Willem van den Ende
Posted on Jul 22, 2020 • Updated on Jul 6, 2021

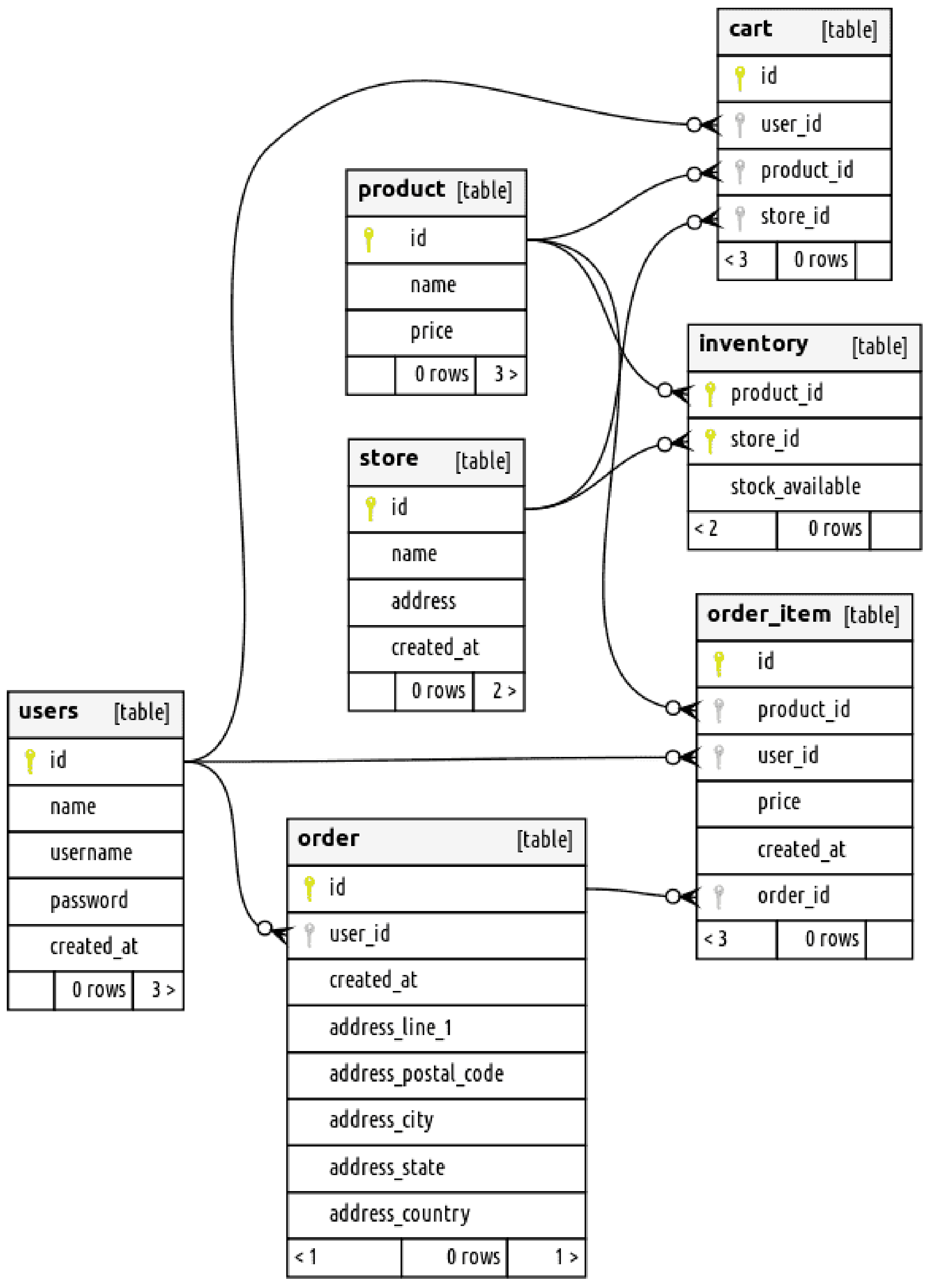
How to visualize a PostgreSQL schema as SVG with SchemaSpy

#postgres #documentation #visualization

TLDR: Use the option `-imageformat svg` to generate an ER diagram in SVG format with SchemaSpy.

SchemaSpy generates HTML documentation from your database schema, including Entity Relationship diagrams. I find visualization helpful as a database model grows.

Out of the box it produces diagrams in png format, like this one in the Hasura ecommerce example:



Generated by SchemaSpy

I prefer SVG, because it scales, is smaller, and it is all text.

Using SchemaSpy to see what the relations are like is not complicated, *once you have downloaded the dependencies and figured out the command line options*. And there are quite a lot of command line parameters. I had to download the postgresql jdbc driver by hand, and specify the path with the `-dp` option:

```
<schemaspy-path>$ java -jar target/schemaspy-6.1.1-SNAPSHOT.jar \
  -t pgsql -host localhost -db postgres -o /tmp -u postgres -dp . -p postgres -imageformat
```

The -t option is mandatory. Probably works for anything that has a JDBC Driver. I use a local database inside a docker container that is only open to my machine, so I left the passwords and the port at the default settings the docker-compose configuration came with.

See the comment by Jon Lauridsen below on how to do all of this with a docker one-liner.

If you forget an option, Schema Spy will tell you what was missing. I stuck the above in a shell script inside my project, so I can easily generate it again. I left the output as `/tmp`, so when I reboot (rarely), I am forced to create a fresh diagram. Next step: integrate it into the build script, so I get a fresh diagram whenever I add database migration files in the `migrations` directory.

SchemaSpy uses graphviz to generate images, the .dot files it uses as input, and the generated images can be found in `<outputpath>/summary/`

At first I had a one liner for this, and then while writing the post, I found that SchemaSpy supports SVG natively. It is 'just' another output parameter:

```
-imageformat svg
```

For more details, see the Get Started page in the SchemaSpy documentation .

I hope you found this helpful. It took me a bit of searching to find SchemaSpy, and the svg option. I'd be interested to know better alternatives, if there are any.

Top comments (3)

Jon Lauridsen • Jan 12 '21 • Edited on

Thanks for the article. I didn't want to download SchemaSpy so for anyone else interested you can run it straight via Docker:

```
docker run --rm -it schemaspy/schemaspy:latest --help
```

I've cooked up this command to generate report from my locally-running DB:

```
docker run --rm -it -v "$(pwd)/output":/output  --network host schemaspy/schemaspy:latest -t
```

Documentation is on schemaspy.readthedocs.io.

andreasneuman • Sep 28 '20

SchemaSpy is an interesting solution, I will definitely try it out, thanks. Now I'm using Studio for PostgreSQL with its reporting and visualizing tools, it works pretty well.

Sualeh Fatehi • Dec 29 '21

Also take a look at How to Visualize Your PostgreSQL Database with One Command (and Nothing to Install) for another approach.

Thank you.

Thanks for visiting DEV, we've worked really hard to cultivate this great community and would love to have you join us. If you'd like to create an account, you can sign up here.

Willem van den Ende

Systemic thinker and doer, Software developer.

LOCATION
Bath, UK

EDUCATION
University of Twente, MSc Computer Science

WORK
Full circle developer at QWAN.co.uk - Quality Without A Name

JOINED
May 22, 2020

Trending on DEV Community 👩‍💻👨‍💻 🔥

Always over estimate effort

#discuss #devrel #productivity

Defect life cycle in API testing ⚙️

#beginners #api #codenewbie #productivity

CHALLENGE: Describe AI in five words or less.

#discuss #ai #watercooler