

課題検討WG 2016年度活動報告

PostgreSQLデータ連携大全 - PostgreSQL間の連携から異種データベースとの連携まで -

PostgreSQL エンタープライズ・コンソーシアム WG3 (課題検討WG)

(株)アシスト 竹内/ NTTテクノクロス(株) 原田

アジェンダ

■ WG3の紹介

■レプリケーション

■ ストリーミングレプリケーション、Bi-Directional Replication (BDR)

■ 異種DB間連携

■ 外部データラッパ、ロジカルデコーディング、DB同期ツール、ETLツール

■ クロージング

WG3 (課題検討WG)の紹介

活動テーマの推移

- 2016年度の活動テーマ
 - レプリケーション
 - 異種データベース連携

昨年度までの「設計運用WG」から活動範囲を広げるため「課題検討WG」に改名した。

課題領域	概要	2013	2014	2015	2016
性能	性能評価手法、性能向上手法、チューニングなど			ツール	
可用性	高可用性クラスタ、BCP	可用性全般	SR(BCP)		レプリケー ション
保守性	保守サポート、トレーサビリティ				
運用性	バックアップ運用、監視運用	バックアップ 監視		ツール	
セキュリティ	監査、認証、アクセス制御、 暗号化		セキュリ ティ	セキュリ ティ	
互換性	データ、スキーマ、SQL ストアドプロシージャ				
接続性	他ソフトウェアとの連携				異種DB 連携

WG3 2016年度活動テーマ

■ レプリケーション

■ ストリーミングレプリケーション (SR)

PostgreSQL 9.6時点における、ストリーミングレプリケーションの各種注目機能や運用ポイントを整理

■ BDR (Bi-Directional Replication)

マルチマスタ・レプリケーションシステムであるBDRの、アーキテクチャの解説、設定例、競合解消方法、注意点について調査

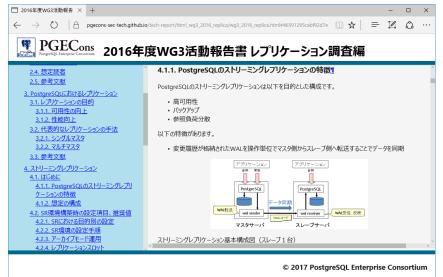
■ 異種データベース連携

データベース連携手法のうち外部データラッパ、ロジカルデコーディング、DB同期ツール、ETLツールの4種類を取り上げ比較調査

WG3 2016年度の活動成果

■「2016年度WG3活動報告書」として、 PGEConsのサイトにドキュメントを公開中







レプリケーション調査編

異種DB間連携 連携方式調査編

WG3 2016年度活動体制

- レプリケーション班 (SR および BDRを担当)
 - □ 株式会社アシスト【主査】
 - □ 株式会社オージス総研
 - □ TIS株式会社 【副査】
 - □ 日本電信電話株式会社
 - □ 株式会社富士通ソーシアルサイエンスラボラトリ
- 異種データベース連携班
 - □ NTTテクノクロス株式会社 【 副査 】
 - □ 日本電気株式会社
 - □ 株式会社日立製作所
 - □ 株式会社日立ソリューションズ
 - □ 富士通株式会社

(企業名50音順・敬略称)

レプリケーション 成果紹介

レプリケーション:活動目的

■ 設定/監視など運用に必要なノウハウを整理する。また今後も 新機能やツールが追加されるため、定点観測として続ける。

今まで

2013年と2014年にレプリケーションについて調査→現在、レプリケーションの機能追加や新ツールの開発が進行

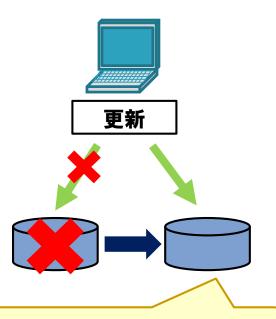
2016年度

- ストリーミングレプリケーションの既存情報整理と最新機能検証
- マルチマスタ・レプリケーションシステムのBDRの調査
 - ◆ 実例や注意点を含め、基本的な設定と運用が分かるドキュメントを作成

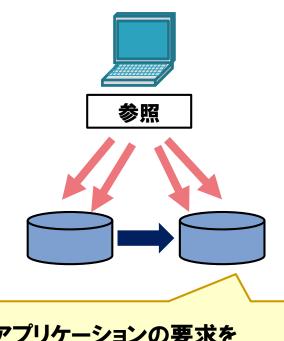
レプリケーションとは何か

複数のデータベース・サーバ間で、可用性や性能の向上を目的に 一貫性を保ちながら内容を複製する手法を指す。

冗長化による可用性向上



同一のデータを持つサーバへ 接続することでサービスを継続 スケールアウトによる性能向上



アプリケーションの要求を 複数サーバに分散し負荷軽減

レプリケーションとは何か

データの一貫性が • シングルマスタ 保証されやすい 更新 参照 同期 同期 マスタ

PostgreSQLの機能である ストリーミングレプリケーションにより実現

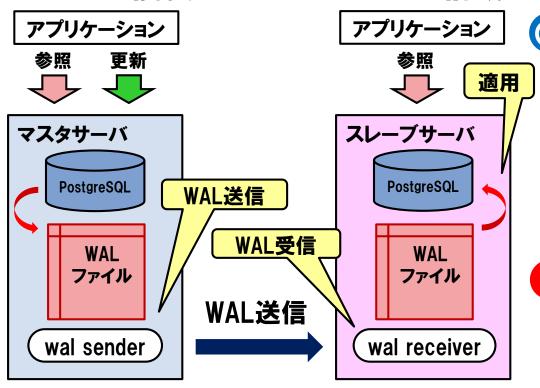
更新操作の負荷を • マルチマスタ 分散できるが、デー タの同期が困難 参照 更新 同期 同期 マスタ マスタ マスタ

BDRが論理レプリケーションを用いて実現

レプリケーション 成果紹介 - ストリーミングレプリケーション -

PostgreSQLのストリーミングレプリケーション

マスタの更新情報(WALレコード)をスレーブに送信し、 データを複製するシングルマスタ構成の1つ。



● 特徴

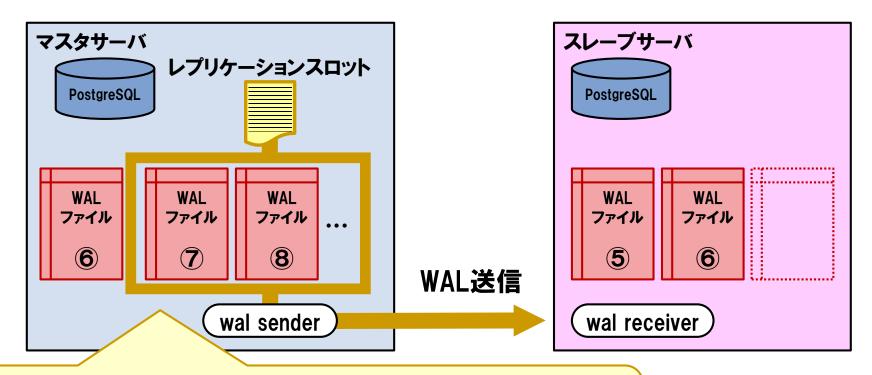
- ・受信サーバ側に送信サーバと 同一のデータベースを複製可能
- ・送信側サーバでコミット済みの データを、受信側へ確実に反映 できるため、データ保護に適切
- ・参照負荷分散が可能

1) 注意点

- ・特定の表のみの複製不可
- ・メジャーバージョンが異なる PostgreSQL間では利用不可
- ◆ 今年の活動で調査・検証した、特に注目していただきたい 3つの機能とその注意点を紹介します。

(1)レプリケーションスロットの設定

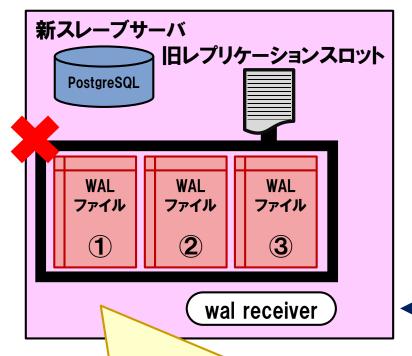
マスタ側がスレーブへ未送信のWALを保持することで、 レプリケーションの同期を保証する。



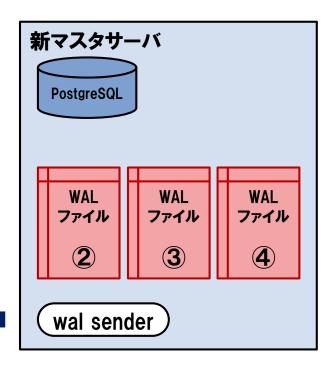
レプリケーションスロットが未送信のWALを把握している。 →スレーブに必要なWALが循環によって消失することを防止する。

(1)レプリケーションスロットの注意点

スイッチバックで正常にマスタ/スレーブを切り替えた際に、 新スレーブに旧マスタ時代のスロットが残る。



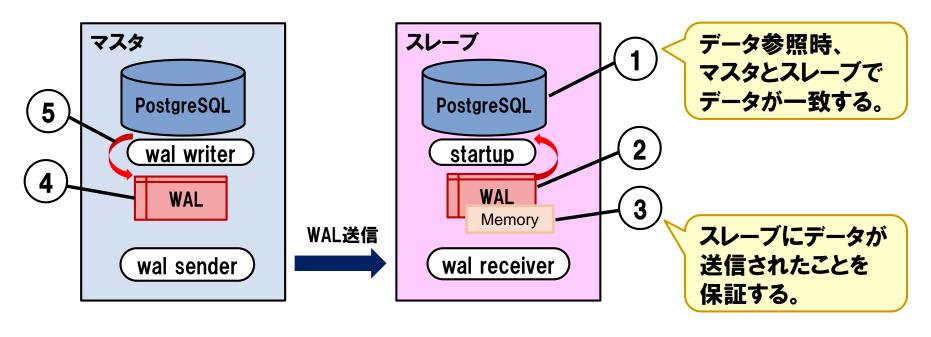
WAL送信



送信先がないため、旧レプリケーションスロットが WALを保持し続け、ディスクを圧迫する

(2) ストリーミングレプリケーションの同期レベル

synchronous_commitに目的に応じた値を指定する。



① remote_apply:スレーブ側へのデータ適用まで保証

② on : 送信したWALのディスク書き込みまで保証

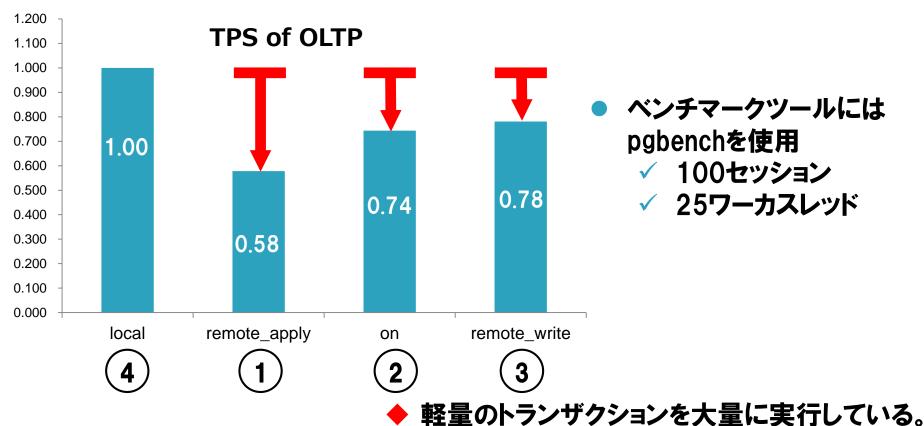
③ remote_write : 送信したWALのメモリ書き込みまで保証

④ local(非同期): ローカルへの書き込みまで保証

5 off : ローカルへの書き込みも保証しない ※非推奨設定

(2) ストリーミングレプリケーションの同期レベル

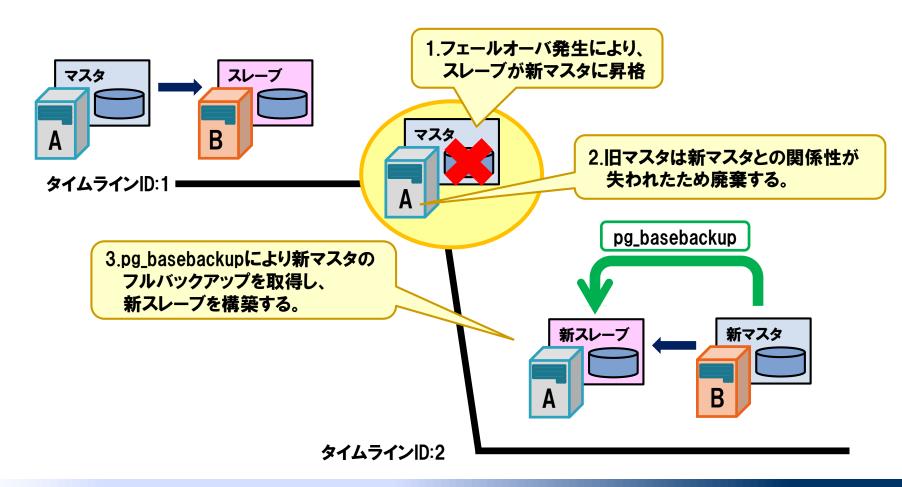
- 同期レベルと更新性能はトレードオフ
 - 非同期(synchronous_commit = local)の値を1.00とした相対値



→性能影響を受けやすい

(3) pg_rewindによる巻き戻し

■ マスタ/スレーブの再構築(9.4以前) フルバックアップの再取得が必要なため時間を要する。



(3) pg_rewindによる巻き戻し

■ マスタ/スレーブの再構築(9.5以降) pg_rewindにより障害が発生した旧マスタを巻き戻し、 2.pg_rewind実行のため 新スレーブとして再構築する。 旧マスタを再起動後、 正常停止 1.フェールオーバ発生により、 スレーブが新マスタに昇格 マスタ スレーブ 旧マスタ マスタ В タイムラインID:1 4.新スレーブに新マスタのWALを 3.pg rewindによる巻き返し 適用し、新マスタへ追い付く 新スレーブ 新マスタ

タイムラインID:2

ストリーミングレプリケーション:まとめと今後

- PostgreSQLのストリーミングレプリケーション機能は年々追加されているため、既存情報の整理と新機能を調査した。
- この発表資料では注目していただきたい事項として3つの機能を取り上げた。

① レプリケーションスロット : 基本動作と注意点

② 同期レベル : 同期レベル毎のパフォーマンス

③ pg_rewind : 従来手法と比べた優位点

■ 次期メジャーバージョンのPostgreSQL 10でもレプリケーション 機能の拡充が予定されているので、重要分野と位置づけ、新機能検証を継続する。

レプリケーション 成果紹介 - Bi-Directional Replication -

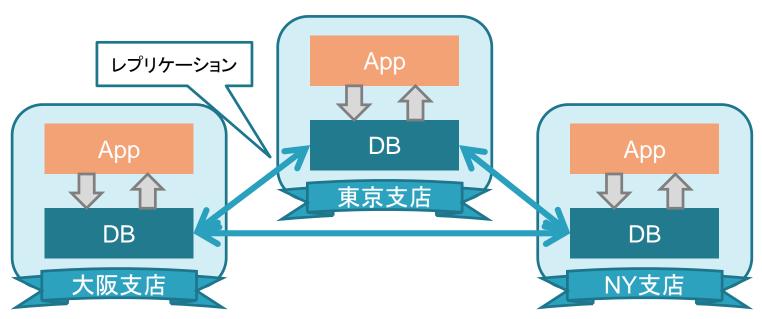
BDR概要

- 「2ndQuadrant」によって開発されたオープンソース(PostgreSQL License)のマルチマスタ・レプリケーションシステム。
- 双方向の非同期論理レプリケーションを使用し、地理的に分散したクラスタで使用するために設計されている。
- サポートについては以下が存在。

No.	サポート	概要
1	無償サポート	BDRコミュニティへのメール、BDRのGoogleグループが存在 email: bdr-list@2ndQuadrant.com Googleグループ: https://groups.google.com/a/2ndQuadrant.com/forum/#!forum/bdr-list
2	有償サポート	2ndQuadrant社によるサポートを受けることが可能 ※ 2ndQuadrant社について BDRの製造元で、PostgreSQLの専門家(コミッター等)が多数在籍する企業です。 PostgreSQLのコンサルティングサービス等を提供。 https://2ndquadrant.com/en/

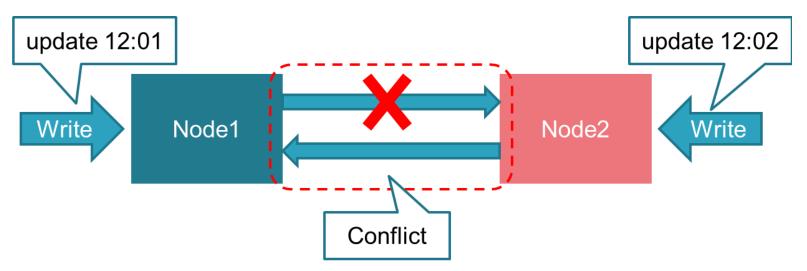
ユースケース

- BDRは以下に該当する場合に有用である。
 - 1. 遠隔地や高レイテンシ環境でクラスタを構成している場合
 - 2. 各ノードが書き込み処理を実施する場合
 - 3. クラスタ間でデータが非同期であることを許容できる場合
- 例としては以下のようにレスポンスを向上させるために各地でアプリケーション 及びDBを動作させるようなケースが考えられる。



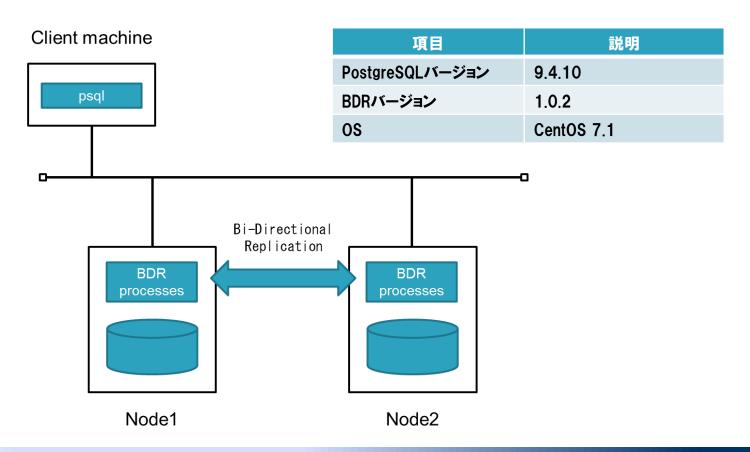
整合性

- マルチマスタ構成を取る場合、各ノードが持つ情報に不整合が発生しないように 管理する仕組みが必要となる。
- BDRでは結果整合性(Eventual Consistency)と呼ばれる一貫性モデルを採用し、整合性を確保。
 - □ デフォルトでは競合が発生した場合、最後の更新処理が適用される。(last_update_wins)
 - □ 競合結果はテーブル「bdr.bdr_conflict_history」で確認可能。



実機検証環境構成図

■ BDRの機能や性能向上を確認するために、下図のようなマルチマスタ環境を構築し、実機検証を実施した。



実機検証結果(1/2)

■ 実機検証の結果は下記の通り。

【 凡例 (結果): 【 ○: 問題なし

| △: 利用時に問題になるケースがある | ×: 対応していない。事実上使えない

項番	検証項目	概要	結果	実施結果補足
1	ノードの追加/切り離し	既存のマルチマスタ環境に対するノード追加/切り離し手順を明確にする。	Δ	ノード切り離す場合に、システムカタログを操作する必要がある。 また、ノード追加時には、トランザクションの停止が必要。
2	更新処理競合時の 動作確認	各ノードで実施された、 更新処理が競合した場 合の挙動を確認する。	Δ	基本的には最後に実行された更新処理が反映される。 ただし、いくつかの競合パターンで、予期せぬエラーが発生する。
3	ノード毎のシーケンス の 独立	ノード毎に払い出される シーケンス番号を独立 させ、シーケンス番号の 競合が抑止できるか確 認する。	0	ノード毎にシーケンス番号を発行 し、 競合を防ぐことが可能

実機検証結果(2/2)

■ 実機検証の結果は下記の通り。

□ 凡例(結果): ■ ○: 問題なし

△: 利用時に問題になるケースがある

| × : 対応していない。事実上使えない

項番	検証項目	概要	結果	実施結果補足
4	選択的レプリケーション	データベース内の一部の テーブルのみを レプリケーション対象とする 手順を確認する。	0	テーブル毎に柔軟なレプリケーション設 定が可能
5	ノード障害時の挙動確認	複数ノードで構成されるクラスタ環境内の1ノードに 障害が発生した場合の挙動を確認する。	0	他ノードで実行中のトランザクションに 影響が及ばない
6	障害ノードの復旧確認	障害が発生したノードをク ラスタ環境に復旧させる手 順を確認する。	Δ	ノード追加時には、トランザクションの 停止が必要。 また、システムカタログの操作が必要。
7	性能検証	遠隔地間のデータレプリ ケーションを想定し、レスポ ンスタイムが改善するか確 認する。	0	遠隔地や帯域幅が細い等の高レイテンシ(遅延)環境では、 BDRを利用することによって、レスポンスタイム及びTPSを大幅に向上させることができる。(日本-シンガポール間の環境を擬似した結果は5.4倍向上)

Bi-Directional Replication:まとめと今後

- BDRを用いることで、SRでは実現できないマルチマスタ・レプリケーションや、 選択的レプリケーション(テーブル単位のレプリケーション)を実現できることを 確認できた。
 - □ 選択的レプリケーションを用いて柔軟なデータ連携(データ集約等)が実現可能
- 性能試験において、BDRのユースケースで想定している「遠距離拠点間で双方向に 更新する」場合にレスポンスタイムの低減と処理向上につながることが確認できた。
- 一方、本検証で利用したBDR バージョン 1.0.2では更新処理が競合するパターンでは、 意図しない動作が発生する場合が確認された。
 - □ BDRを適用する場合、競合が発生しないようなアプリケーション設計が必要と考えられる。 例)更新するテーブルを拠点ごとに分ける等
- ノード障害時の運用においても、一般的でないシステムカタログの操作を必要とするといった、今後の改善が必要と思われる結果が確認された。
- BDRの次期バージョン(2.0)がリリースされた後、 PGEConsにおいて再度検証を実施したいと考えている。

レプリケーションのまとめ

データの一貫性が シングルマスタ 保証されやすい 更新 参照 同期 同期 マスタ

PostgreSQLの機能である ストリーミングレプリケーションにより実現

更新操作の負荷を マルチマスタ 分散できるが、デー タの同期が困難 更新 参照 同期 同期 マスタ マスタ マスタ

BDRが論理レプリケーションを用いて実現

◆目的に応じて機能を選択 → 設定方法はドキュメントを参考に!

異種DB連携班成果紹介

異種DB連携班の活動目的

- ■背景
 - □ ICTシステムの進化によるシステム統合/拡張は不可避
- 問題点
 - □ システム全体を統合/拡張するのは高コスト
 - 既存システムを活用したままサブシステムを拡張
 - データベースも既存のデータベースと連携させて拡張
- 目的
 - □ PostgreSQLと異種DBMSの連携方式をピックアップ
 - □ 各連携方式の使いどころや注意点を整理する

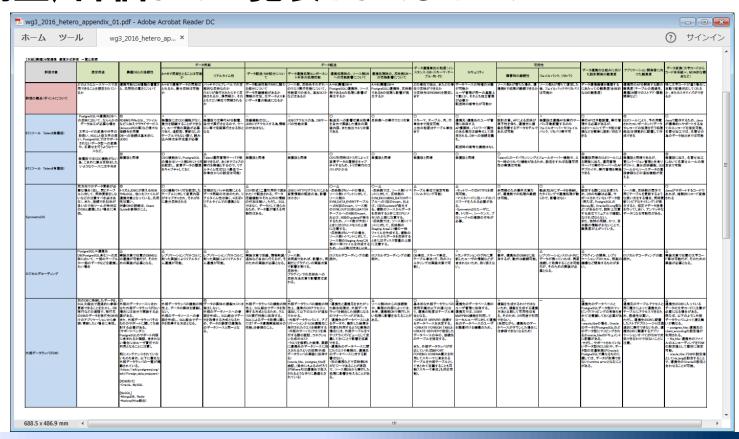
検証した異種DB連携方式

連携方式	想定用途	検証対象ツール
外部データ ラッパ	PostgreSQL側でデータを持たず、データを共有したい場合	PostgreSQL本体機能 +拡張機能
ロジカル デコーディング	DB (PostgreSQL含む) への反映において、一部のテーブルや一部のデータなどを連携したい場合	PostgreSQL本体機能 +拡張機能
DB同期ツール	ある程度の同期性を確保しつつ、異 種DB間でデータを共有したい場合	SymmetricDS
ETLツール	PostgreSQL⇒連携先DBへの反映に おいて、なんらかのデータ加工が必要 な場合	Talend

→上記に挙げた異種DB連携方式について、調査・検証を行った

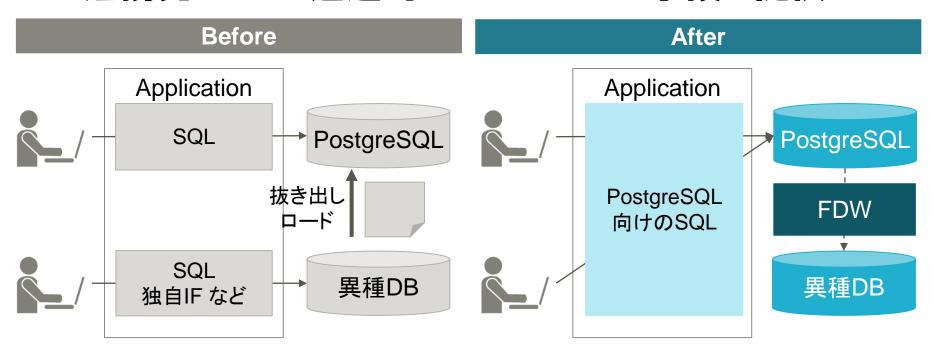
異種DB連携の調査/評価観点

■ 想定用途、接続性、同期性、性能、セキュリティ、構築・開発時の難易度等の観点で異種DB連携方式を調査/評価して一覧表としてまとめた



外部データラッパ

■ 連携先のDBに透過的にアクセスする手段を提供



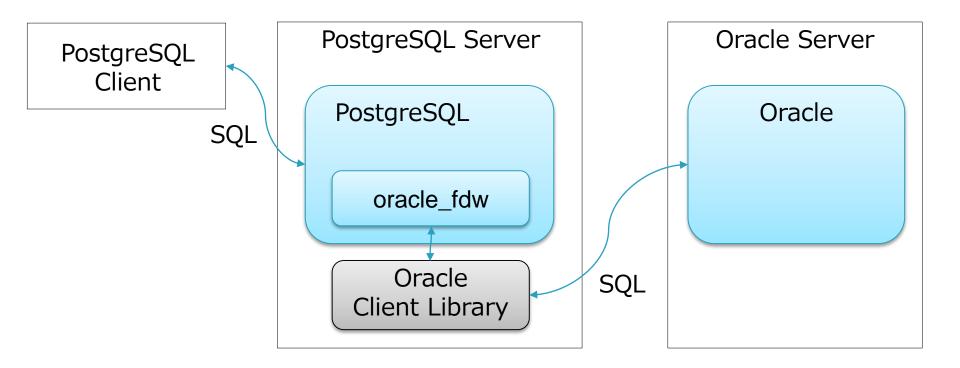
連携先のDB(ファイル等のデータソース含) に応じてSQL、独自IFを使い分けが必要

使い慣れたSQLを利用でき アプリ開発、学習コストの低減

既存Dから直接データを 利用する際には抜き出し & ロードが必要 PostgreSQLのSQL文を利用し 透過的にアクセス可能

外部データラッパ

■ 例:oracle_fdw



■ PostgreSQL SQL内容を元に、Oracleのクライアントライブラリを経由してOracle SQLを発行して情報を取得する

参考:主な外部データラッパ

- 汎用インターフェース
 - **ODBC**
 - **JDBC**
- 主なRDBMSとの連携向け
 - **PostgreSQL**
 - Oracle
 - **MySQL**
- NoSQLとの連携向け
 - Cassandra
 - CouchDB
 - MongoDB
 - Neo4i
 - Redis

出典:Foreign data wrappers より抜粋

https://wiki.postgresql.org/wiki/Foreign_data_wrappers

- ビッグデータ基盤との連携向け
 - **Elastic Search**
 - Hadoop

Foreign data wrappers



Foreign Data Wrappers

In 2003, a new specification called SQL/MED ("SQL Management of External Data") was added to the SQL standard. It is a standardized way of handling access to remote objects from SQL databases. In 2011, PostgreSQL 9.1 was released with read-only support of this standard, and in 2013 write support was added with PostgreSQL 9.3.

There are now a variety of Foreign Data Wrappers (FDW) available which enable PostgreSQL Server to different remote data stores, ranging from other SQL databases through to flat file. This page list some of the wrappers currently available. Another fdw list a can be found at the PGXN website a.

Please keep in mind that most of these wrappers are not officially supported by the PostgreSQL Global Development Group (PGDG) and that some of these projects are still in Beta version

Generic SQL Database Wrappers

Data Source	Туре	Licence	Code	Install	Doc	Notes
ODBC	Native		github 🙆	PGXN 🔒	example @	Does not compile with PostgreSQL >= 9.2!
ODBC	Native		github 👸			CartoDB took over active development of the ODBC FDW for PG 9.5+
JDBC	Native		github 🙆			Not maintained ?
JDBC2	Native		github 🙆			
SQL_Alchemy 🗗	Multicorn @	PostgreSQL	GitHub@	PGXN@	documentation @	Can be used to access data stored in any database supported by the sqlalchemy python toolki
VirtDB	Native	GPL	GitHub 8			A generic FDW to access VirtDB data sources (SAP ERP, Oracle RDBMS)

Specific SQL Database Wrappers

Data Source	Туре	Licence	Code	Install	Doc	Notes
PostgreSQL 4	Native	PostgreSQL	git.postgresql.org		documentation 🔒	

外部データラッパ

■ <u>ここが便利!</u>

□ PostgreSQL標準のSQLを使って、異種DBMSへの参照、あるいは更新が可能になる

■ <u>ここに注意!</u>

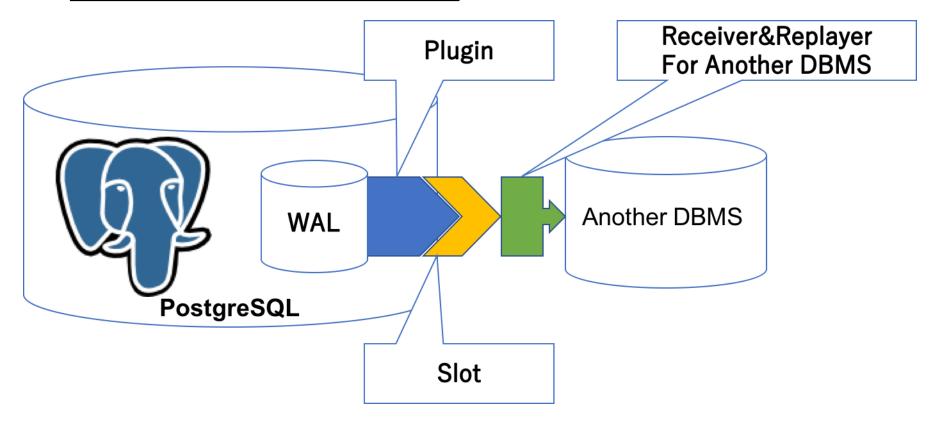
- □ データベースによってはFDWが実装されていないものがある。
- □ FDWの実装によっては、連携先のデータベースの能力を使えないケース がある
- 連携先のストアドプロシージャやユーザファンクションはFDW経由では実行できない

■ こういうときに使おう!

□ データそのものはPostgreSQLに持たせずに連携したい場合

ロジカルデコーディング

■ ロジカルデコーディングの概要



- Pluginにより論理ログに変換
- 変換された論理ログを元に別DBMSへ更新を反映させる

ロジカルデコーディング

- ここが便利!
 - □ PostgreSQL以外の異種DBMSへ更新内容を自由に伝播できる
 - □ 一部のスキーマ/テーブルの更新のみを伝播できる

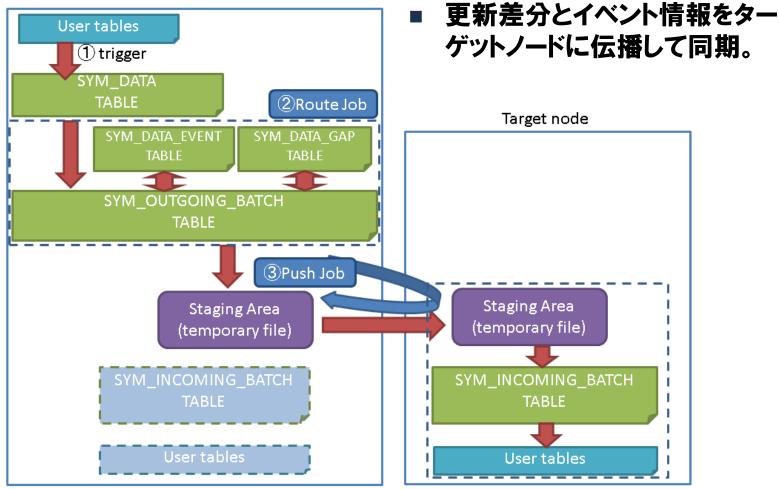
- ここに注意!
 - □ プラグインや更新反映用のプログラムの開発が必要

- こういうときに使おう!
 - □ PostgreSQL上の一部のテーブルを、異種DBと共有したい場合。

DB同期ツール

■ 例:SymmetricDS

Source node



■ ソースノードの更新を、更新差

分とイベント情報に変換

参考:主なサポートDBの機能対応

Database	Versions	Transaction Identifier	Data Capture	Conditional Sync	Update Loop Prevention	BLOB Sync	CLOB Sync
DB2	9.5	N	Υ	Υ	Υ	Y	Υ
DB2 for IBM i	6	N	Υ	Υ	N	Υ	Υ
Derby	10.3.2.1	Υ	Υ	Υ	Y	Υ	Υ
Firebird	2.0	Υ	Υ	Υ	Y	Y	Υ
Greenplum	8.2.15 and above	N	N	N	Υ	N	N
H2	1.x	Υ	Υ	Y	Υ	Υ	Υ
HSQLDB	1.8	Υ	Υ	Y	Υ	Υ	Υ
HSQLDB	2.0	N	Υ	Y	Υ	Υ	Υ
Informix	11	N	Υ	Y	Υ	N	N
Interbase	9.0	N	Υ	Υ	Υ	Υ	Υ
MySQL	5.0.2 and above	Υ	Υ	Y	Υ	Y	Υ
MariaDB	5.1 and above	Υ	Υ	Υ	Υ	Υ	Υ
Oracle	10g and above	Υ	Υ	Y	Υ	Y	Υ
PostgreSQL	8.2.5 and above	Y (8.3 and above only)	Υ	Υ	Υ	Y	Y
Redshift	1.0	N	N	N	Υ	N	N
SQL Anywhere	9	Y	Υ	Y	Y	Υ	Y
SQL Server	2005 and above	Υ	Υ	Y	Y	Y	Υ
SQL Server Azure	Tested on 11.00.2065	Y	Υ	Y	Y	Y	N
SQLite	3.x	N	Υ	Y	Υ	Υ	Υ
Sybase ASE	12.5	Υ	Υ	Y	Υ	Υ	Υ

DB同期ツール

ここが便利!

□ PostgreSQLと異種DB間で、双方向に、差分反映によるデータ更新ができる。

■ ここに注意!

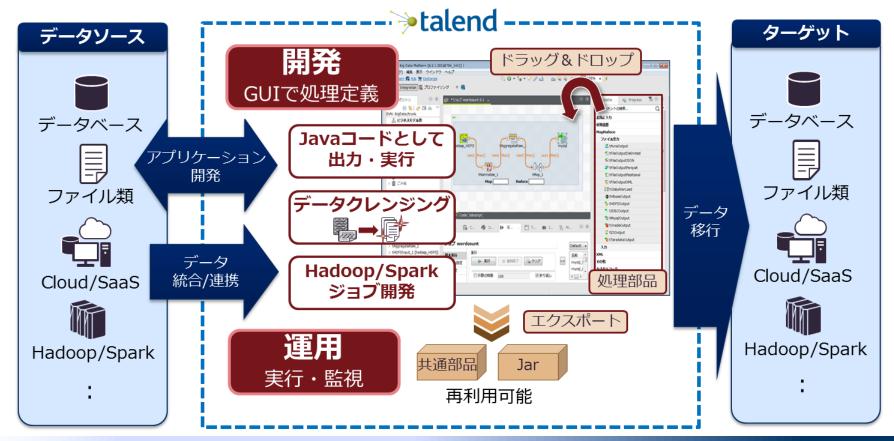
- □ 同期といいつつ、リアルタイムではなく、1秒程度の時差が発生する
- □ 同じテーブルに同時更新が発生すると、データアンマッチが発生する場合が・・・。
- □ ドキュメントの品質や、コミュニティが不活性なので、トラブル発生時には ハマる可能性が・・・。

■ こういうときに使おう!

- □ データ連携したいDBが、FDWに未対応だがJDBCに対応している場合
- □ マイグレーション等で、一部のDBはPostgreSQLに移行したが、残っている異種DBと、相互連携が必要な場合

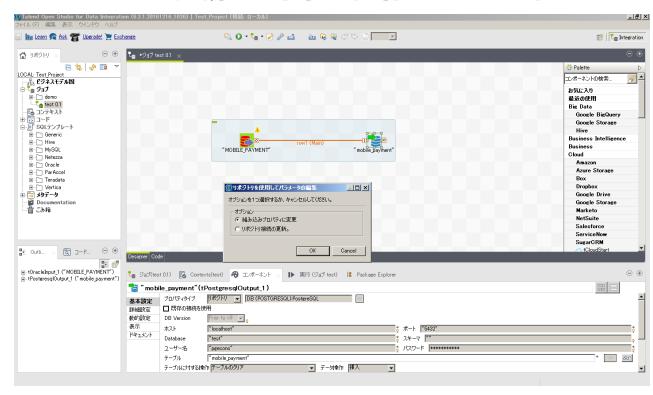
ETLツール (Talend) について

- Talend社が提供するデータ統合/連携製品
 - □ 機能追加された有償版と無償のOSS版(Talend Open Studio)があるが、 メインとなるETL開発に使えるコンポーネント(DBへの接続、データの抽出、 加工、反映)については 無償版でもほぼ差はない



Talend Open Studioの動作検証

- 成果物内で簡単な動作例を記載
 - □ Talendは、Eclipseベースで開発されておりインターフェースのつくりや操作 感覚もそれに近い
 - □ GUIでジョブを作成した後、jar形式の実行ファイル、batファイルとスクリプトまでビルドしてくれるのでデータ連携をお手軽に試行することが可能



参考 Talendの対応DB

データベース ビッグデータ NoSQL	Access	AS400	DB2	eXist	EXAsolution	Firebird	Greenplum
	HSQLDB	Informix	Ingres	Interbase	JavaDB	LDAP	MaxDB
	MS SQL Server	MySQL	Netezza	OLE DB	Oracle	ParAccel	Postgres Plus
	PostgreSQL	Redshift	VectorWise	SAS	SQLite	Sybase	Teradata
	Vertica	汎用JDBC					
	Cassandra	Couchbase	CouchDB	Google BigQuery	SAP HANA	HBase	HCatalog
	HDFS	Hive	Impala	MarkLogic	MemSQL	MongoDB	Neo4j
ファイル	Pig	Riak	Sqoop				
	ARFF	Excel	LDIF	XML	区切りファイル	マルチレイアウト ファイル	Avro
	Parquet						
アプリケーション	Alfresco	Bonita	CentricCRM	Marketo	Microsoft AX	Microsoft CRM	NetSuite
アプリケーション 	Open Bravo	SageX3	SAP	SugarCRM	VtigerCRM		
クラウド	Amazon RDS (Aurora, MySQL, Oracle)			Amazon S3	Azure Storage	Box	Amazon EMR
	Amazon Redshift	Dropbox	Google Drive	Google Storage	Marketo	Salesforce	ServiceNow
ドットネット	DotNET						
標準規約	EDIFACT	HL7					
インターネット	JBoss	Petals	FTP	Kafka	MOM and JMS	SCP	HTTP Request
	POP	REST	RSS	Kerberos	Proxy	Socket	SVN
	WebService	XML RPC	SMTP	SOAP	JSON		
ビジネスインテリ ジェンス	Jasper	OLAPキューブ(Mondrian, Palo)		SPSS			
Hadoop distribution	Hortonworks Data Platform	Apache	Cloudera	MapR	Amazon EMR	Pivotal HD	Microsoft HDInsight

ETLツール (Talend) について

ここが便利!

- □ 数少ない無償でも使えるETLツールながら機能が豊富
- □ 数多くのDBに対応した コンポーネントが標準で用意されている (NoSQLや クラウド環境まで)

■ ここに注意!

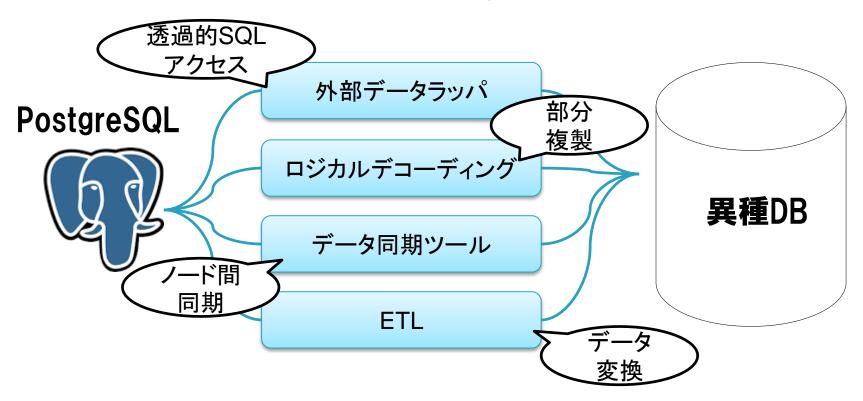
□ OSS版ではCDCの機能が使えないため、差分反映を設定することができない。 反映時に都度truncateするなど運用による対策が必要

■ こういう時に使おう!

- □ 対象DBでのシステム開発の経験が少ない
- □ 開発期間が短い、開発工数が少ない
- □ APの継続的な改修があり、処理の可視化などのメンテナンス性が必要

異種DB連携のまとめ

- 異種DB連携方式の整理
 - □ さまざまな異種DB連携方式をPostgreSQLでは利用できる。



- □ 用途に応じた異種DB連携方式の選択が必要。
- □ 既存のデータ・資産をそのまま残しつつ、システムの拡張が可能に!



2016年度の活動を振り返って

- 参加企業の皆様からいただいたコメント
 - □ 初めてWGに参加し、他社との議論やレビューで新たな観点 や考え方を学ぶことができた。
 - □ WG活動に貢献しながらストリーミング・レプリケーションの知見を深めることができた。
 - □ 今まで使っていなかったロジカルデコーディングやBDRについての知見を得られました。
 - □ Githubの機能を活用し、昨年度より活発な議論ができた。

2017年度のWG3活動

- 課題検討WG
 - □課題検討WGという自由度のある部会としてPostgreSQL 10の新機能や各種運用ツール類を対象にテーマを設定し 検討。

一緒にPGECons WG活動しませんか?



PGECons PostgreSQL Enterprise Consortium