Database migration
from Sybase ASE to PostgreSQL

Achim Eisele and Jens Wilke
1&1 Internet AG
30.10.2013
Agenda

- Introduction
- Analysis
- Differences between Sybase ASE and PostgreSQL
- Porting the application
- Migration strategies
- Retrospects
Introduction

Analysis

Differences between Sybase ASE and PostgreSQL

Porting the application

Migration strategies

Retrospects
Why we had been dissatisfied

- many DBs on one System
- the availability was depending on third party Systems
- many downtimes
- performance dependencies

Our goals

- autonomy
- performance improvements
- high availability
- \( \Rightarrow \) satisfied customers :-}
Introduction

Why PostgreSQL

- Most advanced Open Source RDBMS
- Best TCO
Analysis

Agenda

- Introduction
- Analysis
- Differences between Sybase ASE and PostgreSQL
- Porting the application
- Migration strategies
- Retrospects
3 Applications

- 250 GB size
- 20 Mio. Queries per day
- up to 1.500 transactions per second
- **Sybase**
  - all reading and writing operations on one system
  - many databases, user and tools on a single system

- **PostgreSQL**
  - Heartbeat / DRBD
  - WAL / Streaming Replication *(not synchronous!)*
  - SR-Standby for dedicated Statements
Differences between Sybase ASE and PostgreSQL

Agenda

- Introduction
- Analysis
- Differences between Sybase ASE and PostgreSQL
- Porting the application
- Migration strategies
- Retrospects
Sybase offers bi-directional replication using the replication agent

on Sybase it is possible to switch the replication direction

Sybase offers replication of single tables with few impact, using the transaction log

Sybase offers a replication solution for Road Warriors

RAW Devices, advantage on modern file systems is doubtable

I/O can be audited by process

Plans of running queries can be shown
Postgres’ MVCC vs Sybase’s locking
Postgres is conforming to SQL-standard
nested transactions vs savepoints
Postgres does know schemes
Postgres does support grant for columns, not just revoke
on Postgres it is possible to restrict the connection limit per user
Logins can be restricted to IP addresses in Postgres
On Sybase ASE

- strings longer than varchar/char or a page are cropped or fail
- different behaviour on insert and update
- default ASE pagesize is 2k
- Sybase creates a fixed size container for the DB and the transaction log.
- shrinking the DB container is only possible with version 15.7
Postgres’ sequences are not limited to a single column as Sybase’s identity columns are and the value can be overwritten.

- bcp (COPY) is neither capable to escape nor to quote.
- Postgres does support multiple trigger for one event.
- Sybase is planning to implement advisory locks for future versions.
- Index concurrently since ASE 15.7 SP 100, May 2013.
Porting the application

Agenda

- Introduction
- Analysis
- Differences between Sybase ASE and PostgreSQL
- Porting the application
- Migration strategies
- Retrospects
Java (Hibernate)
  ■ loading JDBC driver
  ■ setting Dialect and DriverClass

PHP
  ■ On Debian:
    apt-get install php5-pgsql
  ■ PDO only need another data source name
  ■ “Plain” PHP
    ■ sybase_connect(), sybase_select_db() => pg_connect()
    ■ sybase_query() => pg_query()
    ■ sybase_fetch_object() => pg_fetch_object()
Date/time

- **Sybase**
  
  ```sql
  SELECT getdate();                  -- Oct 30 2013 03:52PM
  SELECT convert(varchar(30), getdate(), 109);
  -- Oct 30 2013 03:52:34:126PM
  ```

- **PostgreSQL**
  
  ```sql
  SELECT current_timestamp;
  -- 2013-10-30 15:52:34.126305+01
  SELECT to_char(current_timestamp,
     'Mon DD YYYY HH12:MI:SS:MSAM');
  ```
- **concatenation**
  - **Sybase**
    ```sql
    SELECT zip + ' ' + city FROM address
    ```
  - **PostgreSQL**
    ```sql
    SELECT zip || ' ' || city FROM address
    ```

- **field length**
  - **Sybase**
    ```sql
    set textsize 512000
    -- limits the size of a text or image column that is
    -- returned through a select statement. The default is 32KB.
    ```
  - **PostgreSQL**
    n/a
Migration strategies

Agenda

- Introduction
- Analysis
- Differences between Sybase ASE and PostgreSQL
- Porting the application
- Migration strategies
- Retrospects
Migration strategies:

- Dump Reload
- Replication
Migration strategies

Dump - Reload 1/5

- adjust and import the DDL
- stop the application/write access
- export Data, search and replace dates and times, import
- create indices, foreign keys and set sequences
- import and index/key creation by multiple processes in parallel
- logical checks
- re-configure and start the application
Advantages

- simple
- less effort
- rollback is also simple

Disadvantages

- longer downtime
- limits regarding quoting, escaping and content (date/time)
- empty strings are converted to NULL
- Export with bcp
  `bcp $db..$tablename out $fifo_path/$tablename.pipe -Jutf8 -c -t ','`
- Sybase bcp is not capable to write in unnamed pipes, just in files
- Date and time formats of Sybase are not conforming to any standard: milliseconds are separated by colons and a white space is missing before AM/PM
Import in Postgres using COPY and fix the timestamps:

```
mkfifo $fifo_path/$tablename.pipe

cat $fifo_path/$tablename.pipe | sed

s/\([^;\[A-Z]\]\\([^;\]\]+):\([^0-9]+\)\([AM\|PM]\)\([^;\]\]+\)/\1\2.\3 \4\5/g

| psql -d $IMPORTDB -c

"COPY $tablename from stdin CSV DELIMITER ",\n"
```

- alter sequence $seq restart with max(id)+1

- create indices and foreign keys
- the image type is converted to binary

- image is imported into a text field and decoded after that

- CREATE TABLE $tablename2 as (select id,
  decode(image, 'hex') as image,
  ... from $tablename);

- DROP TABLE $tablename;

- ALTER TABLE $tablename2 RENAME to $tablename;
Migration strategies

Sybase → PostgreSQL - Replication

Closed source replication tool based on transaction logs. Setup replication, wait until it’s synchronised and restart the re-configured application.

Advantages

- flexible point of time for the switch
- minimal downtime
- → less organisational overhead :)
- no impact on the production DB

Disadvantages

- expensive
- closed source without warranty
Top requirement: minimal downtime!

- should we write our own trigger based replication tool?
  too much resource consuming on the server
- analysis of the table usage
  - static tables
  - autoincrement tables
  - non-static tables
- tool based migration
- Optimizations (indices, field lengths)
Automated Export of DDL

- sp_tables
- sp_columns
- sp_pkeys
- sp_fkeys
- sp_statistics
- sp_helpuser
- sp_helprotect
// iterate over all tables
for $table in sp_tables
    "CREATE TABLE $table->name ("

// now iterate over all columns of table
for $column in sp_columns $table->name
    "$column->name $column->type DEFAULT $column->default"

// add primary key
"ALTER TABLE $table->name ADD PRIMARY KEY " + sp_pkeys

// grant statements
for $right in sp_helprotect
    "GRANT $right->action ON $table->name TO $right->user"
Data import

- to reduce the system load, batch jobs were suspended
- import by 10 processes in parallel (horizontal partition)
- chunks of 5,000 rows per select
- Setup Up-Sync (auto increment)
- 80 tables migrated by triggers (non static tables) using a transitional migration table (table name, primary key, action)
Finalisation

- create indices
- during downtime
  - create sequences
  - create foreign keys
  - content checks
  - re-configure the applications’ DB connection parameter
  - tests

result: 10 minutes downtime
Introduction

Analysis

Differences between Sybase ASE and PostgreSQL

Porting the application

Migration strategies

Retrospects
achievement of objectives

- autonomy: dedicated hardware
- stability: not one single outage in 12 months
- performance: dedicated hardware, read-only standby

```
====== anomaly for job 'cron_detect_refund_fraud.php'
trouble type: job duration

2012-07-19 06:45:01 - 2012-07-19 06:47:28 (147s)
2012-07-23 06:45:01 - 2012-07-23 06:48:02 (181s)
[...]
2012-12-10 06:45:01 - 2012-12-10 06:45:02 (1s)
2012-12-13 06:45:01 - 2012-12-13 06:45:03 (2s)
```
???