
SchemaSpy Documentation

Release 6.0.0

SchemaSpy Contributors

Apr 02, 2017

1	What is SchemaSpy ?	3
2	Sample documentation	5
3	SchemaSpy GUI	7
3.1	SchemaSpy	7
3.1.1	What is SchemaSpy ?	7
3.1.2	Sample documentation	8
3.1.3	SchemaSpy GUI	8
3.2	Installation	8
3.2.1	Java 8	8
3.2.2	Graphviz	8
3.3	Get Started	8
3.3.1	Configuration	8
3.3.2	Running SchemaSpy	9
3.3.2.1	Parameters priority:	9
3.3.2.2	Commonly used parameters:	9
4	Indices and tables	11

Document your database simply and easily

Do you hate starting on a new project and having to try to figure out someone else's idea of a database? Or are you in QA and the developers expect you to understand all the relationships in their schema? If so then this tool's for you.

What is SchemaSpy ?

SchemaSpy is a Java-based tool (requires Java 8 or higher) that analyzes the metadata of a schema in a database and generates a visual representation of it in a browser-displayable format. It lets you click through the hierarchy of database tables via child and parent table relationships as represented by both HTML links and entity-relationship diagrams. It's also designed to help resolve the obtuse errors that a database sometimes gives related to failures due to constraints. It is free software that is distributed under the terms of the MIT License.

If you like SchemaSpy then please give the start on project.

SchemaSpy uses the dot executable from [Graphviz](#) to generate graphical representations of the table/view relationships. This was initially added for people who see things visually. Now the graphical representation of relationships is a fundamental feature of the tool. Graphvis is not required to view the output generated by SchemaSpy, but the dot program should be in your PATH (not CLASSPATH) when running SchemaSpy or none of the entity relationship diagrams will be generated (or use the -gv option).

SchemaSpy uses JDBC's database metadata extraction services to gather the majority of its information, but has to make vendor-specific SQL queries to gather some information such as the SQL associated with a view and the details of check constraints. The differences between vendors have been isolated to configuration files and are extremely limited. Almost all of the vendor-specific SQL is optional.

CHAPTER 2

Sample documentation

Browse some [sample documentation](#) generated by SchemaSpy. Note that this was run against an extremely limited schema so it doesn't show the full power of the tool.

SchemaSpy is a command line tool. If you're more comfortable with the point-and-click approach then try out [Joachim Uhl's SchemaSpyGUI](#).

SchemaSpy was mentioned in one of th O'Reilly's book

SchemaSpy

Document your database simply and easily

Do you hate starting on a new project and having to try to figure out someone else's idea of a database? Or are you in QA and the developers expect you to understand all the relationships in their schema? If so then this tool's for you.

What is SchemaSpy ?

SchemaSpy is a Java-based tool (requires Java 8 or higher) that analyzes the metadata of a schema in a database and generates a visual representation of it in a browser-displayable format. It lets you click through the hierarchy of database tables via child and parent table relationships as represented by both HTML links and entity-relationship diagrams. It's also designed to help resolve the obtuse errors that a database sometimes gives related to failures due to constraints. It is free software that is distributed under the terms of the MIT License.

If you like SchemaSpy then please give the start on project.

SchemaSpy uses the dot executable from [Graphviz](#) to generate graphical representations of the table/view relationships. This was initially added for people who see things visually. Now the graphical representation of relationships is a fundamental feature of the tool. Graphvis is not required to view the output generated by SchemaSpy, but the dot program should be in your PATH (not CLASSPATH) when running SchemaSpy or none of the entity relationship diagrams will be generated (or use the -gv option).

SchemaSpy uses JDBC's database metadata extraction services to gather the majority of its information, but has to make vendor-specific SQL queries to gather some information such as the SQL associated with a view and the details of check constraints. The differences between vendors have been isolated to configuration files and are extremely limited. Almost all of the vendor-specific SQL is optional.

Sample documentation

Browse some [sample documentation](#) generated by SchemaSpy. Note that this was run against an extremely limited schema so it doesn't show the full power of the tool.

SchemaSpy GUI

SchemaSpy is a command line tool. If you're more comfortable with the point-and-click approach then try out [Joachim Uhl's SchemaSpyGUI](#).

SchemaSpy was mentioned in one of th O'Reilly's book

Installation

Before you can start using SchemaSpy you must have installed two things in your system environment.

Java 8

Download instructions for all operating systems: <https://java.com/en/download/manual.jsp>

Graphviz

- **Windows** The easiest way to install Graphviz is to download the msi package from http://www.graphviz.org/Download_windows.php

Warning: Remember to add the folder containing Graphviz's dot.exe application to your system PATH variable, eg.

```
C:\Program Files (x86)\Graphviz2.38\bin
```

- **Linux, Mac OS** Please read carefully the detailed instructions on how to install Graphviz on your os version <http://www.graphviz.org/Download.php>.

Get Started

Welcome to SchemaSpy. We will do the best to simplify documentation process of your database.

Configuration

Parameters can be specified in the comand line (described below) or you can predefine configuration in the file. SchemaSpy will search configuration file in `<current-dir>/schemaspyspy.properties` To use an alternative configuration file run SchemaSpy with parameter: `java -jar schemaspyspy.jar -configFile path/to/config.file`

Config file example:

```
# type of database. Run with -dbhelp for details
schemaspyspy.t=mssql
# optional path to alternative jdbc drivers.
schemaspyspy.dp=path/to/drivers
# database properties: host, port number, name user, password
schemaspyspy.host=server
schemaspyspy.port=1433
schemaspyspy.db=db_name
schemaspyspy.u=database_user
schemaspyspy.p=database_password
# output dir to save generated files
schemaspyspy.o=path/to/output
# db scheme for which generate diagrams
schemaspyspy.s=dbo
```

Running SchemaSpy

You can easily run SchemaSpy from the command line:

```
java -jar schemaspyspy.jar -t dbType -dp C:/sqljdbc4-3.0.jar -db dbName -host server -
  -port 1433 [-s schema] -u user [-p password] -o outputDir
```

Parameters priority:

It is important to notice, that command-line parameters **override** those configured in schemaspyspy.properties file.

Commonly used parameters:

[-t databaseType] Type of database (e.g. ora, db2, etc.). Use -dbhelp for a list of built-in types. Defaults to ora.

[-db dbName] Name of database to connect to

[-u user] Valid database user id with read access. A user id is required unless -sso is specified.

[-s schema] Database schema. This is optional if it's the same as user or isn't supported by your database. Use -noschema if your database thinks it supports schemas but doesn't (e.g. older versions of Informix).

[-p password] Password associated with that user. Defaults to no password.

[-o outputDirectory] Directory to write the generated HTML/graphs to

[-dp pathToDrivers] Looks for drivers here before looking in driverPath in [databaseType].properties. The drivers are usually contained in .jar or .zip files and are typically provided by your database vendor.

[-hq] or [-lq] Generate higher or lower-quality diagrams. Various installations of Graphviz (depending on OS and/or version) will default to generat /ing either higher or lower quality images. That is, some might not have the "lower quality" libraries and others might not have the "higher quality" libraries. Higher quality output takes longer to generate and results in significantly larger image files (which take longer to download / display), but the resultant Entity Relationship diagrams generally look better.

CHAPTER 4

Indices and tables

- `genindex`
- `modindex`
- `search`