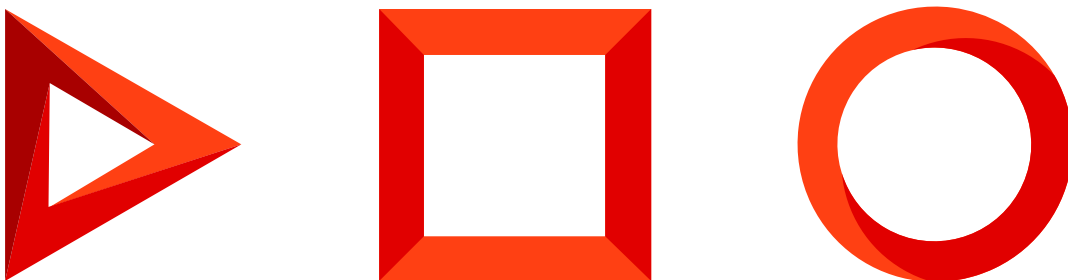


Database server

Version 7.17



This documentation is provided under restrictions on use and are protected by intellectual property laws. Except as expressly permitted in your license agreement or allowed by law, you may not use, copy, reproduce, translate, broadcast, modify, license, transmit, distribute, exhibit, perform, publish or display any part, in any form, or by any means. Reverse engineering, disassembly, or decompilation of this documentation, unless required by law for interoperability, is prohibited.

The information contained herein is subject to change without notice and is not warranted to be error-free. If you find any errors, please report them to us in writing.

Table of Contents

Deploy Microsoft SQL database for Creatio	4
Deploy Oracle Database for Creatio	6
Deploy PostgreSQL database (Linux)	8
I. Install PostgreSQL	8
II. Create PostgreSQL user	9
III. Restore PostgreSQL database	11
IV. Change the database owner (optional)	13
Deploy PostgreSQL database (Windows)	13
I. Install PostgreSQL	14
II. Create PostgreSQL user	14
III. Restore PostgreSQL database	16

Deploy Microsoft SQL database for Creatio

PRODUCTS: **ALL CREATIO PRODUCTS**

Install Microsoft SQL Server Management Studio on the database server. Installation instructions are available in the [Microsoft SQL Server documentation](#).

Note. Microsoft SQL has been tested for deployment of clustered Creatio databases. Using Microsoft SQL Always On availability groups is a recommended method of setting up a high availability configuration. For more information on Microsoft SQL Always On technology, please refer to the [Microsoft SQL documentation](#).

In Microsoft SQL Server Management Studio, create two database users.

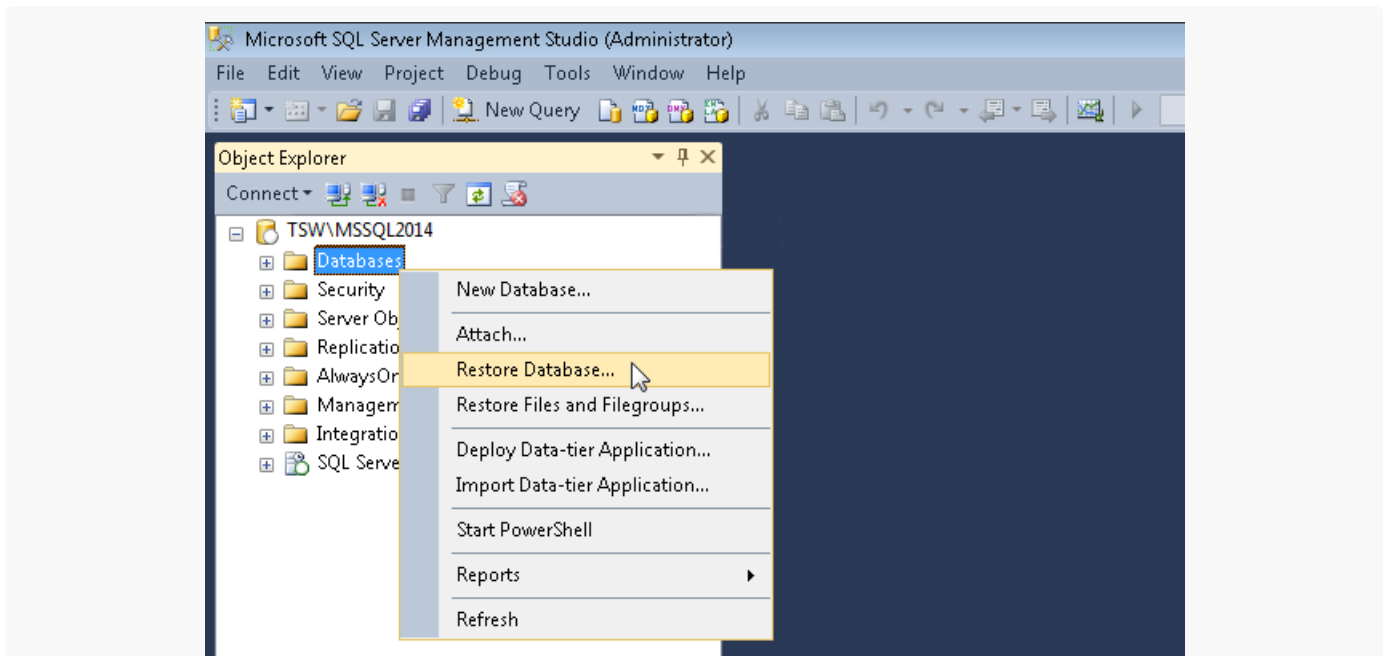
- A user with the “**sysadmin**” role, who has maximum access privileges on the database server level. This user will restore the Creatio database from a backup file and assign access permissions.
- A user with the “**public**” role, whose permissions are limited. You will need this user to set up a secure connection to the restored Creatio database using Microsoft SQL authentication.

For more on creating users and access permissions on the database server, see [Microsoft SQL Server documentation](#).

To restore a database:

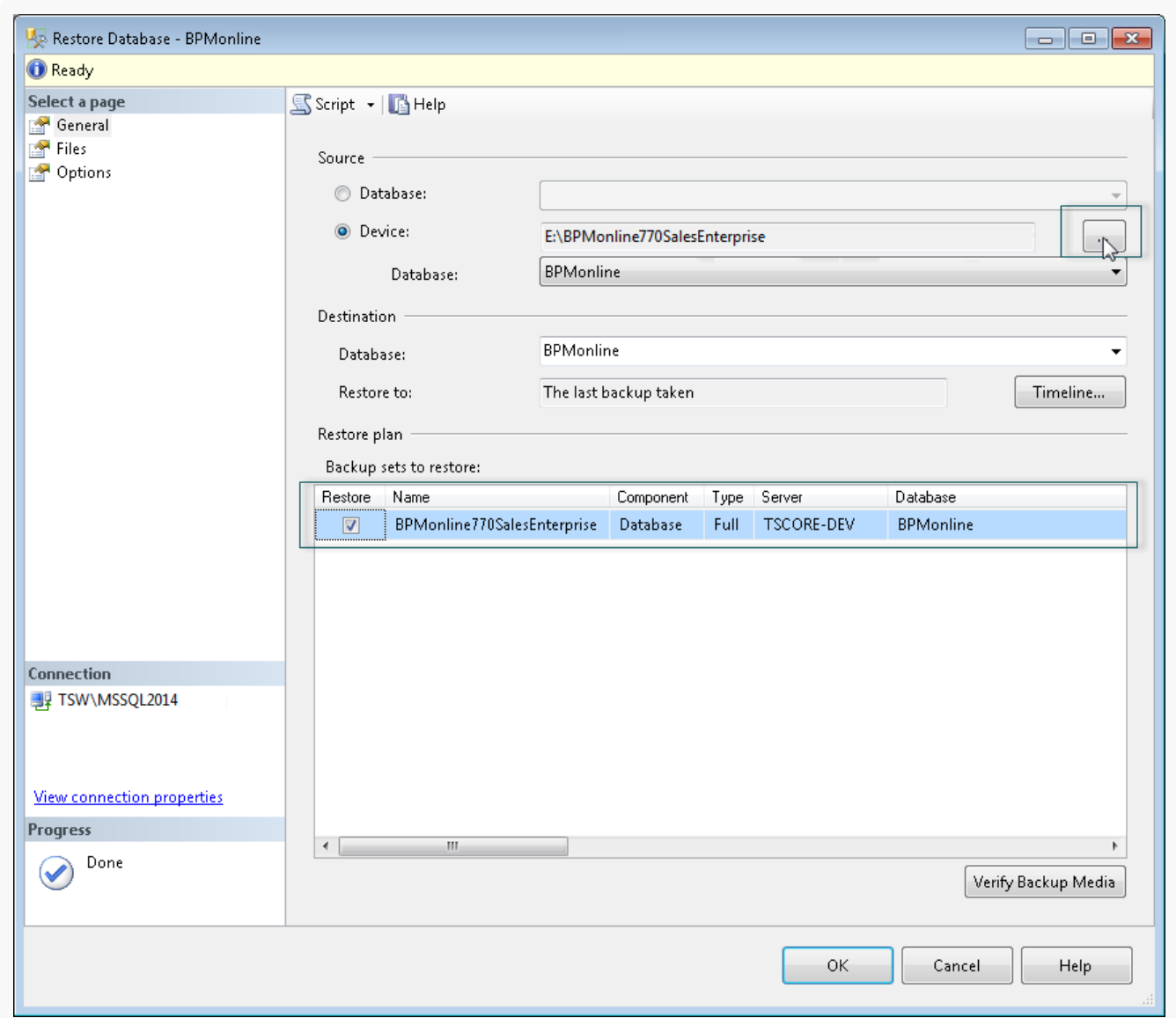
1. Authenticate in Microsoft SQL Server Management Studio as a “**sysadmin**” user.
2. Click the [*Databases*] catalog and select the [*Restore Database*] option from the context menu (Fig. 1).

Fig. 1 Selecting database backup command



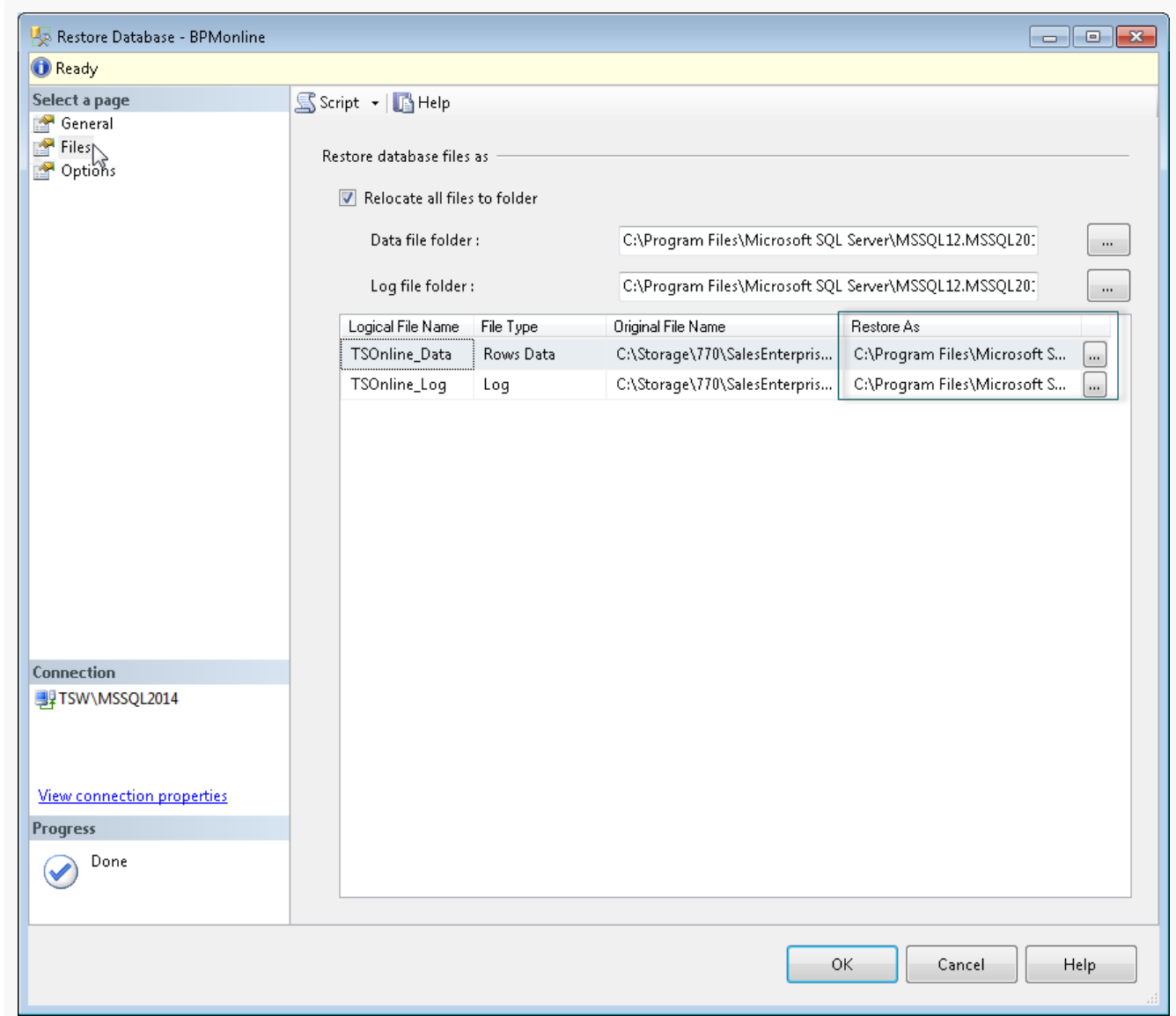
3. In the [*Restore Database*] window:
 - a. Specify the name of the database in the [*Database*] field;
 - b. Specify the [*Device*] checkbox and specify the path to the database backup copy file. The database backup file is supplied together with executable files and is located in the `~\db` folder (Fig. 2).

Fig. 2 Selecting database backup



4. Specify a folder on the database server where the Creatio database will be restored. Creating a folder to restore database files is required beforehand, as the SQL Server is not permitted to create directories.
 - a. Go to the [*Files*] tab.
 - b. In the [*Restore the database files as*] area, select the [*Relocate all files and folders*] checkbox.
 - c. Specify paths to the folders where SQL Management Studio will save the **TS_Data.mdf** and **TS_Log.ldf** files (Fig. 3).

Fig. 3 Specifying the names and paths to TS_Data.mdf and TS_Log.ldf files.



5. Click the [*OK*] button and wait for the database restore process to be finished.
6. Enable connection for the **public** Microsoft SQL user who Creatio will use to access the database.
 - a. Locate the restored Creatio database in Microsoft SQL Server Management Studio.
 - b. Click the [*Security*] tab.
 - c. Add the user to the [*Users*] list.
 - d. Click [*Membership*] and specify the **db_owner**, which will grant the user full access to the restored Creatio database.

Deploy Oracle Database for Creatio

PRODUCTS: [ALL CREATIO PRODUCTS](#)

Install Oracle Database on the database server. Installation instructions are available in the [Oracle Database Online Documentation](#). The “sqlplus” and “impdp” utilities required for restoring the application database from

backup are installed with Oracle Database.

Note. Please refer to the [Oracle Database Online Documentation](#) for more details on Oracle Database clustering.

In Oracle Database, create two database users.

- A user with the “**admin**” role, who has maximum access privileges on the database server level. This user will restore the Creatio database from a backup file and assign access permissions.
- A user with the “**public**” role, whose permissions are limited. You will need this user to set up a secure connection to the restored Creatio database using Oracle authentication.

For more on creating users and access permissions on the database server, see [Oracle Database Online Documentation](#).

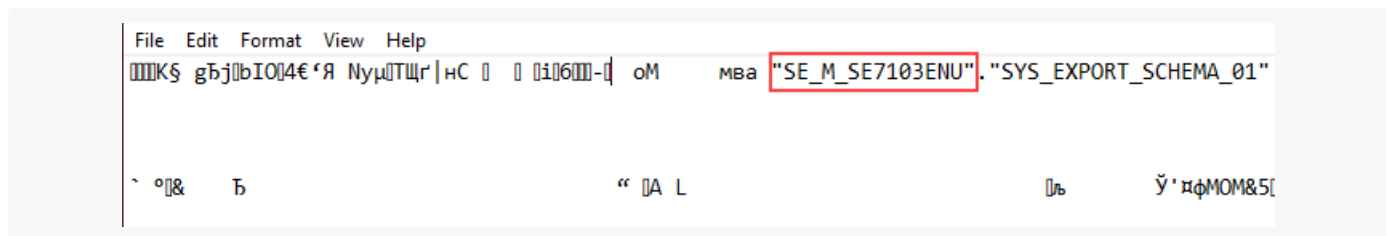
Download and unzip the [archive with the SQL scripts](#) that are used to restore the database from the Oracle backup file.

By default the Oracle DB backup file is located in the ~\db folder with the Creatio executable files. If the backup file is located not on the Oracle server, it should be located in the network folder with general access.

To restore the database:

1. Open the CreateUser.sql and RecompileSchema.sql scripts in the editor and modify the following macros:
 - a. YOUR_SCHEMA_NAME - schema name
 - b. YOUR_SCHEMA_PASSWORD - schema password
 - c. \\your_server.com\Share - path to the backup (*.dmp file).
2. Open the backup file in a text editor, find and save the name of the used schema located before the “.SYS_EXPORT_SCHEMA” record (Fig. 1).

Fig. 1 The schema name in the backup file



3. Remove the modified scripts on the Oracle server. To create a new schema, execute the following command from the folder with the scripts:

```
sqlplus.exe "SYS/SYS_PASSWORD@your_server.com:1521/YOUR_SERVICE_NAME AS SYSDBA" @CreateUser.s
```

1. SYS_PASSWORD - a password for authorization on the Oracle server
2. your_server.com - network address of the Oracle server
3. YOUR_SERVICE_NAME - Oracle service name.

4. Run import of the DB backup copy in the created schema:

```
impdp "YOUR_SCHEMA_NAME/YOUR_SCHEMA_NAME@//your_server.com:1521/BPMBUILD"
  REMAP_SCHEMA=ORIGINAL_SCHEMA_NAME:YOUR_SCHEMA_NAME
  DIRECTORY=BACKUPDIR DUMPFILE=filename.dmp NOLOGFILE=YES
```

1. YOUR_SCHEMA_NAME - the name of the schema specified in the CreateUser.sql
2. your_server.com - network address of the Oracle server
3. ORIGINAL_SCHEMA_NAME - the name of the schema from the backup file (step 2).
4. Consistently run:

```
sqlplus.exe "YOUR_SCHEMA_NAME/YOUR_SCHEMA_PASSWORD@your_server.com:1521/YOUR_SERVICE_NAME"
@tspkg_UtilitiesGlobalTypes.sql
```

```
sqlplus.exe
"YOUR_SCHEMA_NAME/YOUR_SCHEMA_PASSWORD@your_server.com:1521/YOUR_SERVICE_NAME"
@tspkg_UtilitiesGlobalTypes.sql
```

```
sqlplus.exe "YOUR_SCHEMA_NAME/YOUR_SCHEMA_PASSWORD@your_server.com:1521/ YOUR_SERVICE_NAME"
@RecompileSchema.sql
```

Deploy PostgreSQL database (Linux)

PRODUCTS: [ALL CREATIO PRODUCTS](#)

Use one of two database configurations to deploy Creatio:

- Use a remote DBMS (recommended)
- Use a local PostgreSQL server.

If you already have a PostgreSQL server running on the intended machine, skip to step II.

If you have set up sysadmin (with privileges to log in, create and modify databases) and public (unprivileged) user roles, skip to step III.

I. Install PostgreSQL

PostgreSQL is unavailable in most standard repositories. To install PostgreSQL on Linux:

1. Log in as root:

```
sudo su
```


2. Add the PostgreSQL repository:

```
echo -e "deb http://apt.postgresql.org/pub/repos/apt/ $(lsb_release -sc)-pgdg main" > /etc/ap
```

3. Import the signing key of the PostgreSQL repository:

```
wget --quiet -O - https://www.postgresql.org/media/keys/ACCC4CF8.asc | apt-key add -
```

4. Update the package lists:

```
apt-get update
```

5. Install PostgreSQL:

```
apt-get install -y postgresql-12
```

6. Log out from your root session:

```
exit
```

Note. Please refer to the [PostgreSQL documentenation](#) for details on PostgreSQL clustering.

II. Create PostgreSQL user

A fresh installation of PostgreSQL is not ready for deploying Creatio immediately. If you plan to use a fresh installation of PostgreSQL, you need to create a user that can log in to the database using a password and has sufficient privileges to create and update a database. By default, no such user will be available.

We recommend to create two database users in PostgreSQL:

- A user with the “**sysadmin**” role, who has maximum access privileges on the database server level. This user will restore the Creatio database from a backup file and assign access permissions.
- A user with the “**public**” role, whose permissions are limited. You will need this user to set up a secure connection to the restored Creatio database using PostgreSQL authentication.

If your PostgreSQL instance already has sysadmin (privileged) and public (unprivileged) user roles, skip this step.

To create PostgreSQL users:

1. Log in as **postgres**:

```
sudo su - postgres
```

-
2. Open PostgreSQL shell:

```
psql
```

-
-
3. Create a sysadmin user:

```
CREATE USER pg_sysadmin;
```

pg_sysadmin – user who will be granted sysadmin privileges. This user will restore the Creatio database from a backup file and assign access permissions

-
-
-
4. Make **pg_sysadmin** a system administrator:

```
ALTER ROLE pg_sysadmin WITH SUPERUSER;
```

-
-
-
-
5. Allow **pg_sysadmin** to log in:

```
ALTER ROLE pg_sysadmin WITH LOGIN;
```

-
-
-
-
-
6. Set a password for **pg_sysadmin**:

```
ALTER ROLE pg_sysadmin WITH PASSWORD 'pg_syspassword';
```

pg_password – sysadmin user password for connecting to the PostgreSQL server.

-
-
-
-
-
-
7. Create a public user:

```
CREATE USER pg_user;
```

pg_user – public user for connecting to the PostgreSQL server. You will need this user to set up a connection to the restored Creatio database.

-
-
-
-
-
-
8. Allow **pg_user** to log in:

```
ALTER ROLE pg_user WITH LOGIN;
```

-
-
-
-
-
-
-
9. Set a password for **pg_user**:

```
ALTER ROLE pg_user WITH PASSWORD 'pg_password';
```

pg_password - public user password for connecting to the PostgreSQL server.

10. Exit the PostgreSQL shell:

```
\q
```

11. Log out from your postgres session:

```
exit
```

III. Restore PostgreSQL database

To restore a PostgreSQL database from a backup file, you will need **psql** and **pg_restore** utilities. Both are part of the **postgresql-client-common** package.

If you install **postgresql-12** locally using **apt-get**, APT will install **postgresql-client-common** as a dependency for **postgresql-12**.

If you plan to use a remote PostgreSQL database without installing the PostgreSQL DBMS on your server, install the **postgresql-client-common** package manually by running:

```
sudo apt-get install postgresql-client-common
```

To restore the Creatio database from a backup file:

1. Enter DB connection password in the environment variable:

```
export PGPASSWORD=pg_syspassword
```

pg_syspassword - pg_sysadmin user password for connecting to the PostgreSQL server.

2. Create a database where the backup data will be restored:

```
psql --host=pg_server_address --port=pg_server_port --username=pg_sysadmin --dbname=pg_dbname
```

pg_server_address - PostgreSQL server address

pg_server_port - PostgreSQL server port

pg_sysadmin - sysadmin user for connecting to the PostgreSQL server

pg_dbname - name of the PostgreSQL DB where the instructions will be executed

Note. If you have not created any databases yet or an attempt to connect to a database triggers the "FATAL: database "pg_dbname" does not exist" error, use the default database "template1".

pg_dbname_creatio - name of the PostgreSQL DB which will host Creatio tables

pg_user - the "public" user who will be granted permission to use and update the Creatio database

3. If you are using AWS RDS:

- a. Download the [ChangeTypesOwner.sql](#) script.
- b. In the script, replace the "postgres" value with a valid Postgres username.
- c. Run the updated ChangeTypesOwner.sql script.

4. Navigate to the application directory:

```
cd /path/to/application/directory/
```

/path/to/application/directory/ - the directory with Creatio setup files.

5. Navigate to the database directory:

```
cd db
```

6. Restore the database from the backup file:

```
pg_restore --host pg_server_ip --port pg_server_port --username=pg_sysadmin --dbname=pg_dbname
```

pg_server_address - PostgreSQL server address

pg_server_port - PostgreSQL server port

pg_sysadmin - sysadmin user for connecting to the PostgreSQL server

pg_dbname_creatio - name of the PostgreSQL DB to insert backup tables. Use the name you specified in the "CREATE DATABASE" command on step 2.

7. Download the [CreateTypeCastsPostgreSql.sql file](#).

8. Execute type conversion:

```
psql --host=pg_server_address --port=pg_server_port --username=pg_sysadmin --dbname=pg_dbname
```

pg_server_ip - PostgreSQL server address

pg_server_port - PostgreSQL server port

pg_sysadmin - sysadmin user for connecting to the PostgreSQL server

pg_dbname_creatio – name of the PostgreSQL DB where the instructions will be executed

/path/to/CreateTypeCastsPostgreSql.sql – path to the CreateTypeCastsPostgreSql.sql file.

IV. Change the database owner (optional)

Creatio lets you replace the owner of the database and its objects to a non-administrator user (not a superuser) during the restoration. Use the ChangeDbObjectsOwner script for that. For Postgres version 10 and earlier: [Download the script](#). For Postgres version 11 and later: [Download the script](#).

To restore the database on behalf of a non-administrator user:

1. Replace the database owner:

```
psql --host pg_server_address --port pg_server_port --username=pg_sysadmin --dbname=pg_dbname
```

pg_server_address – PostgreSQL server address

pg_server_port – PostgreSQL server port

pg_sysadmin – sysadmin user for connecting to the PostgreSQL server This user must be an administrator (superuser) or have the “ALTER DATABASE” privileges.

pg_user – the placeholder to replace with the actual username of the new database owner. You will need this user to set up a connection to the Creatio database.

pg_dbname_creatio – the name of the database whose owner is being changed.

2. Replace the owner of the database objects:

```
psql --host pg_server_address --port pg_server_port --username=pg_sysadmin --dbname=pg_dbname
```

pg_server_address – PostgreSQL server address

pg_server_port – PostgreSQL server port

pg_sysadmin – sysadmin user for connecting to the PostgreSQL server This user must be an administrator (superuser) or the Creatio database owner.

pg_user – the placeholder to replace with the actual username of the new database owner. You will need this user to set up a connection to the Creatio database.

pg_dbname_creatio – the name of the database whose owner is being changed.

/path/toChangeDbObjectsOwner.sql – the path to the previously saved toChangeDbObjectsOwner.sql file.

You can ignore this step. In that case, the user who ran the pg_restore command will remain the owner of the database and its objects. Normally, this is the postgres user.

Deploy PostgreSQL database (Windows)

PRODUCTS: [ALL CREATIO PRODUCTS](#)

Use one of two database configurations to deploy Creatio:

- Use a remote DBMS (recommended)

- Use a local PostgreSQL server.

If you already have a PostgreSQL server set up, skip to step II.

If you have already set up `sysadmin` (with privileges to log in, create and modify databases) and `public` (unprivileged) user roles, skip to step III.

I. Install PostgreSQL

PostgreSQL setup files are available for download at [postgresql.org](https://www.postgresql.org).

Note. High-availability PostgreSQL configurations have not been tested with Creatio. Please refer to the [PostgreSQL documentation](#) for details on PostgreSQL clustering.

II. Create PostgreSQL user

A fresh installation of PostgreSQL Server is not ready for deploying Creatio immediately. If you plan to use a fresh installation of PostgreSQL Server, you need to create a user that can log in to the database using a password and has sufficient privileges to create and update a database. By default, no such user will be available.

We recommend to create two database users in PostgreSQL:

- A user with the “**sysadmin**” role who has maximum access privileges on the database server level. This user will restore the Creatio database from a backup file and assign access permissions. These instructions use **pg_sysadmin** as a placeholder username, but you can set the username to any value.
- A user with the “**public**” role whose permissions are limited. You will need this user to set up a secure connection to the restored Creatio database using PostgreSQL authentication. These instructions use **pg_user** as a placeholder username, but you can set the username to any value.

To create the two PostgreSQL users:

1. Open the Command Prompt.
2. Navigate to the PostgreSQL software install folder:

```
cd /D "\\path\to\PostgreSQL\folder"
```

- **\\path\to\PostgreSQL\folder** - the path to the PostgreSQL software install folder.

3. Navigate to the folder with the Command Line Tools component:

```
cd bin
```

4. Enter the DB connection password in the environment variable.

```
set PGPASSWORD=pg_password
```

- **pg_password** – password of the **postgres** user for connecting to the PostgreSQL server.

5. Run PostgreSQL shell as **postgres**:

```
psql.exe --username postgres
```

6. Create a sysadmin user, e. g. **pg_sysadmin**:

```
CREATE USER pg_sysadmin;
```

- **pg_sysadmin** – placeholder name for a sysadmin user. The sysadmin will restore the Creatio database from a backup file and assign access permissions.

7. Make **pg_sysadmin** a system administrator:

```
ALTER ROLE pg_sysadmin WITH SUPERUSER;
```

8. Allow **pg_sysadmin** to log in:

```
ALTER ROLE pg_sysadmin WITH LOGIN;
```

9. Set a password for **pg_sysadmin**:

```
ALTER ROLE pg_sysadmin WITH PASSWORD 'pg_syspassword';
```

- **pg_syspassword** – sysadmin user password for connecting to the PostgreSQL server.

10. Create a public user, e. g. **pg_user**:

```
CREATE USER pg_user;
```

- **pg_user** – placeholder name for a public user. This user will set up a connection to the restored Creatio database.

11. Allow **pg_user** to log in:

```
ALTER ROLE pg_user WITH LOGIN;
```

12. Set a password for **pg_user**:

```
ALTER ROLE pg_user WITH PASSWORD 'pg_password';
```

- **pg_password** – public user password for connecting to the PostgreSQL server.

13.Exit the PostgreSQL shell:

```
\q
```

III. Restore PostgreSQL database

To restore a PostgreSQL database from a backup file, you will need **psql.exe** and **pg_restore.exe** utilities. Both are part of the Command Line Tools PostgreSQL component that comes with the PostgreSQL Server. They are located in the PostgreSQL software install folder.

If you plan to use a remote PostgreSQL database without installing the PostgreSQL Server on your machine, take the following steps:

1. Get a PostgreSQL binary package. Binary packages are available for download at [postgresql.org](https://www.postgresql.org).
2. Select the Command Line Tools component during installation. Selecting the other components is optional.

To restore the Creatio database from a backup file:

1. Open Command Prompt.
2. Navigate to the PostgreSQL software install folder:

```
cd /D "\\path\to\PostgreSQL\folder"
```

- **\\path\to\PostgreSQL\folder** – the path to the PostgreSQL software install folder.

3. Navigate to the folder with executables:

```
cd bin
```

4. Enter the DB connection password in the environment variable:

```
set PGPASSWORD=pg_syspassword
```

- **pg_syspassword** – sysadmin user password for connecting to the PostgreSQL server.

5. Create a database where the backup data will be restored.

For Creatio version 7.16.3 or higher:


```
psql.exe --host pg_server_ip --port pg_server_port --username=pg_sysadmin --command "CREATE D
```

- **pg_server_ip** - PostgreSQL server address.
- **pg_server_port** - PostgreSQL server port.
- **pg_sysadmin** - user for connecting to the PostgreSQL server. The user must have either superuser (administrator) privileges or “CREATE DATABASE” privileges.
- **pg_user** - the application will use this user's credentials to connect to the database. You can specify any user when creating the database. To change the user data, follow **step 10** of this instruction.

For Creatio version 7.16.0 - 7.16.2:

```
psql.exe --host pg_server_address --port pg_server_port --username=pg_sysadmin --command "CRE
```

- **pg_server_address** - PostgreSQL server address.
- **pg_server_port** - PostgreSQL server port.
- **pg_sysadmin** - user for connecting to the PostgreSQL server. The user must have either superuser (administrator) privileges or “CREATE DATABASE” privileges.
- **pg_user** - the “public” user who will be granted permission to use and update the Creatio database

6. If you are using AWS RDS:

- Download the [ChangeTypesOwner.sql](#) script.
- In the script, replace the “postgres” value with a valid Postgres username.
- Run the updated ChangeTypesOwner.sql script.

7. Restore the Creatio database from the backup file:

For Creatio version 7.16.3 or higher:

```
pg_restore --host pg_server_ip --port pg_server_port --username=pg_sysadmin --dbname=pg_dbnam
```

- **pg_server_ip** - PostgreSQL server address.
- **pg_server_port** - PostgreSQL server port.
- **pg_sysadmin** - user for connecting to the PostgreSQL server. The user must have either superuser (administrator) privileges or sufficient access permissions to run the **pg_restore** utility.
- **pg_dbname_creatio** - name of the PostgreSQL DB to insert backup tables.

For Creatio version 7.16.0 - 7.16.2:

```
pg_restore.exe --host pg_server_address --port pg_server_port --username=pg_sysadmin --dbname
```

- **pg_server_address** - PostgreSQL server address.
- **pg_server_port** - PostgreSQL server port.
- **pg_sysadmin** - user for connecting to the PostgreSQL server. The user must have either superuser (administrator) privileges or sufficient access permissions to run the pg_restore utility.
- **pg_dbname_creatio** - name of the PostgreSQL DB to insert backup tables.

8. Download the [CreateTypeCastsPostgreSql.sql file](#).

9. Execute type conversion:

```
psql.exe --host pg_server_ip --port pg_server_port --username=pg_sysadmin --dbname=pg_dbname_
```

- **pg_server_ip** - PostgreSQL server address.
- **pg_server_port** - PostgreSQL server port.
- **pg_sysadmin** - user with administrator privileges for connecting to the PostgreSQL server.
- **pg_dbname_creatio** - name of the PostgreSQL DB where the instructions will be executed.
- **\\path\to\CreateTypeCastsPostgreSql.sql** - path to the downloaded CreateTypeCastsPostgreSql.sql file.

10. Creatio version **7.16.3** supports changing the owner of the database and database objects to a non-administrator user (i. e. not a superuser). To do this, use the ChangeDbObjectsOwner script. [Download the script](#).

To restore the database from a backup as a regular user:

a. Change the owner of the database:

```
psql.exe --host pg_server_ip --port pg_server_port --username=pg_sysadmin --dbname=pg_dbname_
```

- **pg_server_ip** - PostgreSQL server address.
- **pg_server_port** - PostgreSQL server port.
- **pg_sysadmin** - user for connecting to the PostgreSQL server. The user must have either administrator (superuser) privileges or "CREATE DATABASE" privileges.
- **pg_user** - new database owner.
- **pg_dbname_creatio** - the name of the database whose owner is changed.

g. Change the owner of the database objects:

```
psql.exe --host pg_server_ip --port pg_server_port --username=pg_sysadmin --dbname=pg_dbname_
```

- **pg_server_ip** - PostgreSQL server address.
- **pg_server_port** - PostgreSQL server port.

- **pg_sysadmin** – user for connecting to the PostgreSQL server. The user must have either administrator (superuser) privileges or “CREATE DATABASE” privileges.
- **pg_user** – new database owner.
- **pg_dbname_creatio** – the name of the database whose owner is changed.
- **\\path\to\ChangeDbObjectsOwner.sql** – path to the downloaded ChangeDbObjectsOwner.sql file.

Skip this step to leave the default owner of the database and database objects, which is the user who runs the **pg_restore** utility (usually **postgres**)