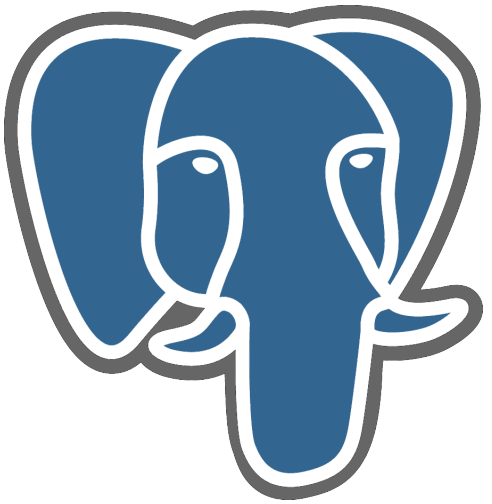
How have PostgreSQL community and Market evolved in Japan

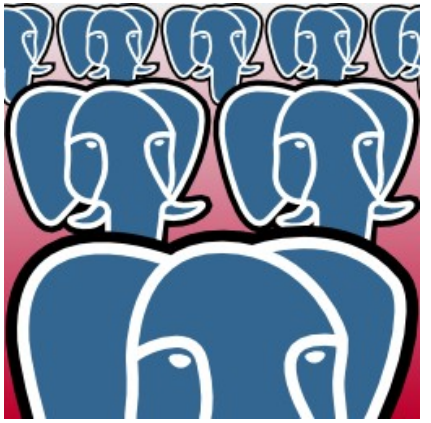
Tatsuo Ishii
SRA OSS, Inc. Japan

SRA OSS, INC.

Who am I?

- OSS developer
  - PostgreSQL committer
  - pgpool-II community lead
- Community activity
  - Co founder of Japan PostgreSQL Users Group
  - Secretary general of PostgreSQL Enterprise Consortium
- Branch manager of SRA OSS, Inc. Japan





About SRA OSS, Inc.

- Running PostgreSQL business from 1999
- Established in 2005 as SRA OSS, 100% owned by SRA, Inc.
- Our main businesses areas are follows:
  - Open source software support
    - PostgreSQL, pgpool-II, Hinemos, Zabbix and many more
    - No MySQL:-)
    - 600 support contracts
  - PostgreSQL based product: PowerGres
    - Transparent encryption of data, redundant WAL
  - Training and Consulting
- Contributions
  - pgbench
  - multi byte characters
  - Windows port
  - Recursive query

PowerGres

ZABBIX

CERTIFIED PARTNER

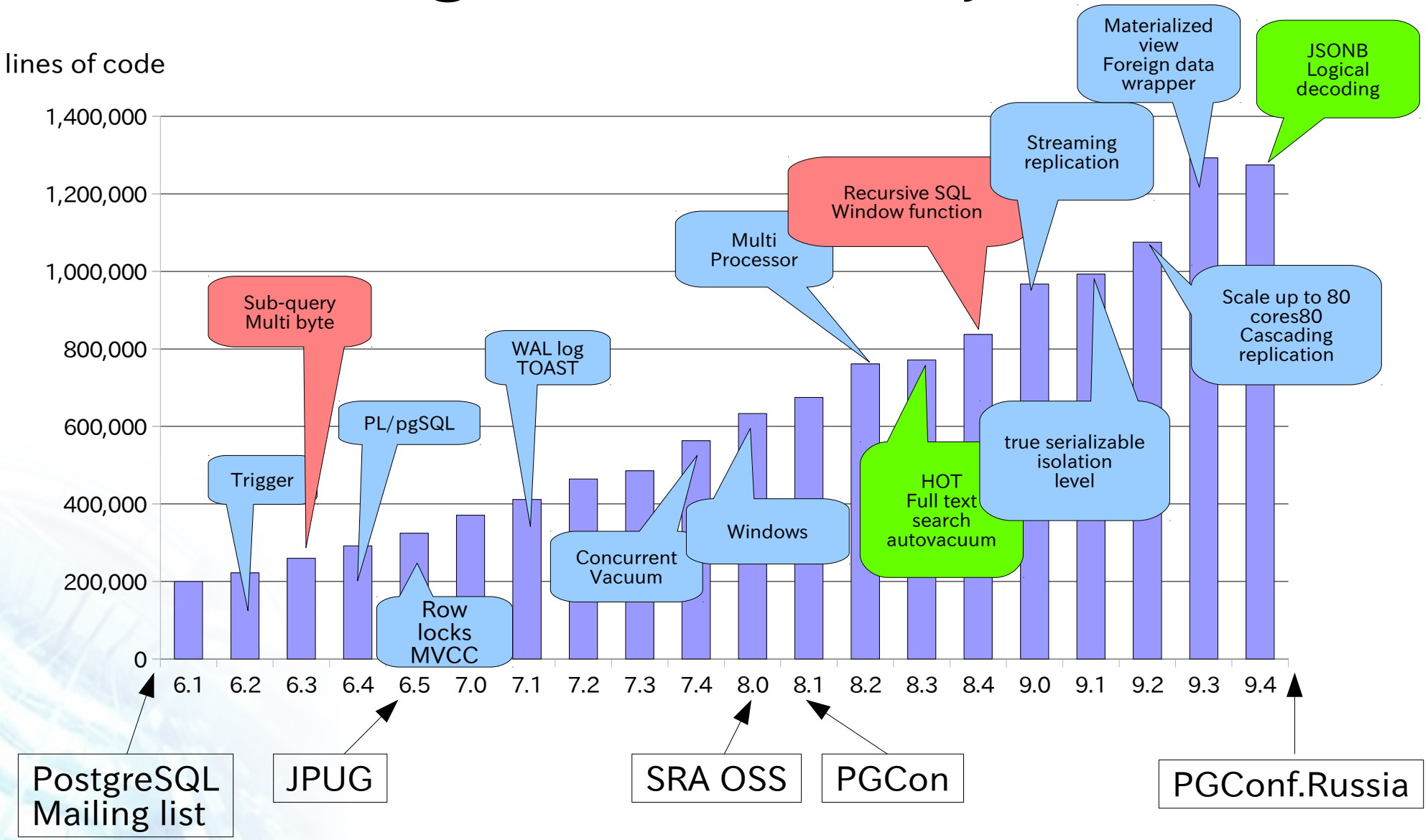
Hinemos

SRA OSS, INC.

PostgreSQL community history in Japan

- In 1995 Postgres95 mailing list started (turned into PostgreSQL mailing list in 1996)
- In 1998 PostgreSQL 6.3.1 merged the multi byte patch
- PostgreSQL documentation translated into Japanese
- In 1998 the first PostgreSQL book published
- In 1999 Japan PostgreSQL User's Group established
- In 2012 PostgreSQL Enterprise Consortium estableished

PostgreSQL history



SRA OSS, INC.

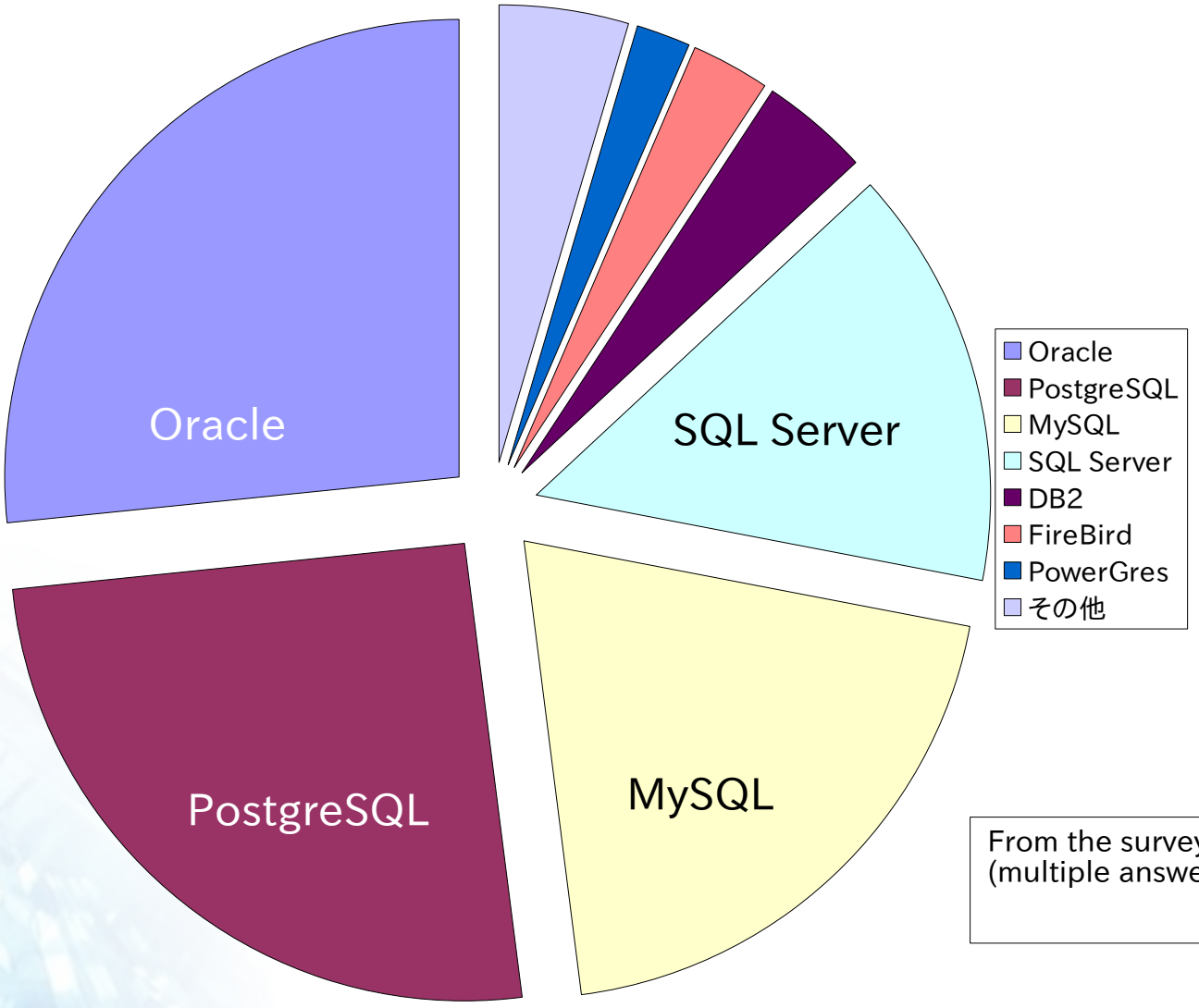
Japan PostgreSQL Users Group

SRA OSS, INC.

Activity of Japan PostgreSQL Users Group (JPUG)

- Established in 1999
- Major activities include:
  - Translating PostgreSQL documents
  - Mailing lists
  - Seminars
  - Regional branches
  - Tech blogs
- JPUG helps:
  - to promote PostgreSQL to people who are not familiar with English
  - beginners of PostgreSQL in ML and seminars
    - Beginners are often spoiled in OSS community :-)
  - to promote PostgreSQL in the early days of OSS in Japan

PostgreSQL is in the second position in the Japan's database market



From the survey result of “Linux World 2007”
(multiple answers was permitted)

PostgreSQL Enterprise Consortium

SRA OSS, INC.

About PostgreSQL Enterprise Consortium

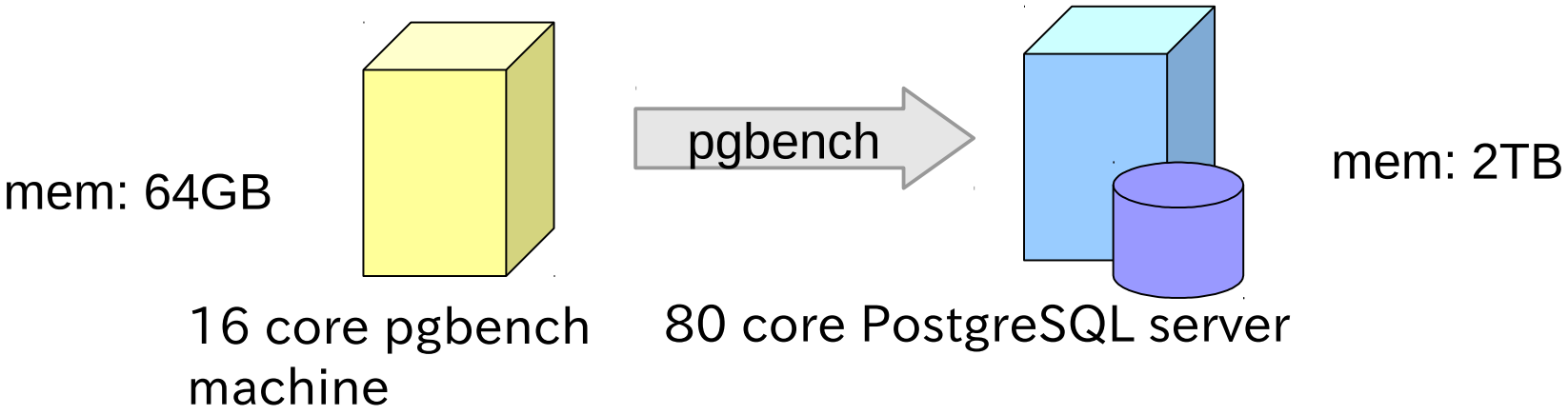
- What is PostgreSQL Enterprise Consortium?
  - PostgreSQL Enterprise Consortium (PGECons) is a non-profit organization aiming at promoting PostgreSQL in production use, especially in mission critical area
- Who are the members of PGECons?
  - Leading IT/OSS companies in Japan
  - 17 regular members and 30 general members (as of 2015/1/26)

Major activities of PGECons

- Collaborative verification
  - PGECons performs necessary verification colla boratively by using resources provided by our member companies
- Promoting PostgreSQL
  - Through seminars PGECons presents technical reports created by the activities above. Also PG ECons provides various case studies, which are important for those who are trying to adopt Pos tgreSQL.
- http://www.pgecons.org/en/about

Evaluation details

- Increase the number of concurrent clients on many core machines
- pgbench -h [host] -p [port] [dbname] -c [c] -j [j] -T 30 -n -f custom.sql
  - Increase number of clients (-c)
  - Number of threads is ½ of -c
  - pgbench dedicated machine is prepared

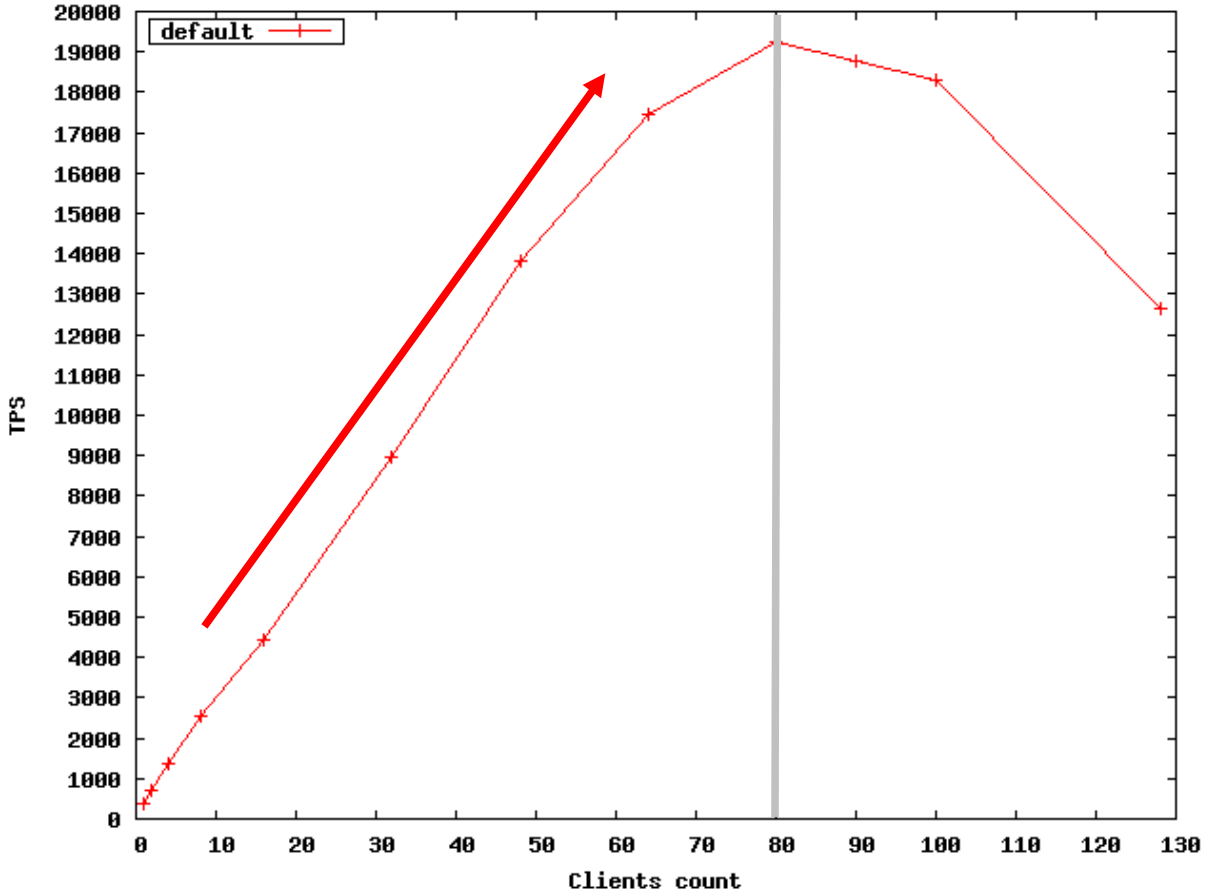


Custom.sql

```
\set nbranches :scale
\set ntellers 10 * :scale
\set naccounts 100000 * :scale
\set row_count 10000
\set aid_max :naccounts - :row_count
\setrandom aid 1 :aid_max
\setrandom bid 1 :nbranches
\setrandom tid 1 :ntellers
\setrandom delta -5000 5000

SELECT count(abalance) FROM pgbench_accounts WHERE
aid BETWEEN :aid and :aid + :row_count;
```

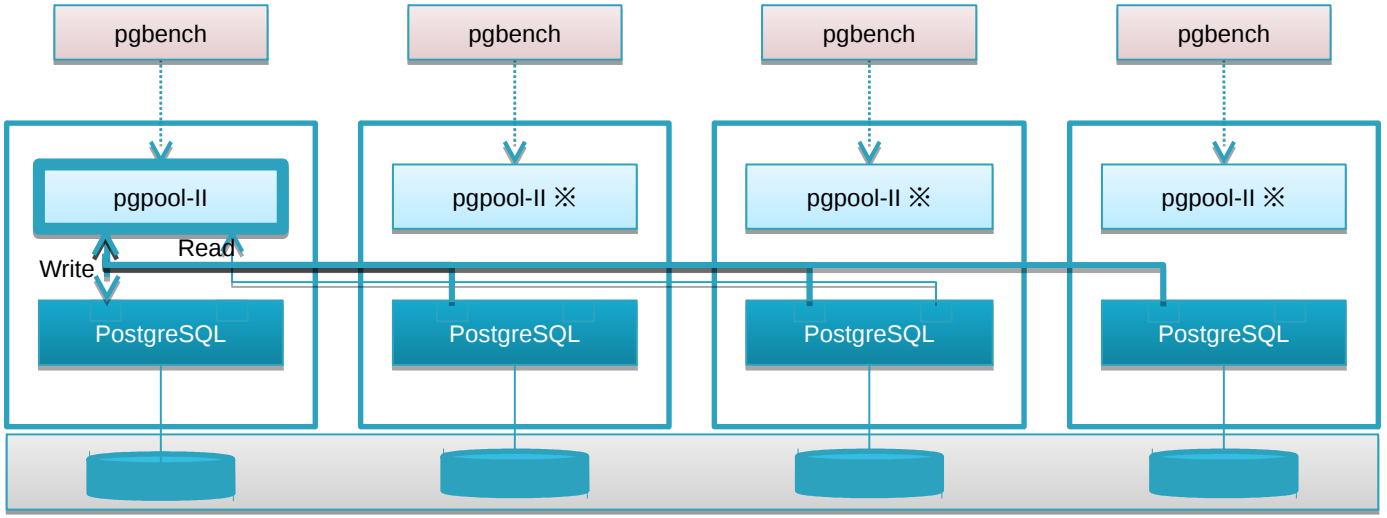
The result!



Scale factor: 1000
Number of rows:
a hundred million(15GB)

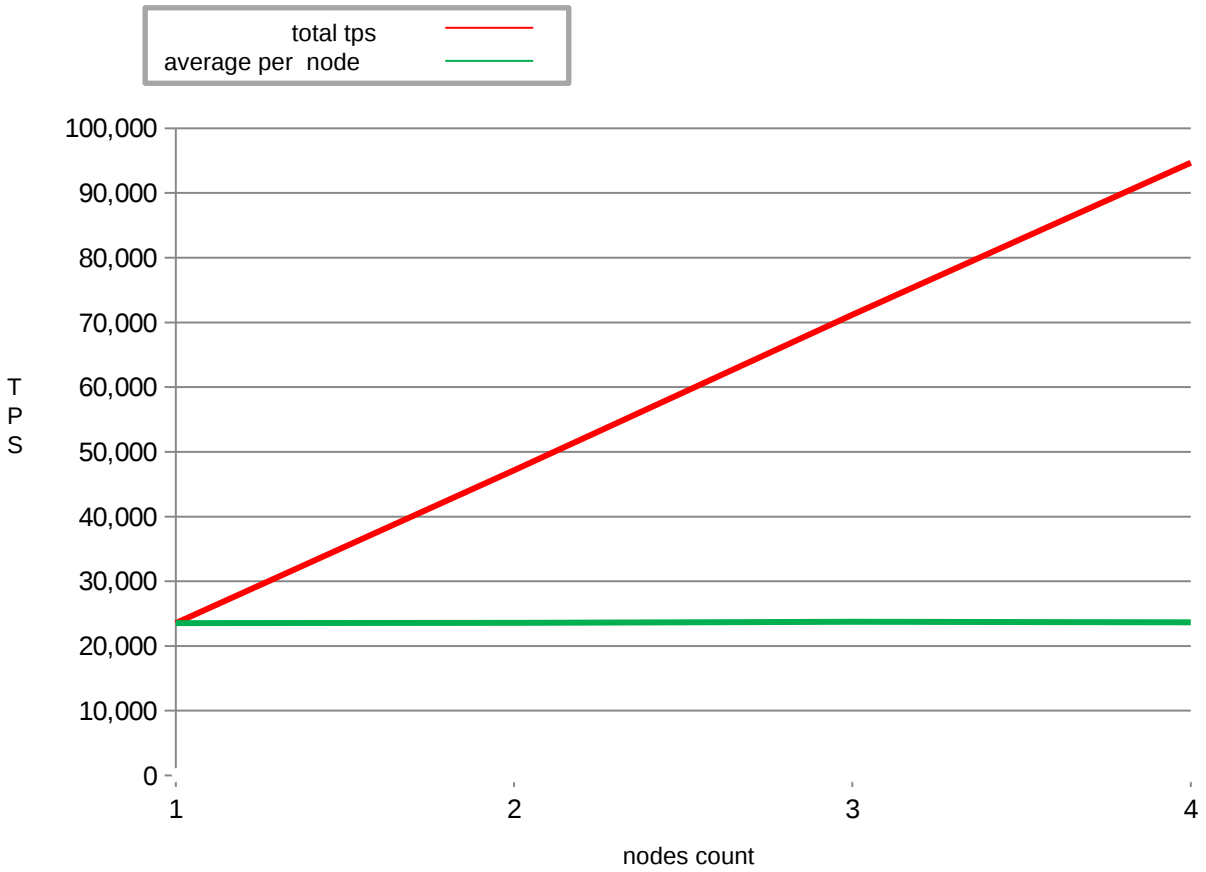
Benchmarking with shared nothing clusters

- pgpool-II configuration
  - Using pgpool-II(3.2.1) & PostgreSQL(9.2.1)
  - Pgpool-II is configured to use native replication mode(synchronous re plication)
- Pgbench configuration
  - Up to 4 pgbench clients used. TPS is defined as the summary of each pgbench TPS
  - Pgbench default scenario is used for write query test
  - Read test uses custom scenario because default scenario(-S) is too subtle



※　Arrows for second node or above is omitted to avoid complexity of the figure

Read query result



- Settings
  - PostgreSQL shared memory: 32GB
  - Pgbench scale factor: 1,000
  - Each duration(-T): 300 seconds
  - Number of concurrent sessions (-c): 100
  - Number of worker threads (-j): 20

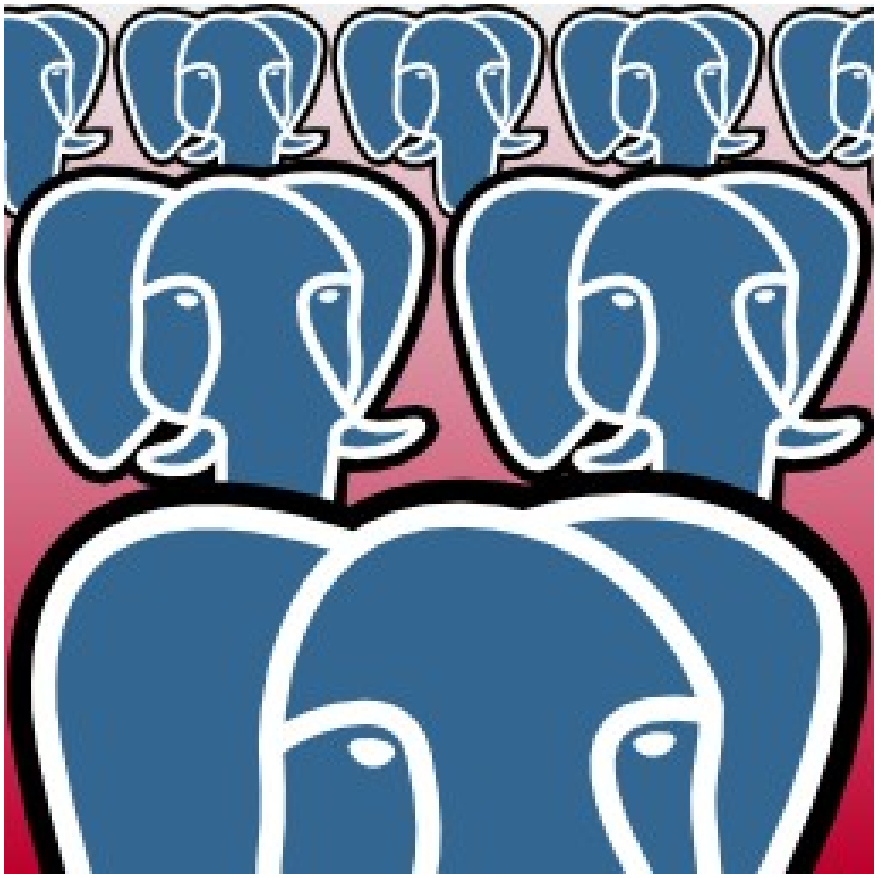
- The custom scenario. Randomly extracts 2,000 rows.

```
\set nbranches :scale
\set ntellers 10 * :scale
\set naccounts 100000 * :scale
\set range 2000
\set aidmax :naccounts - :range
\setrandom aid 1 :aidmax
\setrandom bid 1 :nbranches
\setrandom tid 1 :ntellers
\setrandom delta -5000 5000
SELECT count(abalance) FROM
pgbench_accounts WHERE aid
BETWEEN :aid and :aid + :range;
```

- Total TPS increases as number of nodes increases(scale out)

What is pgpool-II?

- Proxy like cluster management tool for PostgreSQL and compatible servers
- BSD licensed OSS
- Completely separated from PostgreSQL (not modified PostgreSQL like Postgres-XC)
- Can be used with wide variety of PostgreSQL versions
- Rich function sets to enhance performance and availability of PostgreSQL
- Actively developed
  - Once a year major version up
  - 2-3 times a year of minor version up



How pgpool-II works(1)

Looks like as if PostgreSQL from application's point of view

SELECT/UPDATE etc.

Check if the query can be load balanced

pgpool-II

Distribute SELECTs to servers to enhance performance (load balancing)

SELECT
UPDATE etc.

SELECT

SELECT

PostgreSQL primary

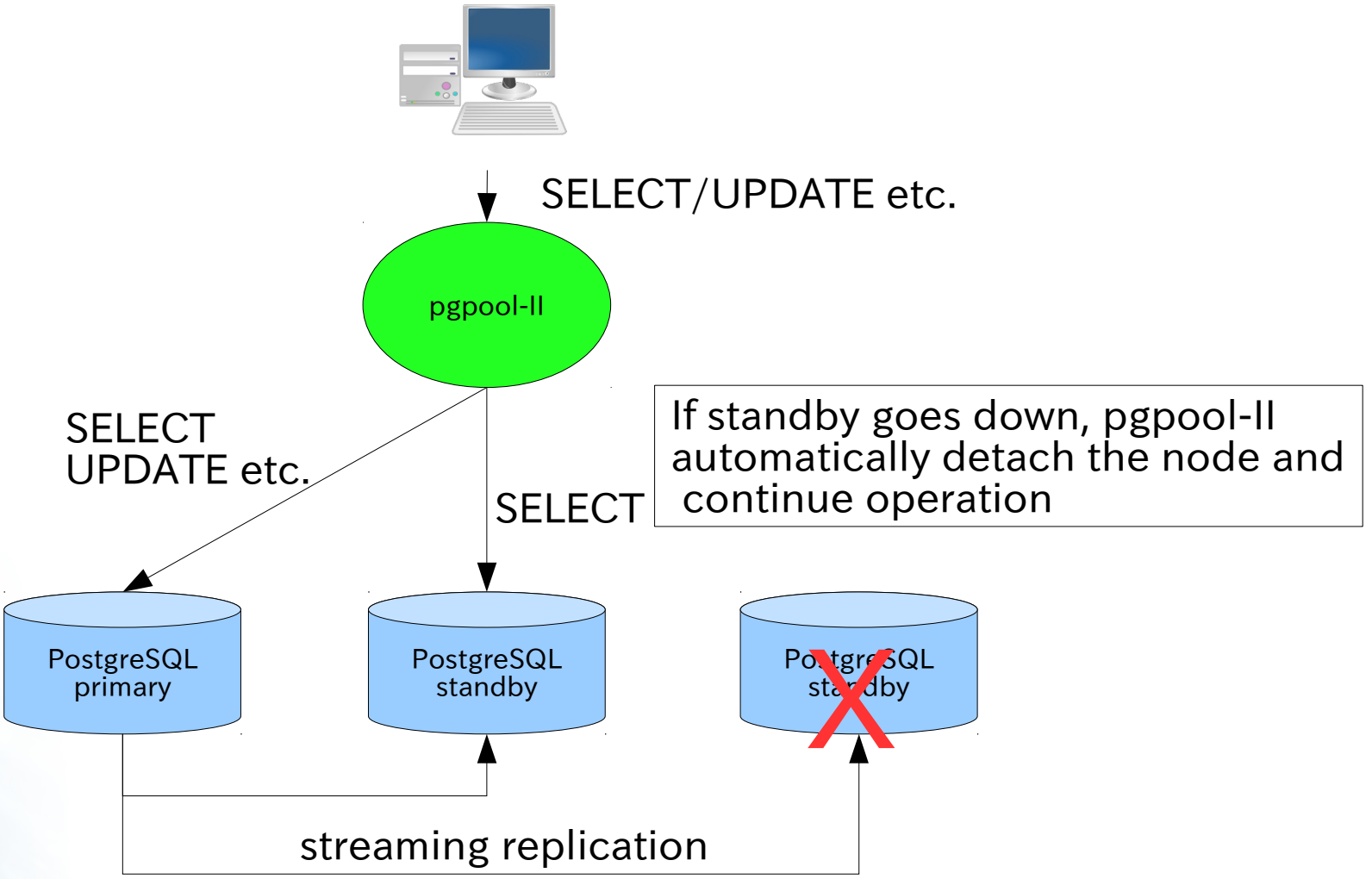
PostgreSQL standby

PostgreSQL standby

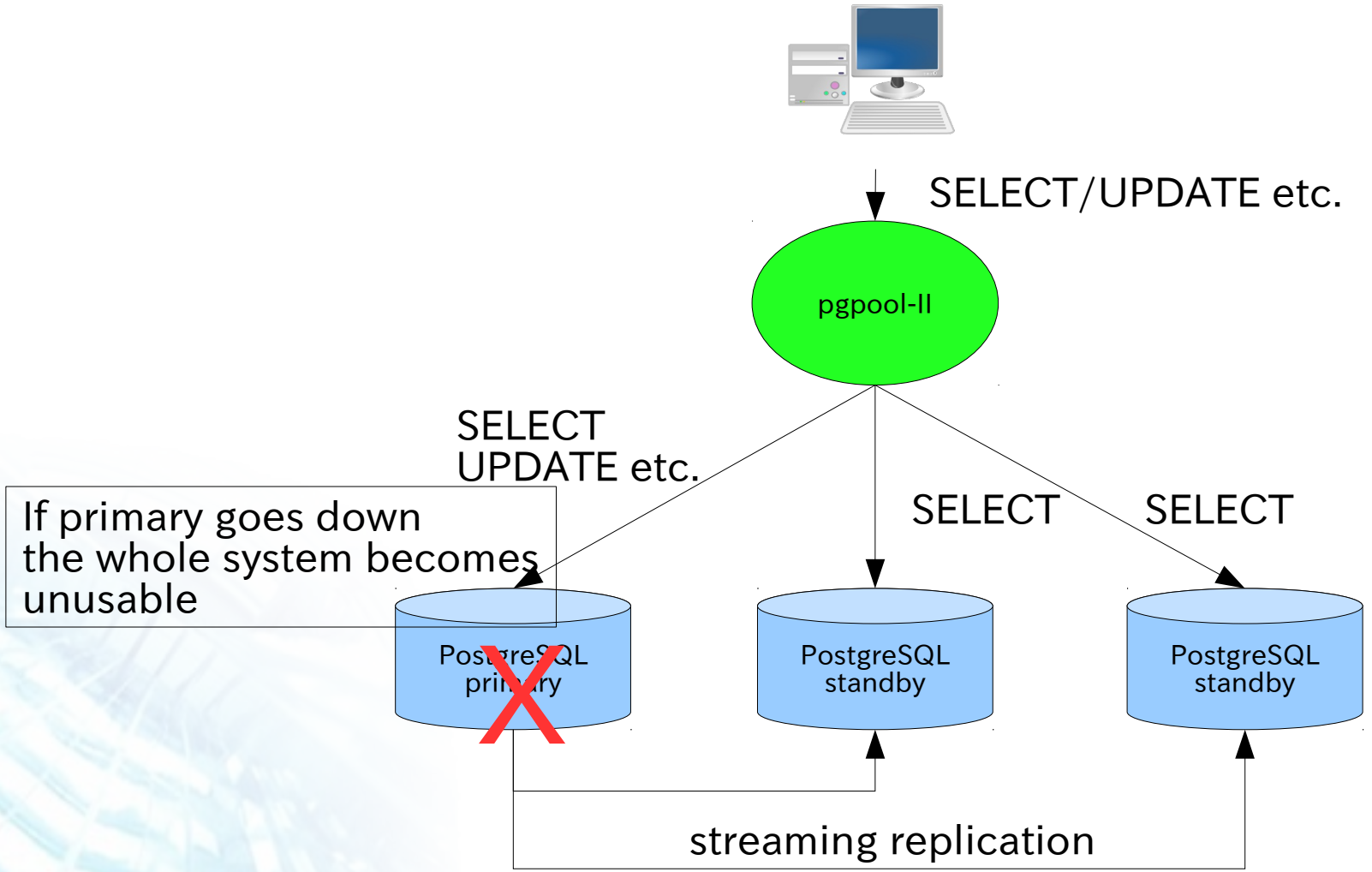
UPDATE is propagated by streaming replication

streaming replication

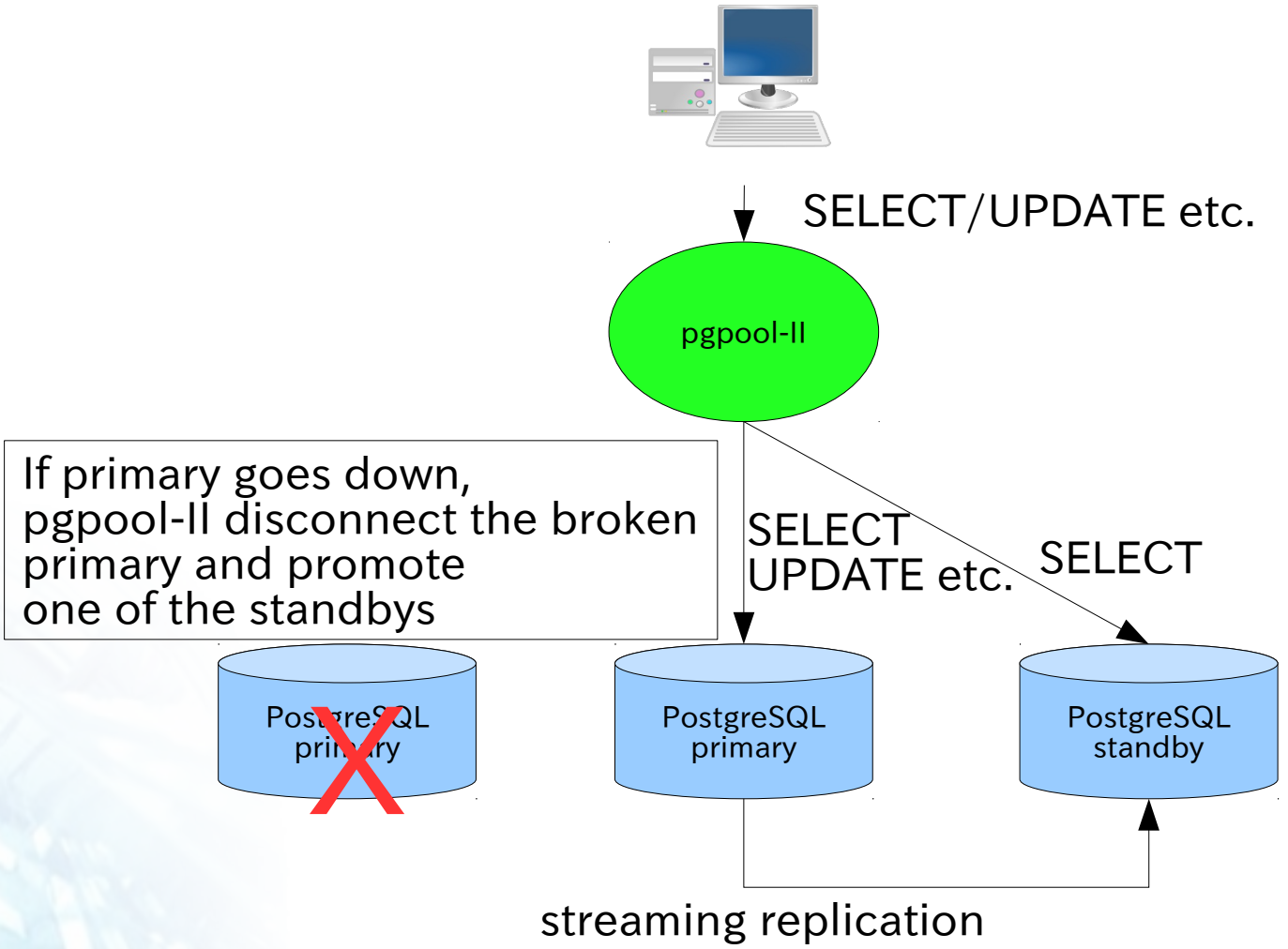
How pgpool-II works(2)



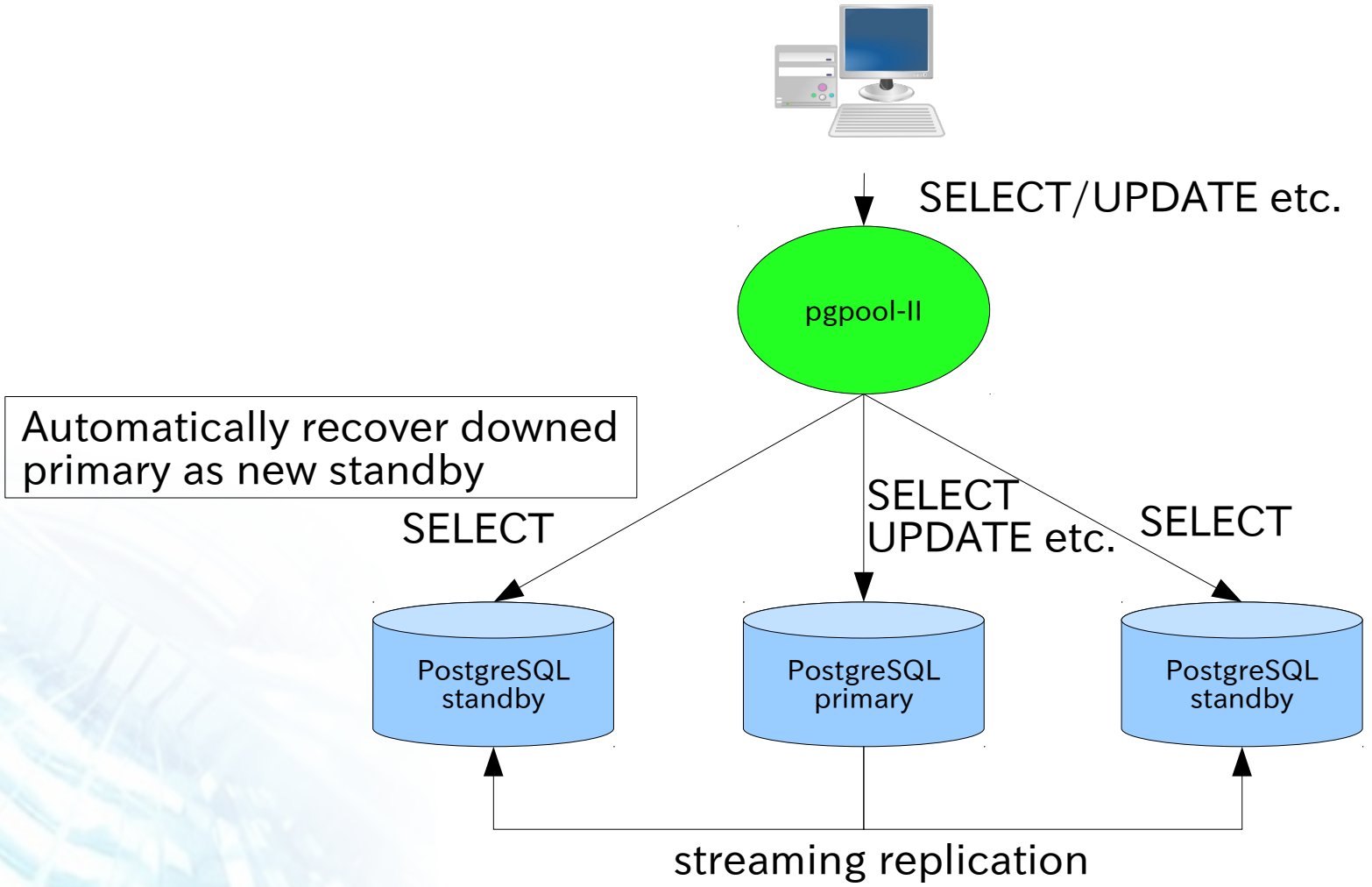
How pgpool-II works(3)



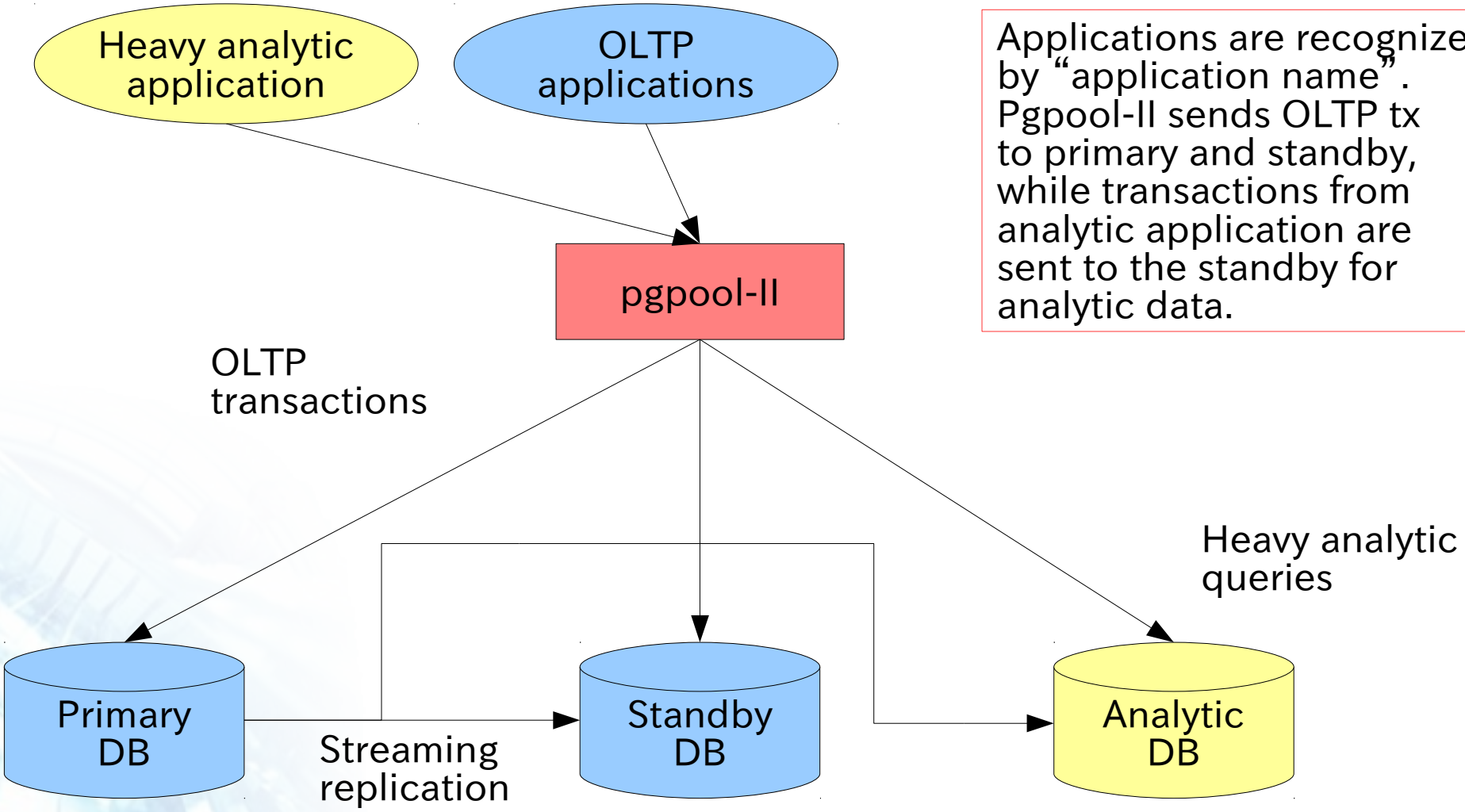
How pgpool-II works(4)



How pgpool-II works(5)



Run heavy analytic queries without disturbing OLTP transactions



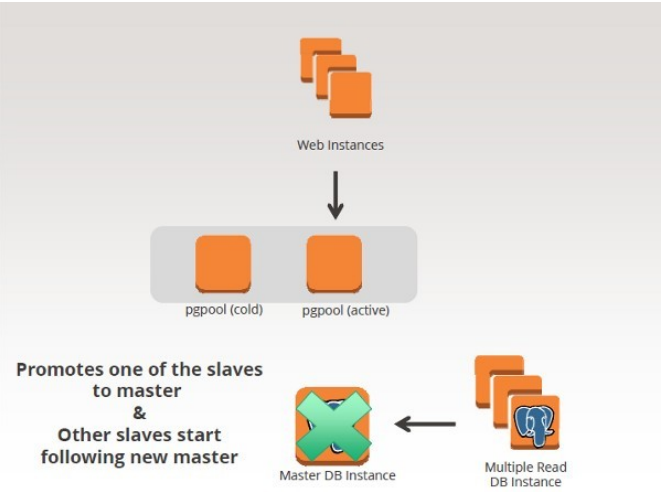
Applications are recognized by “application name”. Pgpool-II sends OLTP tx to primary and standby, while transactions from analytic application are sent to the standby for analytic data.

How to use query redirect in pgpool-II

- Parameters to control read query load balancing
  - database_redirect_preference_list
    - DB_name:node_spec,DB_name:node_spec...
    - Regular expression can be used for DB name
    - “Node spec” can be either of:
      - “primary”, “standby”, or node number
  - app_name_redirect_preference_list
    - app_name:node_spec,app_name:node_spec...
    - Regular expression can be used for app name
    - “Node spec” can be either of:
      - “primary”, “standby”, or node number
  - If database_redirect_preference_list and app_name_redirect_preference_list conflicts each other,は app_name_redirect_preference_list is used

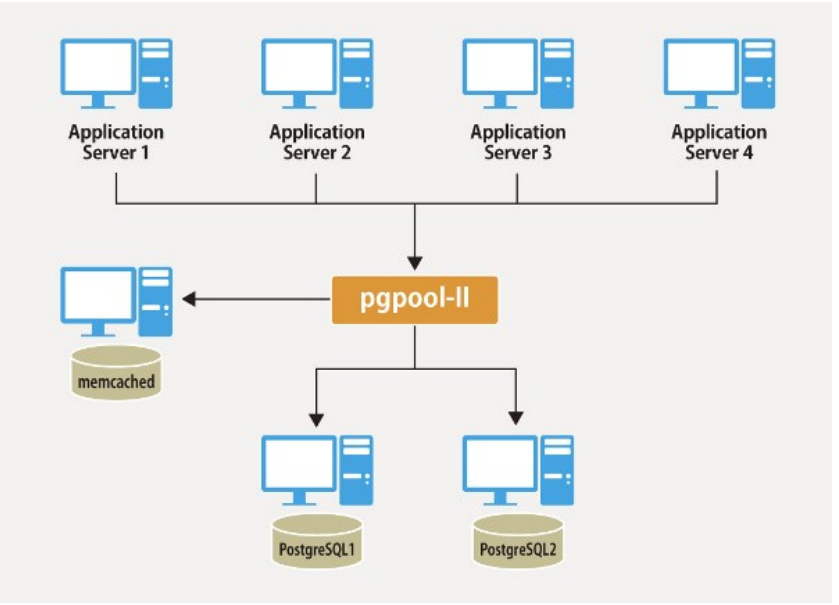
Case study:Gengo

- Provides translation service via Web or API for over 37 languages. 13,000 translators all over the world
- All servers are running on Amazon Web Services (AWS) using PostgreSQL
- Had troubles w occasional DB down and performance problem
- To achieve more reliability and performance, pgpool-II is used with streaming replication
- In addition to this they need to move to the recent PostgreSQL version
- All migration efforts went smoothly in the planned maintenance window
- http://www.sraoss.co.jp/case_study/gengo_en.php



Case study: DAI-ICHI HOKI

- DAI-ICHI HOKI co., ltd. is a company that sells law-related books and provides online data
- They employed PostgreSQL as a database, and pgpool-II for redundancy and to improve performance
- Before the data update was once a month which was frustrating for their customers
- With the new system data update is once a day!
- The auto fail over is managed by pgpool-II, which allows the system to continue the operation when the master DB goes down (it actually happened)
- pgpool-II's in memory query cache boosts the performance
- http://www.sraoss.co.jp/case_study/daiichihoki_en.php



Summary

- PostgreSQL is quite popular in Japan
- The reason behind this is user's community (JPUG) and company's community (PGECons)
- pgpool-II
  - pgpool-II: http://pgpool.net
- Case studies
- URLs