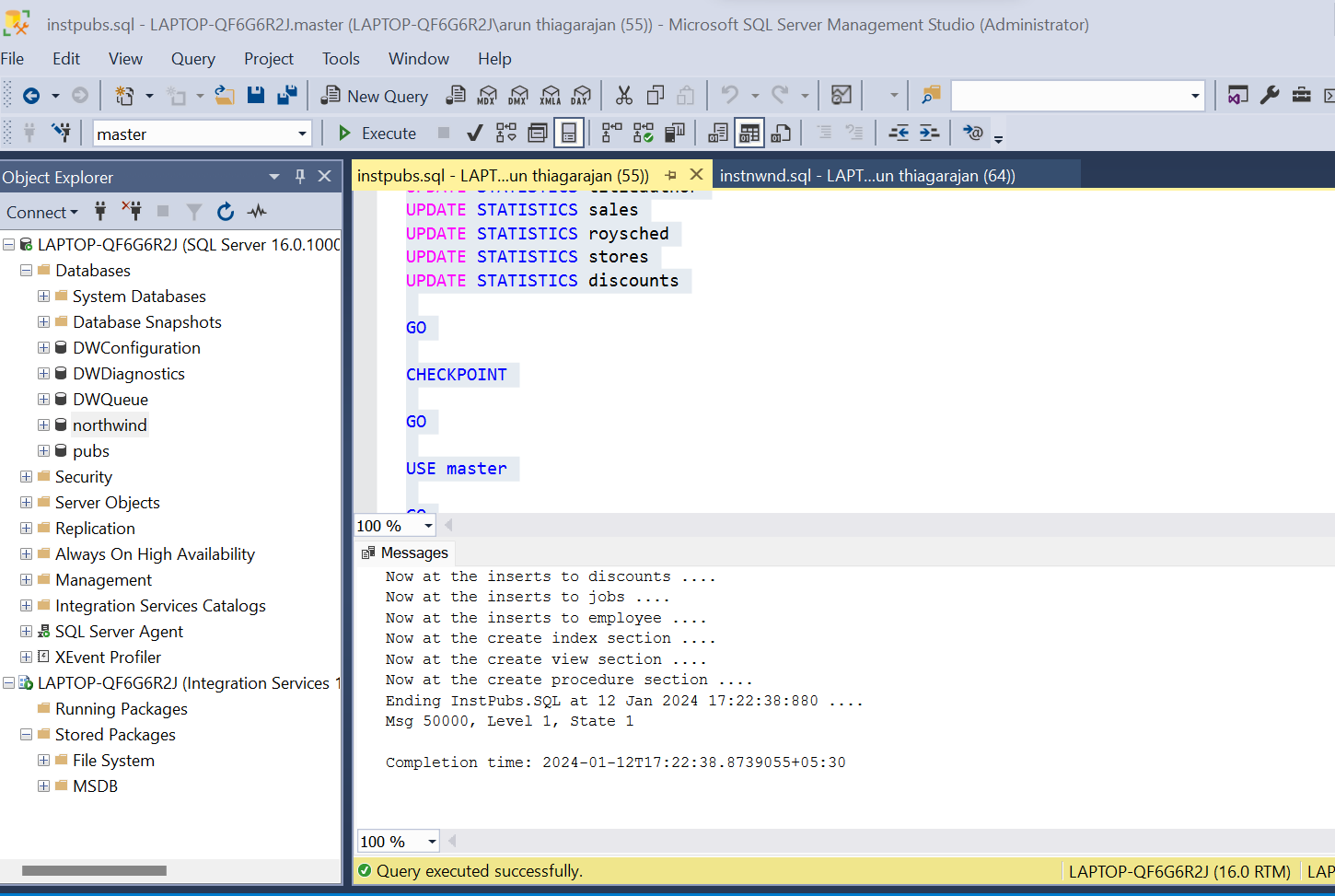
## Detailed migration steps

1. Creation and loading of northwind and pubs database

### Northwind:



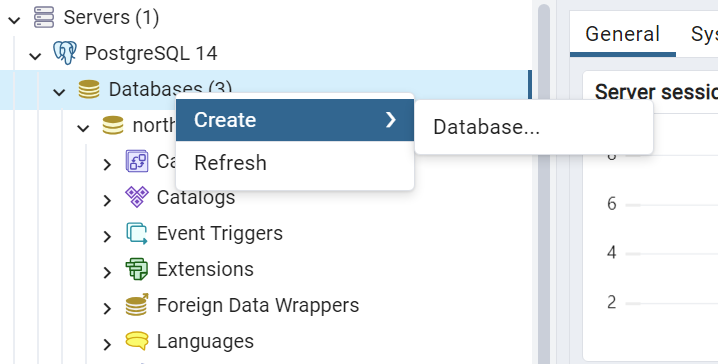
Pubs:



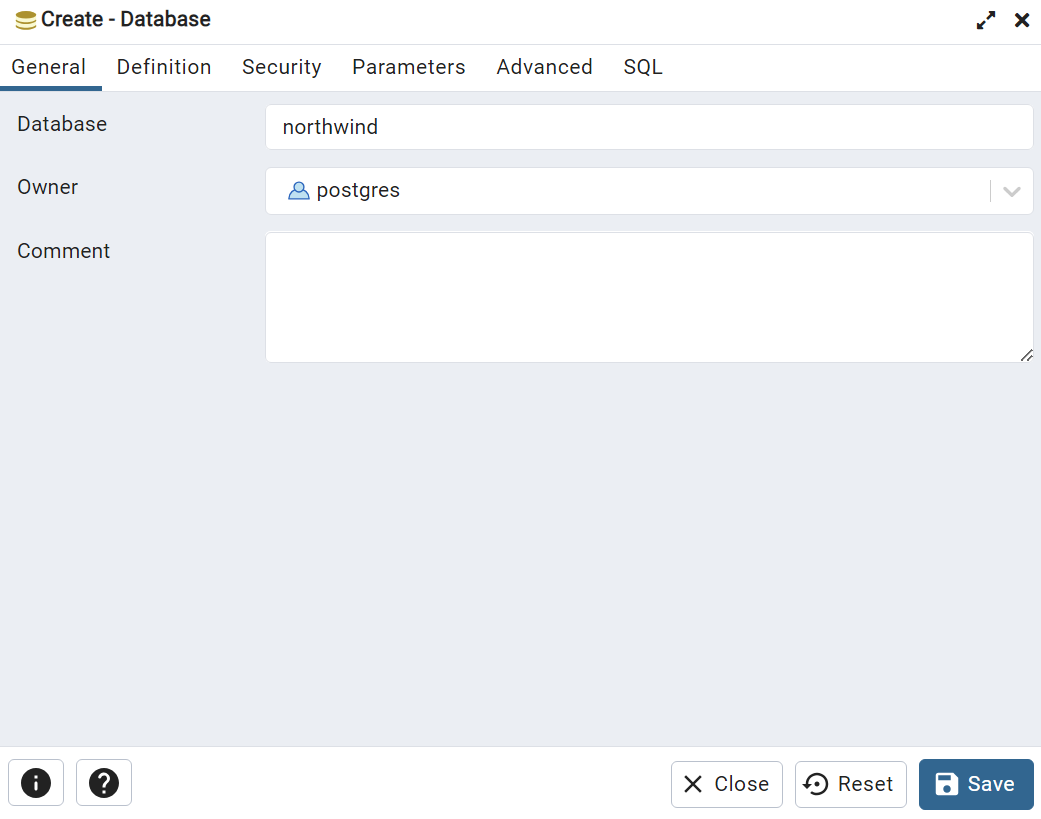
1. Creating dummy database northwind and pubs in PostgreSQL 14.10 database

Connect to PostgreSQL database using pgadmin4 tool

Right click database and select Create -> Database

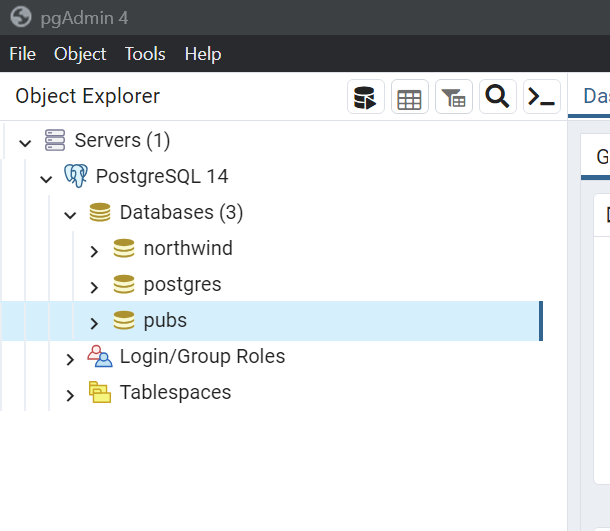


Provide database name northwind and click Save

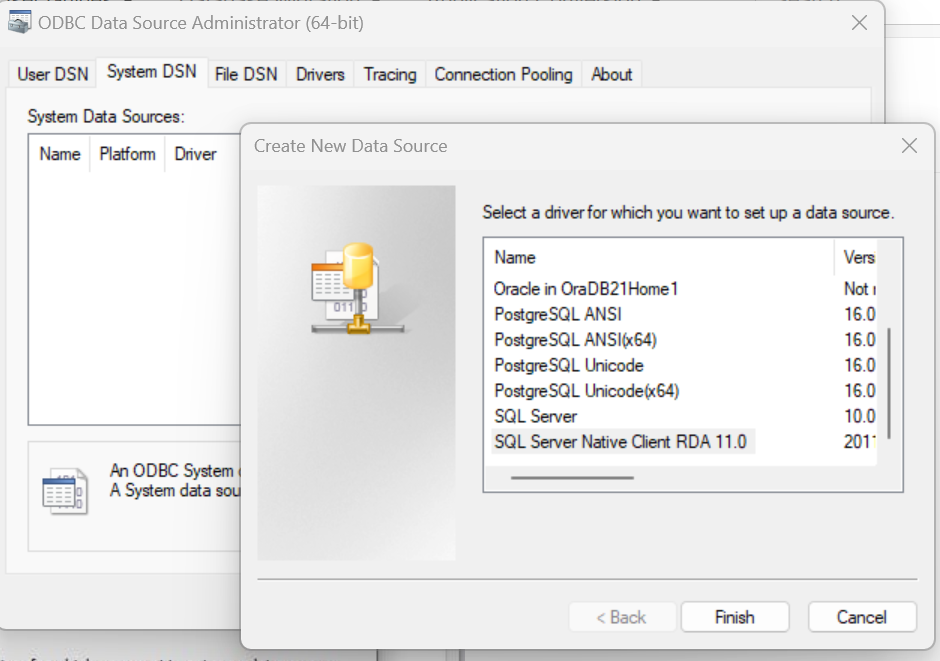


Do same step for creating pubs database

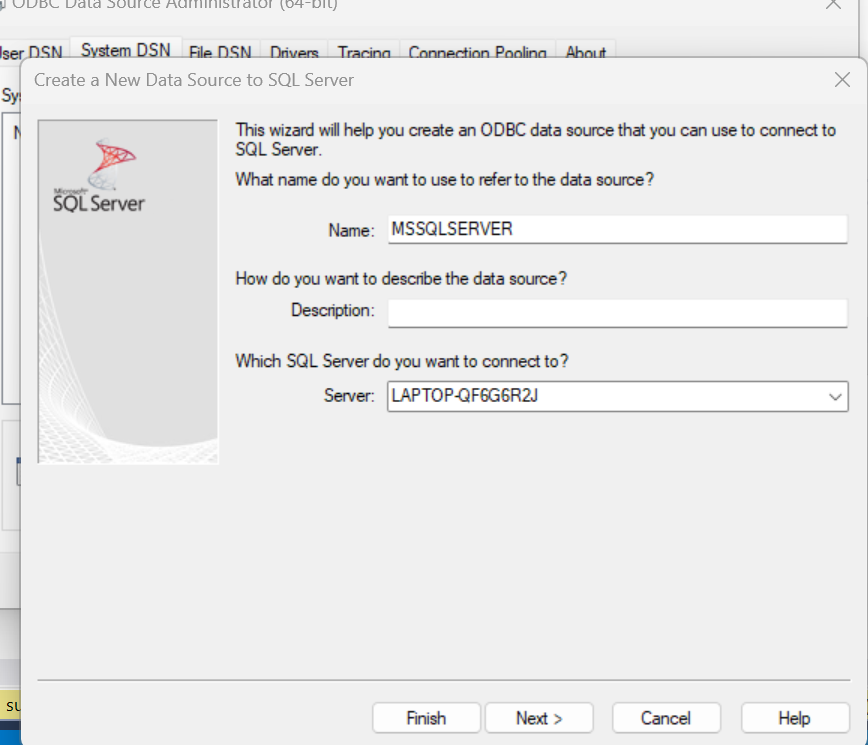
After creating new northwind and pubs database it should look like below.



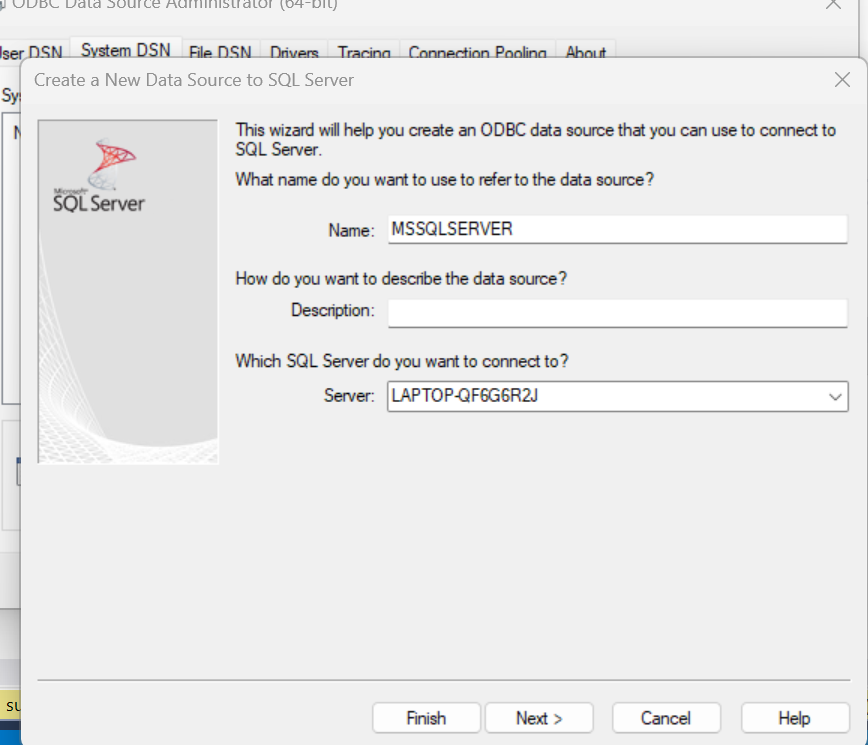
1. Migration of SQL server Database to PostgreSQL using Ispirer Toolkit 10
2. Configure MS SQL server driver ODBC (DSN)



Specify name MSSQLSERVER



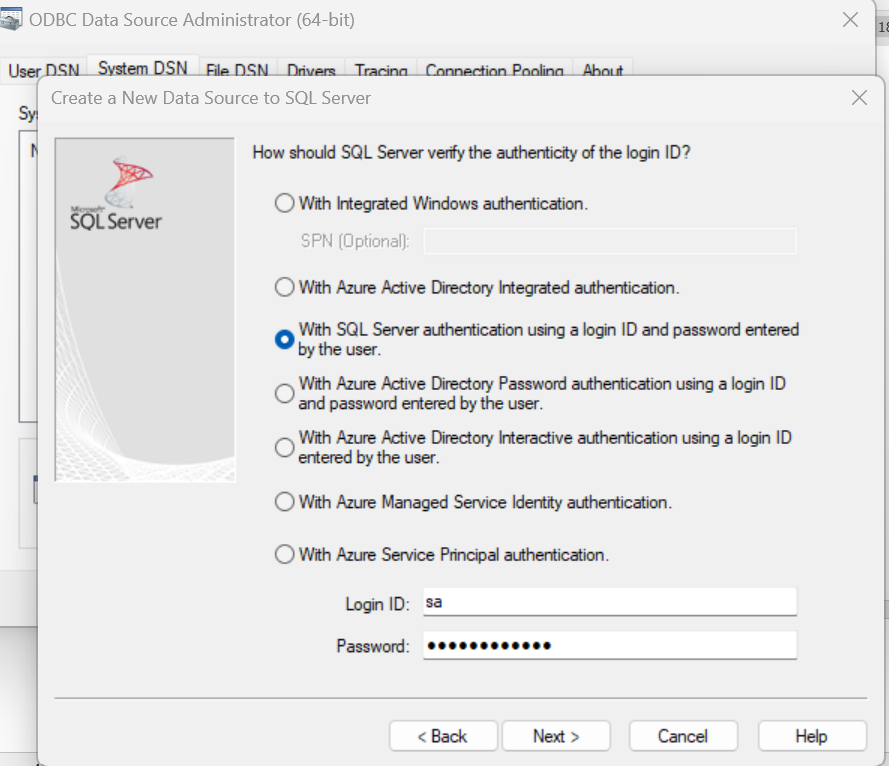
Enter hostname



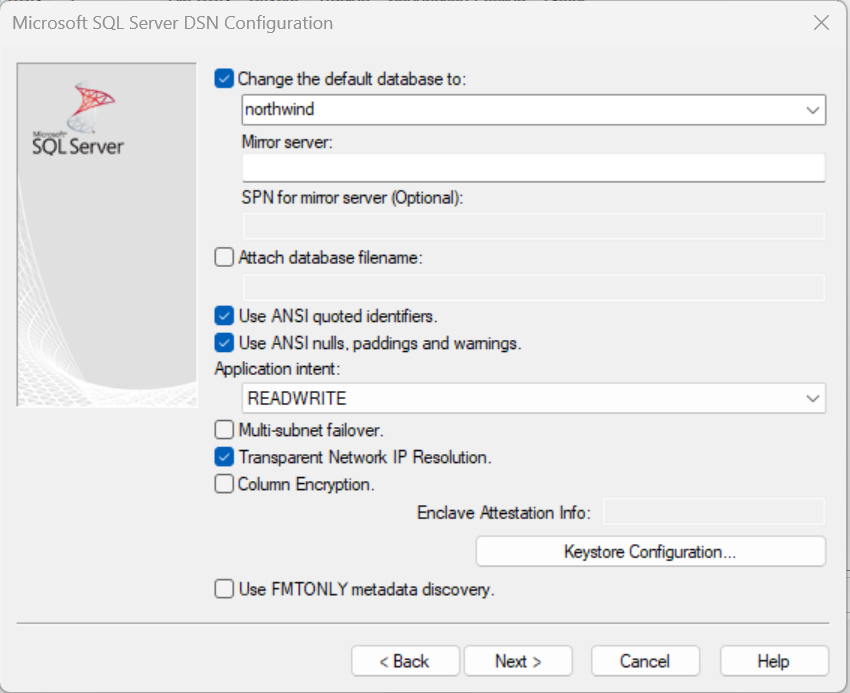
Click next

Select With SQL Server Authentication using a login ID and password entered by the user

Specify Login ID and Password

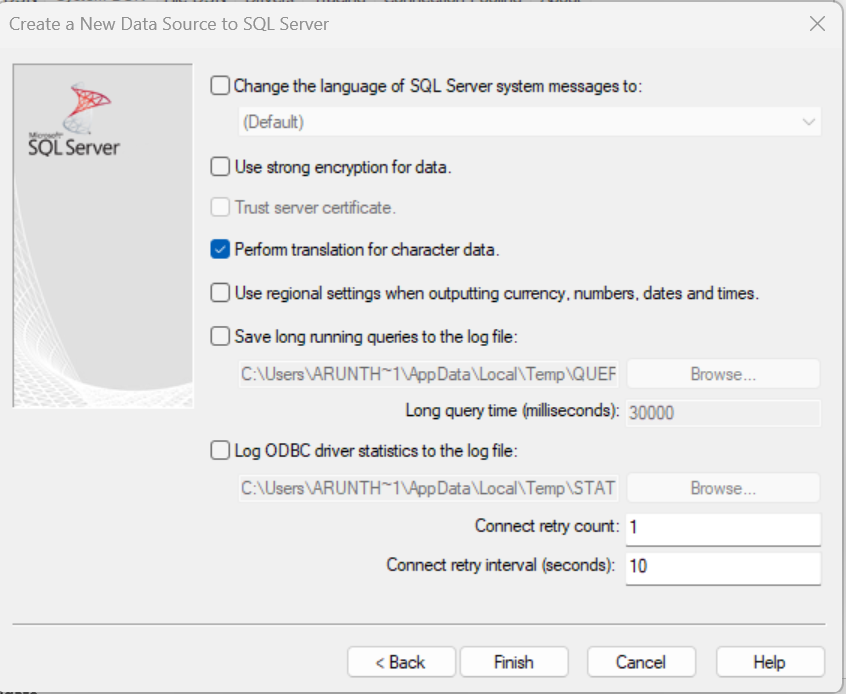


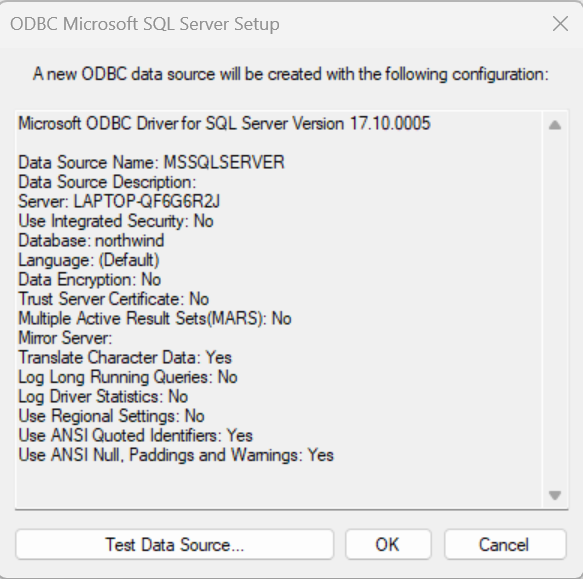
Click Next



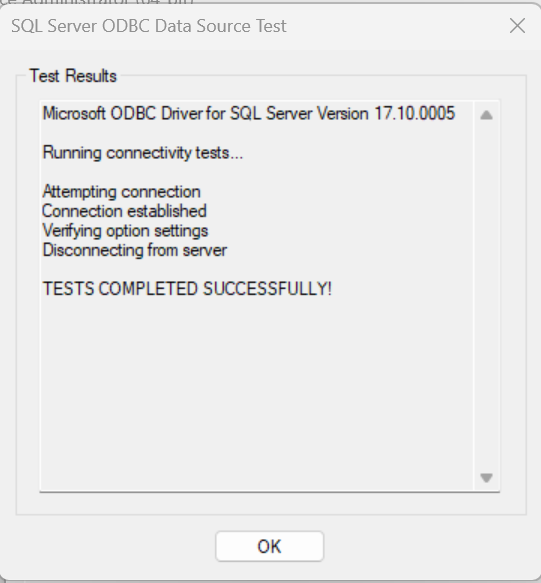
Click Next

Leave default values and click Finish

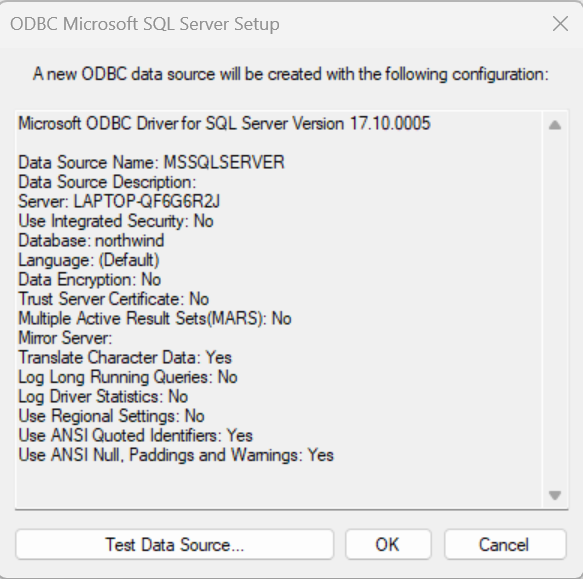




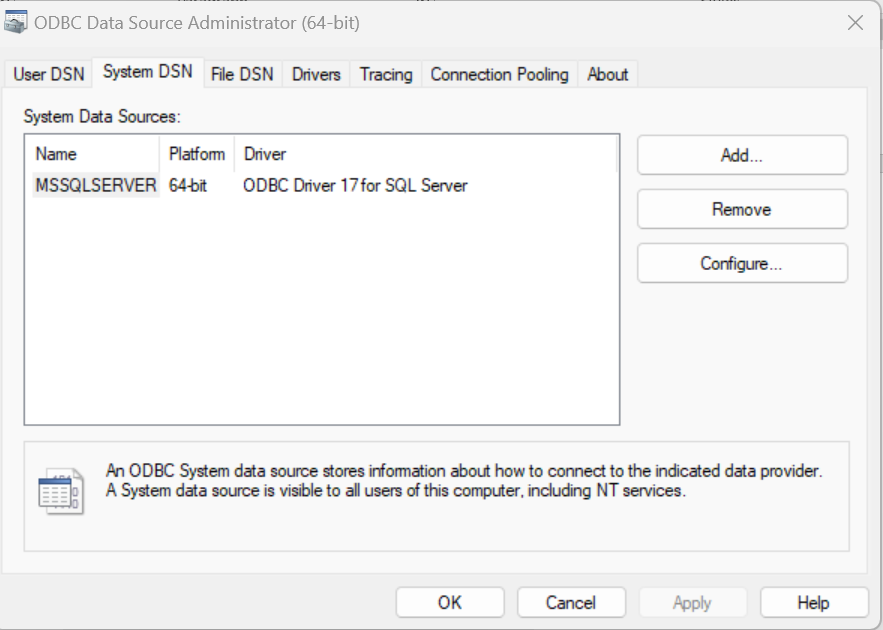
Click Test Data Source and check if connection is successful



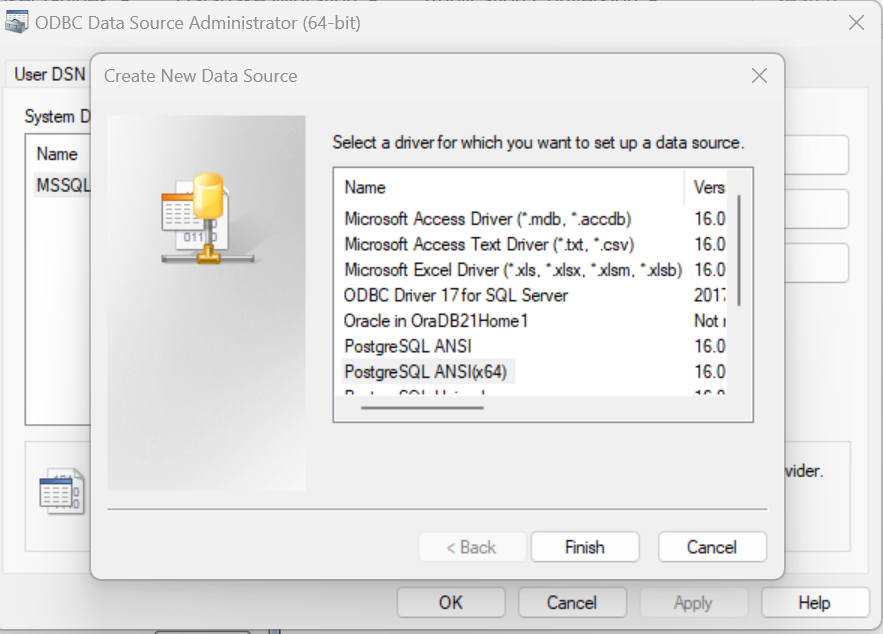
Click Ok



System DSN is now added for SQL server database



Repeat same steps for PostgreSQL



Provide connection details

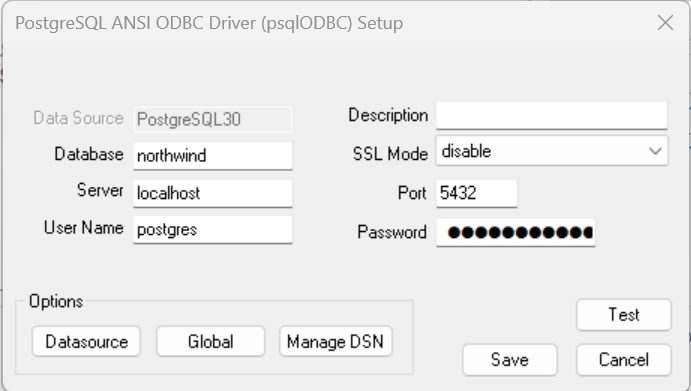
Database: postgres

Server: localhost

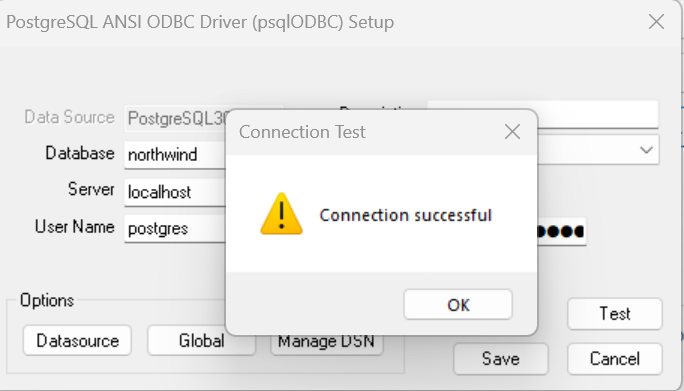
Username: postgres

Port: 5432

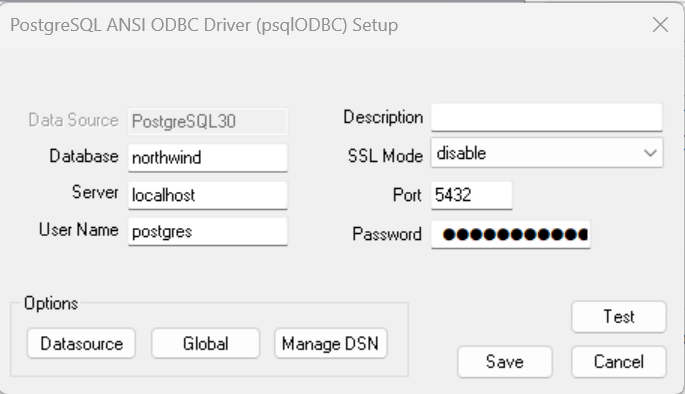
Password: Provided during setup



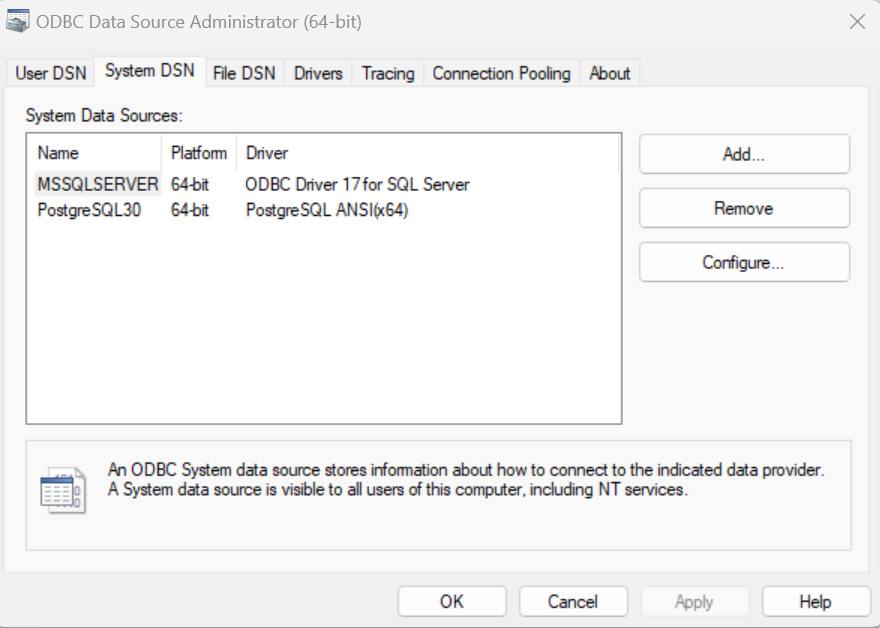
Click Test



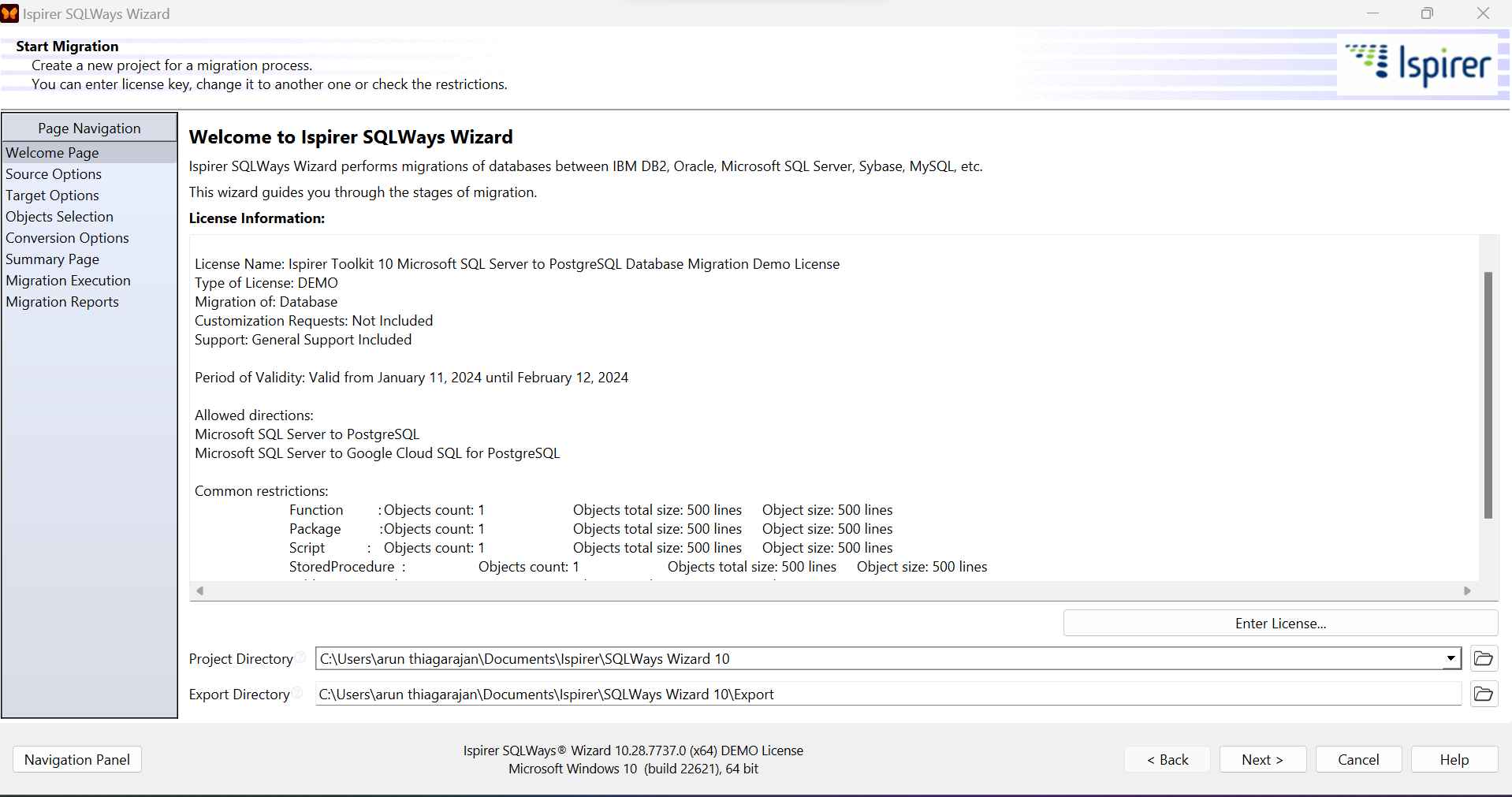
Click ok and Click Save



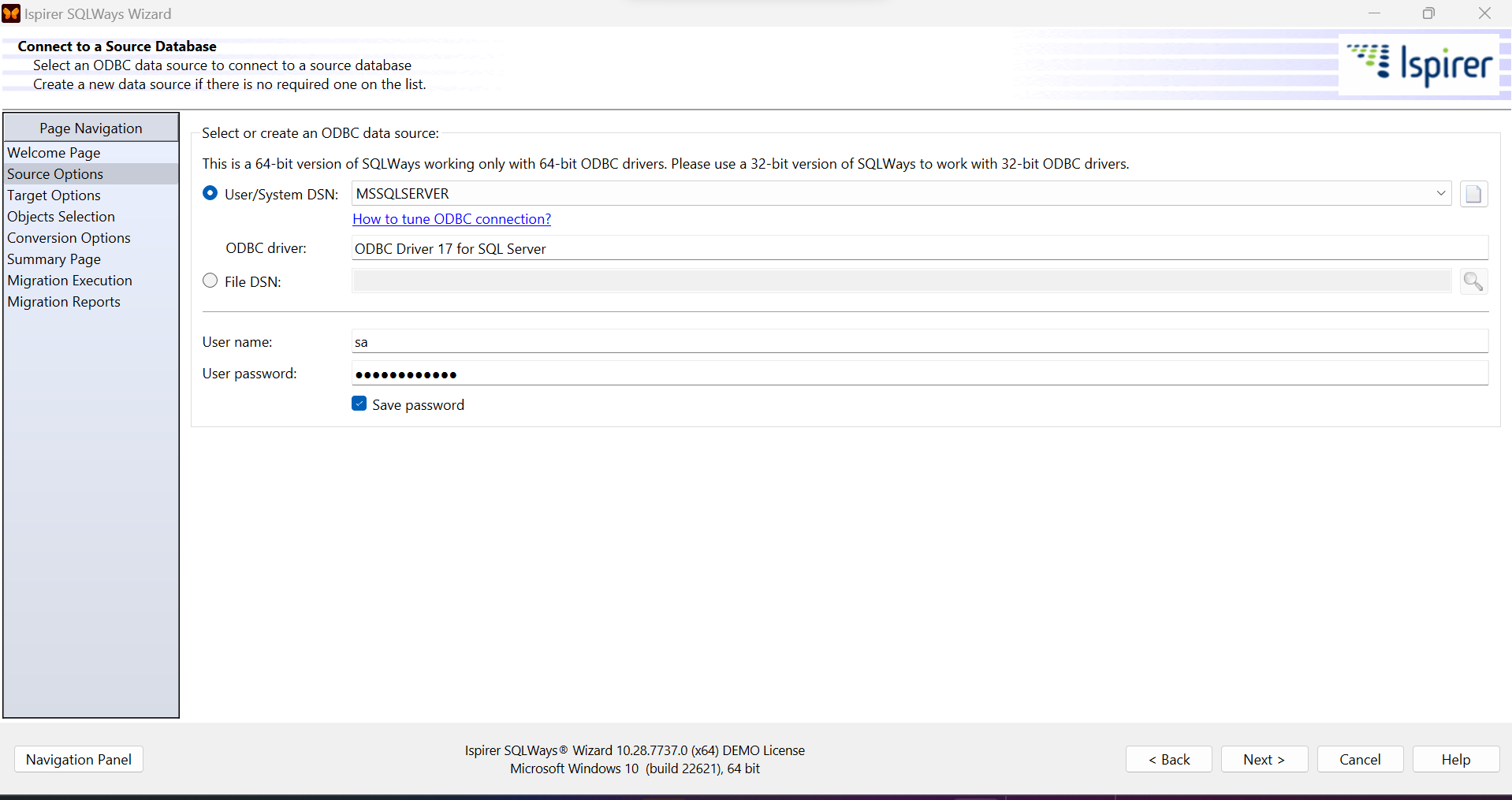
Now there should be 2 DSN’s one for SQL server and one for PostgreSQL



1. Install Ispirer Toolkit 10 free trial version
2. Open Ispirer Toolkit



1. Click Next
2. Choose Source ODBC for MS SQL Server. Provide username and password



1. Specify destination database PostgreSQL and provide connection details

Host: localhost

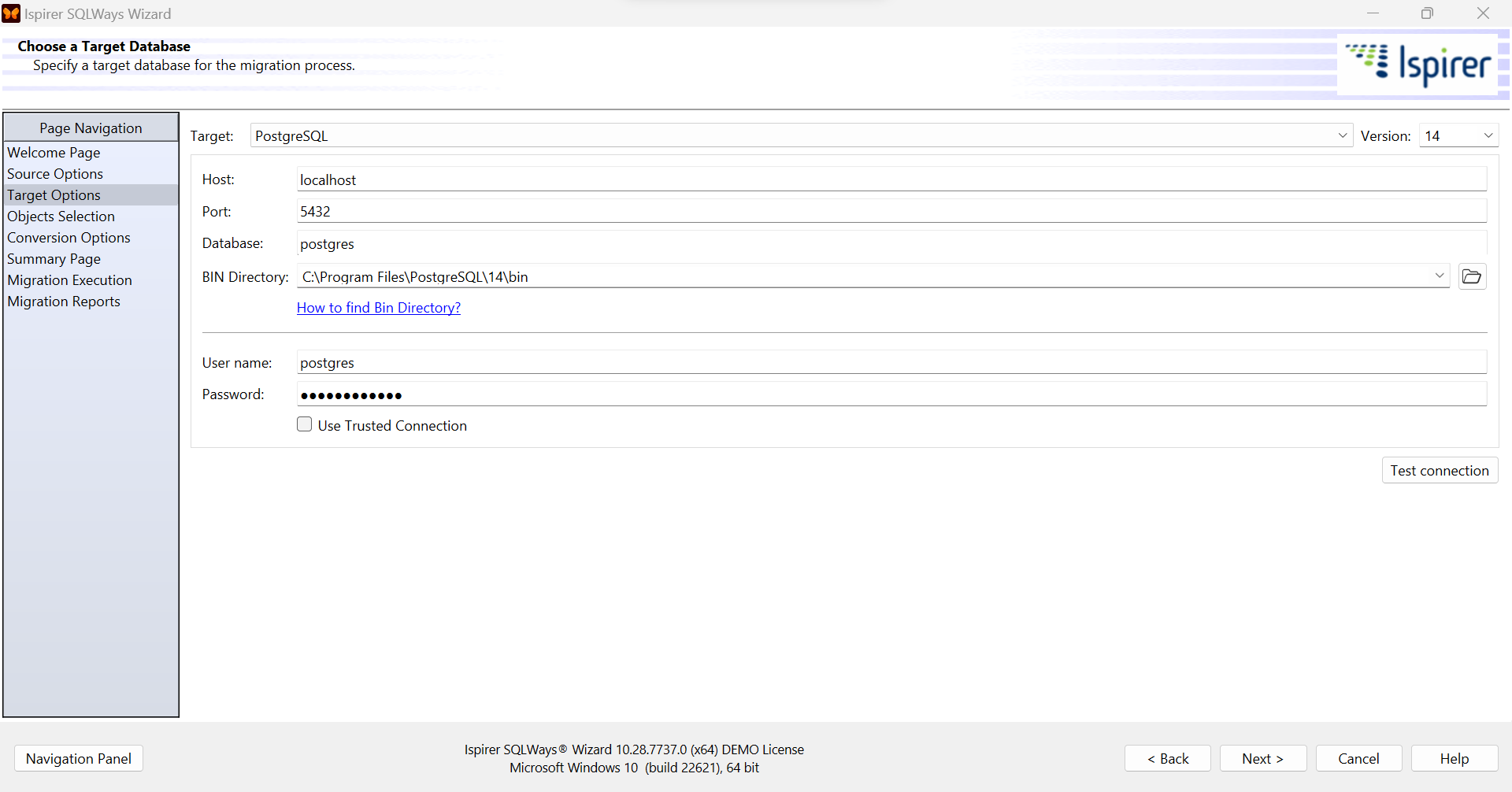
Port: 5432

Database: postgres

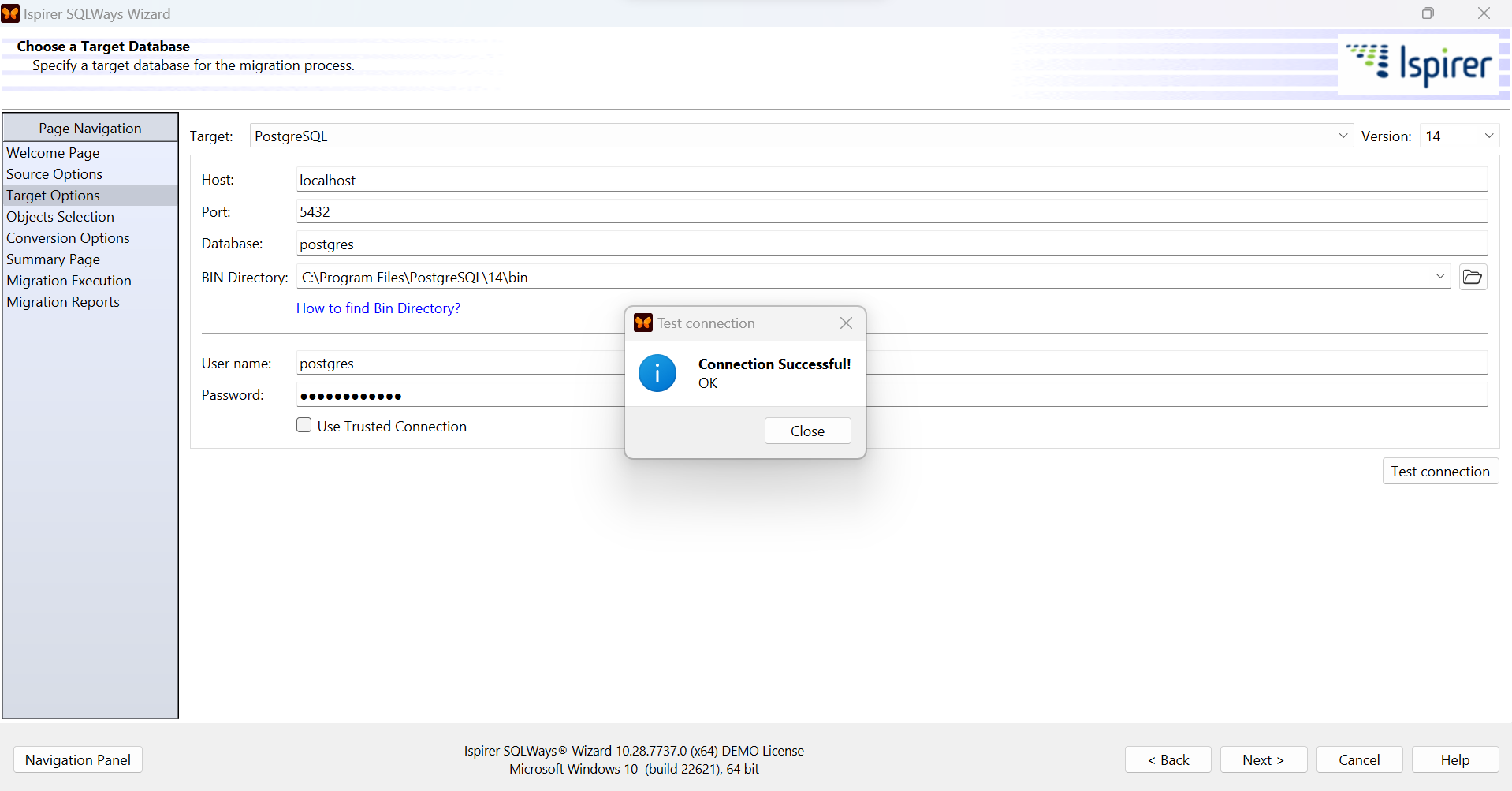
BIN Directory: C:\Program Files\PostgreSQL\14\bin

User name: postgres

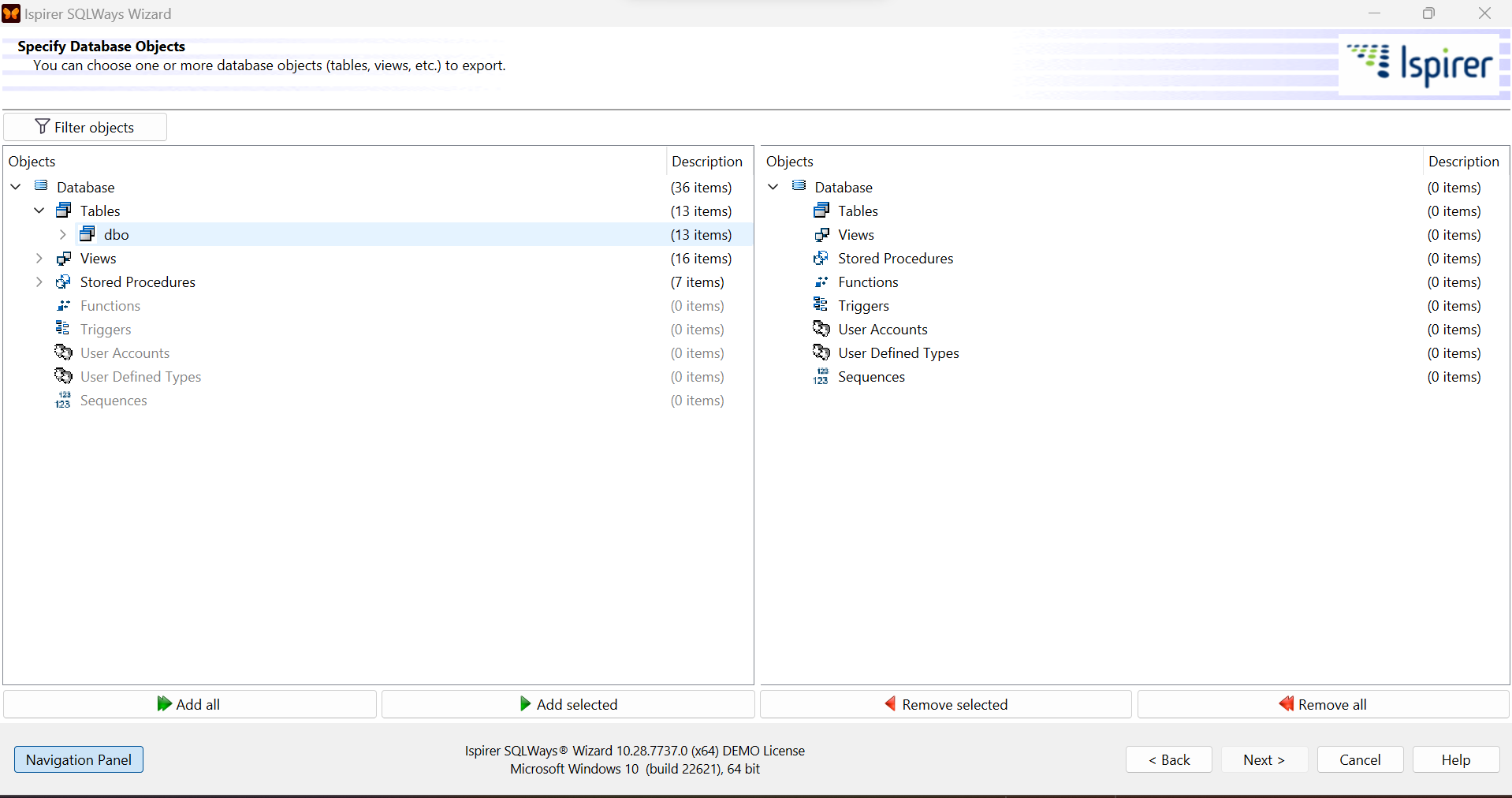
Password: Specify password specified during installation

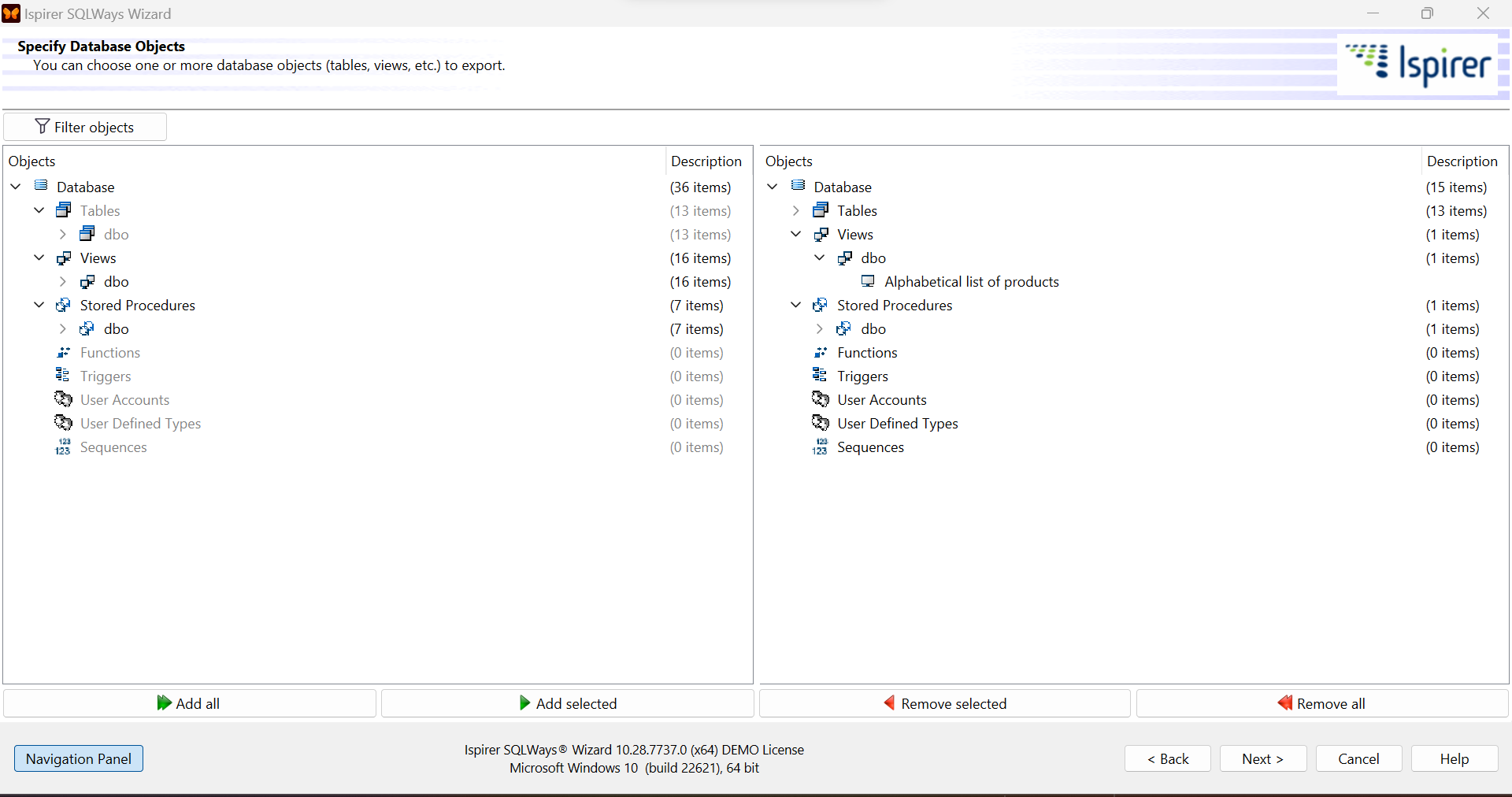


Click Test Connection

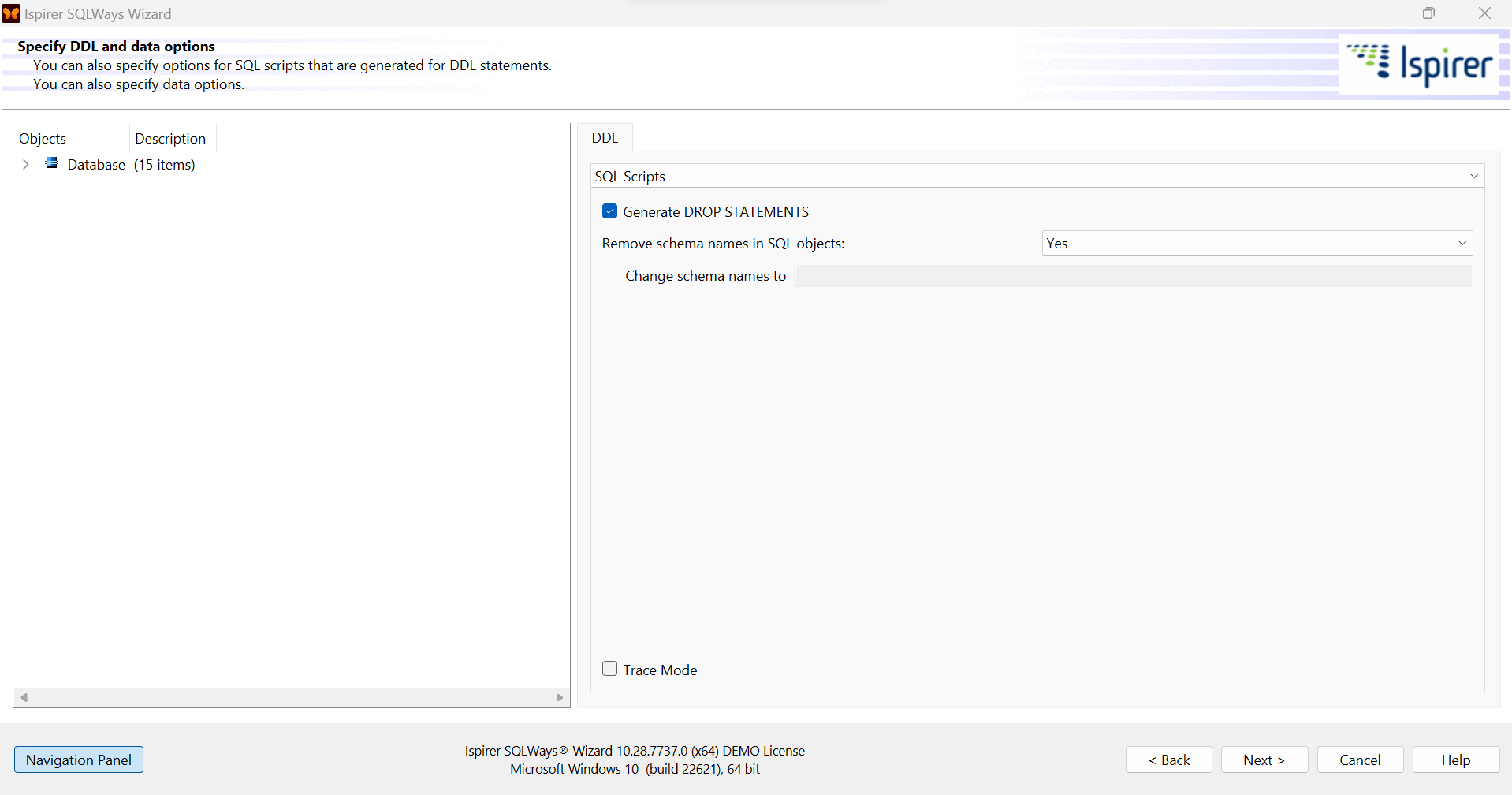


Select objects.

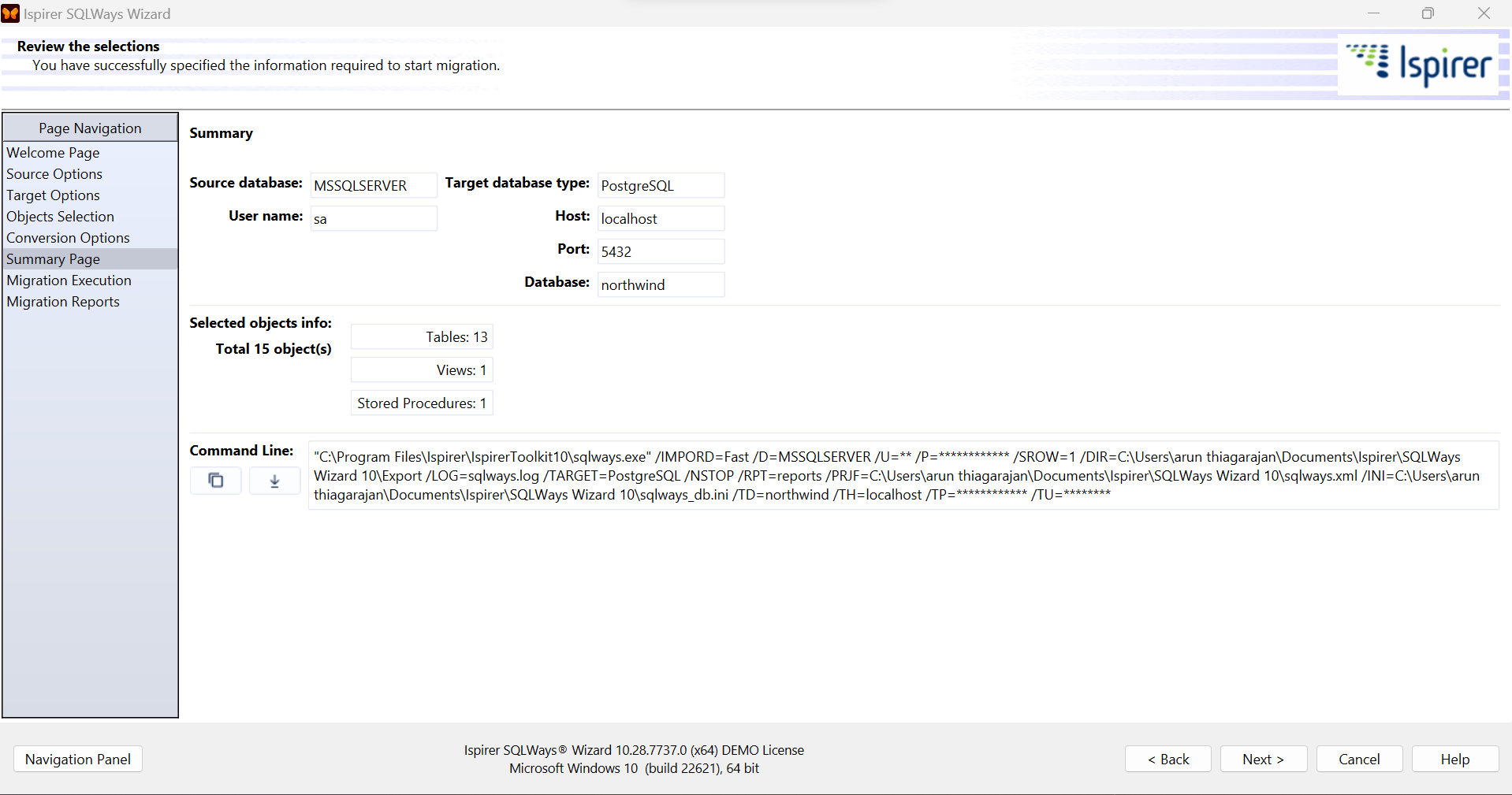




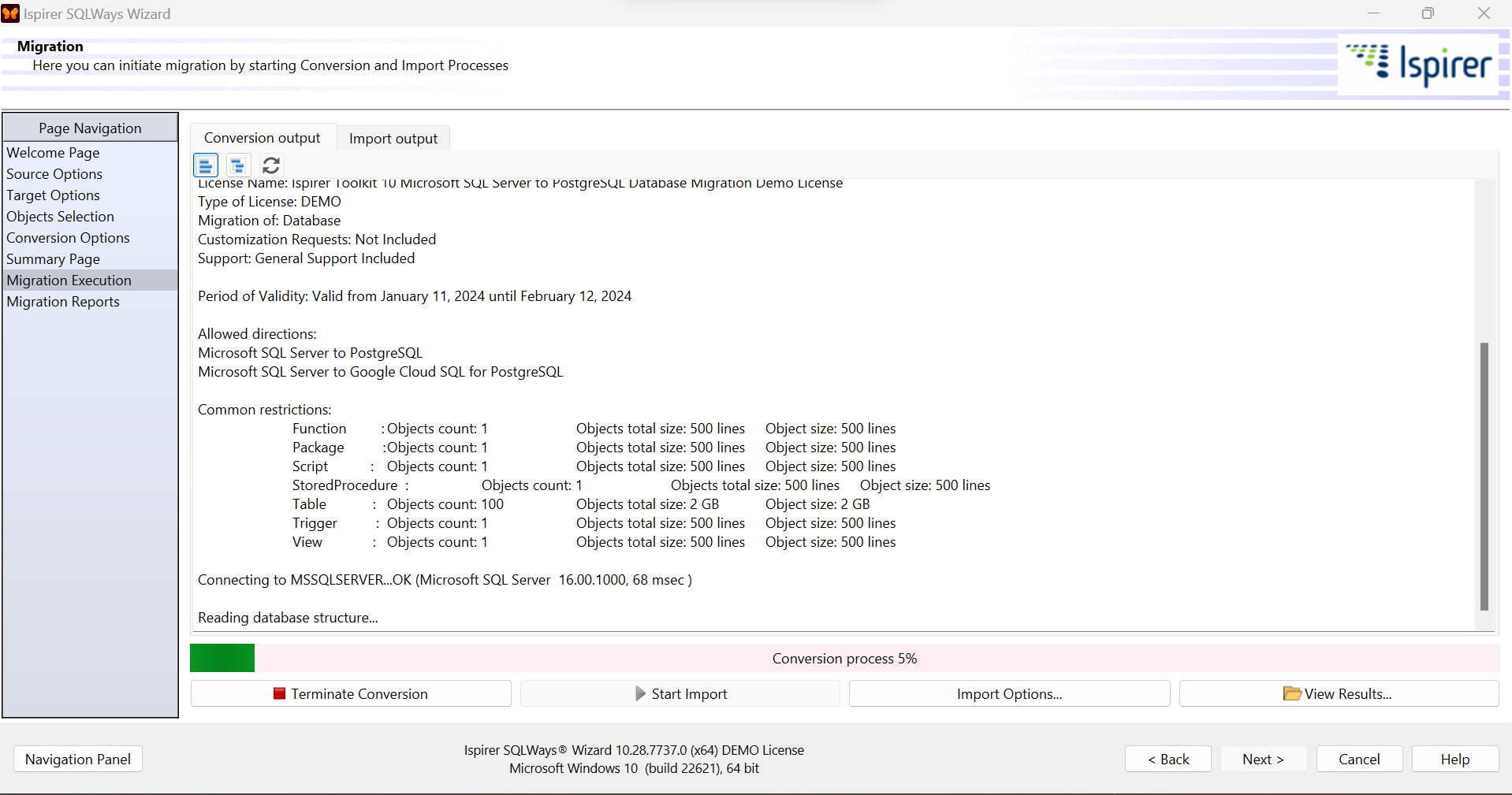
Since it is trail edition it will allow only to select 1 view and 1 stored procedure. Please re-run the above steps to migrate rest if Views and Stored procedures

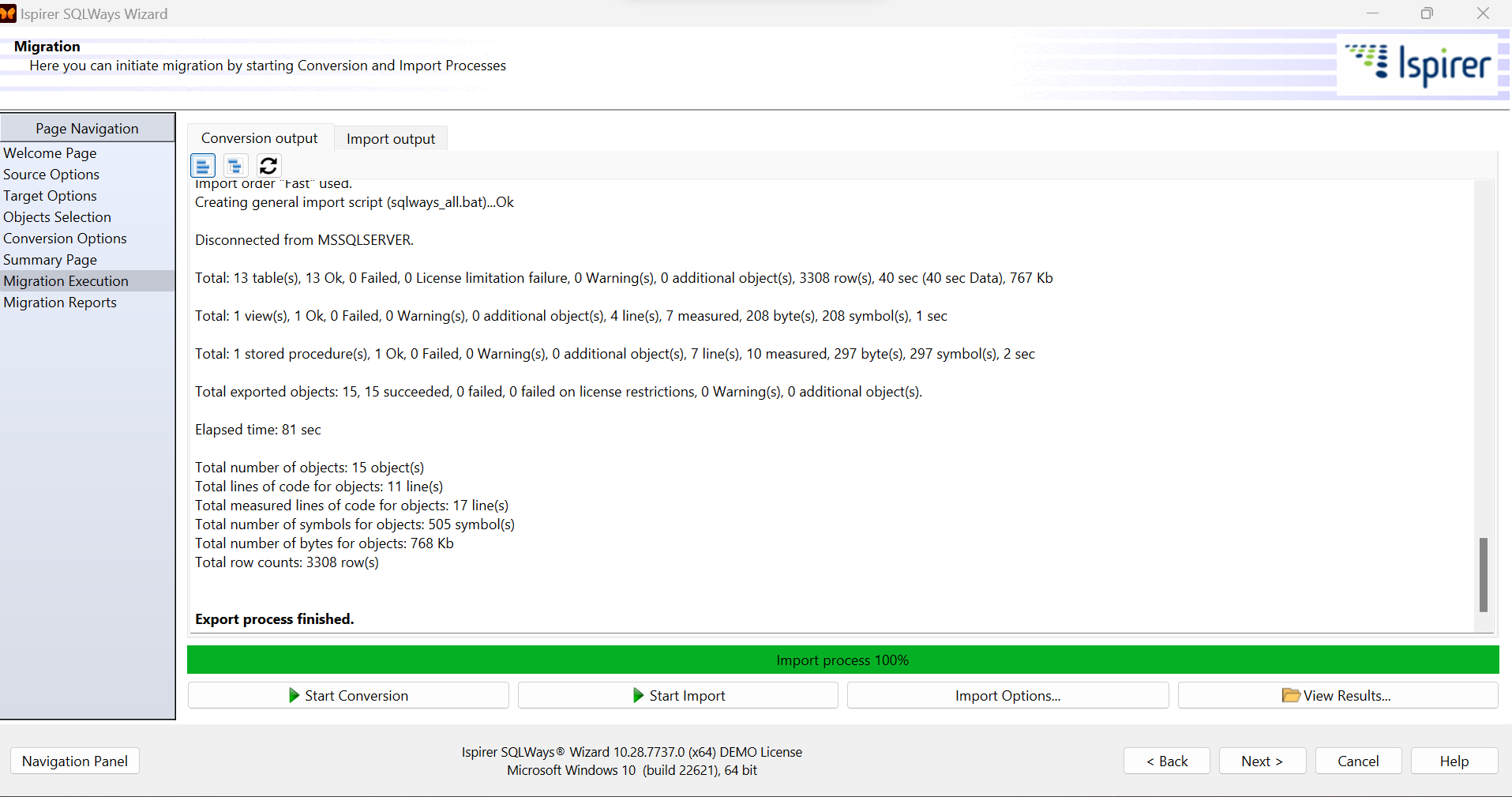


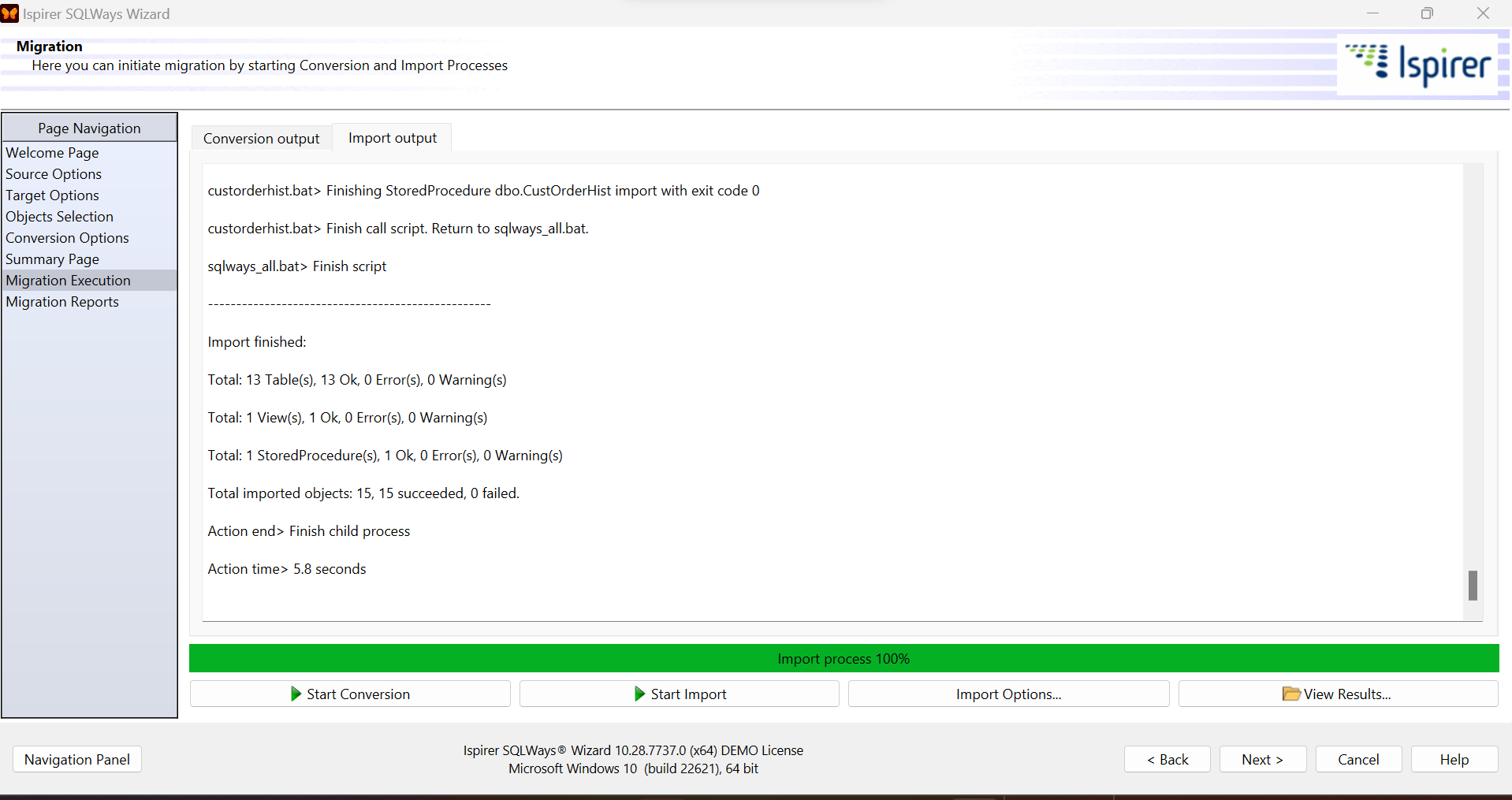
Click Next and verify objects



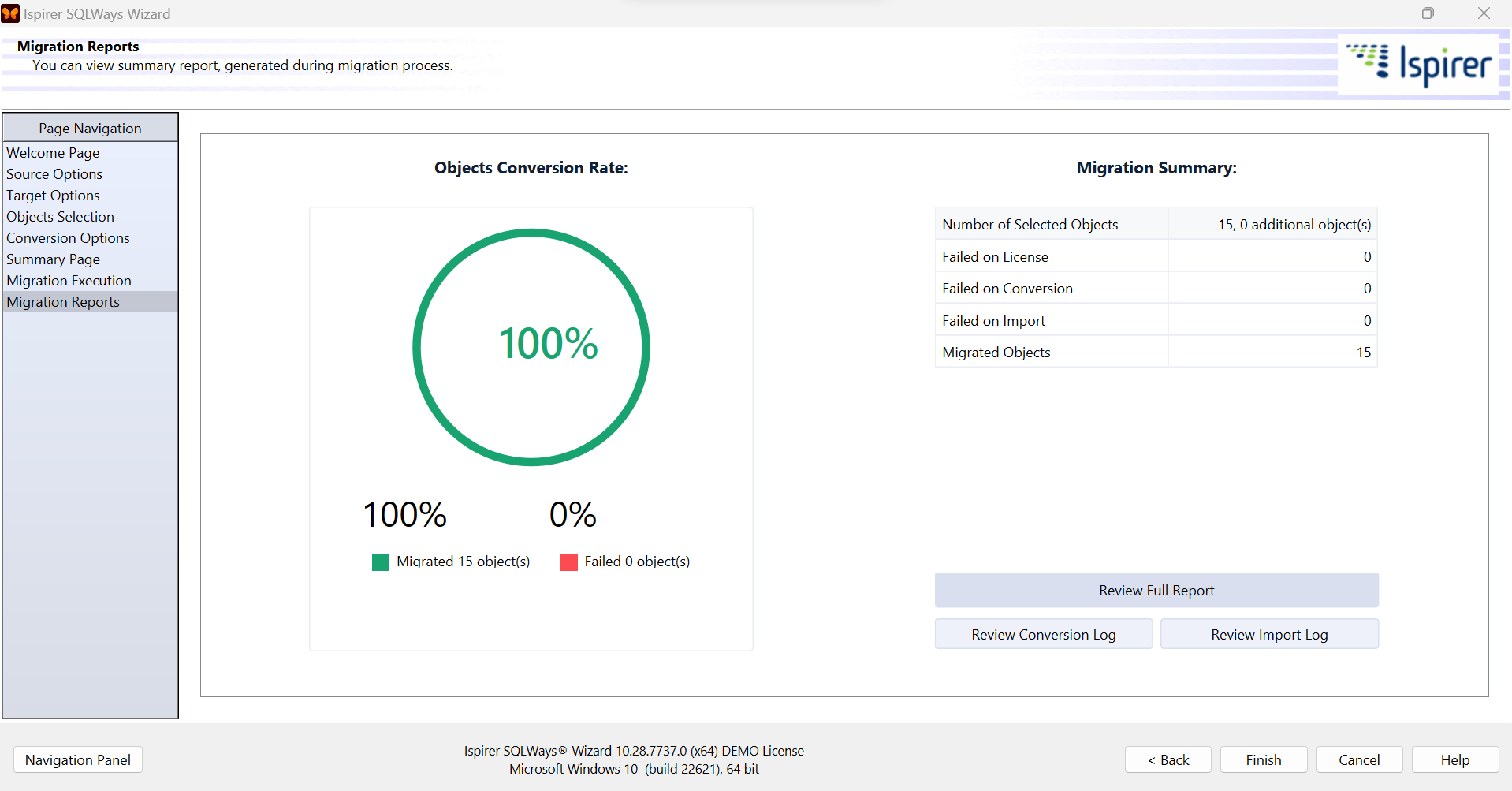
Click on Start conversion







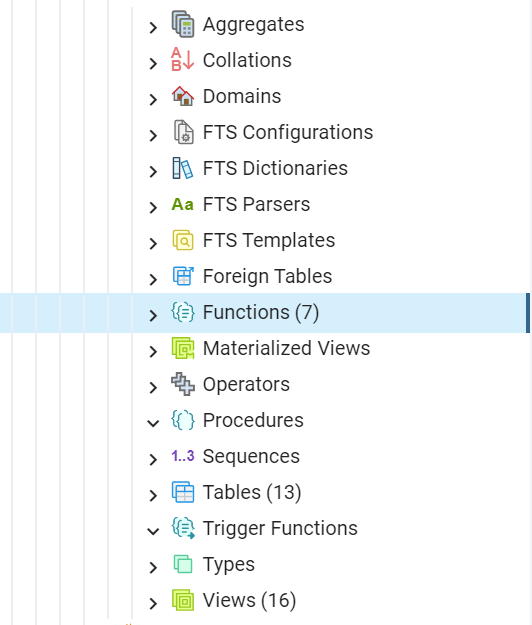
Click Next



Click Finish

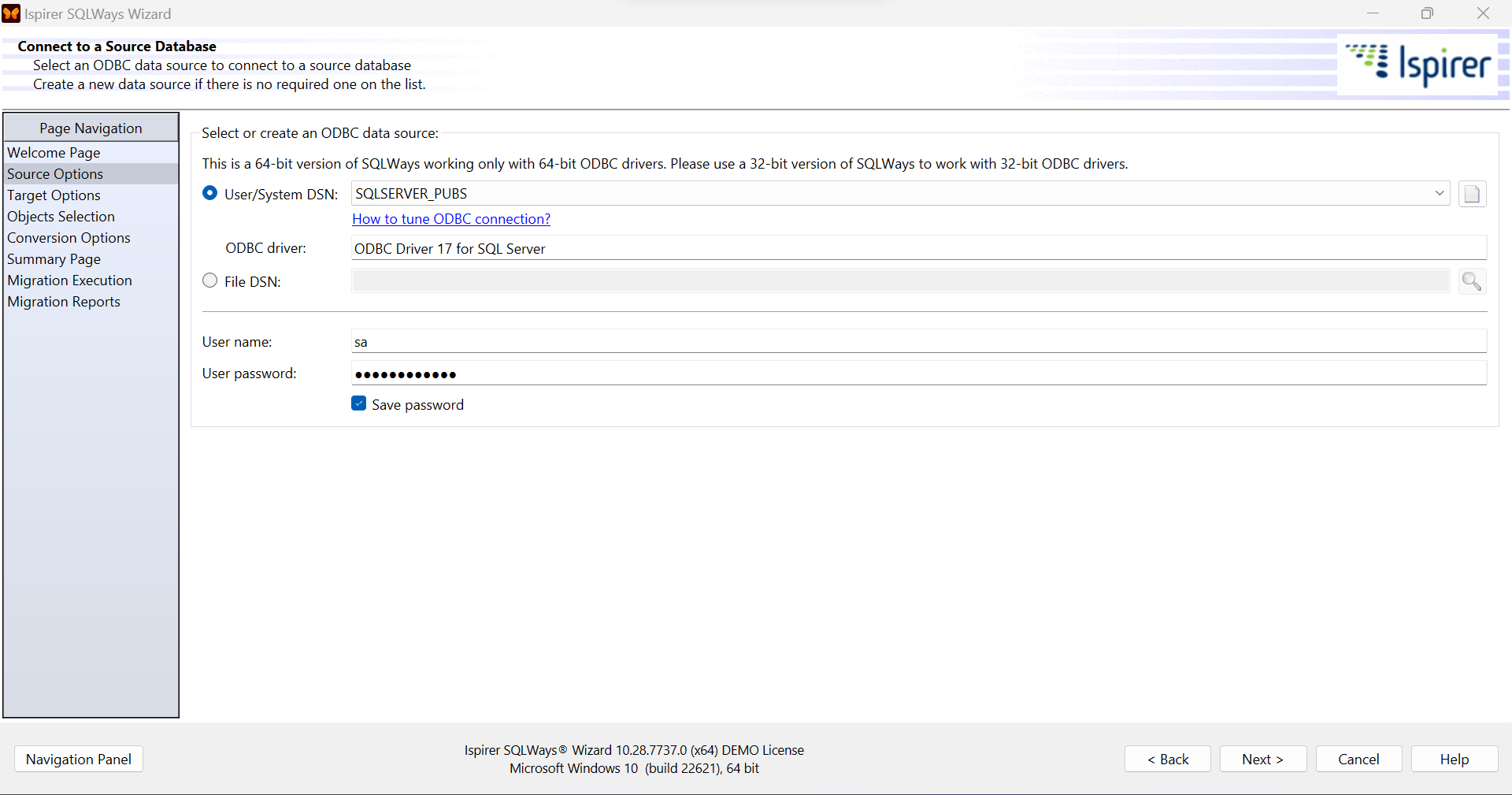
Repeat same steps above to migrate remaining views and stored procedures

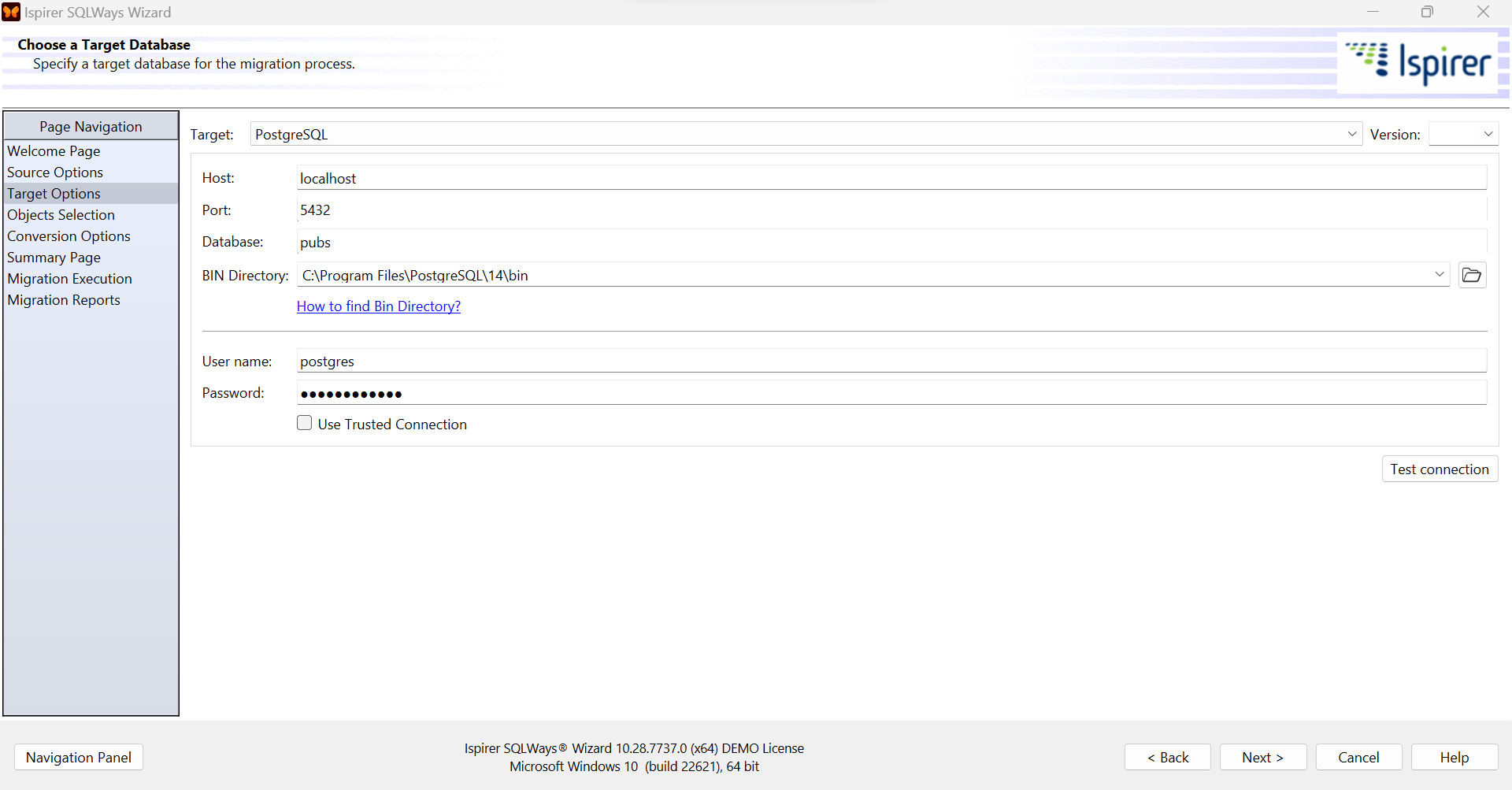
Final objects count in PostgreSQL after migration of northwind database

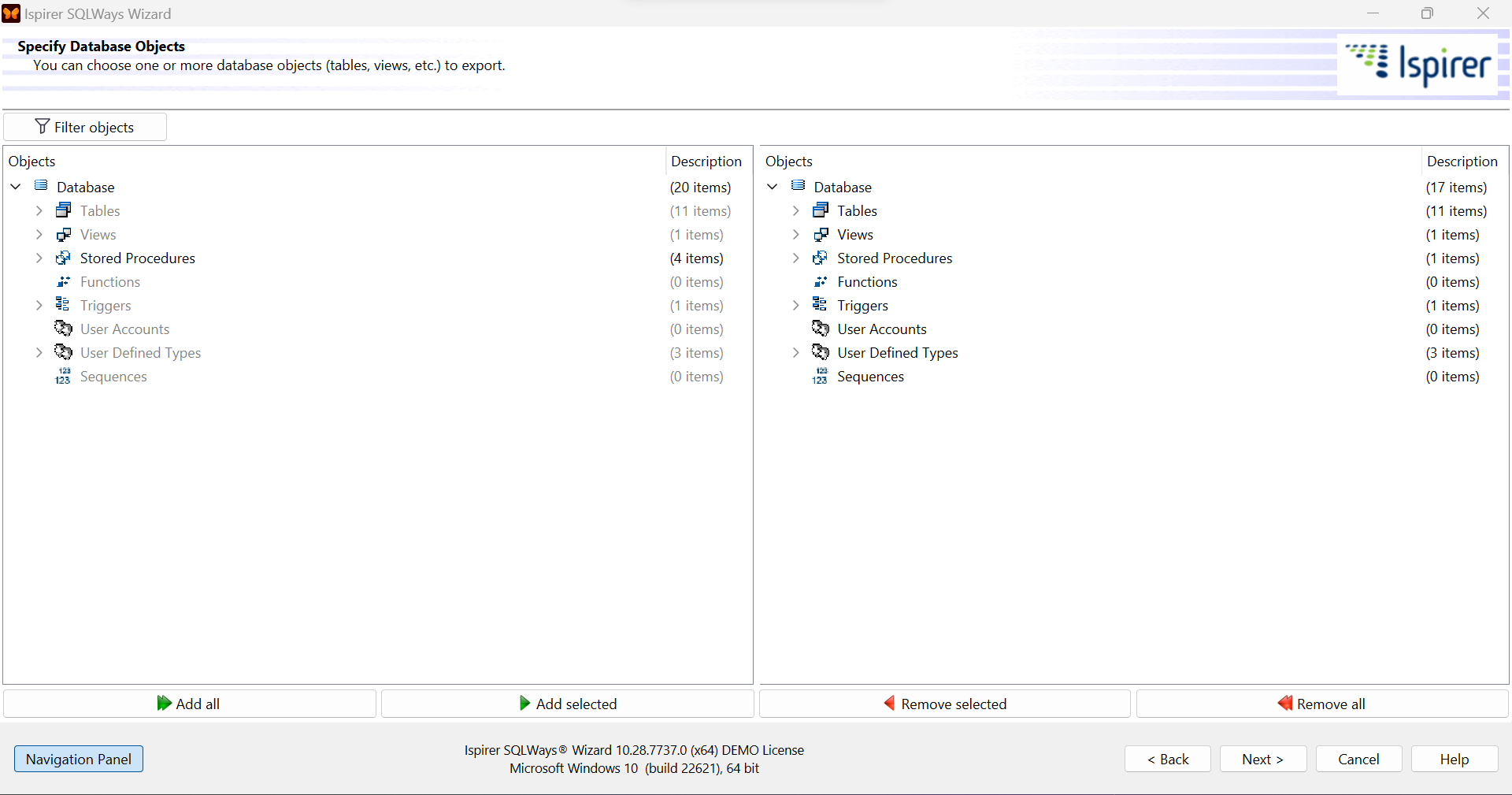


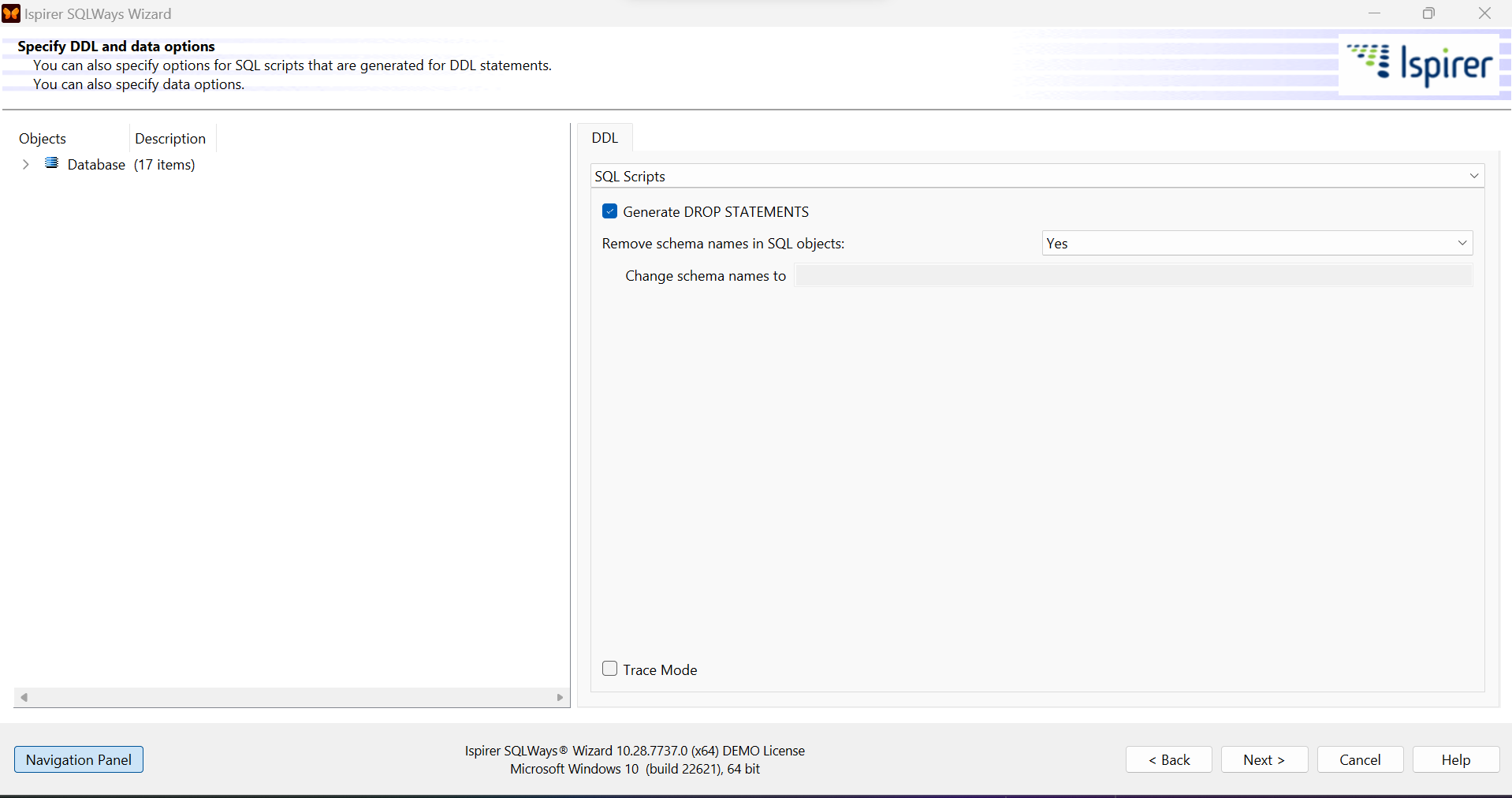
1. We will now follow same steps to migrate pubs database

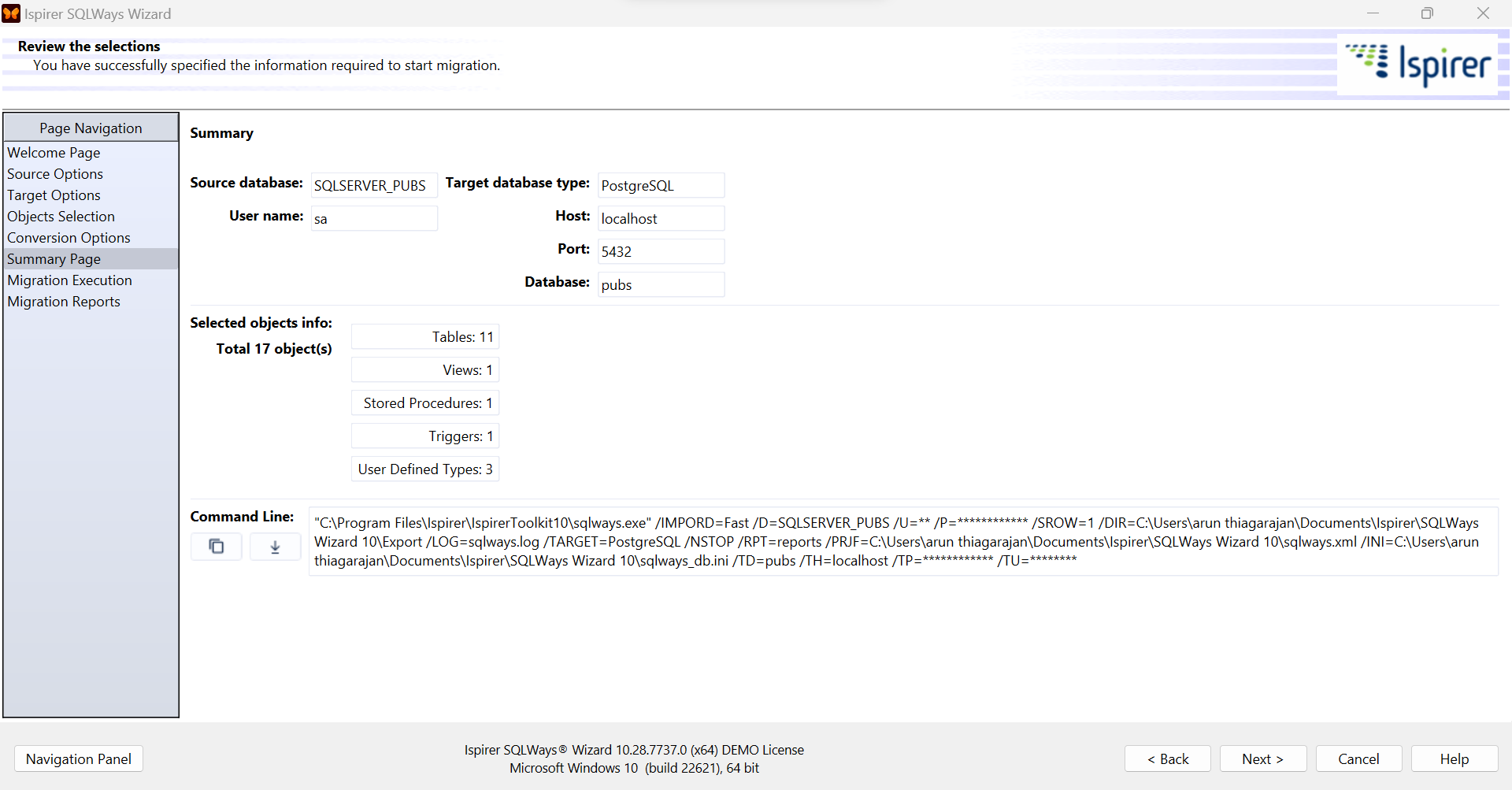
### Pubs

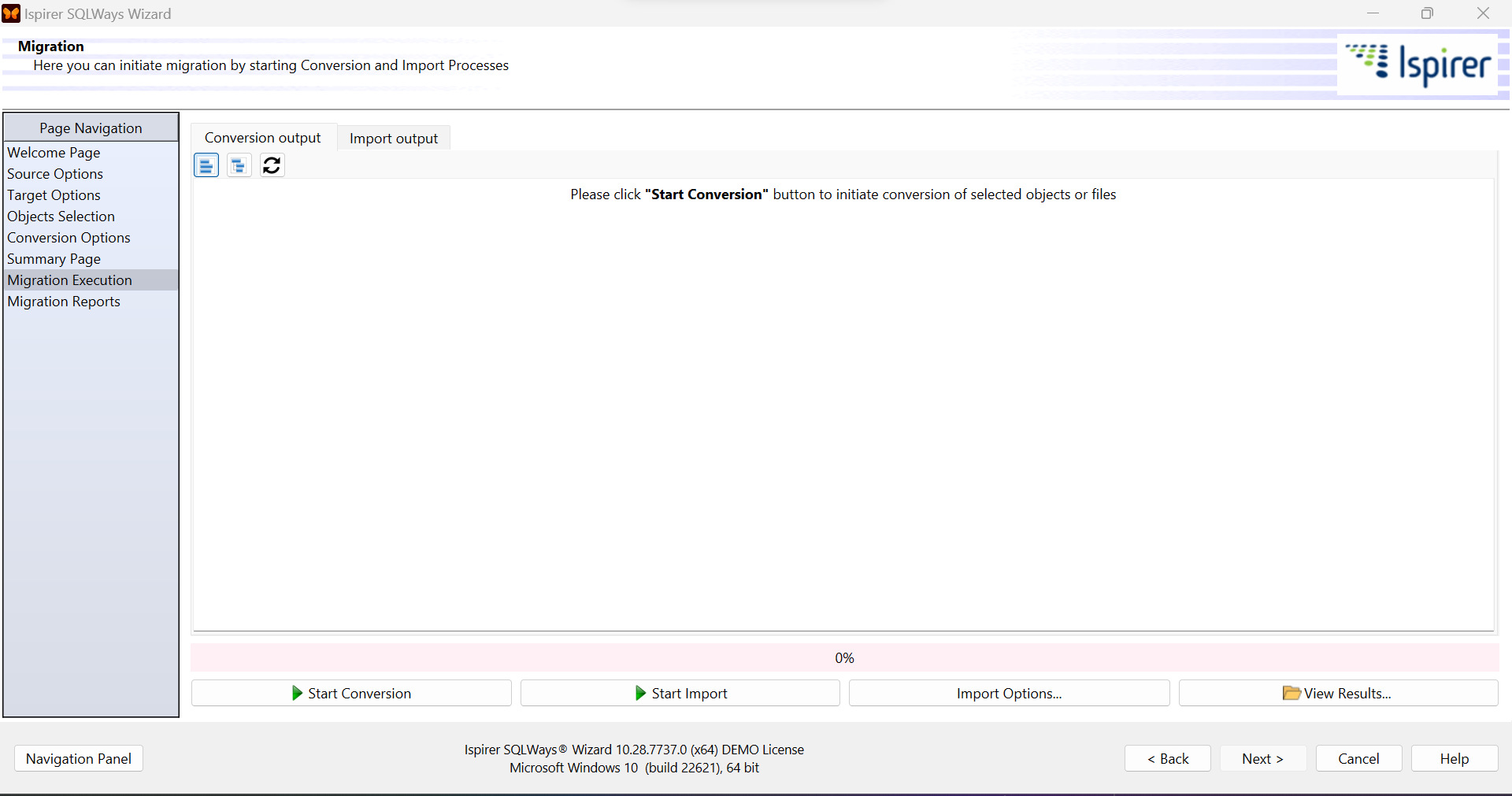




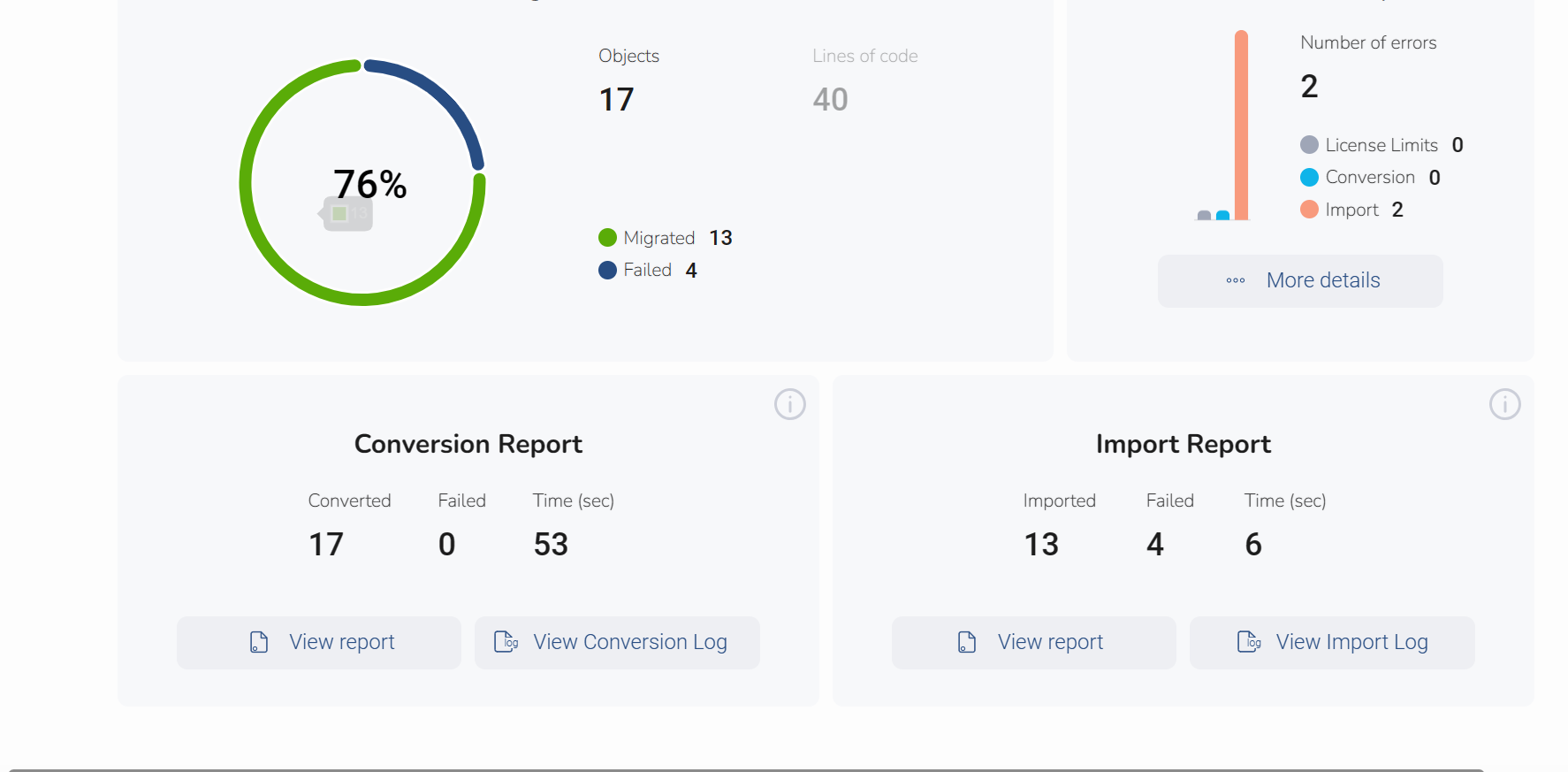


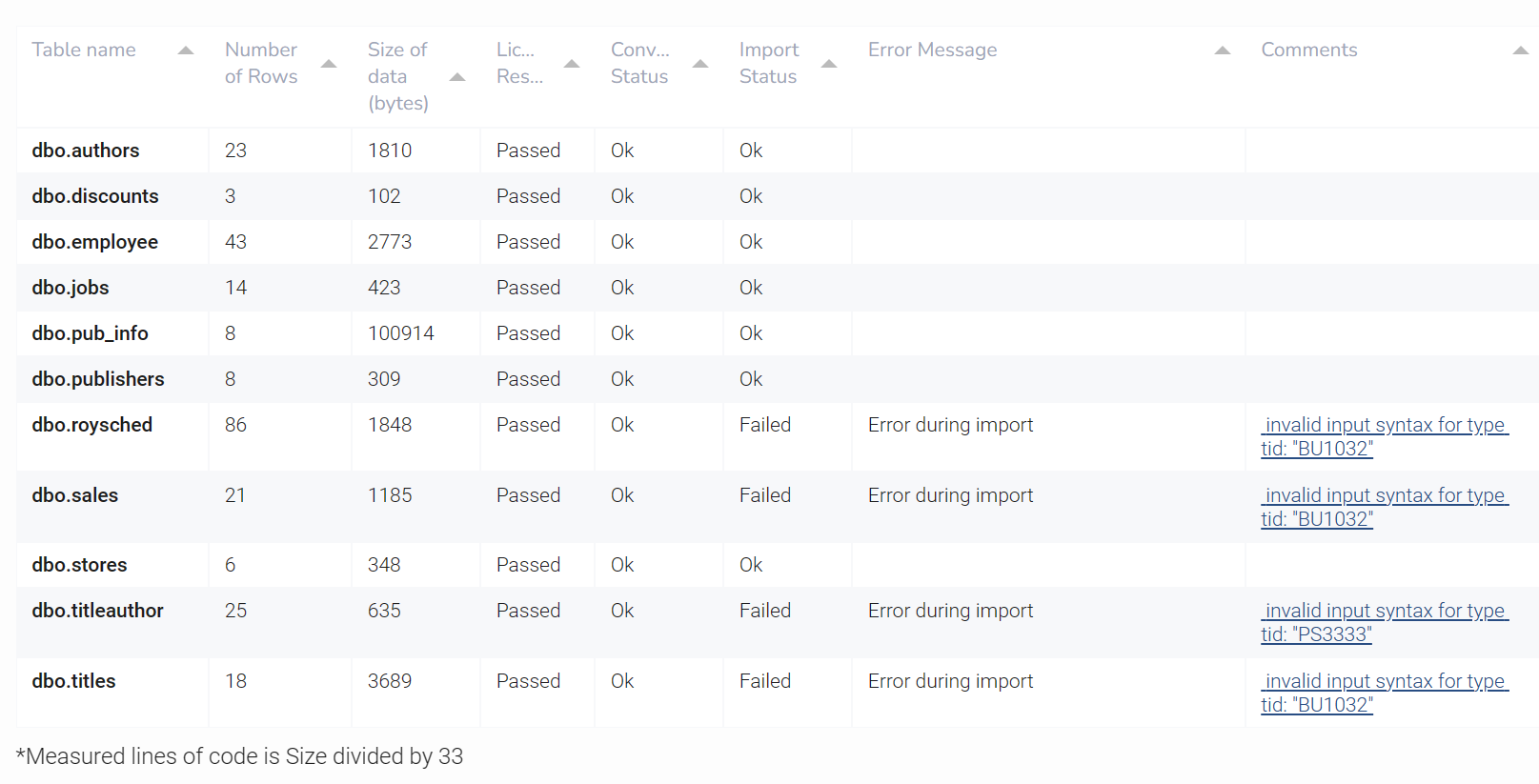






4 tables failed to import





The errors were fixed after migrating User Defined Data Types (UDT’s)

Final view from pgadmin4 after completing migration.

