jts_erd Documentation

Release 0.0.1

ibu radempa

October 18, 2015

Contents

jts_erd creates an ERD (entity-relationship diagram).

It requires an extension of a json-table-schema as input.

Depends on pygraphviz.

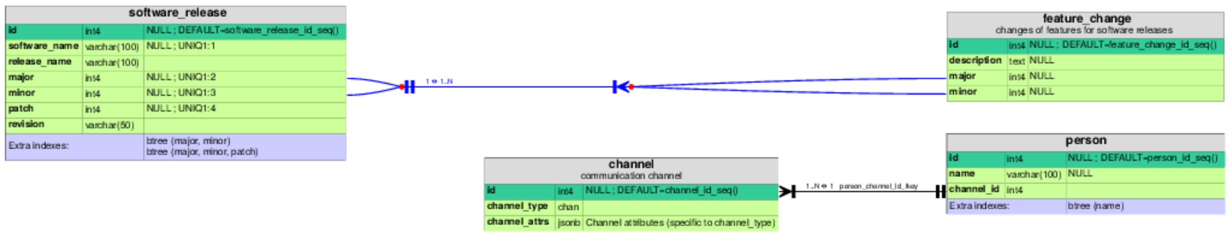
CHAPTER 1

Introduction

For now please look at these slides: `20150927_talk.pdf`

TL;DR Generate an entyty-relationship diagram for a schema given as JSON-table-schema.

Example of a resulting ERD:


software_release
id int4 NULL ; DEFAULT=software_release_id_seq()
software_name varchar(100) NULL ; UNIQ1:1
release_name varchar(100)
major int4 NULL ; UNIQ1:2
minor int4 NULL ; UNIQ1:3
patch int4 NULL ; UNIQ1:4
revision varchar(50)
Extra indexes: btree (major, minor) btree (major, minor, patch)
1 ⊕ 1..N
feature_change
changes of features for software releases
id int4 NULL ; DEFAULT=feature_change_id_seq()
description text NULL
major int4 NULL
minor int4 NULL
channel
communication channel
id int4 NULL ; DEFAULT=channel_id_seq()
channel_type chan
channel_attrs jsonb Channel attributes (specific to channel_type)
1..N ⊕ 1 person_channel_id_fkey
person
id int4 NULL ; DEFAULT=person_id_seq()
name varchar(100) NULL
channel_id int4
Extra indexes: btree (name)

Installation

Install graphviz and build tools:

```
aptitude install pkg-config build-essential graphviz libgraphviz-dev
```

You will need at least PyGraphviz 1.3.1 when using python3.

Currently (as of version 1.3.1 of pygraphviz) on at least debian and ubuntu you need special install options due to a bug (https://github.com/pygraphviz/pygraphviz/issues/71):

```
pip3 install pygraphviz --install-option="--include-path=/usr/include/graphviz" --install-option="--
```

(gcc still throws a warning.)

Prepare a virtualenv with python3:

```
mkdir jts_erd
cd jts_erd
virtualenv -p python3
source bin/activate
```

In the virtualenv root dir:

```
git clone https://github.com/iburadempa/jts_erd.git
```

CHAPTER 3

Usage examples

The main function `jts_erd.get_graph()` creates a graphviz graph from a (JSON-decoded) JSON-table-schema. An accompanying function, `jts_erd.save_svg()`, renders a graph in SVG format and saves it.

For examples look at the examples directory, https://github.com/iburadempa/jts_erd/tree/master/examples

CHAPTER 4

API

4.1 jts_erd

Generate an entity-relationship diagram from an extended JSON table schema.

JSON table schema is a simple schema for describing the structure of tabular data. It can be extended to allow for a comprehensive representation of an SQL relational database schema.

Starting from such a description this python module generates visualizations of the database schema using graphviz via PyGraphviz.

`jts_erd.jts_erd.`**`get_graph`**`(`*json_database_schema*, ***options*`)`

Create and return a graph from the given *json_database_schema*.

All keys from *`options_defaults`* are allowed in *kwargs*.

`jts_erd.jts_erd.`**`options_defaults`** **= {'fontsize_label': 6, 'edge_thickness': 1.0, 'omit_isolated_tables': False, 'display**

Options and their default values.

Options:

- **html_color_default**
- **html_color_highlight**
- **fontname**
- **fontsize**
- **fontsize_title**
- **fontsize_label**
- **bgcolor_indexes**
- **rankdir**: 'LR' or 'RL'; whether dependent tables appear on the right (left) hand side
- **edge_thickness**
- **display_columns**: bool
- **display_indexes**: bool
- **display_crowfoots**: bool
- **omit_isolated_tables**: bool

`jts_erd.jts_erd.`**`save_svg`**(*json_database_schema*, *filepath*, ***options*)

Write an ERD in SVG format for a database to a file.

json_database_schema must be compatible with what pg_jts produces. *filepath* must end in '.svg'.

Source: https://github.com/iburadempa/jts_erd/

Python Module Index

Index