# pgloader

(ql:quickload :pgloader)
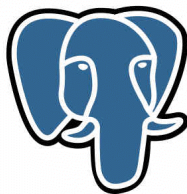
Dimitri Fontaine
@tapoueh

May 6th, 2014

# Dimitri Fontaine

**2ndQuadrant France**
PostgreSQL Major Contributor

- debian
- pgloader
- prefix, skytools, ...
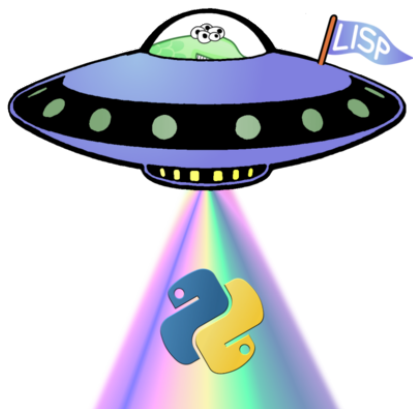- CREATE EXTENSION
- CREATE EVENT TRIGGER

PostgreSQL

# Load data into PostgreSQL. Fast.

http://pgloader.io/

# Used to be *python* code. Now Common Lisp.

# pgloader got faster

```
select rows, v2, v3,
       round((  extract(epoch from v2)
       / extract(epoch from v3))::numeric, 2)
       as speedup
  from timing;

  rows   |         v2          |        v3        | speedup
---------+---------------------+------------------+---------
 4768765 | @ 37 mins 10.878    | @ 1 min 26.917   |   25.67
 3115880 | @ 36 mins 5.881     | @ 1 min 10.994   |   30.51
 3865750 | @ 33 mins 40.233    | @ 1 min 15.33    |   26.82
 3994483 | @ 29 mins 30.028    | @ 1 min 18.484   |   22.55
(4 rows)
```

# pgloader command language & parser (using esrap)

```
LOAD CSV
     FROM inline (x, y, a, b, c, d)
     INTO postgresql:///pgloader?csv (a, b, d, c)

     WITH truncate,
          skip header = 1,
          fields optionally enclosed by '"',
          fields escaped by double-quote,
          fields terminated by ','

      SET client_encoding to 'latin1',
          work_mem to '12MB',
          standard_conforming_strings to 'on'
```

# parse from the REPL

```
(pgloader::with-monitor ()
  (pgloader.parser::parse-commands-from-file
   "/Users/dim/dev/pgloader/test/csv.load"))
```

# Common Lisp Compiler available at run-time

```
((LAMBDA ()
    ...
  (LET ((PGLOADER.PARSER::SOURCE
   (MAKE-INSTANCE 'PGLOADER.CSV:COPY-CSV
       :TARGET-DB "pgloader"
       :SOURCE
       '(:INLINE
         (#P"/Users/dim/dev/pgloader/test/csv.load" . 926))
       :TARGET "csv" :ENCODING :UTF-8
       :FIELDS '(("x") ("y") ("a") ("b") ("c") ("d"))
       :COLUMNS '(("a" NIL NIL) ("b" NIL NIL) ("d" NIL NIL)
                 ("c" NIL NIL))
       :SKIP-LINES 1 :QUOTE #\"
       :ESCAPE "\"\"" :SEPARATOR #\,)))
     (PGLOADER.SOURCES:COPY-FROM PGLOADER.PARSER::SOURCE
                                 :TRUNCATE TRUNCATE)))
```

# Common Lisp Compiler available at run-time

```
$ /usr/bin/pgloader --version
pgloader version "3.0.99"
compiled with SBCL 1.1.17

$ git clone https://github.com/dimitri/pgloader.git \
            /tmp/pgloader

$ /usr/bin/pgloader --self-upgrade /tmp/pgloader --version
Self-upgrading from sources at "/tmp/pgloader/"
pgloader version "3.0.fecae2c"
compiled with SBCL 1.1.17
```

# The headache...