

WG2活動報告書
メジャーバージョンアップ

目次

目次	2
1. 改訂履歴	5
2. ライセンス	6
3. はじめに	7
3.1. 本資料の概要と目的	7
3.2. 本資料の構成	7
3.3. 本資料と『バージョンアップ編』との関係	7
3.4. 補足	7
4. メジャーバージョンアップ手順	9
4.1. 概要	9
4.2. 要件確認	9
4.2.1. 環境	9
4.2.1.1. 同一環境でのバージョンアップ	9
4.2.1.2. 別環境でのバージョンアップ	9
4.2.2. 運用	9
4.3. 非互換確認	10
4.3.1. メジャーバージョン間の差異	10
4.3.1.1. 機能非互換	10
4.3.1.2. 非機能要素	10
4.4. メジャーバージョンアップ手順	10
4.4.1. pg_upgradeを利用する場合	11
4.4.2. pg_dumpallを利用する場合	11
4.4.3. pg_dump/pg_restoreを利用する場合	11
5. メジャーバージョン間の非互換	13
5.1. 概要	13
5.2. メジャーバージョン間の非互換	13
5.2.1. サーバプロセス	13
5.2.2. ディレクトリ、ファイル	13
5.2.3. サーバパラメーター	14
5.2.3.1. カテゴリごとのサーバパラメーター数	15
5.2.3.2. サーバパラメーター初期値	16
5.2.4. システムカタログ	18
5.2.5. システムビュー	20
5.2.6. 機能・非機能	21
5.2.6.1. バックアップ・リカバリーおよびストリーミング	21
5.2.6.2. 監視	22
5.3. 拡張機能	22
5.3.1. pg_dbms_stats	22
5.3.2. pgBadger	23
5.3.3. pg_hint_plan	23
5.3.4. pg_monz	23
5.3.5. pg_repack	24
5.3.6. pg_rman	24
5.3.7. pg_statsinfo	25
5.3.8. pg_bulkload	25
6. メジャーバージョンアップ検証	27
6.1. 概要	27
6.2. 想定するストーリー	27
6.3. 検証パターン	27
6.4. 検証環境	27
6.4.1. サーバ構成	27
6.4.2. 機能拡張	28
6.5. 検証結果	28
7. 著者	29

1. 改訂履歴

版	改訂日	変更内容
1.0	2020/03/31	新規作成

2. ライセンス

本作品はCC-BYライセンスによって許諾されています。ライセンスの内容を知りたい方は[こちら](#)でご確認ください。文書の内容、表記に関する誤り、ご要望、感想等につきましては、[PGECconsのサイト](#)を通じてお寄せいただきますようお願いいたします。

- Eclipseは、Eclipse Foundation,Inc.の米国、およびその他の国における商標もしくは登録商標です。
- IBMおよびDb2は、世界の多くの国で登録されたInternational Business Machines Corporationの商標です。
- Intel、インテルおよびXeonは、米国およびその他の国における Intel Corporation の商標です。
- Javaは、Oracle Corporation 及びその子会社、関連会社の米国及びその他の国における登録商標です。文中の社名、商品名等は各社の商標または登録商標である場合があります。
- Linux は、Linus Torvalds 氏の日本およびその他の国における登録商標または商標です。
- Red HatおよびShadowman logoは、米国およびその他の国におけるRed Hat,Inc.の商標または登録商標です。
- Microsoft、Windows Server、SQL Server、米国 Microsoft Corporationの米国及びその他の国における登録商標または商標です。
- MySQLは、Oracle Corporation 及びその子会社、関連会社の米国及びその他の国における登録商標です。文中の社名、商品名等は各社の商標または登録商標である場合があります。
- Oracleは、Oracle Corporation 及びその子会社、関連会社の米国及びその他の国における登録商標です。文中の社名、商品名等は各社の商標または登録商標である場合があります。
- PostgreSQLは、PostgreSQL Community Association of Canadaのカナダにおける登録商標およびその他の国における商標です。
- Windows は米国 Microsoft Corporation の米国およびその他の国における登録商標です。
- TPC, TPC Benchmark, TPC-C, TPC-E, tpmC, TPC-H, QphHは米国Transaction Processing Performance Councilの商標です
- その他、本資料に記載されている社名及び商品名はそれぞれ各社が 商標または登録商標として使用している場合があります。

3. はじめに

3.1. 本資料の概要と目的

本資料は、PostgreSQLのメジャーバージョンアップを検討される方の参考にしていただくことを目的に、PostgreSQLエンタープライズ・コンソーシアム(以下PGECcons)が作成・公開をしています。PGECconsは、PostgreSQL本体および各種ツールの情報収集と提供、整備などの活動を通じて、ミッションクリティカル性の高いエンタープライズ領域へのPostgreSQLの普及を推進することを目的として設立された団体です。PGECconsの技術部会では、PostgreSQLの普及に対する課題の検討を通じて活動テーマを挙げ、そのテーマごとにワーキンググループを立ち上げて活動しています。その中の一つであるWG2(移行ワーキンググループ)では、「異種DBMSからPostgreSQLへの移行」をテーマとして調査・検証を行い、収集した技術ノウハウを成果として取り纏めた資料を公開しています。

今回、2020年2月に最終リリースとなったPostgreSQL 9.4から、本資料作成時点で最新のPostgreSQL 12へのメジャーバージョンアップをモデルとし、PostgreSQLにおけるメジャーバージョンアップ手順、メジャーバージョン間の主な非互換についてまとめ、メジャーバージョンアップの実機検証をおこないました。

参考として、下表にメジャーバージョンごとの最終リリース日を掲載します。なお、最新情報はPostgreSQL.orgの[バージョンポリシーのページ](#)で確認するようにしてください。

表 3.1 メジャーバージョンごとの最終リリース

メジャーバージョン	最初のリリース	最終リリース
12	2019年10月3日	2024年11月14日
11	2018年10月18日	2023年11月9日
10	2017年10月5日	2022年11月10日
9.6	2016年9月29日	2021年11月11日
9.5	2016年1月7日	2021年2月11日
9.4	2014年12月18日	2020年2月13日

本資料が、皆様のPostgreSQL運用の一助となれば幸いです。

3.2. 本資料の構成

本資料の本文の構成は下表の通りです。

表 3.2 本資料の構成

章	概要
3 はじめに	本資料の位置づけと概要
4 メジャーバージョンアップ手順	PostgreSQLのメジャーバージョンアップ手順について
5 メジャーバージョン間の非互換	メジャーバージョン間の主な非互換について
6 メジャーバージョンアップ検証	メジャーバージョンアップを実機検証した結果について

3.3. 本資料と『バージョンアップ編』との関係

PGECcons WG2にて2014年に作成した『[バージョンアップ編](#)』は、バージョンアップ時に利用するツールおよびその使用方法に主眼をおき、それらツールの実機検証をおこない、まとめた資料です。それに対し、本資料『メジャーバージョンアップ編』は、メジャーバージョンアップ作業に主眼をおいたものになります。

pg_dumpやpg_upgradeなど本資料に登場するツールの詳細については、『[バージョンアップ編](#)』や他のPGECcons成果物、ならびにマニュアルを参照するようにしてください。

3.4. 補足

- PostgreSQLのマニュアルでは、メジャーアップグレードやメジャーバージョンアップグレードなど、『バージョンアップ』ではなく『アップグレード』という表記をしていますが、本資料では『バージョンアップ』と表記しています。

- パブリッククラウドのフルマネージドサービスで提供されるPostgreSQL（例：Azure Database for PostgreSQL、Amazon RDS for PostgreSQL、Cloud SQL for PostgreSQLなど）における、メジャーバージョンアップについては本資料では扱いません。

4. メジャーバージョンアップ手順

4.1. 概要

この章では、メジャーバージョンアップの基本的な流れについて記述します。

PostgreSQLのメジャーバージョンアップでは、バグ修正やセキュリティ修正だけでなく、新機能の追加などがおこなわれます。また、異なるメジャーバージョン間ではデータベースクラスタの互換性が保証されておらず、データベースクラスタを新しいバージョンにあわせて変換する作業が必要です。

つまり、メジャーバージョンアップでは、バイナリを入れ替えるだけで済むマイナーバージョンアップとは異なり、データベースクラスタの移行方法など検討しなければならないポイントが多くなります。

4.2. 要件確認

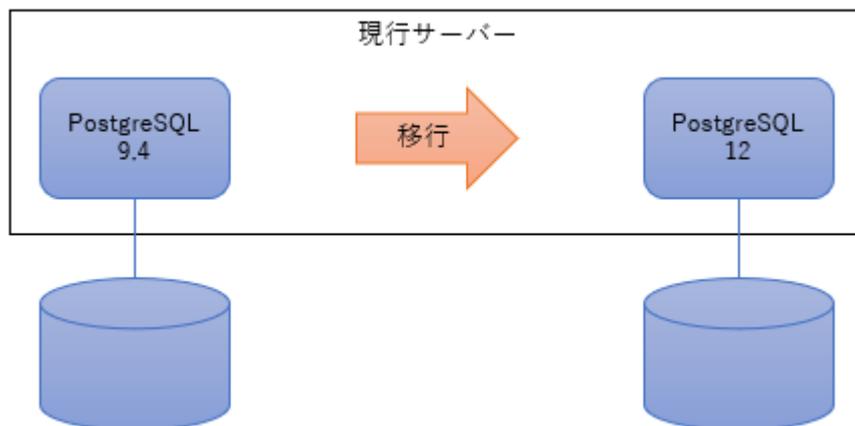
まず、メジャーバージョンアップ作業の要件を確認します。

4.2.1. 環境

今日では、オンプレミスの物理サーバーであったり、パブリッククラウドの仮想サーバーやであったり、またはコンテナであったりと、環境といっても多岐にわたりますが、おおまかには下記の2パターンが考えられます。

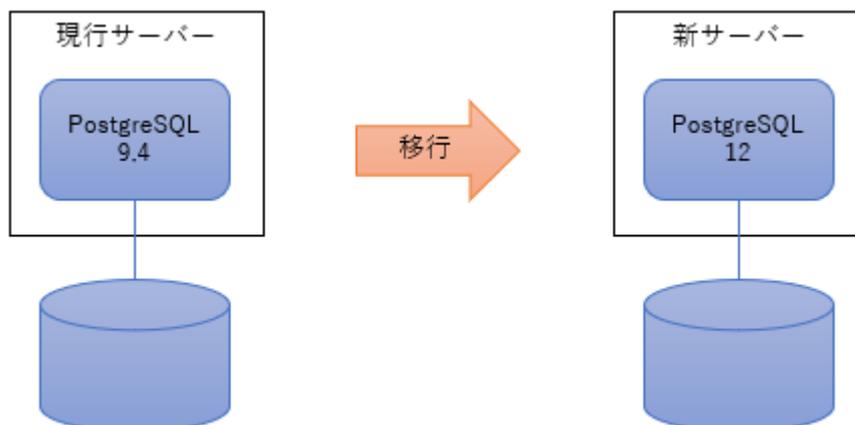
4.2.1.1. 同一環境でのバージョンアップ

同一のサーバーやコンテナなどに、新しいバージョンのPostgreSQLをインストールして、移行するケース



4.2.1.2. 別環境でのバージョンアップ

別のサーバーやコンテナなどに、新しいバージョンのPostgreSQLをインストールして、移行するケース



4.2.2. 運用

運用要件として、メジャーバージョンアップ作業を実施するにあたり、システム停止が可能なかどうかは重要なポイントです。

メジャーバージョンアップでは、データベースクラスタの移行が発生しますので、これにかかる時間を見積もります。そして、システムの稼働率、もしくは年間許容停止時間と照らし合わせ、最適な移行方法を検討します。システムを停止できない場合には、段階的なデータ移行や、データレプリケーション製品を活用するなどの工夫が必要になってきます。

また、許容停止時間が短い重要システムであれば、冗長化構成であったり、ストリーミングレプリケーション構成としている場合も多くなりますので、さらに移行手順の検討要素が増えます。

表 4.1 参考：稼働率と年間許容停止時間

稼働率	年間許容停止時間	考えられる手法
99 %	3.65日	多くの場合、システムを停止したうえでの作業が十分に可能。
99.9 %	8.8時間	データ移行に長時間を要する大規模データであれば、コールドデータを先に移行しておくなど、データ移行を段階的にこなうことを考える。
99.99 %	52.6分	データレプリケーション製品を活用し、データは常に同期したうえで接続先だけを切り替える、などの工夫が必要。バージョンアップ元のバージョンが10以降であれば、ロジカルレプリケーションを利用することも検討する。
99.999 %	5.3分	同上。

4.3. 非互換確認

メジャーバージョン間では新機能の追加だけでなく、機能非互換がある場合があります。それら直接的な機能の非互換や、それに伴う非機能の非互換を調査します。なお、主な非互換については、5章に記述します。

4.3.1. メジャーバージョン間の差異

メジャーバージョンアップのターゲットバージョンとの差異を確認します。

4.3.1.1. 機能非互換

表 4.2 機能非互換

項目	内容
システムカタログ	システムカタログの内容を参照、利用するアプリケーションでは、修正が必要です。
サーバーパラメーター	新規パラメーター、廃止されたパラメーター、初期値が変更になったパラメーターについて調査し、値を再設計、そして動作検証が必要です。
ディレクトリやファイルの配置	PostgreSQL 10においてディレクトリ名が変更、PostgreSQL 12でrecovery.confがpostgresql.confに統合されるなど、変更があります。
プロセス名	プロセス監視をしている場合、プロセス名変更への対処が必要です。
クライアントアプリケーション	pg_ctlなど、オプションの初期値が変更になっているものがあり、それらを利用するシェルなどの対処が必要です。
拡張機能	拡張機能もあわせてバージョンアップが必要になる場合があります。新バージョンに対応していない場合は代替案の検討も必要です。

4.3.1.2. 非機能要素

表 4.3 機能非互換

項目	内容
バックアップ・リカバリ	PostgreSQL 12でrecovery.confがpostgresql.confに統合されるなどの変更があります。
監視	pg_monzにおいて、それ自身のバージョンアップが必要な場合は、監視項目の再確認が必要です。独自に監視している場合、プロセス名の変更やシステムカタログの変更への対処が必要です。

4.4. メジャーバージョンアップ手順

メジャーバージョンアップのおおまかな手順について、まとめます。

留意事項

- すべての作業を網羅しているわけではありません。
- 手順は、別環境に移行するケースを想定しています。
- 手順に書かれているツールのオプションは、一例です。
- ロジカルレプリケーション（pglogicalなどの拡張機能、またはバージョンアップ元がPostgreSQL 10以降）には触れません。

- 本資料はPostgreSQL 9.4からPostgreSQL 12へのメジャーバージョンアップをモデルとしているため、ここでは省略します。
- 本資料作成時点で、pglogicalはPostgreSQL 12未対応です。
- ストリーミングレプリケーションは、異なるメジャーバージョン間では利用できません。

4.4.1. pg_upgradeを利用する場合

- ここでは、リンクモード (--link) を使用することを前提とします。
- リンクモードでは新バージョンのデータベースクラスタに旧バージョンのデータファイルへのハードリンクを張ります。そのため、新旧バージョンのデータクラスタを同一ファイルシステムに配置する必要があります。
- 新バージョンでデータベースクラスタを作成する (initdb) 際、旧バージョンと互換性のあるオプション (文字コードなど) とする必要があります。
- 新バージョンと旧バージョンのポートは同一で構いません。
- 新バージョンと旧バージョンのPostgreSQLインストールユーザー (OSユーザー) は同一である必要があります。
- その他、pg_upgradeの使用方法、注意点については、マニュアルもしくは『バージョンアップ編』を参照してください。

表 4.4 pg_upgrade手順概要

No.	旧環境作業	新環境作業
1		新バージョンのPostgreSQL、拡張機能をインストール
2		旧バージョンのPostgreSQLをインストール
3		新バージョンでデータベースクラスタを作成 (initdb)
4	旧バージョンのPostgreSQLを停止	
5		旧環境からデータベースクラスタをコピー
6		新バージョンのpg_upgradeを実行
7		新バージョンのPostgreSQLを起動
8		新バージョンの拡張機能をバージョンアップ (ALTER EXTENSION ... UPDATE) または再インストール (CREATE EXTENSION)
9		新バージョンでバキュームおよび統計情報の再取得 (e.g. vacuumedb --all --analyze)
10		新バージョンのPostgreSQLで、データ確認、アプリケーション動作確認
11	環境削除	旧バージョンのPostgreSQLおよびデータベースクラスタを削除

4.4.2. pg_dumpallを利用する場合

- 要件に依存しますが、小規模なデータベースであれば十分現実的です。

表 4.5 pg_dumpall手順概要

No.	旧環境作業	新環境作業
1		新バージョンのPostgreSQL、拡張機能をインストール
2		新バージョンでデータベースクラスタを作成 (initdb)
3	旧バージョンで新環境からの管理者アクセス以外を禁止にする (e.g. pg_hba.conf など)	
4		新バージョンのpg_dumpallで、旧バージョンに接続し、ダンプする (e.g. pg_dumpall --host=<旧環境> --if-exists --clean -quote-all-identifiers > olddb.out)
5		psqlで、ダンプファイルをリストアする (e.g. psql --host=<新環境> --file=olddb.out)
6		新バージョンでバキュームおよび統計情報の再取得 (e.g. vacuumedb --all --analyze)
7		新バージョンのPostgreSQLで、データ確認、アプリケーション動作確認
8	環境削除	ダンプファイルを削除

4.4.3. pg_dump/pg_restoreを利用する場合

- 新バージョンに移行するデータを取捨選択できます。
- コールドデータを先に移行するなど、段階的にデータを移行する場合に有効です。

表 4.6 pg_dump/pg_restore手順概要

No.	旧環境作業	新環境作業
1		新バージョンのPostgreSQL、拡張機能をインストール
2		新バージョンでデータベースクラスタを作成 (initdb)
3	旧バージョンで新環境からの管理者アクセス以外を禁止にする (e.g. pg_hba.conf など)	
4		新バージョンのpg_dumpallで、旧バージョンに接続し、ロールなどのグローバルオブジェクトを移行する
5		新バージョンのpg_dumpで、旧バージョンに接続し、スキーマなどを移行する
6		新バージョンのpg_dumpで、旧バージョンに接続し、ダンプする (e.g. pg_dump --host=<旧環境> --format=custom --data-only --schema=hr --file=hr.dmp --quote-all-identifiers sampledb)
7		新バージョンのpg_restoreで、ダンプファイルをリストアする (e.g. pg_restore --host=<新環境> --dbname=sampled -jobs=4 hr.dmp)
8		新バージョンでバキュームおよび統計情報の再取得 (e.g. vacuumedb --all --analyze)
9		新バージョンのPostgreSQLで、データ確認、アプリケーション動作確認
10	環境削除	ダンプファイルを削除

5. メジャーバージョン間の非互換

5.1. 概要

この章では、PostgreSQL 9.4からPostgreSQL 12へのメジャーバージョンアップ時に把握しておくべき差異や非互換について、記述します。

ただし、すべての差異や非互換について書き出したものではないことに注意してください。本章の内容をベースに、マニュアルのリリースノートなどを確認するようにしてください。

5.2. メジャーバージョン間の非互換

下記について、主な差異、非互換を挙げます。

- サーバプロセス
- ディレクトリ、ファイル
- サーバパラメーター
- 機能・非機能

5.2.1. サーバプロセス

稼働するサーバプロセスの名称に、差異があります。

これにより、PostgreSQLサーバの稼働監視（プロセス監視）に影響が発生する可能性があります。下表は、initdb を実行後すぐに pg_ctl start した場合に、実行されるサーバプロセス名を比較したものです。

表 5.1 サーバプロセス名称比較

9.4	12	説明
/usr/pgsql-9.4/bin/postgres	/usr/pgsql-12/bin/postgres	
postgres: logger process	postgres: logger	
postgres: checkpoint process	postgres: checkpoint	
postgres: writer process	postgres: background writer	
postgres: wal writer process	postgres: walwriter	
postgres: autovacuum launcher process	postgres: autovacuum launcher	
postgres: stats collector process	postgres: stats collector	
	postgres: logical replication launcher	論理レプリケーションを担う、PostgreSQL 10からの新規プロセス

5.2.2. ディレクトリ、ファイル

データベースクラスタを構成するディレクトリやファイルに、差異があります。

表 5.2 データベースクラスタの構成要素比較

9.4	12	変更	説明
PG_VERSION	PG_VERSION		PostgreSQLの主バージョン番号を保有するファイル
base/	base/		データベースごとのサブディレクトリを保有するサブディレクトリ
	current_logfiles	○	ログ収集機構が現在書き込んでいるログファイルを記録するファイル
global/	global/		pg_databaseのようなクラスタで共有するテーブルを保有するサブディレクトリ
	pg_commit_ts/	○	トランザクションのコミット時刻のデータを保有するサブディレクトリ
pg_dynshmem/	pg_dynshmem/		動的共有メモリサブシステムで使われるファイルを保有するサブディレクトリ
pg_logical/	pg_logical/		論理デコードのための状態データを保有するサブディレクトリ
pg_multixact/	pg_multixact/		マルチトランザクションの状態のデータを保有するサブディレクトリ（共有行ロックで使用されます）
pg_notify/	pg_notify/		LISTEN/NOTIFY状態データを保有するサブディレクトリ
pg_replslot/	pg_replslot/		レプリケーションスロットデータを保有するサブディレクトリ
pg_serial/	pg_serial/		コミットされたシリアライズブルトランザクションに関する情報を保有するサブディレクトリ
pg_snapshots/	pg_snapshots/		エクスポートされたスナップショットを保有するサブディレクトリ
pg_stat/	pg_stat/		統計サブシステム用の永続ファイルを保有するサブディレクトリ
pg_stat_tmp/	pg_stat_tmp/		統計サブシステム用の一時ファイルを保有するサブディレクトリ
pg_subtrans/	pg_subtrans/		サブトランザクションの状態のデータを保有するサブディレクトリ
pg_tblspc/	pg_tblspc/		テーブル空間へのシンボリックリンクを保有するサブディレクトリ
pg_twophase/	pg_twophase/		プリペアドトランザクション用の状態ファイルを保有するサブディレクトリ
pg_xlog/	pg_wal/	○	WAL（ログ先行書き込み）ファイルを保有するサブディレクトリ
pg_clog/	pg_xact/	○	トランザクションのコミット状態のデータを保有するサブディレクトリ
postgresql.auto.conf	postgresql.auto.conf		ALTER SYSTEMにより設定された設定パラメータを格納するのに使われるファイル
postmaster.opts	postmaster.opts		最後にサーバを起動した時のコマンドラインオプションを記録するファイル
postmaster.pid	postmaster.pid		現在のpostmasterプロセスID（PID）などを記録するロックファイル

5.2.3. サーバーパラメーター

サーバーパラメーターに、新設や廃止、初期値の差異があります。

5.2.3.1. カテゴリごとのサーバーパラメーター数

サーバーパラメーターをカテゴリごと、バージョンごとにカウントした表を下記に示します。

「Connections and Authentication / Security and Authentication」カテゴリのパラメーターは、PostgreSQL 11以降、row_securityパラメーターを除き、「Authentication」と「SSL」カテゴリに分割されており、row_securityパラメーターは、「Client Connection Defaults / Statement Behavior」カテゴリに移動しています。

表 5.3 カテゴリごとのサーバーパラメーター数比較

カテゴリ名	9.4	9.5	9.6	10	11	12
Autovacuum	11	11	11	11	11	11
Client Connection Defaults / Locale and Formatting	14	14	14	14	14	14
Client Connection Defaults / Other Defaults	5	5	5	5	5	6
Client Connection Defaults / Shared Library Preloading	3	3	3	3	4	4
Client Connection Defaults / Statement Behavior	21	22	23	23	25	26
Connections and Authentication / Authentication					5	5
Connections and Authentication / Connection Settings	9	9	9	9	9	9
Connections and Authentication / Security and Authentication	14	14	14	15		
Connections and Authentication / SSL					11	13
Developer Options	10	9	9	10	15	15
Error Handling	3	3	3	3	3	3
File Locations	5	5	5	5	5	5
Lock Management	3	3	3	5	5	5
Preset Options	11	12	12	12	13	14
Process Title		2	2	2	2	2
Query Tuning / Genetic Query Optimizer	7	7	7	7	7	7
Query Tuning / Other Planner Options	5	5	6	6	7	8
Query Tuning / Planner Cost Constants	6	6	9	10	13	13
Query Tuning / Planner Method Configuration	11	11	11	12	17	17
Replication		1	1	1	1	1
Replication / Master Server	2	2	2	2	2	2
Replication / Sending Servers	4	4	4	4	4	4
Replication / Standby Servers	6	7	7	7	7	11
Replication / Subscribers				2	2	2
Reporting and Logging / What to Log	17	18	18	18	18	18
Reporting and Logging / When to Log	3	3	3	3	3	4
Reporting and Logging / Where to Log	11	11	13	13	13	13
Resource Usage / Asynchronous Behavior	2	2	5	6	8	8
Resource Usage / Background Writer	3	3	4	4	4	4
Resource Usage / Cost-Based Vacuum Delay	5	5	5	5	5	5
Resource Usage / Disk	1	1	1	1	1	1
Resource Usage / Kernel Resources	1	1	1	1	1	1
Resource Usage / Memory	10	10	11	11	10	11
Statistics / Monitoring	4	4	4	4	4	4

カテゴリ名	9.4	9.5	9.6	10	11	12
Statistics / Query and Index Statistics Collector	6	5	5	5	5	5
Version and Platform Compatibility / Other Platforms and Clients	1	1	1	1	1	1
Version and Platform Compatibility / Previous PostgreSQL Versions	9	10	10	9	9	8
Write-Ahead Log / Archive Recovery						3
Write-Ahead Log / Archiving	3	3	3	3	3	3
Write-Ahead Log / Checkpoints	4	5	6	6	6	6
Write-Ahead Log / Recovery Target						8
Write-Ahead Log / Settings	10	11	12	12	12	14

5.2.3.2. サーバーパラメーター初期値

各バージョンのサーバーパラメーター初期値について、下表に示します。

表 5.4 サーバーパラメーター初期値比較

パラメーター	9.4	9.5	9.6	10	11	12	説明
extra_float_digits	0	0	0	0	0	1	リリースノート E.2.2. Migration to Version 12 参照
tcp_user_timeout						0	
jit_provider					llvmjit	llvmjit	
default_table_access_method						heap	
gin_pending_list_limit		4096	4096	4096	4096	4096	
idle_in_transaction_session_timeout			0	0	0	0	
row_security		on	on	on	on	on	
vacuum_cleanup_index_scale_factor					0.1	0.1	
password_encryption	on	on	on	md5	md5	md5	onもmd5の別名として扱う
ssl_max_protocol_version							
ssl_min_protocol_version						TLSv1	
ssl_passphrase_command							
ssl_passphrase_command_supports_reload					off	off	
ssl_renegotiation_limit	0						～v9.4までで廃止
debug_assertions	off						～v9.4までで廃止
jit_debugging_support					off	off	
jit_dump_bitcode					off	off	
jit_expressions					on	on	
jit_profiling_support					off	off	
jit_tuple_deforming					on	on	
wal_consistency_checking							
max_pred_locks_per_page				2	2	2	
max_pred_locks_per_relation				-2	-2	-2	
data_directory_mode					700	700	
debug_assertions		off	off	off	off	off	
server_version	9.4.25	9.5.20	9.6.16	10.11	11.6	12.1	
server_version_num	90425	90520	90616	100011	110006	120001	

パラメーター	9.4	9.5	9.6	10	11	12	説明
ssl_library						OpenSSL	
wal_segment_size	2048	2048	2048	2048	16777216	16777216	v11からバイト数に変更されているがすべて16MB
cluster_name							
update_process_title		on	on	on	on	on	
force_parallel_mode			off	off	off	off	
jit					off	on	
plan_cache_mode						auto	
jit_above_cost					100000	100000	
jit_inline_above_cost					500000	500000	
jit_optimize_above_cost					500000	500000	
min_parallel_relation_size			1024				v9.6のみ
min_parallel_index_scan_size				64	64	64	
min_parallel_table_scan_size				1024	1024	1024	
parallel_setup_cost			1000	1000	1000	1000	
parallel_tuple_cost			0.1	0.1	0.1	0.1	
enable_gathermerge				on	on	on	
enable_parallel_append					on	on	
enable_parallel_hash					on	on	
enable_partition_pruning					on	on	
enable_partitionwise_aggregate					off	off	
enable_partitionwise_join					off	off	
track_commit_timestamp		off	off	off	off	off	
max_replication_slots	0	0	0	10	10	10	
max_wal_senders	0	0	0	10	10	10	
hot_standby	off	off	off	on	on	on	
primary_conninfo							
primary_slot_name							
promote_trigger_file							
recovery_min_apply_delay						0	
wal_retrieve_retry_interval		5000	5000	5000	5000	5000	
max_logical_replication_workers				4	4	4	
max_sync_workers_per_subscription				2	2	2	
log_line_prefix	< %m >	< %m >	< %m >	%m [%p]	%m [%p]	%m [%p]	
log_replication_commands		off	off	off	off	off	
log_transaction_sample_rate						0	
log_directory	pg_log	pg_log	pg_log	log	log	log	v10からpg_logからlogに変更
syslog_sequence_numbers			on	on	on	on	
syslog_split_messages			on	on	on	on	
backend_flush_after			0	0	0	0	
max_parallel_maintenance_workers					2	2	
max_parallel_workers				8	8	8	
max_parallel_workers_per_gather			0	2	2	2	

パラメーター	9.4	9.5	9.6	10	11	12	説明
old_snapshot_threshold			-1	-1	-1	-1	
parallel_leader_participation					on	on	
bgwriter_flush_after			64	64	64	64	
replacement_sort_tuples			150000	150000			
shared_memory_type						mmap	
update_process_title	on						
default_with_oids	off	off	off	off	off		
operator_precedence_warning		off	off	off	off	off	
sql_inheritance	on	on	on				
archive_cleanup_command							
recovery_end_command							
restore_command							
checkpoint_segments	3						
checkpoint_flush_after			32	32	32	32	
max_wal_size		64	64	1024	1024	1024	v9.5新設で、v10から1GBに
min_wal_size		5	5	80	80	80	v9.5新設で、v10から80MBに
recovery_target							
recovery_target_action						pause	
recovery_target_inclusive						on	
recovery_target_lsn							
recovery_target_name							
recovery_target_time							
recovery_target_timeline						latest	
recovery_target_xid							
wal_compression		off	off	off	off	off	
wal_init_zero						on	
wal_level	minimal	minimal	minimal	replica	replica	replica	archiveやhot_standbyと設定してもreplicaにマップされる
wal_recycle						on	
wal_writer_flush_after			128	128	128	128	

5.2.4. システムカタログ

システムカタログに、新しいカタログがあります。

ここでは、PostgreSQL 9.4とPostgreSQL 12との違いについて、下表にまとめます。

表 5.5 PostgreSQL 9.4には存在しないシステムカタログ

カタログ名	用途
pg_init_privs	オブジェクトの初期権限
pg_partitioned_table	テーブルのパーティションキーについての情報
pg_policy	行単位セキュリティポリシー
pg_publication	論理レプリケーションのパブリケーション
pg_publication_rel	リレーションとパブリケーションの対応
pg_replication_origin	登録されたレプリケーション起点
pg_sequence	シーケンスについての情報
pg_statistic_ext	プランナ拡張統計情報（定義）
pg_statistic_ext_data	プランナ拡張統計情報（構築統計情報）
pg_subscription	論理レプリケーションのサブスクリプション
pg_subscription_rel	サブスクリプションについてのリレーションの状態
pg_transform	変換（データ型を手続き言語に変換）

また、同じカタログでも、機能追加などの影響から、列構成が異なるものがあります。ここでは、一例としてpg_catalog.pg_classについて下記に挙げます。

表 5.6 pg_catalog.pg_class比較

列名	9.4	12	説明
oid		○	行識別子
relname	○	○	
relnamespace	○	○	
reltype	○	○	
reloftype	○	○	
relowner	○	○	
relam	○	○	
relfilenode	○	○	
reltablespace	○	○	
relpages	○	○	
reltuples	○	○	
relallvisible	○	○	
reltoastrelid	○	○	
relhasindex	○	○	
relisshared	○	○	
relpersistence	○	○	
relkind	○	○	
relnatts	○	○	
relchecks	○	○	
relhasoids	○		リレーションの行毎にOIDを生成する場合は真
relhaspkey	○		テーブルにプライマリーキーが含まれている（あるいは過去含まれていた）場合に真
relhasrules	○	○	
relhastriggers	○	○	
relhassubclass	○	○	
relrowsecurity		○	行単位セキュリティが有効なテーブルでは真
relforcerowsecurity		○	行単位セキュリティが（有効にされているとして）テーブルの所有者にも適用されるなら真
relispopulated	○	○	
relreplident	○	○	
relfrozenxid	○		この値より以前のトランザクションIDはすべて、このテーブルで永遠（"凍結"）トランザクションIDに置き換えられます。
relminmxid	○		このテーブル内のトランザクションIDによって置換される前のすべてのマルチトランザクションID
relispartition		○	テーブルあるいはインデックスがパーティションなら真
relrewrite		○	テーブルの書き換えが必要なDDL操作中に書き込みが行われる新しいリレーションでは、これは元のリレーションのOIDを持ちます
relfrozenxid		○	この値より以前のトランザクションIDはすべて、このテーブルで永続的な（「凍結された」）トランザクションIDに置き換えられています
relminmxid		○	このテーブル内のトランザクションIDによって置換される前のすべてのマルチトランザクションID
relacl	○	○	
reloptions	○	○	
relpartbound		○	テーブルがパーティションの場合（relispartition参照）のパーティション境界の内部表現

5.2.5. システムビュー

システムビュー（統計情報ビュー含む）に、新しいビューがあります。

ここでは、PostgreSQL 9.4とPostgreSQL 12との違いについて、下表にまとめます。

表 5.7 PostgreSQL 9.4には存在しないシステムビュー

ビュー名	用途
pg_config	コンパイル時の設定パラメータ
pg_file_settings	設定ファイルの内容の要約
pg_hba_file_rules	クライアント認証の設定ファイルの内容の要約
pg_policies	ポリシー
pg_publication_tables	パブリケーションとそれに関連するテーブル
pg_replication_origin_status	レプリケーションの進捗を含めたレプリケーション起点に関する情報
pg_replication_slots	レプリケーションスロットの情報
pg_sequences	シーケンス
pg_stat_gssapi	接続（通常およびレプリケーション）あたり1行の形式で、接続に使われるGSSAPI認証と暗号化に関する情報を表示します
pg_stat_progress_cluster	CLUSTERまたはVACUUM FULLを実行している各バックエンドごと1行の形式で、現在の進捗を表示します
pg_stat_progress_create_index	CREATE INDEXまたはREINDEXを実行する各バックエンド実行ごとに1行の形式で、現在の進捗を表示します
pg_stat_progress_vacuum	VACUUMを実行している（自動バキュームワーカプロセスを含んだ）各バックエンドごとに1行の形式で、現在の進捗を表示します
pg_stat_ssl	接続（通常およびレプリケーション）あたり1行の形式で、接続に使われるSSLの情報を表示します
pg_stat_subscription	1つのサブスクリプションにつき少なくとも1行の形式で、サブスクリプションワーカに関する情報を表示します
pg_stat_wal_receiver	1行の形式で、受信サーバが接続したサーバからWALレシーバに関する統計情報を表示します
pg_stats_ext	プランナの拡張統計情報

システムカタログと同様に、同じビューでも、機能追加などの影響から、列構成が異なるものがあります。ここでは、一例としてpg_catalog.pg_tablesについて下記に挙げます。

表 5.8 pg_catalog.pg_tables比較

列名	9.4	12	説明
schemaname	○	○	
tablename	○	○	
tableowner	○	○	
tablespace	○	○	
hasindexes	○	○	
hasrules	○	○	
hastriggers	○	○	
rowsecurity		○	テーブルの行セキュリティが有効なら真

5.2.6. 機能・非機能

5.2.6.1. バックアップ・リカバリーおよびストリーミング

PostgreSQL 12で、recovery.confの設定がpostgresql.confへ移動されました。recovery.confは廃止されたため、データベースクラスタにrecovery.confというファイルが存在すると、サポートしていない旨のエラー出力し、PostgreSQLサーバが起動しません。

以下の場合などに、設計変更やドキュメントの修正といった対処が必要です。

- 移行元環境でストリーミングレプリケーション構成をとっていた場合
- バックアップ・リカバリーの手順書がある場合

なお、マニュアルでは、recovery.confのパラメーターについて書かれていた『第27章 リカバリーの設定』が削除され、『第19章 サーバの設定』

に移動しています。

5.2.6.2. 監視

サーバープロセス名やディレクトリ構成、システムカタログおよびシステムビューなどの変更が、PostgreSQL監視に影響があるかどうか、確認が必要です。

以下のような場合に、対処が必要となる場合があります。

- 独自のシェルで、サーバープロセスの存在を監視している
- ディレクトリの容量を監視している
- システムカタログやシステムビューから、PostgreSQLサーバーの性能情報を収集している

拡張機能のpg_monzを利用してPostgreSQLを監視している場合は、pg_monzに新バージョンがないかどうかあわせて確認しましょう。

5.3. 拡張機能

ここでは、主な機能拡張について、どのバージョンがどのPostgreSQLバージョンをサポートしているか、をまとめました。

以下、留意点です。

- 機能拡張のリリース日や、それがサポートしているPostgreSQLバージョンは、GitHubのリリース情報、README、Makefileなどから抜粋しています。
- 機能拡張のリリースされたすべてのバージョンを掲載しているわけではなく、省略している場合もあります。
- なお、本資料が、そのバージョンでの稼働を保証するものではありません。利用する機能拡張のドキュメントを確認し、テストするようにしてください。
 - マニュアルに記載がない場合、基本的には、メジャーバージョンアップ前に機能拡張をアンインストールし、メジャーバージョンアップ後に再度インストールします。

表 5.9 調査した主だった機能拡張

名称	概要
pg_dbms_stats	PostgreSQLの統計情報を固定する拡張機能。通常ANALYZEで採取する統計情報をダミー情報で固定化することで、予期しない実行計画の変更を防ぐことが可能。
pg_hint_plan	PostgreSQLにヒントを実装する拡張機能。ヒント句を使用してスキャンや結合の方式を指定することで、SQLの実行計画を制御することが可能。
pgBadger	PostgreSQLのログファイルを解析して、SQL実行状況などのHTMLレポートを生成。
pg_monz	ZabbixによるPostgreSQL監視のためのテンプレートを提供。
pg_repack	通常は排他ロックを必要とするテーブルやインデックスの再編成を、排他ロックをかけずに実行可能とする拡張機能。
pg_rman	PostgreSQLのバックアップ・リカバリ実行の簡易化、バックアップの世代管理など、バックアップ運用を補助するツール。
pg_statsinfo	PostgreSQLの稼働統計情報のスナップショットを定期的に収集・蓄積し、データベースの処理状況、性能傾向などの確認に利用可能。レポート出力機能も提供。
pg_bulkload	PostgreSQLに対して大量データの高速ロードを可能とするツール。

5.3.1. pg_dbms_stats

- https://github.com/oss-c-db/pg_dbms_stats
- 本書作成時点で、PostgreSQL 12対応版はリリースされていません。
- 専用のスキーマ、オブジェクトが作成されるため、データベースクラスタ移行時、対処が必要です。
- 下表の対応バージョンは、Makefileから抜粋しています。

表 5.10 pg_dbms_stats

バージョン	リリース日	PostgreSQLバージョン	ノート
1.3.11.PG11	2019年8月26日	9.3, 9.4, 9.5, 9.6, 10, 11	
1.3.11	2018年9月26日	9.3, 9.4, 9.5, 9.6, 10	
1.3.10	2018年4月5日	9.3, 9.4, 9.5, 9.6, 10	
1.3.9	2017年11月13日	9.2, 9.3, 9.4, 9.5, 9.6, 10	
1.3.8	2017年10月10日	9.2, 9.3, 9.4, 9.5, 9.6, 10	
1.3.7	2016年9月5日	9.1, 9.2, 9.3, 9.4, 9.5, 9.6	

5.3.2. pgBadger

- <https://github.com/darold/pgbadger>
- リリースノートなどに、PostgreSQLのバージョンに関する記述がほとんどありません。
- 11.2において、SQL文を100,000文字で切り捨てる仕様がデフォルトになっています。ただし、これはオプションで指定可能です。

表 5.11 pgBadger

バージョン	リリース	ノート
11.2	2020年3月11日	SQL文を100000文字で切り捨てる仕様がデフォルトに
11.1	2019年9月16日	
11.0	2019年6月25日	下位互換性なし
10.3	2019年2月14日	
10.2	2018年12月27日	
10.1	2018年9月12日	
10.0	2018年9月9日	
9.2	2017年7月28日	
9.1	2017年1月24日	
9.0	2016年9月2日	

5.3.3. pg_hint_plan

- https://github.com/ossc-db/pg_hint_plan
- リリースノートに非互換情報の記載はありません。
- 実行計画を制御するものであるため、PostgreSQL本体のメジャーバージョンアップにより、ヒント句を指定しなくても適切な実行計画を生成しないかなど、性能検証は必要です。
- PostgreSQLのメジャーバージョンごとにモジュールが用意されています。（例：pg_hint_plan-REL12_1_3_5）

表 5.12 pg_hint_plan

リリース	PostgreSQL 9.4	9.5	9.6	10	11	12	ノート
2020年2月 20日	1.1.9	1.1.9	1.2.6	1.3.5	1.3.5	1.3.5	PostgreSQL 12サポート
2019年1月 17日	1.1.8	1.1.8	1.2.5	1.3.3	1.3.4		
2018年11月 13日	1.1.7	1.1.7	1.2.4	1.3.2	1.3.2		PostgreSQL 11サポート
2018年6月8 日	1.1.6	1.1.6	1.2.4	1.3.1			
2017年10月 10日				1.3.0			PostgreSQL 10サポート
2017年7月 27日	1.1.5	1.1.5	1.2.2				

5.3.4. pg_monz

- <https://github.com/pg-monz>
- 本書作成時点で、PostgreSQL 12をサポートする旨の記述がありませんが、リリースノート記載の動作環境には、1.0リリース時から一貫してPostgreSQL 9.2以上と書かれています。
- リリースノートに非互換情報の記載はありません。

表 5.13 pg_monz

バージョン	リリース	Zabbixバージョン	PostgreSQLバージョン
2.2	2018年10月31日	2.0以上	9.2以上 (11をサポート)
2.1	2018年3月30日	2.0以上	9.2以上 (9.6および10をサポート)
2.0.1	2016年4月21日	2.0以上	9.2以上
2.0	2015年3月31日	2.0以上	9.2以上
1.0.1	2014年11月17日	2.0以上	9.2以上
1.0	2013年11月5日	2.0以上	9.2以上

5.3.5. pg_repack

- https://github.com/reorg/pg_repack
- PostgreSQL 9.1 ~ 12 をサポート
- リリースノートに非互換情報の記載はありません。
- ドキュメントに、古いバージョンからのアップグレード方法が記載されています。

表 5.14 pg_repack

バージョン	リリース	ノート
1.4.5	2019年10月5日	PostgreSQL 12をサポート
1.4.4	2018年10月18日	PostgreSQL 11をサポート
1.4.3	2018年5月20日	
1.4.2	2017年10月13日	PostgreSQL 10をサポート
1.4.1	2017年8月12日	
1.4.0.1	2017年4月26日	
1.4.0	2017年4月26日	PostgreSQL 9.6をサポート、9.1より前のサポートを削除
1.3.4	2016年5月7日	
1.3.3	2016年2月8日	PostgreSQL 9.5をサポート
1.3.2	2015年8月14日	
1.3.1	2015年3月11日	PostgreSQL 9.4をサポート

5.3.6. pg_rman

- https://github.com/osscc-db/pg_rman
- PostgreSQLのメジャーバージョンごとにモジュールが用意されています。(例: pg_rman-1.3.9-pg12.tar.gz)
- リリースノートに明確な推奨PostgreSQLバージョンが記載されていない場合は、提供されているモジュールから判断しています。
- いくつかのバージョン間で非互換や注意点があるため、バージョンアップ元からのリリースノートを確認する必要があります。
- PostgreSQL 12対応版では、recovery.confではなくpostgresql.confにPITR関連オプションを追加するなどの変更がされています。

表 5.15 pg_rman

バージョン	リリース	推奨PostgreSQLバージョン	ノート
1.3.9	2019年10月29日	9.4.24, 9.5.19, 9.6.15, 10.10, 11.5, 12.0	PostgreSQL 12をサポート
1.3.8	2019年1月22日	9.2.24, 9.3.25, 9.4.20, 9.5.15, 9.6.11, 10.6, 11.1	PostgreSQL 11をサポート
1.3.7	2018年9月27日		pg_rman 1.3.6以前の互換性なし
1.3.6	2018年1月30日	9.2.24, 9.3.20, 9.4.15, 9.5.10, 9.6.6, 10.1	PostgreSQL 10をサポート
1.3.5	2017年7月14日	9.3, 9.4, 9.5, 9.6	pg_rman 1.3.4以前を使用して取得したバックアップにGIN、BRIN、およびSP-GiSTインデックスのいずれかが含まれている場合に注意点あり
1.3.4	2017年4月24日	9.3, 9.4, 9.5, 9.6	
1.3.3	2016年10月13日	9.3, 9.4, 9.5, 9.6	PostgreSQL 9.6をサポート

5.3.7. pg_statsinfo

- <https://sourceforge.net/projects/pgstatsinfo/>
- pg_statsinfo 10.0以降は、単一のPostgreSQLバージョンのみをサポートしています。
- ドキュメントに、古いバージョンからのアップグレード方法が記載されています。

表 5.16 pg_statsinfo

バージョン	リリース	PostgreSQLバージョン	ノート
12.0	2019年10月5日	12	
11.0	2019年2月12日	11	
10.0	2018年2月1日	10	
3.3.0	2017年9月19日	9.2, 9.3, 9.4, 9.5, 9.6	
3.2.3	2017年1月30日	9.1, 9.2, 9.3, 9.4, 9.5, 9.6	
3.2.1	2016年2月26日	9.1, 9.2, 9.3, 9.4, 9.5	

5.3.8. pg_bulkload

- https://github.com/oss-c-db/pg_bulkload
- バージョン3.1.14までは、publicスキーマでオブジェクトが作成されていましたが、CVE-2018-1058対応のため、バージョン3.1.15からはpgbulkloadスキーマが作成され、そこにオブジェクトが作成されます。

表 5.17 pg_bulkload

バージョン	リリース	推奨PostgreSQLバージョン	ノート
3.1.16	2020年1月22日	9.3, 9.4, 9.5, 9.6, 10, 11, 12	PostgreSQL 12をサポート、アクセスメソッドがheapである表のみサポート
3.1.15	2019年1月21日	11.1, 10.6, 9.6.11, 9.5.15, 9.4.20, 9.3.25, 9.2.24	PostgreSQL 11をサポート、CVE-2018-1058対応
3.1.14	2017年11月7日	10.0, 9.6.5, 9.5.9, 9.4.14, 9.3.19, 9.2.23	PostgreSQL 10をサポート
3.1.13	2017年1月27日	9.6.1, 9.5.5, 9.4.10, 9.3.15, 9.2.19	
3.1.12	2016年11月16日	9.6.1, 9.5.5, 9.4.10, 9.3.15, 9.2.19	
3.1.11	2016年11月11日	9.6.1, 9.5.5, 9.4.10, 9.3.15, 9.2.19	
3.1.10	2016年9月29日	9.6.0, 9.5.4, 9.4.9, 9.3.14, 9.2.18	PostgreSQL 9.6をサポート
3.1.9	2016年1月27日	9.5.0, 9.4.5, 9.3.10, 9.2.14	PostgreSQL 9.5をサポート
3.1.8	2015年7月2日	9.4, 9.3, 9.2, 9.1, 9.0, 8.4, 8.3	

6. メジャーバージョンアップ検証

6.1. 概要

この章では、実機でメジャーバージョンアップを行った結果をまとめます。

なお、結果についてはログなどが長大になるため、別紙『Appendix. メジャーバージョンアップのログ』を参照ください。

6.2. 想定するストーリー

保守限界を迎えるPostgreSQL 9.4が稼働するサーバーから、PostgreSQL 12をインストールした新サーバーへ移行（メジャーバージョンアップ）をおこなう。

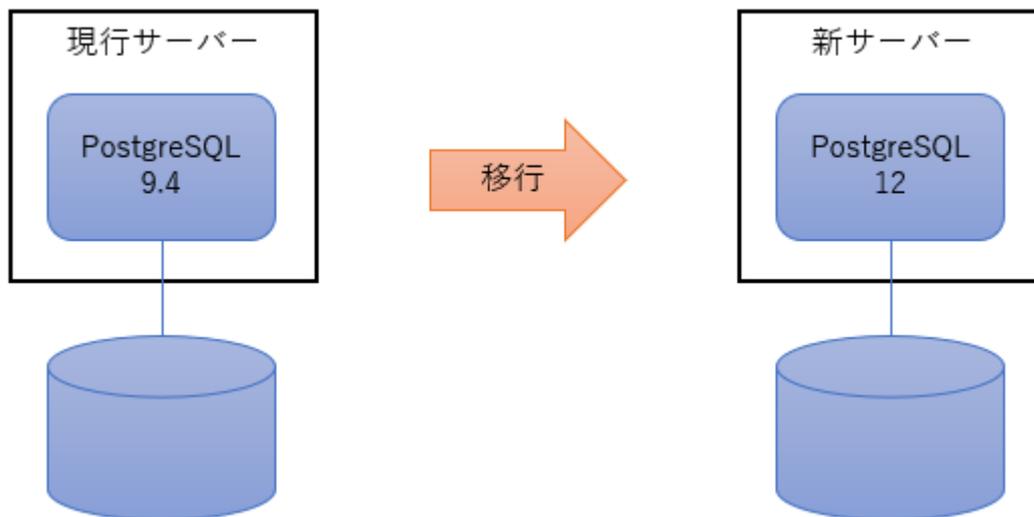


図 6.1 想定するストーリー

6.3. 検証パターン

表 6.1 検証パターン

No.	パターン	説明
1	pg_dump/pg_restoreを利用	ユーザーデータをpg_dumpでエクスポートし、pg_restoreでインポートする
2	pg_upgradeを利用	データベースクラスタをpg_upgradeのリンクモードでバージョンアップする

6.4. 検証環境

6.4.1. サーバー構成

Oracle VirtualBox 6.0にて、移行元、移行先のサーバーとして、2個のVMを作成。

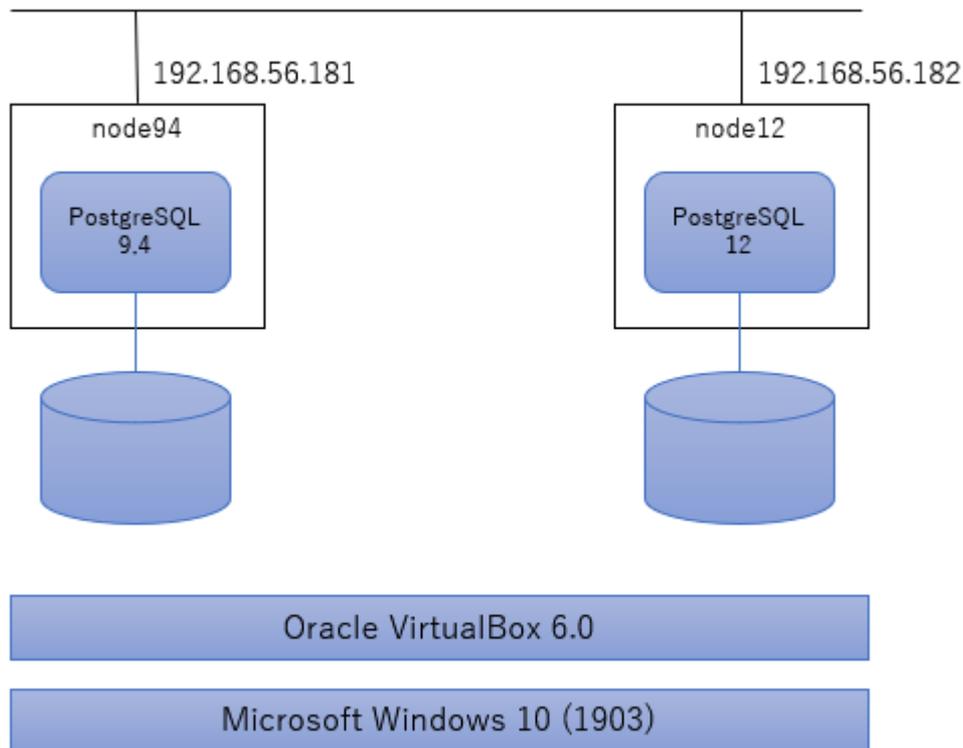


図 6.2 サーバー構成

表 6.2 検証用のサーバー構成

役割	Hostname	OS	M/W	CPU(vCore)	Mem	Disk
移行元	node94	CentOS 7.7	PostgreSQL 9.4.26	2	3GiB	50GiB
移行先	node12	CentOS 7.7	PostgreSQL 12.2	2	3GiB	50GiB

6.4.2. 機能拡張

利用する機能拡張は下記のとおり。

移行元バージョンは、2015年前後にリリースされたバージョンを採用しました。ただし、一部、makeに失敗したものは、yumで入手可能な最古バージョンを採用しました。

表 6.3 検証で利用する機能拡張

機能拡張	移行元バージョン	移行先バージョン
pg_dbms_stats	1.3.7	なし
pg_hint_plan	1.1.5	1.3.5
pgBadger	10.2	11.2
pg_repack	1.4.3	1.4.5
pg_rman	1.3.3	1.3.9
pg_statsinfo	3.2.1	12.0
pg_bulkload	3.1.8	3.1.16

6.5. 検証結果

別紙『Appendix. メジャーバージョンアップのログ』を参照ください。

7. 著者

(企業・団体名順)

版	所属企業・団体名	部署名	氏名
第1.0版 (2019年度WG2)	株式会社中電シーティーアイ	クラウド運営グループ	平林 義博
	株式会社中電シーティーアイ	クラウド運営グループ	森下 陽平