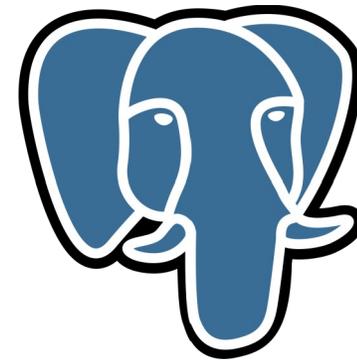


オープンソースカンファレンス2020大阪 PostgreSQL“超”入門

SRA OSS, Inc. 日本支社

OSS事業本部 技術部 PostgreSQL技術グループ

北山 貴広



目次

- 自己紹介
- PostgreSQLの概要
- PostgreSQLへのマイグレーション概要

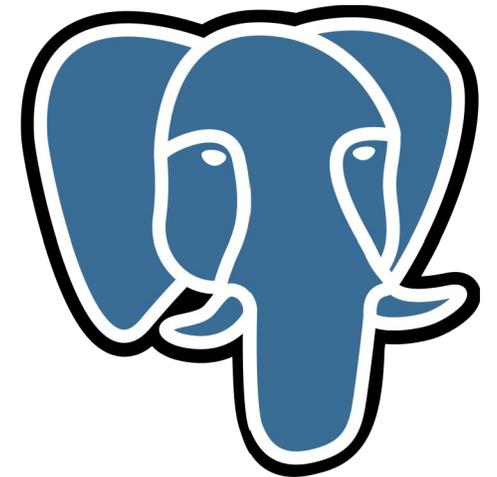
自己紹介

- 北山 貴広 (きたやま たかひろ)
- 2019年6月入社の新入社員（社会人経験31年目）
- PostgreSQLとZabbix他種々のOSSサポート業務がメイン
- データベーススキルを武器に金融公共系でシステム開発運用経験
- 過去にSoftware Designでコラム「温故知新 ITむかしばなし」を連載

PostgreSQLの概要

PostgreSQLとは

- ポストgresキューエルと読む
 - 単にポストgres、日本ではポストグレとも呼ばれる
- 標準SQLの大部分とそのほかの先進的な機能をサポートする本格的なオープンソースRDBMS
- 20年以上の歴史をもち、現在も活発な開発体制
- 豊富なプラットフォームに対応
 - Unix系OS全般、Windows、macOS
- 豊富なサポート言語
 - C、ECPG、C++、Java、Tcl/TK、Python、Perl、PHP、Ruby、ODBC、.NET Data Providerなど



PostgreSQLのライセンス

- The PostgreSQL License
 - <https://www.postgresql.org/about/license/>
- 広告条項が無い修正BSDライセンスに類似
 - 使用、複製、改変、配布の自由
 - 複製においては著作権表示、ライセンス条文、免責事項を含めることが条件
- GPLと比較すると、派生物を再配布する際にソースコード公開、ライセンス継承の義務がない

COPYRIGHTファイル

PostgreSQL Database Management System
(formerly known as Postgres, then as Postgres95)

Portions Copyright (c) 1996-2019, PostgreSQL Global Development Group

Portions Copyright (c) 1994, The Regents of the University of California

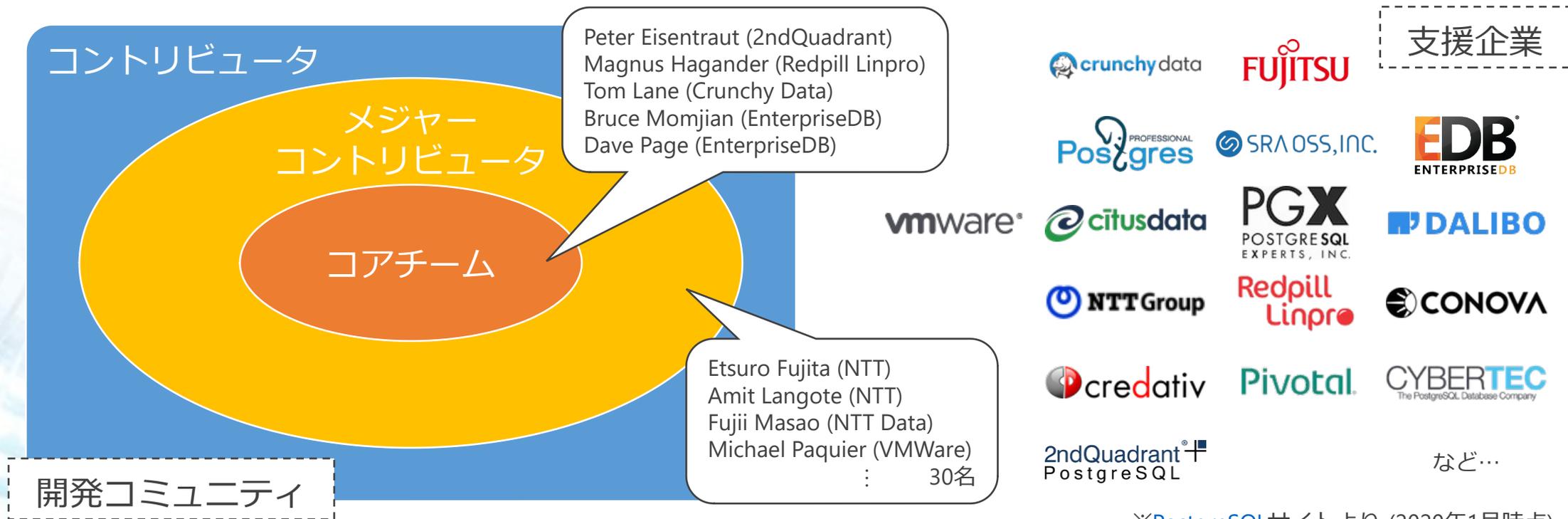
Permission to use, copy, modify, and distribute this software and its documentation for any purpose, without fee, and without a written agreement is hereby granted, provided that the above copyright notice and this paragraph and the following two paragraphs appear in all copies.

IN NO EVENT SHALL THE UNIVERSITY OF CALIFORNIA BE LIABLE TO ANY PARTY FOR DIRECT, INDIRECT, SPECIAL, INCIDENTAL, OR CONSEQUENTIAL DAMAGES, INCLUDING LOST PROFITS, ARISING OUT OF THE USE OF THIS SOFTWARE AND ITS DOCUMENTATION, EVEN IF THE UNIVERSITY OF CALIFORNIA HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

THE UNIVERSITY OF CALIFORNIA SPECIFICALLY DISCLAIMS ANY WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE SOFTWARE PROVIDED HEREUNDER IS ON AN "AS IS" BASIS, AND THE UNIVERSITY OF CALIFORNIA HAS NO OBLIGATIONS TO PROVIDE MAINTENANCE, SUPPORT, UPDATES, ENHANCEMENTS, OR MODIFICATIONS.

PostgreSQLの開発体制

- 特定のオーナー企業をもたない方式
 - 単独の企業に独占されることなく、幅広い層の厚い開発体制



※[PostgreSQL](https://www.postgresql.org/)サイトより (2020年1月時点)

日本のPostgreSQLコミュニティ

ユーザコミュニティ

日本PostgreSQLユーザ会 (JPUG)

- 1999年設立、2006年NPO法人化
- 5,000人強のML会員
- 全国各地の支部
- 普及活動を中心に

日本におけるOSSコミュニティのロールモデルでもある



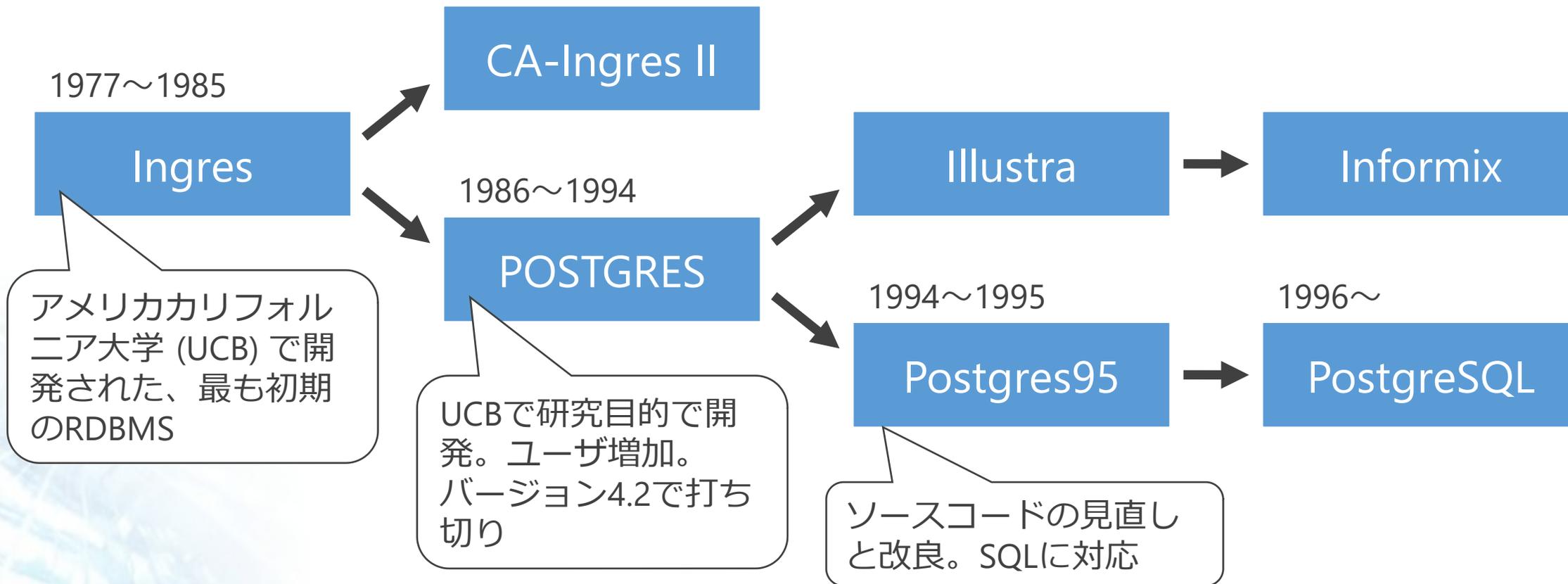
ビジネスコミュニティ

PostgreSQLエンタープライズコンソーシアム (PGECons)

- 2012年設立
- 正会員企業19社、一般会員企業45社 (2020年1月現在)
- エンタープライズ用途に向けた共同検証やプロモーション活動



PostgreSQLの生い立ち



PostgreSQLのバージョン番号

- PostgreSQLのバージョンは6.0から始まる
 - POSTGRESがバージョン4.2まで、Postgres95をバージョン5と見なして

バージョン9.6以前

9.6.14

↑ ↑
メジャー マイナー
バージョン バージョン

バージョン10以降

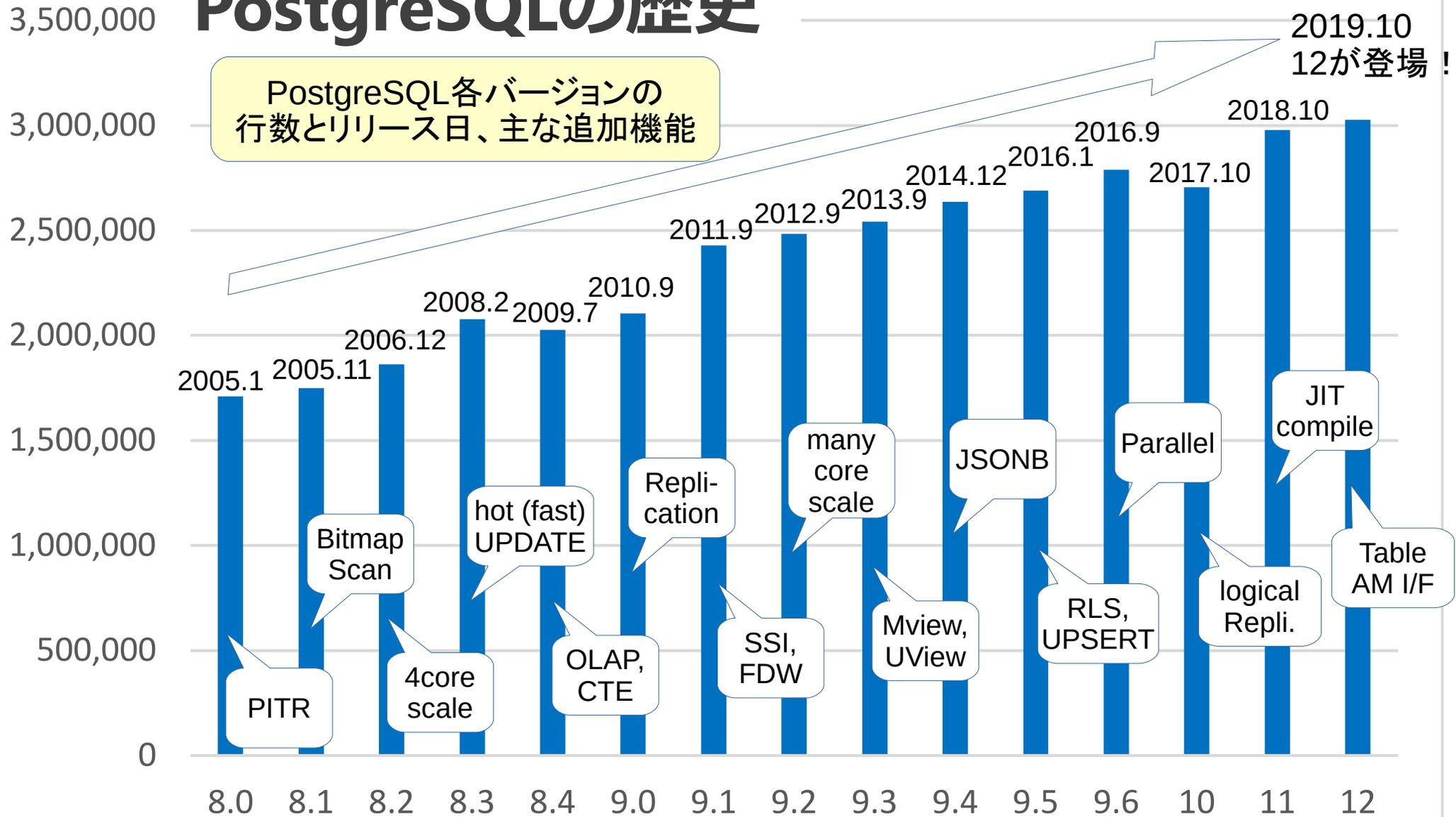
12.1

↑ ↑
メジャー マイナー
バージョン バージョン

- 1年に1度のメジャーバージョンアップでは仕様の追加・変更
 - バージョンアップにはデータ移行が必要
- 1年に数回のマイナーバージョンアップではおもにバグ修正とセキュリティ対応

PostgreSQLの歴史

PostgreSQL各バージョンの
行数とリリース日、主な追加機能



PostgreSQLで出来ること／出来ないこと

• 性能面

- 多数のCPUコアによるOLTP読み書き性能スケール（単純ベンチマーク条件）
- パラレル問い合わせ／JITに対応（OLAPむけに）
- [×]更新でWALボトルネック有

• 運用機能

- PITRバックアップ
- 監査ログ
- クラウドによるマネージドDB
- [×]フラッシュバック／時系列

• 機能面

- SQL SQL:2011 機能をほぼサポート
 - JSON対応での優越
 - PostGIS による GIS対応
 - [×]マテビュー差分更新など
- ## • クラスタ構成
- 論理・物理のレプリケーション
 - アクティブ＝スタンバイ構成
 - 参照負荷分散／シャーディング
 - 両方向論理レプリケーション
 - [×]共有ストレージを用いたアクティブ＝アクティブ構成（いわゆるRAC型）

PostgreSQLの導入事例

国内の導入事例

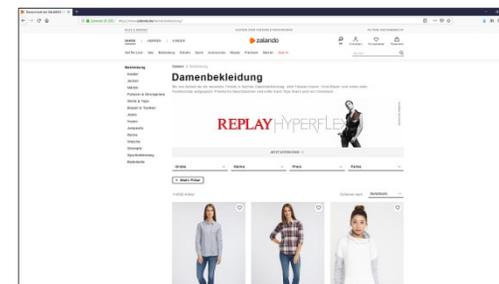
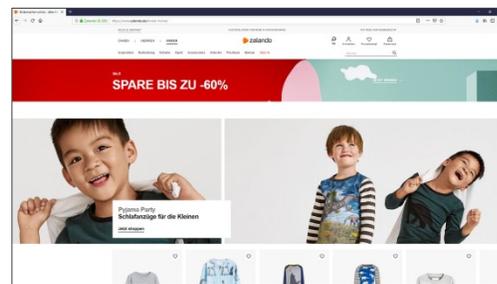
- NTT OSSセンター
 - PostgreSQLの開発まで手掛けるユーザ企業
 - 社内の80%強のシステムにPostgreSQLが採用できると分析
 - システムリプレイスに応じて順次入れ替え
- 中国電力
 - スマートメータのシステムに採用
 - PGConf.ASIA 2016で講演



PGConf.ASIA 2016公開資料: <http://www.pgconf.asia/JP/material/>

海外の導入事例

- CAIXA (ブラジル連邦貯蓄銀行)
 - ATMおよびオンラインシステムで採用
 - 開発者も抱える
- フランス社会保障システム
 - 汎用機のシステムからリプレイス
- Zalando
 - ヨーロッパ最大のオンライン服飾販売サイト
 - 5TB、90台のデータベース



Zalandoのオンライン通販サイト: <https://www.zalando.de/>

PostgreSQLの派生製品

- ライセンス制約が少ないため、商用/OSS製品のベースとして広く利用されている
 - PostgreSQLサイトに記載のものだけで42 (開発終了も含む)
- おもな派生製品
 - SRA OSSのPowerGres
 - EnterpriseDBのEDB Postgres Advanced Server
 - AWSのAmazon Aurora / Redshift
 - 富士通のFUJITSU Enterprise Postgres
 - Microsoft (Citus Data)のCitus



PostgreSQL derived databases

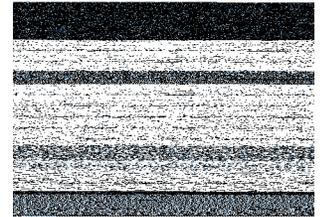
A list of PostgreSQL derived forks and rebranded distributions in alphabetical order.

Name	Vendor	License	Availability	Notes	
AgaveGraph	Blaine	Apache2	2016...	PostgreSQL + Graph Model features (Support graph storage and Cypher query language)	
Adia Data	Teclat	Proprietary	2006...	PostgreSQL + MapReduce	
ADMP	Softchase	BSD	2014...	PostgreSQL Mark Mayer combined entry back to Open PG	
Adigos	Swinsium	BSD	2005-2007	PostgreSQL + BI features	
Adip	Clas Data	AGPL v3	2012...	Sharding and replication across many series with MRP [1] [2]	
Adipulator	Cyberbit	BSD	2007-2010	Clustering (pgCluster fork)	
Adipulator Database	Genpact	Apache2	2005...	PostgreSQL + BI features (formerly known as "Bigres MRP") [1] [2]	
Adipulator	Genpact	Proprietary	2003-2007	PostgreSQL + BI features [1] [2]	
Adipulator Enterprise Postgres	Fujitsu	proprietary	2006...	Full PostgreSQL, compatibility with additional functionality [1] [2]	
Adipulator	MTI DATA	Proprietary	2012...	Database appliance solution based on PostgreSQL [1] [2]	
Adipulator	EnterpriseDB	GPL	2007-2010	PostgreSQL + BI features (formerly known as "EDB Postgres") [1] [2]	
Adipulator	Great Bridge PostgreSQL	Great Bridge LLC	BSD	1999-2001	PostgreSQL re-distribution
Adipulator	Yale University	Apache License V2	2006...	PostgreSQL + sharednothing cluster + Hadoop [1] [2]	
Adipulator	Teclat	Proprietary	2011...	Hardware fork	
Adipulator	Command Prompt	BSD	2005-2010	PostgreSQL + proprietary replication + extensions	
Adipulator	IBM	proprietary	2005...	Appliance based on PostgreSQL, SQL engine	
Adipulator	RedHorse	proprietary	2002-2003	Native Win32 port of PostgreSQL	
Adipulator	Astian	proprietary	2005...	PostgreSQL + BI features [1] [2]	
Adipulator	Peraso	BSD	2005-2006	PostgreSQL re-distribution	
Adipulator	SRA	BSD	2002-2005	Clustering (Share Nothing)	
Adipulator	SRA	BSD	2004-2007	Clustering (Shared Disk)	
Adipulator	pgNet OGG	BSD	2005...	Clustering (Connection Pooling / Replication / Load Balancing)	
Adipulator	Open-X	GPL v3	2014...	Streaming SQL	
Adipulator	MTI DATA	BSD	2009-2010	Clustering / Postgresql-vert is a fork of the JDBC driver, not from the backend code	
Adipulator	EnterpriseDB	proprietary	2006...	PostgreSQL + Oracle compatibility + security + performance tools + developer tools + DBA tools, formerly Postgres Plus Advanced Server / EnterpriseDB AS [1] [2]	
Adipulator	Postgres Professional	proprietary	2010...	PostgreSQL + enterprise features [1] [2]	
Adipulator	PGDG	BSD	2005-2010	Clustering	
Adipulator	PGDG	BSD	2010...	Clustering (formerly Postgres-IC)	
Adipulator	PGDG	BSD	2010-2013	Clustering [1] [2]	
Adipulator	PGDG	BSD	2014...	Clustering	
Adipulator	SRA OSS	proprietary	2003...	Native Win32 port of PostgreSQL, and Linux RPM	
Adipulator	SRA OSS	proprietary	2003...	PostgreSQL + custom storage engine, relational SQL, encrypted database [1] [2]	
Adipulator	Sun	TRIL	2004-2009	PostgreSQL re-distribution	
Adipulator	Open-X	BSD	2011...	Reimplementation engine [1] [2]	
Adipulator	Red Hat	BSD	2000-2003	PostgreSQL re-distribution	
Adipulator	Amazon	Proprietary/Commercial	2013...	Data Warehouse or AWS Spanned on Amazon Redshift [1] [2]	
Adipulator	SQL	GPL	2011-2011	PostgreSQL + BI features (fork of EnterpriseDB)	
Adipulator	UC Berkeley	BSD	2000-2008	Data Stream oriented fork of PostgreSQL	
Adipulator	BitData	AGPL v3	2015...	Turns PostgreSQL into a NoSQL database compatible with MongoDB. It can work as a standalone database or a secondary node of a MongoDB replicant	
Adipulator	Teclat	proprietary	2006-2012	Fork of PostgreSQL	

PostgreSQL derived databases
https://wiki.postgresql.org/wiki/PostgreSQL_derived_databases

クラウドでのPostgreSQL利用

- マネージドサービスも増え、利用しやすくなってきている
 - Microsoft Azure Database for PostgreSQL
 - Amazon RDS for PostgreSQL
 - Google Cloud SQL for PostgreSQL
- クラウドのメリット
 - 資材の調達が不要
 - 従量課金制なので使った分しか費用が発生しない
- マネージドサービスの利点
 - 運用管理を任せられる
 - 耐障害性、可用性構成の作成を任せられる
 - スケールアップ/ダウンが容易



RDBMS市場でのPostgreSQLの位置づけ

- DB-Engines Rankingによる人気ランキングでは
 - Oracle、MySQL、MS SQL Serverに次いで第4位のスコア
 - スコアの伸びでは2017年から2年連続で第1位
- Oracleの移行先として
 - SE1/SEの販売終了、他社クラウドでの値上げもあって、Oracle離れが進んでいる
 - MySQLに比べると、機能が豊富で移行しやすく、システム構造が似ていて技術者の転換がしやすい
- Oracle以外とは棲み分け
 - MySQLはシンプルなWebシステム向け
 - Microsoft SQL ServerはWindows向け

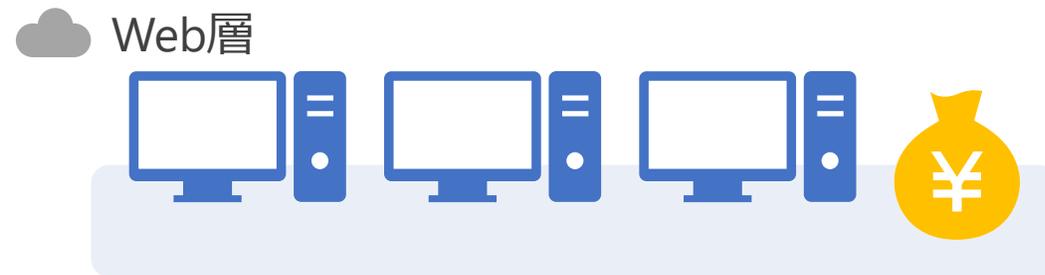
350 systems in ranking, January 2020

Rank			DBMS	Database Model	Score		
Jan 2020	Dec 2019	Jan 2019			Jan 2020	Dec 2019	Jan 2019
1.	1.	1.	Oracle +	Relational, Multi-model	1346.68	+0.29	+77.85
2.	2.	2.	MySQL +	Relational, Multi-model	1274.65	-1.01	+120.39
3.	3.	3.	Microsoft SQL Server +	Relational, Multi-model	1098.55	+2.35	+58.29
4.	4.	4.	PostgreSQL +	Relational, Multi-model	507.19	+3.82	+41.08
5.	5.	5.	MongoDB +	Document, Multi-model	426.97	+5.85	+39.78
6.	6.	6.	IBM Db2 +	Relational, Multi-model	168.70	-2.65	-11.15
7.	7.	↑8.	Elasticsearch +	Search engine, Multi-model	151.44	+1.19	+8.00
8.	8.	↓7.	Redis +	Key-value, Multi-model	148.75	+2.51	-0.27
9.	9.	9.	Microsoft Access	Relational	128.58	-0.89	-13.04
10.	↑11.	10.	SQLite +	Relational	122.14	+1.78	-4.66
11.	↓10.	11.	Cassandra +	Wide column	120.66	-0.04	-2.32
12.	12.	12.	Splunk	Search engine	88.67	-1.85	+7.25
13.	13.	13.	MariaDB +	Relational, Multi-model	87.45	+0.66	+8.63
14.	14.	↑15.	Hive +	Relational	84.24	-1.81	+14.33
15.	15.	↓14.	Teradata +	Relational, Multi-model	78.29	-0.21	+2.10
16.	16.	↑20.	Amazon DynamoDB +	Multi-model	62.02	+0.39	+6.93

※DB-Engines Rankingサイトより (2020年1月時点)

PostgreSQLへのマイグレーション概要

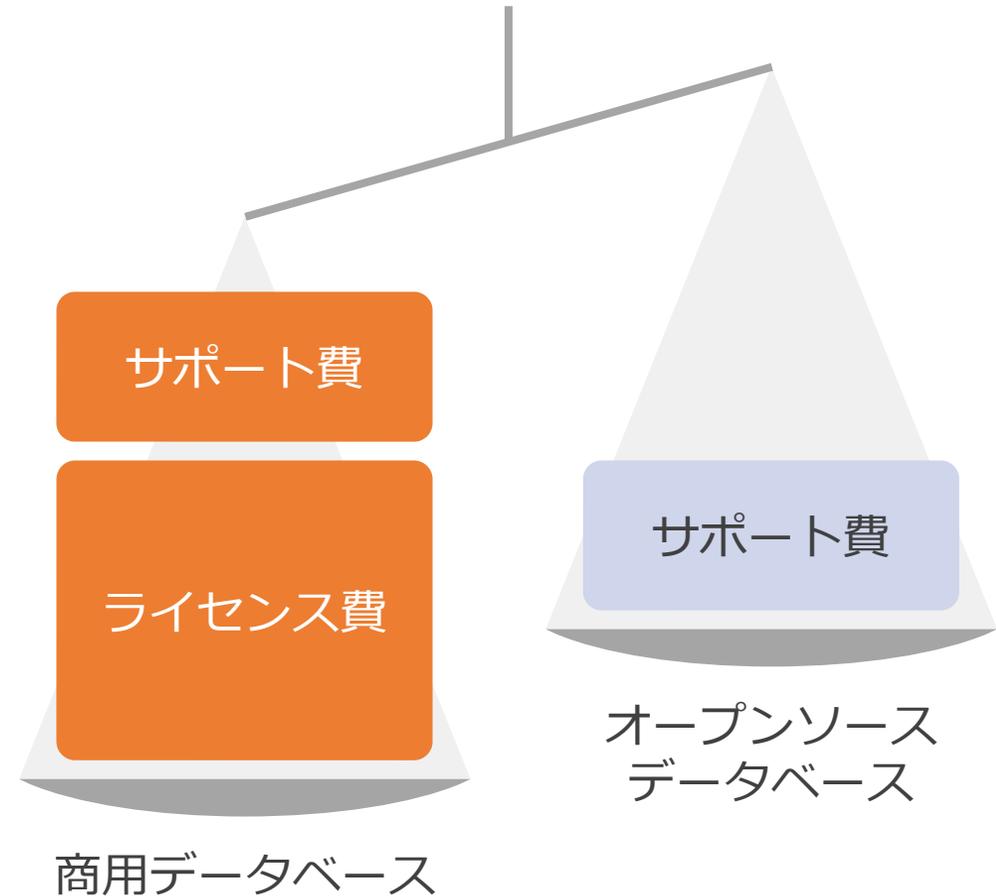
ITシステムのコスト構造



- ITシステムのコストはデータベースの選択がカギを握る
 - データベースの重要性は高い
 - データベースのコストが占める比率も高い
- データベースにオープンソースを使ってコストを削減

データベースのコスト

- 商用データベースではライセンス費、サポート費は必須
- オープンソースデータベースではライセンス費は基本的に不要、サポート費は必要に応じて選択できる
- オープンソースデータベースを使えばライセンス費を削減できる



ベンダロックインのリスク

- 特定のベンダ製品に依存するあまり、ほかの製品への乗り換えが難しくなり、ベンダの言いなりになってしまう状態
- 社内標準としてデータベースを1つに絞っていると…



メリット

- ノウハウがたまりやすい
- まとめ買いで値引きしてもらえらるかも



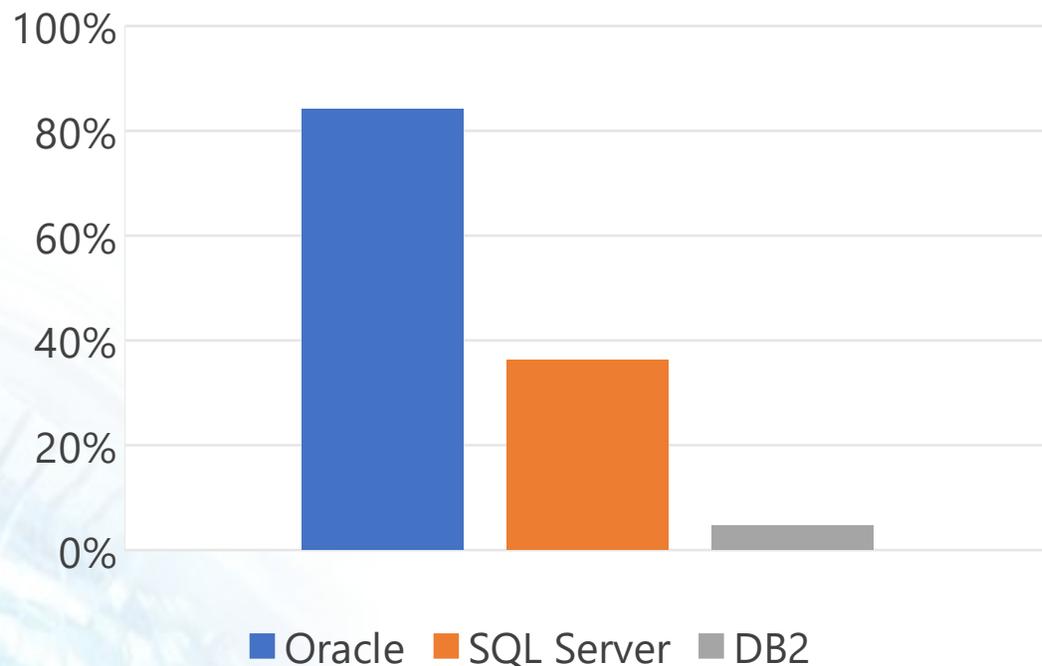
デメリット

- システムに適したデータベースがほかにあっても選べない
- ベンダの都合で不要なバージョンアップを強いられたり、突然サポートを打ち切られたりする

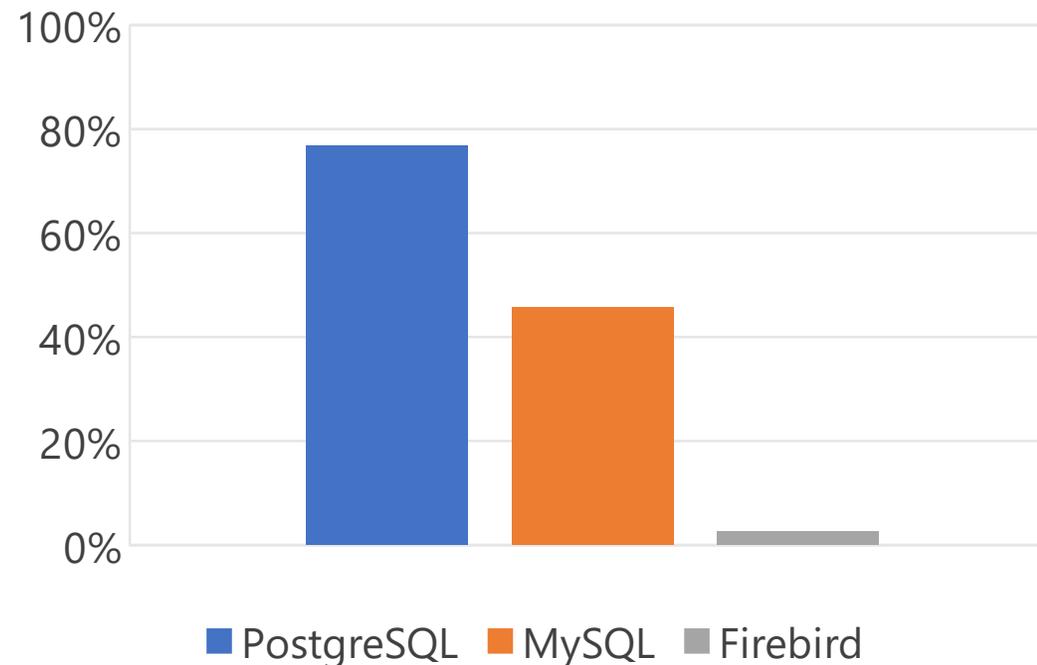
- システムに応じてオープンソースデータベースを使い分ければベンダロックインから抜け出せる

マイグレーション先に選択されるPostgreSQL

マイグレーション元の RDBMS



マイグレーション先の RDBMS



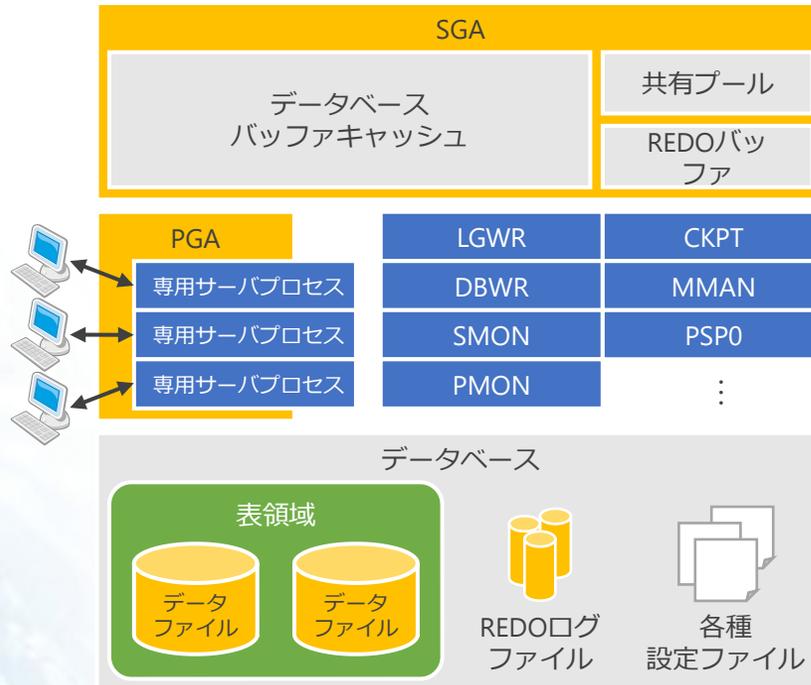
<https://lets.postgresql.jp/documents/tutorial/UserSurvey/1>

PostgreSQLが選択される理由

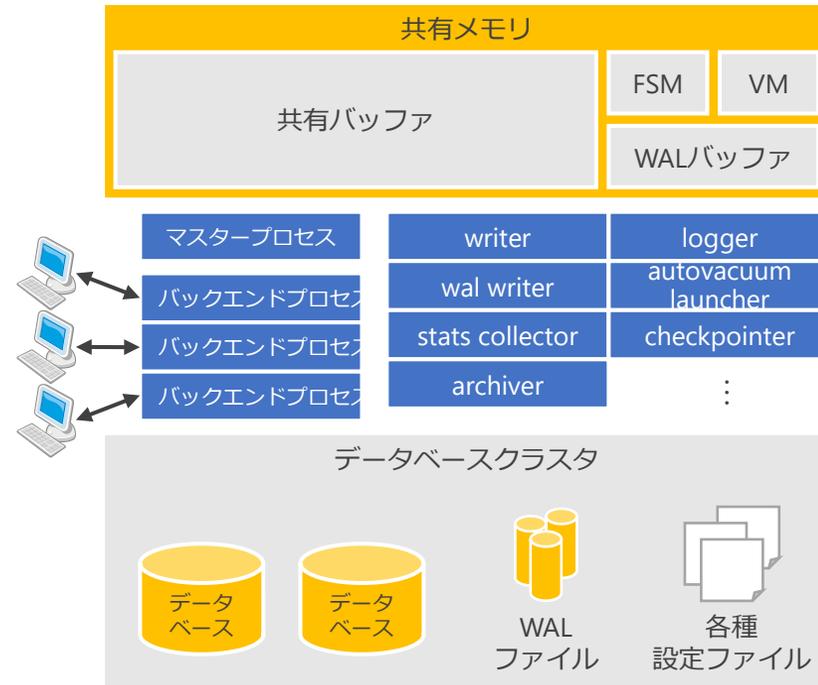
- エンジニアをOracleから転換しやすい

Oracleとは構造が
けっこう似ている

Oracleの構造

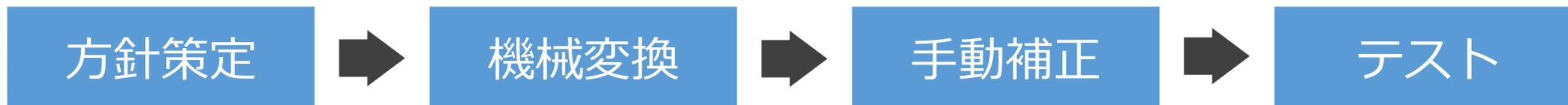


PostgreSQLの構造



Oracleからのマイグレーション

- マイグレーションにコストがかかる場合
 - ストアドプロシージャを多用している場合
 - OCI (Oracle Call Interface) など、独自APIを使用している場合
- 基本手順



- 手動補正の過程で変換方針に追加すべきことが加わっていく
- 一次手動補正 (意図の理解なし) の後に二次手動補正 (意図の理解あり) を行うか？

OracleとのおもなSQL特性の違い

- デフォルトでAUTOCOMMIT/
明示トランザクション
- DDLは暗黙にCOMMIT実行が
含まれない
- DDLはデフォルトでロックタ
イムアウトなし
- トランザクション中のエラー
後にCOMMITしても、成功し
た命令も含めてロールバック
される

```
=> UPDATE t1 SET val = 1 WHERE id = 10;  
UPDATE 1  
=> BEGIN;  
BEGIN  
=> DELETE FROM t1 WHERE id > 100;  
DELETE 9900  
=> COMMIT;  
COMMIT  
=> BEGIN;  
BEGIN  
=> DROP TABLE t1;  
DROP TABLE  
=> ROLLBACK;  
ROLLBACK
```

OracleとのおもなSQLの違い

- SELECTにFROMが不要 (DUAL)
- VARCHAR2型、NUMBER型がない (NUMERIC型がある)
- DATE型に時分秒が含まれない
- NULLと" (空文字列) に区別がある
- VARCHAR(N) 型やLENGTH関数の長さが文字数単位
- NVL関数がない
- ROWNUM列がない

```
SELECT * FROM (  
  SELECT * FROM tbl  
  ORDER BY c  
) WHERE rownum < 10;
```



```
SELECT * FROM tbl  
ORDER BY c LIMIT 10;
```

```
SELECT NVL(c, 0)  
FROM tbl;
```



```
SELECT  
CASE WHEN c IS NULL  
  THEN 0 ELSE c END  
FROM tbl;
```

利用ツール: Ora2Pg

- OracleからPostgreSQLへの移行ツール
 - <http://ora2pg.darold.net/>
- Oracleに接続して、スキーマやデータを抽出し、PostgreSQLで読み込めるように変換
 - 移行時の設定を柔軟に指定できる
 - NUMERICの桁数に応じてSMALLINT、INTEGER、BIGINTを使い分け
 - 移行できない/移行が完璧でないオブジェクトも多い
 - 移行コスト見積もりのレポートも作成できる



利用ツール: orafce

- Oracle互換の関数とパッケージ
 - <https://github.com/orafce/orafce>

算術関数

BITAND、COSH、SINH、TANH

文字列関数

INSTR、LENGTH、LENGTHB、LPAD、LTRIM、NLSSORT、RPAD、RTRIM、SUBSTR、UBSTRB

日付/時刻関数

ADD_MONTHS、DBTIMEZONE、LAST_DAY、MONTHS_BETWEEN、NEXT_DAY、ROUND、SESSIONTIMEZONE、SYSDATE、TRUNC

データ型書式設定関数

TO_CHAR、TO_DATE、TO_MULTIBYTE、TO_NUMBER、TO_SINGLE_BYTE

条件式

DECODE、LNNVL、NANVL、NVL、NVL2

パッケージ

DBMS_ALERT、DBMS_ASSERT、DBMS_OUTPUT、DBMS_PIPE、DBMS_RANDOM、DBMS_UTILITY、UTL_FILE

など

参考ドキュメント

PostgreSQLエンタープライズ・コンソーシアム成果物

- 成果物総索引: https://www.pgecons.org/works_index/

- データベース移行

- アセスメント、システム構成移行、異種DB連携、定義移行、データ移行、アプリケーション移行、移行評価、運用移行、チューニング、機能要件、非機能要件、サポートツール

PGEcons

1. 組み込みデータ型

1.1. 組み込みデータ型の違い
OracleとPostgreSQLでは対応している組み込みデータ型の違いがあり、移行元のデータベースで使われているデータ型が移行先のデータベースに存在しない場合があります。移行元のデータ型が移行先に存在しない場合は、移行先のデータベースにおいて対応するデータ型に設定する必要があります。
OracleとPostgreSQLに存在する組み込みデータ型の対応関係は、別紙「組み込みデータ型対応表(Oracle-PostgreSQL)」を参照してください。

1.2. 変換方針
1.1に記載した相違点に基づき、以下の方法で組み込みデータ型を変換します。

1.2.1. 文字型
Oracleのデータ型に対応するPostgreSQLのデータ型に変換します。
ただし、可変長文字列はサイズ指定がバイト数(文字数)のどちらかで指定されているか注意が必要です。

1.2.2. 数値型
OracleのNUMBER型に相当するデータ型はPostgreSQLに存在しません。
そのための、精度とスケールに応じて、PostgreSQLの数値データ型から変換する型を選択します。
なお、OracleのFLOAT型はPostgreSQLにも存在するため変換は不要ですが、両者が可能な精度が異なる(Oracle 2乗数 126桁、PostgreSQL 2乗数 53桁)点に注意する必要があります。

1.2.3. 日付/時刻データ型
Oracleのデータ型に対応するPostgreSQLのデータ型に変換します。
ただし、DATE型はOracleとPostgreSQLで両方とも制度が異なるため、注意が必要です。

1.2.4. バイナリ型
PostgreSQLでは別称としてbytea型に変換します。
ただし、bytea型で両方とも1GBを超える場合は、ラージオブジェクトを使用するか、外部ファイルへ別途保存する必要があります。

1.2.5. その他
対応するデータ型がPostgreSQLに存在しないため、概念的な変換は困難です。

2. テーブル

2.1. DDLの違い
OracleとPostgreSQLに併用するCREATE TABLE文の違いを比較します。
[OracleのCREATE TABLE文]
CREATE [GLOBAL TEMPORARY] TABLE 表名
(
列名 データ型 [SORT] [ENCRYPT 暗号化方式]
);

1 Oracle Database SQL 書籍リファレンス 11gリリース2
(http://docs.oracle.com/cd/E16338_01/server.112/b56299/elements_7002.htm#2095331)より引用

5/25 Copyright © 2011 PostgreSQL Enterprise Solutions. All Rights Reserved.

データ型の移行

PGEcons

2. OracleからPostgreSQLへの移行(定義関連)

2.1. CREATE FUNCTION 文

CREATE FUNCTION 文の比較

Oracle	PostgreSQL
CREATE OR REPLACE FUNCTION	CREATE OR REPLACE FUNCTION
FUNCTION名 (戻り型) IN データ型	func_f (戻り型 IN データ型)
RETURN 戻り型データ型	RETURNS 戻り型データ型
IS	AS
変数名 データ型	\$\$
BEGIN	DECLARE
; 宣言節	変数名 データ型
END;	BEGIN
	; 宣言節
	END;
	\$\$
	LANGUAGE plpgsql;

PostgreSQLでは括弧内の記述部分(変数名とBEGINからENDまで)を文字列定数として作成する必要があります。
SQL文に引号(ダブルクォーテーション)を付与して記述記述の範囲を指定します。
単一引用符で範囲を指定する方法も可能ですが、この場合には関数の本体部分で使用される単一引用符(バックslash)は二重引用符にする必要があります。
括弧内の記述に使用している変数の指定は、LANGUAGE句で指定します。
変数名前にDECLAREが必要ですので追加する必要があります。
引号を付けないFUNCTIONを作成するときはOracleでは"()"を省略できますが、PostgreSQLでは"()"の記述が必要です。
上記以外では
RETURN → RETURNS
IS → AS
に変換する必要があります。

2.2. CREATE PROCEDURE 文
PostgreSQLにはPROCEDUREは実装されていません。
FUNCTIONで代替する事になります。

2.3. CREATE PACKAGE 文
PROCEDUREと同等にPACKAGEは実装されていません。
FUNCTIONで代替することになります。
PACKAGEレベルで共通使用する定数は、一時テーブルに格納するなどの方法を検討する必要があります。

7/34 Copyright © 2011 PostgreSQL Enterprise Solutions

ストアドプロシージャの移行

オープンソースとともに



SRA OSS, INC.