

pgloader

PGConf US 2016, NYC

Dimitri Fontaine
@tapoueh

April 18, 2016

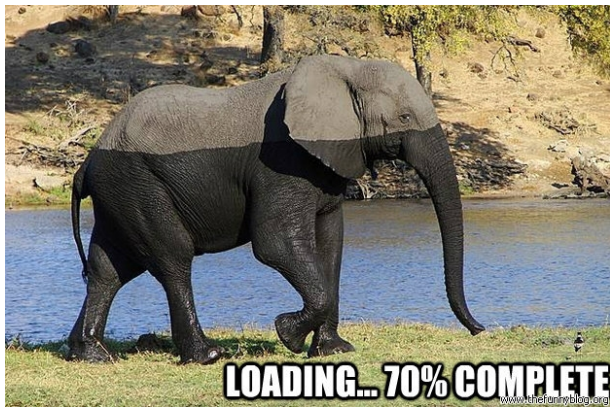
Load data into PostgreSQL. Fast.

<http://pgloader.io/>



Load data into PostgreSQL. Fast.

<http://pgloader.io/>

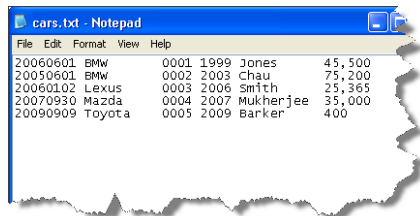


pgloader: Open Source, github

`https://github.com/dimitri/pgloader`



File based formats



20060601	BMW	0001	1999	Jones	45,500
20050601	BMW	0002	2003	Chau	75,200
20060102	Lexus	0003	2006	Smith	25,365
20070930	Mazda	0004	2007	Mukherjee	35,000
20090909	Toyota	0005	2009	Barker	400



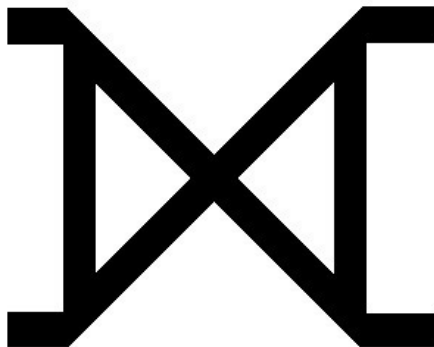
Load from a live connection

`http://pgloader.io/howto/mysql.html`



pgloader : Transform data on the fly

`http://pgloader.io/`



Using a command language

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 work_mem to '12MB',  
    standard_conforming_strings to 'on'
```


with extra before/after sections

```
BEFORE LOAD DO
  $$ drop table if exists csv; $$,
  $$ create table csv (
    a bigint,
    b bigint,
    c char(2),
    d text
  );
  $$;
```

Some data source examples

```
FROM stdin
```

```
FROM inline (a, b, c)
```

```
FROM data/2013_Gaz_113CDs_national.txt
```

```
FROM FILENAME MATCHING ~/GeoLiteCity-Location.csv/
```

```
FROM ALL FILENAMES MATCHING ~/ALIOR/
```

```
FROM ALL FILENAMES MATCHING ~/F[A-Z]{4}1[45]|OZ20/
```

```
FROM http://www.census.gov/geo/maps-data/  
      data/docs/gazetteer/places2k.zip
```

```
FROM http://www.insee.fr/fr/methodes/nomenclatures/  
      cog/telechargement/2013/dbf/historiq2013.zip
```

On the fly data transformations

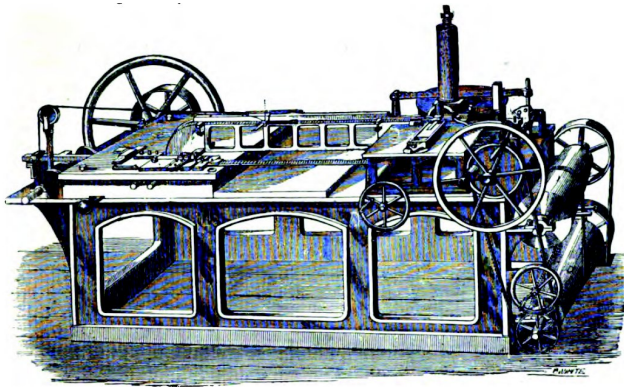
```
FROM FILENAME MATCHING ~/GeoLiteCity-Blocks.csv/  
  WITH ENCODING iso-8859-1  
  (  
    startIpNum, endIpNum, locId  
  )  
INTO postgresql:///ip4r?geolite.blocks  
  (  
    iprange ip4r using (ip-range startIpNum endIpNum),  
    locId  
  )
```



On the fly processing useful for CASTing too

Empty string and NULL, default values, zero *dates* 0000-00-00, int(11), float(20,2), tinyint rather than boolean, sets, ...

Oh, and *encodings* too



JOHNSON AND ATKINSON'S TYPE CASTER.

Digitized by Google

Here's how to migrate from MySQL to PostgreSQL

In **one** command line.

```
$ pgloader mysql://user@localhost/sakila \  
pgsql:///pagila
```

pgloader mysql://root@localhost/sakila pgsql:///pagila

table name	read	imported	errors	total time	read	write

before load	3	3	0	0.008s		
fetch meta data	86	86	0	0.184s		
create, drop	0	18	0	0.244s		

actor	200	200	0	0.011s	0.021s	0.010s
address	603	603	0	0.038s	0.040s	0.037s
category	16	16	0	0.007s	0.006s	0.007s
city	600	600	0	0.037s	0.034s	0.036s
country	109	109	0	0.009s	0.026s	0.009s
customer	599	599	0	0.022s	0.077s	0.021s
films	1000	1000	0	0.052s	0.097s	0.051s
.....						
inventory	4581	4581	0	0.057s	0.190s	0.057s
language	6	6	0	0.003s	0.011s	0.003s
payment	16049	16049	0	0.246s	0.500s	0.245s
rental	16044	16044	0	0.329s	0.623s	0.329s
staff	2	2	0	0.025s	0.007s	0.025s
store	2	2	0	0.094s	0.010s	0.094s
actor_info	200	200	0	0.013s	1.020s	0.012s
mv.customer_list	599	599	0	0.022s	0.047s	0.022s
mv.film_list	997	997	0	0.046s	0.172s	0.046s

COPY Threads Completion	69	69	0	2.076s		
Create Indexes	41	41	0	0.489s		
Index Build Completion	41	41	0	0.002s		
Reset Sequences	0	13	0	0.030s		
Primary Keys	16	16	0	0.018s		
Foreign Keys	22	44	0	0.156s		
Install comments	0	0	0	0.000s		

Total import time	50086	50086	0	2.788s	3.572s	1.151s



And more to come

File formats with on-the-fly normalisation

<?xml?>

{JSON}

Other database systems

ORACLE®

 INFORMIX

 Microsoft®
SQL Server®

 SYBASE®

You can become a sponsor!

<http://pgloader.io/pgloader-moral-license.html>



SPONSORING



Load data into PostgreSQL. Fast.

<http://pgloader.io/>

