What are the similarities and differences of using PostgreSQL between Japan and Europe?

PGConf.ASIA 2016
K.K.ASHISUTO
Yoko Takase
Who am I?

Yoko Takase

Working for K.K.Ashisuto as a database engineer
Engaging in PostgreSQL since 2009 and
EDB Postgres since 2011
Living in UK for 2.5 years
writing up how database things are in Europe
Agenda

- PostgreSQL in Europe
- User Cases in Europe and Japan
- Points of views
Activities for PostgreSQL in Europe

- Develop some main features and popularize PostgreSQL
  - AXLE Project
  - Various events
    - Conference
    - MeetUP

Copyright © 2016 K.K. Ashisuto
Applying to Big Data

- AXLE Project
  (Advanced Analytics for Extremely Large European Databases)
  - Database Size 10TB - 100TB
  - Fast and Highly Secure Business Intelligence
  - Great contribution to major features PostgreSQL 9.5 onward
    - BRIN Index
    - tablesample
    - Bi-Directional Replication
    - Column store/compression(COAST)

http://axleproject.eu/

Copyright © 2016 K.K. Ashisuto
PostgreSQL Events in Europe

- PGDayUK
  - 1Day Event in UK
  - 100 attendees in 2016
  - Fintech user case
- London MeetUp
  - 3 or 4 times in a year

https://wiki.postgresql.org/wiki/Events
PostgreSQL Events in Europe

- PostgreSQL Conference Europe
  - Biggest Conference
  - More than 400 attendees in 2016
  - Major developers
PostgreSQL Events in Europe

- **FOSDEM**
  - *(Free Open Source Development European Meeting)*
  - Event for the Open Source Development (2Days)
  - 5,000-6,000 attendees every year
  - PostgreSQL, MySQL, Container, etc
Event Reports and Technical News

- Euro IT Journal (Web)
  - http://www.ashisuto.co.jp/corporate/column/technical-column/
- ASHISUTO (Magazine)
Agenda

- PostgreSQL in Europe
- User Cases in Japan and Europe
- Points of views
User Case in Japan and Europe

- High Availability
- Performance
- Migration
- Management
High Availability
Public System (Macedonia)

- National Health Management System

**Background**
- Need to select software under various constraints and tight budget
- Need to **build high availability system in minimum downtime**

**Effectiveness**
- Achieve a required level of high availability by **Streaming Replication**
- **170,000 prescriptions** per day

Source: https://raw.githubusercontent.com/sorsix/pgconfeu2015/master/MojTerminPGConfEU.pdf
GoCardless (UK)

- Online Payment System

**Background**
- Build the system with a small start
- *Not allowed even ten-odd seconds of API downtime*

**Effectiveness**
- High Availability by *Streaming Replication and Pacemaker*
- Reduce impact of user transactions by setting statement-level timeout

Source: https://www.youtube.com/watch?v=Tu-cf-Jki60
● Replace OracleDB EE RAC to EDB Postgres in all store systems

Background

- Reconsider IT cost with system replacement
- Need **same level of high availability as OracleDB EE RAC**
- Performance for thousands of concurrency

Effectiveness

- Sharp cost reduction by subscription license
- High Availability by **Streaming Replication and pgpool-II**
- Scale-up by multi CPU
Major Technical Features for HA User Case

- Streaming Replication
  - PostgreSQL Standard Replication, implemented since 9.0

- Complementary modules

<table>
<thead>
<tr>
<th>Product</th>
<th>Explanation</th>
</tr>
</thead>
<tbody>
<tr>
<td>HA Proxy</td>
<td>TCP/HTTP load-balancing proxy <a href="http://www.haproxy.org/">http://www.haproxy.org/</a></td>
</tr>
<tr>
<td>pgpool-II</td>
<td>Middleware that works between PostgreSQL servers and a PostgreSQL database client <a href="http://www.pgpool.net/mediawiki/jp/">http://www.pgpool.net/mediawiki/jp/</a></td>
</tr>
<tr>
<td>pacemaker</td>
<td>HA Clusterware <a href="http://linux-ha.osdn.jp/wp/">http://linux-ha.osdn.jp/wp/</a></td>
</tr>
<tr>
<td>Repmgr</td>
<td>A tool suite for managing replication and failover <a href="http://repmgr.org/">http://repmgr.org/</a></td>
</tr>
<tr>
<td>Patroni</td>
<td>Template to create your own customized, high-availability solution</td>
</tr>
<tr>
<td></td>
<td><a href="https://github.com/zalando/patroni">https://github.com/zalando/patroni</a></td>
</tr>
<tr>
<td>Stolon</td>
<td>A cloud native PostgreSQL manager for PostgreSQL high availability</td>
</tr>
<tr>
<td></td>
<td><a href="https://github.com/sorintlab/stolon">https://github.com/sorintlab/stolon</a></td>
</tr>
</tbody>
</table>

Transfer wal consecutively...

Master  Slave
Performance
TomTom (Netherlands)

- GPS Navigation System with global large market share

**Background**

- Need **GIS features quality**
- Need **high performance of several hundred thousand requests / sec**

**Effectiveness**

- Provide **highly precise geographical information** by PostGIS and PostgreSQL
- Achieve **600,000 queries and 15,000 data insertions / sec**

Source: https://wiki.postgresql.org/images/e/e2/Postgresql_at_Tomtom_-_lessons_learned.pdf
Yammer (US)

- Enterprise Social Networking Service

**Background**
- Build a system in minimum cost
- Need DBMS with high reliability and stability
- Need **performance for high workload and increasing requests**

**Effectiveness**
- High Availability by **Streaming Replication**
- Achieve **30,000 requests per sec to Master server**

Source: https://wiki.postgresql.org/images/7/7a/Nguyen_and_Gul_-_PGConf.EU_2016.pdf
Checking customer shop visit system

**Background**

- Need **DBMS with high quality and reliability of GIS features**
- Need **enough performance for thousands of concurrency at peak time**

**Effectiveness**

- **PostGIS**, having plenty of operators for geographical information
- Achieve **3,000 transactions and 16,000 queries/sec**

Source: https://www.pgecons.org/wp-content/uploads/2013/12/7c14ac1727a38c22295af840fc613321.pdf
Energy Management System (EMS)

**Background**
- Need reliability and availability for IoT System
- **Scale up** and license style for the growth of business
- Need technical support for enterprise service

**Effectiveness**
- 1.3 million sensor data per day
- Flexible scale up and scale out
- 24h365d support for enterprise system

Source: https://www.ashisuto.co.jp/case/industry/information/__icsFiles/afieldfile/2016/02/05/PPlus_Daiwa-Soken-BI_20160205b.pdf
Major Technical Features for Performance User Case

- Improvement of performance features as DBMS

- **Shared buffers contention**
  - by increasing read transactions

- **Data file extension**
  - by increasing write transactions

- Revealed new bottleneck

- Multi CPU Disk Acceleration

- More than 10GB shared buffers

- Concurrent writing of WAL

- Improved by 9.6

- Extend multi blocks of data files
Migration
- Migrate OracleDB EE of stock management system for 13 distribution centers to EDB Postgres

**Background**
- Place stock management system to virtual environment and need to reduce license cost of OracleDB on virtual environment
- Need **DBMS to use OracleDB application and skills efficiently**

**Effectiveness**
- Succeed 90% cost reduction of license and support
- **Smooth migration by OracleDB compatibility**

Source: https://www.ashisuto.co.jp/case/industry/retail/edb_coopnet.pdf
Migrate OracleDB EE of ISP account management system to EDB Postgres

**Background**
- **Keep using current OracleDB applications**
- **No stress to OracleDB engineers** by leaning new DBMS technology

**Effectiveness**
- **Change only connection statements of program** due to compatibility of SQL and procedures
- **No need of specific technical training** to OracleDB engineers due to similar interfaces

Source: https://www.ashisuto.co.jp/case/industry/information/__icsFiles/afieldfile/2015/10/23/PPAS_user_01J_0626_x4.pdf
- Migrate OracleDB EE of EDI System, used by 300 companies to EDB Postgres

**Background**

- Rework IT cost due to reaching maximum number of companies, using system
- Need to reduce license cost of OracleDB on virtual environment
- Minimum cost for rewriting application

**Effectiveness**

- **98% of SQL compatibility** in 30,000 steps of program
- **Smooth migration** by EDB Migration Toolkit
## Major Technical Features for Migration User Case

- Use OracleDB applications efficiently
  - Can create stored procedures and packages
  - Support OracleDB specific SQL and PL/SQL syntax

### Supported Major Objects

<table>
<thead>
<tr>
<th>Supported Major Objects</th>
<th>SQL, PL/SQL</th>
</tr>
</thead>
<tbody>
<tr>
<td>Tables (with partitions)</td>
<td>Outer join(+)</td>
</tr>
<tr>
<td>Indexes(B*tree)</td>
<td>Commit / rollback in stored procedures</td>
</tr>
<tr>
<td>Constraints</td>
<td>Set operators(minus)</td>
</tr>
<tr>
<td>Views</td>
<td>Concatenated NULL and empty string</td>
</tr>
<tr>
<td>Sequences</td>
<td>ROWNUM</td>
</tr>
<tr>
<td>Functions</td>
<td>No alias for sub-query in FROM clause</td>
</tr>
<tr>
<td></td>
<td>PL/SQL syntax</td>
</tr>
<tr>
<td></td>
<td>OracleDB specific function(nvl)</td>
</tr>
<tr>
<td></td>
<td>OracleDB built-in packages</td>
</tr>
<tr>
<td></td>
<td>SQL Hint</td>
</tr>
</tbody>
</table>
Management
Zalando (Germany)

- Largest online fashion website in Europe

**Background**

- Need to achieve required performance
- **Efficient DB maintenance and monitoring**

**Effectiveness**

- **Shorten the duration time of DB backup and migration** by distributing data into several servers
- Develop unique performance monitoring tool **PGObserver**

Source: https://docs.google.com/presentation/d/1gJL93iGNxbo9B2Y2sVnWVQfPBGIIeJetoZlp6UitUwXM/present?ueb=true#slide=id.g11d1f2970_01
Yammer (US)

● Enterprise Social Networking Service

Background

• Need to use DBMS with reliability and stability
• Need **plenty of tools for DB management**

Effectiveness

• Good use of **OSS modules, Barman and pgBadger**
• Develop **own tool for monitoring replication lag**

Source: [https://wiki.postgresql.org/images/7/7a/Nguyen_and_Gul_-_PGConf.EU_2016.pdf](https://wiki.postgresql.org/images/7/7a/Nguyen_and_Gul_-_PGConf.EU_2016.pdf)
ASHISUTO - Attendance System - (Japan)

- MosP, Domestic Open Source Attendance System

**Background**

- Need DBMS with high reliability and stability
- **Reduce time for checking the bottleneck of performance issues**

**Effectiveness**

- **Regularly monitoring** of DB Server by EDB Postgres Enterprise Manager (PEM)
- **Suggestions for improving performance** by PEM components, Index Advisor and SQL Profile

Source: https://www.ashisuto.co.jp/case/industry/information/__icsFiles/afieldfile/2016/03/03/EDB_ashisuto_160205-2.pdf
## Major Technical Features for Management User Case

- **Tools for Management monitoring and SQL analysis**

<table>
<thead>
<tr>
<th>Product</th>
<th>Explanation</th>
</tr>
</thead>
<tbody>
<tr>
<td>pgAdmin</td>
<td>Management tool <a href="https://www.pgadmin.org/">https://www.pgadmin.org/</a></td>
</tr>
<tr>
<td>PGObservable</td>
<td>Monitoring tool <a href="http://zalando.github.io/PGObserver/">http://zalando.github.io/PGObserver/</a></td>
</tr>
<tr>
<td>pg_statsinfo</td>
<td>Utility of monitoring statistics and the activity <a href="http://pgstatsinfo.projects.pgfoundry.org/index_ja.html">http://pgstatsinfo.projects.pgfoundry.org/index_ja.html</a></td>
</tr>
<tr>
<td>pgBadger</td>
<td>Log analysis report <a href="https://github.com/dalibo/pgbadger">https://github.com/dalibo/pgbadger</a></td>
</tr>
</tbody>
</table>

- **Tools for Maintenance**

<table>
<thead>
<tr>
<th>Product</th>
<th>Explanation</th>
</tr>
</thead>
<tbody>
<tr>
<td>Barman</td>
<td>Administration tool for disaster recovery <a href="http://www.pgbarman.org/">http://www.pgbarman.org/</a></td>
</tr>
<tr>
<td>pgcompact</td>
<td>A tool to reduce bloat for tables and indexes <a href="https://github.com/grayhemp/pg-toolkit#pgcompact">https://github.com/grayhemp/pg-toolkit#pgcompact</a></td>
</tr>
</tbody>
</table>
Agenda

- PostgreSQL in Europe
- User Cases in Europe and Japan
- Points of view
Similarities

- Increasing the number of high reliability and performance system using PostgreSQL
  - Streaming Replication
  - Basic performance
- PostgreSQL: new system
- EDB Postgres: system replacement

It is widely recognized PostgreSQL sufficient features as standard DBMS

Right DBMS in the right place
Diffrences

- tools
  - Europe
    - Widely use plenty of OSS tools for PostgreSQL
    - Develop and manage specific tools for their own system
  - Japan
    - Limited using PostgreSQL features and major OSS tools

It may be said that this is because of difference of IT structure. The number of general private companies using PostgreSQL has been increasing.

The more information you get, the more widely you can use OSS.
Utilization of OSS (Postgres) accelerates offensive IT

Planned system investment
- System on physical server

Prompt action against changes of business
- Server Virtualization
- Cloud
- IoT

Server integration to virtual environment
Quick deploy
Small start
Unpredicted business volume
Efforts to utilize OSS(Postgres)

Making the most of information in conference

- PGConf.ASIA

PostgreSQL
- Sufficient features as standard DBMS

New

Replace

EDB Postgres
- Efficiently utilization of OracleDB application and skills

PostgreSQL communities in Japan
- Japan PostgreSQL User Group(JPUG)
- PostgreSQL Enterprise Consortium(PGECons)
アシスト
「お客様の最高」のために

※The names of companies and products written in this document are trademarks of each company or registered trademarks.
※Oracle and Java are the registered trademarks of Oracle Corporation, and its subordinate or subsidiary companies in USA and other countries.