cstore_fdw
column store pro PostgreSQL

Prague PostgreSQL Developer Day 2015
Jan Holčapek
holcapek@gmail.com

cstore_fdw

- foreign data wrapper pro ORC* formát
  - ORC, Optimized Row Columnar
  - jeden z formátů, který je podporován Apache Hive
  - (*) `cstore_fdw` se odchyluje od specifikace ORC
- open source
  - `https://github.com/citusdata/cstore_fdw`
  - `http://pgxn.org/dist/cstore_fdw/`
- verze 1.0 ~ duben 2014
- verze 1.1 ~ srpen 2014

Citus Data

- Y Combinator (seed accelerator) alumnus
  - léto 2011
  - `http://yclist.com/`
- Silicon Valley, Istanbul
- Josh Berkus, poradce
  - člen PostgreSQL Core Teamu
  - `http://www.citusdata.com/people`
- CitusDB ~ PostgreSQL-{XC,XL}

Row-oriented store

Id	…	price	…	…	quant	…	…	last_stm	…	…	…	…	…	weight
1	…	3.90	…	…	31	…	…	2013-…	…	…	…	…	…	0.6
2	…	13	…	…	70	…	…	2010-…	…	…	…	…	…	0.8
3	…	4.25	…	…	432	…	…	2013-…	…	…	…	…	…	1
4	…	4	…	…	45	…	…	2013-…	…	…	…	…	…	6
…														
4…	…	95	…	…	37	…	…	2013-…	…	…	…	…	…	0.6
4…	…	59	…	…	90	…	…	2012-…	…	…	…	…	…	1.5

citusdata

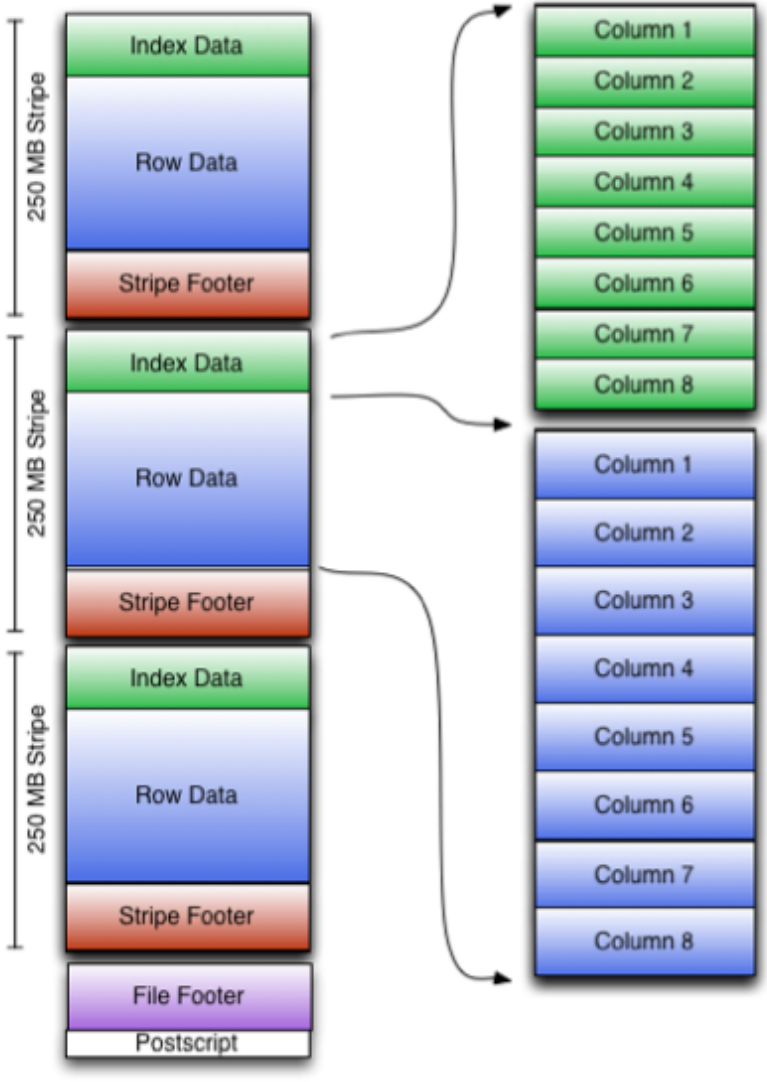
Column-oriented store

Id	sz	price	…	…	quant	…	…	last_stm	…	…	…	…	…	weight
1	4	3.90	…	…	31	…	…	2013-…	…	…	…	…	…	0.6
2	3	13	…	…	70	…	…	2010-…	…	…	…	…	…	0.8
3	2	4.25	…	…	432	…	…	2013-…	…	…	…	…	…	1
4	4	4	…	…	45	…	…	2013-…	…	…	…	…	…	6
…														
4…	19	95	…	…	37	…	…	2013-…	…	…	…	…	…	0.6
4…	2	59	…	…	90	…	…	2012-…	…	…	…	…	…	1.5

citusdata

ORC

https://cwiki.apache.org/



ORC, Optimized Row Columnar

> `orc-file := {stripe*, orc-footer}`
> `stripe := {index, data, stripe-footer}`
> `index :=` min, max, sum sloupců ve stripe
> `stripe-footer :=` kódování sloupců
> `orc-footer :=` seznam stripes, # řádků ve stripe, typy sloupců a jejich agregace (count, min, max, sum) v celém souboru

Odlišnosti `cstore_fdw` od ORC

- posuďte sami
  - `https://github.com/apache/hive/blob/trunk/ql/src/protobuf/org/apache/hadoop/hive/ql/io/orc/orc_proto.proto`
  - `https://github.com/citusdata/cstore_fdw/blob/master/cstore.proto`
- chybí sum agregace
- chybí kódování
- jiné komprese

Demo

- instalace `cstore_fdw`
- vyrobíme nativní a `cstore_fdw` tabulky
- nalijeme do nich data
- posbíráme statistiky
- srovnáme velikost vstupu a velikost tabulek
- podíváme se na exekuční plány

Instalace

> „ručně“

```
git clone \ git@github.com:
citusdata/cstore_fdw.git
make
sudo make install
```

> pgxn

```
sudo pgxn install cstore_fdw
```

Instalace

> doplnit `cstore_fdw` do `shared_preload_libraries`
> restartovat PostgreSQL

Instalace

- v příslušné databázi jako superuživatel

```
CREATE EXTENSION cstore_fdw;
CREATE SERVER cstore_server FOREIGN
DATA WRAPPER cstore_fdw;
GRANT USAGE ON FOREIGN SERVER
cstore_server TO <user>;
```

- jako běžný uživatel tamtéž

```
CREATE FOREIGN TABLE <table def>;
```

```
DROP FOREIGN TABLE IF EXISTS cust_rev_cstore;
CREATE FOREIGN TABLE cust_rev_cstore
(
    customer_id TEXT,
    review_date DATE,
    review_rating INTEGER,
    review_votes INTEGER,
    review_helpful_votes INTEGER,
    product_id CHAR(10),
    product_title TEXT,
    product_sales_rank BIGINT,
    product_group TEXT,
    product_category TEXT,
    product_subcategory TEXT,
    similar_product_ids CHAR(10)[]
)
SERVER cstore_server
OPTIONS(compression 'pglz');
```

```
DROP TABLE IF EXISTS cust_rev_native;
CREATE TABLE cust_rev_native
(
    customer_id TEXT,
    review_date DATE,
    review_rating INTEGER,
    review_votes INTEGER,
    review_helpful_votes INTEGER,
    product_id CHAR(10),
    product_title TEXT,
    product_sales_rank BIGINT,
    product_group TEXT,
    product_category TEXT,
    product_subcategory TEXT,
    similar_product_ids CHAR(10)[]
);
```

```
\copy cust_rev_cstore from '/tmp/customer_reviews_1999.csv' with csv;
\copy cust_rev_native from '/tmp/customer_reviews_1999.csv' with csv;

ANALYZE cust_rev_cstore;
ANALYZE cust_rev_native;

\! ls -lh /tmp/customer_reviews_1999.csv

SELECT pg_size_pretty(cstore_table_size('cust_rev_cstore'));
SELECT pg_size_pretty(pg_table_size('cust_rev_native'));
```

```
EXPLAIN SELECT 1 FROM cust_rev_cstore;
EXPLAIN SELECT 1 FROM cust_rev_native;

EXPLAIN SELECT customer_id FROM cust_rev_cstore;
EXPLAIN SELECT customer_id FROM cust_rev_native;

EXPLAIN SELECT customer_id FROM cust_rev_cstore WHERE review_date = '1999-06-01';
EXPLAIN SELECT customer_id FROM cust_rev_native WHERE review_date = '1999-06-01';

CREATE INDEX cust_rev_native_rev_date ON cust_rev_native(review_date);
-- tohle vam neprojde
CREATE INDEX cust_rev_cstore_rev_date ON cust_rev_cstore(review_date);

-- jeste jednou
EXPLAIN SELECT customer_id FROM cust_rev_cstore WHERE review_date = '1999-06-01';
EXPLAIN SELECT customer_id FROM cust_rev_native WHERE review_date = '1999-06-01';

EXPLAIN SELECT customer_id FROM cust_rev_cstore WHERE review_date >= '1999-06-01';
EXPLAIN SELECT customer_id FROM cust_rev_native WHERE review_date >= '1999-06-01';

EXPLAIN SELECT customer_id FROM cust_rev_cstore WHERE review_date >= '1999-06-01' AND review_date <=
'1999-09-30';
EXPLAIN SELECT customer_id FROM cust_rev_native WHERE review_date >= '1999-06-01' AND review_date <=
'1999-09-30';
```

Závěr

- nižší I/O
  - menší soubory (komprese)
  - skip indexy
  - projekce
- nalívání pomocí COPY
  - plánují implementovat INSERT, UPDATE, DELETE
- Foreign Data Wrapper API
  - namá callbacky na agregace