PGECons
PostgreSQL Enterprise Consortium

設計運用WG 2013年度WG3活動報告

―走り続けるPostgreSQLシステム構築のために―

PostgreSQL エンタープライズ・コンソーシアム
WG3 (設計運用WG)

アジェンダ

- WG3について
  - WG3設立の背景
  - WG3の位置づけ、活動概要
  - 2013年度の活動テーマ
  - 実施体制(14社)
- 2013年度活動報告
  - 企業システムに求められる非機能要求
  - DBMSに求められる要求(要件)
  - DB要件を実現するPostgreSQLの代表的なシステム構成
  - PostgreSQLシステム構成の分類
  - バックアップ／リカバリ、監視に求められる要件
  - 運用技術検証　基礎検証、高負荷下での可用性検証
    バックアップ／リカバリ検証、監視ケーススタディ
- エンディング　～WG3からのメッセージ～
  - エンタープライズなDB要件に応えられるか
  - 2013年度を振り返って
  - 2014年度活動予定

WG3(設計運用WG)設立の背景-PGEConsに寄せられたコメント-

- **運用面の注意**なども知りたい。
- PostgreSQL普及の課題は、移行ではなく、**ユーザーを納得させられる信頼性と可用性の情報**だと思う。そのあたりでOracleとの比較や代替手段等も今後期待します。
- **HA構成の案**について、さまざまな視点からの評価をいただければと思いました。
- **障害時の動作検証の結果**が欲しかった。
- 移行というテーマも興味があるが、それ以前に**PostgreSQLを選定して安心だと言える**ようなテーマを検証してほしい。
- 既にpostgreSQLを導入しているため、設計運用WG(WG3)の今後の活動に期待している。
- WG3においてストリーミングレプリケーション機能をpacemaker+Redhatの連携が可能か知りたい　等

※2012年度成果報告会アンケートより抜粋

運用性、可用性に対するニーズが大

WG3の位置づけ、活動概要

- 技術部会で設定された課題に具体的に取り組む実働部隊の一つで主に設計・運用面に関する課題をテーマとする
  - 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適用に向けて、安定稼動のための技術ノウハウを提供

PGEConsにおける課題領域

性能	性能評価手法、性能向上手法、チューニングなど
可用性	高可用クラスタ、BCP
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
セキュリティ	監査
互換性	データ、スキーマ、SQL、ストアドプロシージャの互換性
接続性	他ソフトウェアとの連携

WG1テーマ

WG2テーマ

高可用、バックアップ、監視に着目して活動開始

2013年度の活動テーマ

- 可用性
  - PostgreSQLの代表的なシステム構成(シングル・HA・レプリケーション)の概要・特徴・考慮事項を調査し、適用領域を整理
  - これらの実機検証
- バックアップ
  - システム構成ごとにバックアップ手法を洗い出し、バックアップ要件の対応度合いを整理
  - 実機検証による各システム構成ごとの運用例
- 監視
  - DBサーバとDBの死活・性能監視に必要な情報の洗い出し
  - 収集した情報からの分析および対処方針を各システム構成ごとに整理
  - 実機検証による監視ケーススタディと対処法の効果

PostgreSQLの代表的構成に対して3つの視点で整理

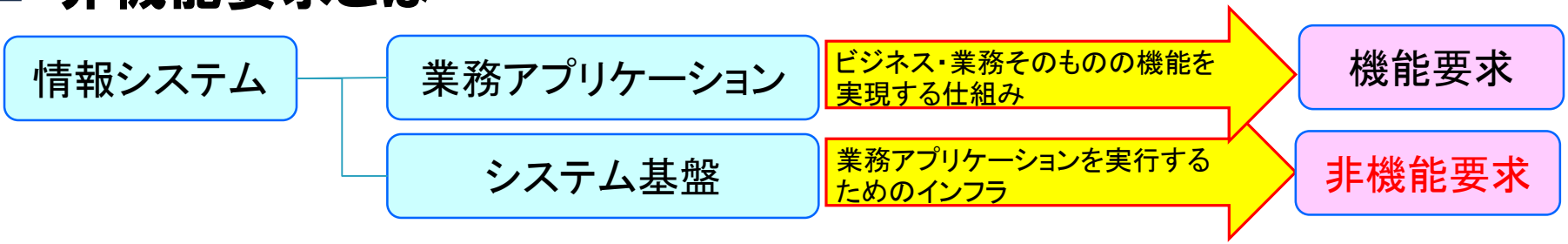
実施体制(14社)

- 株式会社アイ・アイ・エム
- 株式会社アシスト（主査）
- 株式会社インフォメーションクリエーティブ
- SRA OSS, Inc. 日本支社
- NTTソフトウェア株式会社（副査）
- クオリカ株式会社
- TIS株式会社（副査）
- 日本電気株式会社
- 日本電信電話株式会社（副査）
- 日本ヒューレット・パッカード株式会社
- 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
- フューチャーアーキテクト株式会社　　※企業名順

志士14社で課題に挑みます

企業システムに求められる非機能要求(１)

■ 非機能要求とは



大項目	概要	要求例
可用性	システムサービスを継続的に利用可能とするための要求	・運用スケジュール(稼働時間・停止予定など) ・障害、災害時における稼働目標
性能・拡張性	システムの性能および将来のシステム拡張に対する要求	・業務量および今後の増加見積 ・システム化対象業務の特性(ピーク時、通常時、縮退時)
運用・保守性	システム運用と保守サービスに関する要求	・運用中に求められるシステム稼働レベル ・問題発生時の対応レベル
移行性	現行システム資産の移行に関する要求	・新システムへの移行期間および移行方法 ・移行対象資産の種類および移行量
セキュリティ	情報システムの安全性の確保に関する要求	・利用制限 ・不正アクセスの防止
システム環境・エコロジー	システムの設置環境やエコロジーに関する要求	・耐震/免震、重量/空間、温度/湿度、騒音などシステム環境に関する事項 ・CO_2排出量や消費エネルギーなどエコロジーに関する事項

「可用性」と「運用・保守」に着目！

※独立行政法人　情報処理推進機構「非機能要求グレード」より

企業システムに求められる非機能要求(２)

■ 可用性とは　－トラブルが発生しても何事もなかったかのようにサービスを継続－

項目	概要
継続性	システムが正常稼働している状態を表す「運用スケジュール」や「業務継続性：対象業務範囲」、故障発生時の復旧目標である「目標復旧水準」や「業務継続性：サービス切替時間」への要求
耐障害性	故障への耐性に関してシステムの構成要素の観点から整理 「サーバ」の冗長化や「端末」、「ネットワーク機器」、「データ」のバックアップなどの要素
災害対策	情報システムを設置した施設や地域全体が使用不能になるような大規模災害に対する「システム」、「外部保管データ」、「付帯設備」への要求
回復性	故障や災害発生後のシステム機能回復と必要なデータ復旧に対する要求

■ 運用・保守性とは　－運用方法や保守サービスの項目(軽視されがちで後々トラブル発生)－

項目	概要
通常運用	運用時間や時刻同期といった項目に加え、「バックアップ」や「運用監視」といった可用性の実現に重要な項目
保守運用	計画停止、パッチ適用、活性保守等の項目
障害時運用	復旧作業、復旧自動化の範囲、異常検知時の対応、交換用部材の確保、等の項目

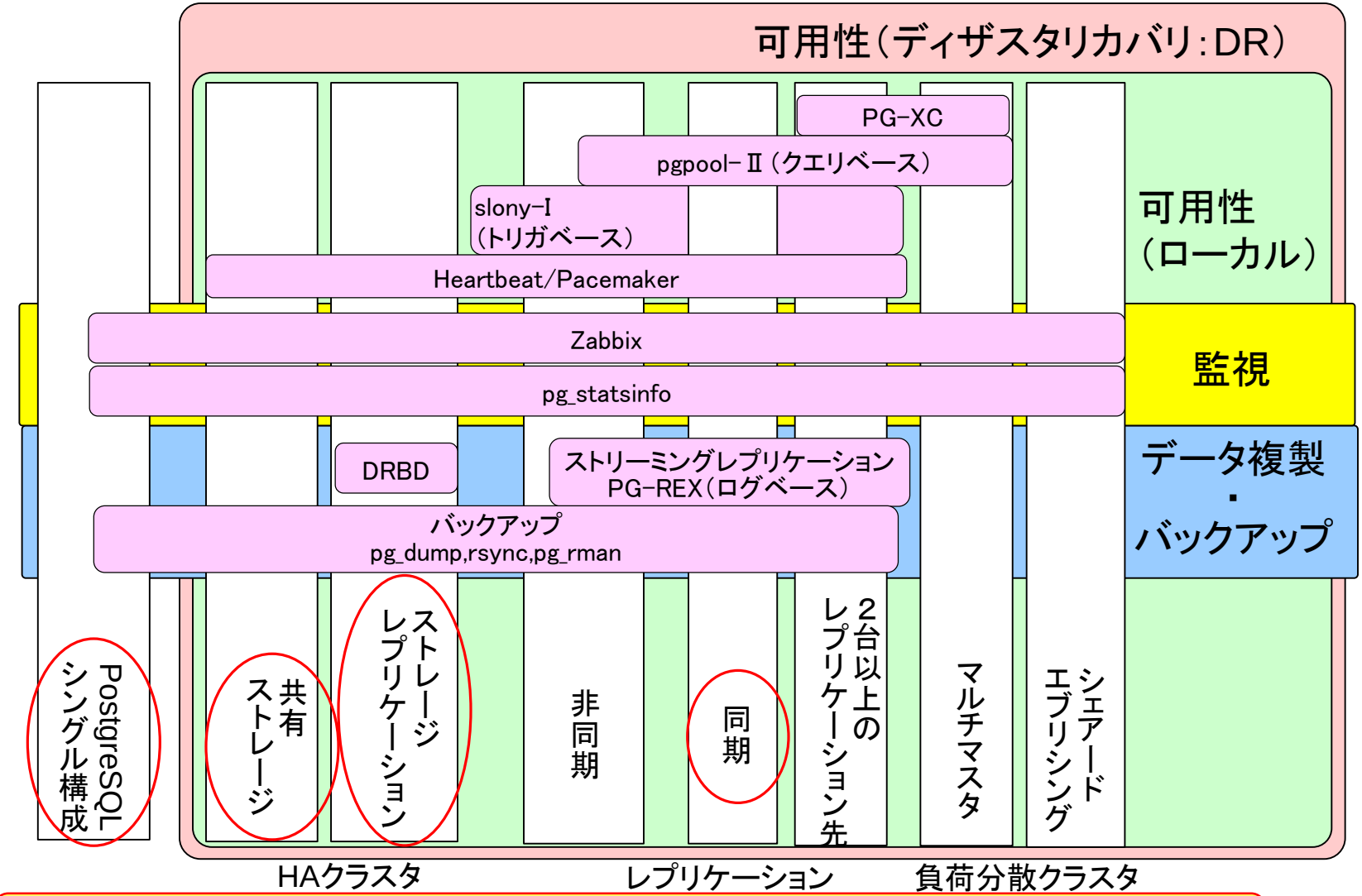
ポイントは「可用性(冗長化)」と「バックアップ」「監視」

DBMSに求められる要求(要件)

- 可用性
  - 「冗長化」で安心してはダメ！
    - 単一ノード時の環境設定、運用手順、運用スキルは、可用性に先立ち重要！
    - 稼働・コストをかけて切替えても、各種設定を含めたベースがダメならトラブル再発
  - DBサーバやDB構成ファイル、WAL格納ディスクの冗長化、オンライン再編成、更新・削除による不要データ増を押さえるVACUUM、等
- バックアップ
  - 冗長化やレプリケーション、高信頼なハード活用でバックアップは不要？
    - ソフトバグやオペレーションミスによるデータ削除や論理矛盾の復旧はバックアップが必須！
  - システム稼働中のオンラインバックアップ、遠隔地バックアップ、任意の時点へのリカバリ(Point In Time Recovery)、等
- 監視
  - 監視間隔と監視情報をきめ細かくチェックすれば大丈夫？
    - 期待通りに動いている正常状態を把握し、「変化」を捉えることが重要！
  - 死活監視、エラー監視、リソース監視、パフォーマンス監視、等

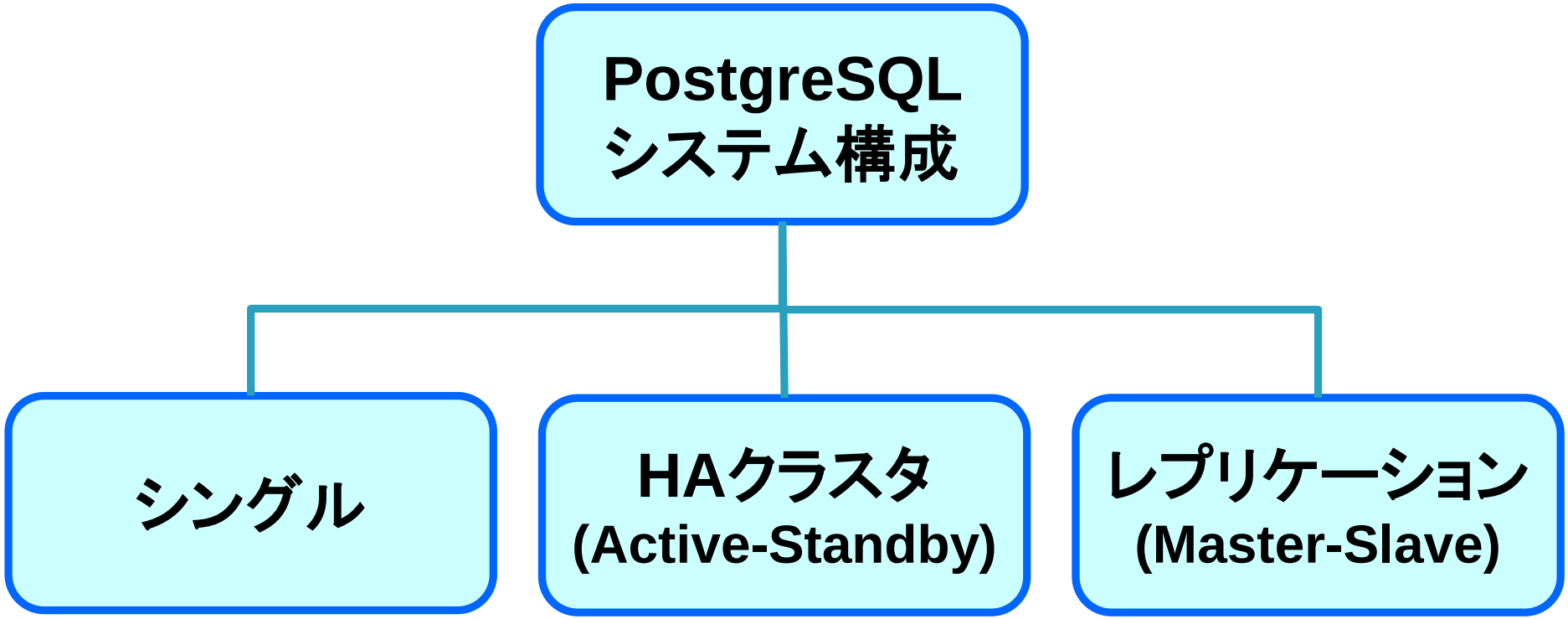
DBMSに特徴的な問題への考慮が大事

DB要件を実現するPostgreSQLの代表的なシステム構成

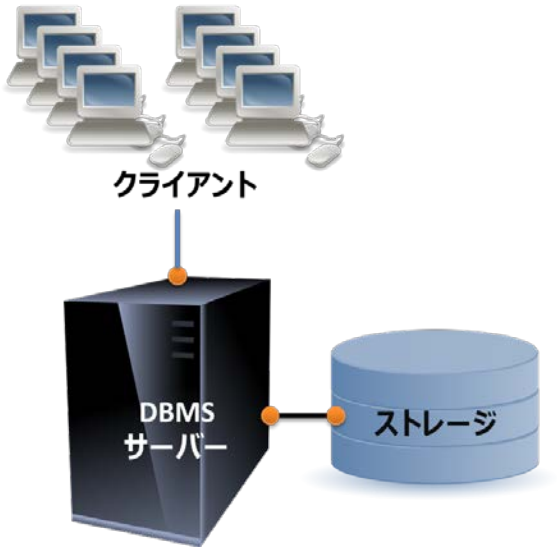


PostgreSQLと様々なツールの組合せで選択肢が拡大

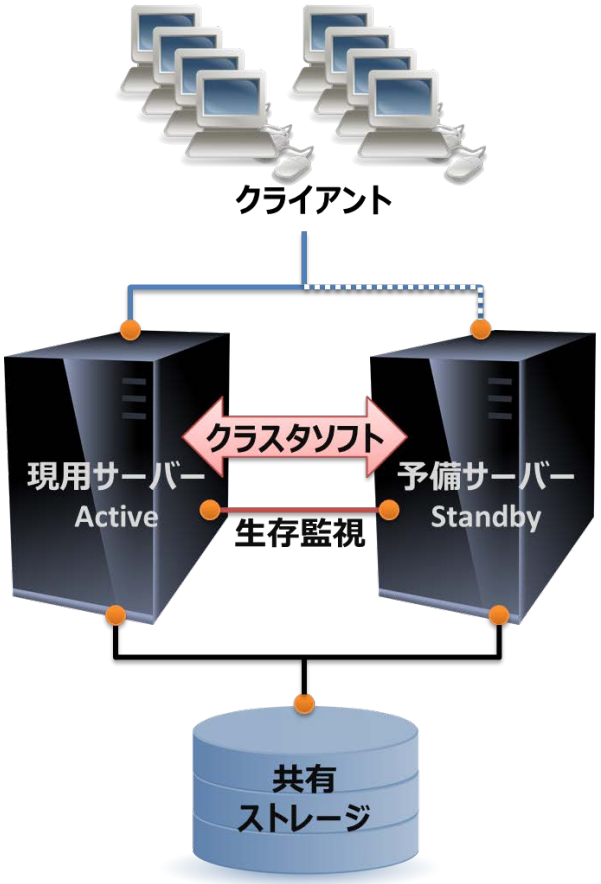
PostgreSQLシステム構成の分類



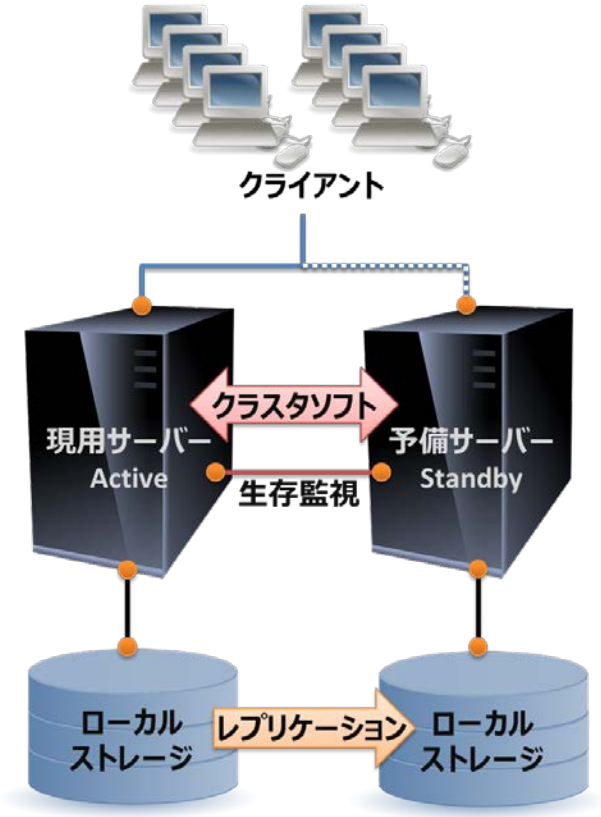
シングル構成／HAクラスタ構成図



シングル構成



共有ストレージ構成



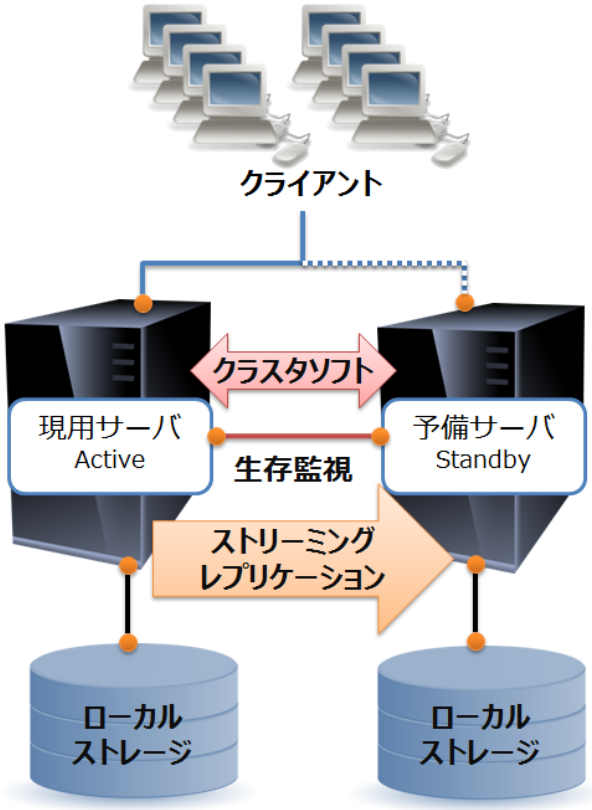
ストレージレプリケーション構成

シングル構成／HAクラスタ構成の可用性

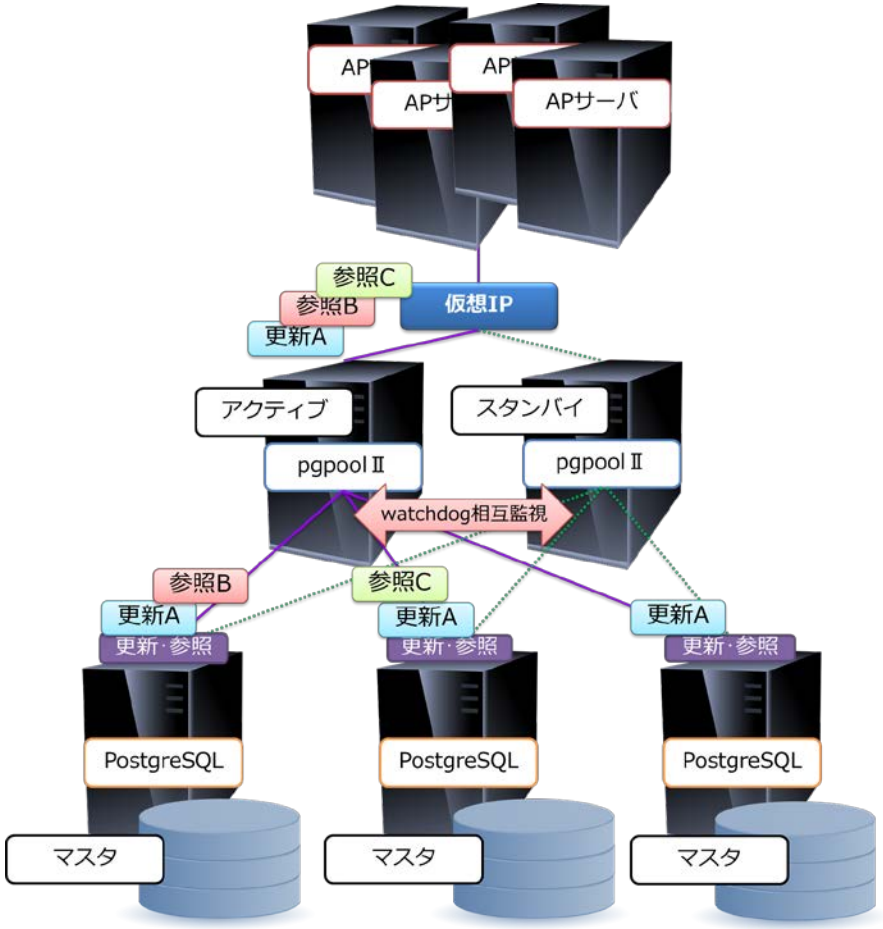
	シングル	HAクラスタ	
		共有ストレージ	ストレージレプリケーション
データ同期性	なし	なし （同期不要）	あり （ブロック単位で同期・非同期）
障害発生時のサービス継続性	なし （サービス停止）	低 （共有ストレージがSPOF、 アクティブサーバ障害のとき影響有）	中 （アクティブサーバ障害のとき影響有）
復旧時の運用性	なし （復旧中停止） ⇒WALファイル冗長化、 バックアップは必須	中 （スタンバイサーバとアクティブサーバを切り替えるとき影響有）	中 （スタンバイサーバとアクティブサーバを切り替えるとき影響有）
拡張性	なし	なし （PostgreSQLサーバの追加可能）	なし （最大4台までレプリケーション可能）
コスト	必要最低限	高信頼性のストレージが求められるため高価になりがち	高価なH/Wは不要だが、スタンバイサーバは使用不可のためリソースの利用効率は低い
オーバーヘッド	発生しない	発生しない （同期不要）	発生する （同期方式に依存）

これまでの主流は共有ストレージ、これからは「レプリケーション」に注目 ！

レプリケーション構成図(１)

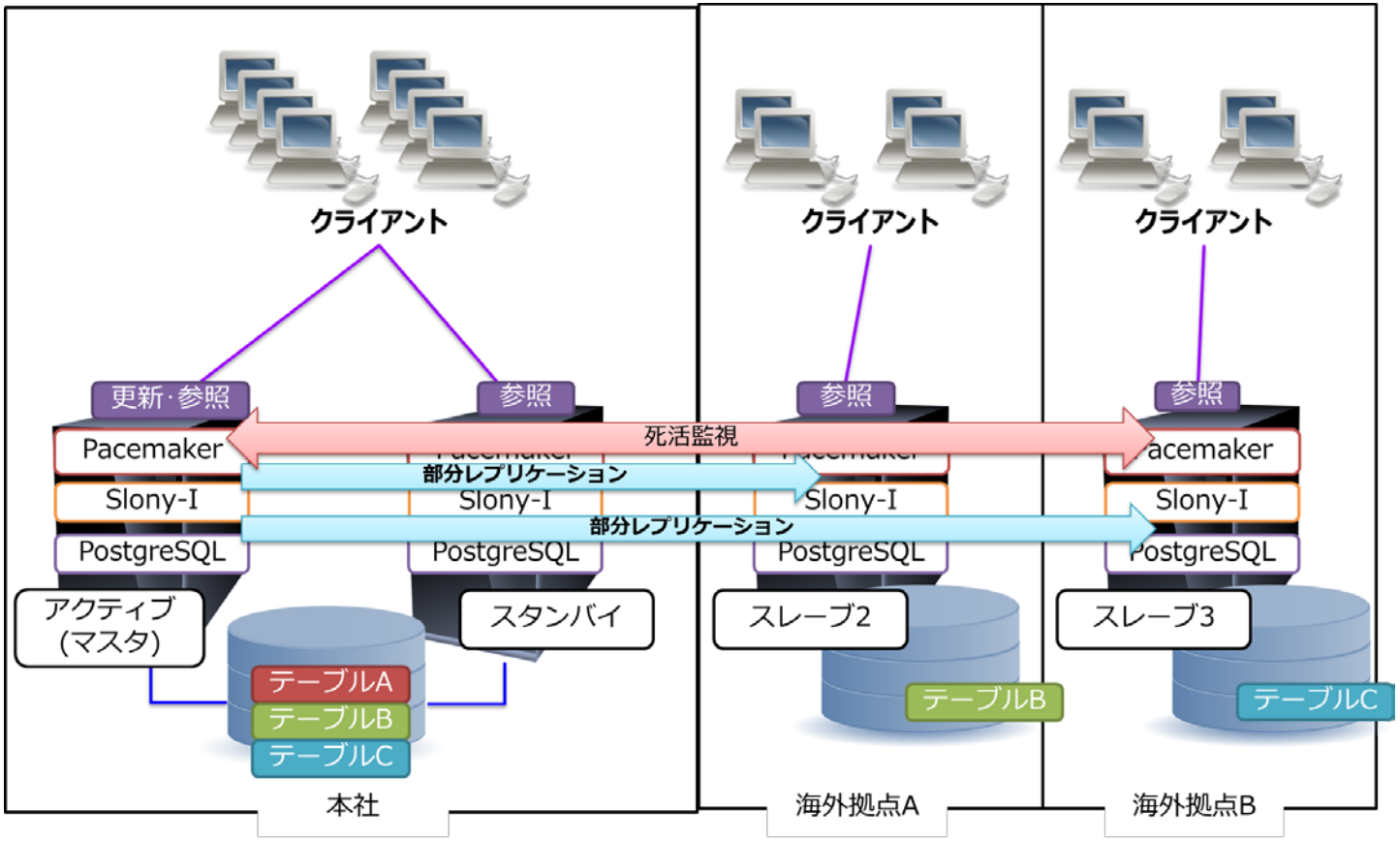


ストリーミング
レプリケーション構成



pgpool-II(レプリケーション)構成

レプリケーション構成図（２）



Slony-I構成

レプリケーション構成の可用性

	レプリケーション		
	ストリーミングレプリケーション（PG標準機能）	pgpool-II (レプリケーションモード)	Slony-I
データ同期性	あり（操作単位のWALを同期・非同期）	あり（SQL単位で同期）	あり（トリガーベースで非同期）
障害発生時のサービス継続性	中（マスタサーバ、同期スレーブサーバ障害のとき影響有）	高（どのサーバに障害が発生しても影響なし）	中（マスタサーバ障害のとき影響有）
復旧時の運用性	中（スレーブサーバとマスタサーバを切り替えるとき影響有）	中（リカバリ中にDBに変更が加えられると、その変更を反映するとき影響有）	高（遅延が発生する可能性があるが、切断されない）
拡張性	あり（スレーブサーバも参照可能）	あり（全サーバがマスタとして参照・更新が可能）	あり（スレーブサーバも参照可能）
コスト	高価なH/Wは不要。スレーブサーバが使えるため利用効率が高い	高価なH/Wは不要。全サーバが参照・更新可能なので利用効率が極めて高い	高価なH/Wは不要。スレーブサーバが使えるため利用効率が高い
オーバーヘッド	発生する（同期方式に依存）	発生する	発生する

ストリーミングレプリケーションとpgpool-IIの組合せに着目して検証を実施

バックアップ／リカバリに求められる要件

- バックアップ
  - 冗長化やレプリケーション、高信頼なハード活用でバックアップは不要？
    - ソフトバグやオペレーションミスによるデータ削除や論理矛盾の復旧はバックアップが必須！
  - システム稼働中のオンラインバックアップ、遠隔地バックアップ、任意の時点へのリカバリ（Point In Time Recovery）、等

- 非機能要求を整理し適切なバックアップ／リカバリ方式を採用することが重要
  - バックアップ／リカバリに関する代表的な非機能要求
    - 障害時にどの時点までデータを戻す必要があるか（RPO）
    - 復旧にどれくらい時間をかけられるか（RTO）
  - バックアップ／リカバリの特徴
    - 業務システムへの影響度合い
    - バックアップ／リカバリの単位
    - バックアップ／リカバリにかかる時間
    - オペレーションミスによるデータ消失からも復旧可能か

主なバックアップ／リカバリ方式(1/2)

	オフライン バックアップ	オンライン バックアップ (論理)	オンラインバックアップ(物理)		レプリケーション	
			OSレベルの ファイルコピー	ストレージ ローカルコピー	非同期	同期
バックアップ概要	データベースを停止し、cp,rsyncなどを用いたファイルコピー	pg_dumpコマンド等を使用	pg_start_backupとcp,rsyncなどの組み合わせ、またはpg_basebackupを使用	I/Oの静止点でストレージの機能を用いてデータを複製	ストリーミングレプリケーション構成のスタンバイをバックアップとみなす	
業務への影響度	バックアップ中は業務停止	バックアップ中はストレージ負荷が高まる	バックアップ中はストレージ負荷が高まる	大きな影響なし	大きな影響なし	性能面での影響大
バックアップ最小単位	データベースクラスタ	テーブル	データベースクラスタ	データベースクラスタ	データベースクラスタ	
バックアップ取得時間	ファイルコピー時間と同等	ファイルコピーと比較して長時間(検証では約2倍)	ほぼファイルコピー時間と同等(※1)	短時間(数秒～数分)	ー	
リカバリ時間(RTO)	ファイルコピー時間と同等	ファイルコピーと比較して長時間(検証では約5～10倍)	ファイルコピー時間＋WALのロールフォワード時間	短時間(数秒～数分)＋WALのロールフォワード時間	ー	

※1 pg_basebackupを使用した場合、検証ではデータベースサーバ上のファイルコピー(cpなど)よりは処理時間がかかるが、リモートコピー(scp)よりは短時間でバックアップ可能であった。

サービス影響が少ないオンラインバックアップ(物理:OSレベルのファイルコピー)が主流

主なバックアップ／リカバリ方式（2/2）

	オフライン バックアップ	オンライン バックアップ（論理）	オンラインバックアップ（物理）		レプリケーション	
			OSレベルの ファイルコピー	ストレージ ローカルコピー	非同期	同期
障害時にどの時点までデータを復旧できるか（RPO）	バックアップ取得時点（※2）	バックアップ取得時点	障害発生時点	障害発生時点	障害発生時点（※3）	障害発生時点
オペレーションミスによるデータ障害への対応	基本的に不可	基本的に不可	PITRにより可	PITRにより可	不可	
まとめ・注意点	• データベースを停止できる場合に限られる • ストレージローカルコピーでバックアップすることも可	• テーブル単位、データベース単位にバックアップすることが可能 • 物理バックアップよりも処理時間が長くなる	• 可用性（特に継続性）要件のレベルのシステムで推奨 • 不要となったアーカイブされたWALを定期的に削除する必要あり	• バックアップ／リカバリを短時間で実行する要件がある場合に使用する • I/Oの静止点でバックアップを実施する • アーカイブされたWALについては左記と同様	• 非同期レプリケーションでは、マスタ障害時にコミット済データが欠損する可能性がある • 他のバックアップ方式と併用することを推奨	

※2 バックアップ取得後、障害発生時点までのWALファイルが全て残っている場合、障害発生時点までの復旧が可能
※3 マスタサーバに障害が発生し、スタンバイにレプリケーションされていないWALデータがある場合、マスタ上のWALファイルを手動でスタンバイにコピーするなどの対応が必要になる。障害でマスタ上のWALファイルが失われた場合、レプリケーションされていないデータは復旧できない。

レプリケーションは便利だが異常データ混入時にバックアップ側も汚染されるため注意！」

監視についてDBMSに求められる要求（要件）

- 監視
  - 監視間隔と監視情報をきめ細かくチェックすれば大丈夫？
    - 期待通りに動いている**正常状態を把握**し、「変化」を捉えることが重要！
  - 死活監視、エラー監視、リソース監視、パフォーマンス監視、等

- 監視の目的に応じて要求を整理すること
  - 可用性を重視
    - 死活監視、エラー監視、リソース監視などサービス継続に大きな影響を与える項目を重点的かつ頻繁に監視。不要な情報は即削除する。
  - 運用保守性を重視
    - パフォーマンス監視などサービス維持に大きな影響を与える項目を重点的かつ定期的に監視。必要に応じて、情報を集約して改善策を講じる。

これらをバランスよく整理し、
必要な情報を必要な形式で
必要な期間監視し、「変化」をとらえること！

監視要求に応える

- サーバ/PostgreSQL、両方の監視を行いましょう
  - OSのコマンドやPostgreSQLのコマンド/関数/システムビューで必要な情報はそろう

観点	監視	頻度	ポイント
可用性向上	サーバ 死活監視	数秒間隔	OSのコマンド、ログを用いて正常/異常を把握する
	PostgreSQL 死活監視		PostgreSQLのコマンドやログ、システムビューを用いて正常/異常を把握する
運用保守性向上	サーバ 性能監視	数分間隔	OSのコマンド、ログを用いて性能情報を収集する
	PostgreSQL 性能監視		PostgreSQLの関数やログ、システムビューを用いて性能情報を収集する

可用性と運用保守、2つの視点で
サーバとPostgreSQLの監視が重要

運用技術検証
～可用性を担保する機能の基礎検証～

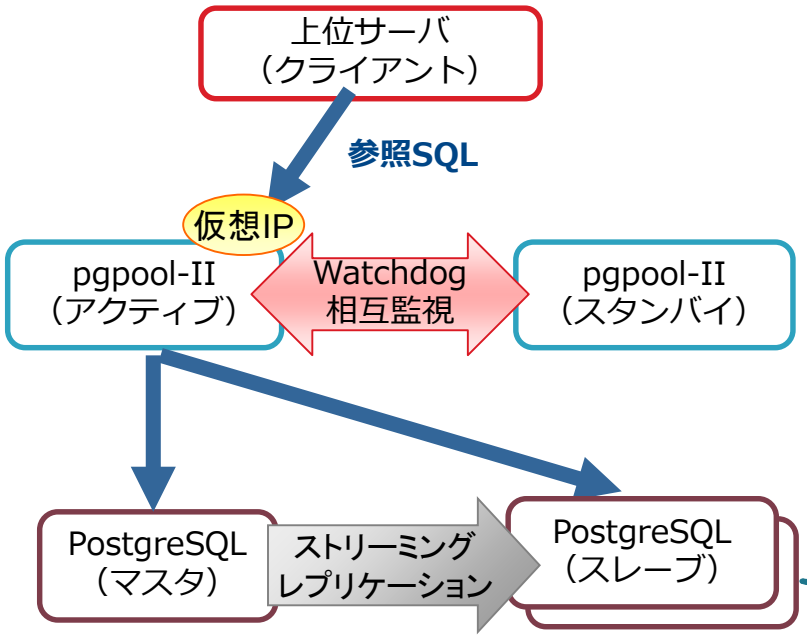
検証概要

PostgreSQL のストリーミングレプリケーション
＋ pgpool-II の自動フェイルオーバ機能
& watchdog 機能による pgpool-II 自体の冗長化

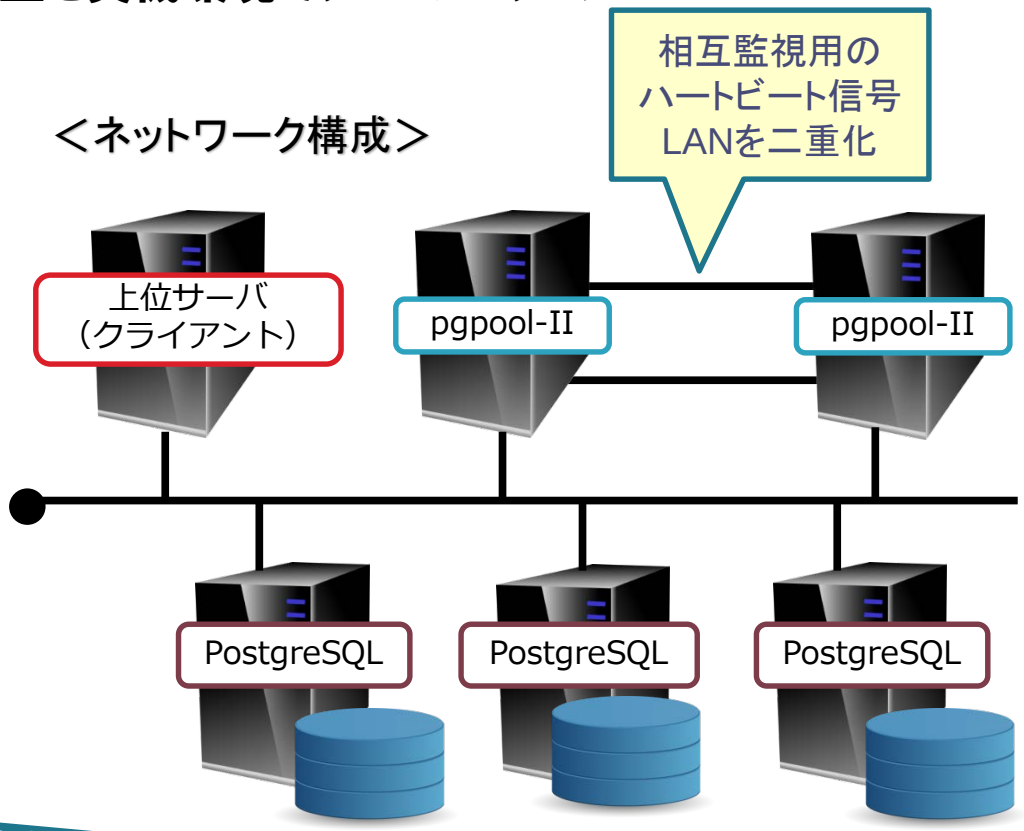
- **目的**
  - PostgreSQL + pgpool-II 構成の高可用性性能を評価
  - 実際の運用で起きうる障害発生を実機環境でシミュレーション
- **検証環境**

使用バージョン
- PostgreSQL 9.3.1
- pgpool-II 3.3.2

＜システム構成＞



＜ネットワーク構成＞



検証中に3台目のサーバをスレーブとして追加

検証対象機能

- PostgreSQL の障害検出と自動フェイルオーバ
- pgpool-II の障害検出と仮想 IP アドレスの自動付け替え
- 障害が発生した PostgreSQL のオンラインリカバリ
- スレーブサーバの動的追加、および削除

検証ケース(１)

- PostgreSQL の障害ケース
  - 検証したシナリオは以下の3通り
    - サーバの電源ダウン：　電源ボタンの長押しによる強制電源断
    - プロセス停止：　pg_ctl stop コマンドを実行
    - ネットワーク障害：　PostgreSQL サーバから LAN ケーブルを抜く
  - 障害が発生したサーバは pgpool-II から切り離される
    - 全ての子プロセスが切断＆再起動されるため、**接続中のセッションは一旦切断される**
      （切断時間 5～10秒）
  - マスタに障害が発生した場合は、**自動的にスレーブがマスタに昇格**する
  - 障害が発生したサーバは、**オンラインリカバリ**によりスレーブとして復旧させる
    - 新しいセッションが開始されたときに
      子プロセスが再起動されるので、
      接続中のセッションに影響は無し

PostgreSQL のトラブルに対し、サービスの中断を最小限に抑えられることを確認

検証ケース(２)

- pgpool-II の障害ケース
  - 以下のシナリオで障害を発生させる
    - 本体プロセス異常停止:kill コマンドで pgpool-II のプロセスを強制終了
    - ネットワーク障害: pgpool-II サーバからサービス LAN のケーブルを抜く
  - アクティブ pgpool-II に障害発生の場合、現スタンバイが新アクティブに昇格
    ⇒ 仮想 IP アドレスの付け替えが発生
    - その際には、接続中のセッションは一旦切断される （切断時間 5～60秒）
  - 障害が発生した pgpool-II は再起動により自動的にスタンバイとして復旧
  - ハートビート LAN に障害が発生した場合
    - ２本のうち１本が切断されても問題はない
    - ただし、２本とも切断された場合には、両方の pgpool-II がアクティブとなってしまう
      ⇒ ネットワーク上に同じ仮想 IP アドレスが２つ存在する状態となる
    - 片方の pgpool-II を再起動によりスタンバイとして復旧させる

pgpool-II のトラブルに対しても watchdog 機能を用いた冗長化構成によりサービスが継続できることを確認

検証ケース(３)

- スレーブサーバの追加
  - 追加するサーバの情報を設定ファイルに追記し、両 pgpool-II に読み込ませた後に、オンラインリカバリを実行
    - この際には、
      接続中のセッションに
      影響は無し

- スレーブサーバの削除
  - 削除対象の PostgreSQL を停止し、設定ファイルからサーバ情報を削除した後に、両 pgpool-II を再起動する
    - この際には、接続中のセッションは切断される

pgpool-II のバックエンドへの PostgreSQL サーバの追加および削除が容易に行えることを確認

可用性を担保する機能の基礎検証 –まとめ–

PostgreSQLのストリーミングレプリケーションと pgpool-II の機能を組み合わせた高可用性構成の実機検証

- PostgreSQL 障害時
  - 自動で障害を検出し、障害が発生したサーバの切り離し、およびマスタ・スレーブの切り替えを行う（自動フェールオーバ）
  - 接続中のセッションに影響を与えずにサーバ復旧が可能（オンラインリカバリ）
    - ⇒ サービス停止時間の短縮
- pgpool-II 障害時
  - 仮想 IP アドレスの自動切り替えによるサービス継続（watchdog 機能）
    - ⇒ pgpool-II が単一障害点とならない
- スレーブサーバの追加および削除
  - 接続中のセッションに影響を与えずにサーバを追加可能
    - ⇒ 参照負荷分散性能のスケールアウトが可能

高可用性システムを構築・運用する手段としての高い有用性が示された

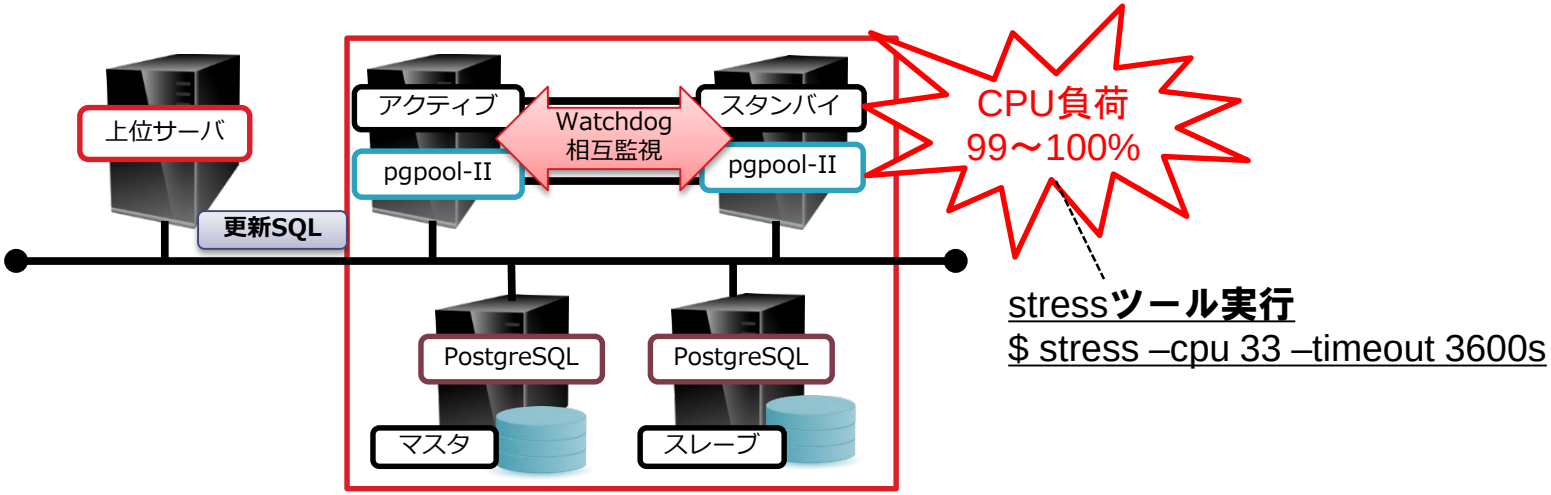
WG3 活動風景(その1)

- 実機検証風景

運用技術検証
～高負荷下での可用性検証～

検証目的と環境

- 目的
  - エンタープライズ領域では、大規模データ処理による高負荷状態でも正常に運用できることが重要
  - PostgreSQLサーバ、pgpool-IIサーバともにCPU負荷が高い状況での運用技術検証を実施
- 検証マシン構成
  - 2台のPostgreSQLによるストリーミングレプリケーションと、pgpool-IIの自動フェイルオーバ機能を組み合わせた高可用構成



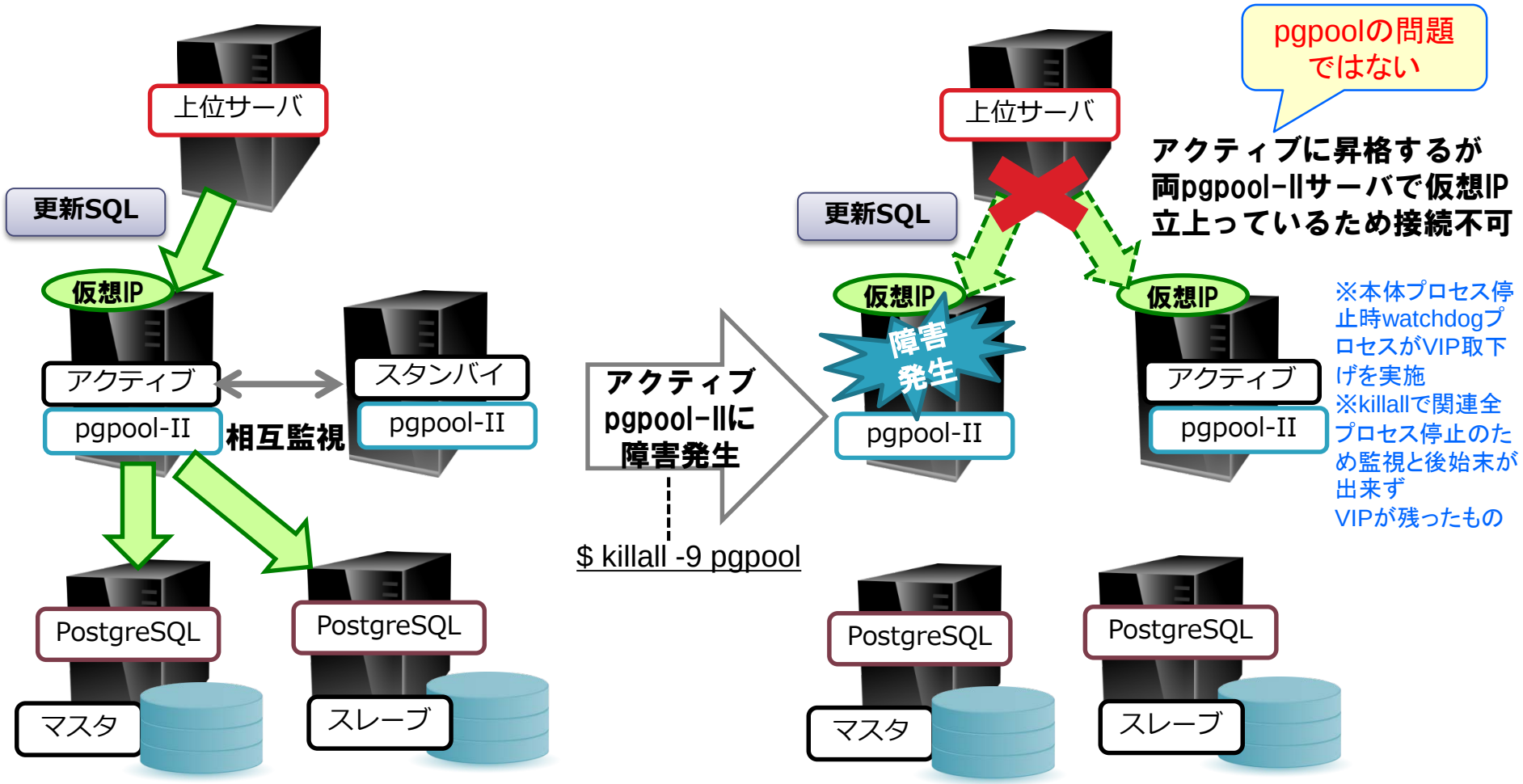
検証ケース・方法・結果

高負荷状況でINSERT実行中に、以下4ケースの挙動を確認

	検証ケース	検証方法	検証結果
1	PostgreSQL プロセス障害	pg_ctlコマンドでPostgreSQL プロセス停止	● pgpool-IIがPostgreSQLのダウンを検知しフェイルオーバ実施 ● 切り替え時間（※）：7秒 ※切り替え時間＝PGダウン検知～クエリ処理再開
2	PostgreSQL プロセス復帰	ケース1からpcp_recoveryコマンドでオンラインリカバリ実行	● 2秒でリカバリ完了、接続中のセッションに影響なし
3	pgpool-II プロセス障害	killallコマンドでpgpool-IIの全プロセス終了	● pgpool-IIが親プロセスの異常を検知しフェイルオーバ実施 ● セッション復帰不可
4	pgpool-II サーバ障害	pgpoolコマンドでアクティブpgpool-IIサーバを停止	● pgpool-IIが親プロセスの異常を検知しフェイルオーバ実施 ● 切り替え時間：50秒

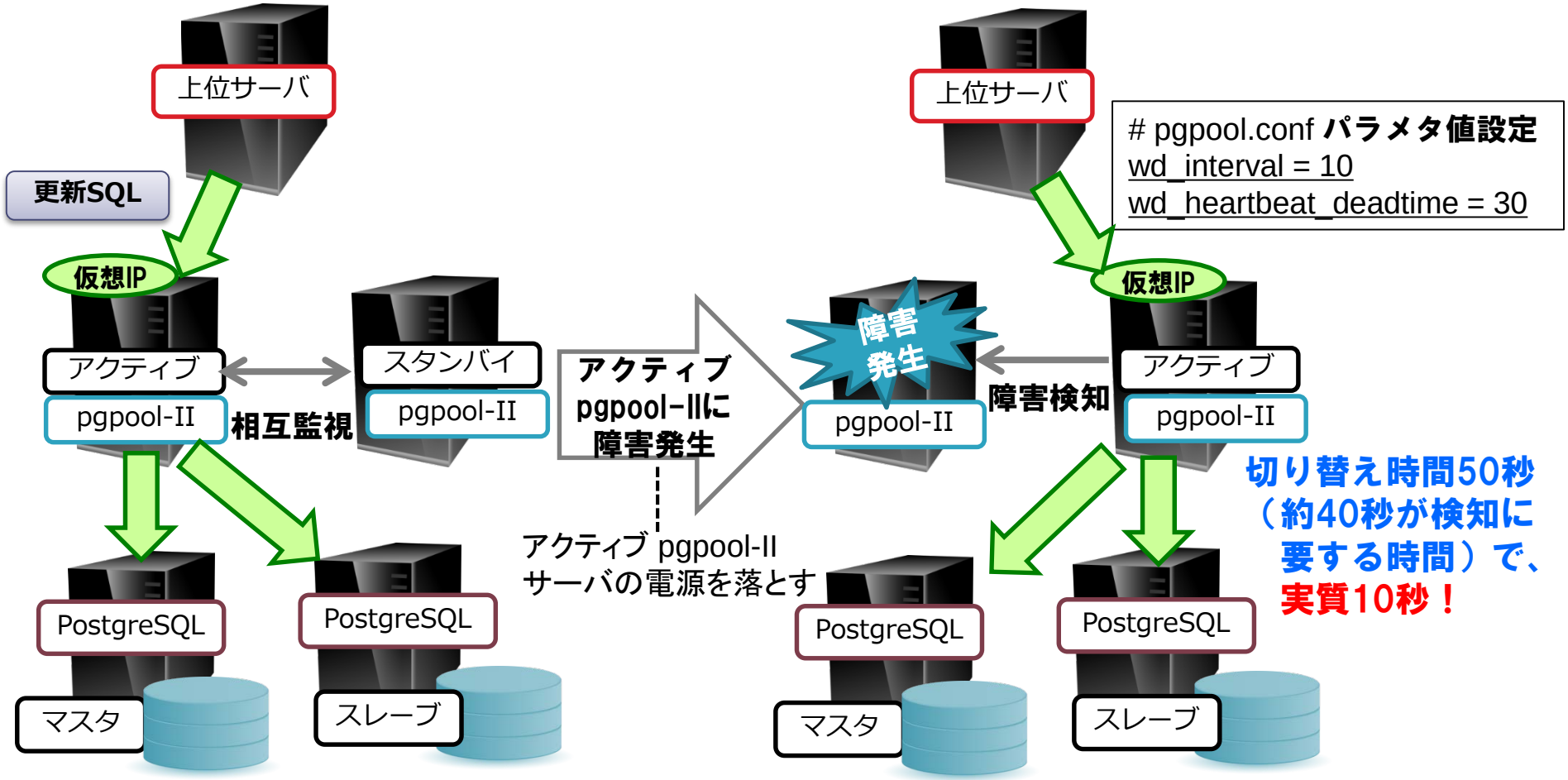
高負荷下でのPostgreSQL故障の短時間復旧動作を確認、pgpool-IIについては後述

ケース3 pgpool-IIプロセス障害：killallコマンド実行によるセッション復帰不可



killallコマンド実施で仮想IPアドレスを停止させる処理が起動せずセッション復帰不可
⇒pgpool-IIの仕様に従った運用(例 $pgpool -m fast stop)を行うよう、注意が必要

ケース4 pgpool-IIサーバ障害：pgpool-II障害時の切り替え時間50秒



- 切り替え時間50秒のうち、約40秒が障害検知にかかる時間（pgpool.confの設定値）
  - ⇒死活監視間隔：10秒／障害発生とみなす閾値：30秒

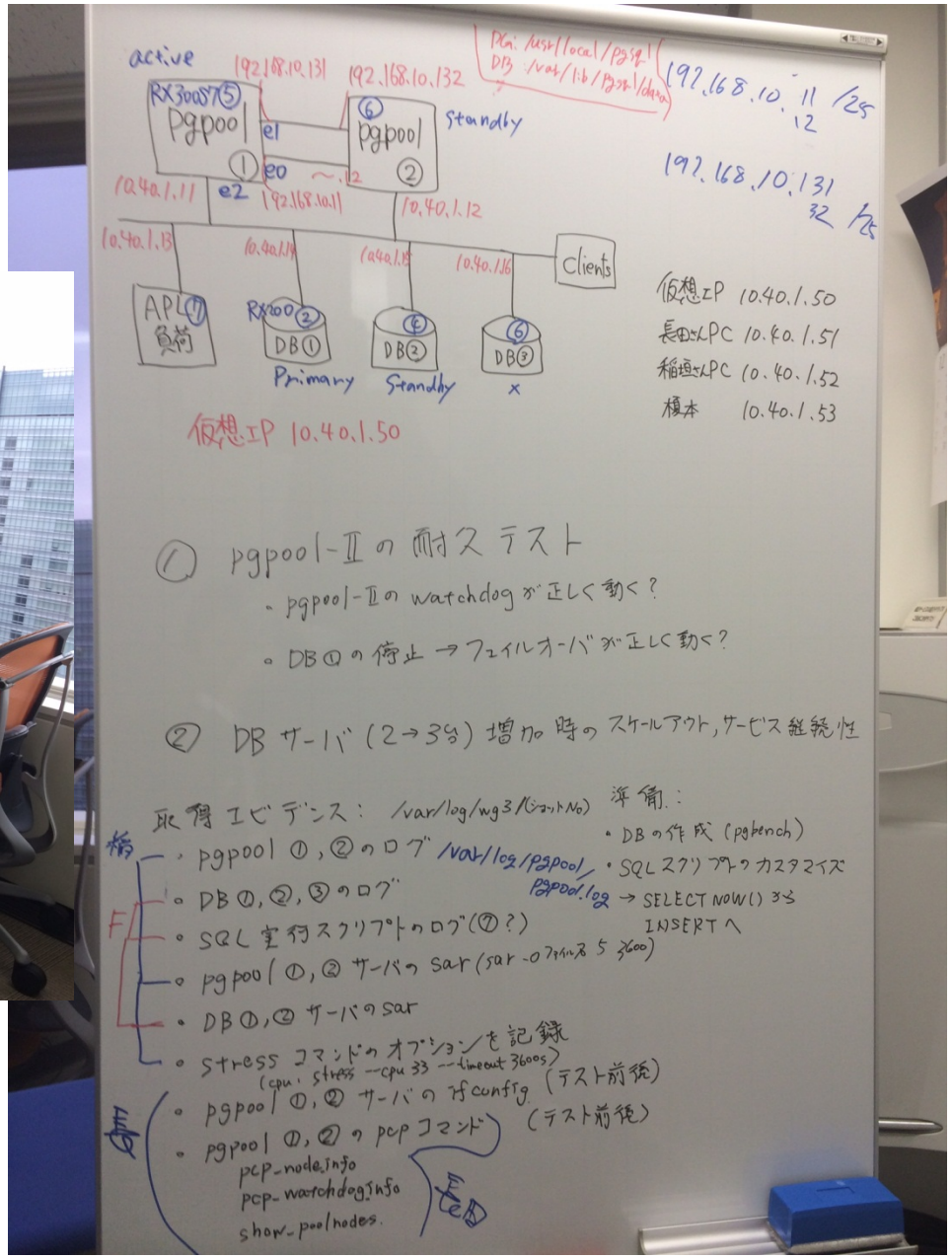
検証結果まとめ

- CPU高負荷状況でのpgpool-IIの
  自動フェイルオーバとwatchdog機能の運用性を検証

- 自動フェイルオーバ機能による速やかな切り替えを確認
- watchdog機能を用いた仮想IPアドレスの自動切り替えにより、サービスを継続提供可能
- ただし切り替え時間は、watchdogのハートビート受信に関するパラメータ値によるものが大きいため、目的に合ったpgpool.conf設定を行う必要あり
- プロセス終了させる場合、pgpool-IIの仕様に従った運用を行うよう注意する必要あり(例 $pgpool -m fast stop)
- **高負荷環境においても
  PostgreSQL+pgpool-IIの基本動作に問題がないことを確認**

WG3 活動風景(その2)

- 実機検証風景



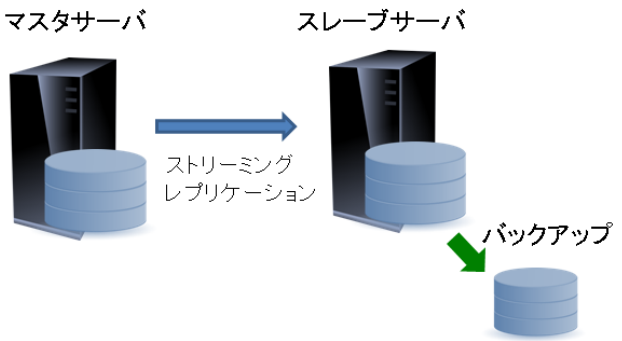
運用技術検証
～バックアップ／リカバリ検証～

検証概要

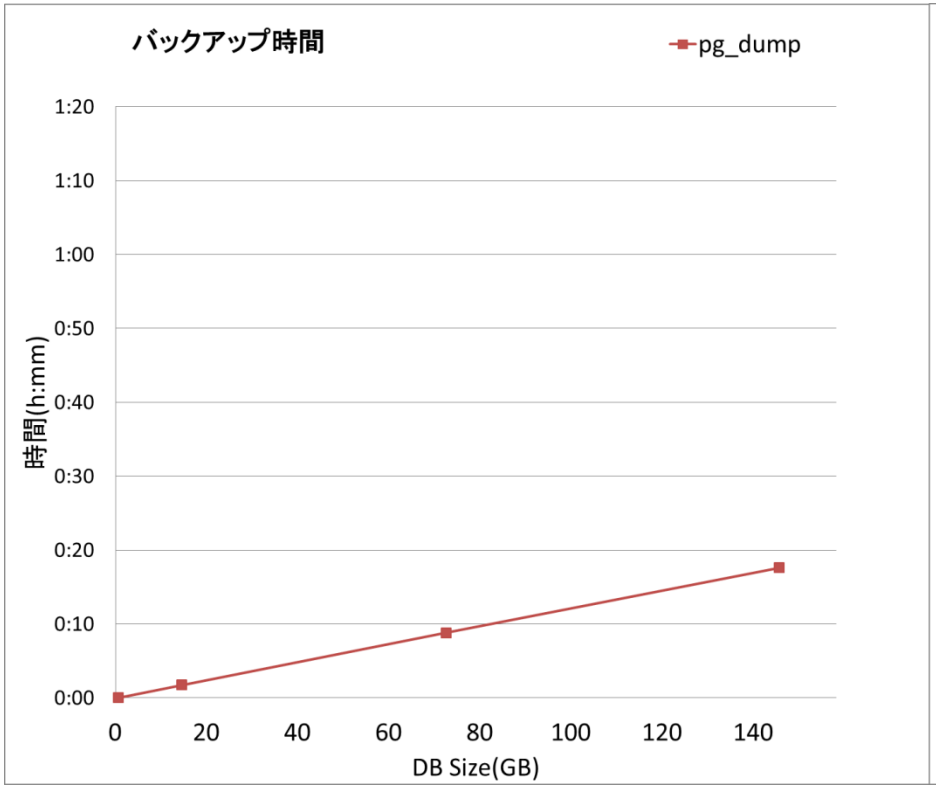
- 構成
  - 付録：検証環境1において「シングル構成」、「ストリーミングレプリケーション構成」の2構成で検証実施
- 検証シナリオ
  - 比較的非機能要求レベルの低いシステムを想定し、シングル構成で以下の検証を実施
    - クラッシュリカバリ
    - 論理バックアップからのリカバリ
    - オンラインバックアップとPITR

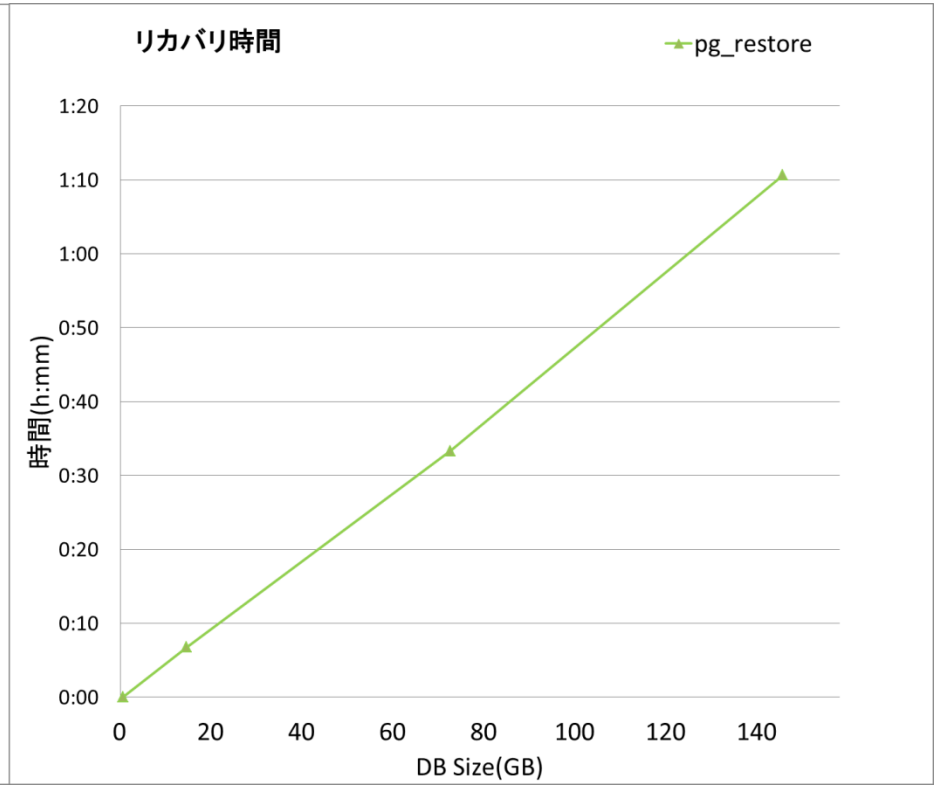
実行時間の測定も実施

  - 非機能要求レベルの高いシステムを想定し、ストリーミングレプリケーション構成においてマスタサーバに障害が発生したというシナリオで、以下の検証を実施
    - スレーブをマスタに昇格させて業務継続
    - スレーブで取得していたバックアップを使用してマスタをリカバリ



pg_dump／pg_restore実行時間測定結果

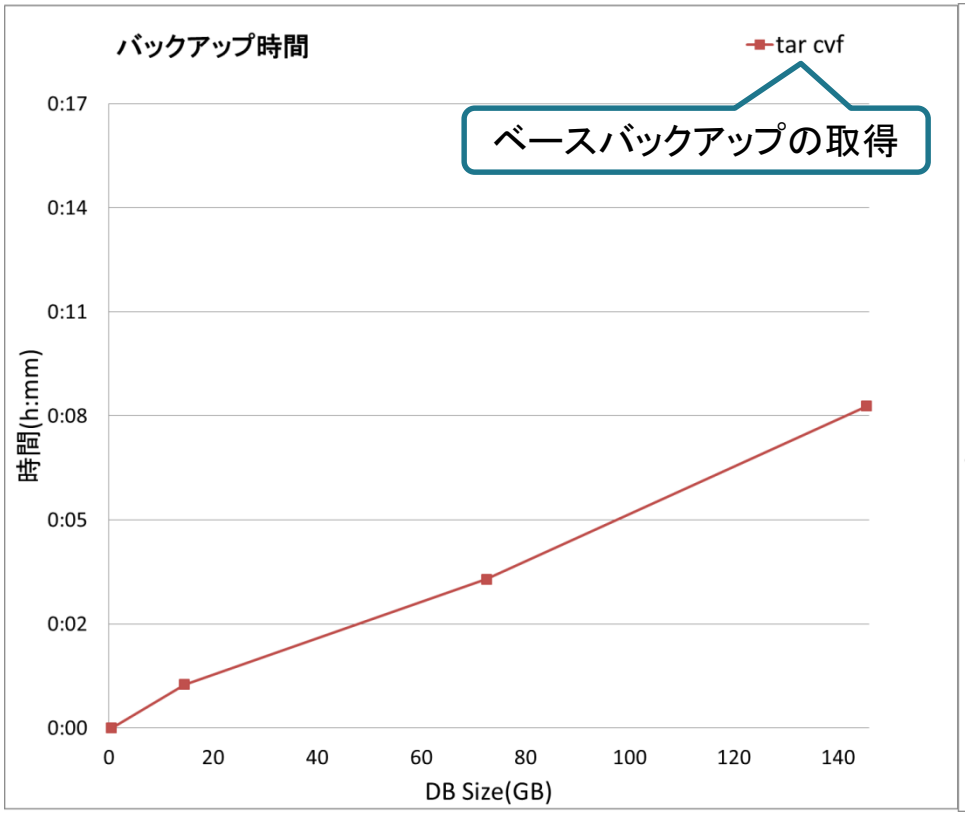


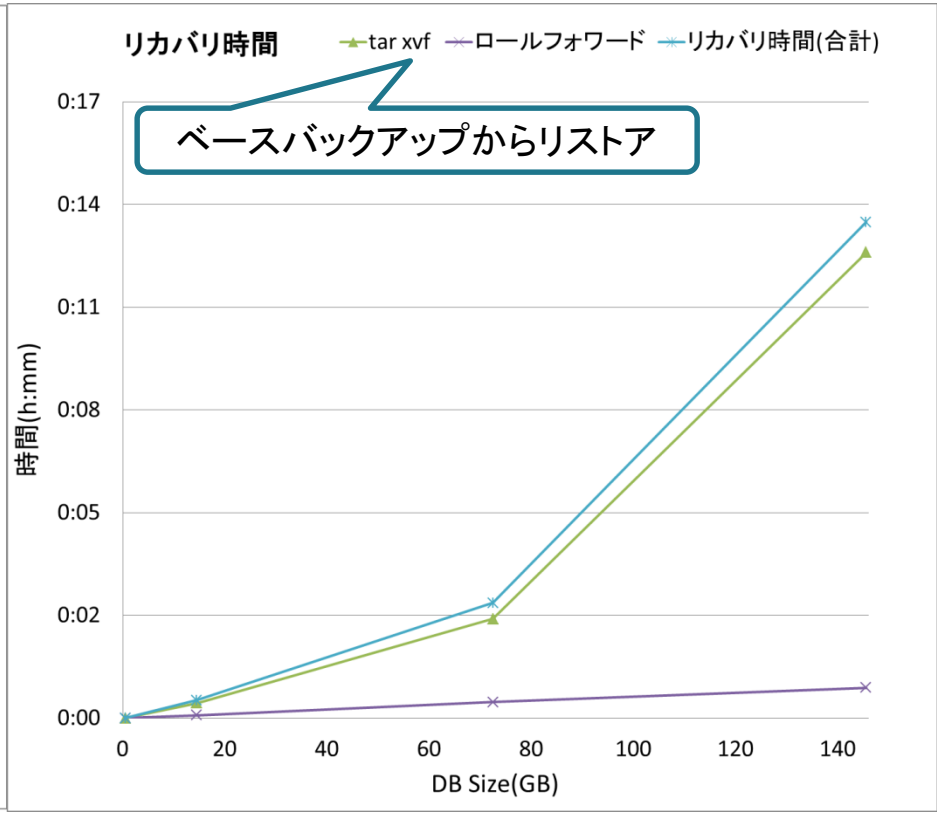


※実行時間はCPU性能やストレージ性能等の環境によって変わるため、実行時間の目安として考えてください。

- pg_dumpによる論理バックアップ時間はデータ量にほぼ比例
- 検証環境では100GB程度のバックアップが約12分で完了
- リカバリはデータ更新やそれにともなうWALファイル出力処理のため、バックアップよりも時間がかかる

オンラインバックアップ／リカバリ実行時間測定結果





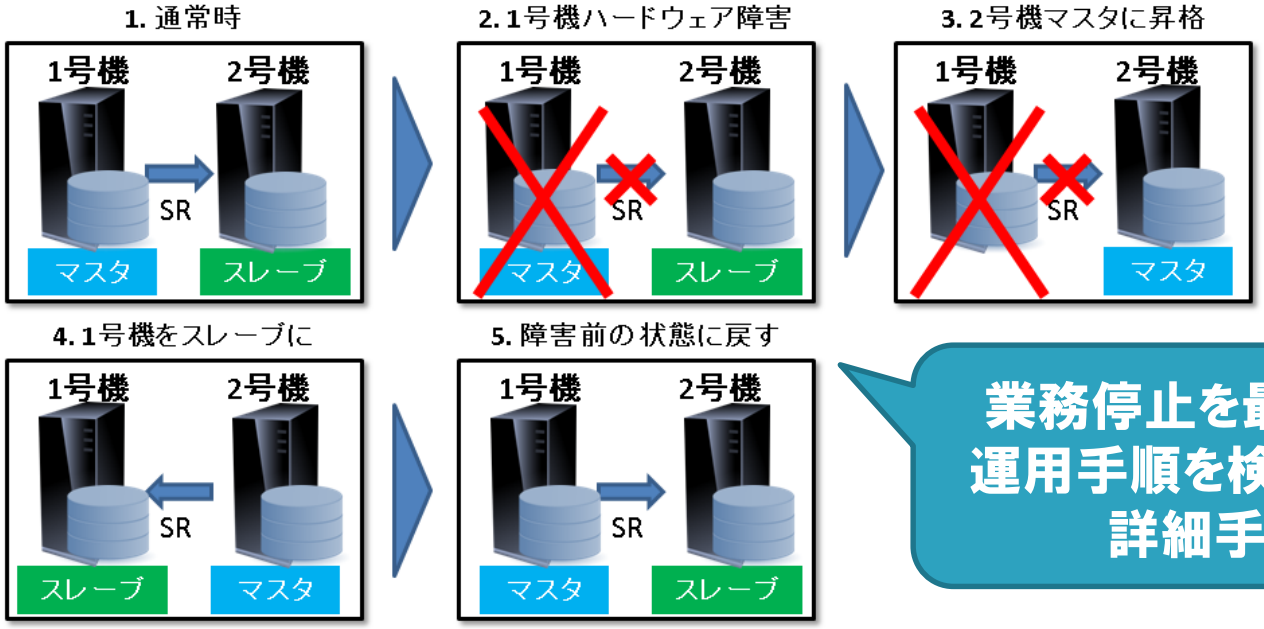
※実行時間はCPU性能やストレージ性能等の環境によって変わるため、実行時間の目安として考えてください。
※ロールフォワードするログのサイズはデータサイズの1/10で検証

- バックアップにかかる時間はデータ量にほぼ比例
- 検証環境では100GB程度のバックアップが約5分で完了

ストリーミングレプリケーション構成検証 –スレーブ昇格による業務継続–

■ 想定シナリオ

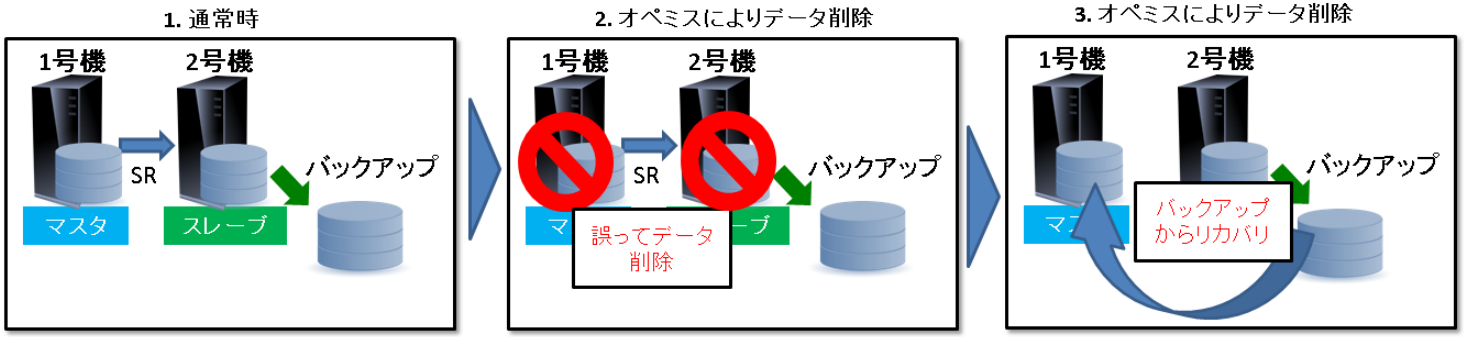
- □ マスタサーバのハードウェア障害が発生し、マスタダウン。
- □ 障害状況を調査・分析の結果、マスタの復旧は困難と判断し、スレーブをマスタに昇格(フェイルオーバー)させ業務継続する。
- □ バックアップから旧マスタを新スレーブとするストリーミングレプリケーション(SR)構成を構築。
- □ 新スレーブをマスタに昇格(スイッチバック)させ、障害前の構成に戻す。

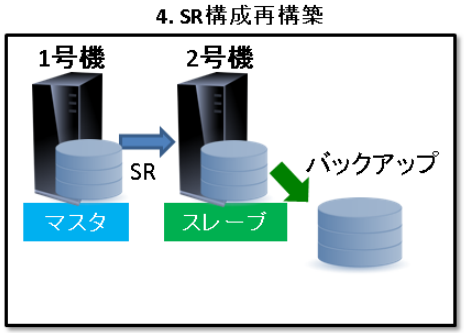


業務停止を最小限に抑える運用手順を検証により確認。詳細手順を公開

ストリーミングレプリケーション構成検証 –マスタをバックアップからリカバリ–

- **想定シナリオ**
  - ○○日△△時××分、オペレーションミスにより、データを誤って削除。
  - スレーブサーバの状況を確認したところ、スレーブサーバでもすでに削除。
  - スレーブサーバで毎日取得しているバックアップからマスタを○○日△△時××分の直前までリカバリ。
  - ストリーミングレプリケーション(SR)構成再構築。





スレーブのバックアップによりマスタのリカバリが可能であることを検証により確認。
詳細手順を公開

バックアップ／リカバリ検証 –まとめ–

代表的な構成である「シングル構成」と「ストリーミングレプリケーション構成」で、バックアップ／リカバリの手順や所要時間の目安を検証

- **シングル構成**
  - 一時的なサーバ障害は、再起動時に自動的にクラッシュリカバリによりデータ復旧される
  - pg_dumpによる論理バックアップは、100GBあたり10数分程度で可能（構成により変動）
  - tarコマンドを使用したオンラインバックアップは、100GBあたり5分程度で可能（構成により変動）
- **ストリーミングレプリケーション構成**
  - 障害時の業務停止を最小限に抑える運用手順を検証し公開
  - スレーブのバックアップを使用したマスタのリカバリ手順を検証し公開

適切なバックアップにより障害時にも
データ消失の不安なくPostgreSQLを運用できます！

運用技術検証
～監視ケーススタディ～

ケーススタディ概要

- 全体の流れ
  - 問題発見から収集した情報やエラーメッセージの分析、対処方針案の検討を以下の4つの観点で確認

作業項目	作業内容
アラート	異常を検知する
現状把握	現在の状況を確認し、早急な対処が必要か否かを判断する
切り分け	どこで問題が生じているか切り分ける
調査・対処	問題に対する具体的な解決策を施す

- 前提
  - シングル構成およびストリーミングレプリケーション構成を想定
- シナリオ
  - キャッシュヒット率低下
  - HAクラスタの更新性能低下

キャッシュヒット率の低下

- アラート
  - 監視ツールよりキャッシュヒット率低下のアラートがあがる
- 現状把握
  - 過去に収集した情報および下記SQLで、キャッシュヒット率の推移を確認する

```
SELECT relname,
      round(heap_blks_hit*100/(heap_blks_hit+heap_blks_read), 2) AS cache_hit_ratio
FROM pg_statio_user_tables
WHERE heap_blks_read > 0
ORDER BY cache_hit_ratio;
```

  - 上記の結果、いくつかのテーブルでキャッシュヒット率の低下が継続していた
- 切り分け
  - PostgreSQLの設定見直し
    - shared_buffersやwork_memの値を確認

    →デフォルトのまま運用していたことが発覚。
- 調査・対処
  - 設定値の修正(一般的に利用されている値としてshared_buffersを物理メモリの25%、work_memを8MBに設定)と再起動を行い、経過を観測。他の処理への影響とキャッシュヒット率の低下がみられなくなったことを確認

どちらの設定も、大きくしすぎても
逆効果になるので要注意！

HAクラスタの更新性能低下

- アラート
  - 更新を伴うバッチ処理が想定よりも長いという報告を受けた
- 現状把握
  - ログから確かに処理時間が想定よりかかっていたことを確認
  - 大幅なデータ量の増減もなかった
- 切り分け
  - 以下のように更新処理が行われるマスタの特定を行い問題個所を切り分けた
    - pg_is_in_recovery関数の実行　　→falseが返ればマスタと推測できる
    - transaction_read_onlyの値を確認 →offであればマスタと推測できる
    - pg_stat_replicationビューの情報を確認→結果が返ればマスタと推測できる
    - recovery.confファイルの有無を確認→ファイルが存在しなければマスタと推測できる
    - PostgreSQLのログを確認→「entering standby mode」のメッセージがなければマスタと判断できる
  - 上記および性能監視で取得していた情報から、バッチ実行と同時刻帯にマスタサーバのiowaitが高騰していたことが分かった。
- 調査・対処
  - 本環境は仮想サーバを用いており、別の仮想サーバと物理ディスクを共用していた。本事象は他の仮想サーバのディスクアクセスと競合していたことが原因であったため、他の仮想サーバをマイグレーションすることで対処した

監視ケーススタディのまとめ

- さまざまなケースがあるが、主な監視のケースを「アラート」「現状把握」「切り分け」「調査・対処」といった流れで、問題発見から対処まで行った
- 特に「現状把握」「切り分け」では必要なOSコマンドやPostgreSQL関数の実例を挙げ、真の問題点を的確に把握して対処することが重要

PostgreSQLの問題に見えて、
別の問題であるケースも。
面倒でも状況を整理することが最短ルートに！

エンディング
～　WG3からのメッセージ～

エンタープライズなDB要件に応えられるか

- 可用性、バックアップ、監視とも不安なし
  - 今回の検証では想定どおりの結果が得られた
  - 業務用データベースとして、普通に使用できる

  商用データベースにも引けを取りません！

- エンタープライズな非機能要件への対応
  - セキュリティなど残項目については、来期以降の課題

2013年度を振り返って（1）

- 各社からいただいたコメントを抜粋します。
  - 主査という大役を拝命し、当初は大所帯のWGの運営方法に悩みましたが、参加企業の皆様が積極的に動いてくださり助けていただきました。ありがとうございました。
  - サブグループに分かれての並行活動により、各社のノウハウと共同検証の結果が凝縮された成果物が完成した事を嬉しく思います。
  - 本来なら競争相手になるメンバのみなさんと、同じ目標に向かって議論して作業を進めていくことってとっても楽しいと感じました。

2013年度を振り返って（2）

- 各社からいただいたコメントを抜粋します。
  - WG3は活動初年度ということもあり活動の方向性をまとめるのに苦労しましたが、報告書を公開するところまで到達したのは感慨深いですね。
  - 検討会、検証作業、成果物執筆といったWG活動と実業務の両立は大変でしたが、PostgreSQLに詳しい技術者の方々と定期的に議論する場を得られたのは良かったです。
  - 来年度は今回の成果をベースに技術検証の守備範囲を増やしていけたら良いと思っています。

2014年活動予定

- 設計運用WG（WG3）
  - 災害対策など可用性の検証範囲を拡大
  - セキュリティ(監査)など新テーマに着手

PGEConsにおける課題領域

性能	性能評価手法、性能向上手法、チューニングなど
可用性	高可用クラスタ、BCP
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
セキュリティ	監査
互換性	データ、スキーマ、SQL、ストアドプロシージャの互換性
接続性	他ソフトウェアとの連携

可用性の適用範囲拡大

新テーマに着手

設計運用分野の適用範囲を拡大！

謝辞

検証用の機器を以下の各社よりご提供いただきました。
PostgreSQLエンタープライズ・コンソーシアムとして、この場を借りて厚く御礼を申し上げます。

株式会社日立製作所

SRA OSS, Inc. 日本支社

富士通株式会社

WG3活動メンバ　オールキャスト

お疲れ様でした♪

PGECons

PostgreSQL Enterprise Consortium

ご清聴ありがとうございました

付録：検証環境１ （提供： 株式会社 日立製作所）

■ 実機検証での使用機器／設備

ハーモニアス・コンピテンス・センター

★BS500 (BladeSymphony BS540A)

PostgreSQLサーバとして使用

CPU:インテルXeon E5-4610 (2.40GHz) 6コア×4

メモリ:256GB

内蔵HDD:900GB×2 RAID1

★HA8000 (HA8000/RS220-h HM)

クライアントとして使用

CPU:インテルXeon E5-2690 (2.90GHz) 8コア×2

メモリ:32GB

内蔵HDD:900GB×4 RAID6

検証ルーム

LAN

FC接続

サーバルーム

JR品川駅 港南口より徒歩約3分

京浜急行品川駅 高輪口より徒歩約5分

★HUS100 (Hitachi Unified Storage 130)

ストレージとして使用

HDD (600GB x 5 RAID5) データ用

SSD (200GB x 3 RAID5) WAL用

HDD (600GB x 5 RAID5) アーカイブ用

HDD (600GB x 5 RAID5) バックアップ用

HDD (600GB x 5 RAID5) データ用

HDD (600GB x 5 RAID5) WAL用

HDD (600GB x 5 RAID5) アーカイブおよびバックアップ用

付録：検証環境2 （提供： 富士通株式会社）

■ 実機検証での使用機器／設備

富士通トラステッド・クラウド・スクエア

★PRIMERGY RX300 S7

上位サーバ、pgpool-IIサーバとして使用

CPU:インテルXeon E5-2690 (2.90GHz) 8コア×2

メモリ:8GB

内蔵HDD:600GB×2 RAID1

★PRIMERGY RX200 S7

PostgreSQLサーバとして使用

CPU:インテルXeon E5-2690 (2.90GHz) 8コア×2

メモリ:8GB

内蔵HDD:600GB×2 RAID1

検証ルーム

サーバルーム

最新の当社サーバ、ストレージ機器を約300台完備し、事前導入検証やベンチマーク、ICTシステム検証が可能です。

詳細情報は以下をご確認ください。
http://jp.fujitsu.com/facilities/tcs/