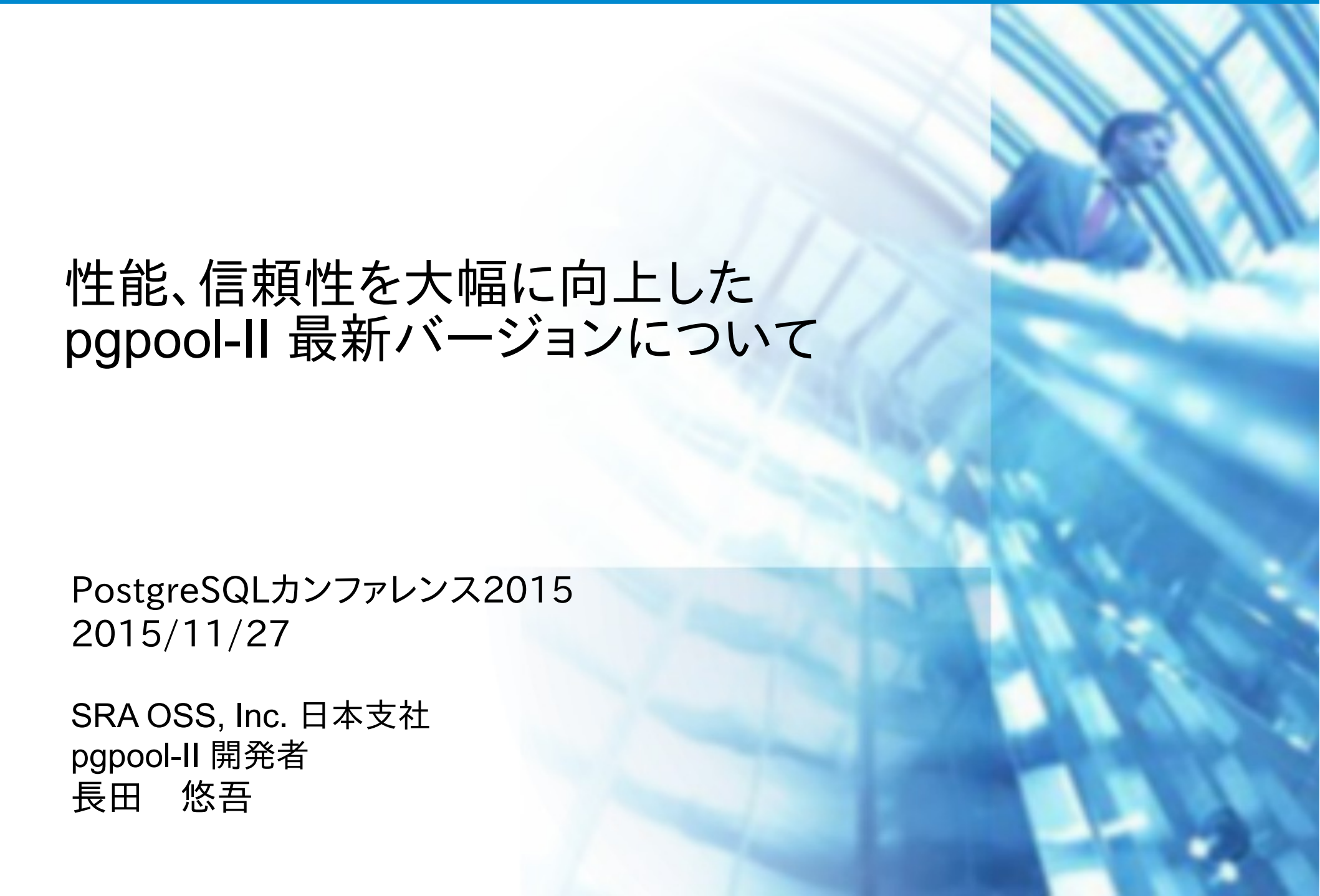
SRA OSS, INC.



性能、信頼性を大幅に向上した pgpool-II 最新バージョンについて

PostgreSQLカンファレンス2015
2015/11/27

SRA OSS, Inc. 日本支社
pgpool-II 開発者
長田　悠吾

自己紹介

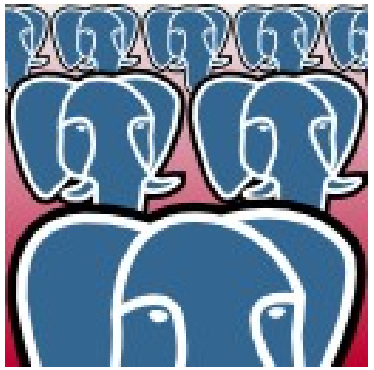
- 長田　悠吾　（ナガタ　ユウゴ）
  - SRA OSS, Inc. 日本支社
  - マーケティング部 OSS技術グループ
  - OSS の技術サポート
  - pgpool-II 開発者
  - PostgreSQL 関連の技術調査・開発

SRA OSS, Inc. 日本支社

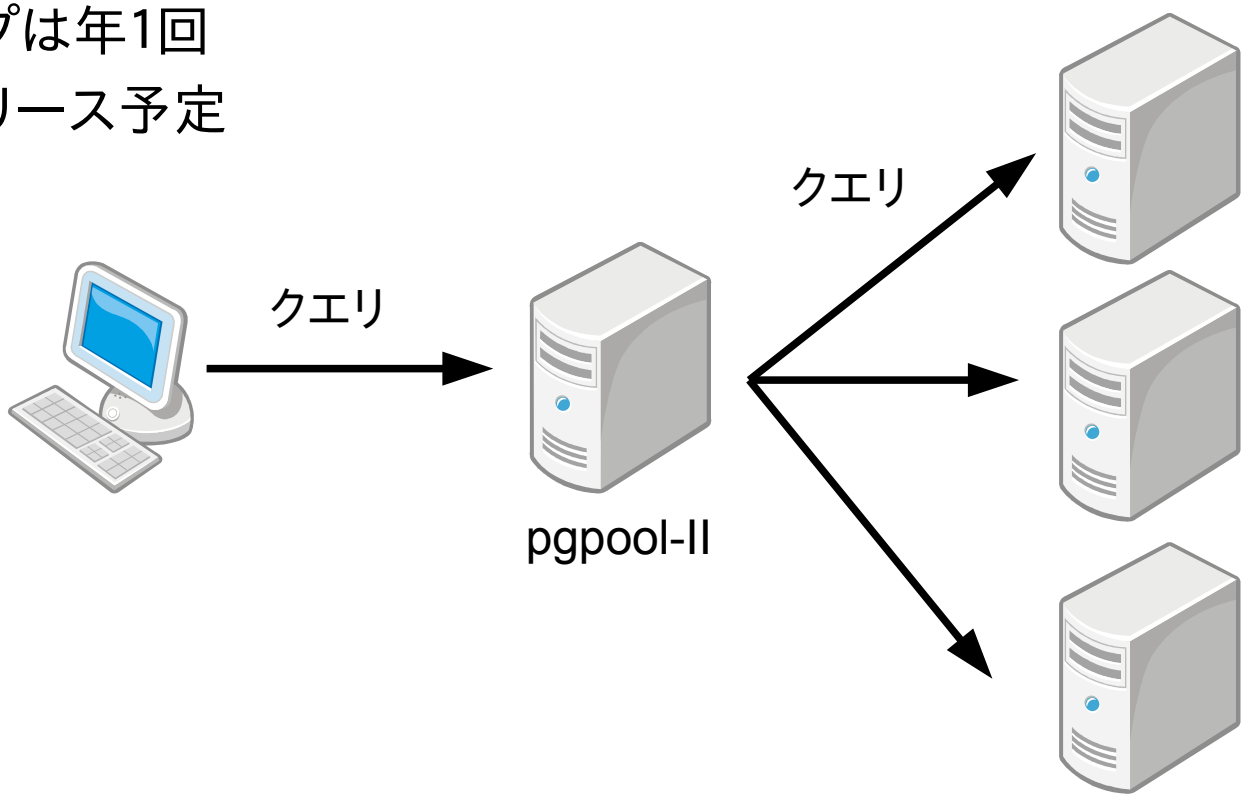
- 1999年よりPostgreSQLサポートを中心にOSSビジネスを開始

- PostgreSQL、Hinemos、Zabbix などのOSSサポート
- PowerGresファミリーの開発、販売
- トレーニング、導入、設計コンサルティングサービス

pgpool-II とは



- アプリケーションとPostgreSQLの間に入って、クラスタリング機能を提供するミドルウェア
  - アプリケーションからは普通のPostgreSQLに見える
- オープンソースソフトウェア（BSDライセンス）
  - メジャーバージョンアップは年1回
  - 今年の冬に 3.5.0 がリリース予定
- 多彩な機能
  - コネクションプーリング
  - 参照負荷分散
  - クエリキャッシュ
  - ヘルスチェック
  - 自動フェイルオーバ
  - オンラインリカバリ

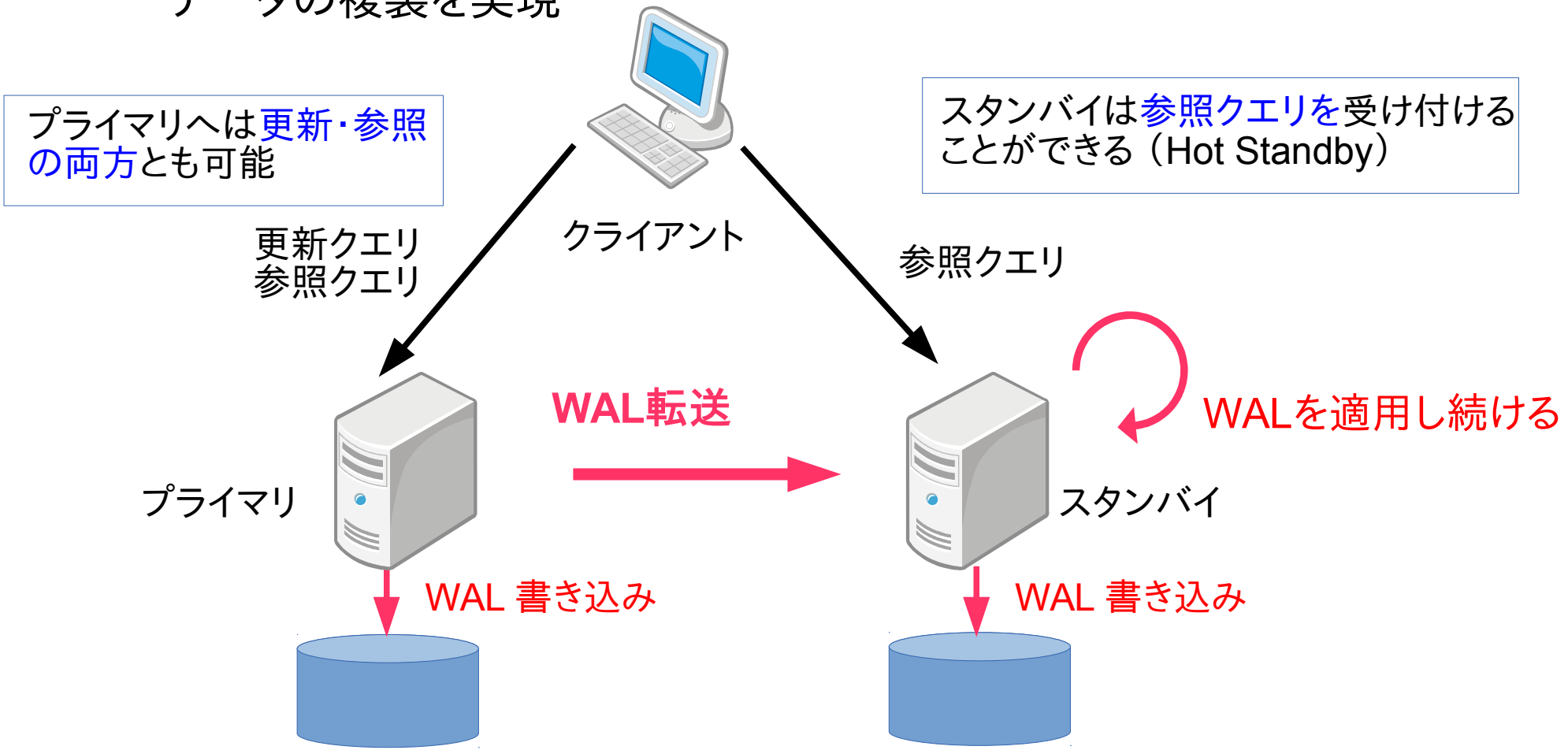


PostgreSQLのクラスタ技術

- HAクラスタ
  - Pacemaker+DRBD、共有ストレージなどを利用
  - 待機側はサービス停止
- ストリーミングレプリケーション
  - PostgreSQL自体が持つ、非同期レプリケーション機能
  - プライマリ（更新可能） + 複数のスタンバイDB（検索のみ）
  - 簡単、確実、速い
- pgpool-II
  - クライアントとPostgreSQLの間に入ってレプリケーション機能を提供
  - コネクションプーリング、負荷分散、自動フェイルオーバなど他の機能もある
- Postgres-XC/XL
  - PostgreSQLを改造したクラスタシステム
  - 書き込み性能の負荷分散

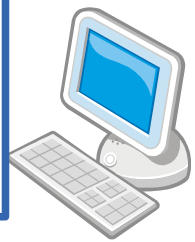
PostgreSQLのストリーミングレプリケーション

- ストリーミングレプリケーション（PostgreSQL 9.0 ～）
  - マスタからスレーブにトランザクションログ（WAL）を転送することにより データの複製を実現



ストリーミングレプリケーションの課題

負荷分散はどうすればよい？
更新クエリ、参照クエリの振り分けは？
アプリケーションを書き換えなきゃだめ？



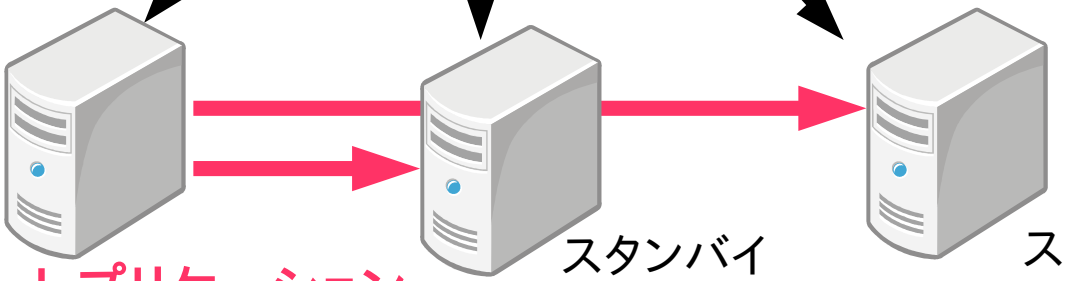
クライアント

更新クエリ
参照クエリ

参照
クエリ

参照クエリ

DBサーバに障害が発生したら？
手動で対応するの？



プライマリ

レプリケーション

スタンバイ

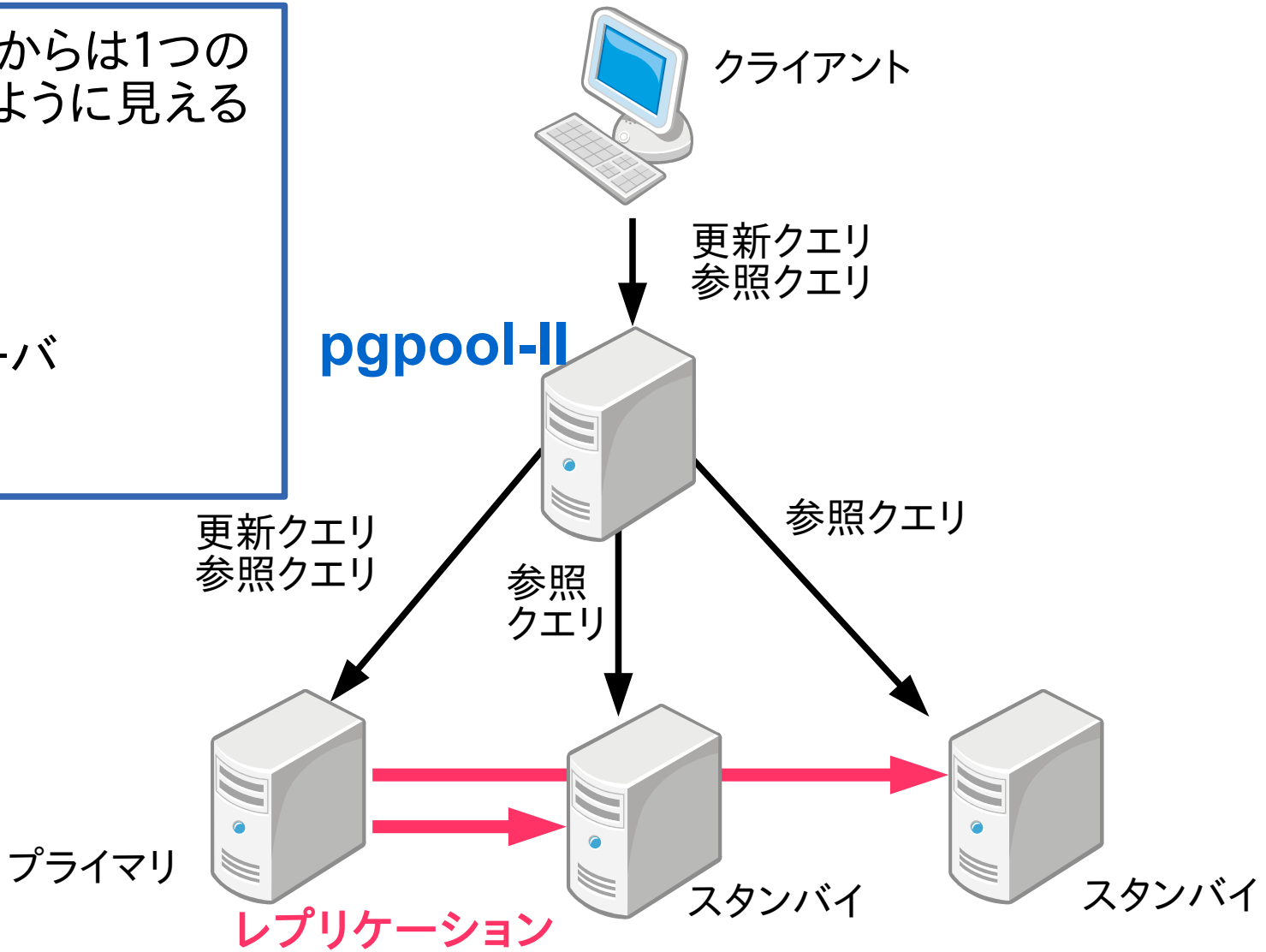
スタンバイ

プライマリがダウンしたら更新ができなくなる?!
サービスが停止してしまう!?

新しいスタンバイの追加は？

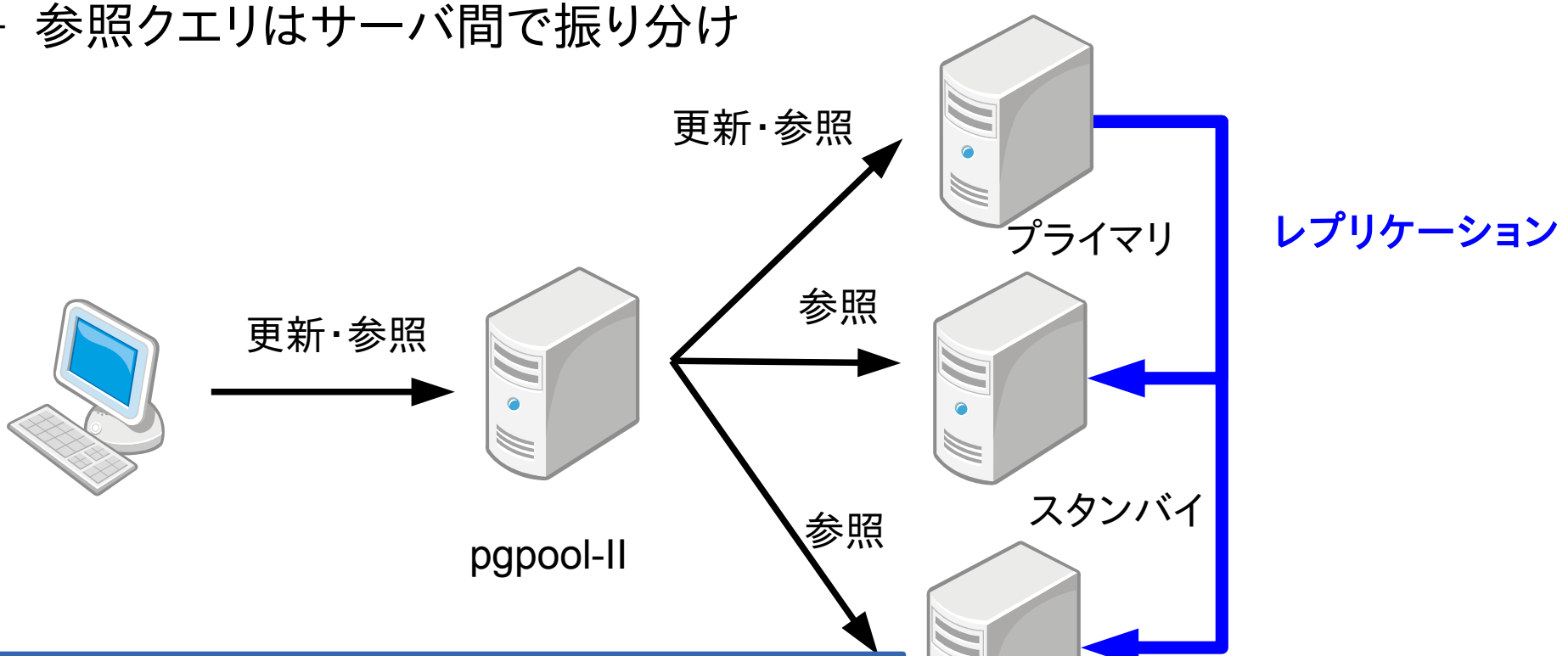
pgpool-II の導入

- アプリケーションからは1つのPostgreSQL のように見える
- クエリ振り分け
- 負荷分散
- 自動ファイルオーバ
- ・・・



参照負荷分散

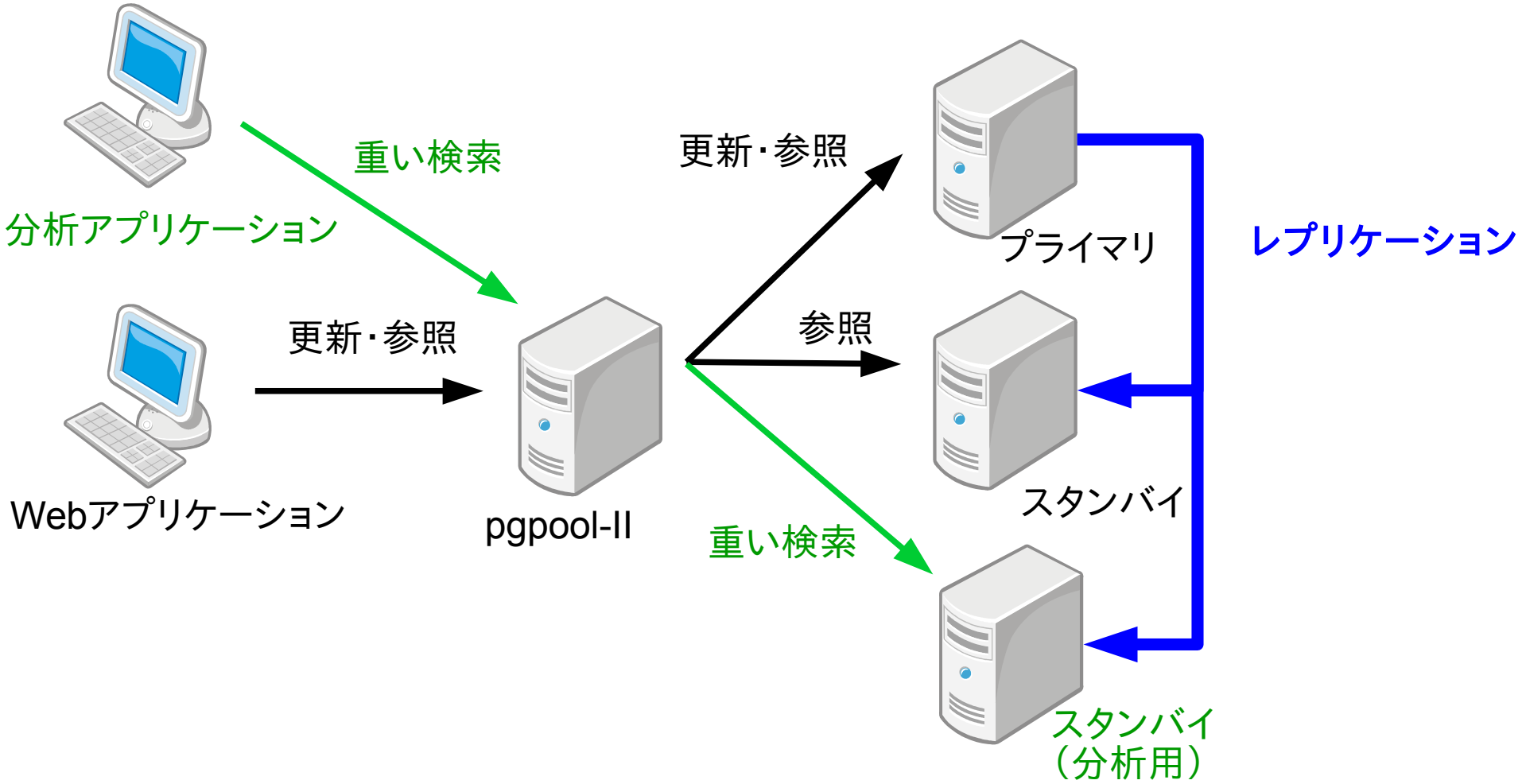
- クエリの自動振り分け
  - 更新クエリはプライマリサーバへ
  - 参照クエリはサーバ間で振り分け



振り分けの重みを指定可能
レプリケーション遅延が大きいサーバには振り分けない
3.4.0 からはよりきめ細やかな振り分けが可能に!

きめ細かな負荷分散（3.4～）

- アプリケーション名、DB名によって接続先が指定できる

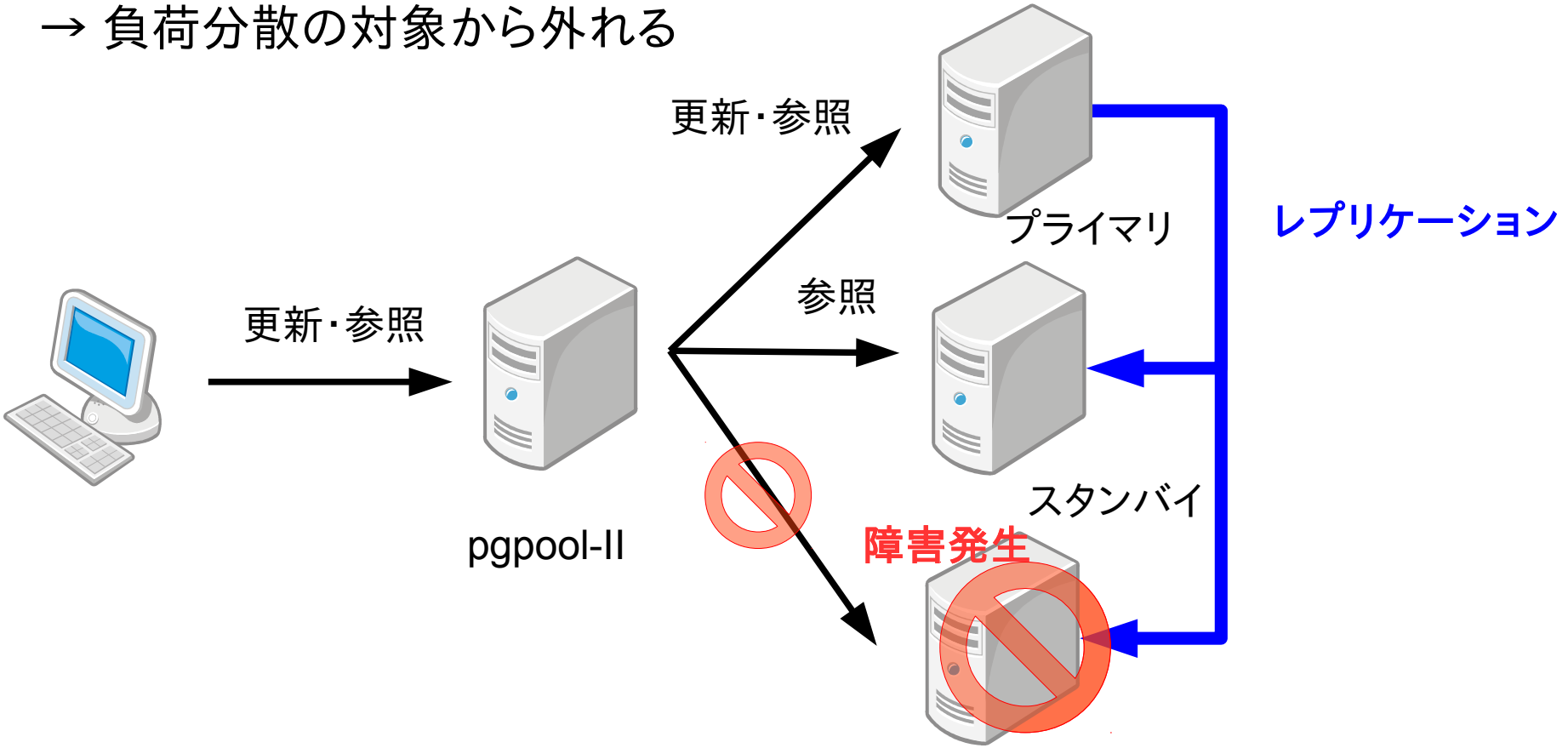


負荷分散の設定例

```
# PostgreSQL のホスト名、ポート番号、ロードバランスの重みを設定

backend_hostname0 = 'host1'
backend_port0 = 5432
backend_weight0 = 1

backend_hostname1 = 'host2'
backend_port1 = 5432
backend_weight1 = 1

backend_hostname2 = 'host3'
backend_port2= 5432
backend_weight2 = 0

# 接続アプリケーションによって、接続先DBを変更

app_name_redirect_preference_list = 'analyze_app:2'
```

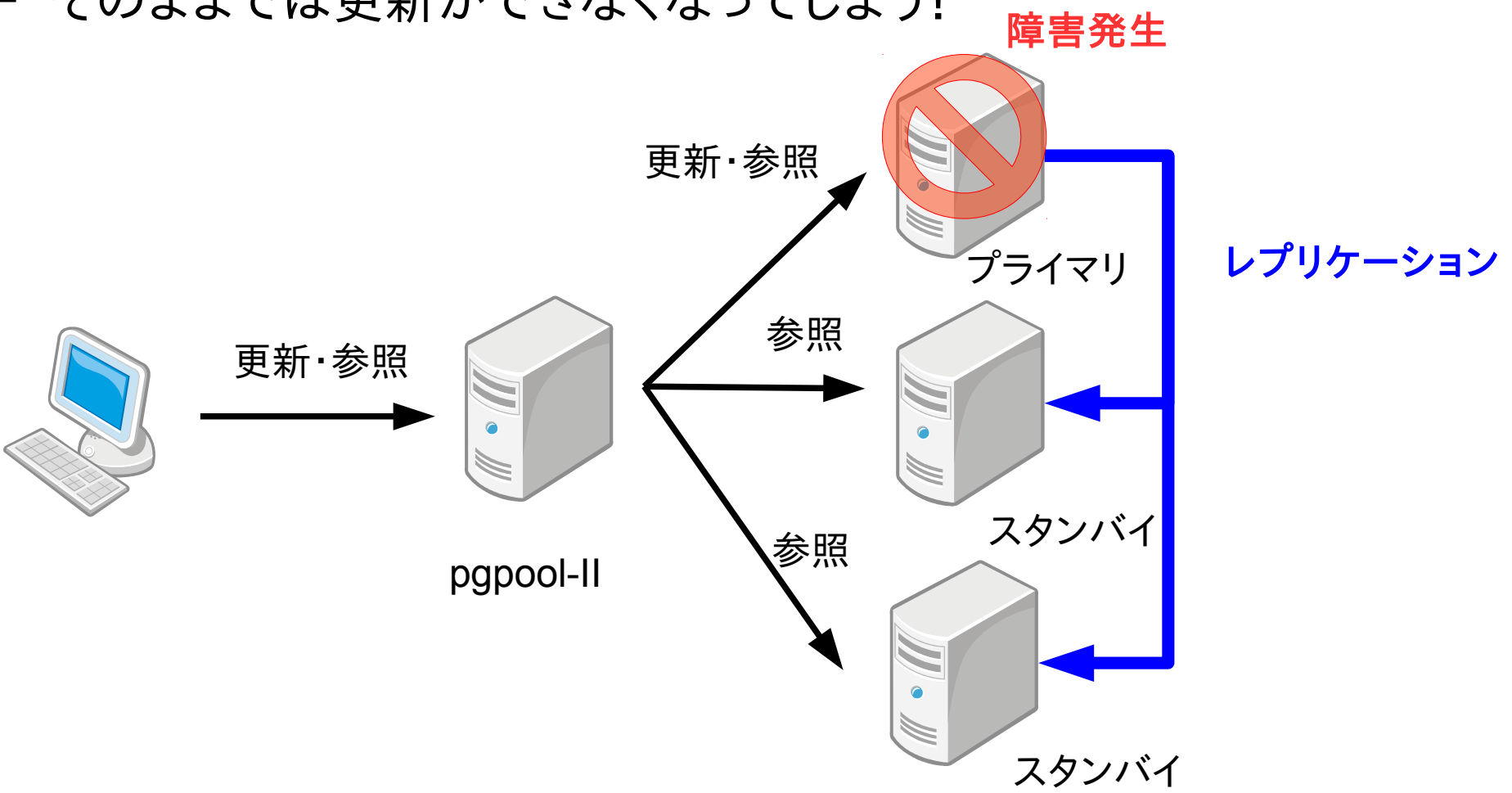
自動フェイルオーバ

- DBサーバの障害を自動検出（ヘルスチェック機能）
  - ダウンしたPostgreSQLを切り離す
    → 負荷分散の対象から外れる



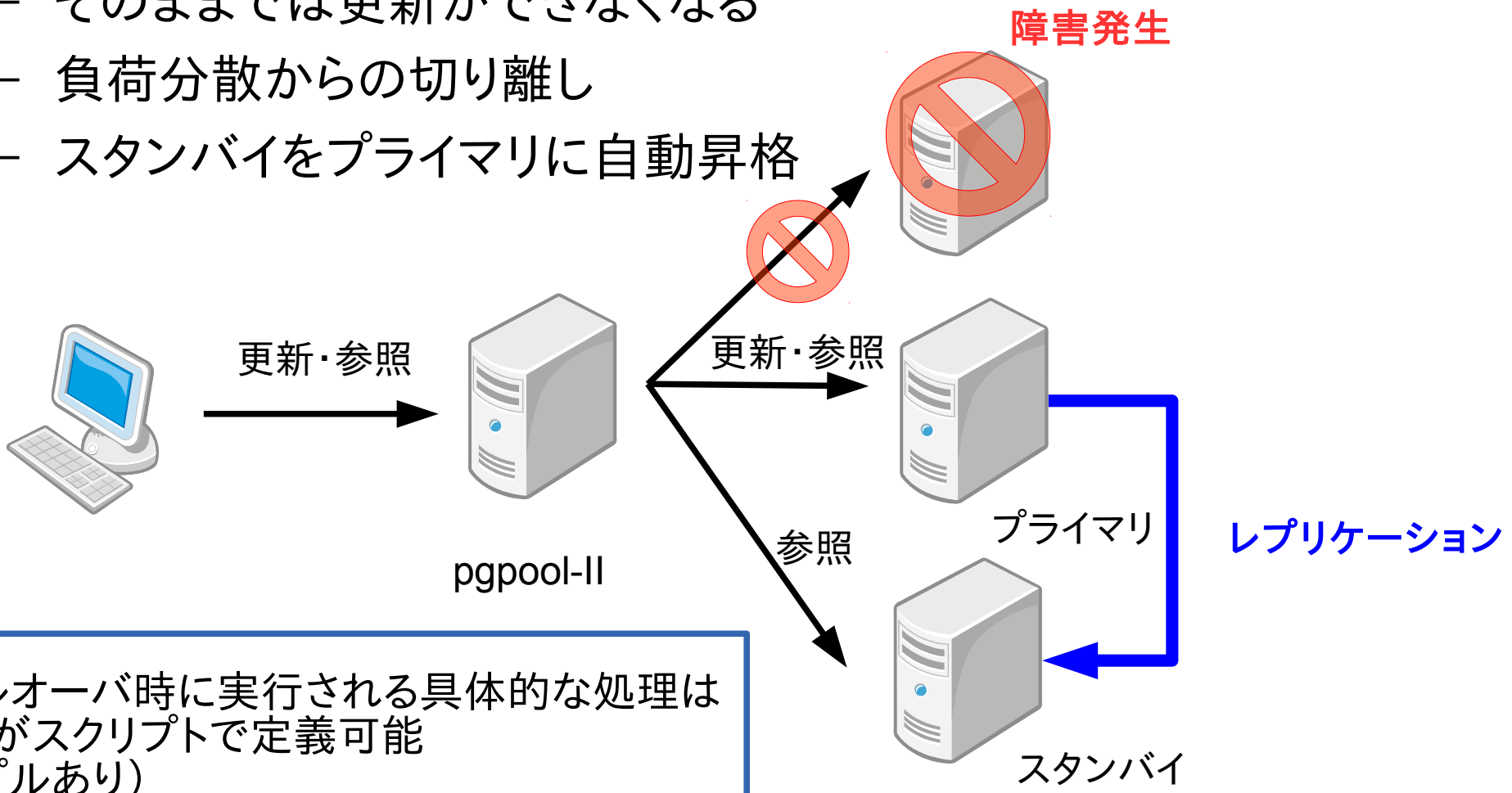
自動フェイルオーバ

- プライマリサーバに障害が発生した場合は？
  - そのままでは更新ができなくなってしまう!



自動フェイルオーバ

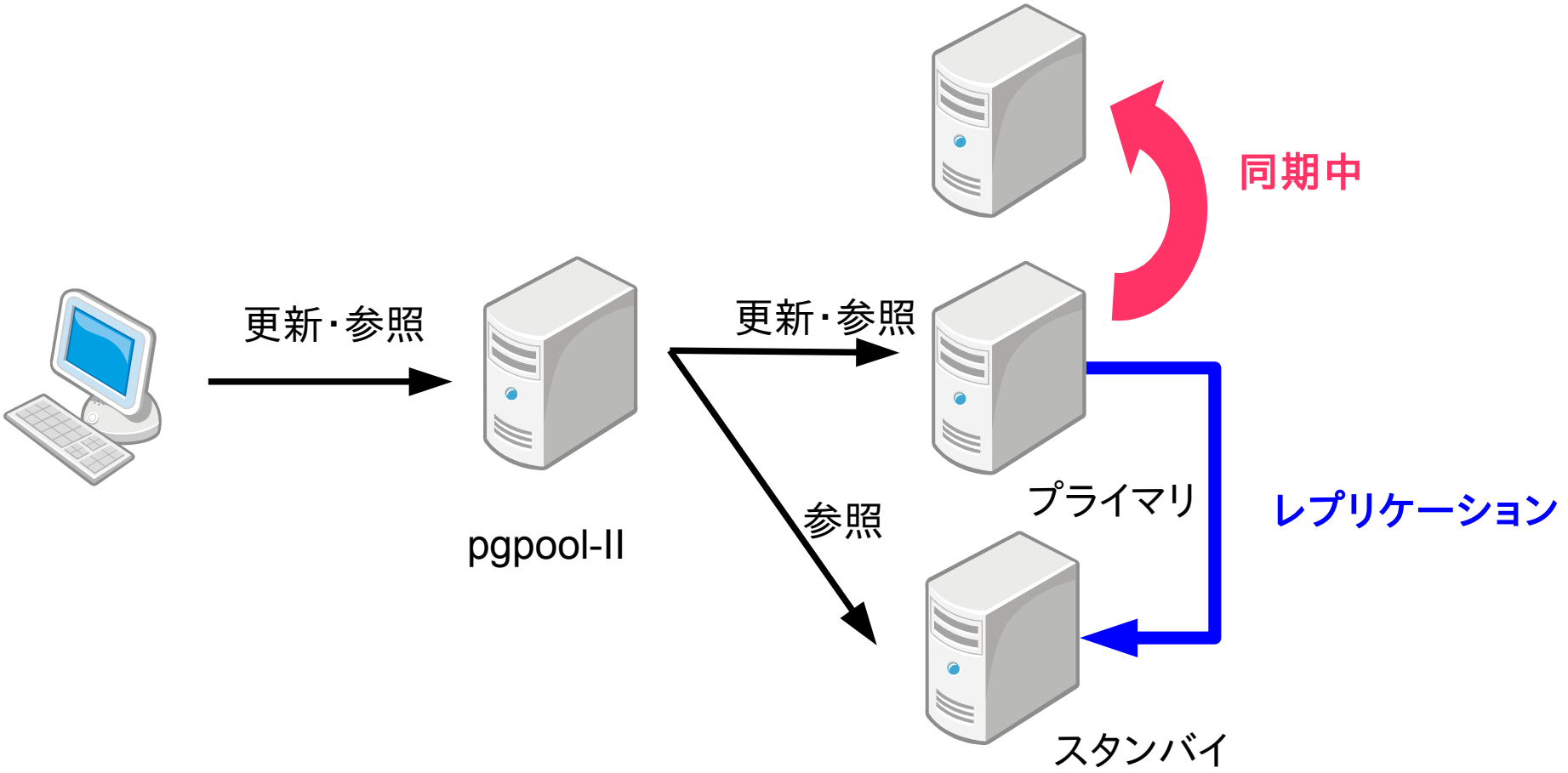
- プライマリサーバに障害が発生した場合
  - そのままでは更新ができなくなる
  - 負荷分散からの切り離し
  - スタンバイをプライマリに自動昇格



フェイルオーバ時に実行される具体的な処理は
ユーザがスクリプトで定義可能
（サンプルあり）

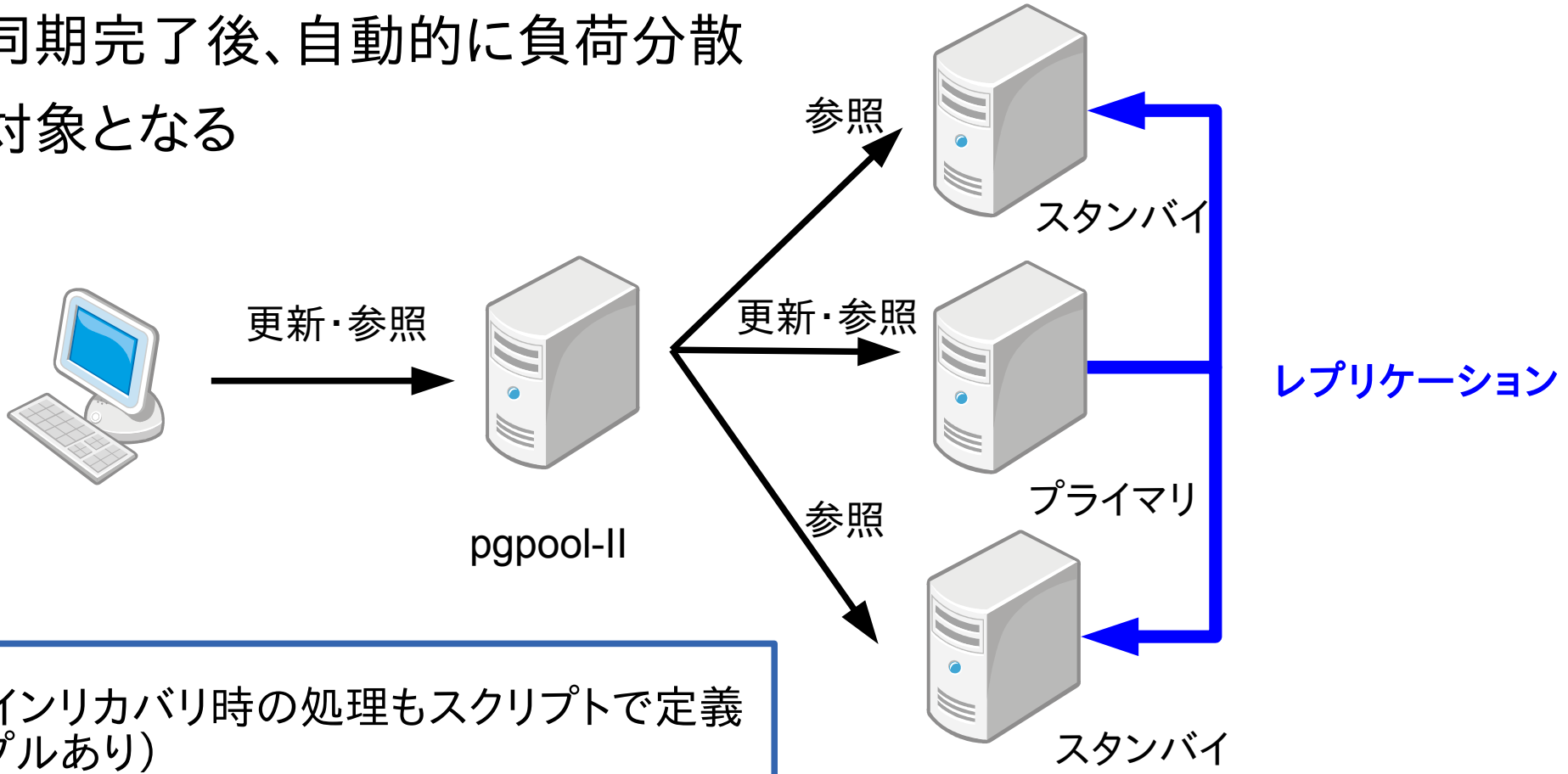
オンラインリカバリ

- ダウンしたスタンバイをプライマリに同期させる
- 同期中も更新が可能



オンラインリカバリ

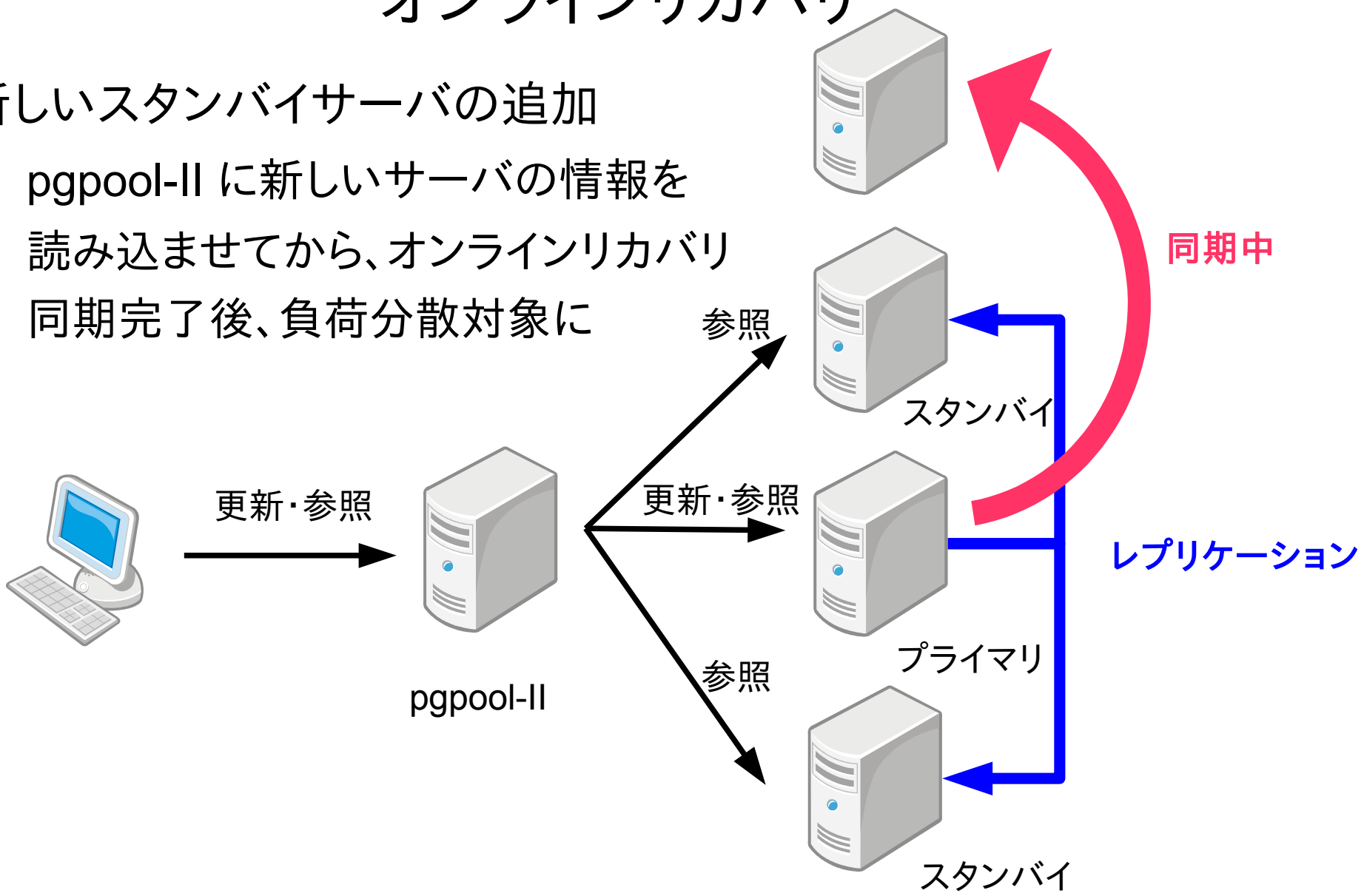
- ダウンしたスタンバイをプライマリに同期させる
- 同期中も更新可能
- 同期完了後、自動的に負荷分散対象となる



オンラインリカバリ時の処理もスクリプトで定義
（サンプルあり）

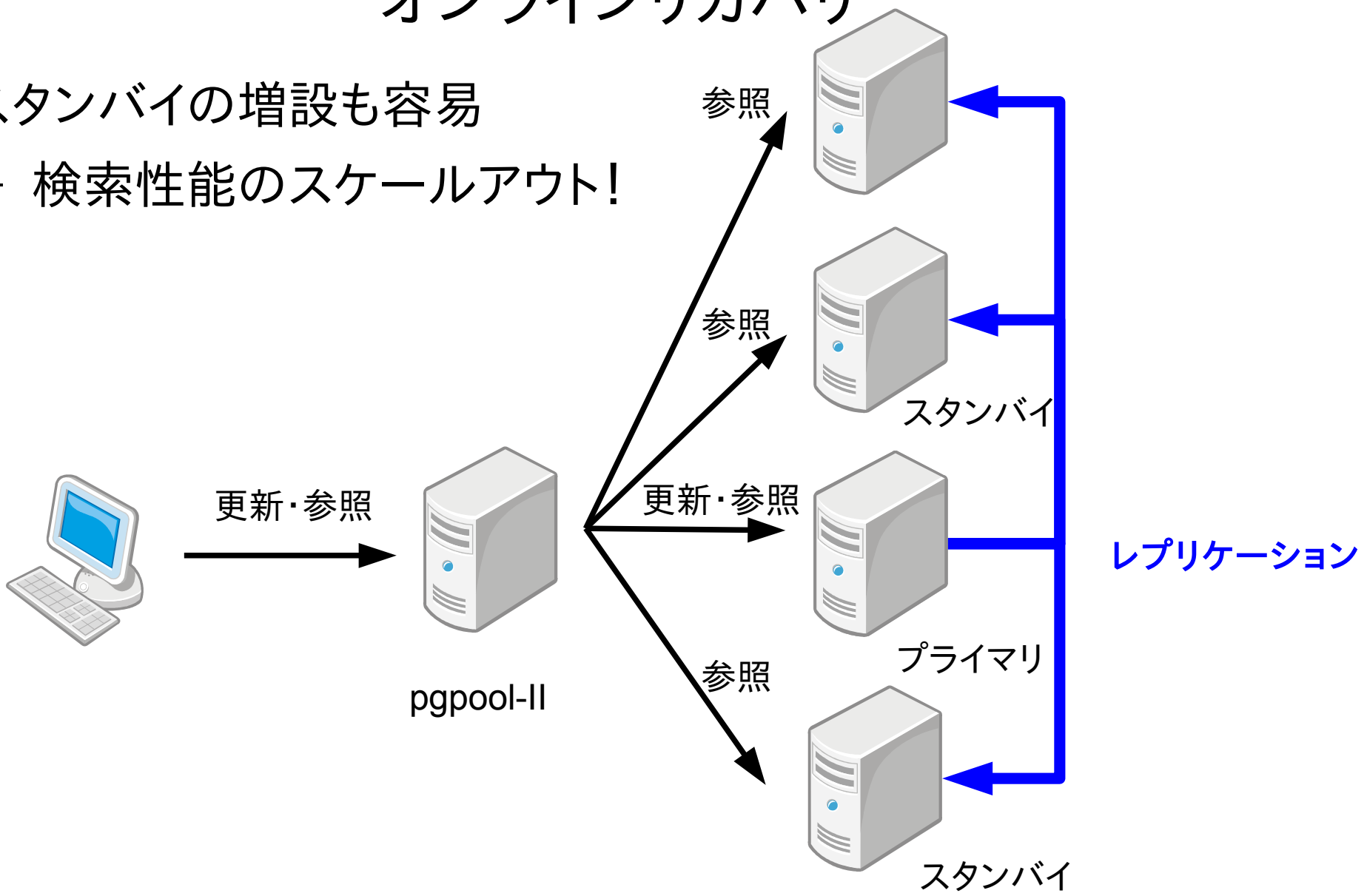
オンラインリカバリ

- 新しいスタンバイサーバの追加
  - pgpool-II に新しいサーバの情報を読み込ませてから、オンラインリカバリ
  - 同期完了後、負荷分散対象に



オンラインリカバリ

- スタンバイの増設も容易
  - 検索性能のスケールアウト!

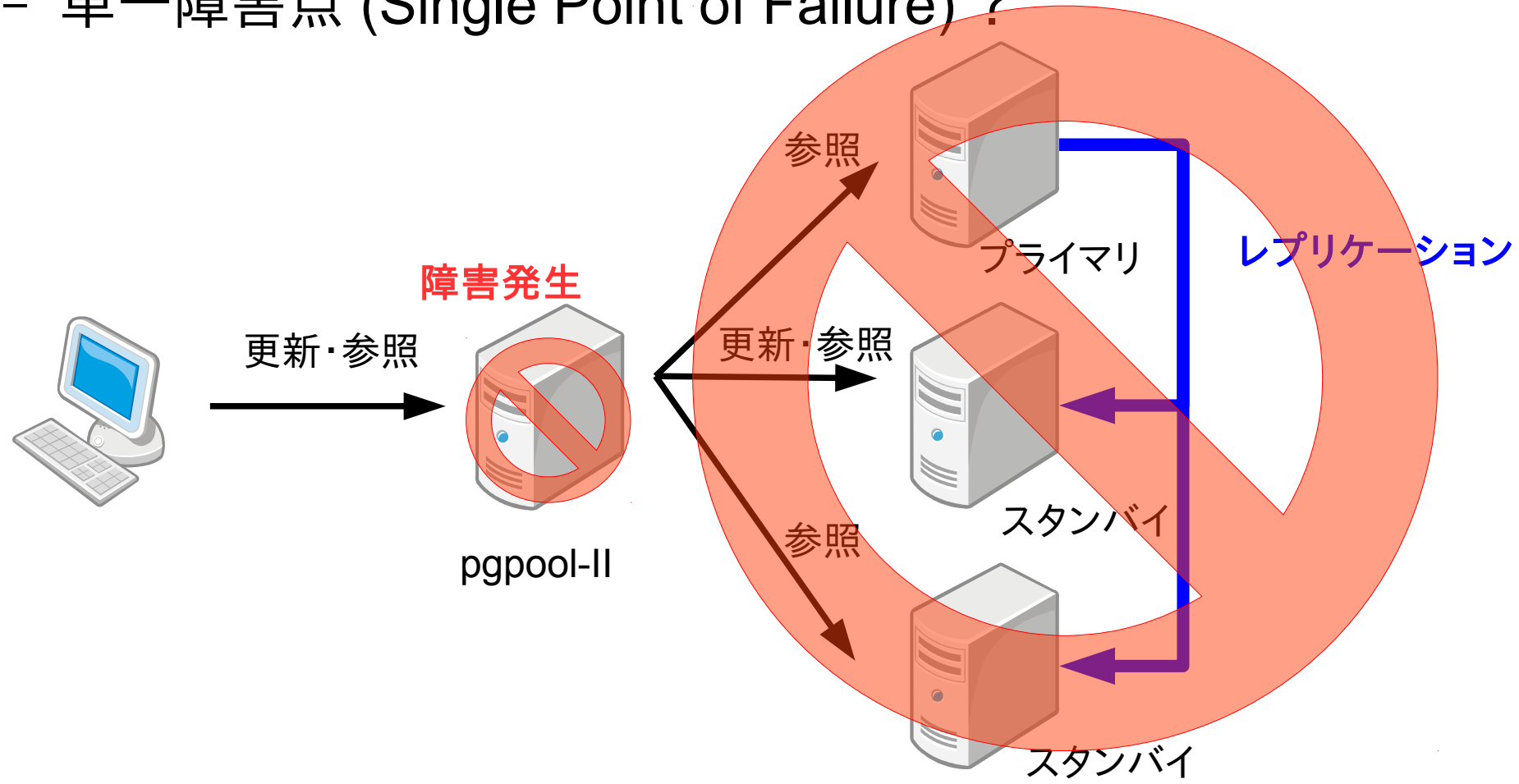


ここまでのまとめ

- 参照性能の負荷分散
  - 更新クエリと参照クエリの適切な振り分け
  - アプリケーション名やデータベース名での振り分け
- 自動フェイルオーバ
  - データベース障害の自動検出＆切り離し
  - プライマリがダウンしたら、スタンバイが新プライマリに昇格
- オンラインリカバリ
  - サービスを停めずにダウンしたサーバを復帰
  - 新しいスタンバイの追加も簡単

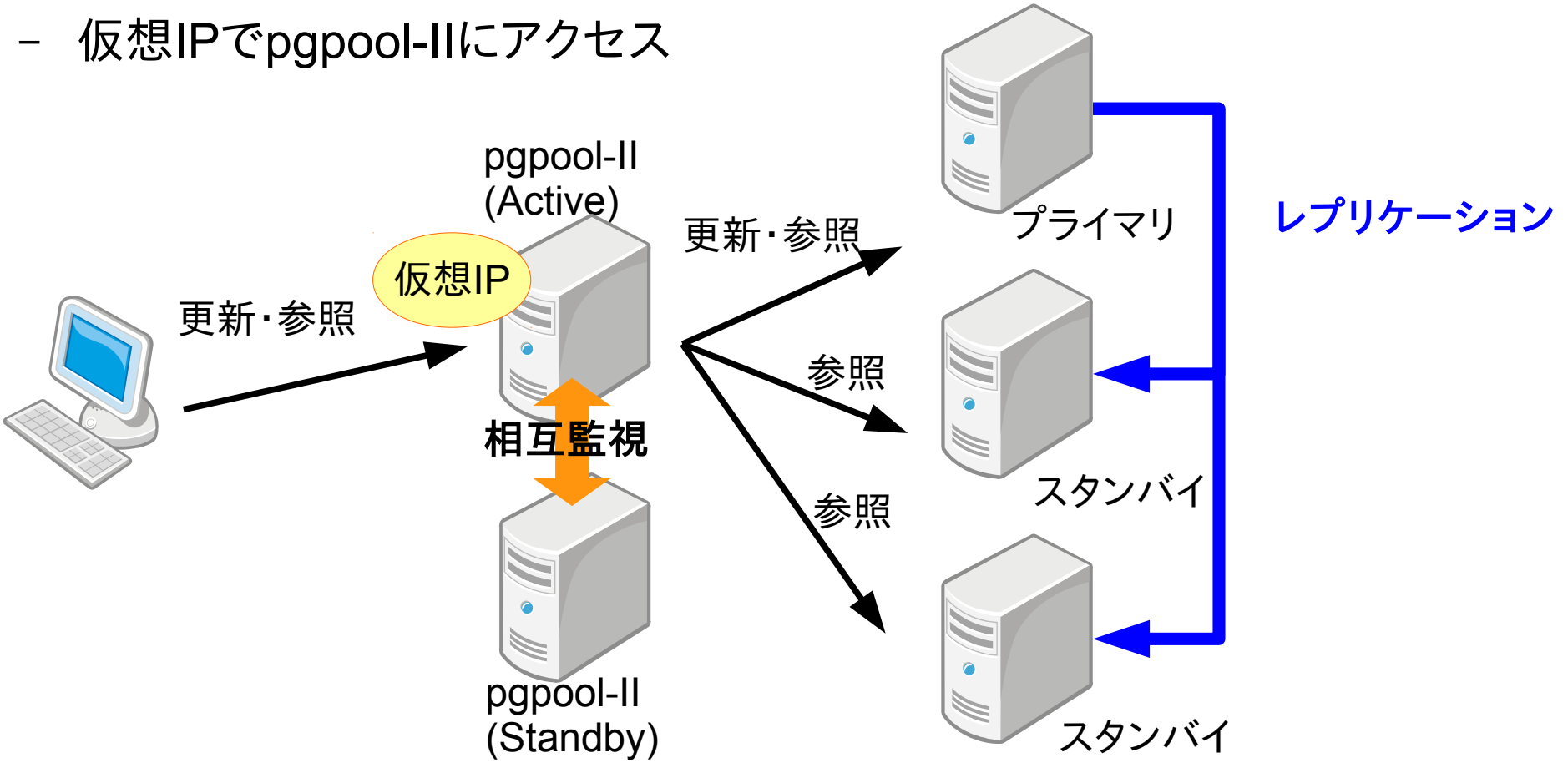
pgpool-II の単一障害点回避

- もし、pgpool-II に障害が発生したら?!
  - 単一障害点 (Single Point of Failure) ?



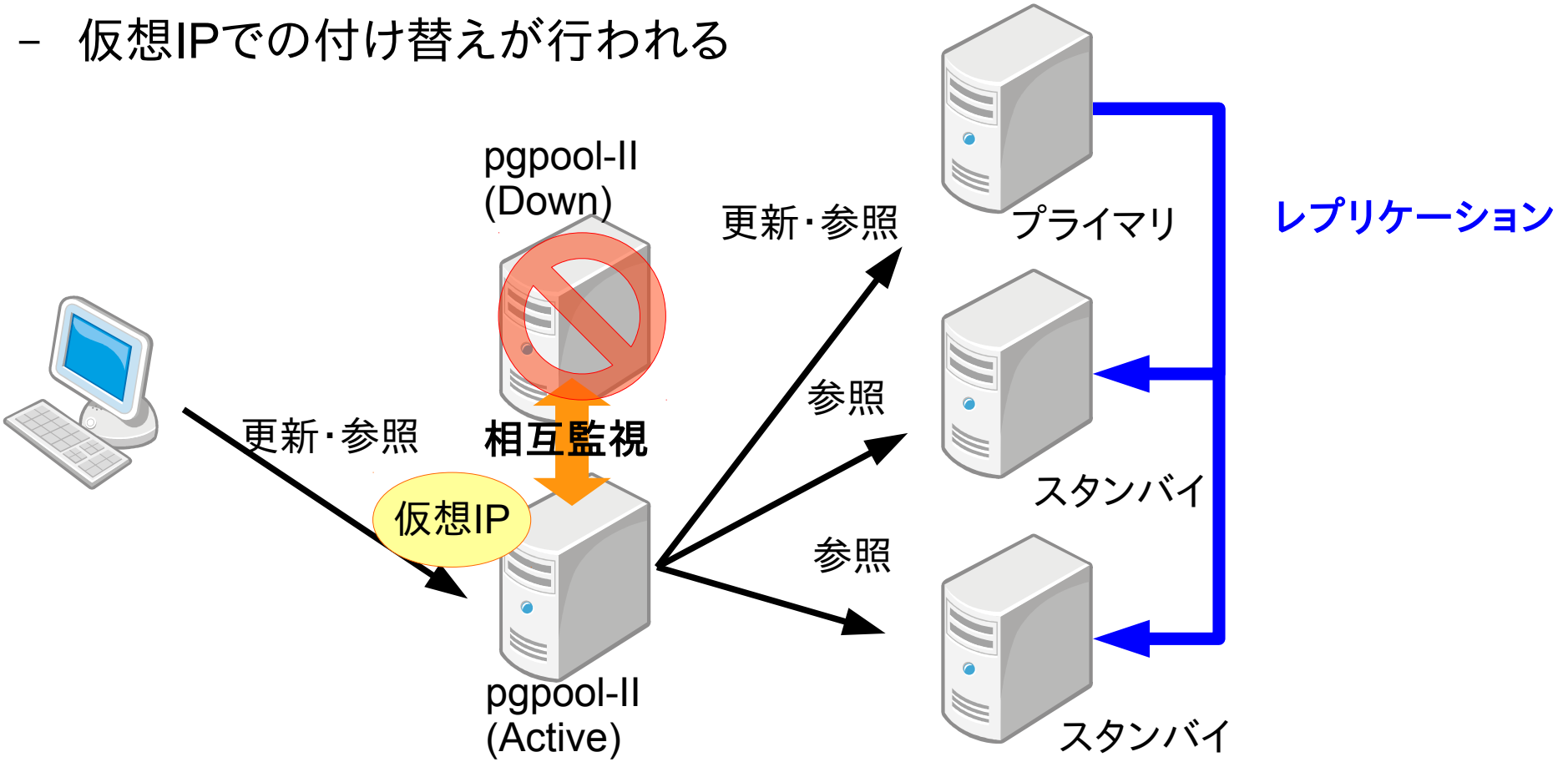
watchdog

- pgpool-II 組み込みのHA機能
  - pgpool-II を Active/Standby 構成にする
  - 仮想IPでpgpool-IIにアクセス



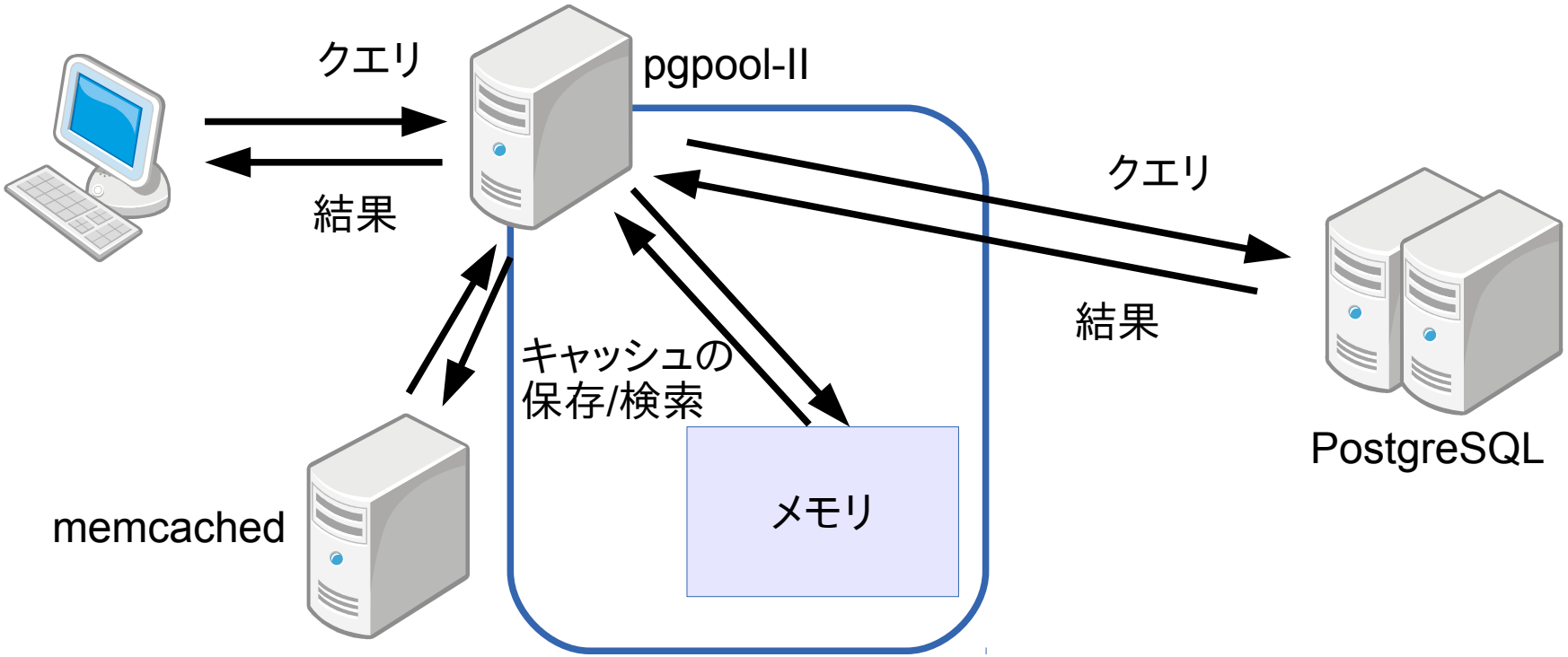
watchdog

- Active pgpool-II に障害発生すると・・・
  - Standby pgpool-II が Active に昇格
  - 仮想IPでの付け替えが行われる



インメモリクエリキャッシュ

- SELECTクエリの結果をメモリ内にキャッシュする機能
  - 同じクエリが来たときに再利用する
  - DBへのアクセスが減り、応答速度が向上

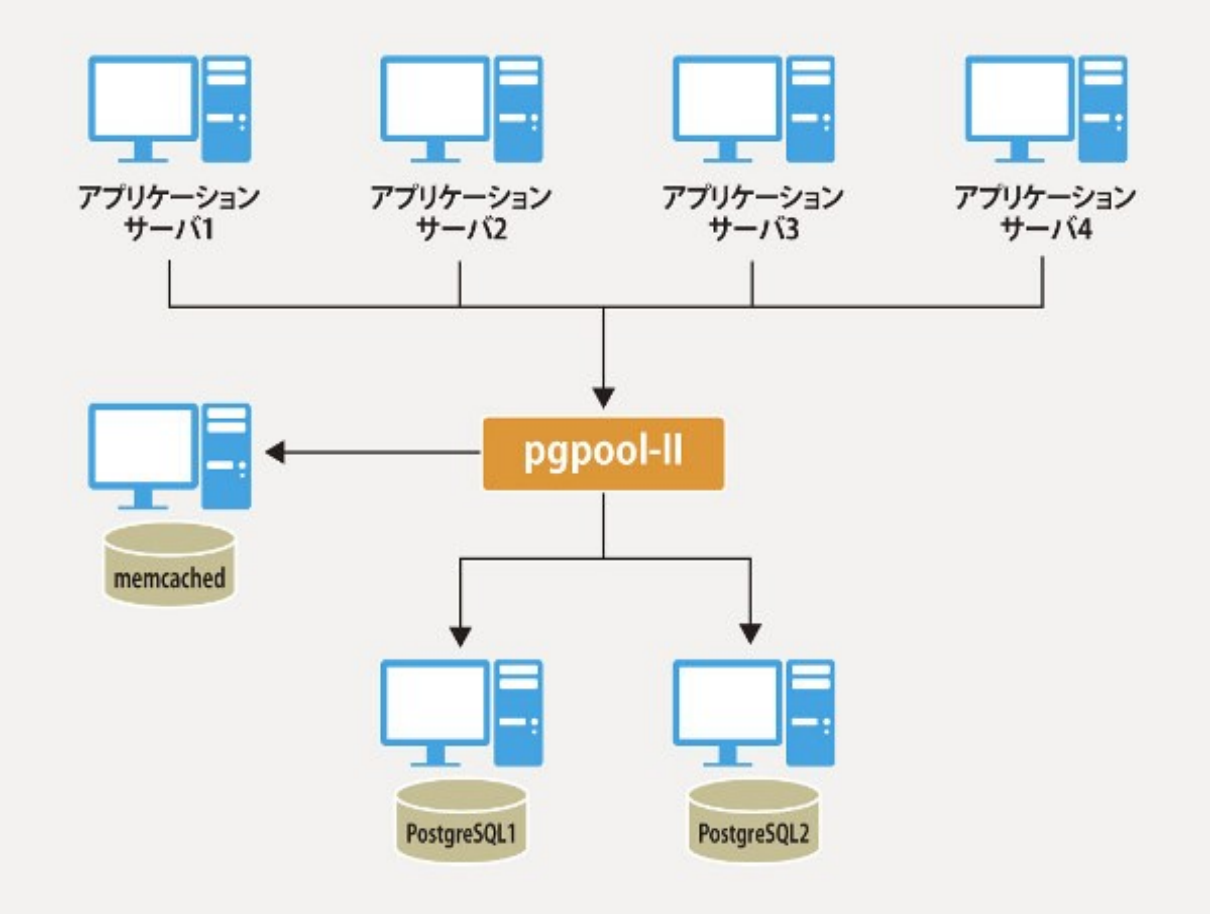


pgpool-II 機能まとめ

- 性能向上
  - 参照負荷分散
  - インメモリキャッシュ
- 信頼性向上
  - 自動フェイルオーバ
  - Watchdog（= pgpool-II 自体の高可用化）

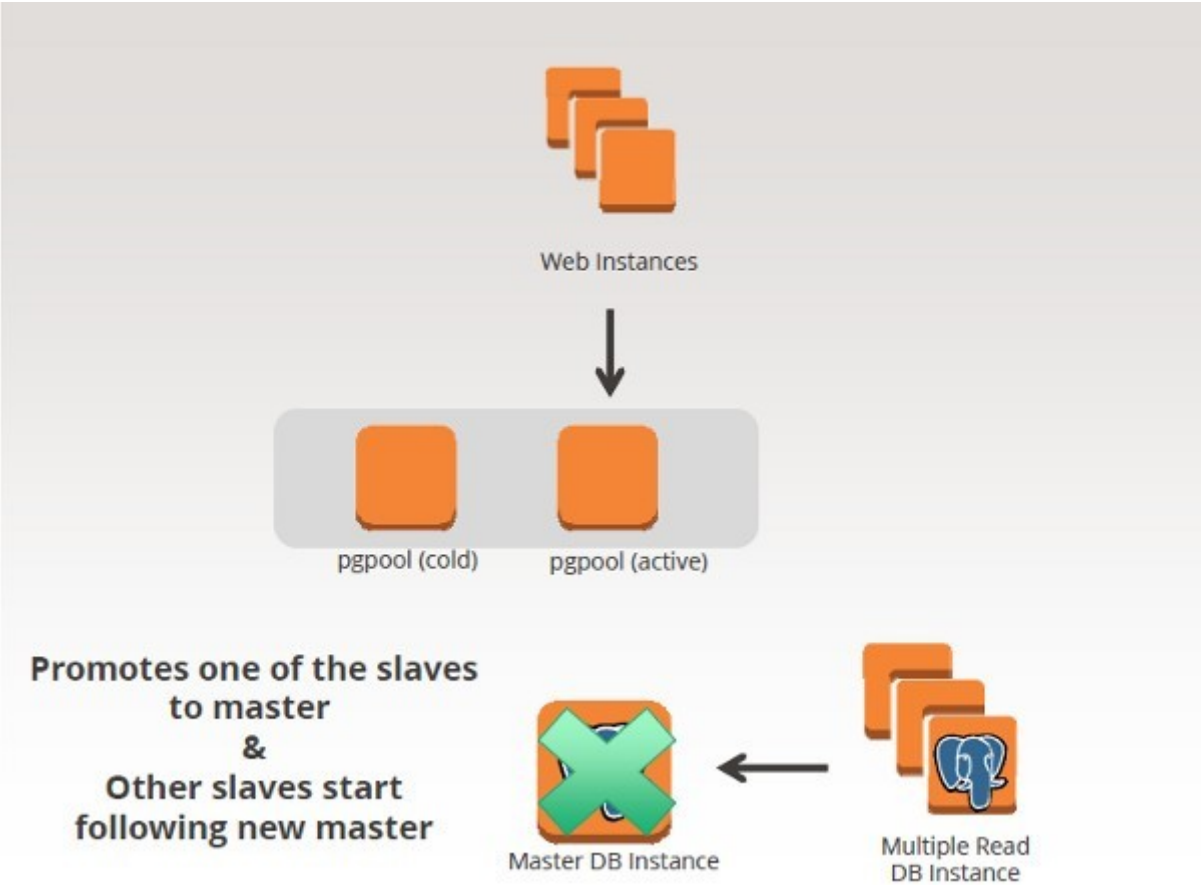
- その他クラスタ管理など
  - クエリの自動振り分け
  - オンラインリカバリ

第一法規株式会社様 事例



- 大量の判例などを検索するシステム
- PostgreSQLのストリーミングレプリケーション + pgpool-II で、負荷分散による性能向上、可用性向上
- インメモリクエリキャッシュ機能を活用して検索性能を向上
- 一度発生したDB障害でもpgpool-II の自動フェイルオーバ機能により、サービスは停止することなく継続できた。

株式会社 Gengo様 事例



- 翻訳サービスのクラウドソーシング企業
- AWS上でシステム構築
- 3万トランザクション/日
- 同時にPostgreSQLのバージョンアップ。オンラインリカバリ機能を活用しダウンタイムを最小限に。
- AWSによる強制インスタンス再起動メンテナンスも自動フェイルオーバ機能で乗り切った

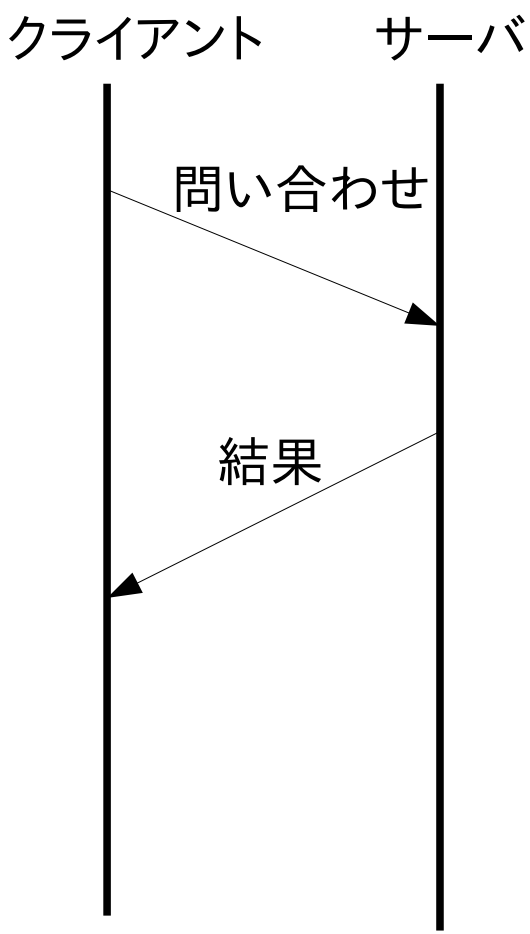
pgpool-II 3.5.0

- 12月にリリース予定

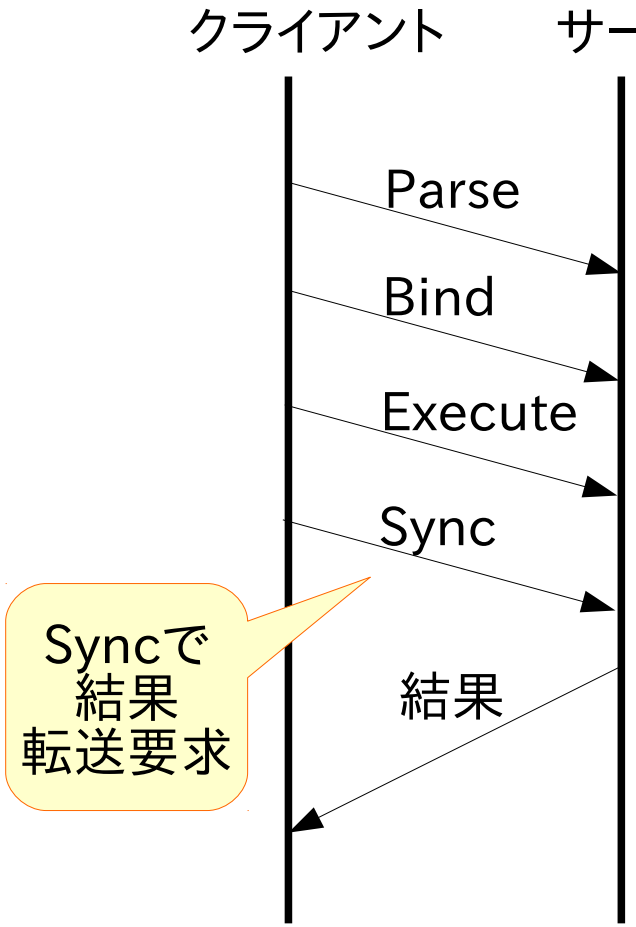
- 主な変更点
  - 性能改善
  - watchdog 機能の改善
  - PostgreSQL 9.5 対応
  - その他

拡張問い合わせプロトコル性能改善(1)

- 単純問い合わせ



- 拡張問い合わせ



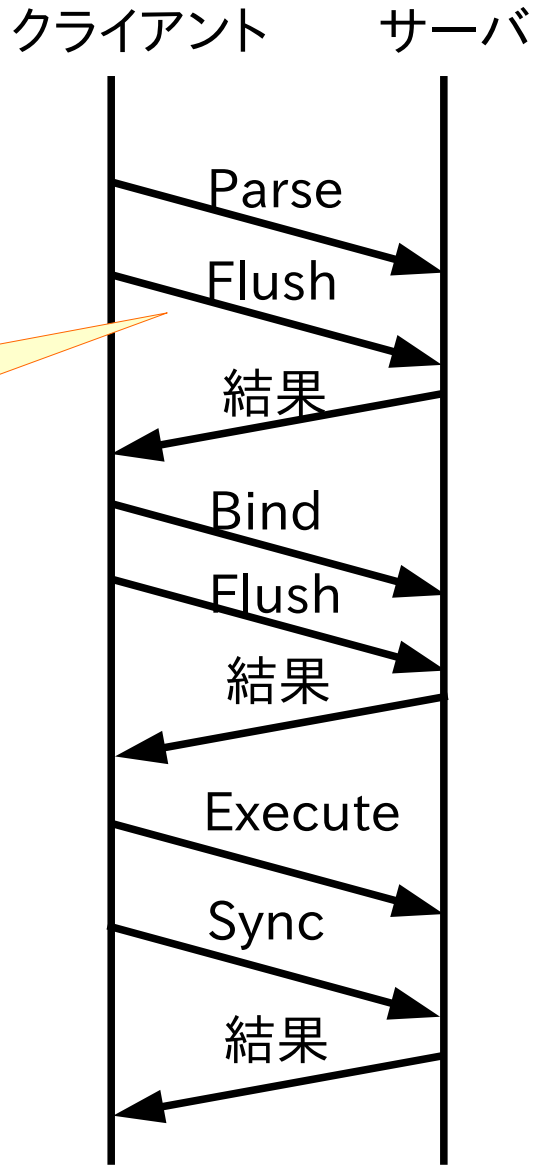
- 複数の段階に分けて処理
  - SQLの解析
  - パラメータ値の結び付け
  - 実行
- Java アプリケーションの JDBC ドライバで使用される

一部やり取りを省略しています

拡張問い合わせプロトコル性能改善(2)

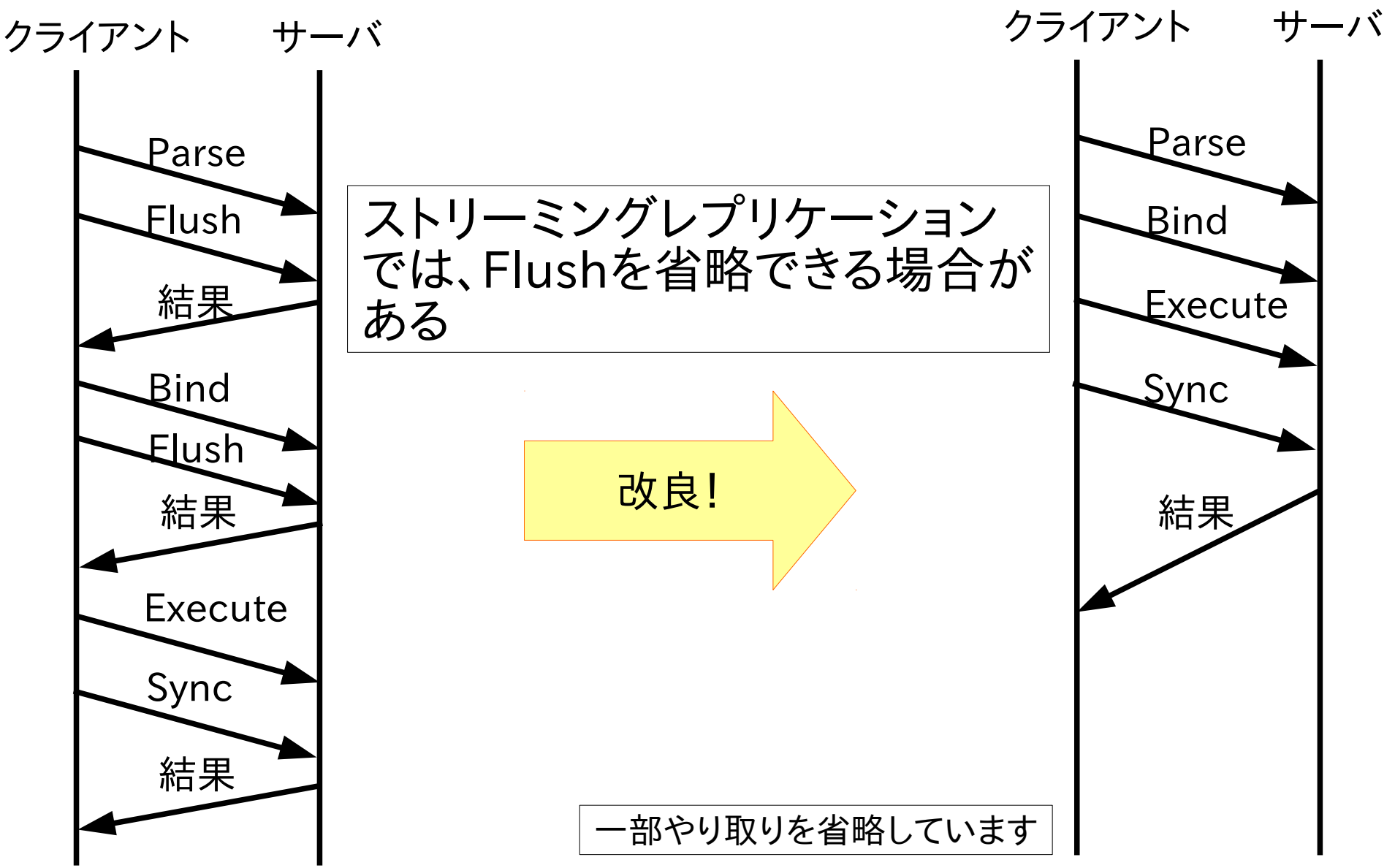
- 今の pgpool-II は拡張問い合わせ使用時の性能が悪い
  - 最悪単純問い合わせ使用時の半分位の性能となってしまう

複数 PostgreSQL の状態を確認するために Flush が必要

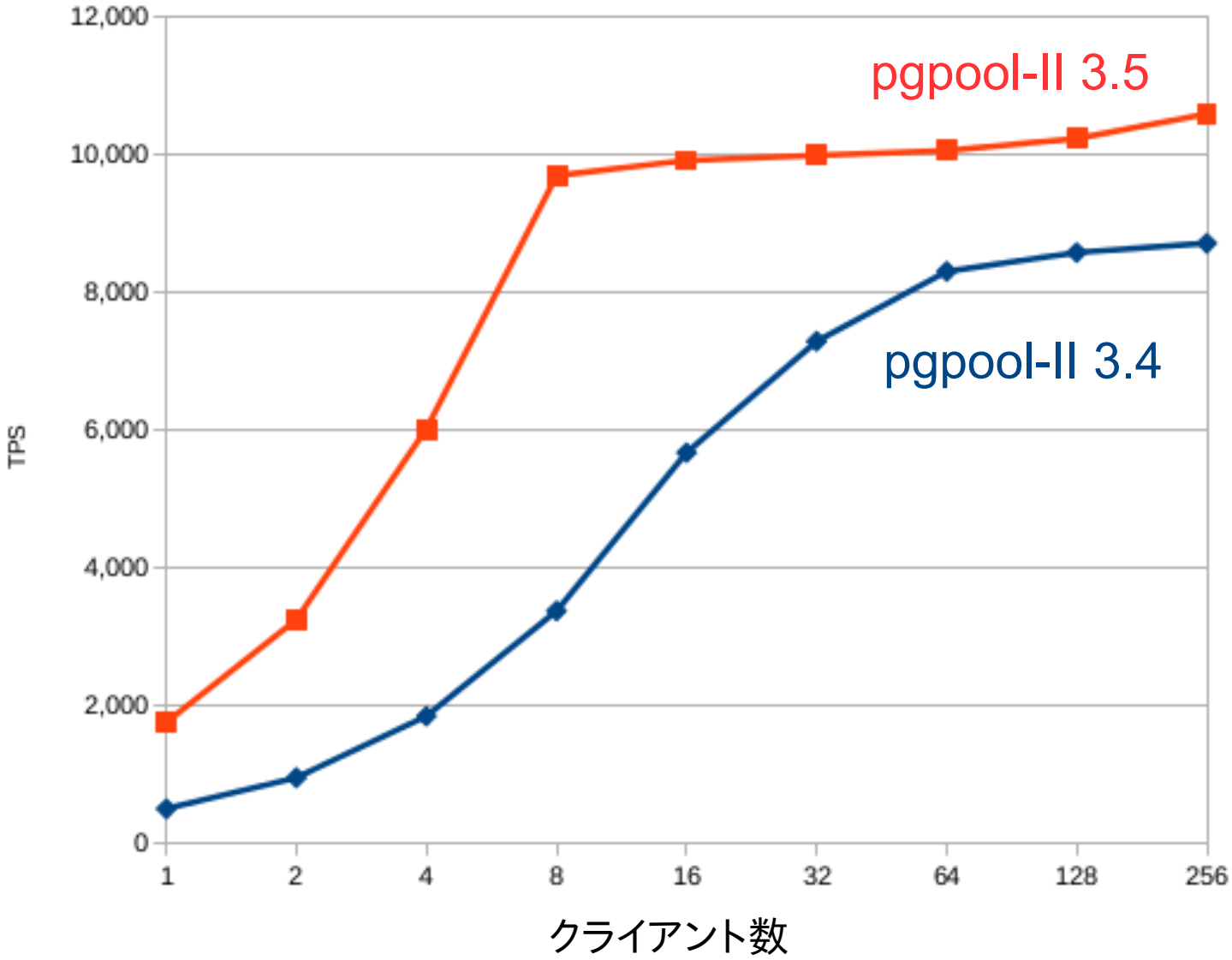


- 性能劣化の原因
  - Flush の発行回数が多い
  - PostgreSQL との通信が増えてしまう

拡張問い合わせプロトコル性能改善(3)



拡張問い合わせプロトコル 性能改善の結果



20% ～ 250% の性能向上!

AWS m4.large instance
CentOS 6
PostgreSQL 9.4 x2
（streaming replication）
pgbench -S

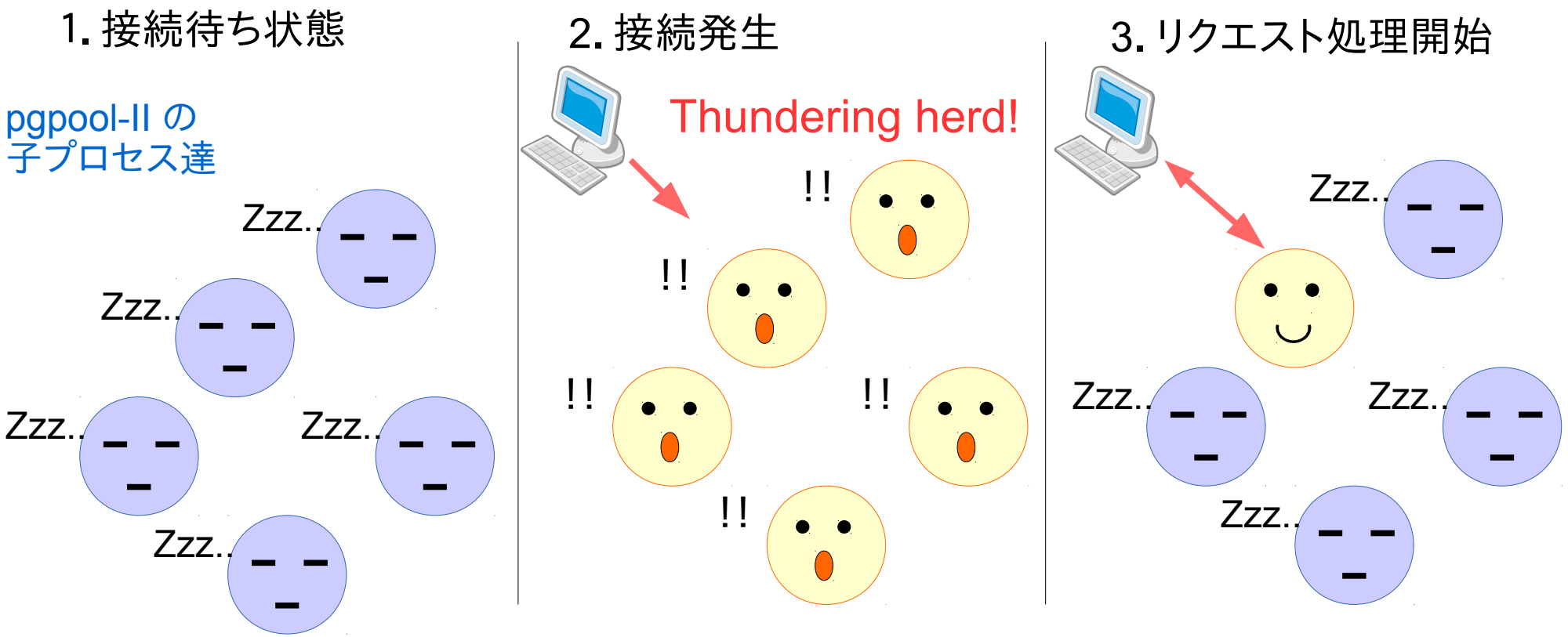
Thundering herd 問題への対応

- Thundering herd（獣群の暴走）問題とは
  - あるイベントを待機している多数のプロセスが、イベントの発生により一度に起こされる
  - プロセスが起こされた後に、その後の処理を続けるプロセスが1つだけ決められる
  - 残りのプロセスは再び眠りにつく
  - 多くのプロセスが、一斉に目覚めては眠ることの繰り返し・・・

  → CPU 資源の無駄遣い！

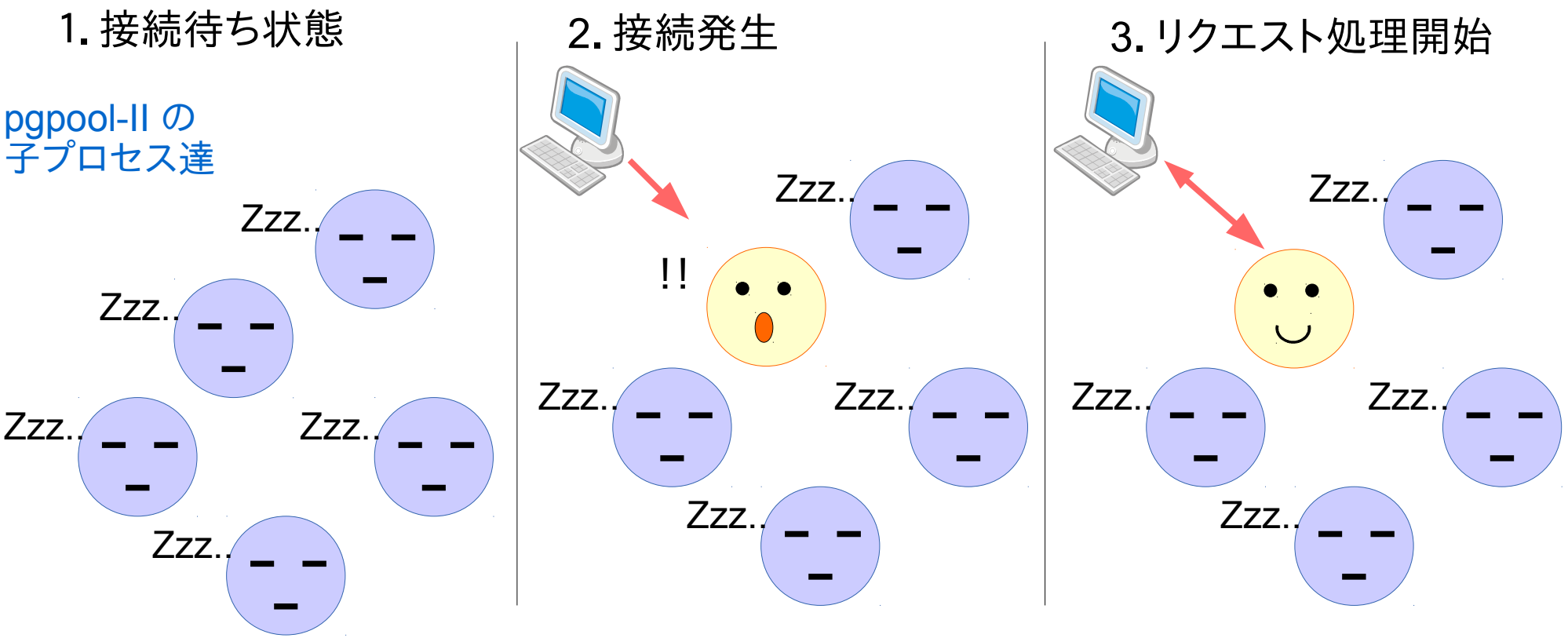
pgpoo-II の場合

- 起動時に fork した複数の子プロセスがクライアントの接続を待ち受けている
  - Thundering herd 問題が発生

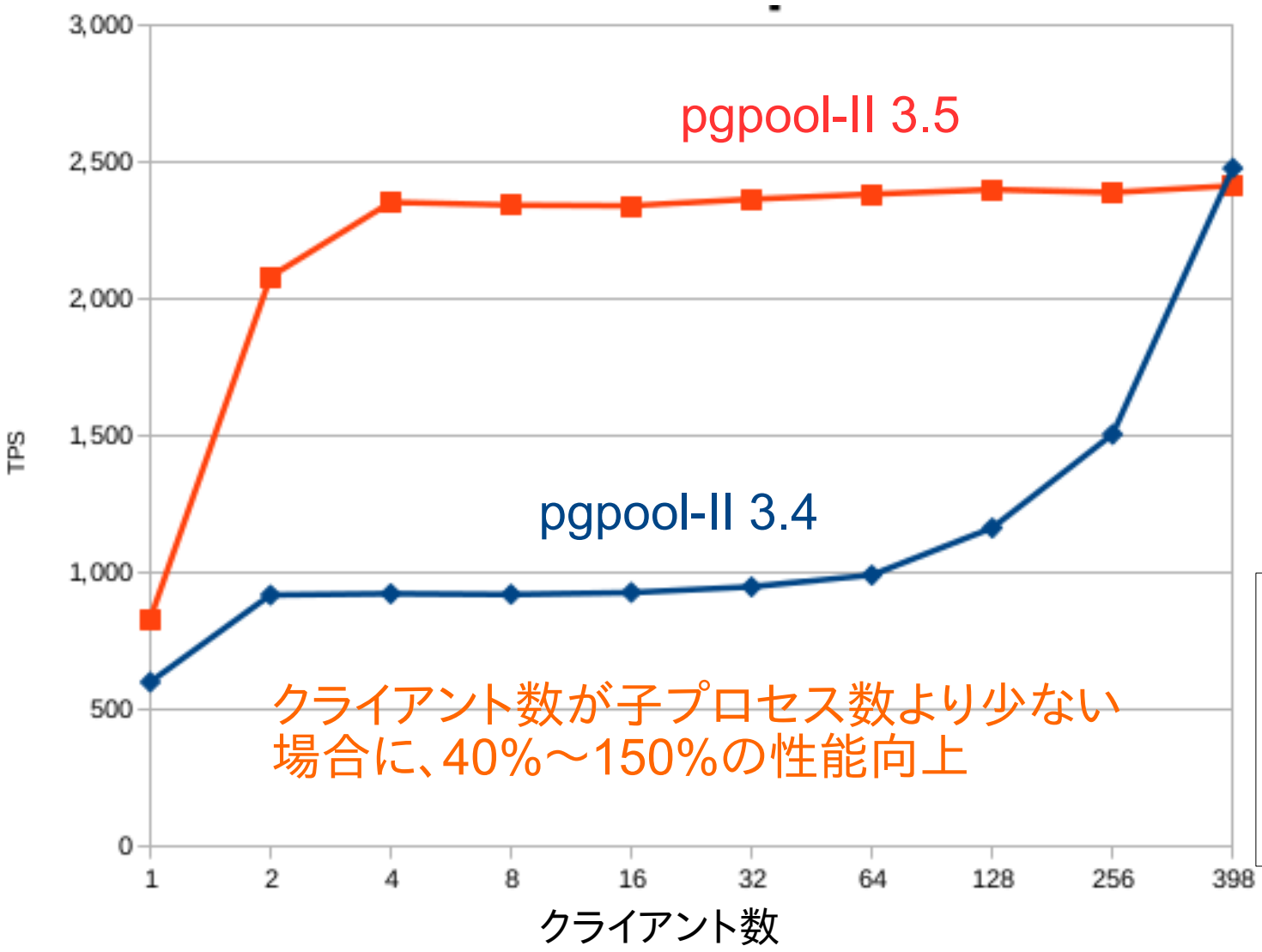


pgpool-II 3.5 で解決

- 接続待ち受けをシリアライズを有効にする新パラメータを追加
  - serialize_accept = on
  - Thundering herb 問題を回避



Thundering herd 問題対応 性能改善の結果



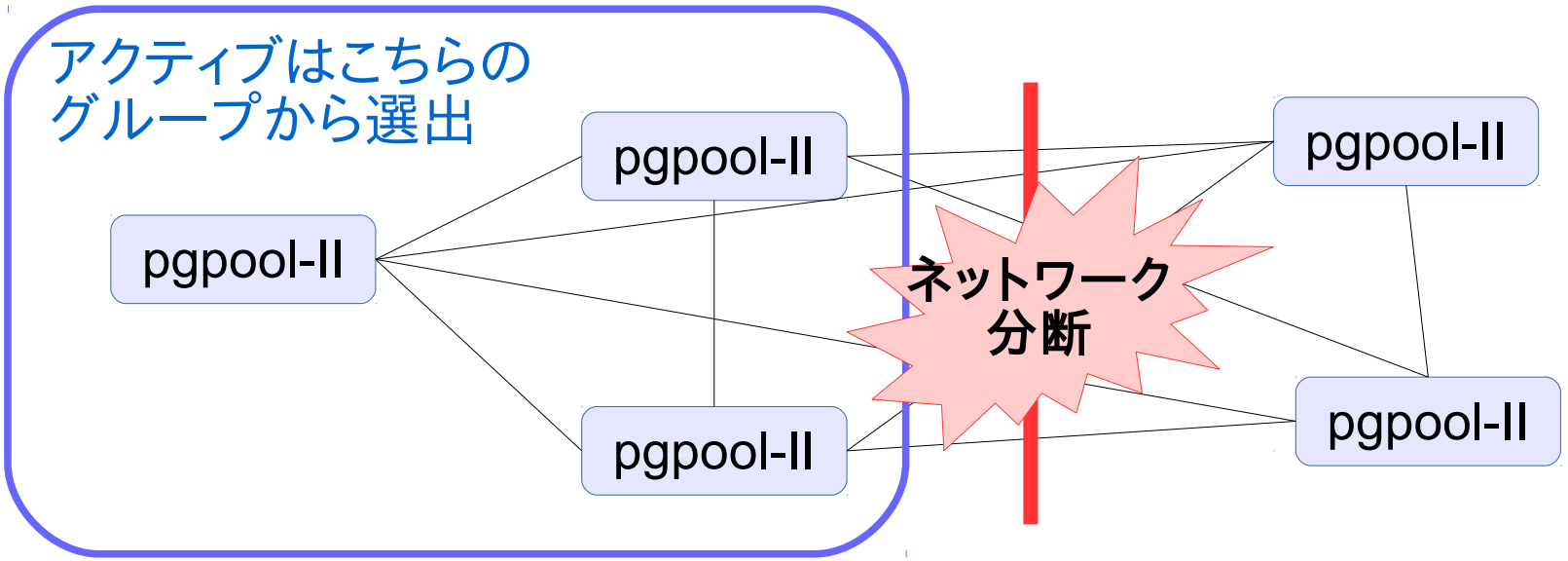
Note PC with 16GB Mem,
CORE i7 x2, 512GB SSD
Ubuntu 14.04
PostgreSQL 9.4 x2
(streaming replication)
pgbench -S -C -T 300

Watchdog 機能の改善

- 内部コードの改善
  - よりデバッグ、メンテナンス、機能拡張がしやすいコードへ
    - 「状態マシン」モデルで全面的に書き換え
- スプリットブレイン対策
  - Quorum のサポート
- プロセス間通信方式の変更
  - UNIXドメインソケット、JSON の採用
  - 外部ツールを使った死活監視
- ノードの優先度
  - pgpool-II が「アクティブ」に選ばれる「優先度」を設定可能

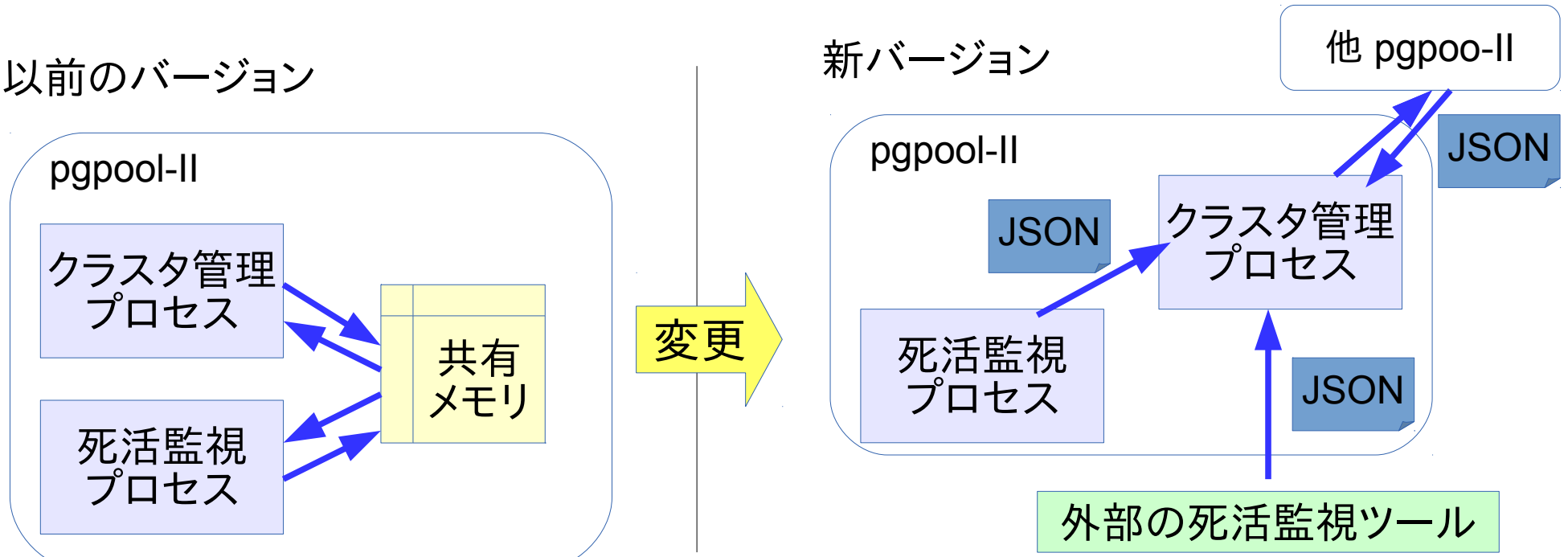
スプリットブレイン問題への対応

- スプリットブレイン問題
  - ネットワークが分離された時に、どの pgpool-II がアクティブとなるべきか決められなくなる問題
  - Quorum（定足数）の充足を常に保証
    - クラスタに参加している全ノードのうち半数以上が自分と同じネットワークに属しているかどうか、をチェックする



プロセス間通信の改善

- watchdog 内部で行われる、プロセス間通信方式の変更
  - UNIX ドメインソケット & JSON 形式データ
- これにより、外部のサードパーティツールとの連携が可能に
  - 外部ツールを使った死活監視モード（external）を追加



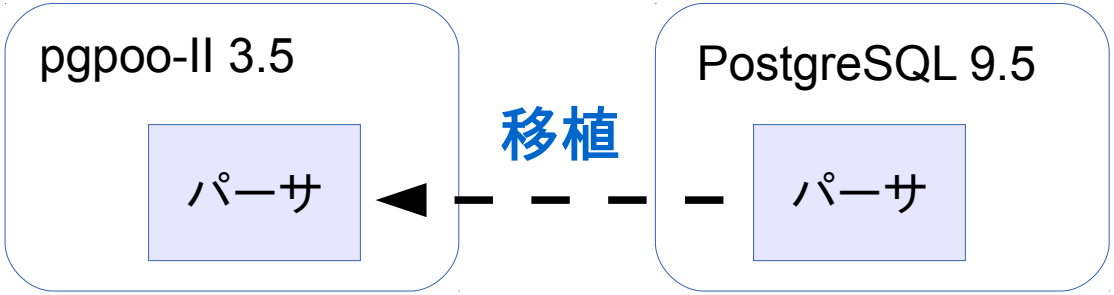
その他のWatchdog 機能改善

- ノードの優先度
  - 各ノードに異なる優先度を付与できる
  - wd_priority パラメータを追加
  - 高い優先度のノードは、マスターに選ばれやすくなる

- ノード間で設定パラメータの一貫性を検証
  - pgpool-II 間で重要なパラメータの値に一貫性を持たせる
  - 設定ミスに起因する問題を軽減

PostgreSQL 9.5 対応

- pgpool-II 3.5 では PostgreSQL 9.5 の SQL パーサを移植



  - 参照負荷分散、クエリキャッシュが、新しい SELECT 構文に対応
    - GROUPING SET, CUBE, ROLLUP
    - TABLESAMPLE
  - レプリケーションモード時のクエリ書き換えが、新しい INSERT/UPDATE 構文に対応
    - INSERT … ON CONFLICT
    - UPDATE tab SET (col1,col2,...) = (SELECT ...), ...

その他の変更

- PCP コマンドの改善
  - 引数の与え方の改善：オプションとして指定可能に

```
旧）  $ pcp_node_info 0 localhost 9898 admin_user admin_pass 0

新）  $ pcp_node_info -h localhost -U admin_user 0
```

  - パスワードをコマンドラインで渡さなくても良いようになった
    - ~/.pcppass ファイルに書いておけばよい
  - 複数pcpコマンドの同時実行
    - たとえば時間のかかる pcp_recovery_node（オンラインリカバリ）の実行中に他のpcpコマンドを実行できる

- パラレルクエリモードの廃止
  - ユーザが少なく、その割にメンテナンスの手間が大きかった

pgpool-II 3.5 まとめ

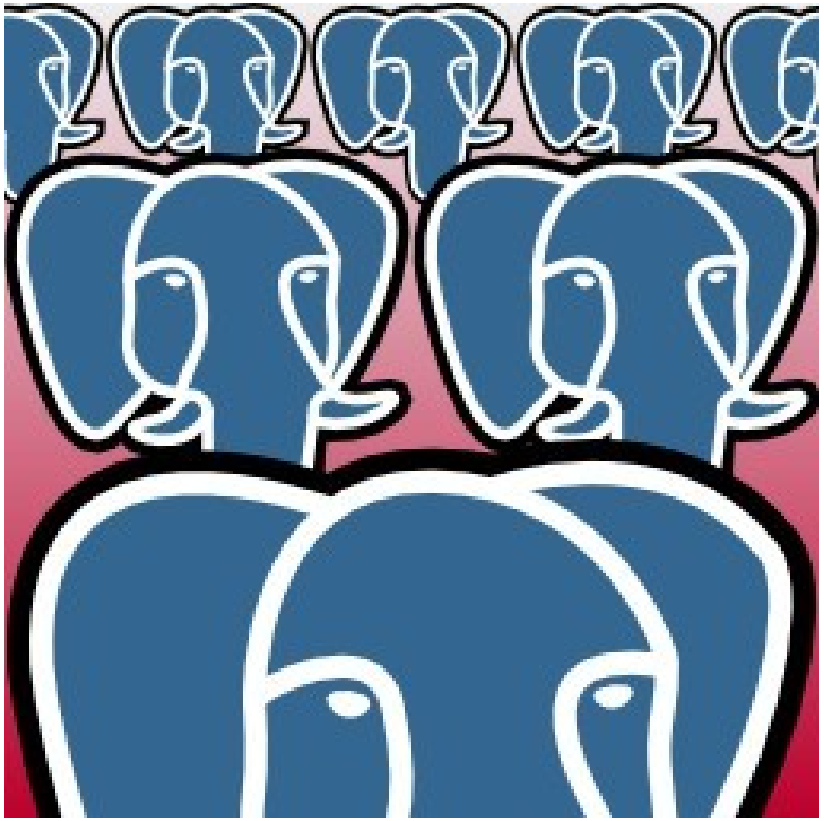
- 性能改善
  - 拡張プロトコル使用時
  - Thundering herd 問題対応
- 信頼性向上
  - Watchdog の改善
  - スプリットブレイン対策
- その他
  - PostgreSQL 9.5 コマンド, pcp コマンド, ・・・

- 今年の冬（12/15）にリリース予定
  - 現在、リリースに向けテスト中
  - alpha1 版が 11/16 にリリース済

参考URL

- pgpool-II オフィシャルサイト
  - http://www.pgpool.net/
  - http://www.pgpool.net/jp/
- SRA OSS, Inc. 日本支社
  - セミナー資料、事例情報、技術情報
  - http://www.pgecons.org/
- Let's Postgres
  - PostgreSQL 情報のポータルサイト
  - http://lets.postgresql.jp/

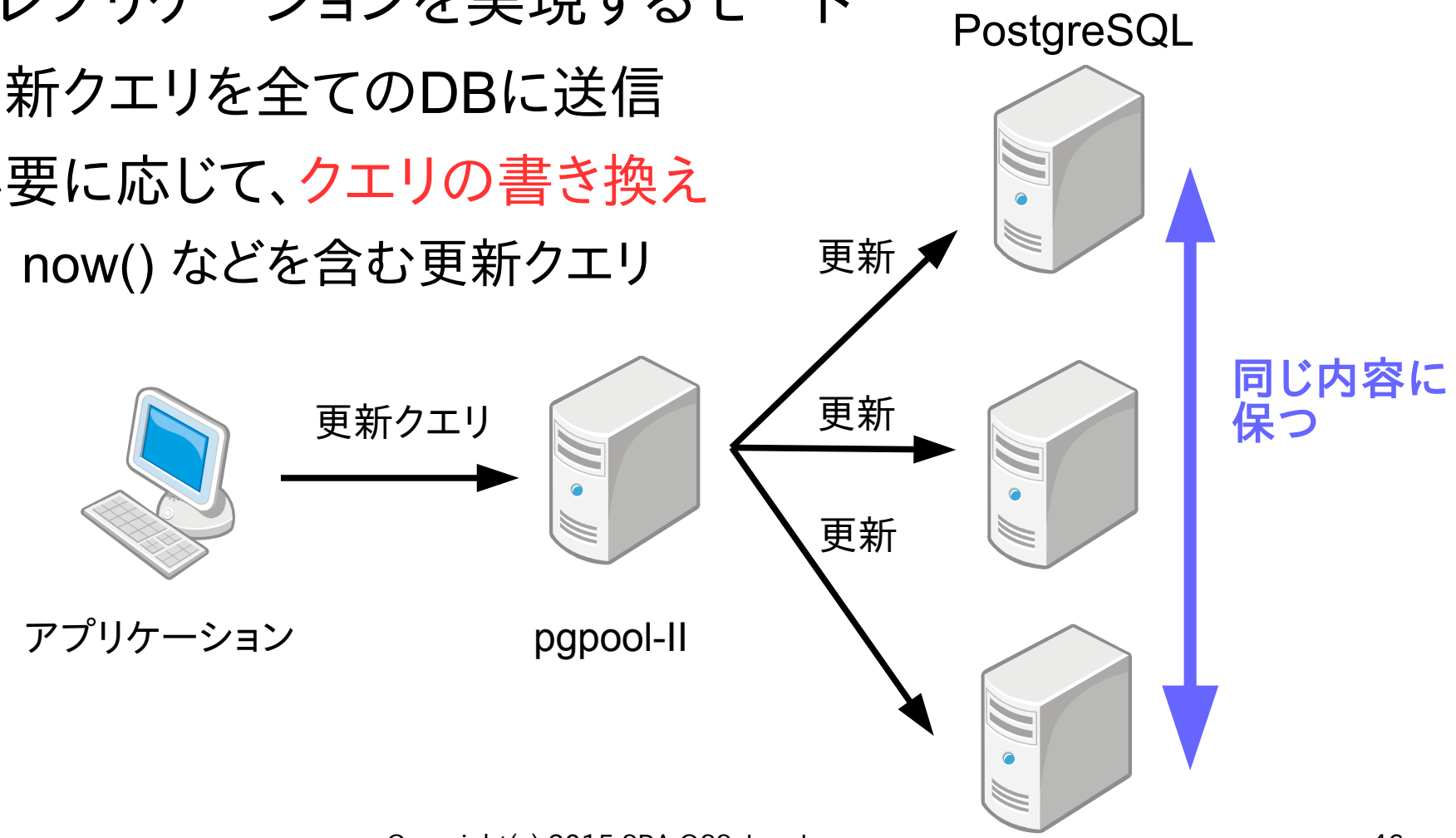
ご静聴ありがとうございました!



付録

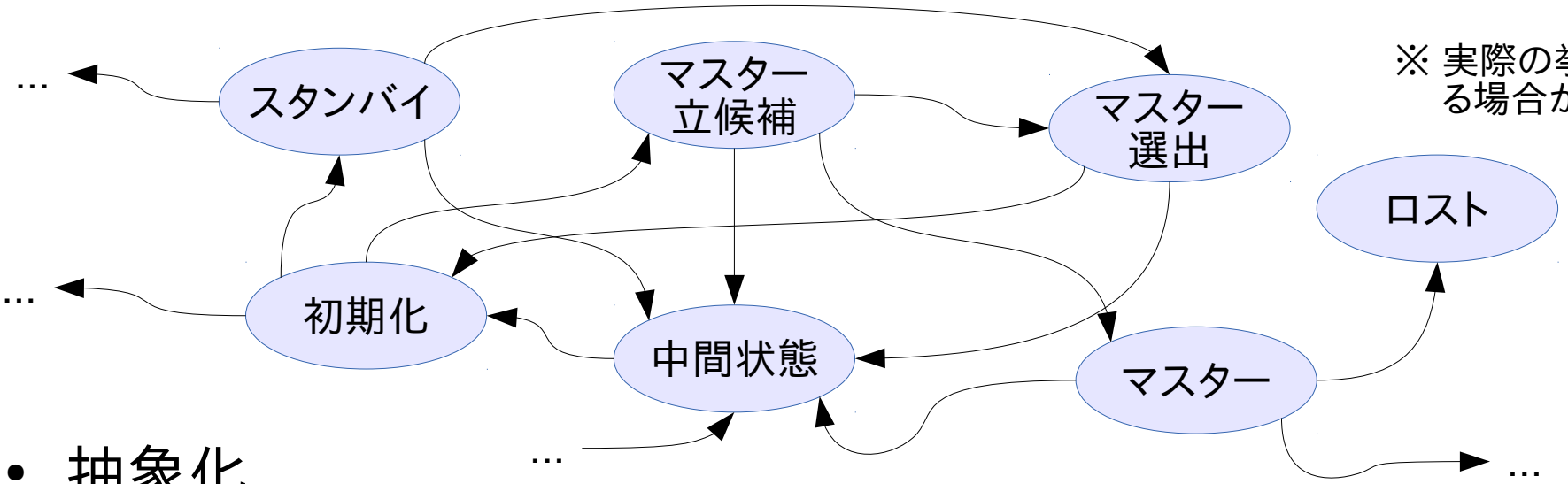
ネイティブレプリケーション

- PostgreSQL のストリーミングレプリケーションを用いずに同期レプリケーションを実現するモード
  - 更新クエリを全てのDBに送信
  - 必要に応じて、クエリの書き換え
    - now() などを含む更新クエリ



Watchdog 改善：状態マシン

- watchdog のクラスタ管理を行うコア部分のコード
  - 「状態マシン」モデルで全面的に書き換え



- 抽象化
  - コードが理解しやすく、デバッグが容易に → 保守性の向上
  - 将来の機能拡張が容易に → 拡張性の向上