

DBSeeder - Relational Database Data Generator.

build passing release v3.0.2 release date last monday github repo or version not found

Table of Contents

- 1. Introduction**
 - 1.1 RDBMS Overview**
 - 1.2 RDBMS Directory**
 - 1.3 Performance Example**
- 2. Data**
 - 2.1 Database Schema**
 - 2.2 Construction of the Dummy Data Content**
- 3. Installation**
- 4. Operating Instructions**
 - 4.1 Scripts**
 - 4.2 Operation Possibilities**
 - 4.3 Control Parameters**
 - 4.4 Statistics**
- 5. RDBMS Specific Technical Details**
- 6. trino**

1. Introduction

DBSeeder allows the flexible generation of large amounts of anonymised random dummy data for selected relational database systems (RDBMS) - useful e.g. for stress testing.

The database schema underlying the data generation can be freely defined. The names of the database, the schema and the user can be freely chosen, unless the respective database management system contains restrictions. If the selected database, schema or user already exist, they are deleted with all including data. **DBSeeder** then creates the selected database, schema or user and generates the desired dummy data. A maximum of 2 147 483 647 rows can be generated per database table. The database schema to be used, that is, the required database tables can be user defined using a JSON file. Details can be found here: [Database Schema](#).

Currently, depending on the capabilities of the specific RDBMS, the following functionalities and data types are supported:

- constraints
 - foreign (referential) key
 - not null constraint
 - primary key
 - unique (alternate) key
- data types
 - BIGINT - large integer
 - BLOB - large binary object
 - CLOB - large character Object
 - TIMESTAMP - timestamp including date
 - VARCHAR - variable text

The database systems considered meet the following conditions:

1. The database system is freely available in a documented docker image for testing purposes.
2. The database system provides a well documented JDBC interface.
3. A complete documentation of the SQL commands is available.

1.1 RDBMS Overview

RDBMS	Ticker Symbol(s)	RDBMS Versions	Latest JDBC
AgensGraph	agens	v2.1.1 - v2.1.3	1.4.2-c1
Apache Derby	derby, derby_emb	10.15.2.0	10.15.2.0
CockroachDB	cockroach	v20.2.5 - v21.1.8	see PostgreSQL

RDBMS	Ticker Symbol(s)	RDBMS Versions	Latest JDBC
CrateDB	cratedb	4.1.6 - 4.6.1	2.6.0
CUBRID	cubrid	10.2 - 11.0	11.0.1.0291
Exasol	exasol	6.2.8-d1 - 7.1.0-d1	7.1.0
Firebird	firebird	3.0.5 - v4.0.0	4.0.3.java11
H2 Database Engine	h2, h2_emb	1.4.200	1.4.200
HSQLDB	hsqldb, hsqldb_emb	2.5.1 - 2.6.0	2.6.0
IBM Db2 Database	ibmdb2	11.5.1.0 - 11.5.5.1	11.5.6.0
IBM Informix	informix	14.10 FC3DE - 14.10.FC5DE-rhm	4.50.4.1
MariaDB Server	mariadb	10.4.13 - 10.6.4-focal	2.7.4
Mimer SQL	mimer	v11.0.3c - v11.0.5a	3.41a
MonetDB	monetdb	Jun2020-SP1 - Jul2021	3.1.jre8
MySQL Database	mysql	8.0.20 - 8.0.26	8.0.26
OmniSciDB	omnisci	5.6.1 - 5.7.0	5.7.0
Oracle Database	oracle	12.1.0.2 - 21.3.0	21.3.0.0
Percona Server for MySQL	percona	8.0.25-15	see MySQL
PostgreSQL	postgresql	12.3 - 13.4-alpine	42.2.23
SQL Server	sqlserver	2019-CU12-ubuntu-20.04	9.4.0.jre16
SQLite	sqlite	3.32.0 - 3.36.0	3.36.0.3
TimescaleDB	timescale	2.3.1-pg13 - 2.4.1-pg13	see PostgreSQL
trino	mysql_trino,	339 - 361	361
	oracle_trino,		
	postgresql_trino,		
	sqlserver_trino		
VoltDB	voltodb	9.2.1	11.0
YugabyteDB	yugabyte	2.2.2.0-b15 - 2.9.0.0-b4	42.2.7-yb-3

1.2 RDBMS Directory

The following database systems are included in the current version of **DBSeeder**:

- [AgensGraph](#)
 - client only version
 - commercial, open source
 - derived from PostgreSQL
 - property graph model and relational model
 - [see technical details here](#)
- [Apache Derby](#)
 - client and embedded version
 - open source
 - relational model
 - [see technical details here](#)
- [CockroachDB](#)
 - client only version
 - commercial, open source
 - compatible with PostgreSQL JDBC
 - relational model
 - [see technical details here](#)
- [CrateDB](#)

- client only version
 - commercial, open source
 - compatible with PostgreSQL
 - relational model
 - [see technical details here](#)
- [CUBRID](#)
 - client only version
 - compatible with MySQL
 - open source
 - relational model
 - [see technical details here](#)
- [Exasol](#)
 - client only version
 - commercial
 - in-memory, column-oriented, relational model
 - [see technical details here](#)
- [Firebird](#)
 - client and embedded (not supported here) version
 - open source
 - relational model
 - [see technical details here](#)
- [H2 Database Engine](#)
 - client and embedded version
 - compatible with HSQLDB, PostgreSQL
 - open source
 - relational model
 - [see technical details here](#)
- [HSQLDB](#)
 - client and embedded version
 - open source
 - relational model
 - [see technical details here](#)
- [IBM Db2 Database](#)
 - client only version
 - commercial
 - relational model
 - [see technical details here](#)
- [IBM Informix](#)
 - client only version
 - commercial
 - relational model
 - [see technical details here](#)
- [MariaDB Server](#)
 - client only version
 - open source
 - derived from MySQL
 - relational model
 - [see technical details here](#)
- [Mimer SQL](#)
 - client only version
 - commercial
 - relational model
 - [see technical details here](#)
- [MonetDB](#)
 - client only version
 - open source
 - column-oriented relational model
 - [see technical details here](#)
- [MySQL Database](#)
 - client only version
 - open source

- relational model
 - [see technical details here](#)
- [OmniSciDB](#)
 - client only version
 - commercial, open source
 - GPU and CPU version
 - relational model
 - [see technical details here](#)
- [Oracle Database](#)
 - client only version
 - commercial
 - relational model
 - [see technical details here](#)
- [Percona Server for MySQL](#)
 - client only version
 - commercial, open source
 - derived from MySQL
 - relational model
 - [see technical details here](#)
- [PostgreSQL](#)
 - client only version
 - open source
 - relational model
 - [see technical details here](#)
- [SQL Server](#)
 - client only version
 - commercial
 - derived from Adaptive Server Enterprise
 - relational model
 - [see technical details here](#)
- [SQLite](#)
 - commercial, open source
 - embedded only version
 - relational model
 - [see technical details here](#)
- [TimescaleDB](#)
 - client only version
 - commercial, open source
 - derived from PostgreSQL
 - relational model
 - [see technical details here](#)
- [trino](#)
 - compatible with Accumulo, Cassandra, Elasticsearch, Hive, Kudu, MongoDB, MySQL, Pinot, PostgreSQL, Redis, Redshift
 - distributed query engine
 - open source
 - [see technical details here](#)

For the RDBMS MySQL, Oracle, PostgreSQL and SQL Server the JDBC driver from trino can optionally be used instead of the original JDBC driver. The prerequisite for this is that trino is either installed locally (Linux) or is available as a Docker container (Linux and Windows). Details can be found here: [trino](#).

- [VoltDB](#)
 - client only version
 - commercial, open source
 - derived from H-Store, HSQLDB
 - in-memory relational model
 - [see technical details here](#)
- [YugabyteDB](#)
 - client only version
 - commercial, open source
 - compatible with Cassandra, PostgreSQL, Redis
 - derived from PostgreSQL, RocksDB

- inspired by Cloud Spanner
- relational model
- [see technical details here](#)

2. Data

2.1 Database Schema

The underlying database schema is defined in a JSON-based parameter file and the associated program code is generated and compiled with the script `scripts/run_db_seeder_generate_schema`. To validate the database schema in the JSON parameter file, the JSON schema file `db_seeder_schema.schema.json` in the directory `src/main/resources` is used.

2.1.1 Structure of the Database Schema Definition File

The definition of a database schema consists of the object `global` with the global parameters and the array `tables`, which contains the definition of the database tables.

2.1.1.1 `globals` - Global Parameters

- `defaultNumberOfRows` - default value for the number of table rows to be generated, if no value is specified in the table definition
- `encodingISO_8859_1` - a string with Western Latin characters is inserted into generated character columns
- `encodingUTF_8` - a string with simplified Chinese characters is inserted into generated character columns specified in the table definition
- `nullFactor` - determines the proportion of NULL values in optional columns and must be between 2 and 99 (inclusive): 2 means 50%, 4 means 25%, 10 means 10%, etc., default value is 4

2.1.1.2 `tables` - Database Table Definitions

- `tableName` - database table name
- `numberOfRows` - number of table rows to be generated
- `columns` - an array of column definitions
 - `columnName` - column name
 - `dataType` - data type, is one of BIGINT, BLOB, CLOB, TIMESTAMP or VARCHAR
 - `size` - for data type VARCHAR the maximum size of the column value
 - `precision` - currently not used
 - `notNull` - is a NULL value allowed ?
 - `primaryKey` - is this the primary key column ?
 - `references` - an array of foreign key definitions
 - `referenceTable` - name of the reference database table
 - `referenceColumn` - name of the reference column
 - `defaultValueInteger` - default value for integer columns
 - `defaultValueString` - default value for alphanumeric columns
 - `lowerRangeInteger` - lower limit for an integer column, requires also an upper limit
 - `lowerRangeString` - lower limit for an alphanumeric column, requires also an upper limit
 - `upperRangeInteger` - upper limit for an integer column
 - `upperRangeString` - upper limit for an alphanumeric column
 - `validValuesInteger` - valid values for an integer column
 - `validValuesString` - valid values for an alphanumeric column
- `tableConstraints` - an array of table constraint definitions
 - `constraintType` - constraint type, is one of FOREIGN, PRIMARY or UNIQUE
 - `columns` - an array with the names of the affected columns
 - `referenceTable` - name of the reference database table, only for foreign keys
 - `referenceColumns` - an array with the names of the affected reference columns, only for foreign keys

Only either a range restriction (`lowerRange...`, `upperRange...`) or a value restriction (`validValues...`) may be specified for each column.

2.1.2 Mapping of Data Types in the JDBC Driver

Data Type	JDBC Method
BIGINT	<code>setLong</code>
BLOB	<code>setBytes</code>
CLOB	<code>setString</code>

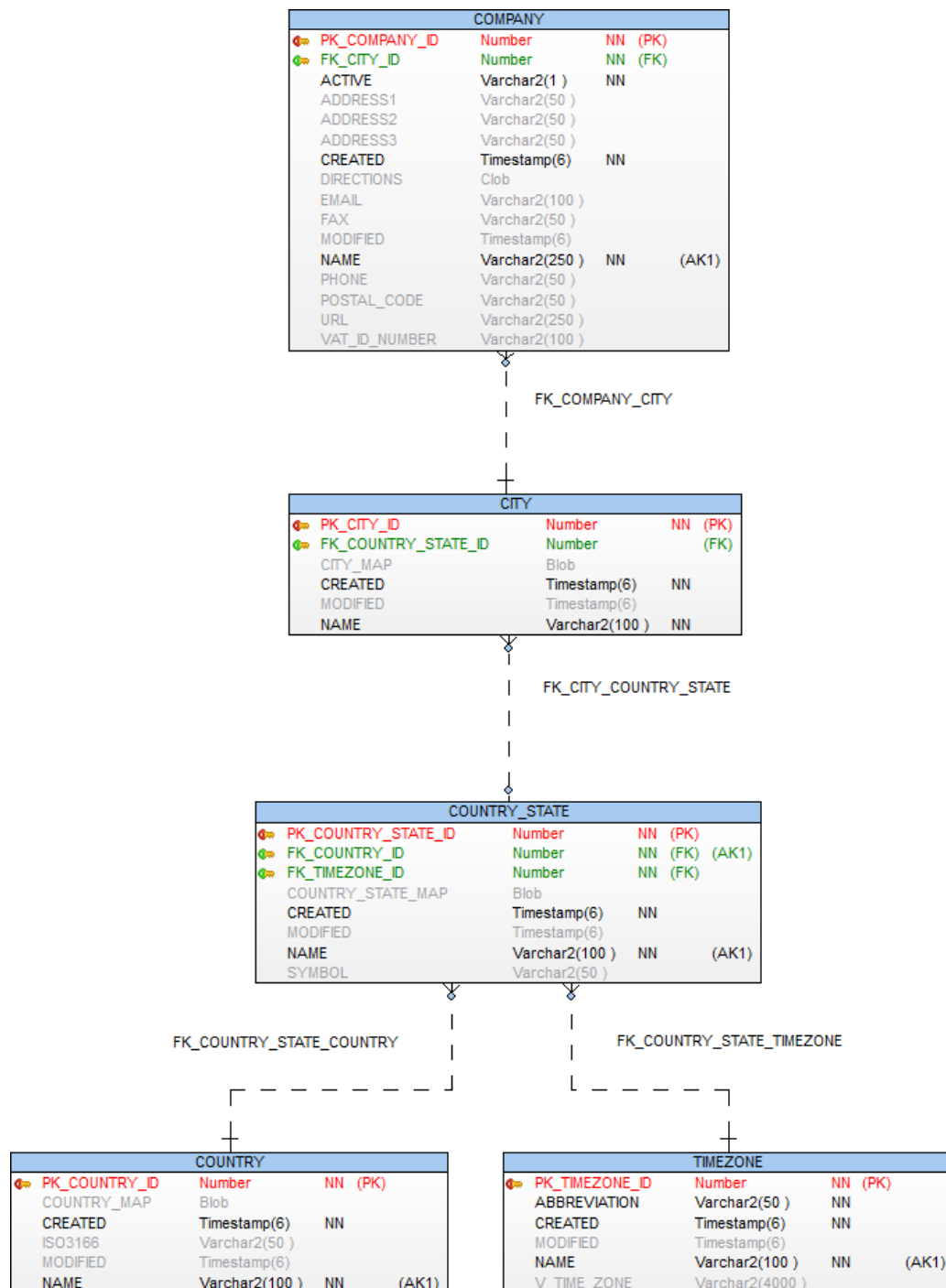
Data Type	JDBC Method
TIMESTAMP	setTimestamp
VARCHAR	setNString (Firebird, MariaDB, MS SQL SERVER and Oracle)
	setString (else)

2.1.3 Example File `db_seeder_schema.company_9...9.json` in the Directory `resources/json`

This file contains the definition of a simple database schema consisting of the database tables CITY, COMPANY, COUNTRY, COUNTRY_STATE and TIMEZONE.

The abbreviations in the following illustration (created with Toad Data Modeler) mean:

- (AK1) - alternate key (unique key)
- FK - foreign key
- NN - not null
- PK - primary key



2.2 Construction of the Dummy Data Content

The proportion of **NULL** values in optional columns is defined by the global parameter **nullFactor**.

All methods for generating column contents can be overwritten if necessary.

2.2.1 BIGINT

Java method: **getContentBigint**

- If the column parameter **validValuesInteger** is defined in the database schema, a random value is taken from it.
- If the column parameters **lowerRangeInteger** and **upperRangeInteger** are defined in the database schema, a random value is taken from this interval.
- Otherwise the counter for the current row (row number) is used.

2.2.2 BLOB

Java method: **getContentBlob**

- The content of the file **blob.png** from the resource directory (**src/main/resources**) is loaded into these columns. This file contains the company logo of Konnexions GmbH.

2.2.3 CLOB

Java method: **getContentClob**

- The content of the file **clob.md** from the resource directory (**src/main/resources**) is loaded into these columns. This file contains the text of the Konnexions Public License (KX-PL).

2.2.4 TIMESTAMP

Java method: **getContentTimestamp**

- A randomly generated timestamp is assigned to all columns that can contain temporal data.


2.2.5 VARCHAR

Java method: **getContentVarchar**


- If the column parameter **validValuesString** is defined in the database schema, a random value is taken from it.
- If the column parameters **lowerRangeString** and **upperRangeString** are defined in the database schema, a random value is taken from this interval.
- Otherwise content of the column is constructed depending on the row number and the encoding flags as follows:
 - ASCII (all rows where the index modulo 3 is 0):
 - column name in capital letters
 - underscore **_**
 - current row number left-justified
 - ISO 8859 1 (all rows where the index modulo 3 is 1) :
 - column name in capital letters
 - underscore **_**
 - a string containing specific Western European characters with accent (e.g. French, Portuguese or Spanish)
 - underscore **_**
 - current row number left-justified
 - the ISO 8859 1 version can be prevented by choosing **encodingISO_8859_1 false** in the database schema definition
 - UTF-8 (all rows where the index modulo 3 is 2):
 - column name in capital letters
 - underscore **_**
 - a string containing simplified Chinese characters
 - underscore **_**
 - current row number left-justified
 - the UTF-8 version can be prevented by choosing **encodingUTF_8 false** in the database schema definition
 - If the resulting value exceeds the permissible column size, the value is shortened accordingly from the left

2.2.6 Examples

1. Table CITY

CITY  Enter a SQL expression to filter results (use Ctrl+Space)									
Grid	123 PK_CITY_ID	123 FK_COUNTRY_STATE_ID	CITY_MAP		CREATED	MODIFIED	NAME		
1	0	417	PNG	IHDR Ö Ü çOä... [19664]	2020-07-07 21:16:36	2020-07-09 13:58:19	NAME_NAME_0		
2	1	154	[NULL]		2020-07-01 15:29:56	2020-07-11 13:22:42	NAME_COMPANÍA_1		
3	2	307	PNG	IHDR Ö Ü çOä... [19664]	2020-07-03 17:37:20	[NULL]	NAME_名称_2		
4	3	270	PNG	IHDR Ö Ü çOä... [19664]	2020-07-03 08:17:33	2020-07-03 09:21:59	NAME_NAME_3		
5	4	201	PNG	IHDR Ö Ü çOä... [19664]	2020-07-09 02:37:50	[NULL]	NAME_COMPANÍA_4		
6	5	119	PNG	IHDR Ö Ü çOä... [19664]	2020-07-13 23:50:00	2020-07-12 20:48:13	NAME_名称_5		
7	6	[NULL]	[NULL]		2020-07-20 18:08:54	[NULL]	NAME_NAME_6		
8	7	[NULL]	PNG	IHDR Ö Ü çOä... [19664]	2020-07-10 02:51:42	2020-07-13 22:11:40	NAME_COMPANÍA_7		
9	8	261	PNG	IHDR Ö Ü çOä... [19664]	2020-07-13 14:53:57	[NULL]	NAME_名称_8		
10	9	584	[NULL]		2020-07-13 20:23:48	[NULL]	NAME_NAME_9		
11	10	[NULL]	PNG	IHDR Ö Ü çOä... [19664]	2020-07-04 17:37:34	[NULL]	NAME_COMPANÍA_10		
12	11	35	PNG	IHDR Ö Ü çOä... [19664]	2020-07-19 01:25:20	2020-06-27 19:33:15	NAME_名称_11		
13	12	553	PNG	IHDR Ö Ü çOä... [19664]	2020-06-28 16:59:18	[NULL]	NAME_NAME_12		
14	13	401	PNG	IHDR Ö Ü çOä... [19664]	2020-07-09 15:50:28	2020-07-11 00:37:42	NAME_COMPANÍA_13		
15	14	127	PNG	IHDR Ö Ü çOä... [19664]	2020-07-15 07:32:40	2020-07-01 18:59:11	NAME_名称_14		
16	15	296	PNG	IHDR Ö Ü çOä... [19664]	2020-07-10 12:51:31	2020-07-17 11:12:31	NAME_NAME_15		
17	16	[NULL]	PNG	IHDR Ö Ü çOä... [19664]	2020-07-18 19:15:29	2020-07-11 12:23:04	NAME_COMPANÍA_16		
18	17	523	PNG	IHDR Ö Ü çOä... [19664]	2020-07-08 23:38:14	2020-07-09 19:58:29	NAME_名称_17		
19	18	230	PNG	IHDR Ö Ü çOä... [19664]	2020-07-01 01:32:05	2020-07-11 15:13:29	NAME_NAME_18		
20	19	275	[NULL]		2020-07-02 19:47:10	2020-06-28 15:22:39	NAME_COMPANÍA_19		
21	20	274	PNG	IHDR Ö Ü çOä... [19664]	2020-07-19 23:43:47	2020-07-04 22:07:54	NAME_名称_20		
22	21	[NULL]	[NULL]		2020-07-11 22:24:56	[NULL]	NAME_NAME_21		
23	22	[NULL]	PNG	IHDR Ö Ü çOä... [19664]	2020-07-08 03:42:43	2020-07-01 21:37:16	NAME_COMPANÍA_22		
24	23	[NULL]	PNG	IHDR Ö Ü çOä... [19664]	2020-06-27 08:43:04	2020-06-29 23:47:39	NAME_名称_23		
25	24	158	PNG	IHDR Ö Ü çOä... [19664]	2020-06-28 20:05:01	2020-07-09 22:41:36	NAME_NAME_24		

2. Table COUNTRY

COUNTRY  Enter a SQL expression to filter results (use Ctrl+Space)									
	123 PK_COUNTRY_ID	COUNTRY_MAP		CREATED	ISO3166	MODIFIED	NAME		
1	0	PNG	IHDR Ö Ü çOä... [19664]	2020-06-28 19:36:46	[NULL]	2020-06-30 00:33:11	NAME_NAME_0		
2	1	PNG	IHDR Ö Ü çOä... [19664]	2020-07-19 07:16:11	ISO3166_CÓDIGO 3166_1	2020-06-27 23:12:01	NAME_COMPANÍA_1		
3	2	PNG	IHDR Ö Ü çOä... [19664]	2020-07-08 22:39:47	ISO3166_ISO 3166标准_2	2020-07-16 18:37:01	NAME_名称_2		
4	3	[NULL]		2020-07-21 14:58:16	ISO3166_ISO3166_3	2020-07-16 01:14:12	NAME_NAME_3		
5	4	PNG	IHDR Ö Ü çOä... [19664]	2020-07-01 19:00:23	ISO3166_CÓDIGO 3166_4	2020-06-29 07:27:24	NAME_COMPANÍA_4		
6	5	PNG	IHDR Ö Ü çOä... [19664]	2020-07-05 00:54:26	ISO3166_ISO 3166标准_5	2020-07-08 02:20:09	NAME_名称_5		
7	6	PNG	IHDR Ö Ü çOä... [19664]	2020-07-06 22:36:40	ISO3166_ISO3166_6	2020-07-04 08:24:18	NAME_NAME_6		
8	7	PNG	IHDR Ö Ü çOä... [19664]	2020-07-20 07:33:21	ISO3166_CÓDIGO 3166_7	2020-06-28 06:10:36	NAME_COMPANÍA_7		
9	8	PNG	IHDR Ö Ü çOä... [19664]	2020-07-21 14:34:52	ISO3166_ISO 3166标准_8	[NULL]	NAME_名称_8		
10	9	PNG	IHDR Ö Ü çOä... [19664]	2020-07-08 08:27:06	ISO3166_ISO3166_9	2020-07-16 00:17:52	NAME_NAME_9		
11	10	[NULL]		2020-06-29 04:43:50	ISO3166_CÓDIGO 3166_10	2020-07-14 19:08:27	NAME_COMPANÍA_10		
12	11	PNG	IHDR Ö Ü çOä... [19664]	2020-06-29 11:03:58	ISO3166_ISO 3166标准_11	[NULL]	NAME_名称_11		
13	12	PNG	IHDR Ö Ü çOä... [19664]	2020-07-16 15:39:56	ISO3166_ISO3166_12	2020-06-27 16:50:58	NAME_NAME_12		
14	13	PNG	IHDR Ö Ü çOä... [19664]	2020-07-12 01:44:07	[NULL]	[NULL]	NAME_COMPANÍA_13		
15	14	PNG	IHDR Ö Ü çOä... [19664]	2020-07-12 07:01:06	ISO3166_ISO 3166标准_14	2020-07-18 06:54:10	NAME_名称_14		
16	15	PNG	IHDR Ö Ü çOä... [19664]	2020-07-09 15:54:56	[NULL]	[NULL]	NAME_NAME_15		
17	16	PNG	IHDR Ö Ü çOä... [19664]	2020-07-16 12:48:24	[NULL]	[NULL]	NAME_COMPANÍA_16		
18	17	PNG	IHDR Ö Ü çOä... [19664]	2020-07-06 08:49:55	ISO3166_ISO 3166标准_17	[NULL]	NAME_名称_17		
19	18	[NULL]		2020-07-19 16:21:58	[NULL]	2020-07-02 11:22:09	NAME_NAME_18		
20	19	PNG	IHDR Ö Ü çOä... [19664]	2020-07-17 04:08:40	ISO3166_CÓDIGO 3166_19	2020-07-04 04:18:40	NAME_COMPANÍA_19		
21	20	PNG	IHDR Ö Ü çOä... [19664]	2020-07-09 16:02:53	[NULL]	2020-07-19 14:04:43	NAME_名称_20		
22	21	PNG	IHDR Ö Ü çOä... [19664]	2020-06-29 20:47:52	[NULL]	[NULL]	NAME_NAME_21		
23	22	[NULL]		2020-07-12 01:36:41	[NULL]	2020-07-03 14:32:35	NAME_COMPANÍA_22		
24	23	[NULL]		2020-07-19 09:13:19	ISO3166_ISO 3166标准_23	2020-07-12 17:02:15	NAME_名称_23		
25	24	PNG	IHDR Ö Ü çOä... [19664]	2020-07-02 10:15:05	[NULL]	2020-07-06 00:07:07	NAME_NAME_24		

3. Table TIMEZONE

TIMEZONE <small>Enter a SQL expression to filter results (use Ctrl+Space)</small>							
	PK_TIMEZONE_ID	ABBREVIATION	CREATED	MODIFIED	NAME	V_TIME_ZONE	
1	0	ABBREVIATION_ABBREVIATION_0	2020-07-06 02:00:35	[NULL]	NAME_NAME_0	V_TIME_ZONE_V_TIME_ZONE_0	
2	1	ABBREVIATION_ABBREVIATION_1	2020-07-19 23:30:25	2020-07-14 16:15:02	NAME_COMPANIA_1	V_TIME_ZONE_FUSO_HORARIO_1	
3	2	ABBREVIATION_缩写_2	2020-06-27 02:48:13	2020-07-05 15:25:17	NAME_名称_2	V_TIME_ZONE_时区_2	
4	3	ABBREVIATION_ABBREVIATION_3	2020-07-06 20:57:16	2020-06-28 20:43:59	NAME_NAME_3	V_TIME_ZONE_V_TIME_ZONE_3	
5	4	ABBREVIATION_ABBREVIATION_4	2020-07-15 16:05:02	2020-07-02 00:19:52	NAME_COMPANIA_4	[NULL]	
6	5	ABBREVIATION_缩写_5	2020-07-13 22:57:44	2020-06-27 08:14:01	NAME_名称_5	V_TIME_ZONE_时区_5	
7	6	ABBREVIATION_ABBREVIATION_6	2020-07-19 22:16:01	2020-07-16 10:19:57	NAME_NAME_6	V_TIME_ZONE_V_TIME_ZONE_6	
8	7	ABBREVIATION_ABBREVIATION_7	2020-07-10 03:25:25	2020-06-27 15:57:09	NAME_COMPANIA_7	V_TIME_ZONE_FUSO_HORARIO_7	
9	8	ABBREVIATION_缩写_8	2020-07-03 05:45:30	2020-07-14 01:20:33	NAME_名称_8	V_TIME_ZONE_时区_8	
10	9	ABBREVIATION_ABBREVIATION_9	2020-07-13 23:10:17	2020-07-05 07:34:24	NAME_NAME_9	V_TIME_ZONE_V_TIME_ZONE_9	
11	10	ABBREVIATION_ABBREVIATION_10	2020-07-03 00:11:26	2020-07-11 17:19:21	NAME_COMPANIA_10	V_TIME_ZONE_FUSO_HORARIO_10	

3. Installation

The easiest way is to download a current release of **DBSeeder** from the GitHub repository. You can find the necessary link [here](#).

To download the repository [Git](#) is needed and for compilation the [Gradle Build Tool](#) and the [open-source JDK](#) are needed. For changes to the **DBSeeder** repository it is best to use an editor (e.g. [Vim](#)) or an IDE (e.g. [Eclipse IDE](#)). For using the Docker Image based databases in operational mode, [Docker Desktop](#) must also be installed. For the respective software versions, please consult the document [release notes](#).

4. Operating Instructions

4.1 Script `run_db_seeder`

Using the **DBSeeder** development and operational Docker image from Docker Hub (see [here](#)) eliminates the need to install the runtime environment.

With the script `run_db_seeder` the complete functionality of the **DBSeeder** application can be used:

- Creating a suitable database
- Generation of any number of dummy data.

All scripts are available in a Windows version (`cmd` / `.bat`) as well as in a Unix version (`bash` / `.sh`). To run the scripts, apart from the prerequisites as release notes ([ReleaseNotes.md](#)), only the libraries in the `lib` directory and the corresponding script of `run_db_seeder` are required. The creation of the databases also requires a working access to [Docker Hub](#).

All control parameters used in **DBSeeder** (see section 4.3) can be adapted in the scripts to specific needs.

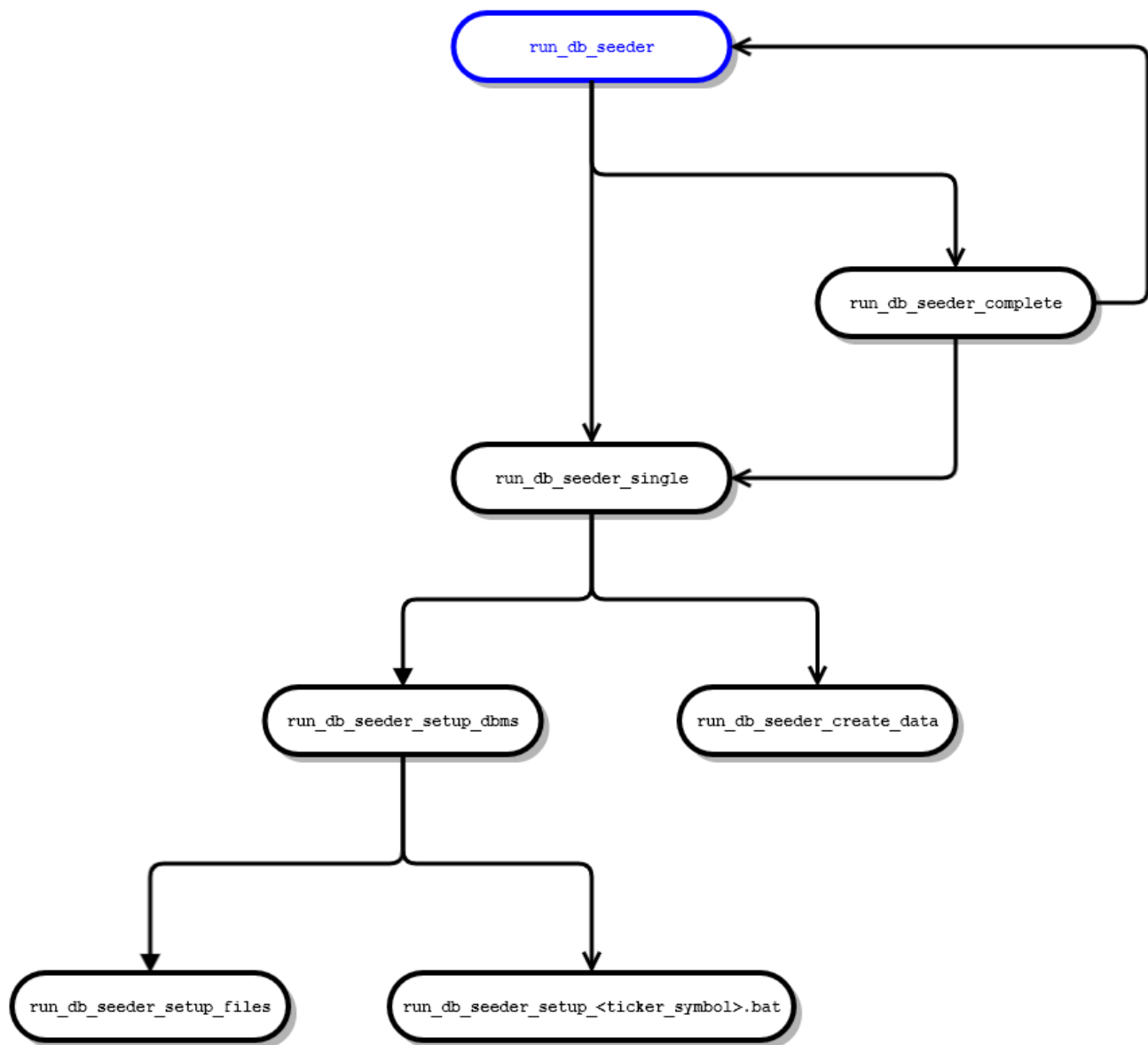
The `run_db_seeder` script is controlled by the following script parameters::

- **DB_SEEDER_DBMS**: the ticker symbol of the desired database management system (default value `sqlite`) or `complete` for all implemented RDBMS.
- **DB_SEEDER_SETUP_DBMS**: should an empty database be created:
 - `yes`: a new database is created based on a suitable Docker image
 - otherwise: no database is created
- **DB_SEEDER_NO_CREATE_RUNS**: Number of dummy data generation runs:
 - 1: one run
 - 2: two runs
 - otherwise: no run

For the run variants `complete`, `complete_client`, `complete_emb` and `complete_trino`, statistics files with the following data name structure are created in the file directory `resources/statistics` by default:

```
db_seeder_<bash|cmd>_<run variant>_unknown_<DBSeeder release>_<vmware|win10|ws12>.<csv|tsv>
```

An overview of the structure of the scripts used can be taken from the following diagram:



4.2 Operation Possibilities

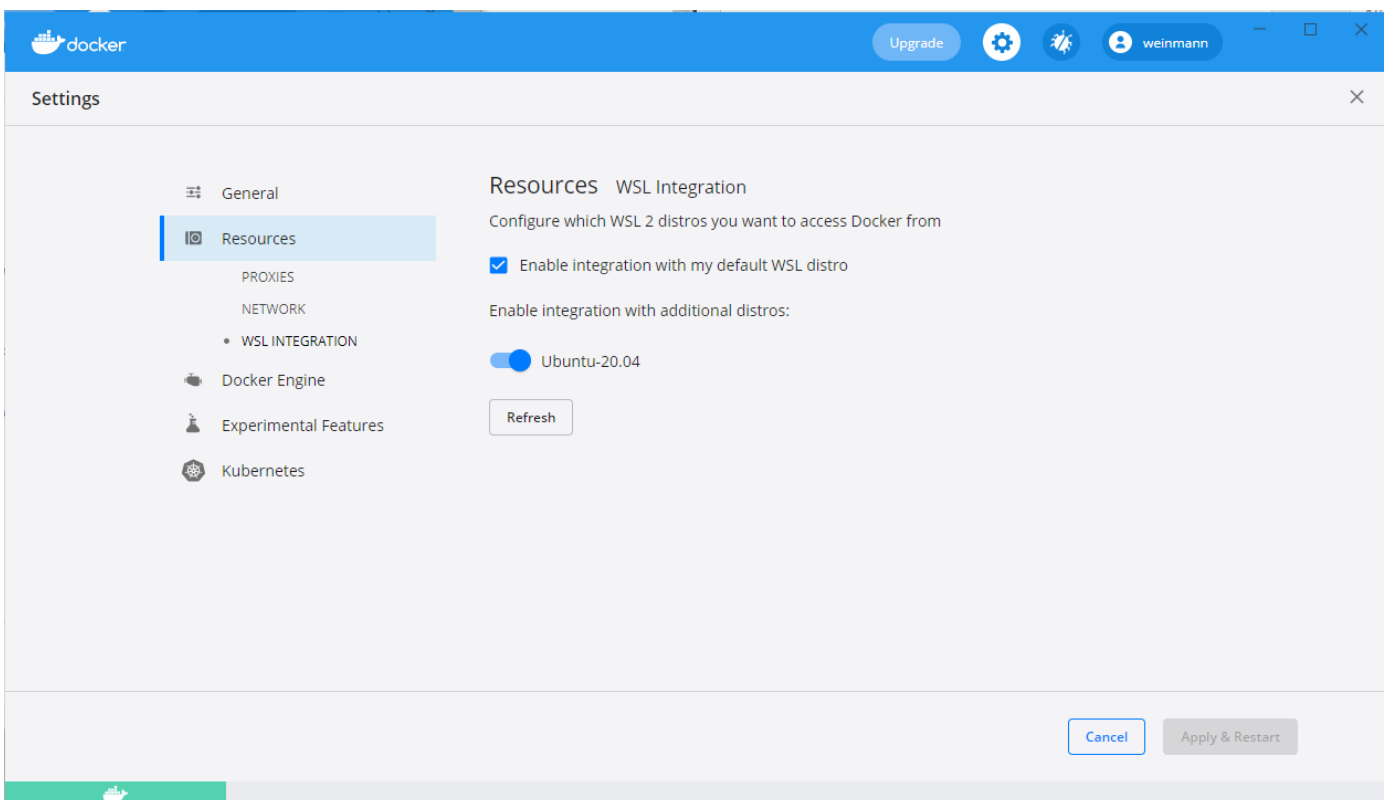
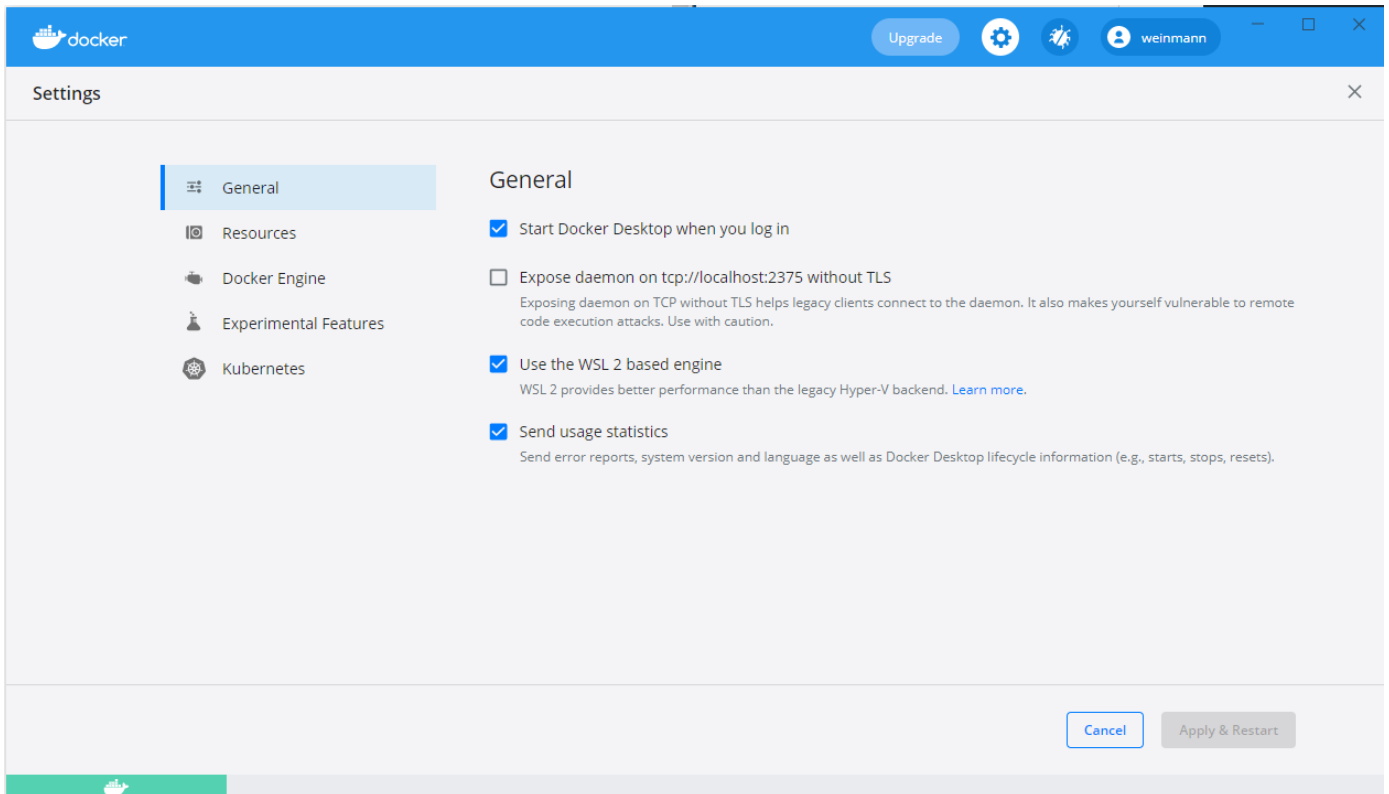
DBSeeder is tested under [Ubuntu](#) and [Microsoft Windows](#). In addition, tests are always performed in Windows with Ubuntu under the [Windows Subsystem for Linux \(WSL\)](#). Besides one of the two operating systems, these are the minimum requirements for running **DBSeeder**:

- [Docker Desktop Community](#)
- [Eclipse IDE](#)
- [Gradle Build Tool](#)
- [Java Development Kit](#)

Details on the required software versions can be found in the [release notes](#).

Special Features for the Operation with Ubuntu

- A suitable image is available on Docker Hub for development and operation, see [here](#).
- In the directory `scripts/3.0.2` are the two scripts `run_install_4_vm_wsl2_1.sh` and `run_install_4_vm_wsl2_1.sh` with which an Ubuntu environment can be prepared for development and operation.
- If the Windows Subsystem for Linux (WSL) is to be used, then the **WSL INTEGRATION** for Ubuntu must be activated in Docker



4.3 Control Parameters

4.3.1 Supported Parameters

The flow control parameters for **DBSeeder** are stored in the properties file `src/main/resources/db_seeder.properties` and can all be overridden by the environment variables defined in the scripts. The following control parameters are currently supported:

```
db_seeder.batch.size=0
db_seeder.character.set.server=
db_seeder.collation.server=
db_seeder.connection.host=
db_seeder.connection.host.trino=
```

```

db_seeder.connection.port=0
db_seeder.connection.port.trino=0
db_seeder.connection.prefix=
db_seeder.connection.service=
db_seeder.connection.suffix=

db_seeder.database.sys=
db_seeder.database=
db_seeder.drop.constraints=

db_seeder.file.configuration.name=yes
db_seeder.file.improvement.header=DBMS;Type;ms;Constraints;Improvement
db_seeder.file.improvement.name=
db_seeder.file.json.name=resources/json/db_seeder_schema_company_5400.json
db_seeder.file.statistics.delimiter=\t
db_seeder.file.statistics.header=ticker symbol;DBMS;db type;total ms;start time;end time;host name;no.
cores;operating system;total DDL ms;drop constr. ms;add constr. ms;total DML ms;constraints
db_seeder.file.statistics.name=resources/statistics/db_seeder_statistics.tsv
db_seeder.file.summary.name=
db_seeder.file.summary.source=resources/statistics

db_seeder.password.sys=
db_seeder.password=

db_seeder.schema=

db_seeder.user.sys=
db_seeder.user=

```

4.3.2 Explanation and Cross-reference

Property incl. Default Value [db.seeder.]	Environment Variable [DB_SEEDER_]	Used By	Description
batch.size=<9...9>	BATCH_SIZE	all RDBMS except	number of insert operations for the bulk operation, default value 0 (a single bulk operation for each database table)
character.set.server=<x...x>	CHARACTER_SET_SERVER	mariadb	default server character set
collation.server=<x...x>	COLLATION_SERVER	mariadb	default server collation
connection.host=<x...x>	CONNECTION_HOST	all client RDBMS	host name or ip address of the database server
connection.host_trino=<x...x>	CONNECTION_HOST_TRINO	trino	host name or ip address of the trino
connection.port=<9...9>	CONNECTION_PORT	all client RDBMS	port number of the database server
connection.port_trino=<9...9>	CONNECTION_PORT_TRINO	trino	port number of the trino
connection.prefix=<x...x>	CONNECTION_PREFIX	all RDBMS	prefix of the database connection string
connection.service=<x...x>	CONNECTION_SERVICE	oracle	service name of the database connection string
connection.suffix=<x...x>	CONNECTION_SUFFIX	firebird, hsqldb, mysql, percona, voltdb	suffix of the database connection string
database.sys=<x...x>	DATABASE_SYS	agens, cockroach, informix, mariadb, mimer, monetdb, mysql, omnisci, percona,	privileged database name
		postgresql, sqlserver, timescale, yugabyte	
database=<x...x>	DATABASE	all RDBMS except cratedb, exasol, monetdb, oracle, voltdb	database name

Property incl. Default Value [db.seeder.]	Environment Variable [DB_SEEDER_]	Used By	Description
drop.constraints=	DROP_CONSTRAINTS	all RDBMS except cockroach, cratedb, h2, omnic, sqlite, trino	drop all constraints before the DML operations and recreate them afterwards
file.configuration.name= <x...x>	FILE_CONFIGURATION_NAME	n/a	directory and file name of the DBSeeder configuration file
file.improvement.header= <x...x>	FILE_IMPROVEMENT_HEADER	all RDBMS	header line of the improvement file created in run_db_seeder_compute_improvement
file.improvement.name= <x...x>	FILE_IMPROVEMENT_NAME	all RDBMS	directory and file name of the DBSeeder improvement file created in run_db_seeder_compute_improvement
file.json.name= <x...x>	FILE_JSON_NAME	scripts/run_db_seeder_generate_schema	directory and file name of the JSON file containing the database schema
file.statistics.delimiter= <x...x>	FILE_STATISTICS_DELIMITER	all RDBMS	separator of the statistics file created in run_db_seeder
file.statistics.header= <x...x>	FILE_STATISTICS_HEADER	all RDBMS	header line of the statistics file created in run_db_seeder
file.statistics.name= <x...x>	FILE_STATISTICS_NAME	all RDBMS	file name of the statistics file created in run_db_seeder
file.summary.name= <x...x>	FILE_SUMMARY_NAME	all RDBMS	file name of the summary statistics file created in run_db_seeder_create_summary
file.summary.source= <x...x>	FILE_SUMMARY_SOURCE	all RDBMS	directory name(s) (separated by semicolon) of the source directories containing statistics files
password.sys= <x...x>	PASSWORD_SYS	agens, exasol, firebird, ibmdb2, informix, mariadb, mimer, monetdb, mysql, omnic,	password of the privileged user
		oracle, percona, postgresql, sqlserver, timescale	
password= <x...x>	PASSWORD	all RDBMS except cockroach, derby, ibmdb2, informix	password of the normal user
schema=kxn_schema	SCHEMA	agens, derby, exasol, h2, hsqldb, ibmdb2, monetdb, postgresql, sqlserver, timescale, yugabyte	schema name
user.sys= <x...x>	USER_SYS	all RDBMS except derby, voltdb	name of the privileged user
user=kxn_user	USER	all RDBMS except derby, ibmdb2, informix	name of the normal user

4.4 Statistics

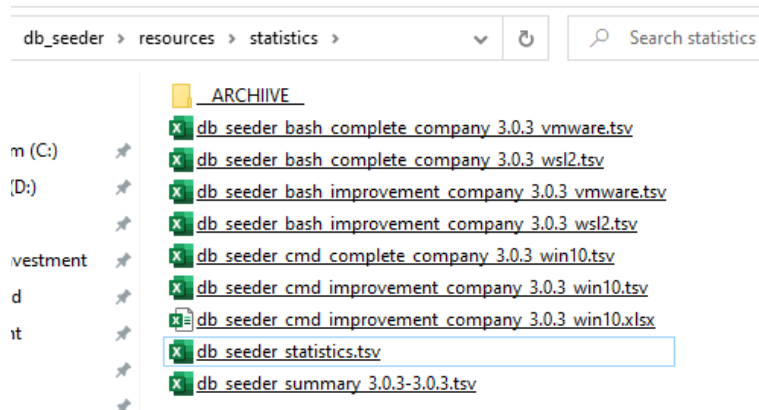
Each new release is completed with the creation of 7 statistics files in the file directory **resources/statistics**. The data contained in these files show the DDL and DML performance of the individual databases under identical conditions:

- Operating systems: Ubuntu with VMware Workstation Player, Ubuntu with WSL (Windows Subsystem for Linux) on Windows and Windows.
 - ..._vmware.tsv**: Ubuntu with VMware Workstation Player on Windows
 - ..._win10.tsv**: Windows 10
 - ..._ws12.tsv**: Ubuntu LTS with Windows Subsystem for Linux 2 on Windows
- DDL: Creation of the database schema consisting of the 5 relational tables CITY, COMPANY, COUNTRY, COUNTRY_STATE and TIMEZONE (see JSON file: **resources/json/db_seeder_schema.company_5400.json**).
- DML: Insert records into these database tables - CITY 1800, COMPANY 5400, COUNTRY 200, COUNTRY_STATE 600 and TIMEZONE 11.

- If possible, two runs are made for each database system: one run with constraints enabled and one run with constraints disabled - see column **constraints**:
 - active**: constraints are enabled
 - active - no choice**: constraints are enabled and disabling is not possible
 - inactive**: constraints are disabled

The creation of these statistics files is managed by the following control parameters ([see here](#)):

```
db_seeder.file.improvement.header=DBMS;Type;ms;Constraints;Improvement
db_seeder.file.improvement.name=
db_seeder.file.statistics.delimiter=\t
db_seeder.file.statistics.header=ticker symbol;DBMS;db type;total ms;start time;end time;host name;no.
cores;operating system;total DDL ms;drop constr. ms;add constr. ms;total DML ms;constraints
db_seeder.file.statistics.name=resources/statistics/db_seeder_statistics.tsv
db_seeder.file.summary.name=
db_seeder.file.summary.source=resources/statistics
```



4.4.1 Detailed statistical data

ticker symbol	DBMS	db type	total ms	start time	end time	host name	no. cores	operating system	total DDL ms	drop constr. ms	add constr. ms	total DML ms	constraints
agens	AgensGraph	client	4128	10.09.2021 09:14	10.09.2021 09:15	fwfw-win-dt-21	32	amd64 / Windows 10 / 10.0	930	0	0	3198	active
cockroach	CockroachDB	client	5630	10.09.2021 09:15	10.09.2021 09:15	fwfw-win-dt-21	32	amd64 / Windows 10 / 10.0	924	0	0	4706	active - no choice
cratedb	Cratedb	client	5342	10.09.2021 09:16	10.09.2021 09:16	fwfw-win-dt-21	32	amd64 / Windows 10 / 10.0	1987	0	0	3355	active - no choice
cubrid	CUBRID	client	6456	10.09.2021 09:16	10.09.2021 09:17	fwfw-win-dt-21	32	amd64 / Windows 10 / 10.0	442	0	0	6014	active
derby	Apache Derby	client	15405	10.09.2021 09:18	10.09.2021 09:18	fwfw-win-dt-21	32	amd64 / Windows 10 / 10.0	3445	0	0	11960	active
derby_emb	Apache Derby	embedded	16202	10.09.2021 09:18	10.09.2021 09:18	fwfw-win-dt-21	32	amd64 / Windows 10 / 10.0	6943	0	0	9259	active
exasol	Exasol	client	4195	10.09.2021 09:20	10.09.2021 09:20	fwfw-win-dt-21	32	amd64 / Windows 10 / 10.0	752	0	0	3443	active
firebird	Firebird	client	93671	10.09.2021 09:21	10.09.2021 09:23	fwfw-win-dt-21	32	amd64 / Windows 10 / 10.0	941	0	0	92730	active
h2	H2 Database Engine	client	7648	10.09.2021 09:24	10.09.2021 09:24	fwfw-win-dt-21	32	amd64 / Windows 10 / 10.0	527	0	0	7121	active - no choice
h2_emb	H2 Database Engine	embedded	2976	10.09.2021 09:24	10.09.2021 09:24	fwfw-win-dt-21	32	amd64 / Windows 10 / 10.0	987	0	0	1989	active - no choice
hsqldb	HSQldb	client	17952	10.09.2021 09:24	10.09.2021 09:25	fwfw-win-dt-21	32	amd64 / Windows 10 / 10.0	602	0	0	17350	active
hsqldb_emb	HSQldb	embedded	2864	10.09.2021 09:25	10.09.2021 09:25	fwfw-win-dt-21	32	amd64 / Windows 10 / 10.0	476	0	0	2388	active
ibmdb2	IBM Db2 Database	client	23578	10.09.2021 09:30	10.09.2021 09:30	fwfw-win-dt-21	32	amd64 / Windows 10 / 10.0	3329	0	0	20249	active
informix	IBM Informix	client	19946	10.09.2021 09:34	10.09.2021 09:34	fwfw-win-dt-21	32	amd64 / Windows 10 / 10.0	1869	0	0	18077	active
mariadb	MariaDB Server	client	3815	10.09.2021 09:35	10.09.2021 09:35	fwfw-win-dt-21	32	amd64 / Windows 10 / 10.0	501	0	0	3314	active
mimer	Mimer SQL	client	9286	10.09.2021 09:36	10.09.2021 09:36	fwfw-win-dt-21	32	amd64 / Windows 10 / 10.0	500	0	0	8786	active
monetdb	MonetDB	client	9525	10.09.2021 09:36	10.09.2021 09:36	fwfw-win-dt-21	32	amd64 / Windows 10 / 10.0	928	0	0	8597	active
mysql	MySQL Database	client	7636	10.09.2021 09:38	10.09.2021 09:38	fwfw-win-dt-21	32	amd64 / Windows 10 / 10.0	798	0	0	6838	active
mysql_trino	MySQL Database	trino	103599	10.09.2021 09:39	10.09.2021 09:57	fwfw-win-dt-21	32	amd64 / Windows 10 / 10.0	1428	0	0	1034571	active - no choice
omnisc	OmniSciDB	client	37538	10.09.2021 09:58	10.09.2021 09:58	fwfw-win-dt-21	32	amd64 / Windows 10 / 10.0	5294	0	0	32244	active - no choice
oracle	Oracle Database	client	12808	10.09.2021 10:00	10.09.2021 10:00	fwfw-win-dt-21	32	amd64 / Windows 10 / 10.0	853	0	0	11955	active
oracle_trino	Oracle Database	trino	1877924	10.09.2021 10:02	10.09.2021 10:33	fwfw-win-dt-21	32	amd64 / Windows 10 / 10.0	866	0	0	1877058	active - no choice
percona	Percona Server for MySQL	client	6614	10.09.2021 10:34	10.09.2021 10:34	fwfw-win-dt-21	32	amd64 / Windows 10 / 10.0	756	0	0	5858	active
postgres	PostgreSQL	client	3940	10.09.2021 10:35	10.09.2021 10:35	fwfw-win-dt-21	32	amd64 / Windows 10 / 10.0	655	0	0	3285	active
postgres_trino	PostgreSQL	trino	510823	10.09.2021 10:36	10.09.2021 10:45	fwfw-win-dt-21	32	amd64 / Windows 10 / 10.0	696	0	0	510127	active - no choice
sqlite	SQLite	embedded	4426	10.09.2021 10:45	10.09.2021 10:45	fwfw-win-dt-21	32	amd64 / Windows 10 / 10.0	1015	0	0	3411	active - no choice
sqlserver	MS SQL Server	client	5300	10.09.2021 10:46	10.09.2021 10:46	fwfw-win-dt-21	32	amd64 / Windows 10 / 10.0	1047	0	0	4253	active
sqlserver_trino	MS SQL Server	trino	1049892	10.09.2021 10:47	10.09.2021 11:04	fwfw-win-dt-21	32	amd64 / Windows 10 / 10.0	1048	0	0	1048844	active - no choice
timescale	TimescaleDB	client	4223	10.09.2021 11:05	10.09.2021 11:05	fwfw-win-dt-21	32	amd64 / Windows 10 / 10.0	915	0	0	3308	active
yugabyte	YugabyteDB	client	13516	10.09.2021 11:06	10.09.2021 11:06	fwfw-win-dt-21	32	amd64 / Windows 10 / 10.0	6728	0	0	6788	active
agens	AgensGraph	client	3987	10.09.2021 11:07	10.09.2021 11:07	fwfw-win-dt-21	32	amd64 / Windows 10 / 10.0	852	0	79	3135	inactive
cubrid	CUBRID	client	6776	10.09.2021 11:07	10.09.2021 11:08	fwfw-win-dt-21	32	amd64 / Windows 10 / 10.0	617	68	94	6159	inactive
derby	Apache Derby	client	15899	10.09.2021 11:08	10.09.2021 11:09	fwfw-win-dt-21	32	amd64 / Windows 10 / 10.0	3972	211	318	11927	inactive
derby_emb	Apache Derby	embedded	18192	10.09.2021 11:09	10.09.2021 11:09	fwfw-win-dt-21	32	amd64 / Windows 10 / 10.0	8913	1402	237	9279	inactive
exasol	Exasol	client	4363	10.09.2021 11:11	10.09.2021 11:11	fwfw-win-dt-21	32	amd64 / Windows 10 / 10.0	902	24	120	3461	inactive
firebird	Firebird	client	76446	10.09.2021 11:12	10.09.2021 11:13	fwfw-win-dt-21	32	amd64 / Windows 10 / 10.0	1462	359	191	74984	inactive
hsqldb	HSQldb	client	18580	10.09.2021 11:14	10.09.2021 11:15	fwfw-win-dt-21	32	amd64 / Windows 10 / 10.0	702	48	69	17878	inactive
ibmdb2	IBM Db2 Database	client	24319	10.09.2021 11:20	10.09.2021 11:20	fwfw-win-dt-21	32	amd64 / Windows 10 / 10.0	4269	923	321	20050	inactive
informix	IBM Informix	client	20057	10.09.2021 11:24	10.09.2021 11:24	fwfw-win-dt-21	32	amd64 / Windows 10 / 10.0	2117	54	191	17940	inactive
mariadb	MariaDB Server	client	6625	10.09.2021 11:25	10.09.2021 11:25	fwfw-win-dt-21	32	amd64 / Windows 10 / 10.0	3201	2461	216	3424	inactive
mimer	Mimer SQL	client	9508	10.09.2021 11:26	10.09.2021 11:26	fwfw-win-dt-21	32	amd64 / Windows 10 / 10.0	798	111	195	8710	inactive
monetdb	MonetDB	client	8844	10.09.2021 11:26	10.09.2021 11:26	fwfw-win-dt-21	32	amd64 / Windows 10 / 10.0	1050	0	137	7794	inactive
mysql	MySQL Database	client	18888	10.09.2021 11:27	10.09.2021 11:28	fwfw-win-dt-21	32	amd64 / Windows 10 / 10.0	12326	11322	269	6562	inactive
oracle	Oracle Database	client	16229	10.09.2021 11:29	10.09.2021 11:29	fwfw-win-dt-21	32	amd64 / Windows 10 / 10.0	4352	128	3368	11877	inactive
percona	Percona Server for MySQL	client	19871	10.09.2021 11:30	10.09.2021 11:31	fwfw-win-dt-21	32	amd64 / Windows 10 / 10.0	14109	12998	350	5762	inactive
postgres	PostgreSQL	client	3804	10.09.2021 11:31	10.09.2021 11:31	fwfw-win-dt-21	32	amd64 / Windows 10 / 10.0	752	0	83	3052	inactive
sqlserver	MS SQL Server	client	5467	10.09.2021 11:32	10.09.2021 11:32	fwfw-win-dt-21	32	amd64 / Windows 10 / 10.0	1519	237	301	3948	inactive
timescale	TimescaleDB	client	3797	10.09.2021 11:33	10.09.2021 11:33	fwfw-win-dt-21	32	amd64 / Windows 10 / 10.0	798	0	83	2999	inactive
yugabyte	YugabyteDB	client	13055	10.09.2021 11:33	10.09.2021 11:33	fwfw-win-dt-21	32	amd64 / Windows 10 / 10.0	6873	0	413	6182	inactive

File name syntax: `db_seeder_<bash|cmd>_complete_<company|syntax>_<DBSeeder version>_<vmware|wsl2|win10>.<csv|tsv>`

Explanation for the columns:

- **ticker symbol** - internal abbreviation used for the database
- **DBMS** - official DBMS name
- **db type** - client version, embedded version or via trino
- **total ms** - total time of DDL and DML operations in milliseconds
- **start time** - date and time when the database operations were started
- **end time** - date and time when the database operations were completed
- **host name** - name of the computer connected to a computer network
- **no. cores** - number of CPU cores used
- **operating system**
- **total DDL ms** - total time of DDL operations in milliseconds
- **drop constr. ms** - total time to drop all constraints
- **add constr. ms** - total time to add the previously dropped constraints
- **total DML ms** - total time of DML operations in milliseconds
- **constraints** - DML operations with enabled (active) or disabled (inactive) constraints (foreign, primary and unique key)

4.4.2 Performance data regarding constraints

DBMS	Type	ms	Constraints	Improvement
CockroachDB	client	5630	active - no choice	
CrateDB	client	5342	active - no choice	
CUBRID	client	6456	active	-5
Apache Derby	client	15405	active	-3.2
Apache Derby	embedded	16202	active	-12.3
Exasol	client	4195	active	-4
H2 Database Engine	client	7648	active - no choice	
H2 Database Engine	embedded	2976	active - no choice	
HSQldb	client	17952	active	-3.5
HSQldb	embedded	2864	active	
IBM Db2 Database	client	23578	active	-3.1
IBM Informix	client	19946	active	-0.6
MariaDB Server	client	3815	active	-73.7
Mimer SQL	client	9286	active	-2.4
MySQL Database	client	7636	active	-147.4
MySQL Database	trino	1035999	active - no choice	
OmniSciDB	client	37538	active - no choice	
Oracle Database	client	12808	active	-26.7
Oracle Database	trino	1877924	active - no choice	
Percona Server for MySQL	client	6614	active	-200.4
PostgreSQL	trino	510823	active - no choice	
SQLite	embedded	4426	active - no choice	
MS SQL Server	client	5300	active	-3.2
MS SQL Server	trino	1049892	active - no choice	
AgensGraph	client	3987	inactive	3.4
Firebird	client	76446	inactive	18.4
MonetDB	client	8844	inactive	7.1
PostgreSQL	client	3804	inactive	3.5
TimescaleDB	client	3797	inactive	10.1
YugabyteDB	client	13055	inactive	3.4

File name syntax: db_seeder_<bash|cmd>_improvement_<company|syntax>_<DBSeeder version>_<vmware|ws12|win10>.<csv|tsv>

Explanation for the columns:

- **DBMS** - official DBMS name
- **Type** - client version, embedded version or via trino
- **ms** - total time of DDL and DML operations in milliseconds
- **Constraints** - DML operations with enabled (active) or disabled (inactive) constraints (foreign, primary and unique key)
- **Improvement** - improvement of total time if constraints are inactive

4.4.3 Historical statistical data

ticker symbol	DBMS	version	creator	db type	constraints	schema	total ms	start time	end time	host name	no. cores	operating system	file name
agens	AgensGraph	v3.0.3	bash	client	active	company	3915	10.09.2021 12:07	10.09.2021 12:07	fjww-win-dt-21	32	amd64 / Linux / 5.4.72-microsoft-standard-WSL2	db_seeder_bash_complete_company_3.0.3_ws12
cockroach	CockroachDB	v3.0.3	bash	client	active - no choice	company	5619	10.09.2021 12:07	10.09.2021 12:07	fjww-win-dt-21	32	amd64 / Linux / 5.4.72-microsoft-standard-WSL2	db_seeder_bash_complete_company_3.0.3_ws12
cratedb	CrateDB	v3.0.3	bash	client	active - no choice	company	5058	10.09.2021 12:08	10.09.2021 12:08	fjww-win-dt-21	32	amd64 / Linux / 5.4.72-microsoft-standard-WSL2	db_seeder_bash_complete_company_3.0.3_ws12
cuprid	CUBRID	v3.0.3	bash	client	active	company	5934	10.09.2021 12:09	10.09.2021 12:09	fjww-win-dt-21	32	amd64 / Linux / 5.4.72-microsoft-standard-WSL2	db_seeder_bash_complete_company_3.0.3_ws12
derby	Apache Derby	v3.0.3	bash	client	active	company	14607	10.09.2021 12:09	10.09.2021 12:10	fjww-win-dt-21	32	amd64 / Linux / 5.4.72-microsoft-standard-WSL2	db_seeder_bash_complete_company_3.0.3_ws12
derby_emb	Apache Derby	v3.0.3	bash	embedded	active	company	12222	10.09.2021 12:10	10.09.2021 12:10	fjww-win-dt-21	32	amd64 / Linux / 5.4.72-microsoft-standard-WSL2	db_seeder_bash_complete_company_3.0.3_ws12
exasol	Exasol	v3.0.3	bash	client	active	company	3819	10.09.2021 12:12	10.09.2021 12:12	fjww-win-dt-21	32	amd64 / Linux / 5.4.72-microsoft-standard-WSL2	db_seeder_bash_complete_company_3.0.3_ws12
firebird	Firebird	v3.0.3	bash	client	active	company	88580	10.09.2021 12:13	10.09.2021 12:14	fjww-win-dt-21	32	amd64 / Linux / 5.4.72-microsoft-standard-WSL2	db_seeder_bash_complete_company_3.0.3_ws12
h2	H2 Database Engine	v3.0.3	bash	client	active - no choice	company	5882	10.09.2021 12:15	10.09.2021 12:15	fjww-win-dt-21	32	amd64 / Linux / 5.4.72-microsoft-standard-WSL2	db_seeder_bash_complete_company_3.0.3_ws12
h2_emb	H2 Database Engine	v3.0.3	bash	embedded	active - no choice	company	3413	10.09.2021 12:15	10.09.2021 12:15	fjww-win-dt-21	32	amd64 / Linux / 5.4.72-microsoft-standard-WSL2	db_seeder_bash_complete_company_3.0.3_ws12
hsqldb	HSQldb	v3.0.3	bash	client	active	company	17767	10.09.2021 12:15	10.09.2021 12:16	fjww-win-dt-21	32	amd64 / Linux / 5.4.72-microsoft-standard-WSL2	db_seeder_bash_complete_company_3.0.3_ws12
hsqldb_emb	HSQldb	v3.0.3	bash	embedded	active	company	2894	10.09.2021 12:16	10.09.2021 12:16	fjww-win-dt-21	32	amd64 / Linux / 5.4.72-microsoft-standard-WSL2	db_seeder_bash_complete_company_3.0.3_ws12
ibmdb2	IBM Db2 Database	v3.0.3	bash	client	active	company	23164	10.09.2021 12:21	10.09.2021 12:21	fjww-win-dt-21	32	amd64 / Linux / 5.4.72-microsoft-standard-WSL2	db_seeder_bash_complete_company_3.0.3_ws12
informix	IBM Informix	v3.0.3	bash	client	active	company	14934	10.09.2021 12:25	10.09.2021 12:25	fjww-win-dt-21	32	amd64 / Linux / 5.4.72-microsoft-standard-WSL2	db_seeder_bash_complete_company_3.0.3_ws12
mariaDB	MariaDB Server	v3.0.3	bash	client	active	company	3881	10.09.2021 12:26	10.09.2021 12:26	fjww-win-dt-21	32	amd64 / Linux / 5.4.72-microsoft-standard-WSL2	db_seeder_bash_complete_company_3.0.3_ws12
mimer	Mimer SQL	v3.0.3	bash	client	active	company	13368	10.09.2021 12:27	10.09.2021 12:27	fjww-win-dt-21	32	amd64 / Linux / 5.4.72-microsoft-standard-WSL2	db_seeder_bash_complete_company_3.0.3_ws12
monetdb	MonetDB	v3.0.3	bash	client	active	company	7916	10.09.2021 12:27	10.09.2021 12:27	fjww-win-dt-21	32	amd64 / Linux / 5.4.72-microsoft-standard-WSL2	db_seeder_bash_complete_company_3.0.3_ws12
mysql	MySQL Database	v3.0.3	bash	client	active	company	5995	10.09.2021 12:28	10.09.2021 12:28	fjww-win-dt-21	32	amd64 / Linux / 5.4.72-microsoft-standard-WSL2	db_seeder_bash_complete_company_3.0.3_ws12
mysql_trino	MySQL Database	v3.0.3	bash	trino	active - no choice	company	1999897	10.09.2021 12:30	10.09.2021 13:03	fjww-win-dt-21	32	amd64 / Linux / 5.4.72-microsoft-standard-WSL2	db_seeder_bash_complete_company_3.0.3_ws12
omniscl	OmniSciDB	v3.0.3	bash	client	active - no choice	company	36638	10.09.2021 13:05	10.09.2021 13:05	fjww-win-dt-21	32	amd64 / Linux / 5.4.72-microsoft-standard-WSL2	db_seeder_bash_complete_company_3.0.3_ws12
oracle	Oracle Database	v3.0.3	bash	client	active	company	12103	10.09.2021 13:06	10.09.2021 13:07	fjww-win-dt-21	32	amd64 / Linux / 5.4.72-microsoft-standard-WSL2	db_seeder_bash_complete_company_3.0.3_ws12
oracle_trino	Oracle Database	v3.0.3	bash	trino	active - no choice	company	1589925	10.09.2021 13:08	10.09.2021 13:35	fjww-win-dt-21	32	amd64 / Linux / 5.4.72-microsoft-standard-WSL2	db_seeder_bash_complete_company_3.0.3_ws12
percona	Percona Server for MySQL	v3.0.3	bash	client	active	company	13039	10.09.2021 13:36	10.09.2021 13:36	fjww-win-dt-21	32	amd64 / Linux / 5.4.72-microsoft-standard-WSL2	db_seeder_bash_complete_company_3.0.3_ws12
postgresql	PostgreSQL	v3.0.3	bash	client	active	company	4030	10.09.2021 13:37	10.09.2021 13:37	fjww-win-dt-21	32	amd64 / Linux / 5.4.72-microsoft-standard-WSL2	db_seeder_bash_complete_company_3.0.3_ws12
postgresql_trino	PostgreSQL	v3.0.3	bash	trino	active - no choice	company	522438	10.09.2021 13:38	10.09.2021 13:47	fjww-win-dt-21	32	amd64 / Linux / 5.4.72-microsoft-standard-WSL2	db_seeder_bash_complete_company_3.0.3_ws12
sqlite	SQLite	v3.0.3	bash	embedded	active - no choice	company	6516	10.09.2021 13:47	10.09.2021 13:47	fjww-win-dt-21	32	amd64 / Linux / 5.4.72-microsoft-standard-WSL2	db_seeder_bash_complete_company_3.0.3_ws12
sqlserver	MS SQL Server	v3.0.3	bash	client	active	company	4631	10.09.2021 13:48	10.09.2021 13:48	fjww-win-dt-21	32	amd64 / Linux / 5.4.72-microsoft-standard-WSL2	db_seeder_bash_complete_company_3.0.3_ws12
sqlserver_trino	MS SQL Server	v3.0.3	bash	trino	active - no choice	company	1050856	10.09.2021 13:49	10.09.2021 14:06	fjww-win-dt-21	32	amd64 / Linux / 5.4.72-microsoft-standard-WSL2	db_seeder_bash_complete_company_3.0.3_ws12
timescale	TimescaleDB	v3.0.3	bash	client	active	company	4094	10.09.2021 14:07	10.09.2021 14:07	fjww-win-dt-21	32	amd64 / Linux / 5.4.72-microsoft-standard-WSL2	db_seeder_bash_complete_company_3.0.3_ws12
yugabyte	YugabyteDB	v3.0.3	bash	client	active	company	13274	10.09.2021 14:08	10.09.2021 14:08	fjww-win-dt-21	32	amd64 / Linux / 5.4.72-microsoft-standard-WSL2	db_seeder_bash_complete_company_3.0.3_ws12
agens	AgensGraph	v3.0.3	bash	client	inactive	company	4018	10.09.2021 14:09	10.09.2021 14:09	fjww-win-dt-21	32	amd64 / Linux / 5.4.72-microsoft-standard-WSL2	db_seeder_bash_complete_company_3.0.3_ws12
cuprid	CUBRID	v3.0.3	bash	client	inactive	company	6320	10.09.2021 14:09	10.09.2021 14:09	fjww-win-dt-21	32	amd64 / Linux / 5.4.72-microsoft-standard-WSL2	db_seeder_bash_complete_company_3.0.3_ws12
derby	Apache Derby	v3.0.3	bash	client	inactive	company	15617	10.09.2021 14:10	10.09.2021 14:10	fjww-win-dt-21	32	amd64 / Linux / 5.4.72-microsoft-standard-WSL2	db_seeder_bash_complete_company_3.0.3_ws12
derby_emb	Apache Derby	v3.0.3	bash	embedded	inactive	company	13797	10.09.2021 14:10	10.09.2021 14:11	fjww-win-dt-21	32	amd64 / Linux / 5.4.72-microsoft-standard-WSL2	db_seeder_bash_complete_company_3.0.3_ws12
exasol	Exasol	v3.0.3	bash	client	inactive	company	3950	10.09.2021 14:13	10.09.2021 14:13	fjww-win-dt-21	32	amd64 / Linux / 5.4.72-microsoft-standard-WSL2	db_seeder_bash_complete_company_3.0.3_ws12
firebird	Firebird	v3.0.3	bash	client	inactive	company	70636	10.09.2021 14:14	10.09.2021 14:15	fjww-win-dt-21	32	amd64 / Linux / 5.4.72-microsoft-standard-WSL2	db_seeder_bash_complete_company_3.0.3_ws12
hsqldb	HSQldb	v3.0.3	bash	client	inactive	company	17986	10.09.2021 14:15	10.09.2021 14:16	fjww-win-dt-21	32	amd64 / Linux / 5.4.72-microsoft-standard-WSL2	db_seeder_bash_complete_company_3.0.3_ws12
ibmdb2	IBM Db2 Database	v3.0.3	bash	client	inactive	company	25507	10.09.2021 14:21	10.09.2021 14:21	fjww-win-dt-21	32	amd64 / Linux / 5.4.72-microsoft-standard-WSL2	db_seeder_bash_complete_company_3.0.3_ws12
informix	IBM Informix	v3.0.3	bash	client	inactive	company	15007	10.09.2021 14:25	10.09.2021 14:25	fjww-win-dt-21	32	amd64 / Linux / 5.4.72-microsoft-standard-WSL2	db_seeder_bash_complete_company_3.0.3_ws12
mariaDB	MariaDB Server	v3.0.3	bash	client	inactive	company	7312	10.09.2021 14:26	10.09.2021 14:26	fjww-win-dt-21	32	amd64 / Linux / 5.4.72-microsoft-standard-WSL2	db_seeder_bash_complete_company_3.0.3_ws12
mimer	Mimer SQL	v3.0.3	bash	client	inactive	company	13970	10.09.2021 14:27	10.09.2021 14:27	fjww-win-dt-21	32	amd64 / Linux / 5.4.72-microsoft-standard-WSL2	db_seeder_bash_complete_company_3.0.3_ws12
monetdb	MonetDB	v3.0.3	bash	client	inactive	company	6856	10.09.2021 14:27	10.09.2021 14:27	fjww-win-dt-21	32	amd64 / Linux / 5.4.72-microsoft-standard-WSL2	db_seeder_bash_complete_company_3.0.3_ws12
mysql	MySQL Database	v3.0.3	bash	client	inactive	company	18638	10.09.2021 14:28	10.09.2021 14:29	fjww-win-dt-21	32	amd64 / Linux / 5.4.72-microsoft-standard-WSL2	db_seeder_bash_complete_company_3.0.3_ws12
oracle	Oracle Database	v3.0.3	bash	client	inactive	company	15614	10.09.2021 14:30	10.09.2021 14:30	fjww-win-dt-21	32	amd64 / Linux / 5.4.72-microsoft-standard-WSL2	db_seeder_bash_complete_company_3.0.3_ws12
percona	Percona Server for MySQL	v3.0.3	bash	client	inactive	company	18334	10.09.2021 14:31	10.09.2021 14:31	fjww-win-dt-21	32	amd64 / Linux / 5.4.72-microsoft-standard-WSL2	db_seeder_bash_complete_company_3.0.3_ws12
postgresql	PostgreSQL	v3.0.3	bash	client	inactive	company	4008	10.09.2021 14:32	10.09.2021 14:32	fjww-win-dt-21	32	amd64 / Linux / 5.4.72-microsoft-standard-WSL2	db_seeder_bash_complete_company_3.0.3_ws12
sqlserver	MS SQL Server	v3.0.3	bash	client	inactive	company	4606	10.09.2021 14:33	10.09.2021 14:33	fjww-win-dt-21	32	amd64 / Linux / 5.4.72-microsoft-standard-WSL2	db_seeder_bash_complete_company_3.0.3_ws12
timescale	TimescaleDB	v3.0.3	bash	client	inactive	company	3996	10.09.2021 14:33	10.09.2021 14:33	fjww-win-dt-21	32	amd64 / Linux / 5.4.72-microsoft-standard-WSL2	db_seeder_bash_complete_company_3.0.3_ws12
yugabyte	YugabyteDB	v3.0.3	bash	client	inactive	company	13343	10.09.2021 14:34	10.09.2021 14:34	fjww-win-dt-21	32	amd64 / Linux / 5.4.72-microsoft-standard-WSL2	db_seeder_bash_complete_company_3.0.3_ws12
agens	AgensGraph	v3.0.3	cmd	client	active	company	4128	10.09.2021 09:14	10.09.2021 09:15	fjww-win-dt-21	32	amd64 / Windows 10 / 10.0	db_seeder_cmd_complete_company_3.0.3_win10
cockroach	CockroachDB	v3.0.3	cmd	client	active - no choice	company	5630	10.09.2021 09:15	10.09.2021 09:15	fjww-win-dt-21	32	amd64 / Windows 10 / 10.0	db_seeder_cmd_complete_company_3.0.3_win10
cratedb	CrateDB	v3.0.3	cmd	client	active - no choice	company	5342	10.09.2021 09:16	10.09.2021 09:16	fjww-win-dt-21	32	amd64 / Windows 10 / 10.0	db_seeder_cmd_complete_company_3.0.3_win10

File name syntax: `db_seeder_summary_<first DBSeeder version>-<current DBSeeder version>.csv|tsv`

Explanation for the columns:

- ticker symbol** - internal abbreviation used for the database
- DBMS** - official DBMS name
- version** - DBSeeder version
- creator** - shell environment: `bash` or `cmd`
- db type** - client version, embedded version or via trino
- constraints** - DML operations with enabled (**active** and **active - no choice**) or disabled (**inactive**) constraints (foreign, primary and unique key)
- schema** - identification term for the scheme definition used: **company** or **syntax**
- total ms** - total time of DDL and DML operations in milliseconds
- start time** - date and time when the database operations were started
- end time** - date and time when the database operations were completed
- host name** - name of the computer connected to a computer network
- no. cores** - number of CPU cores used
- operating system**
- file name** - name of the file with the source data
- total DDL ms** - total time of DDL operations in milliseconds
- drop constr. ms** - total time to drop all constraints
- add constr. ms** - total time to add the previously dropped constraints
- total DML ms** - total time of DML operations in milliseconds

5. RDBMS Specific Technical Details

DBeaver is a great tool to analyze the database content. In the file directory **resources/dbeaver** you will also find a file exported from DBeaver with the connection parameters currently used in DBSeeder.

- [AgensGraph](#)
- [Apache Derby](#)
- [CockroachDB](#)
- [CrateDB](#)
- [CUBRID](#)
- [Exasol](#)
- [Firebird](#)
- [H2 Database Engine](#)

- [HSQLDB](#)
- [IBM Db2 Database](#)
- [IBM Informix](#)
- [MariaDB Server](#)
- [Mimer SQL](#)
- [MonetDB](#)
- [MySQL Database](#)
- [OmniSciDB](#)
- [Oracle Database](#)
- [Percona Server for MySQL](#)
- [PostgreSQL](#)
- [SQL Server](#)
- [SQLite](#)
- [TimescaleDB](#)
- [trino](#)
- [VoltDB](#)
- [YugabyteDB](#)

5.1 AgensGraph

- **data types:**

DBSeeder Type	AgensGraph Database Type
BIGINT	BIGINT
BLOB	BYTEA
CLOB	TEXT
TIMESTAMP	TIMESTAMP
VARCHAR	VARCHAR

- **DDL syntax:**

- CREATE DATABASE: see PostgreSQL
- CREATE SCHEMA: see PostgreSQL
- CREATE TABLE: see PostgreSQL
- CREATE USER: see PostgreSQL

- **Docker image (latest):**

- pull command: `docker pull bitnine/agensgraph:v2.1.3`
- [DockerHub](#)

- **encoding:** see PostgreSQL

- **issue tracking:** [GitHub](#)

- **JDBC driver (latest):**

- version 1.4.2-c1
- [Maven repository](#)

- **source code:** [GitHub](#)

5.2 Apache Derby

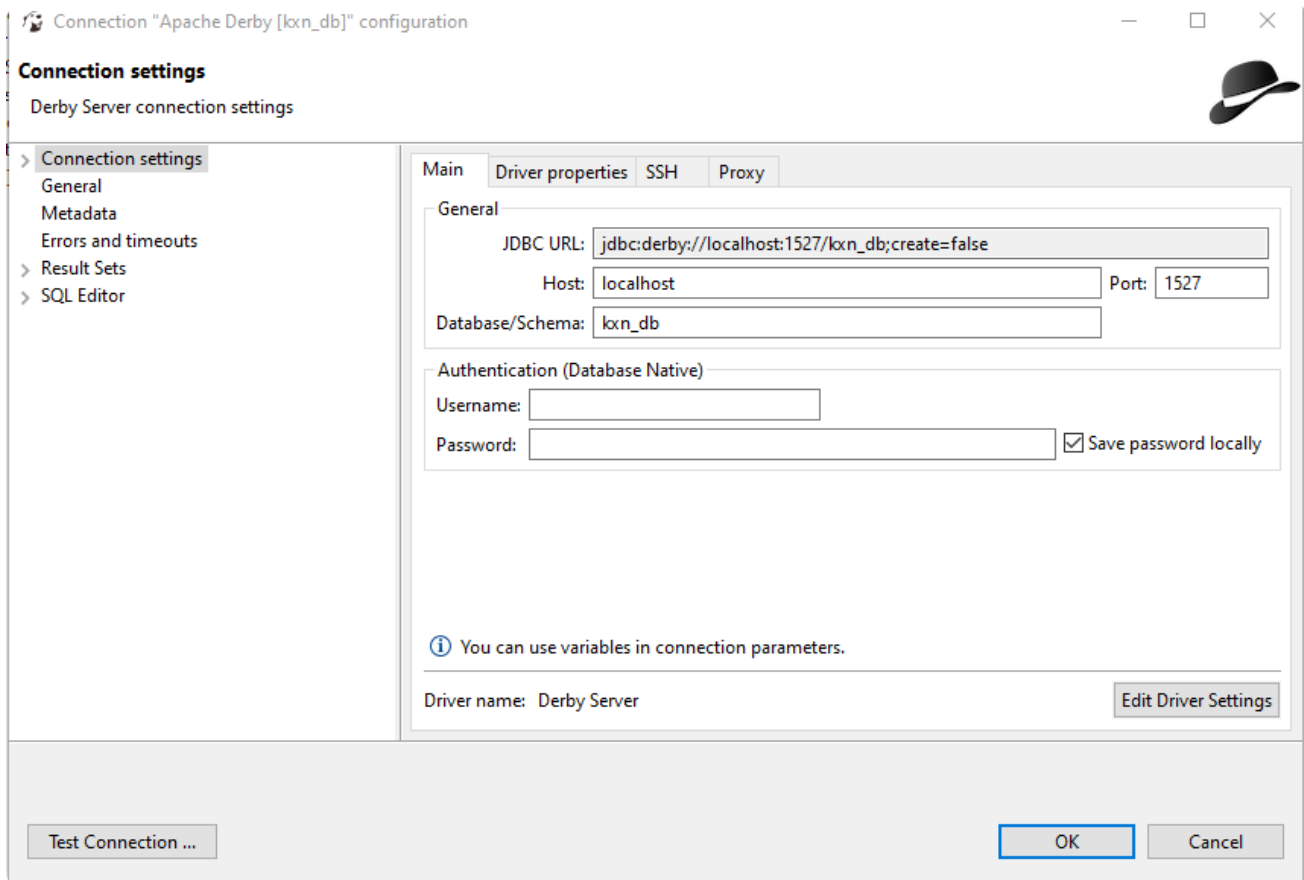
- **data types:**

DBSeeder Type	Apache Derby Type
BIGINT	BIGINT
BLOB	BLOB
CLOB	CLOB
TIMESTAMP	TIMESTAMP

DBSeeder Type Apache Derby Type

VARCHAR VARCHAR

- **DDL syntax:**
 - CREATE DATABASE - n/a
 - [CREATE SCHEMA](#)
 - [CREATE TABLE](#)
 - CREATE USER - n/a
- **Docker image (latest - only client version):**
 - pull command: `docker pull konnexionsgmbh/apache_derby:10.15.2.0`
 - [DockerHub](#)
- **encoding:** by using the following JVM parameter: `-Dderby.ui.codeset=UTF8`
- **issue tracking:** [Jira](#)
- **JDBC driver (latest):**
 - version 10.15.2.0
 - client version: [Maven repository](#)
 - embedded version: [Maven repository](#)
- **source code:** [Apache Derby](#)
- **DBeaver database connection settings:**
 - client version:



5.3 CockroachDB

- **data types:**

DBSeeder Type CockroachDB Type

BIGINT INT

DBSeeder Type	CockroachDB Type
BLOB	BYTES
CLOB	STRING
TIMESTAMP	TIMESTAMP
VARCHAR	STRING

- **DDL syntax:**
 - [CREATE DATABASE](#)
 - [CREATE SCHEMA](#)
 - [CREATE TABLE](#)
 - [CREATE USER](#)
- **Docker image (latest):**
 - pull command: `docker pull cockroachdb/cockroach:v21.1.8`
 - [DockerHub](#)
- **encoding:** by default `utf8` encoding
- **issue tracking:** [GitHub](#)
- **JDBC driver (latest):**
 - same as PostgreSQL
- **privileged database access:** user `root`
- **source code:** [GitHub](#)
- **DBeaver database connection settings:**

Connection "CockroachDB [kxn_user - kxn_db]" configuration

Connection settings
CockroachDB connection settings

▼ Connection settings

- Initialization
- Shell Commands
- Client identification
- Transactions
- General
- Metadata
- Errors and timeouts
- > Data editor
- > SQL Editor

Main CockroachDB Driver properties SSH Proxy SSL

Server

Host: localhost Port: 26257

Database: kxn_db

Authentication

Authentication: Database Native

Username: kxn_user

Password: ☒ Save password locally

Advanced

User role: Local Client:

You can use variables in connection parameters.

Driver name: CockroachDB [Edit Driver Settings](#)

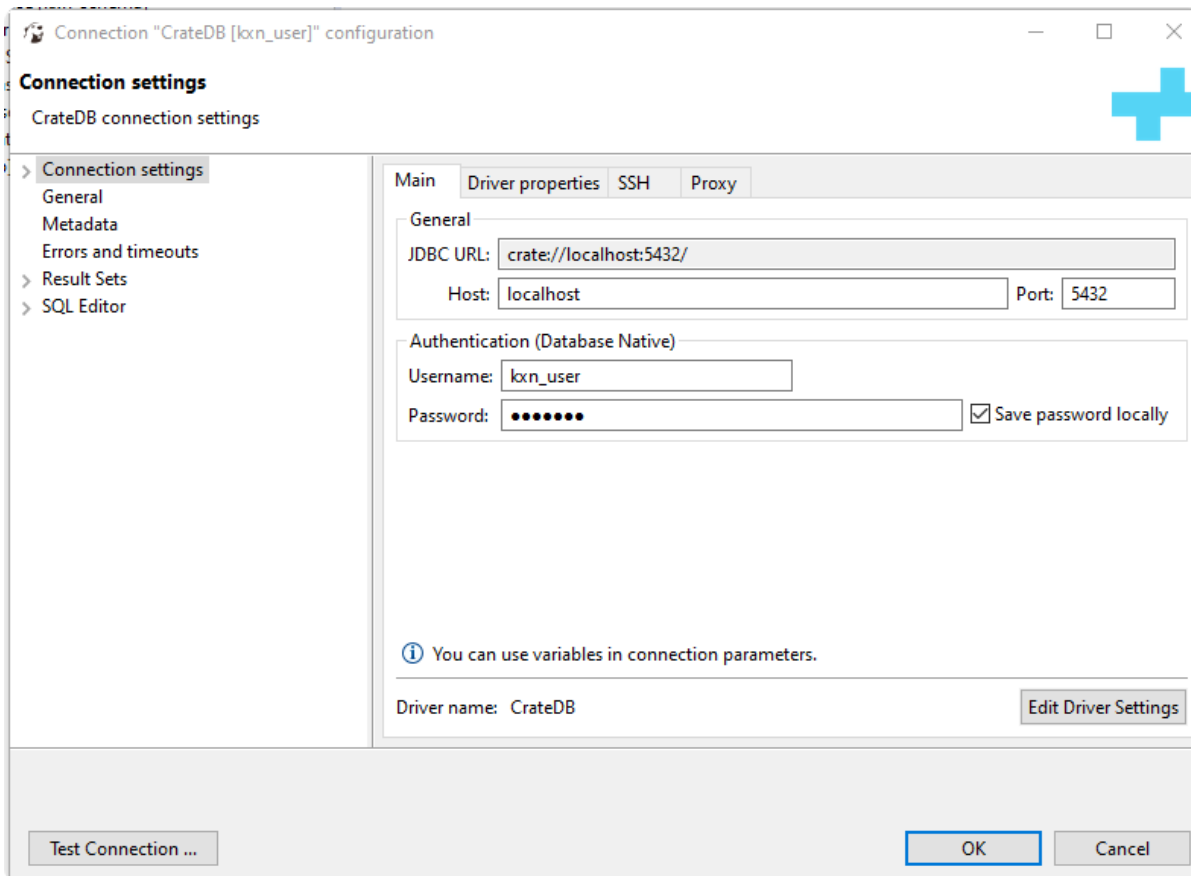
[Test Connection ...](#) [OK](#) [Cancel](#)

5.4 CrateDB

- **data types:**

DBSeeder Type	CrateDB Type
BIGINT	BIGINT
BLOB	OBJECT
CLOB	TEXT
TIMESTAMP	TIMESTAMP
VARCHAR	TEXT

- **DDL syntax:**
 - CREATE DATABASE - n/a
 - CREATE SCHEMA - n/a
 - [CREATE TABLE](#)
 - [CREATE USER](#)
- **Docker image (latest):**
 - pull command: `docker pull crate:4.6.1`
 - [DockerHub](#)
- **encoding:** by default `utf8` encoding
- **issue tracking:** [GitHub](#)
- **JDBC driver (latest):**
 - version 2.6.0
 - [JFrog Bintray repository](#)
- **privileged database access:** user `crate`
- **restrictions:**
 - no constraints (e.g. foreign keys or unique keys)
 - no transaction concept
 - no triggers
 - only a very proprietary BLOB implementation
- **source code:** [GitHub](#)
- **DBeaver database connection settings:**



5.5 CUBRID

- **data types:**

DBSeeder Type	CUBRID Type
BIGINT	INT
BLOB	BLOB
CLOB	CLOB
TIMESTAMP	TIMESTAMP
VARCHAR	VARCHAR

- **DDL syntax:**

- CREATE DATABASE - n/a
- CREATE SCHEMA - n/a
- [CREATE TABLE](#)
- [CREATE USER](#)

- **Docker image (latest):**

- pull command: `docker pull cubrid/cubrid:11.0`
- [DockerHub](#)

- **encoding:** by specifying after the database name when database is created: `kxn_db de_DE.utf8`

- **issue tracking:**

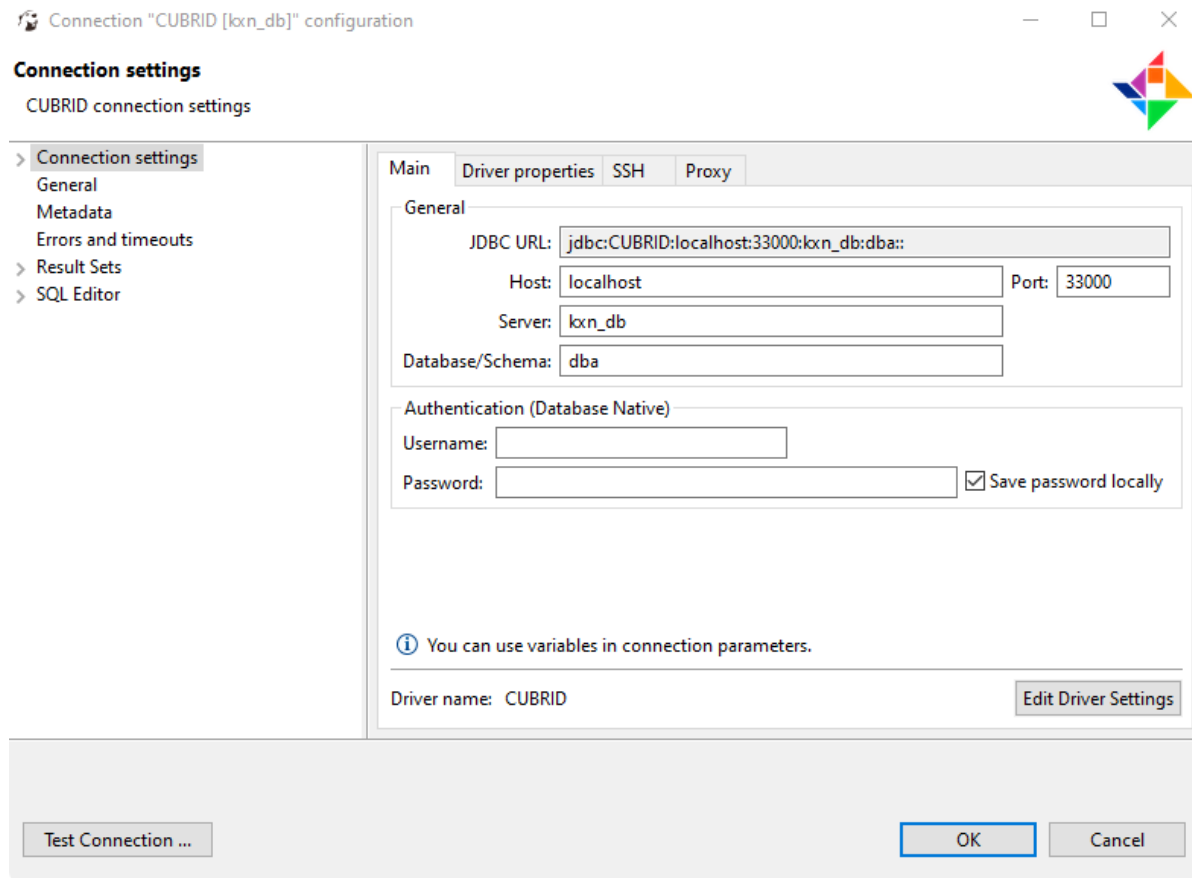
- [Jira](#)

- **JDBC driver (latest):**

- version 11.0.1.0291
- [Maven repository](#)

- **privileged database access:** users `DBA` and `PUBLIC`

- **restrictions:** no full UTF-8 support
- **source code:** [GitHub](#)
- **DBeaver database connection settings:**



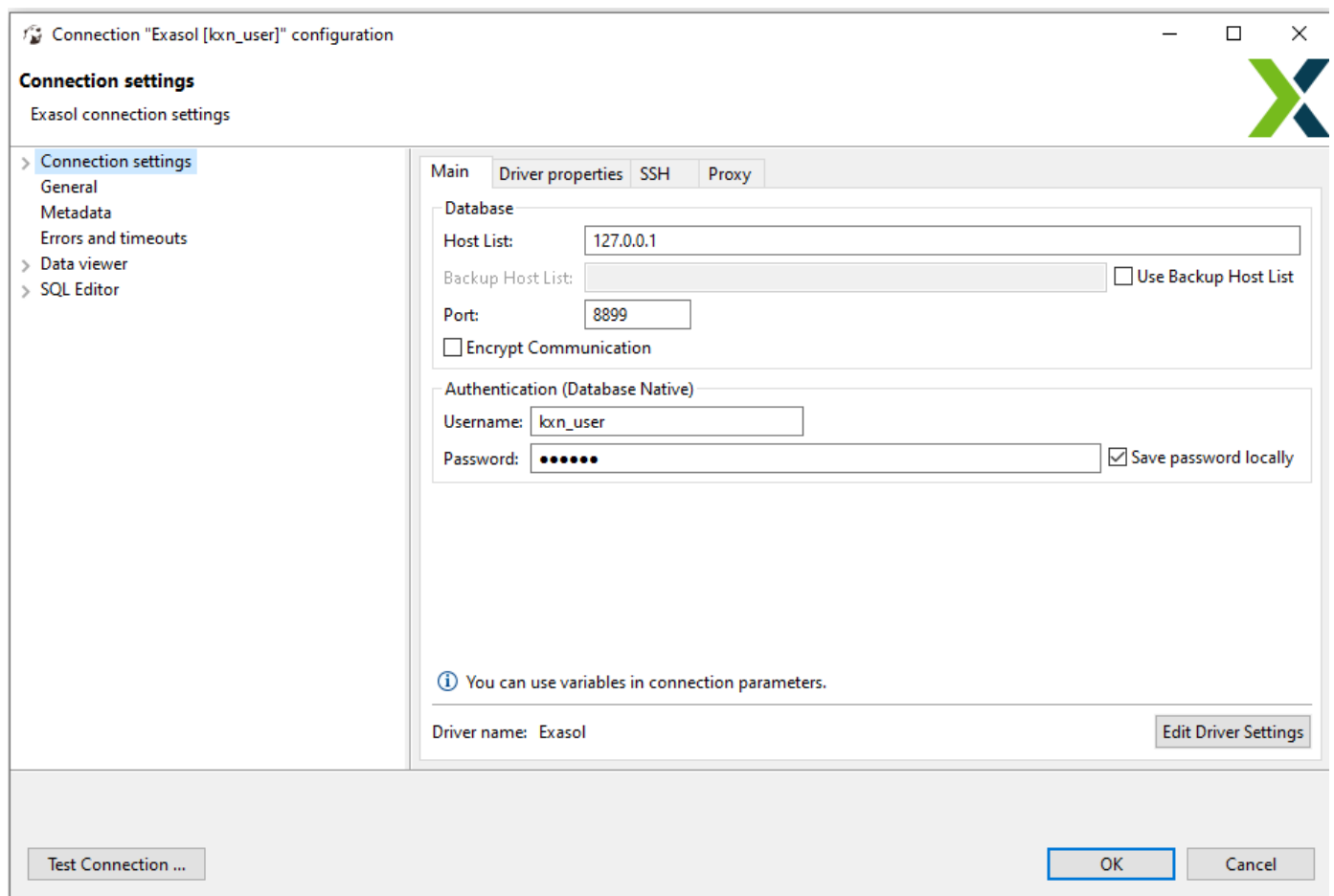
5.6 Exasol

- **data types:**

DBSeeder Type	Exasol Type
BIGINT	BIGINT
BLOB	VARCHAR(2000000)
CLOB	VARCHAR(2000000)
TIMESTAMP	TIMESTAMP
VARCHAR	VARCHAR

- **DDL syntax:**
 - CREATE DATABASE - n/a
 - [CREATE SCHEMA](#)
 - [CREATE TABLE](#)
 - [CREATE USER](#)
- **Docker image (latest):**
 - pull command: `docker pull exasol/docker-db:7.1.0-d1`
 - [DockerHub](#)
- **JDBC driver (latest):**
 - version 7.1.0
 - [Maven repository](#)
- **privileged database access:** user `sys` password `exasol`

- **restrictions:**
 - no unique key constraints
- **DBeaver database connection settings:**



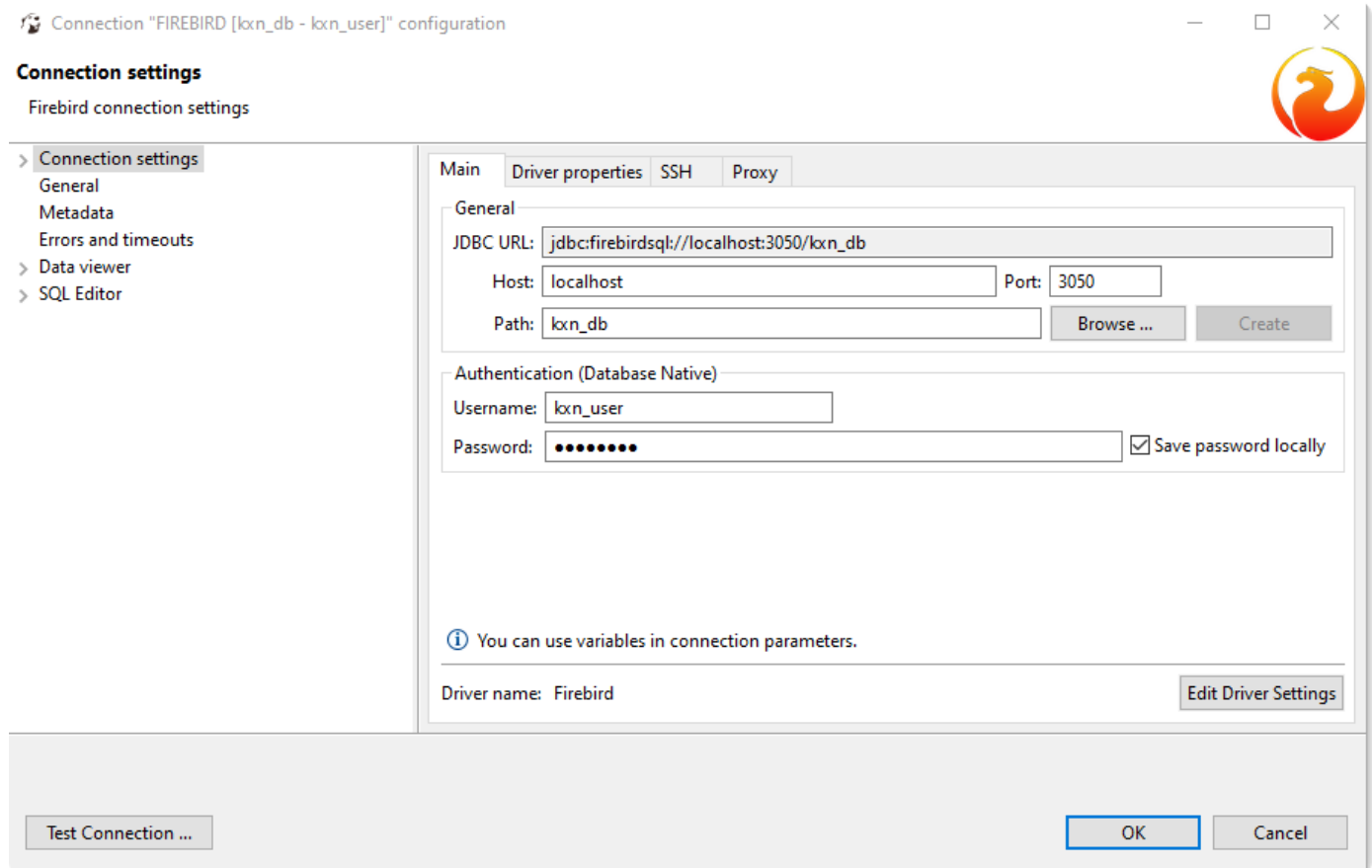
5.7 Firebird

- **data types:**

DBSeeder Type	Firebird Type
BIGINT	INTEGER
BLOB	BLOB
CLOB	BLOB SUB_TYPE 1
TIMESTAMP	TIMESTAMP
VARCHAR	VARCHAR

- **DDL syntax:**
 - [CREATE DATABASE](#)
 - CREATE SCHEMA - n/a
 - [CREATE TABLE](#)
 - [CREATE USER](#)
- **Docker image (latest):**
 - pull command: `docker pull jacobalberty/firebird:v4.0.0`
 - [DockerHub](#)
- **encoding:** by using the following JDBC URL parameter: `encoding=UTF8`
- **issue tracking:** [GitHub](#)

- **JDBC driver (latest):**
 - version 4.0.3.java11
 - [Maven repository](#)
- **privileged database access:** user **SYSDBA**
- **source code:** [GitHub](#)
- **DBeaver database connection settings:**



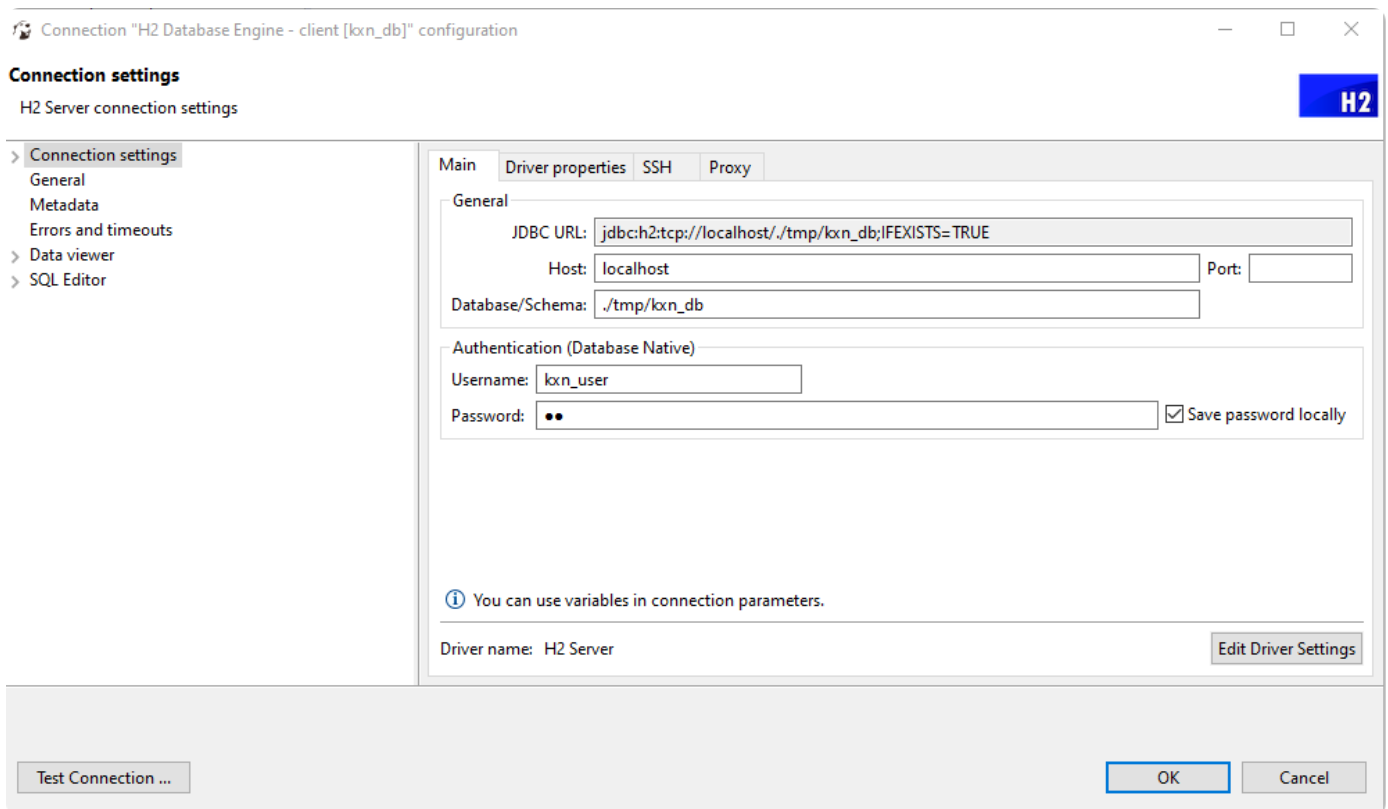
5.8 H2 Database Engine

- **data types:**

DBSeeder Type	H2 Database Engine Type
BIGINT	BIGINT
BLOB	BLOB
CLOB	CLOB
TIMESTAMP	TIMESTAMP
VARCHAR	VARCHAR

- **DDL syntax:**
 - CREATE DATABASE - n/a
 - [CREATE SCHEMA](#)
 - [CREATE TABLE](#)
 - [CREATE USER](#)
- **Docker image (latest):**
 - pull command: `docker pull konnexionsgmbh/h2_database_engine:1.4.200`
 - [DockerHub](#)

- **encoding:** H2 internally uses Unicode, and supports all character encoding systems and character sets supported by the virtual machine you use.
- **issue tracking:** [GitHub](#)
- **JDBC driver (latest):**
 - version 1.4.200
 - [Maven repository](#)
- **privileged database access:** user **sa**
- **source code:** [GitHub](#)
- **DBeaver database connection settings:**
 - client version:



5.9 HSQLDB

- **data types:**

DBSeeder Type	HSQLDB Type
BIGINT	BIGINT
BLOB	BLOB
CLOB	CLOB
TIMESTAMP	TIMESTAMP
VARCHAR	VARCHAR

- **DDL syntax:**
 - CREATE DATABASE - n/a
 - [CREATE SCHEMA](#)
 - [CREATE TABLE](#)
 - [CREATE USER](#)
- **Docker image (latest):**

- pull command: `docker pull konnexionsgmbh/hypersql_database:2.6.0`
- [DockerHub](#)
- **encoding:** by using the following system property `sqlfile.charset=UTF-8`.
- **issue tracking:** [SourceForge](#)
- **JDBC driver (latest):**
 - version 2.6.0
 - [Maven repository](#)
- **privileged database access:** user `SA`
- **source code:** [SourceForge](#)
- **DBeaver database connection settings:**
 - client version:

Connection "HyperSQL - client [kxn_db]" configuration

Connection settings

HSQldb / HSQL Server connection settings

Connection settings

HSQldb / HSQL Server connection settings

General

JDBC URL: `jdbc:hsqldb:hsqldb://localhost:9001/kxn_db`

Host: `localhost` Port: `9001`

Database/Schema: `kxn_db`

Authentication (Database Native)

Username: `KXN_USER`

Password: `.....` ☒ Save password locally

You can use variables in connection parameters.

Driver name: HSQldb / HSQL Server

Test Connection ... OK Cancel

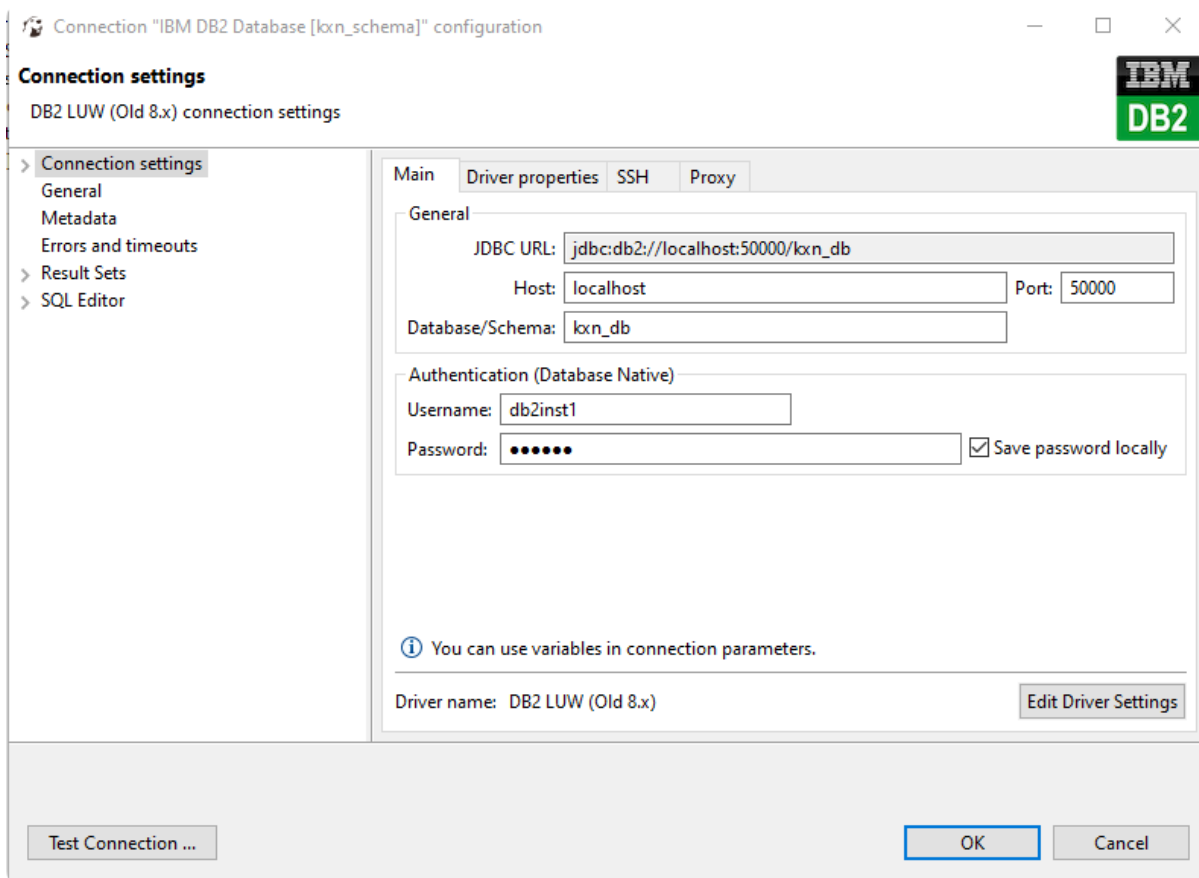
5.10 IBM Db2 Database

- **data types:**

DBSeeder Type	IBM Db2 Database Type
BIGINT	BIGINT
BLOB	BLOB
CLOB	CLOB
TIMESTAMP	TIMESTAMP
VARCHAR	VARCHAR

- **DDL syntax:**
 - [CREATE DATABASE](#)
 - [CREATE SCHEMA](#)

- [CREATE TABLE](#)
- [CREATE USER](#)
- **Docker image (latest):**
 - pull command: `docker pull ibmcom/db2:11.5.5.1`
 - [DockerHub](#)
- **encoding:**
 - by using the CCSID clause in the CREATE statements for any of the following objects:
 - Database
 - Table space
 - Table
 - procedure or function
- **JDBC driver (latest):**
 - version 11.5.6 .0
 - [Maven repository](#)
- **privileged database access:** user `db2inst1`
- **restrictions:** the IBM Db2 DBMS only accepts operating system accounts as database users
- **DBeaver database connection settings:**



5.11 IBM Informix

- **data types:**

DBSeeder Type	IBM Informix Database Type
BIGINT	BIGINT
BLOB	BLOB
CLOB	CLOB
TIMESTAMP	DATETIME YEAR TO FRACTION

DBSeeder Type	IBM Informix Database Type
VARCHAR	VARCHAR (1-254) / LVARCHAR

- **DDL syntax:**

- [CREATE DATABASE](#)
- CREATE SCHEMA - n/a
- [CREATE TABLE](#)
- [CREATE USER](#)

- **Docker image (latest):**

- pull command: `docker pull ibmcom/informix-developer-database:14.10.FC5DE-rhm`
- [DockerHub](#)

- **encoding:**

- code-set conversion value is extracted from the DB_LOCALE value specified at the time the connection is made

- **JDBC driver (latest):**

- version 4.50.4.1
- [Maven repository](#)

- **privileged database access:**

- user `informix`
- password `in4mix`
- database / schema `sysmaster`
- INFORMIXSERVER `informix`

- **restrictions:**

- the IBM Informix DBMS only accepts operating system accounts or users mapped to operating system accounts as database users
- no named constraints in ALTER TABLE ADD CONSTRAINT

- **DBeaver database connection settings:**

Connection "IBM Informix [kxn_db]" configuration

Connection settings
Informix connection settings

General
Metadata
Errors and timeouts
Data viewer
SQL Editor

Main | Driver properties | SSH | Proxy

General

JDBC URL: `jdbc:informix-sqli://localhost:9088/kxn_db:INFORMIXSERVER=informix`

Host: `localhost` Port: `9088`

Server: `informix`

Database/Schema: `kxn_db`

Authentication (Database Native)

Username: `informix`

Password: `.....` ☒ Save password locally

Information icon You can use variables in connection parameters.

Driver name: Informix [Edit Driver Settings](#)

[Test Connection ...](#) [OK](#) [Cancel](#)

5.12 MariaDB Server

- **data types:**

DBSeeder Type	MariaDB Type
BIGINT	BIGINT
BLOB	LONGBLOB
CLOB	LONGTEXT
TIMESTAMP	DATETIME
VARCHAR	VARCHAR

- **DDL syntax:**

- [CREATE DATABASE](#)
- CREATE SCHEMA - n/a
- [CREATE TABLE](#)
- [CREATE USER](#)

- **Docker image (latest):**

- pull command: `docker pull mariadb:10.6.4-focal`
- [DockerHub](#)

- **encoding:**

- server level: `SET character_set_server = 'latin2';`
- database level: `CHARACTER SET = 'keybcs2'`
- table level: `CHARACTER SET 'utf8'`
- column level: `CHARACTER SET 'greek'`

- **issue tracking:** [Jira](#)

- **JDBC driver (latest):**

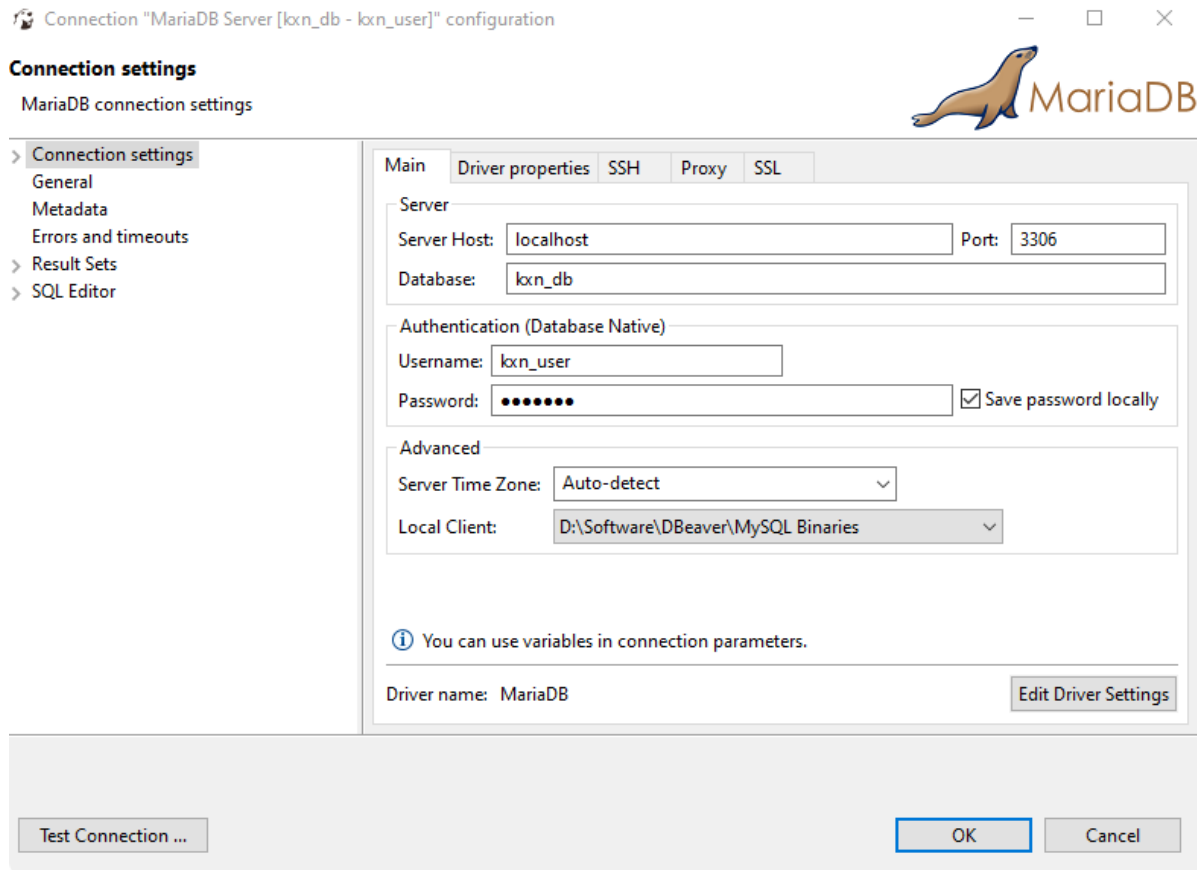
- version 2.7.4
- [Maven repository](#)

- **privileged database access:**

- user: `mysql`
- password: `root`

- **source code:** [GitHub](#)

- **DBeaver database connection settings:**



5.13 Mimer SQL

- **data types:**

DBSeeder Type	MimerSQL Type
BIGINT	BIGINT
BLOB	BLOB
CLOB	CLOB
TIMESTAMP	TIMESTAMP
VARCHAR	NVARCHAR

- **DDL syntax:**

- [CREATE DATABASE](#)
- CREATE SCHEMA - n/a
- [CREATE TABLE](#)
- [CREATE USER](#)

- **Docker image (latest):**

- pull command: `docker pull mimersql/mimersql_v11.0.5a`
- [DockerHub](#)

- **encoding:** NCHAR, NVARCHAR

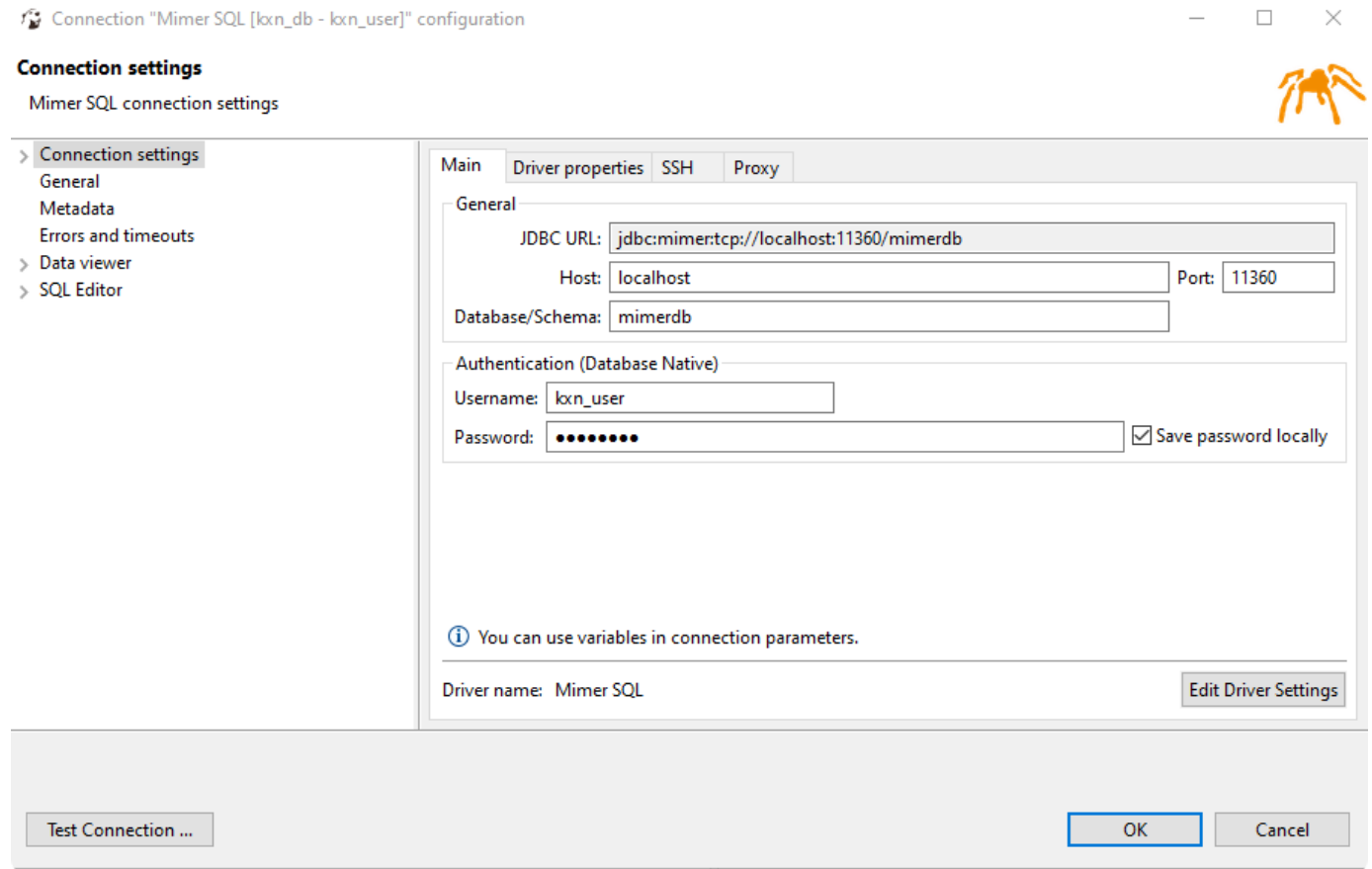
- **JDBC driver (latest):**

- version 3.41a
- [Mimer Website](#)

- **privileged database access:**

- database: `mimerdb`
- user: `SYSADM`

- **DBeaver database connection settings:**



5.14 MonetDB

- **data types:**

DBSeeder Type	MonetDB Type
BIGINT	BIGINT
BLOB	BLOB
CLOB	CLOB
TIMESTAMP	TIMESTAMP
VARCHAR	VARCHAR

- **DDL syntax:**

- CREATE DATABASE - n/a
- [CREATE SCHEMA](#)
- [CREATE TABLE](#)
- [CREATE USER](#)

- **Docker image (latest):**

- pull command: `docker pull monetdb/monetdb:Jul2021`
- [DockerHub](#)

- **encoding:** no special configuration should be needed

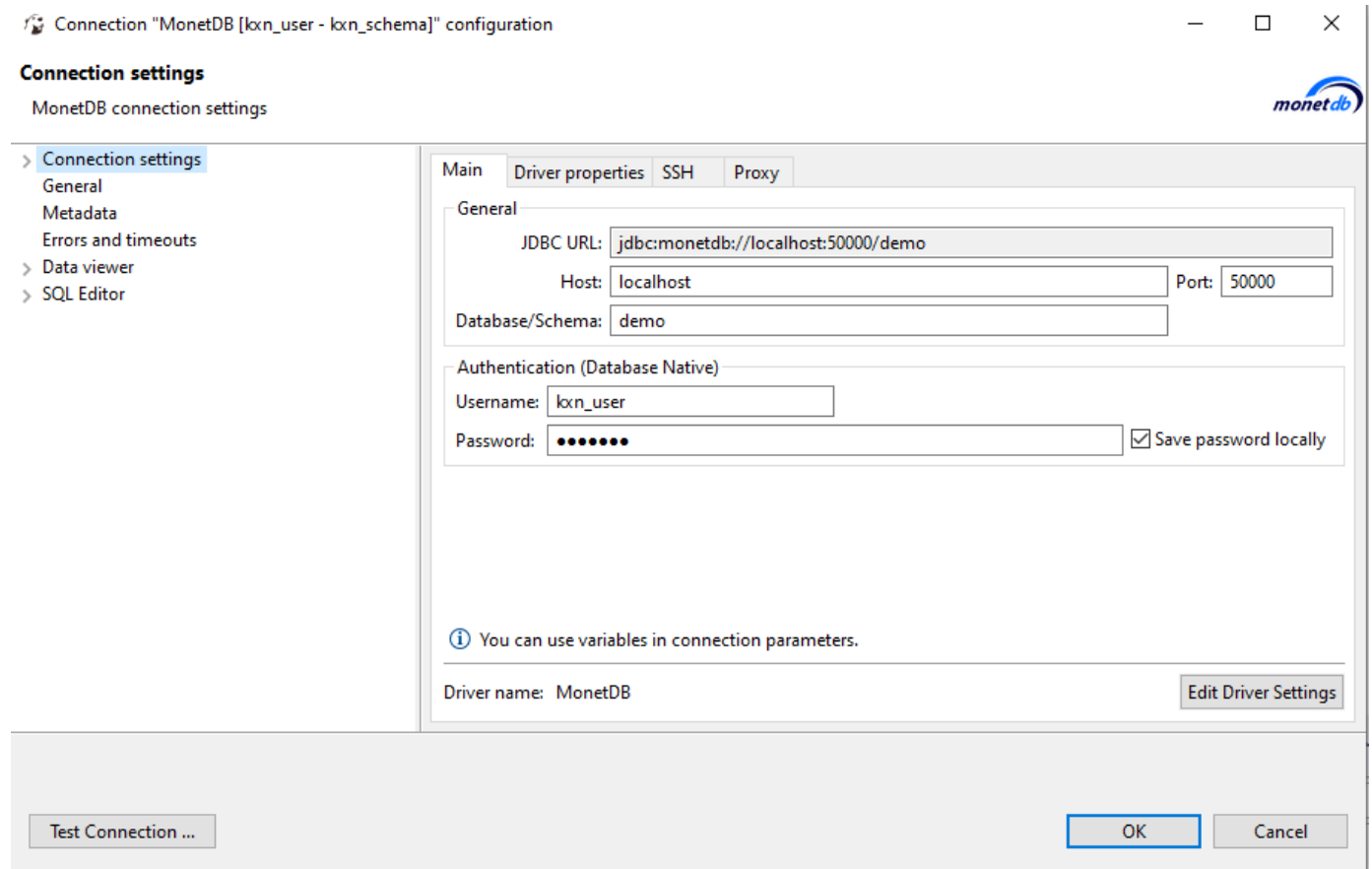
- **issue tracking:** [GitHub](#)

- **JDBC driver (latest):**

- version 3.1.jre8
- [MonetDB Java Download Area](#)

- **privileged database access:**

- database: **demo**
- user: **monetdb**
- password: **monetdb**
- **source code:** [GitHub](#)
- **DBeaver database connection settings:**



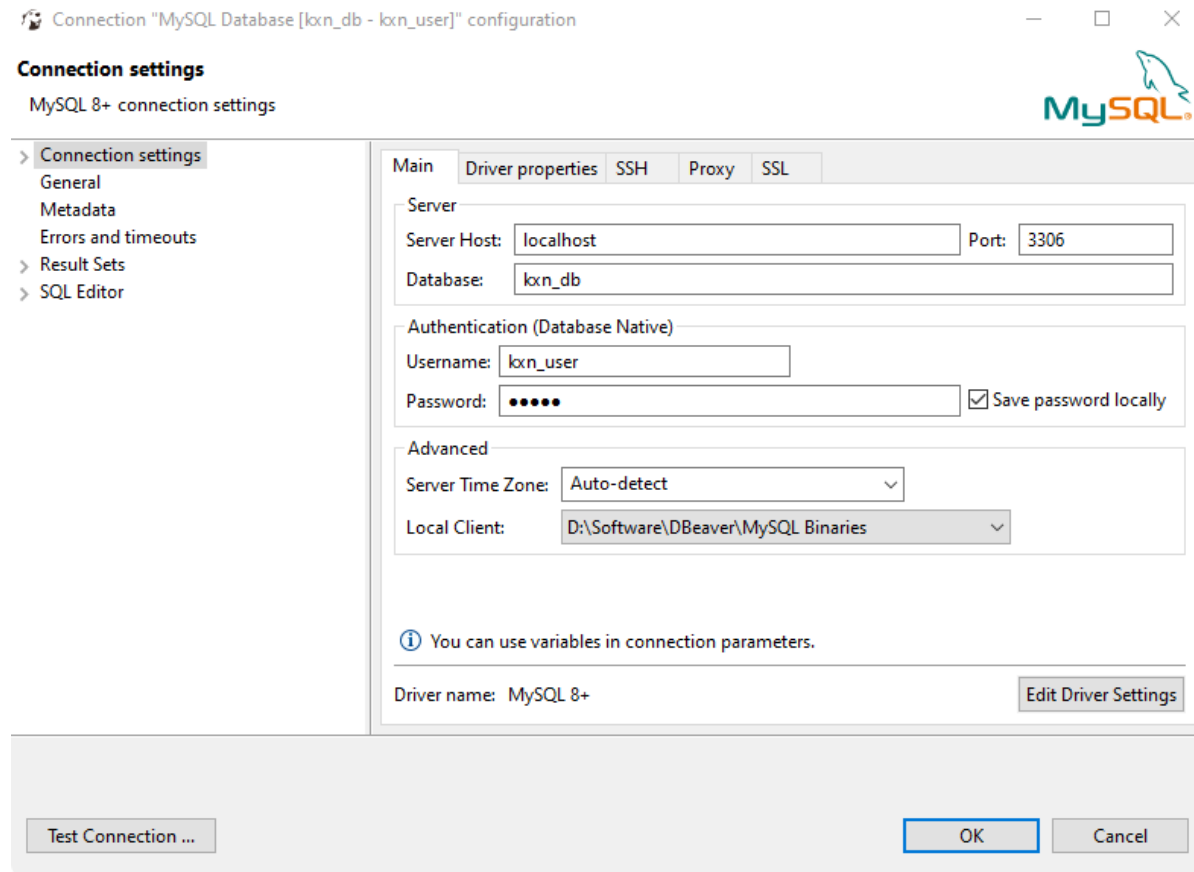
5.15 MySQL Database

- **data types:**

DBSeeder Type	MySQL Database Type
BIGINT	BIGINT
BLOB	LONGBLOB
CLOB	LONGTEXT
TIMESTAMP	DATETIME
VARCHAR	VARCHAR

- **DDL syntax:**
 - [CREATE DATABASE](#)
 - CREATE SCHEMA - n/a
 - [CREATE TABLE](#)
 - [CREATE USER](#)
- **Docker image (latest):**
 - pull command: **docker pull mysql:8.0.26**
 - [DockerHub](#)
- **encoding:** for applications that store data using the default MySQL character set and collation (utf8mb4, utf8mb4_0900_ai_ci), no special configuration should be needed
- **JDBC driver (latest):**

- version 8.0.26
- [Maven repository](#)
- **privileged database access:**
 - database: `sys`
 - user: `root`
- **source code:** [GitHub](#)
- **DBeaver database connection settings:**



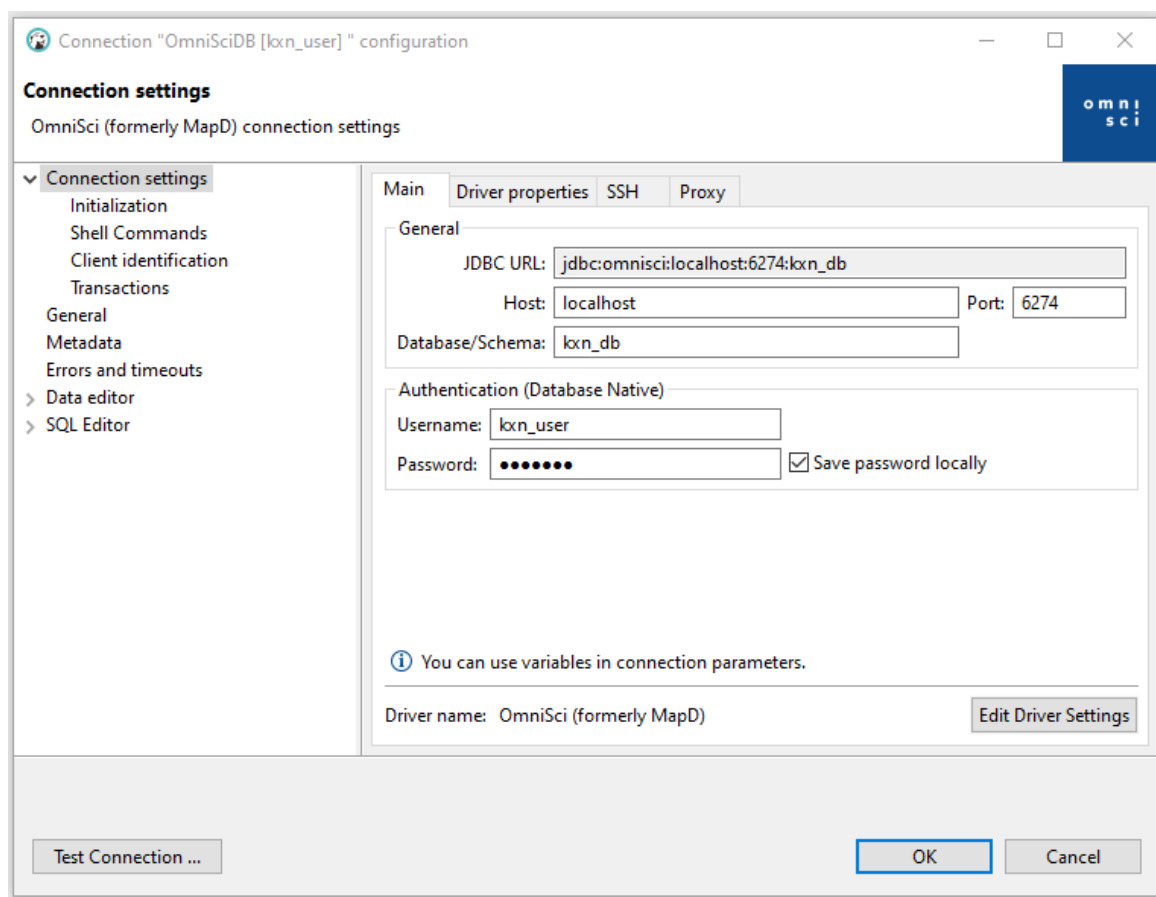
5.16 OmniSciDB

- **data types:**

DBSeeder Type	OmniSciDB Type
BIGINT	BIGINT
BLOB	TEXT ENCODING NONE
CLOB	TEXT ENCODING NONE
TIMESTAMP	TIMESTAMP(0)
VARCHAR	TEXT ENCODING NONE

- **DDL syntax:**
 - [CREATE DATABASE](#)
 - CREATE SCHEMA - n/a
 - [CREATE TABLE](#)
 - [CREATE USER](#)
- **Docker image (latest):**
 - pull command: `docker pull omnisci/core-os-cpu`
 - [DockerHub](#)

- **encoding:** no special configuration should be needed
- **issue tracking:** [GitHub](#)
- **JDBC driver (latest):**
 - version 5.7.0
 - [Maven repository](#)
- **privileged database access:**
 - database: **omnisci**
 - user: **admin**
- **restrictions:**
 - column and table names case sensitive
 - max. column length 32767 bytes
 - no binary columns
 - no constraints, e.g. unique keys
 - no foreign / referential keys
 - no primary key
 - no triggers
- **source code:** [GitHub](#)
- **DBeaver database connection settings:**



5.17 Oracle Database

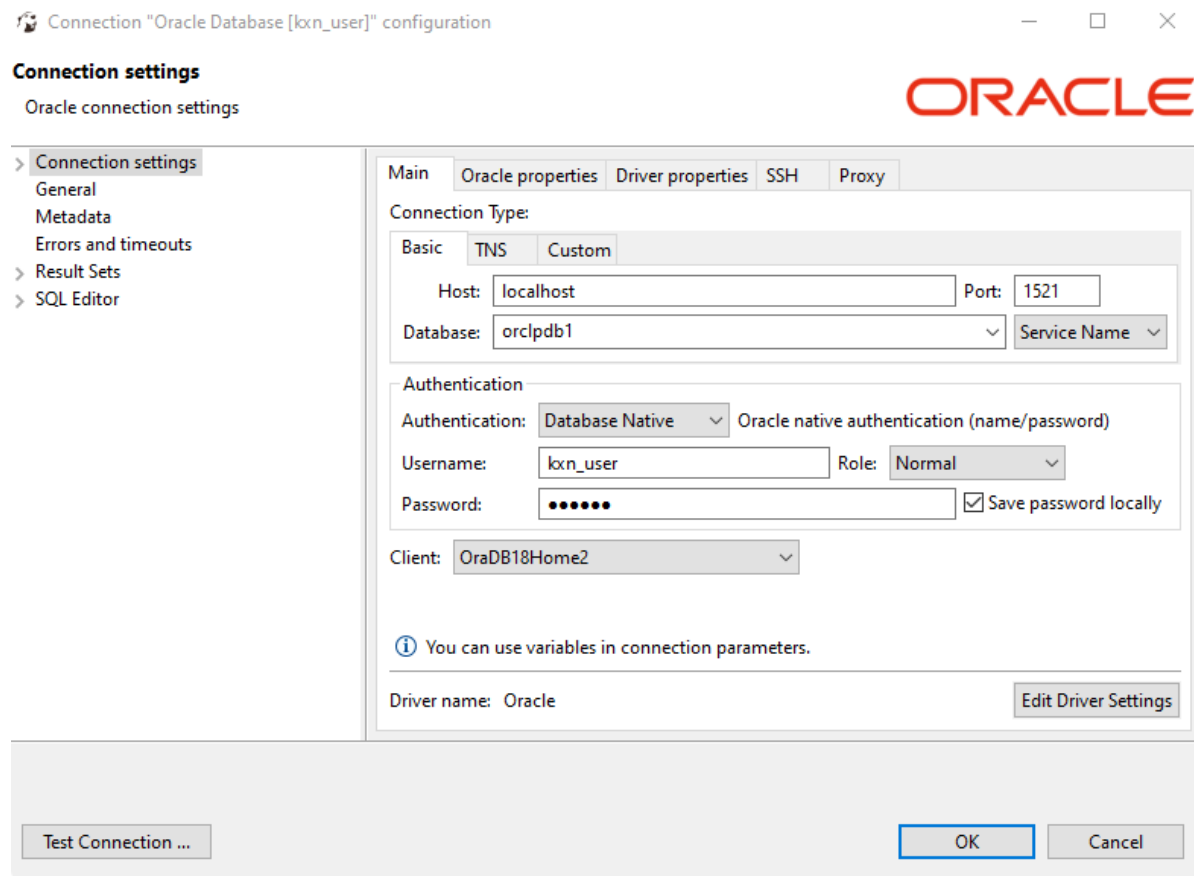
- **data types:**

DBSeeder Type	Oracle Database Type
BIGINT	NUMBER
BLOB	BLOB
CLOB	CLOB

DBSeeder Type	Oracle Database Type
TIMESTAMP	TIMESTAMP
VARCHAR	VARCHAR2

- **DDL syntax:**
 - CREATE DATABASE - n/a
 - CREATE SCHEMA - n/a
 - [CREATE TABLE](#)
 - [CREATE USER](#)
- **Docker image:** [DockerHub](#)
- **encoding:** since Oracle Database 12c Release 2 the default database character set used is the Unicode character set AL32UTF8
- **JDBC driver (latest):**
 - version 21.3.0.0
 - [Maven repository](#)
- **privileged database access:**
 - database: `orclpdb1`
 - user: `SYS AS SYSDBA`

- **DBeaver database connection settings:**



5.18 Percona Server for MySQL

- **data types:**

DBSeeder Type	Percona Server Type
BIGINT	BIGINT
BLOB	LONGBLOB
CLOB	LONGTEXT

DBSeeder Type	Percona Sercver Type
TIMESTAMP	DATETIME
VARCHAR	VARCHAR

- **DDL syntax:**
 - CREATE DATABASE: see MySQL Database
 - CREATE SCHEMA - n/a
 - CREATE TABLE: see MySQL Database
 - CREATE USER: see MySQL Database
- **Docker image (latest):**
 - pull command: `docker pull percona/percona-server:8.0.25-15`
 - [DockerHub](#)
- **encoding:** for applications that store data using the default MySQL character set and collation (utf8mb4, utf8mb4_0900_ai_ci), no special configuration should be needed
- **issue tracking:** [Jira](#)
- **JDBC driver (latest):**
 - same as MySQL
- **privileged database access:**
 - database: `sys`
 - user: `root`
- **source code:** [GitHub](#)
- **DBeaver database connection settings:**

Connection "Percona Server for MySQL [lxn_db - lxn_user]" configuration

Connection settings
MySQL connection settings

MySQL connection settings

Connection settings

Initialization

Shell Commands

Client identification

Transactions

General

Metadata

Errors and timeouts

> Data editor

> SQL Editor

Main Driver properties SSH Proxy SSL

Server

Server Host: localhost Port: 3306

Database: lxn_db

Authentication (Database Native)

Username: lxn_user

Password: •••••••• ☒ Save password locally

Advanced

Server Time Zone: Auto-detect

Local Client:

i You can use variables in connection parameters.

Driver name: MySQL [Edit Driver Settings](#)

Test Connection ... OK Cancel

5.19 PostgreSQL

- **data types:**

DBSeeder Type	PostgreSQL Type
BIGINT	BIGINT
BLOB	BYTEA
CLOB	TEXT
TIMESTAMP	TIMESTAMP
VARCHAR	VARCHAR

- **DDL syntax:**

- [CREATE DATABASE](#)
- [CREATE SCHEMA](#)
- [CREATE TABLE](#)
- [CREATE USER](#)

- **Docker image (latest):**

- pull command: `docker pull postgres:13.4-alpine`
- [DockerHub](#)

- **encoding:** when creating the database: `CREATE DATABASE testdb WITH ENCODING 'EUC_KR' ...`

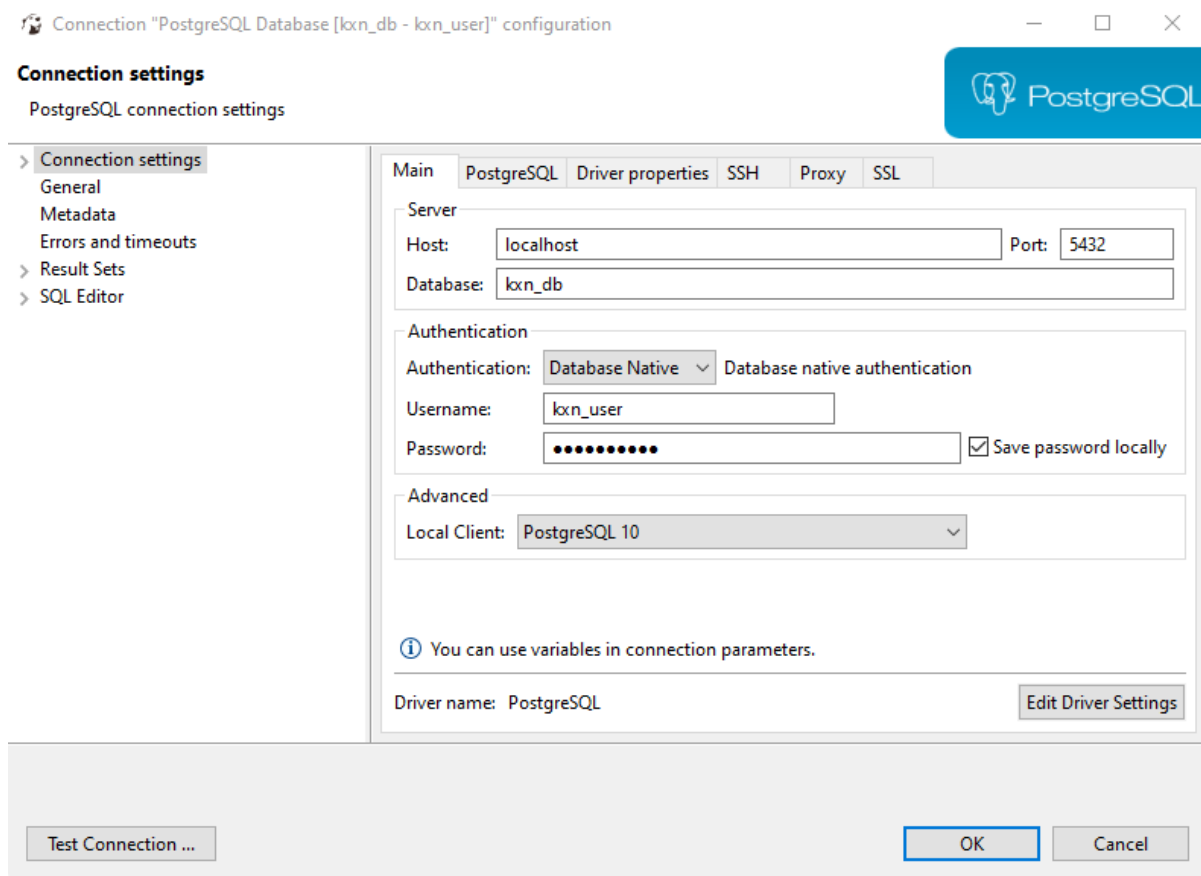
- **issue tracking:** [PostgreSQL](#)

- **JDBC driver (latest):**

- version 42.2.23
- [Maven repository](#)

- **source code:** [GitHub](#)

- **DBeaver database connection settings:**



5.20 SQL Server

- **data types:**

DBSeeder Type	SQL Server Type
BIGINT	BIGINT
BLOB	VARBINARY (MAX)
CLOB	VARCHAR (MAX)
TIMESTAMP	DATETIME2
VARCHAR	VARCHAR

- **DDL syntax:**

- [CREATE DATABASE](#)
- [CREATE SCHEMA](#)
- [CREATE TABLE](#)
- [CREATE USER](#)

- **Docker image (latest):**

- pull command: `docker pull mcr.microsoft.com/mssql/server:2019-CU12-ubuntu-20.04`
- [DockerHub](#)

- **encoding:** to use the UTF-8 collations that are available in SQL Server 2019 (15.x), you must select UTF-8 encoding-enabled collations (_UTF8)

- **JDBC driver (latest):**

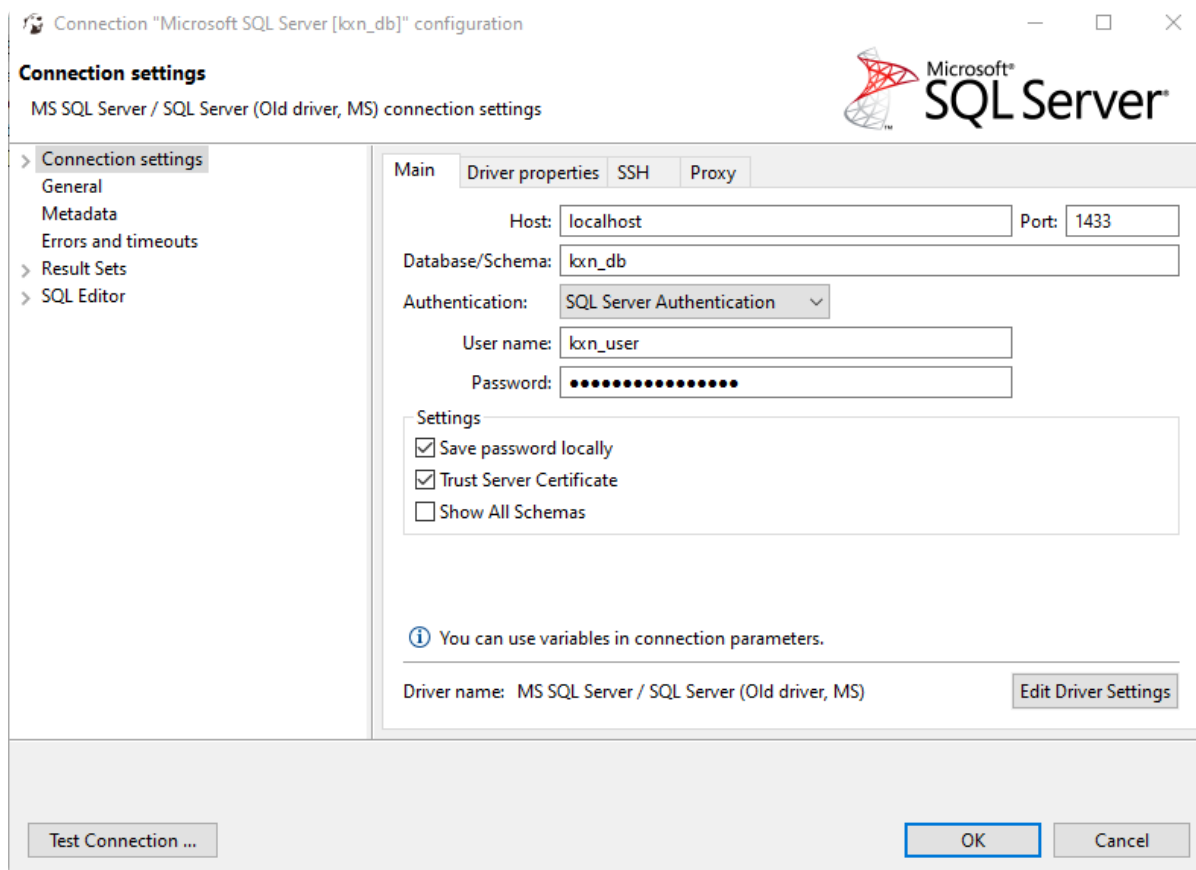
- version 9.4.0.jre16
- [Maven repository](#)

- **privileged database access:**

- database: `master`
- user: `sa`

- **restrictions:** no full UTF-8 support in the given Docker images

- **DBeaver database connection settings:**



5.21 SQLite

- **data types:**

DBSeeder Type	SQLite Type
BIGINT	INTEGER
BLOB	BLOB
CLOB	CLOB
TIMESTAMP	DATETIME
VARCHAR	VARCHAR2

- **DDL syntax:**

- CREATE DATABASE - n/a
- CREATE SCHEMA - n/a
- [CREATE TABLE](#)
- CREATE USER - n/a

- **encoding:** by using the following parameter: `PRAGMA encoding='UTF-8';`

- **issue tracking:** [SQLite](#)

- **JDBC driver (latest):**

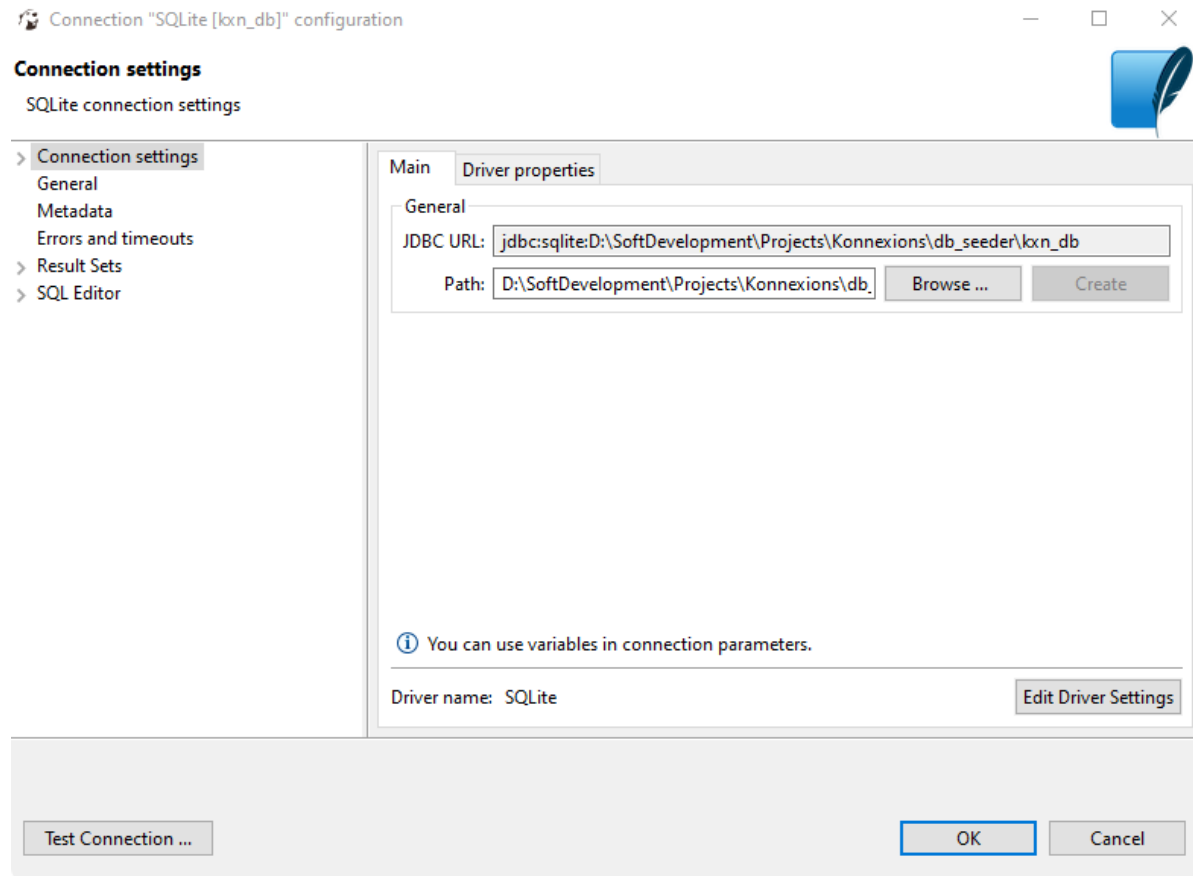
- version 3.36.0.3
- [Maven repository](#)
- determines also the DBMS version

- **restrictions:**

- no Docker image necessary, hence not available
- no user management

- **source code:** [SQLite](#)

- **DBeaver database connection settings:**



5.22 TimescaleDB

- **data types:**

DBSeeder Type	AgensGraph Database Type
BIGINT	BIGINT
BLOB	BYTEA
CLOB	TEXT
TIMESTAMP	TIMESTAMP
VARCHAR	VARCHAR

- **DDL syntax:**

- CREATE DATABASE: see PostgreSQL
- CREATE SCHEMA: see PostgreSQL
- CREATE TABLE: see PostgreSQL
- CREATE USER: see PostgreSQL

- **Docker image (latest):**

- pull command: `docker pull timescale/timescaledb:2.4.1-pg13`
- [DockerHub](#)

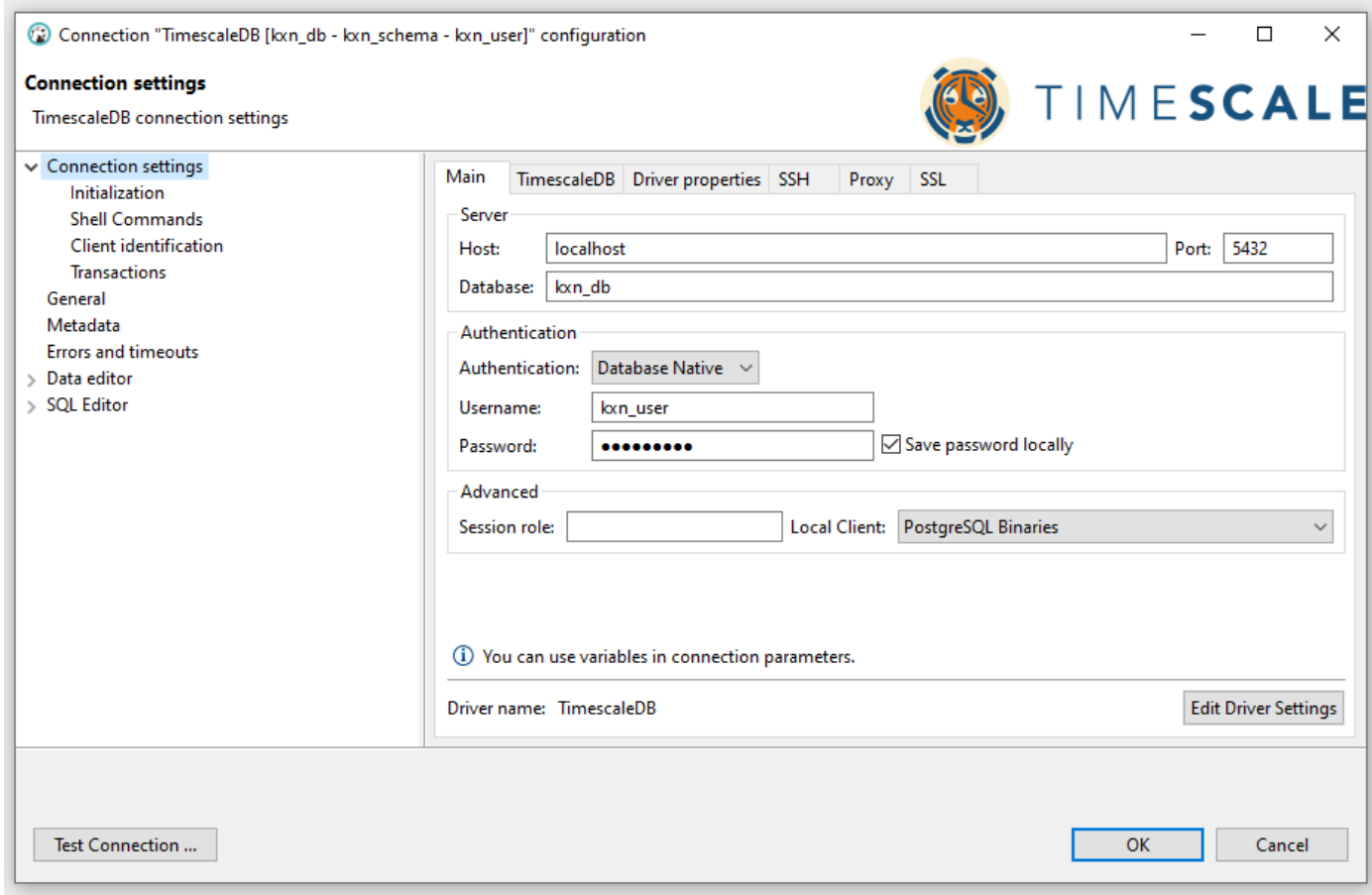
- **encoding:** see PostgreSQL

- **issue tracking:** [GitHub](#)

- **JDBC driver (latest):**

- same as PostgreSQL

- **source code:** [GitHub](#)



5.23 trino

- **data types:**

DBSeeder Type	trino Type
BIGINT	BIGINT
BLOB	BLOB
CLOB	CLOB
TIMESTAMP	TIMESTAMP
VARCHAR	VARCHAR

- **DDL syntax:**

- CREATE DATABASE - n/a
- [CREATE SCHEMA](#)
- [CREATE TABLE](#)
- CREATE USER - n/a

- **Docker image (latest):**

- pull command: `docker pull trinodb/trino:361`
- [DockerHub](#)

- **encoding:** full support of UTF-8 (see [here](#))

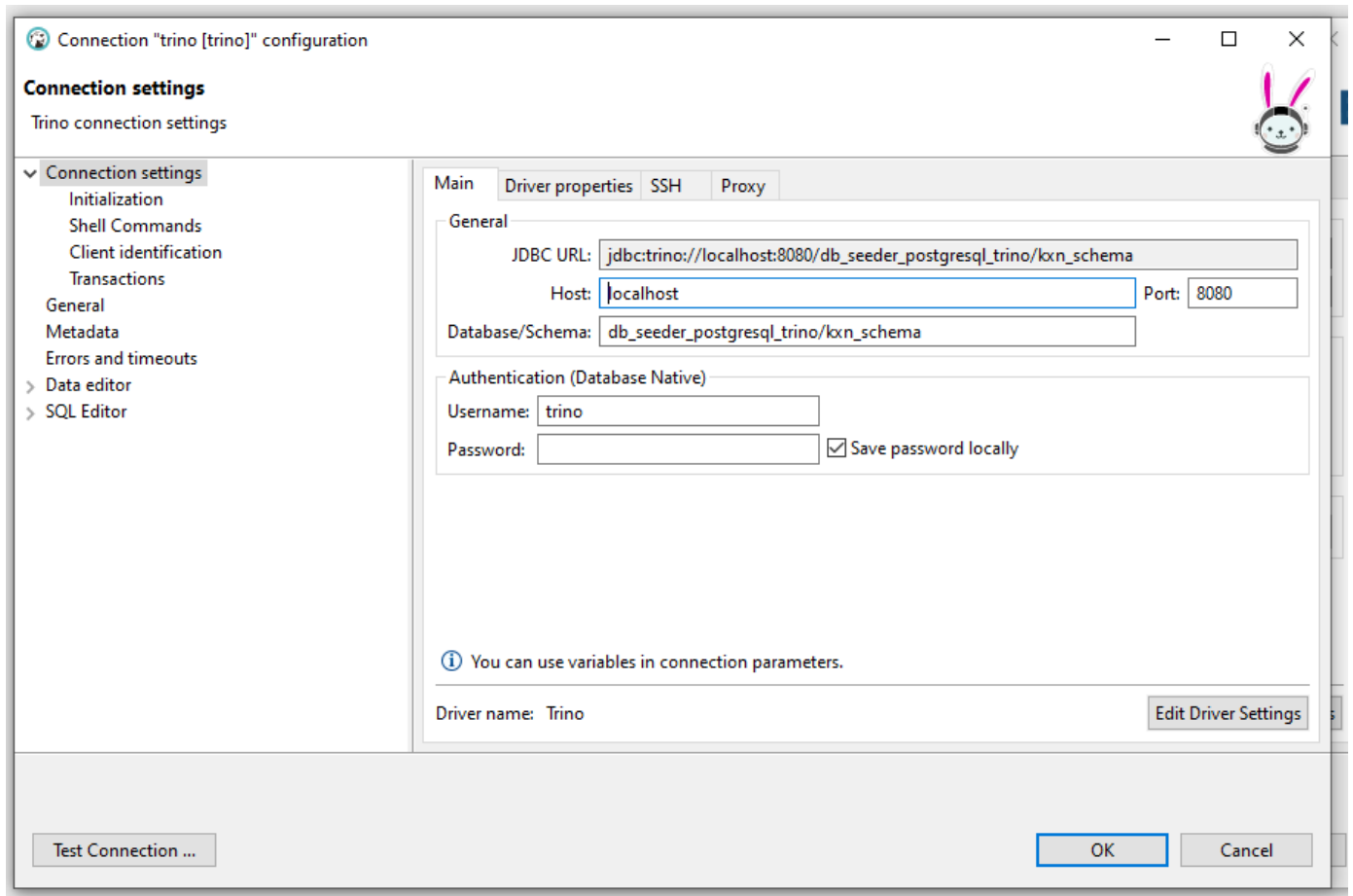
- **issue tracking:** [GitHub](#)

- **JDBC driver (latest):**

- version 361
- [Maven repository](#)

- **source code:** [GitHub](#)

- **DBeaver database connection settings:**



5.24 VoltDB

- **data types:**

DBSeeder Type	VoltDB Type
BIGINT	BIGINT
BLOB	VARBINARY(1048576)
CLOB	VARCHAR(1048576)
TIMESTAMP	TIMESTAMP
VARCHAR	VARCHAR

- **DDL syntax:**

- CREATE DATABASE - n/a
- CREATE SCHEMA - n/a
- [CREATE TABLE](#)
- CREATE USER - n/a

- **Docker image (latest):**

- pull command: `docker pull voltdb/voltdb-community:9.2.1`
- [DockerHub](#)

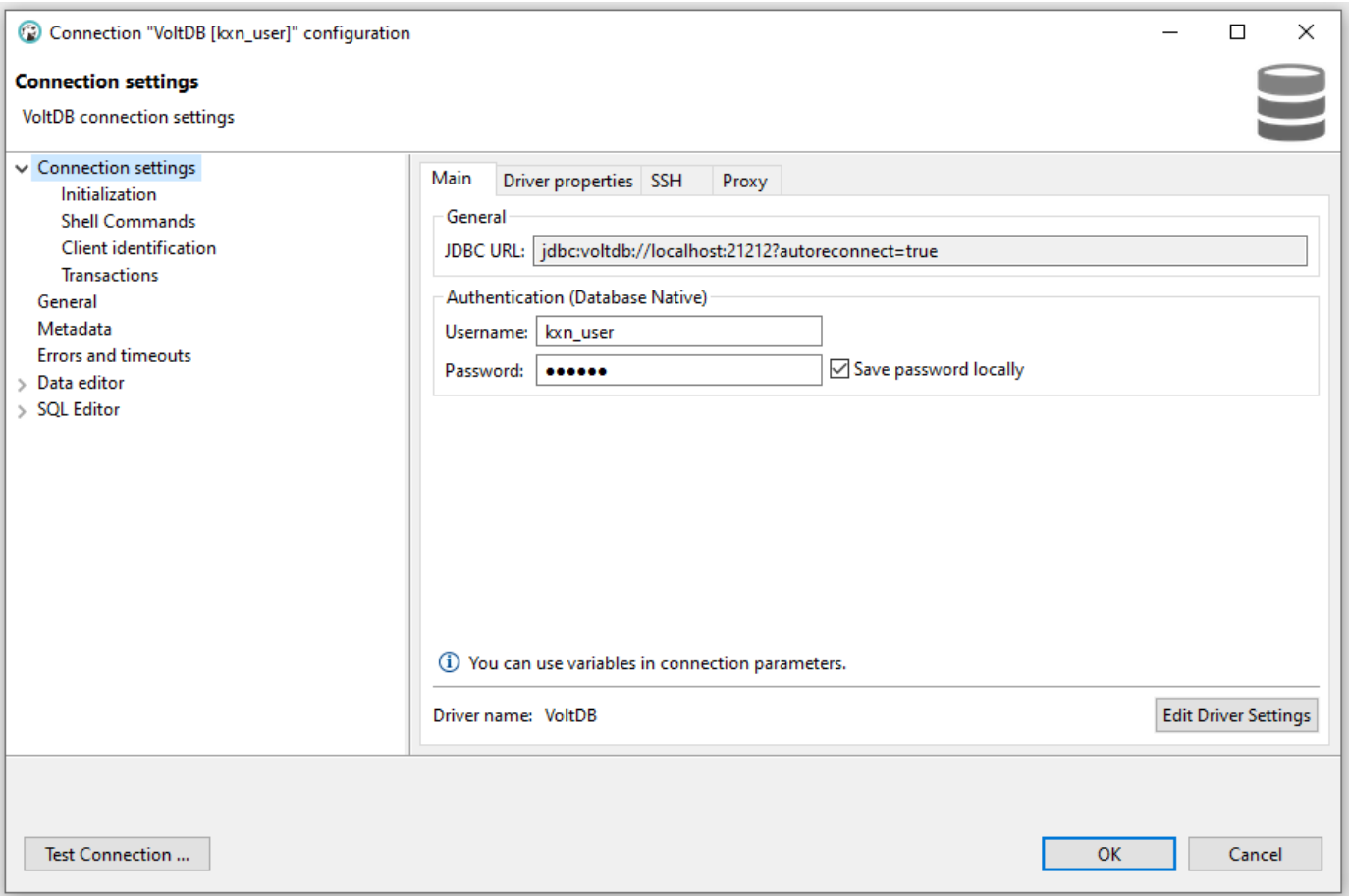
- **issue tracking:** [Jira](#)

- **JDBC driver (latest):**

- version 11.0
- [Maven repository](#)

- **restrictions:** no support of autoincrement, check constraints or foreign keys

- **source code:** [GitHub](#)
- **DBeaver database connection settings:**



5.25 YugabyteDB

- **data types:**

DBSeeder Type	YugabyteDB Database Type
BIGINT	BIGINT
BLOB	BYTEA
CLOB	TEXT
TIMESTAMP	TIMESTAMP
VARCHAR	VARCHAR

- **DDL syntax:**
 - [CREATE DATABASE](#)
 - [CREATE SCHEMA](#)
 - [CREATE TABLE](#)
 - [CREATE USER](#)
- **Docker image (latest):**
 - pull command: `docker pull yugabytedb/yugabyte:2.9.0.0-b4`
 - [DockerHub](#)
- **encoding:** see PostgreSQL
- **issue tracking:** [GitHub](#)
- **JDBC driver (latest):**
 - version 42.2.7-yb-3

- [Maven repository](#)
- **source code:** [GitHub](#)
- **DBeaver database connection settings:**

Connection "yugabyte [kxn_db - kxn_user]" configuration

Connection settings
YugabyteDB connection settings

YugabyteDB

Main | YugabyteDB | Driver properties | SSH | Proxy | SSL

Server

Host: localhost Port: 5433

Database: kxn_db

Authentication

Authentication: Database Native Database native authentication

Username: kxn_user

Password: [masked] ☒ Save password locally

Advanced

Local Client: PostgreSQL 10

i You can use variables in connection parameters.

Driver name: YugabyteDB [Edit Driver Settings](#)

[Test Connection ...](#) [OK](#) [Cancel](#)

6. trino

[trino](#) can integrate the following DBMS, among others:

- MySQL via the [MySQL Connector](#),
- Oracle via the [Oracle Connector](#), and
- PostgreSQL via the [PostgreSQL Connector](#).
- SQL Server via the [SQL Server Connector](#),

DBSeeder makes it possible to use trino's JDBC driver and the corresponding connectors as an alternative to the JDBC drivers of the DBMS suppliers. To use the trino JDBC driver, a trino server is required. With the script `db_seeder_trino_environment` a trino server can be set up. Since trino does not support the Windows operating system, a suitable Docker image is created for Windows. For Linux, e.g. Ubuntu, the script can alternatively be used to perform a local installation of the trino server.