

# Jak ušetřit místo a zároveň mít data snadno dostupná

David Turoň  
david.turon@linuxbox.cz

16.2.2017

Proč:

- šetří místo na disku
- automatické mazání starých nepotřebných dat
- nad staršími daty rychlejší čtení/agregace i nad pomalejšími disky

Využívá moduly:

- pg\_partman (automatický partitioning)
- cstore\_fdw (přístup k sloupcovému úložišti CSTORE)

Odkazy:

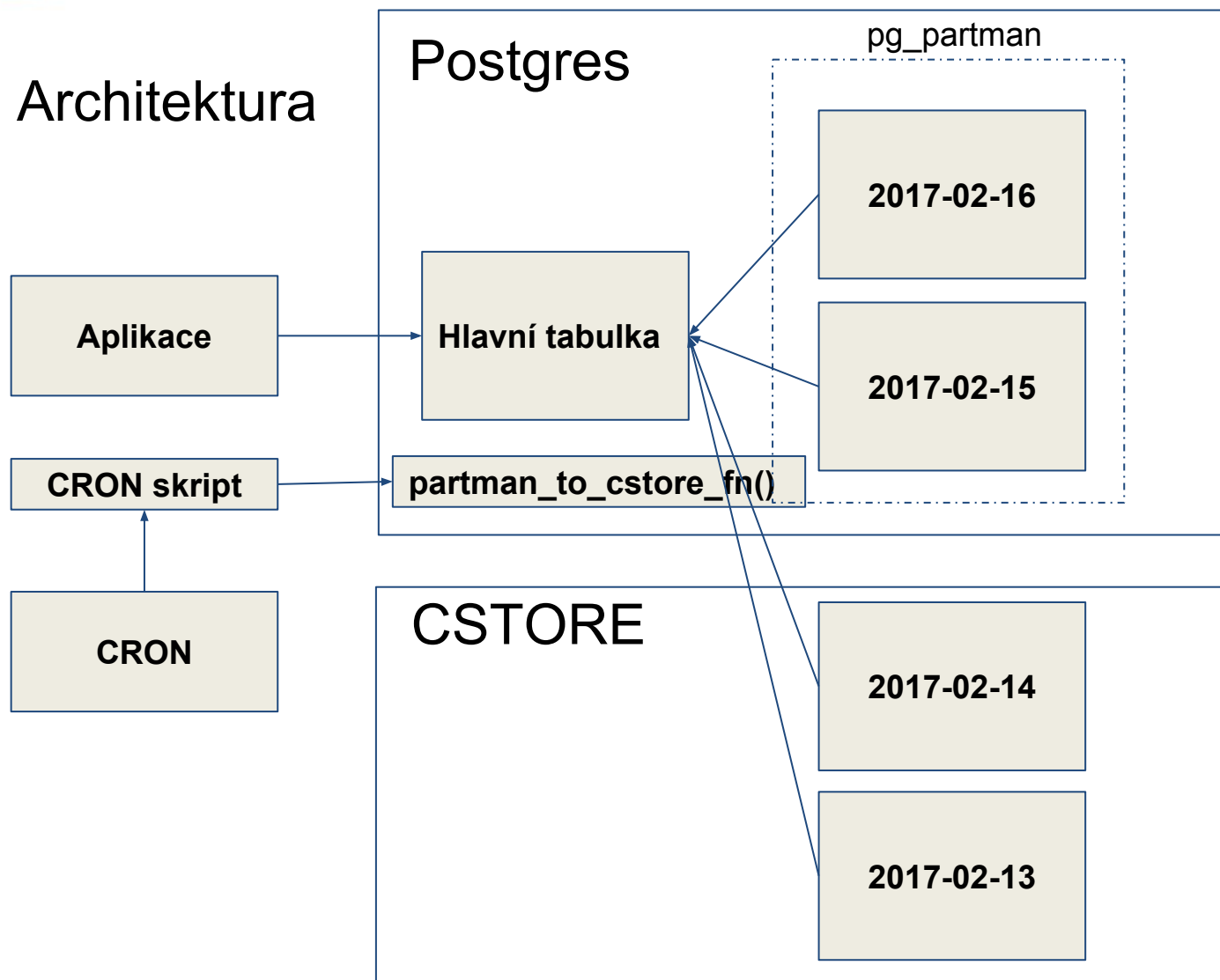
- [https://github.com/keithf4/pg\\_partman](https://github.com/keithf4/pg_partman)
- [https://github.com/citusdata/cstore\\_fdw](https://github.com/citusdata/cstore_fdw)
- [https://github.com/dturon/partman\\_to\\_cstore](https://github.com/dturon/partman_to_cstore)

Kde je možno ji použít:

- tabulka s rotacemi (sloupec timestamp/timestamptz)
- senzorová data
- log událostí
- starší data beze změn (UPDATE/DELETE)
- ve většině případů bez zásahu v aplikaci

# Estenze partman\_to\_cstore

## Architektura



## Volby převzaté z CSTORE:

- *compression* - komprese pglz/vypnuta
- *stripe\_row\_count*
- *block\_row\_count*

## Volby:

- *move\_int* (přesuň do CSTORE tabulky  
starší než 'x days':interval)
- *drop\_int* (odstraň -||-)

## Vlastnosti:

- konfigurační tabulka *move\_config*
- přesun pomocí cron skriptu (nejlépe mimo prac. dobu)
- vytvoří se kopie tabulky *table\_pYYYYMMDD* jako *table\_pYYYYMMDD\_cstore*
- převedou se data, práva, omezení
- nastaví se dědičnost z hlavní tabulky
- odstraní se zdrojová tabulka *table\_pYYYYMMDD*
- nekopírují se triggery
- **pozor na zálohy** - pg\_dump nekopíruje data z FDW tabulek (COPY/pg\_dump před přesunem tabulky)
- **pozor na upgrade** (dump/copy jinak změna OID v cestách k CSTORE)

## Cron skript:

- 1x denně projde všechny databáze
- v databázích kde je nainstalovaná extenze partman\_to\_cstore
- na základě konfigurační tabulky spustí funkci pro přesun dat do CSTORE
- možno upravit dle libosti

# DEMO



```
CREATE SCHEMA demo;
-- modul pg_partman
CREATE SCHEMA partman;
CREATE EXTENSION pg_partman WITH SCHEMA partman;

-- modul pro přístup k CSTORE
CREATE EXTENSION cstore_fdw;

-- modul pro automatický převod partitions do CSTORE
CREATE SCHEMA partman_to_cstore;
CREATE EXTENSION partman_to_cstore WITH SCHEMA partman_to_cstore;

CREATE TABLE demo.test(
    ts timestamptz PRIMARY KEY,
    some_data text,
    magic_number int
);
--naplníme daty
INSERT INTO demo.test
SELECT ts, repeat(ts::text,10), random()*100
FROM generate_series(now()-'5d'::interval,now(),'0.1s'::interval) AS x(ts);
```

--vytvoříme partitions

```
SELECT * FROM partman.create_parent(  
    p_parent_table:='demo.test',  
    p_control:='ts',  
    p_type:='time',  
    p_interval:='daily',  
    p_premake:=4,  
    p_use_run_maintenance:=NULL::boolean,  
    p_inherit_fk:=true,  
    p_jobmon:=false,  
    p_debug:=false  
);
```

-- přesuneme data do partitions

```
SELECT partman.partition_data_time('demo.test',p_batch_count:=1,p_batch_interval:='1day');  
\watch
```

-- uložení nastavení migrace do CSTORE v konfigurační tabulce (pro cron skript)

```
INSERT INTO partman_to_cstore.move_config(parent_table, move_int, drop_int)  
VALUES('demo.test','2days','15days');
```

--manuální spuštění přesunu (SET client\_min\_messages TO debug;)

```
SELECT partman_to_cstore.part_to_cstore(p_parent_table:='demo.test', move_int:='2days',  
    drop_int:='15days');
```

```
--data v CSTORE  
EXPLAIN --(ANALYZE, BUFFERS)  
SELECT sum(magic_number)  
FROM demo.test  
WHERE ts>='2017-02-12' AND ts<'2017-02-13';
```

```
--data v PG  
EXPLAIN --(ANALYZE, BUFFERS)  
SELECT sum(magic_number)  
FROM demo.test  
WHERE ts>='2017-02-15' AND ts<'2017-02-16';
```

Pro účely převedení libovolné partition(nemusí být spravovaná pg\_partmanem) do CSTORE je vytvořena funkce:

```
SELECT partman_to_cstore.move_singlepart_to_cstore(  
    'part_schema',  
    'part_table'  
);
```

-- měli jsme někde vlastní partitioning a nezávislá funkce se nám hodila

Dotazy?

Děkuji za pozornost  
a  
ať Vám chutná

**LinuxBOX.cz**