PostgreSQL 最前線

～ 今大体どうなっているの？ ～

オープンソースカンファレンス Online Spring
2021-03-06 10:00 - 10:45 (D 会場)
日本 PostgreSQL ユーザ会 理事 高塚 遥

PostgreSQL
http://www.postgresql.jp

TOC ：

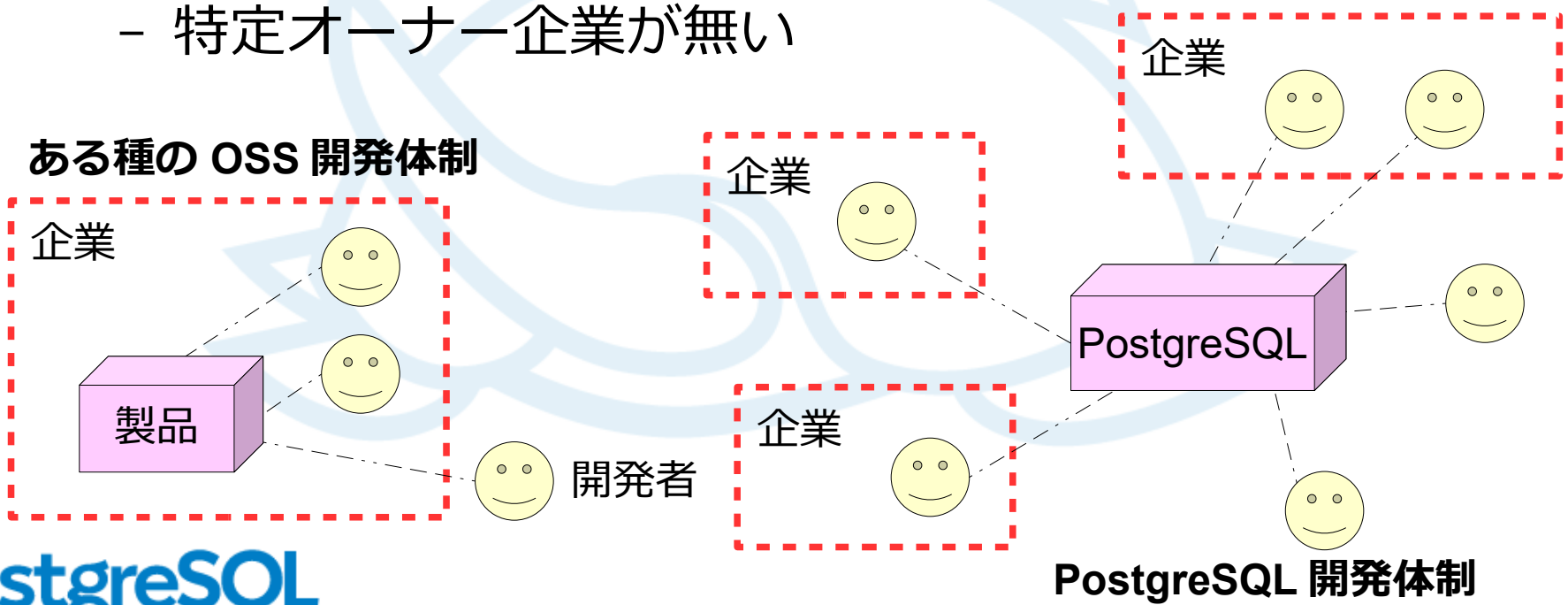
- 「 PostgreSQL は今どうなっている？」を 45 分で解説
- 「 PostgreSQL ってなんだったっけ？」という人でも大丈夫

講演者 :

- 高塚 遥
- 日本 PostgreSQL ユーザ会理事
- 仕事ではヘルプデスク、コンサルティングなど、PostgreSQL 支援業務

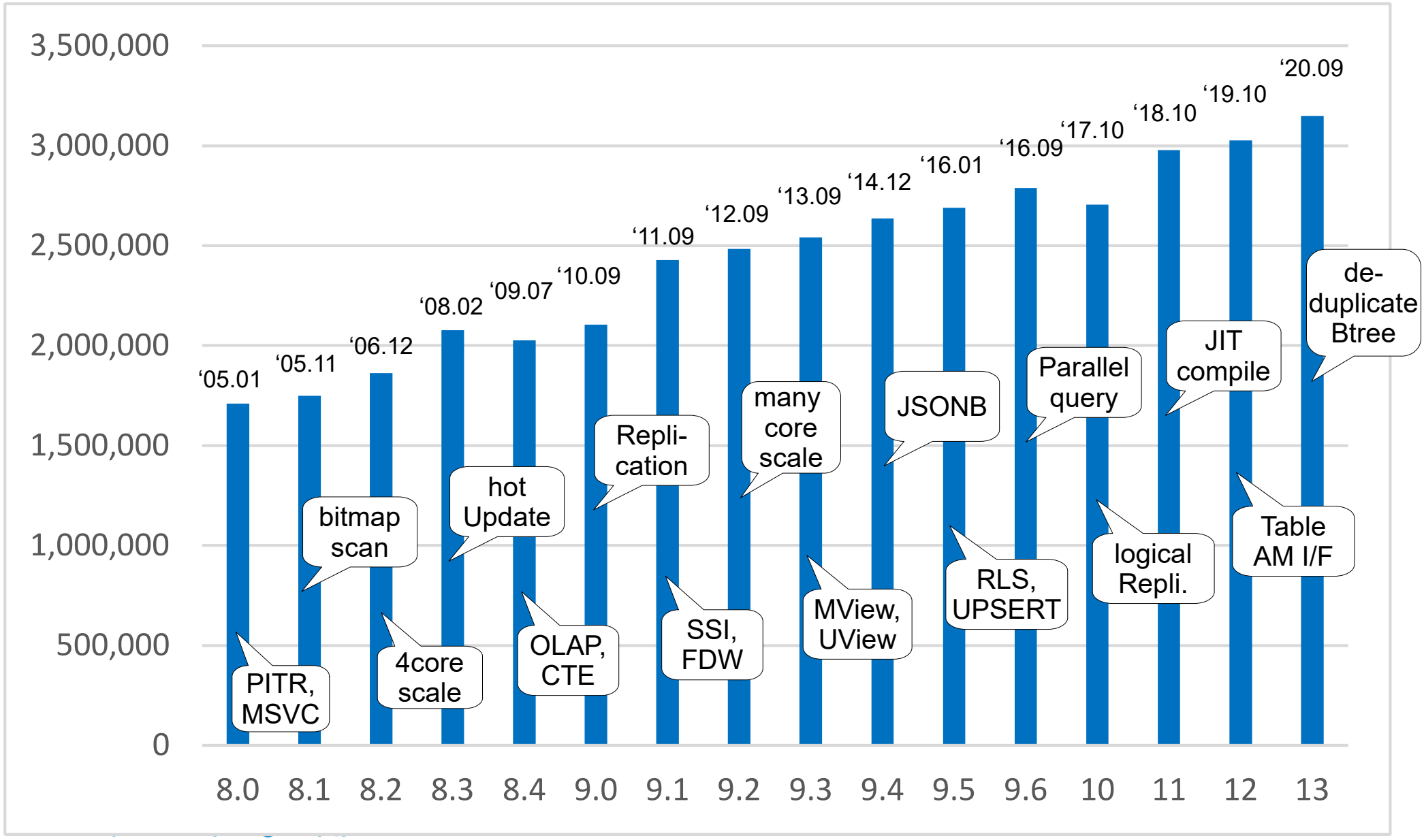
PostgreSQL とは

- 多機能、高性能、かつオープンソースの
  リレーショナルデータベース管理システム
  - INGRES('70),POSTGRES('80) 由来の歴史
  - BSD タイプのライセンス
  - 特定オーナー企業が無い



PostgreSQL
http://www.postgresql.jp

PostgreSQL リリース



最近の大きなエンハンスメント

- JIT コンパイル（11）
- Table Access Method インタフェース (12)
- Btree インデックスの重複除去 (13)
- パラレルクエリ対応と強化（ 9.6 - 13)
- テーブルパーティショニング対応と強化（10 - 13)
- ロジカルレプリケーション対応（10)

バージョン 14 の展望

- 例年の流れ：5 月ごろ仕様確定 → 秋ごろリリース
  - CommitFest 2020-07, 2020-09, 2020-11, 2021-01, 2021-03
- 各種拡張
  - マテリアライズドビュー差分更新
  - テンポラルテーブル
  - Index Skip Scan （ Loose Index Scan)
  - スキーマ変数（パッケージ変数相当）
  - グローバル一時テーブル
  - MERGE 文 ／ **SEARCH 、 CYCLE 句** ／ 標準 SQL 関数構文
  - パラレル INSERT を含む並列化対応強化
  - WAL 無効化オプション ／ 不揮発 WAL バッファ（不揮発メモリ用）
  - その他にも改善多数

⇒ **これらの大部分は未確定であることに注意**

追加候補新機能（1）

- マテリアライズドビュー差分更新
  - 元テーブルの更新時にマテビューを差分更新（即座に反映）
- テンポラルテーブル
  - 行データの変更を全て記録
  - 過去時点のデータを取り出すことができる
- Index Skip Scan
  - SELECT DISTINCT や GROUP BY の最適化
  - Btree インデックスで異なる値を調べる
- スキーマ変数

⇒ Oracle フラッシュバッククエリや、SQL Server の同名機能に相当

⇒ MySQL のルースインデックススキャン、Oracle 同名機能に相当

⇒ SQL Server 、MySQL のセッション変数、Oracle PL/SQL のパッケージ変数に相当

追加候補新機能 (2)

- グローバル一時テーブル
  - 他のセッションからも読み書きできる一時テーブル
- MERGE 文 ⇒ Oracle の同名機能に相当
- SEARCH / CYCLE 句
  - CTE 再帰検索のオプション、グラフ検索を実現
- SQL 標準に沿った SQL 関数定義

```
CREATE PROCEDURE insert_data(a integer, b integer)
  LANGUAGE SQL
  BEGIN ATOMIC
    INSERT INTO tbl VALUES (a);
    INSERT INTO tbl VALUES (b);
  END;
```

現在の PostgreSQL と周辺

SQL 機能： • SQL:2016 の大部分に対応 • 各種のストアド言語 • 地理情報対応 (PostGIS) • JSON 対応 • 豊富な拡張インタフェース	クラスタ構成： • Streaming Replication • Logical Replication • HA クラスタ（active/standby) • MPP クラスタ (shared nothing) • RAC 型 (shared disk) は不可 ×
性能： • 参照更新で多コア性能スケール（ベンチマークベース） • パーティション／パラレル対応 • Just In Compile 対応 • インメモリ対応は無し ×	運用： • 運用監視ツール pg_statsinfo / pg_badger / pg_monz • クライアントツール PgAdmin4 / SI Object Browser • 各種クラウド、 k8s 対応

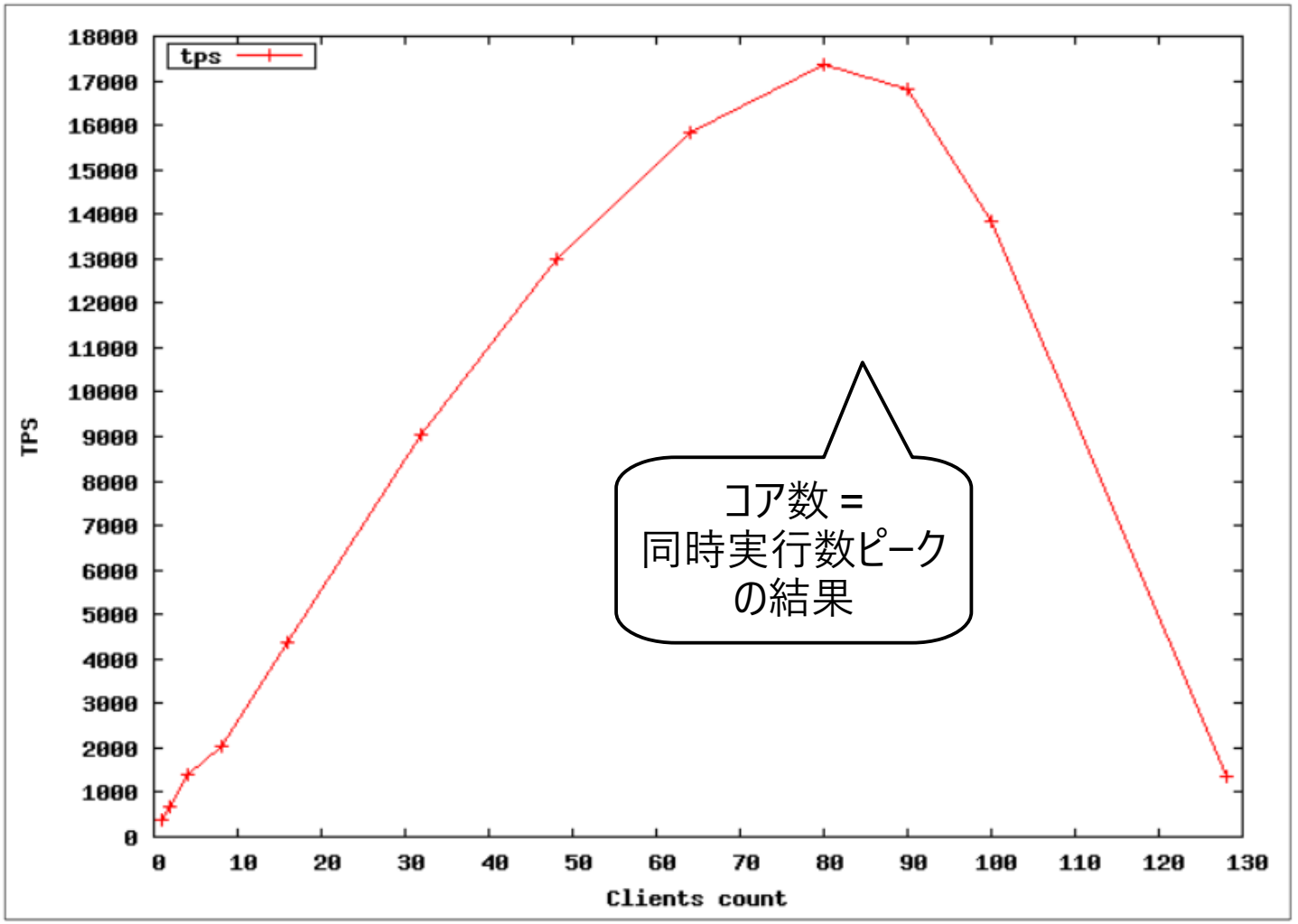
SQL 機能面でのアドバンテージ

- ストアド言語
  - PL/pgSQL
  - Perl 、 Python 、 Tcl
  - V8 (JavaScript)
- JSON 対応
  - JSON の内部要素にインデックス
  - JSON Path 関数
- PostGIS
  - OSS 拡張
  - 地理情報のデファクトスタンダード

現在の PostgreSQL と周辺

SQL 機能： • SQL:2016 の大部分に対応 • 各種のストアド言語 • 地理情報対応 (PostGIS) • JSON 対応 • 豊富な拡張インタフェース	クラスタ構成： • Streaming Replication • Logical Replication • HA クラスタ（active/standby) • MPP クラスタ (shared nothing) • RAC 型 (shared disk) は不可 ×
性能： • 参照更新で多コア性能スケール（ベンチマークベース） • パーティション／パラレル対応 • Just In Compile 対応 • インメモリ対応は無し ×	運用： • 運用監視ツール pg_statsinfo / pg_badger / pg_monz • クライアントツール PgAdmin4 / SI Object Browser • 各種クラウド、 k8s 対応

CPU スケール（参照系）

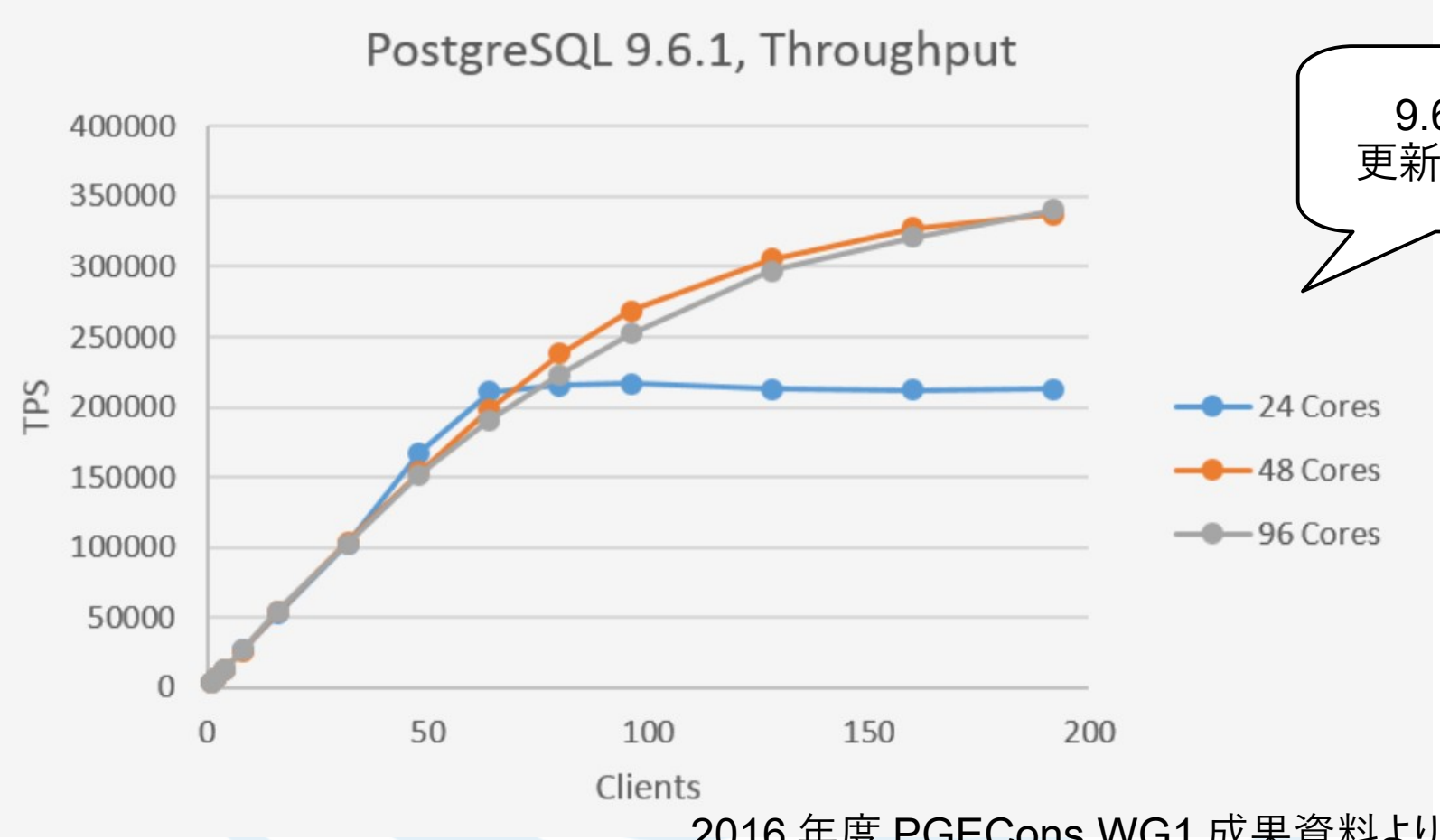


9.2.x の段階で
参照はスケール

図 2.1: 負荷の高い参照系スクリプトでクライアント数を変動させたときの結果 tps

2012 年度 PGECons WG1 成果資料より

CPU スケール（更新系）



9.6.x で
更新スケール

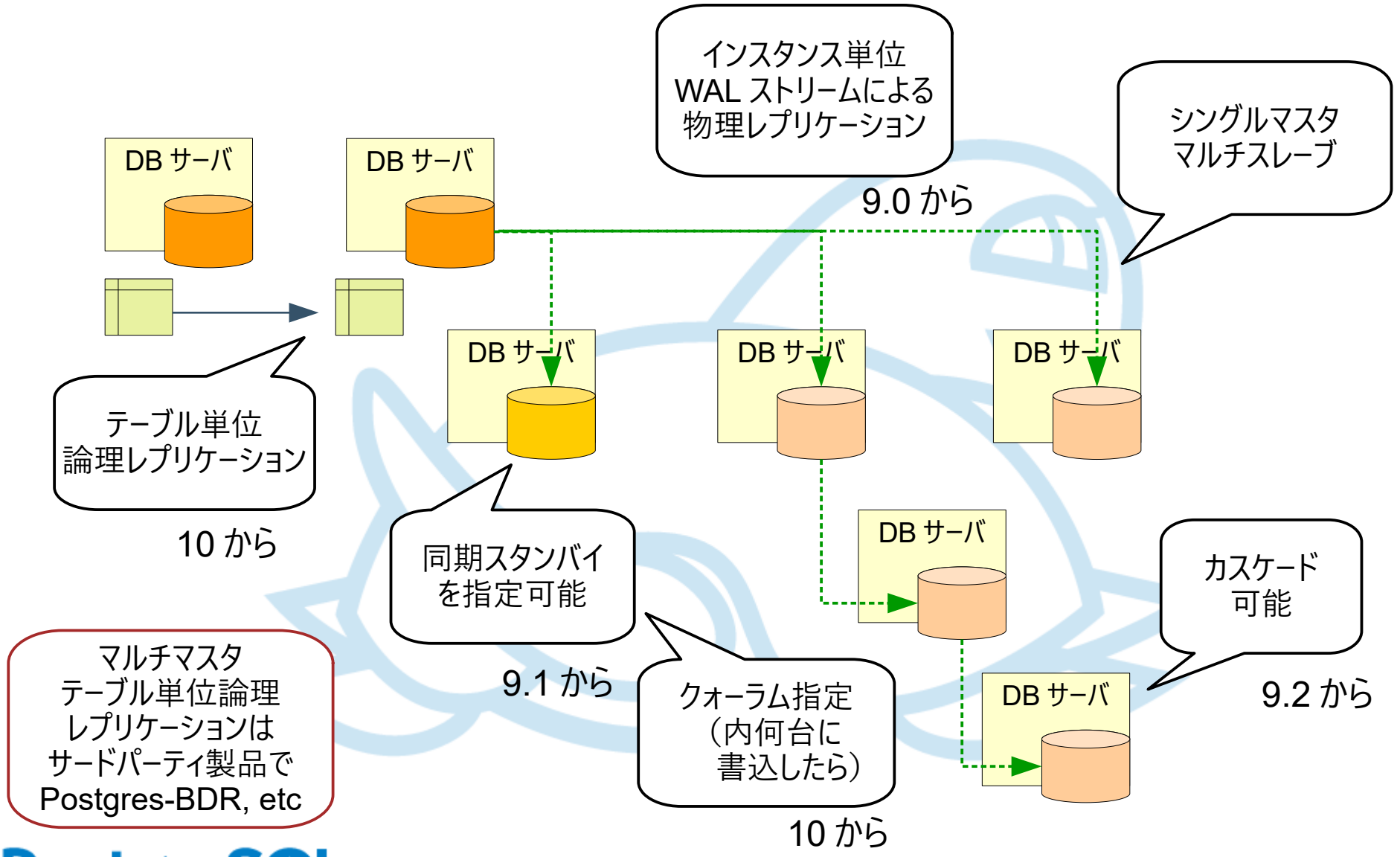
2016 年度 PGECons WG1 成果資料より

あくまでベンチマーク結果であることに注意！

現在の PostgreSQL と周辺

SQL 機能： • SQL:2016 の大部分に対応 • 各種のストアド言語 • 地理情報対応 (PostGIS) • JSON 対応 • 豊富な拡張インタフェース	クラスタ構成： • Streaming Replication • Logical Replication • HA クラスタ（active/standby) • MPP クラスタ (shared nothing) • RAC 型 (shared disk) は不可 ×
性能： • 参照更新で多コア性能スケール（ベンチマークベース） • パーティション／パラレル対応 • Just In Compile 対応 • インメモリ対応は無し ×	運用： • 運用監視ツール pg_statsinfo / pg_badger / pg_monz • クライアントツール PgAdmin4 / SI Object Browser • 各種クラウド、 k8s 対応

PostgreSQL のレプリケーション



PostgreSQL
http://www.postgresql.jp

PostgreSQL クラスタ構成

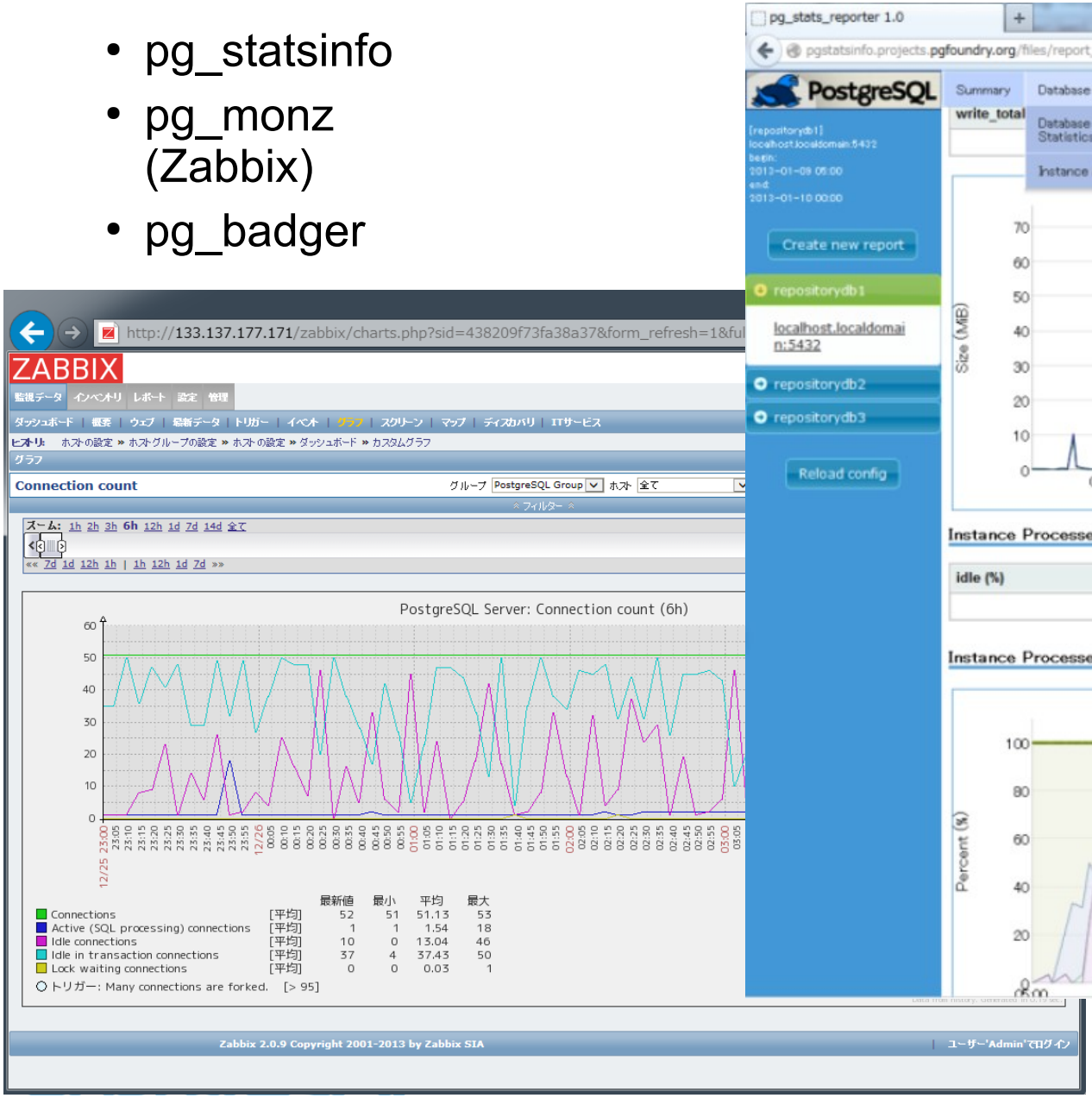
- HA クラスタ
  - Pacemaker 等の各種 HA クラスタソフトで対応
- MPP クラスタ（シャーディング - データ分割格納）
  - Greenplum
  - Citus (Azure Hyperscale)
  - Postgres-XL
- k8s オペレータ
- Pgpool-II

現在の PostgreSQL と周辺

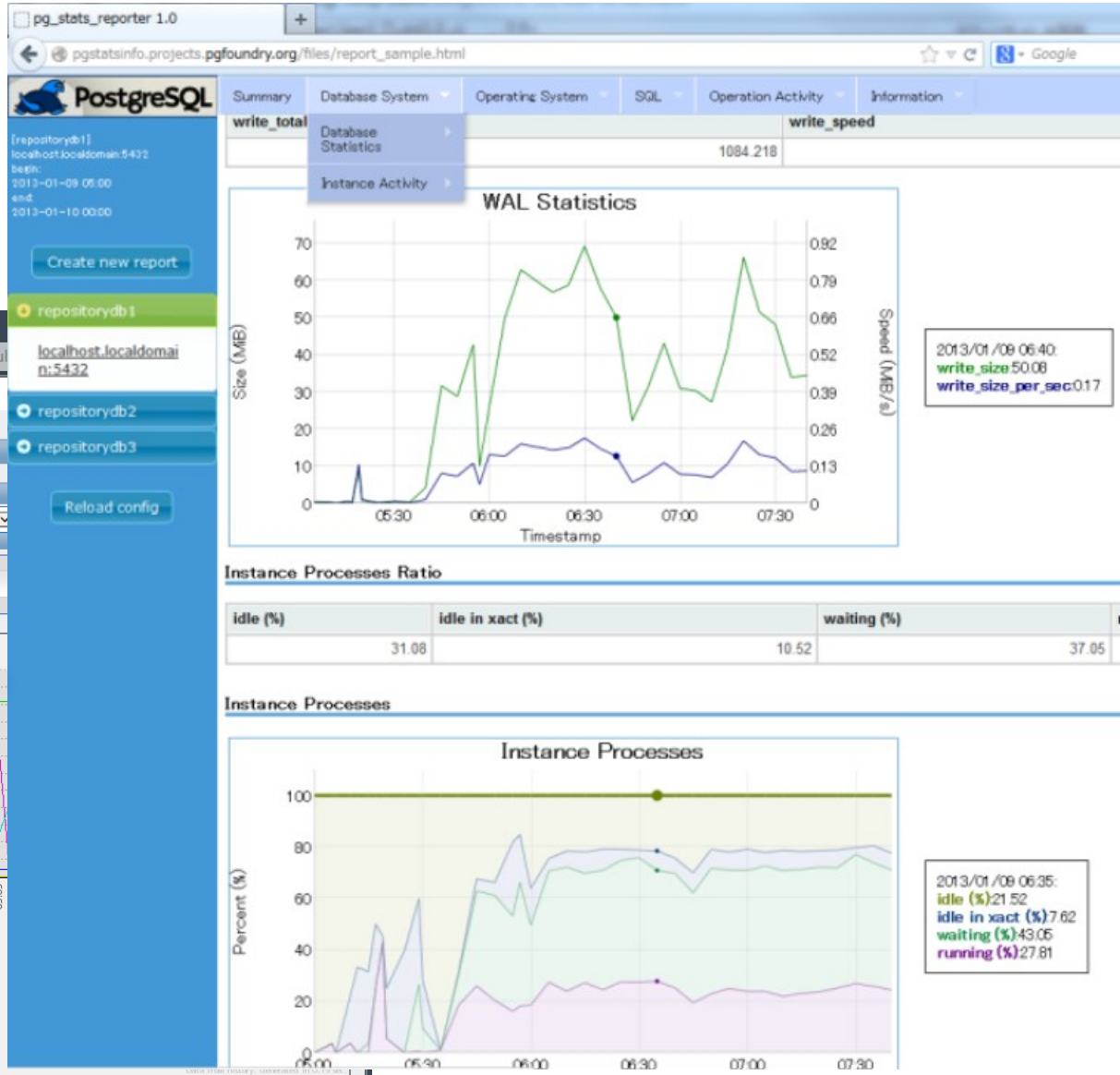
SQL 機能： • SQL:2016 の大部分に対応 • 各種のストアド言語 • 地理情報対応 (PostGIS) • JSON 対応 • 豊富な拡張インタフェース	クラスタ構成： • Streaming Replication • Logical Replication • HA クラスタ（active/standby) • MPP クラスタ (shared nothing) • RAC 型 (shared disk) は不可 ×
性能： • 参照更新で多コア性能スケール（ベンチマークベース） • パーティション／パラレル対応 • Just In Compile 対応 • インメモリ対応は無し ×	運用： • 運用監視ツール pg_statsinfo / pg_badger / pg_monz • クライアントツール PgAdmin4 / SI Object Browser • 各種クラウド、 k8s 対応

PostgreSQL の運用監視

- pg_statsinfo
- pg_monz (Zabbix)
- pg_badger

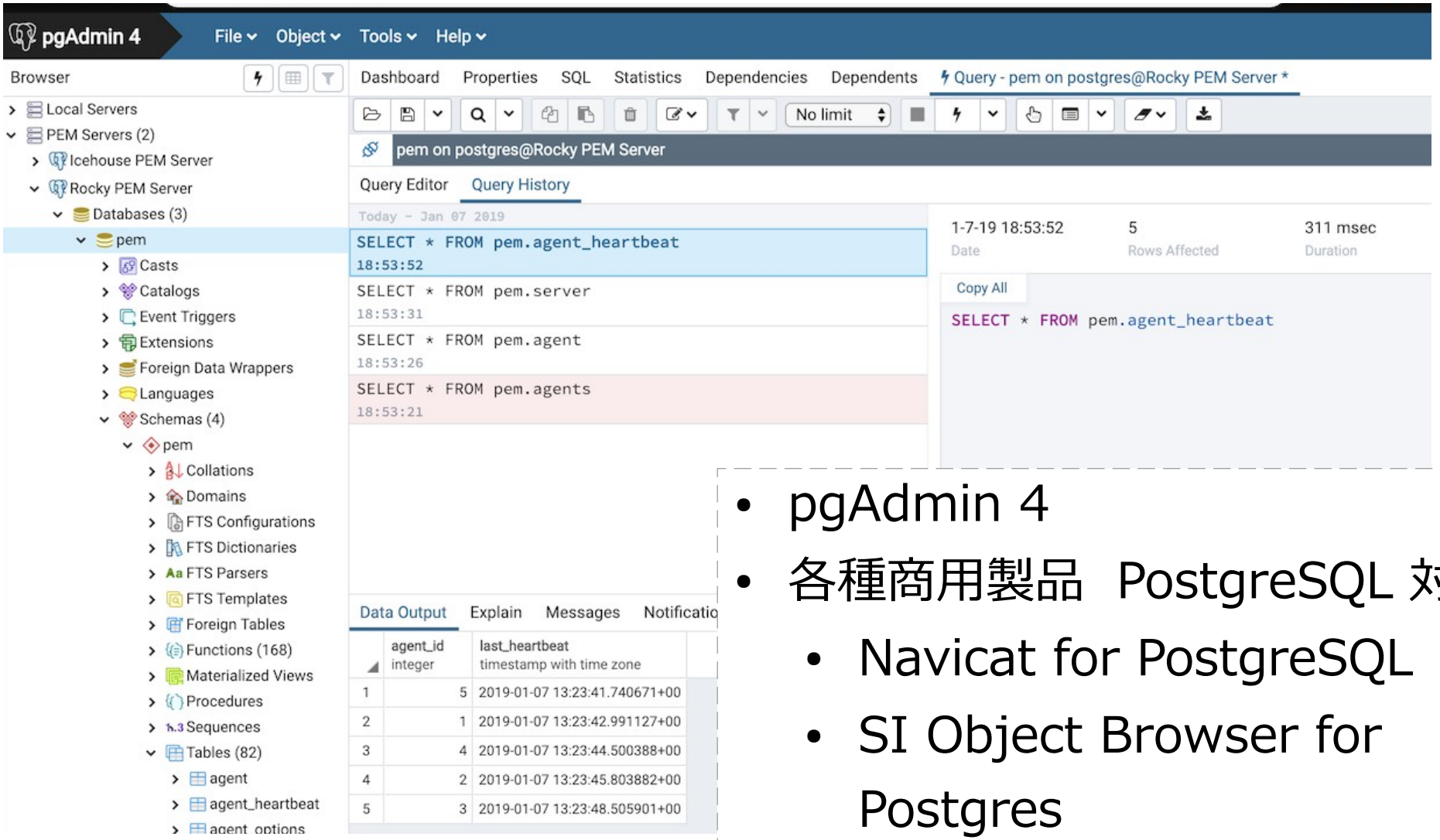


https://www.sraoss.co.jp/technology/zabbix/introduction/pg_monz.php より



https://www.postgresql.jp/sites/default/files/2017-01/B1_PGCON_JP_kondo_nttoss.pdf より

PostgreSQL のクライアントツール



https://www.pgadmin.org/screenshots より

- pgAdmin 4
- 各種商用製品　PostgreSQL 対応
  - Navicat for PostgreSQL
  - SI Object Browser for Postgres
- A5:SQL Mk-2

PostgreSQL クラウド / コンテナ

- クラウドサービス
  - Azure Database
  - AWS RDS / Aurora
  - GCP Cloud SQL
- K8s オペレータ
  - KubeDB
  - CrunchyData/ postgres-operator
  - Zalando/postgres-operator

PostgreSQL で困難なケース

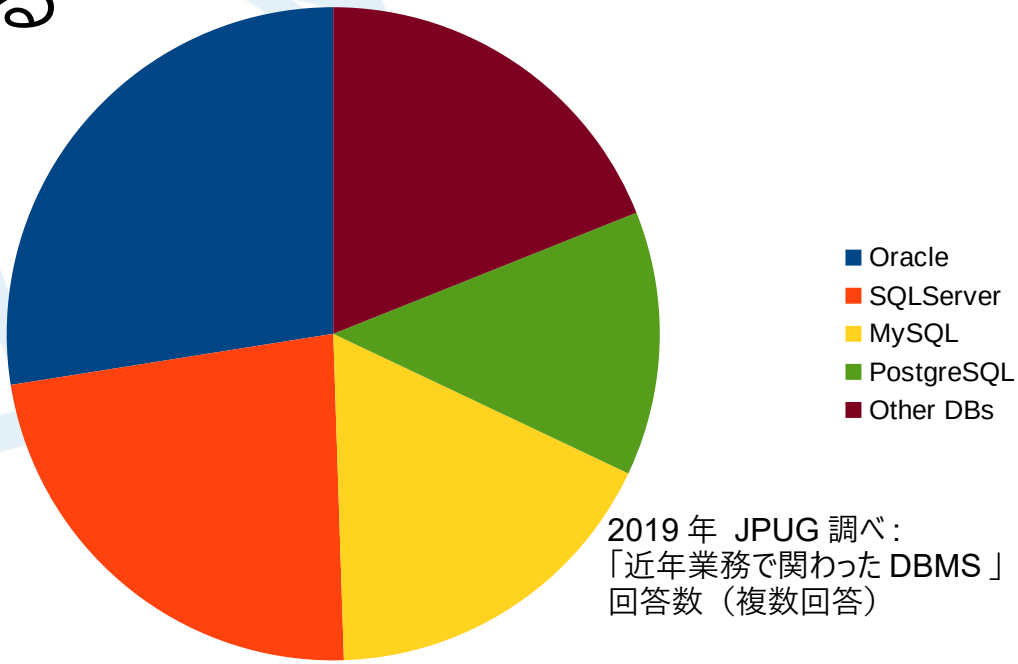
- データ投入性能の限界
  - IOT 方面／投入量要件と構成によっては専用製品に
    - WAL に直列的に書く設計であるため
- OLTP 性能の限界
  - 大メモリや多 CPU コアを活かしきれない場合
    - 遅いストレージ格納を前提とした基本設計に起因
  - スケールアウトが難しい
- HA クラスタにおける高度要件
  - 障害復旧時間の最小化要件
    - 15 秒以内、など

コミュニティと商用サポート

- 開発コミュニティ (Mailing List, Git, Slack)
  (PostgreSQL Global Development Group)
- 国内のコミュニティ活動
  - 日本 PostgreSQL ユーザ会 (JPUG)
  - PostgreSQL エンタープライズ・コンソーシアム
    （ PGECons ）
- 国内の商用サポート
  - 提供会社いくつかあり、取り扱い会社多数
    https://www.pgecons.org/postgresql-info/services/

ユーザ動向

- 「より大規模」「よりクリティカル」は一巡
  - 世界中どこで使われていても珍しくない
- Oracle Database からの移行、最終組がスタート
  - 「 RAC 同様」の壁は残る
- JPUG による利用調査
  - PostgreSQL は
    ４大人気 DBMS の一角



2019 年 JPUG 調べ :
「近年業務で関わった DBMS 」
回答数（複数回答）

JPUG の活動

- イベント
  - PostgreSQL アンカンファレンス - 最近はオンラインで毎月
  - PostgreSQL カンファレンス
  - PostgreSQL 勉強会 - 各地支部でそれぞれ年１回～数回
  - 合宿 （年 0 ～ 1 回程度）
  - OSC など各種イベントに出展（通年）
- 文書翻訳　https://github.com/pgsql-jp/jpug-doc
- ML 運営　pgsql-jp@postgresql.jp, jpug-users@postgresql.jp
  - 最近は Slack postgresql-jp が人口増加中
- Web 運営（ www.postgresql.jp 、 lets.postgresql.jp ）