Pgpool-IIの軌跡、Pgpool-IIが創り出す未来

2021/12/15

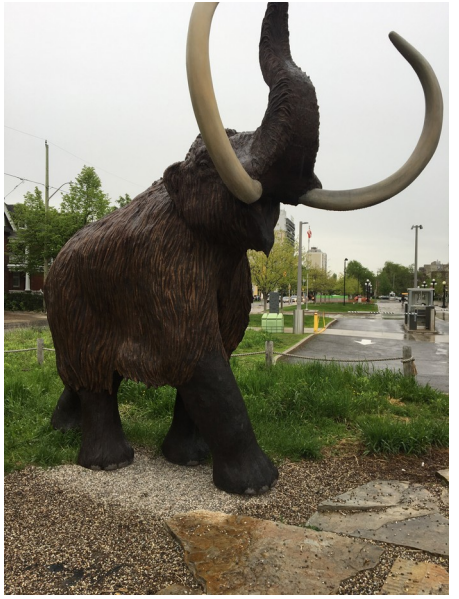
SRA OSS, Inc. 日本支社

石井 達夫

SRA OSS,INC.

自己紹介

- 石井達夫
  - SRA OSS, Inc.日本支社で経営者兼OSS開発者をしています
  - Pgpool-IIの開発者、PostgreSQLコミッタ
  - 趣味は音楽鑑賞、SF、下手な料理



2019年PGCon開催地カナダにて

Pgpool-IIの生い立ち

Pgpool-IIの誕生

- 2003年PostgreSQL用のコネクションプーリングソフトとして公開
- フェイルオーバー機能も同時に実装
- バックアップ用のPostgreSQLサーバの指定も可能
- 現在のRawモードに相当する

その後のPgpool-IIの進化

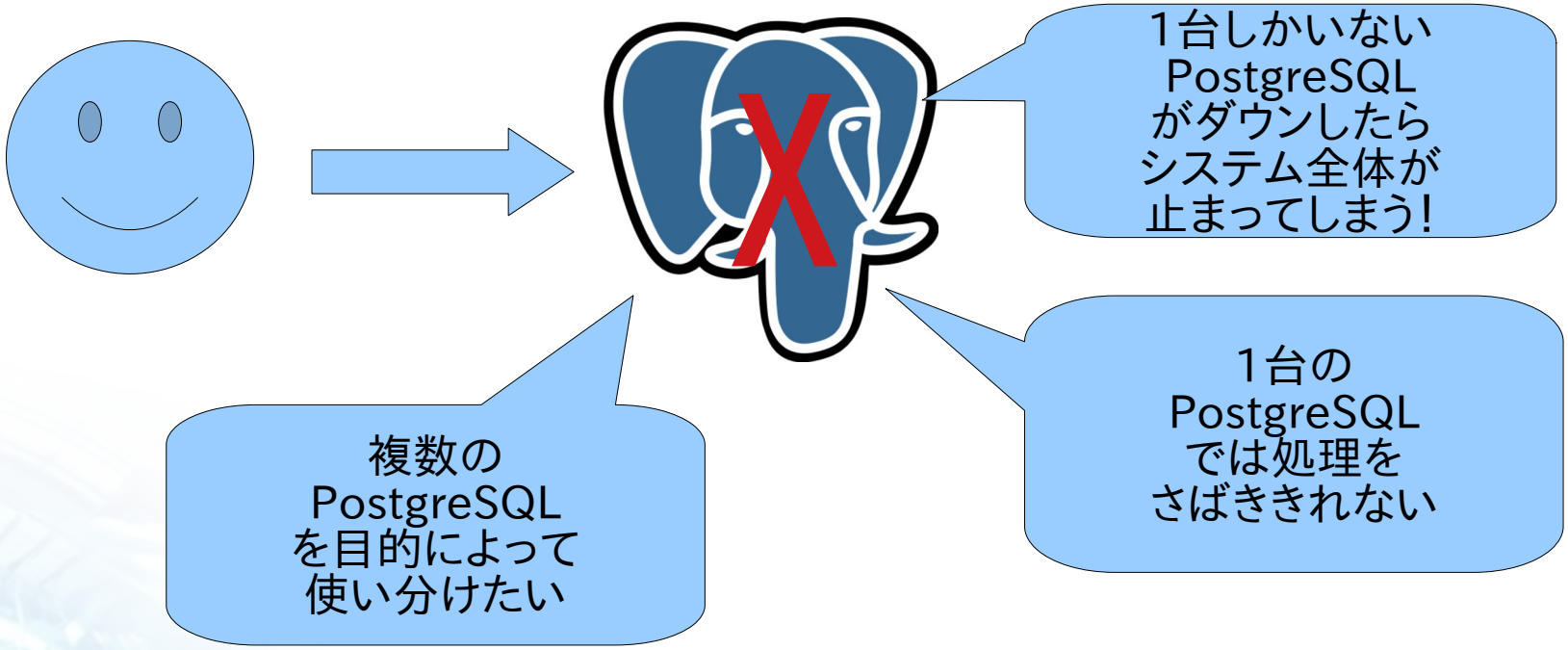
- 2004年、現在の「ネィティブレプリケーションモード」に相当する機能を実装
- 2006年、SQLパーサ搭載、PostgreSQLサーバ台数制限を緩和
- 2010年、ストリーミングレプリケーションへの対応
- 2012年、Watchdogの実装
- 2017年、AWS Aurora対応

その後のPgpool-IIの進化

- 2018年、SCRAM認証、証明書認証対応
- 2019年、文レベルの負荷分散
- 2020年、LDAP認証、ログ収集プロセス
- 2021年、Pgpool-II 4.3リリース
  - 詳細はこの後のセッションで!

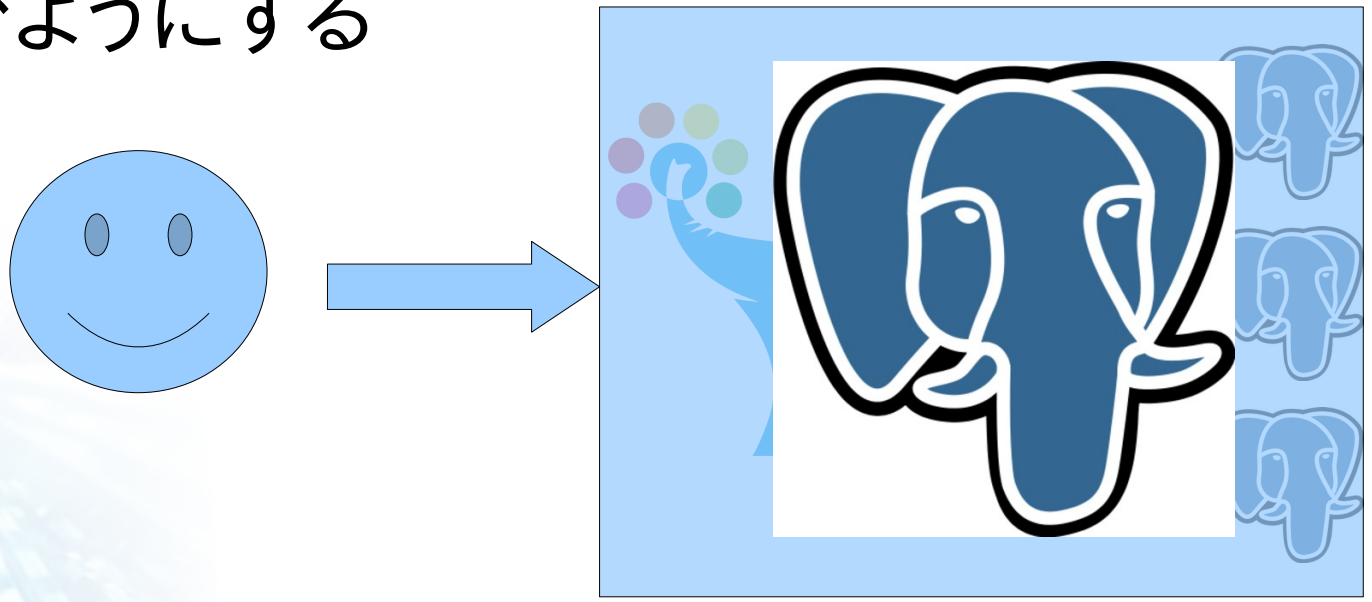
Pgpool-IIの機能早わかり

Pgpool-IIはユーザの課題の解決を目指す

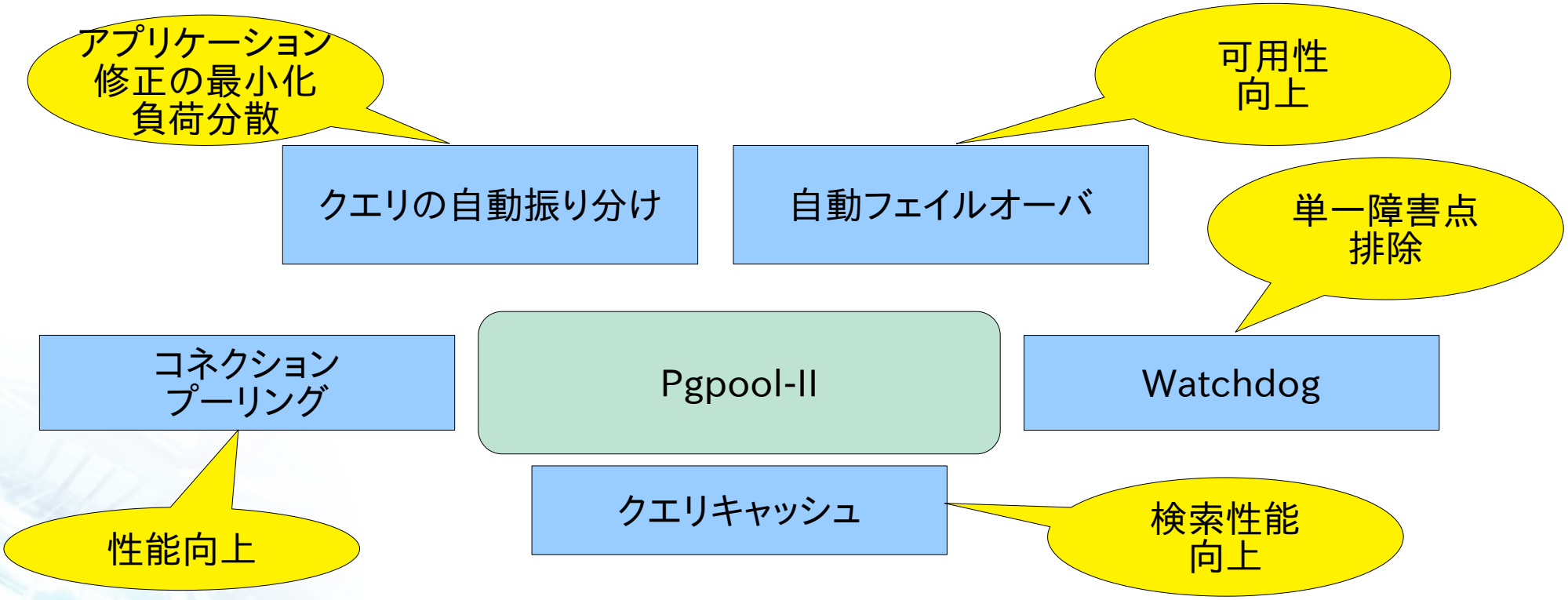


Pgpool-IIのコンセプト

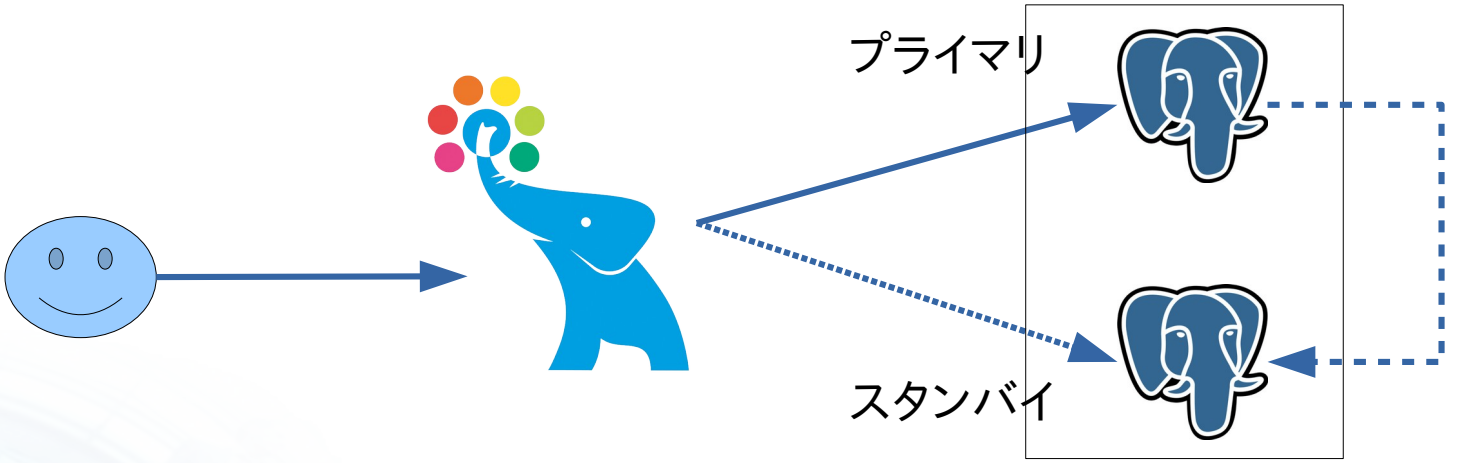
- データベースが複数PostgreSQLから構成されていることをなるべくアプリケーションが意識しないで済むようにする



Pgpool-IIの機能の全体像



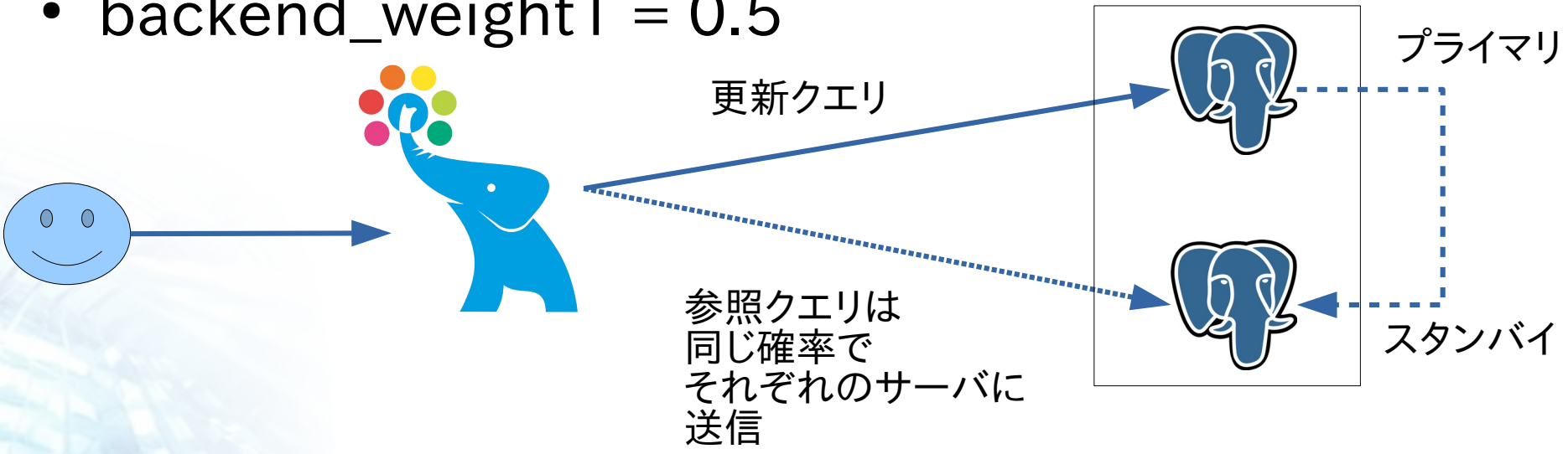
クエリの自動振り分け



ストリーミング
レプリケーション
モード
更新クエリを
プライマリにのみ
投げる

検索クエリの負荷分散

- 負荷分散の重み付けが設定可能
  - backend_weight0 = 0.5
  - backend_weight1 = 0.5

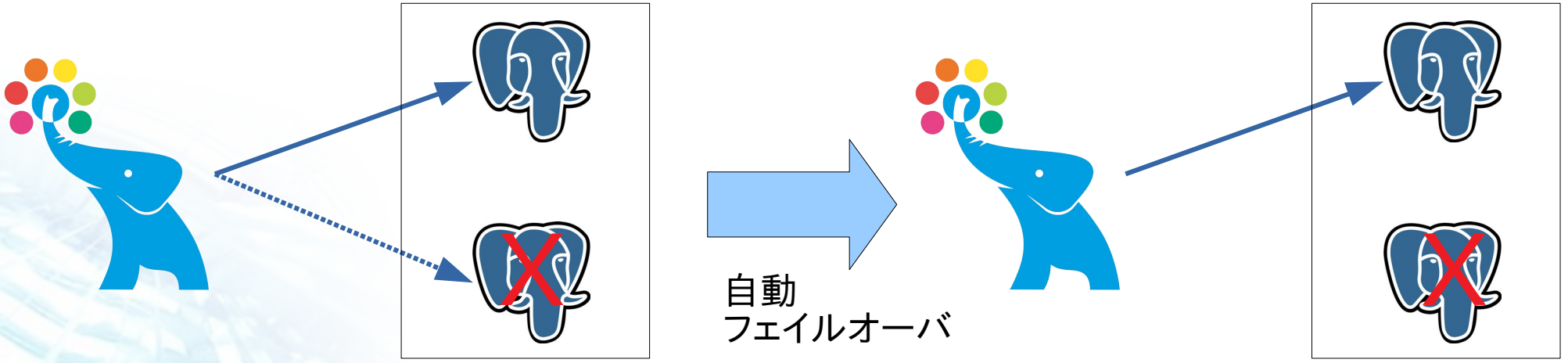


その他のクエリ振り分けの設定

- クエリの振り分けをセッション単位ではなく、SQL文単位で決定
  - statement_level_load_balance
- データベース名によってクエリの振り分け先を決定
  - database_redirect_preference_list
- アプリケーション名によってクエリの振り分け先を決定
  - database_redirect_preference_list
- トランザクション内のクエリの振り分けの挙動を制御
  - disable_load_balance_on_write

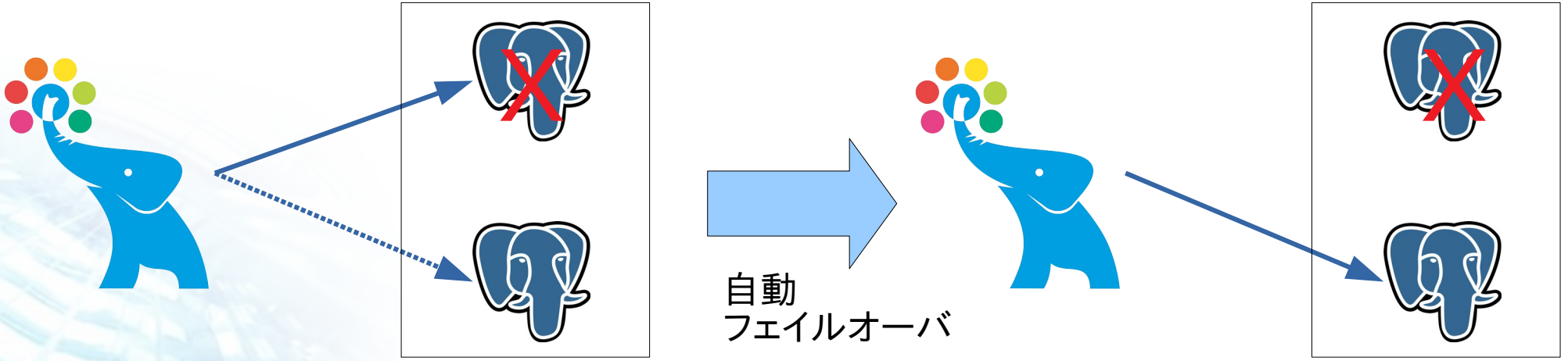
自動フェイルオーバ(1)

- PostgreSQLサーバがダウンしたらそのサーバを切り離し、残りのサーバで処理を継続
  - 定期的にヘルスチェックプロセスがPostgreSQLサーバと通信ができるかどうかを確認
  - 通信ができなければ自動的にそのサーバを切り離す



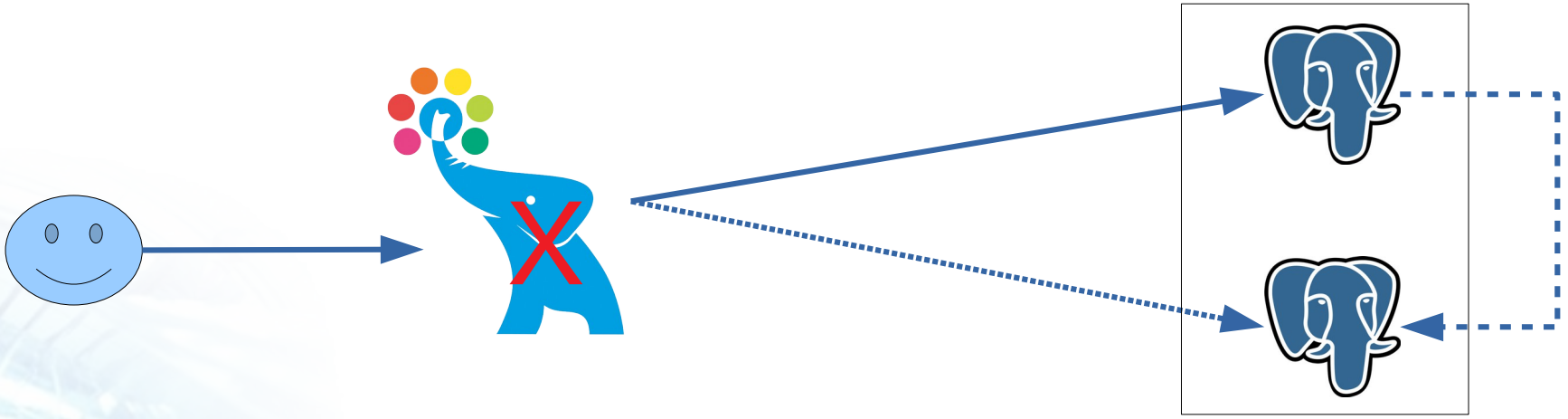
自動フェイルオーバ(2)

- プライマリサーバがダウンした時は、スタンバイサーバを自動昇格して新しいプライマリサーバで処理を続行する
  - 昇格処理を行うカスタムスクリプトを指定できる



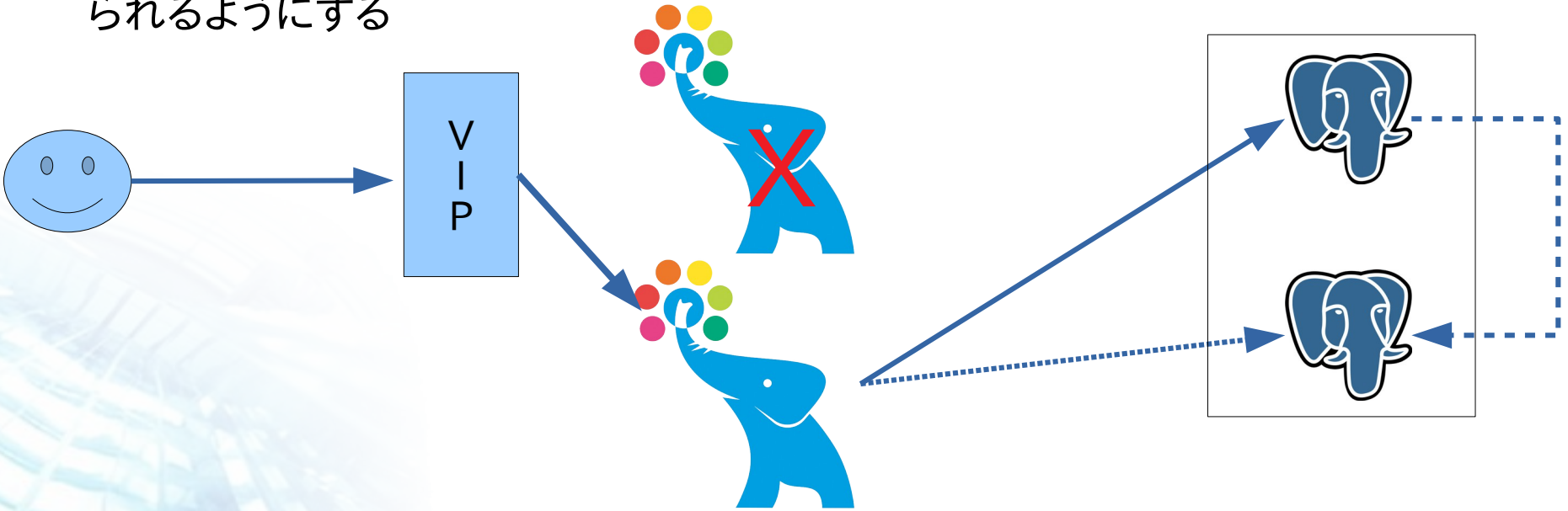
単一障害点排除

- Pgpool-IIがダウンすると、PostgreSQLにアクセスできなくなってしまう（単一障害点、SPOF: Single point of failure)

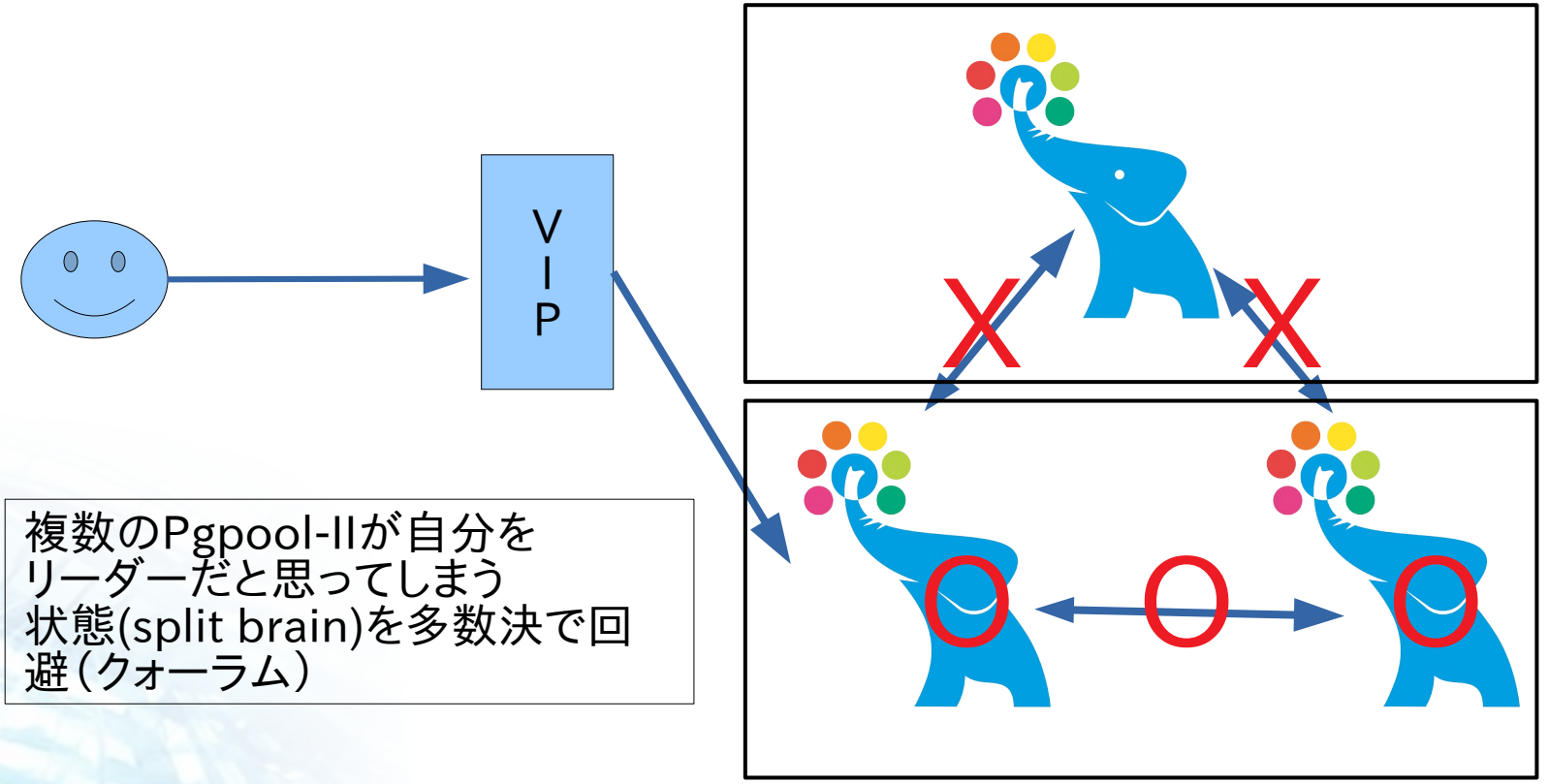


単一障害点排除

- 複数のPgpool-IIを用意してお互いに監視し、リーダーがダウンしたときには待機系のPgpool-IIを昇格して運用を継続する(リーダーがVIPを保持)
- Pgpool-IIへのアクセスはVIP経由にして、クライアント側では同一IPでアクセスし続けられるようにする



Split brainの回避

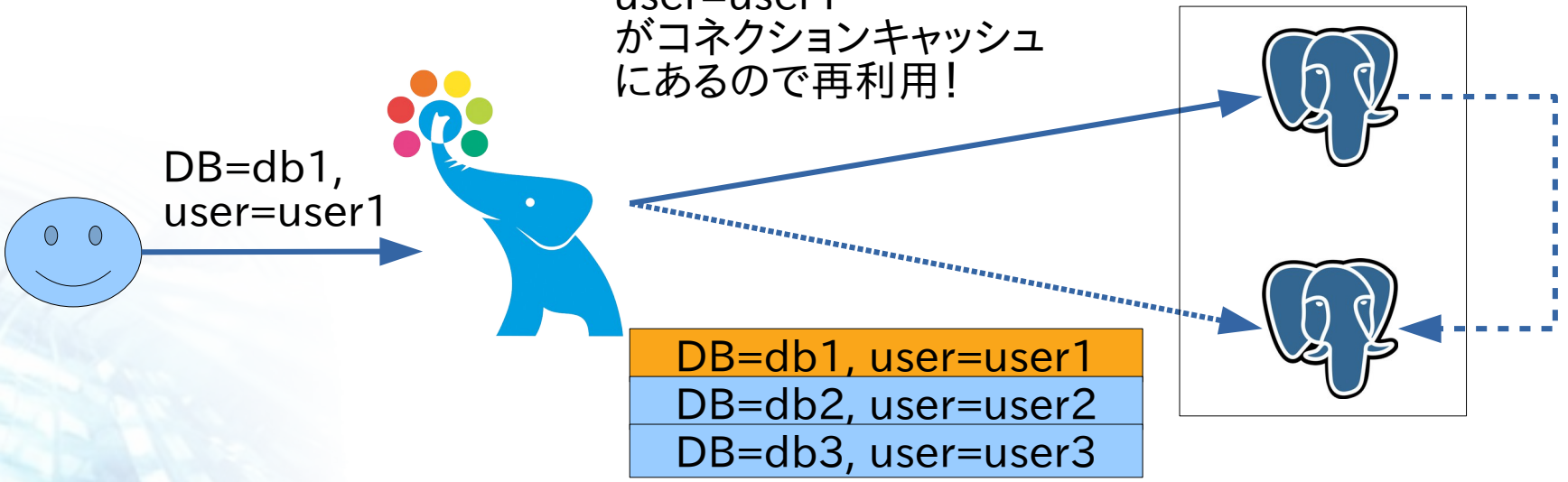


複数のPgpool-IIが自分を
リーダーだと思ってしまう
状態(split brain)を多数決で回
避(クォーラム)

ネットワークの
分割状態
発生!

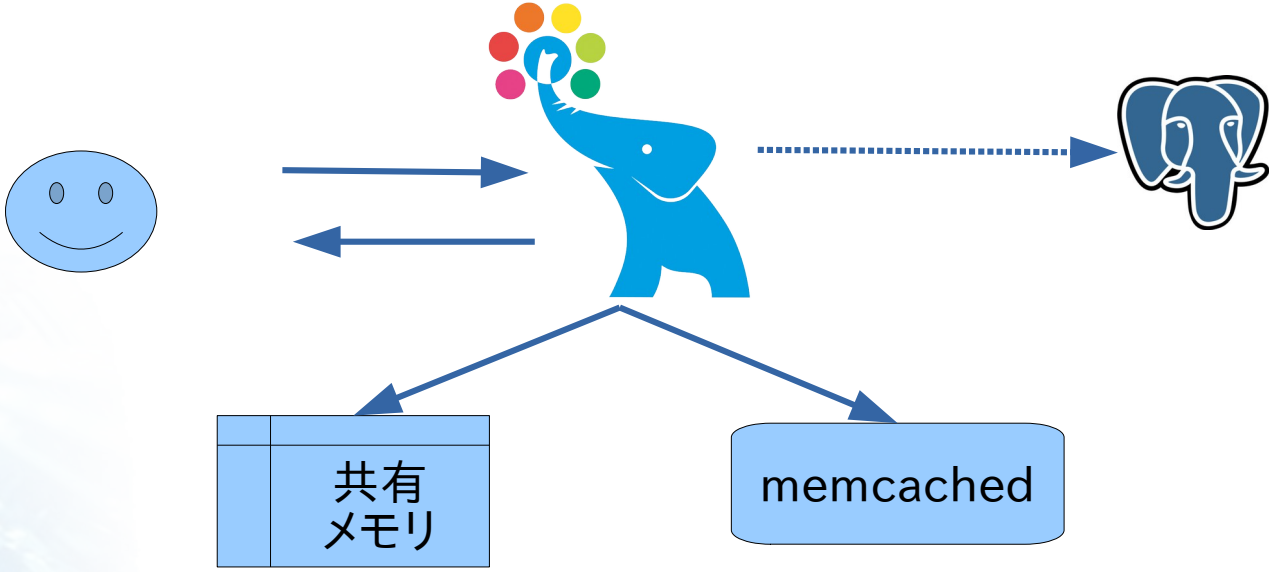
コネクションプーリング

- 一度接続したDBへのコネクションを取っておき、条件が合えば再利用する
- DBへの接続時間を節約



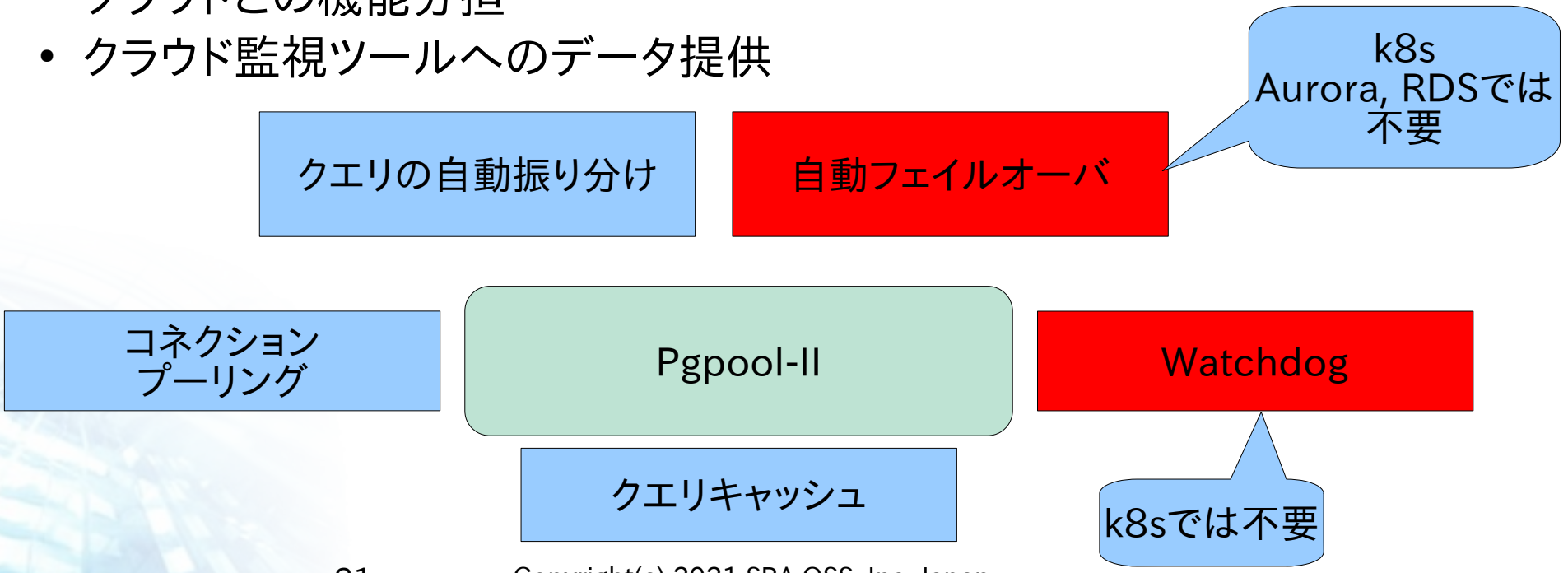
クエリキャッシュ

- 検索結果を共有メモリ(またはmemcached)に保存し、同じクエリ文字列が送られてきたら2回目以降はキャッシュから結果を返す
  - SQLパーサを経由せず、DBアクセスも発生しないので超高速
- キャッシュはテーブルに変更があったらクリアされる
- 更新の多いシステムには向かない(キャッシュヒット率70%以上での利用を推奨)

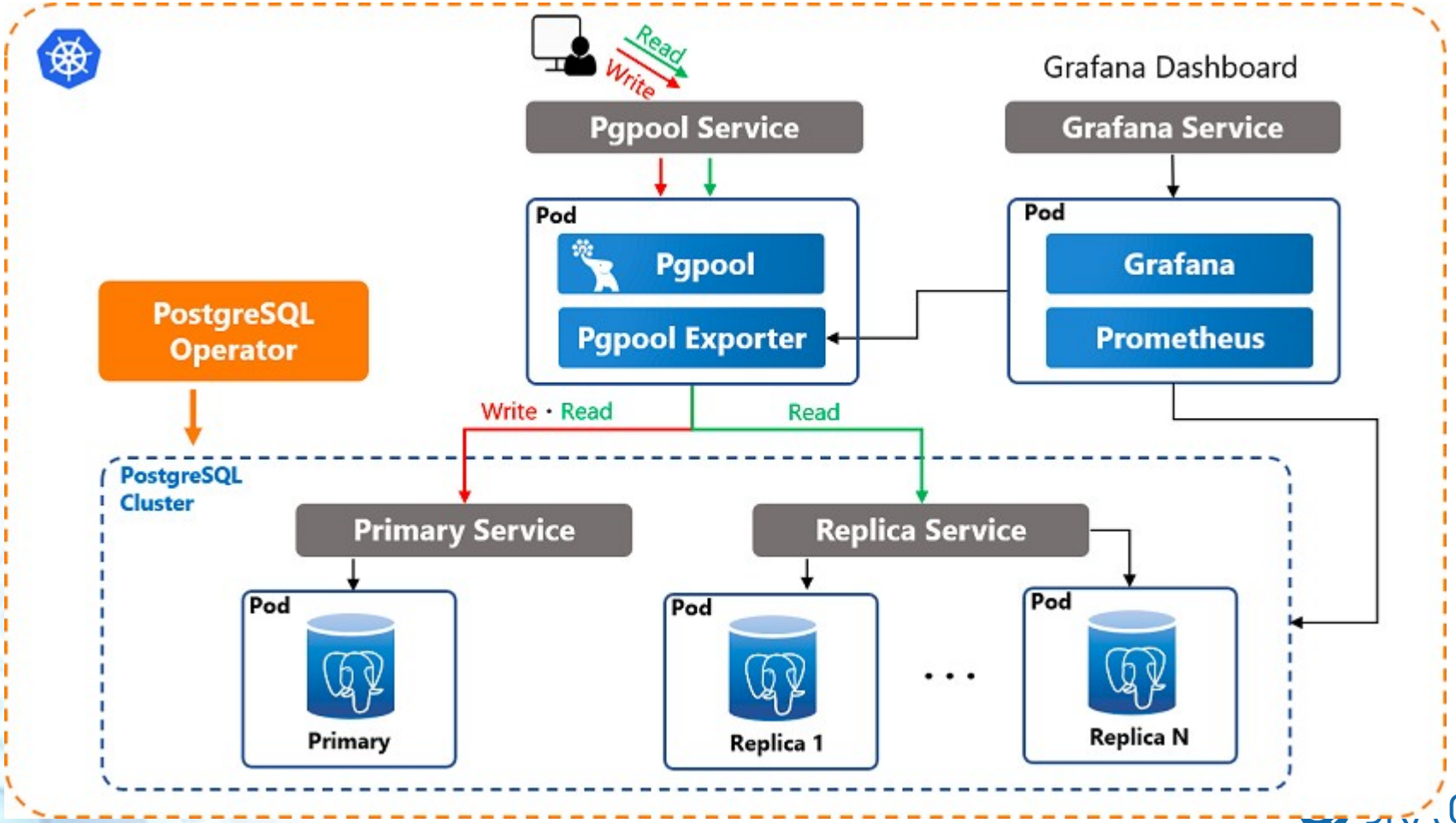


クラウド対応

- Kubernetes、AWS Aurora、RDSなどへの対応の強化
  - クラウドとの機能分担
  - クラウド監視ツールへのデータ提供



KubernetesでのPgpool-II利用例



クラウド用Pgpool-IIモジュール

- Pgpool-II on Kubernetes
  - https://github.com/pgpool/pgpool2_on_k8s
- Pgpool-II exporter
  - https://github.com/pgpool/pgpool2_exporter

Pgpool-IIの今後

リリースポリシー

- メジャーリリースを年に1回
  - 新しい機能の追加、機能変更
- マイナーリリースを少なくとも年に4回
  - 必要に応じて追加マイナーリリースを行うこともあり
- 5メジャーバージョンの保守を維持
  - 今なら4.3, 4.2, 4.1, 4.0, 3.7が保守対象
- 対応するPostgreSQL
  - 7.4以降であれば使えるはず（一部の機能はサポートされません）

次期メジャーバージョンで計画中の機能

- 超大量同時接続への対応
  - 動的なプロセス起動
- 新しい認証方式への対応
  - GSSAPI対応
- より使いやすく
  - レプリケーション遅延の単位をバイトから時間へ

将来の計画

より使いやすく

- 管理ツール(PCPコマンド)の見直し
- より豊富な統計情報の提供

テストの充実

- クラスタ管理は複雑な仕組み
- 不具合が発生したときの影響が大きい
- テストを支援する仕組みを作り込んでいく
  - 障害を人工的に発生させる仕組みなど

皆さんの声をおよせください

- Pgpool-IIはたくさんのユーザの皆様からの声で成長してきました
- 新しいアイデアや、「こんなことで困っている」など、のご意見をお待ちしています
- https://www.pgpool.net