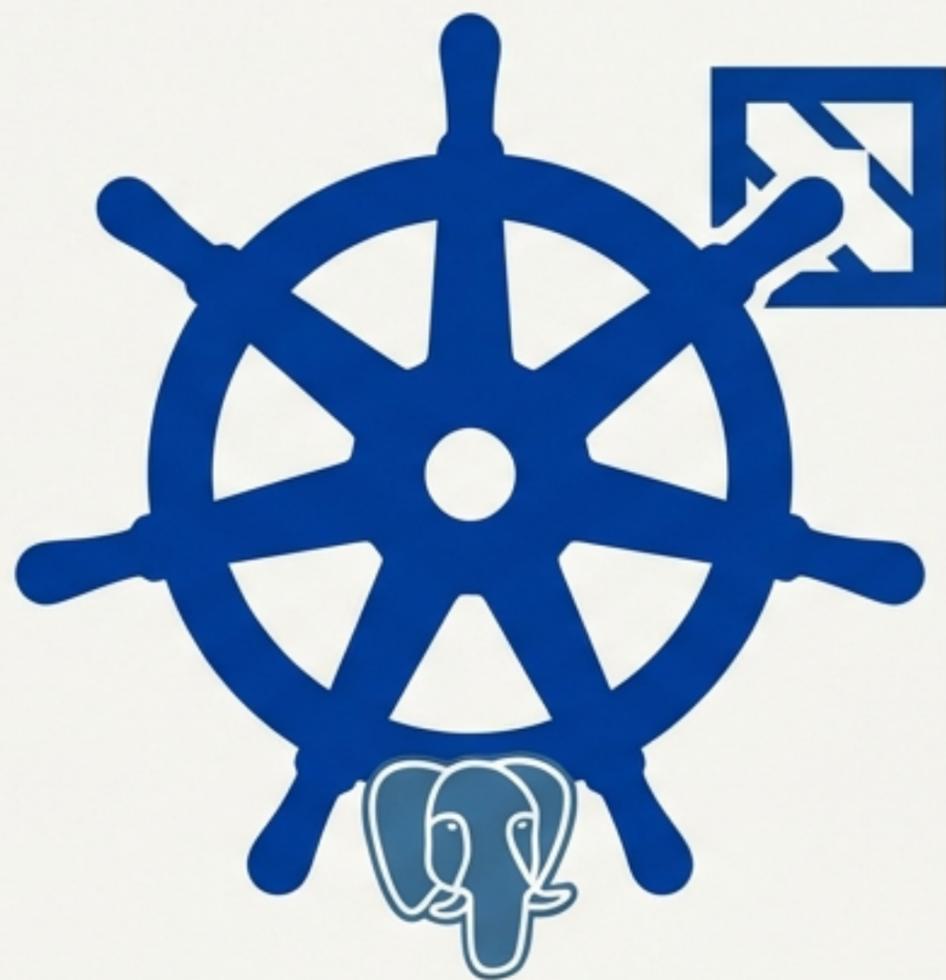


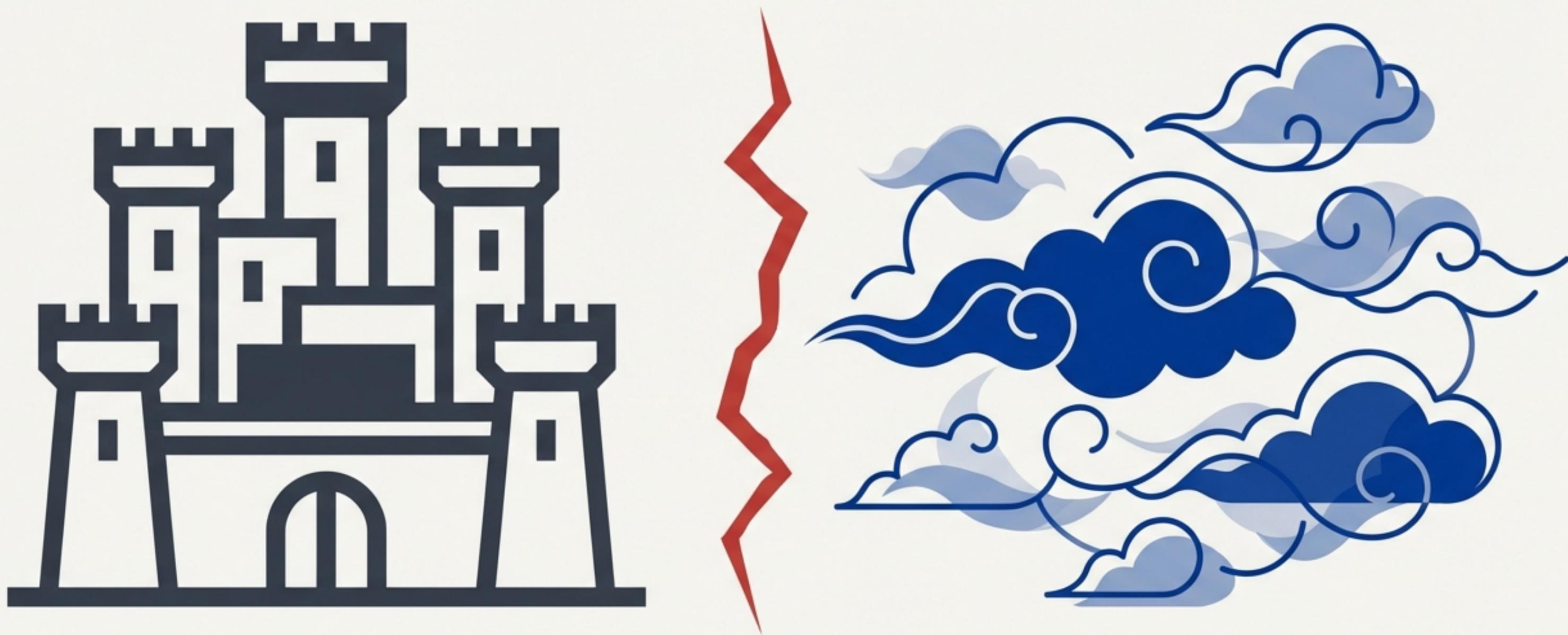
CloudNativePG: Kubernetesネイティブな PostgreSQLを、本来あるべき姿へ

Kubernetes上でのPostgreSQL運用に関する長年の課題を解決する、決定的なソリューション。



なぜPostgreSQL on Kubernetesはこれほど難しいのか？

ステートフルなデータベースと、本来ステートレスであるKubernetes。
この二つの設計思想の衝突が、運用の複雑さを生み出している。



痛みの根源: StatefulSetの限界

汎用的なStatefulSetは、データベース特有の複雑なライフサイクル管理には柔軟性が足りず、運用上の苦痛の源泉となっている。

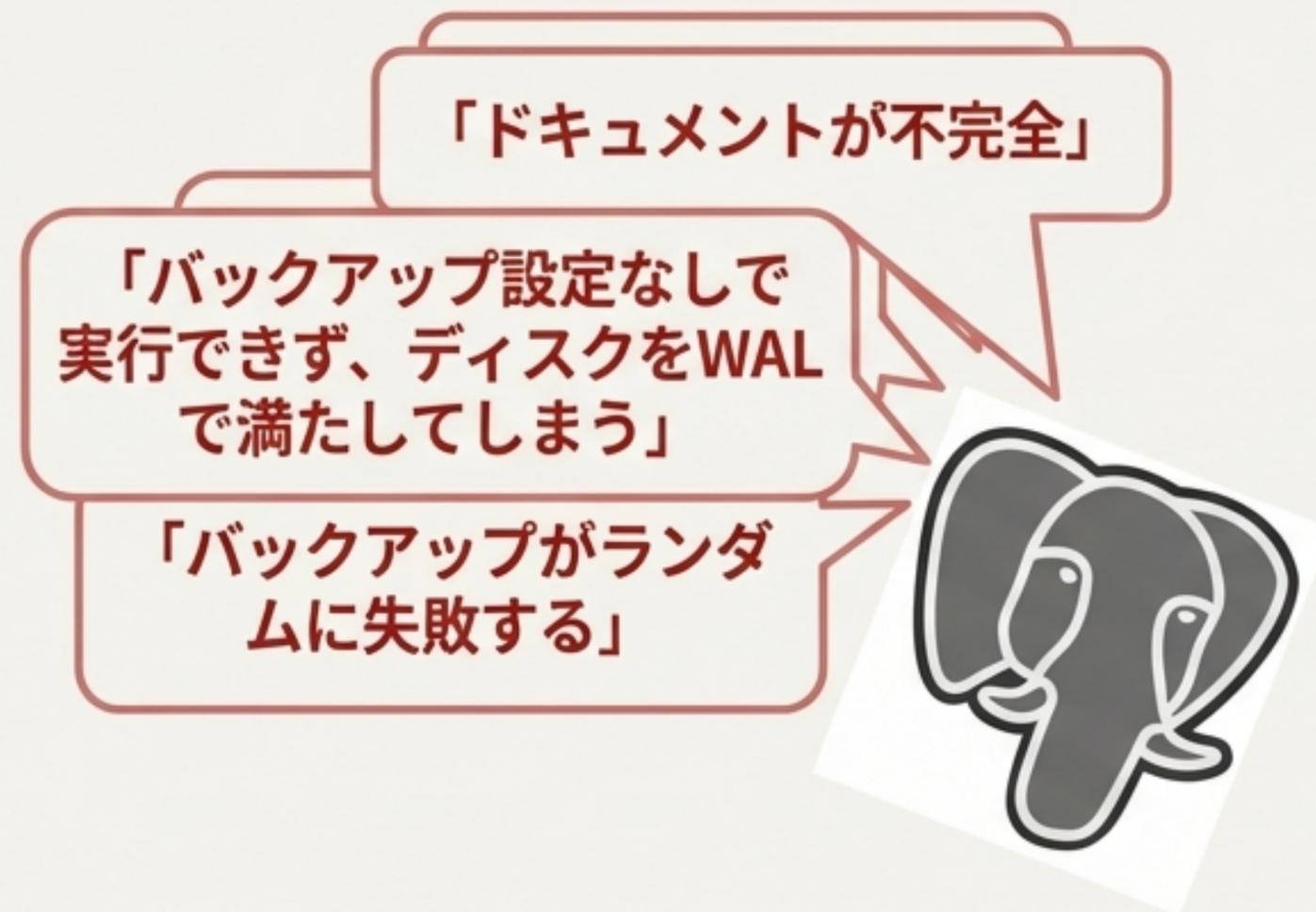
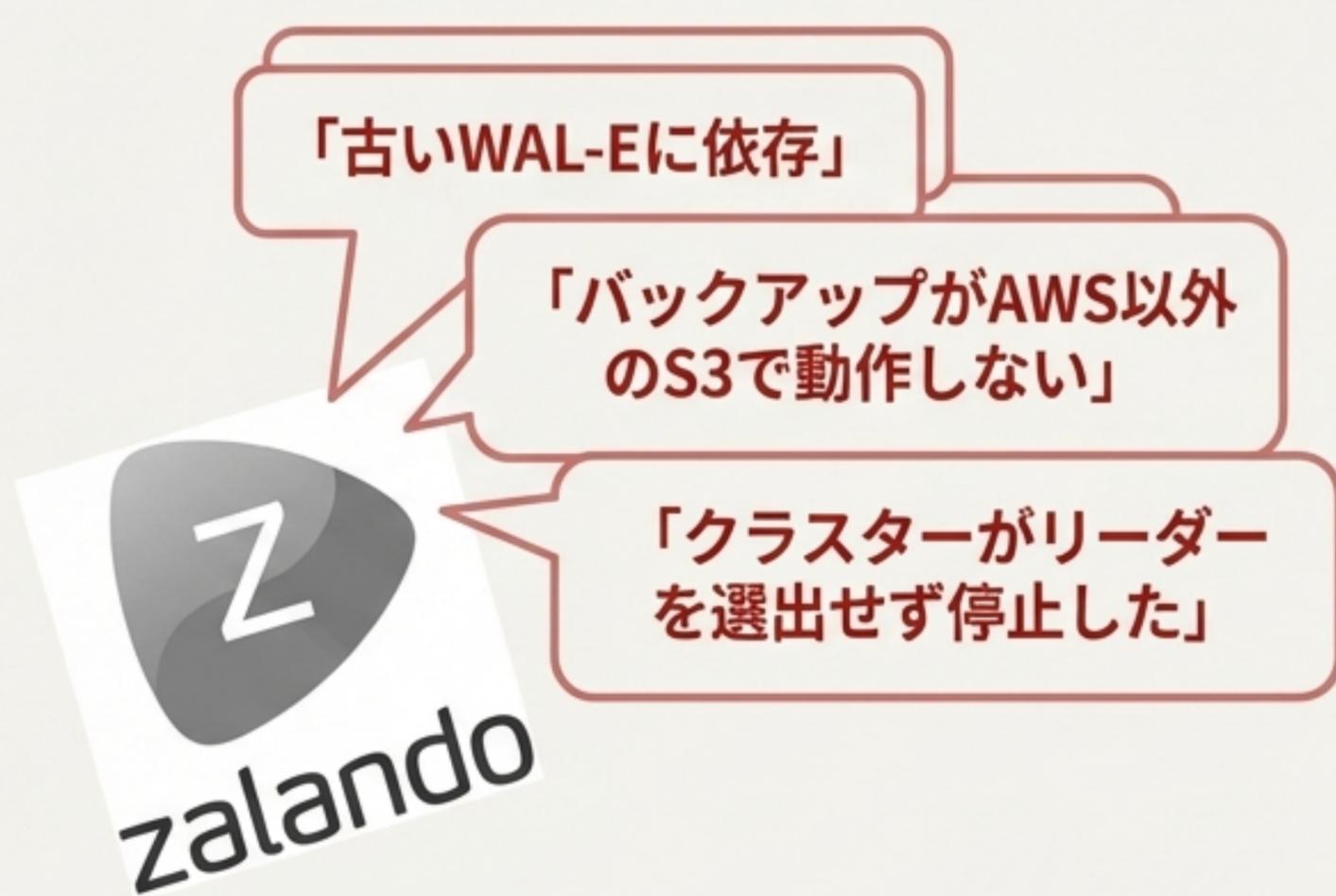


Hacker Newsの声:
*"StatefulSets are an
endless source of pain."*

- * StatefulSetはPodの永続性と順序性を保証するが、DBのフェイルオーバー、アップグレード、バックアップといった高度な操作には最適化されていない。
- * 結果として、複雑なシェルスクリプトや手動介入が必要になり、"The Kubernetes Way"から逸脱してしまう。

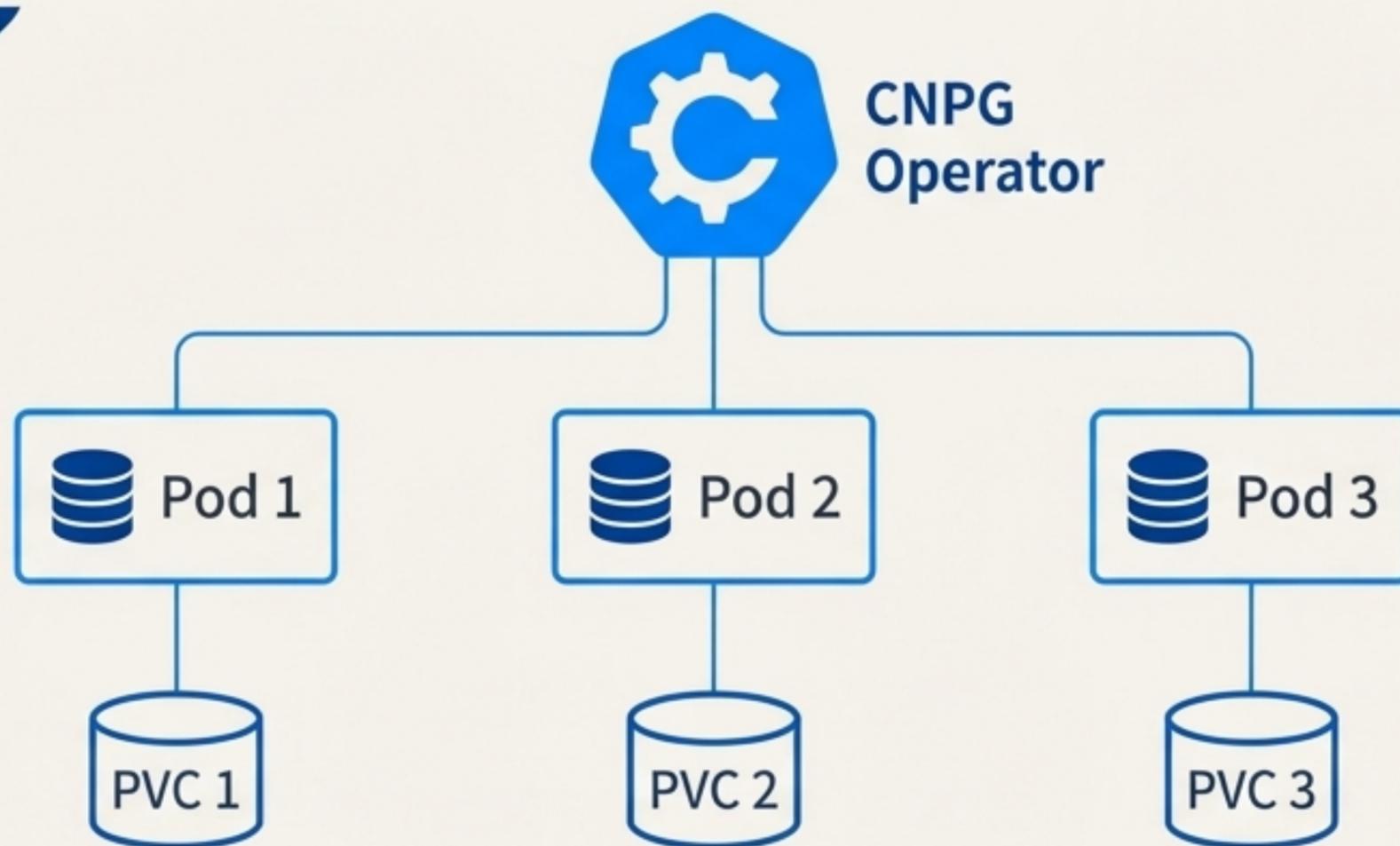
混沌とするPostgreSQL Operatorの世界

多くの先行Operatorは、古い依存関係や独自規約といった課題を抱え、真のクラウドネイティブ体験を妨げてきた。



新しいパラダイム: CloudNativePG

CloudNativePGは、StatefulSetを使わない。代わりにPostgreSQLに最適化された専用コントローラーで、真の「Kubernetesネイティブ」な体験を実現する。



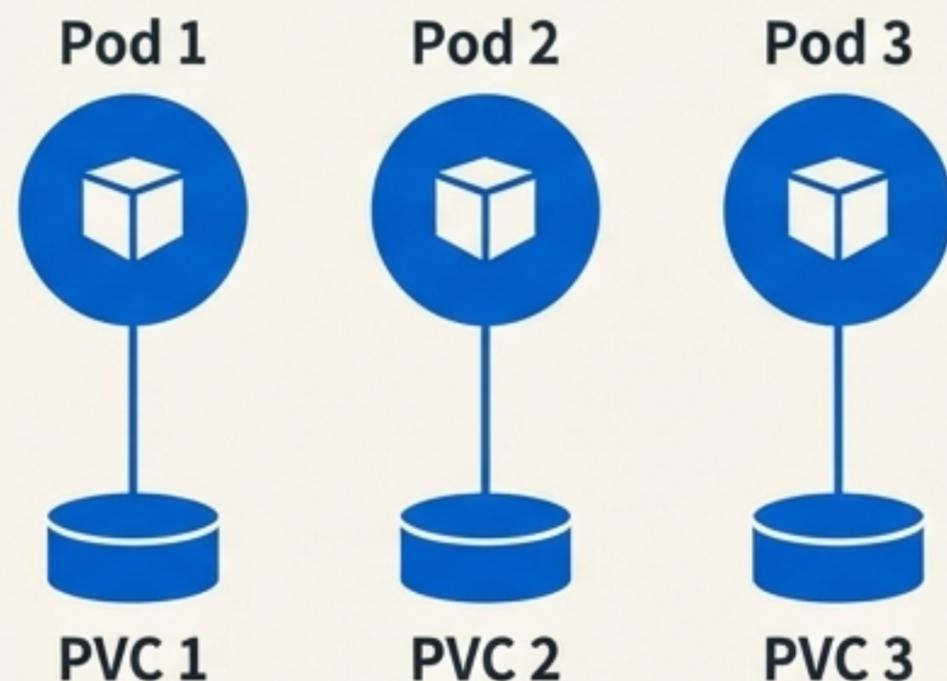
- CNPGは、Kubernetes APIを直接利用してクラスタの状態を管理するカスタムPodコントローラーを実装。
- DBAのベストプラクティスをコード化（Operator化）することで、複雑な運用を自動化。
- PostgreSQLのライフサイクル全体を、一つのYAMLファイルで宣言的に管理可能に。

特徴①: "あるべき姿"を定義するだけの宣言的設定管理

`cluster.yaml`という一つのファイルで、インスタンス数、ストレージ、PostgreSQLパラメータまで、クラスタの全てをコードとして管理できる。

```
apiVersion: postgresql.cnpg.io/v1
kind: Cluster
metadata:
  name: cluster-example
spec:
  instances: 3

  storage:
    size: 10Gi
```



- `kubectl apply -f cluster.yaml`だけで、高可用性クラスタを構築。
- パラメータ変更も、YAMLを修正して`apply`するだけ。CNPGがローリングアップデートを自動実行。
- GitOpsワークフローに完璧にフィット。

特徴②: 眠れる夜を取り戻す、高可用性と自動フェイルオーバー

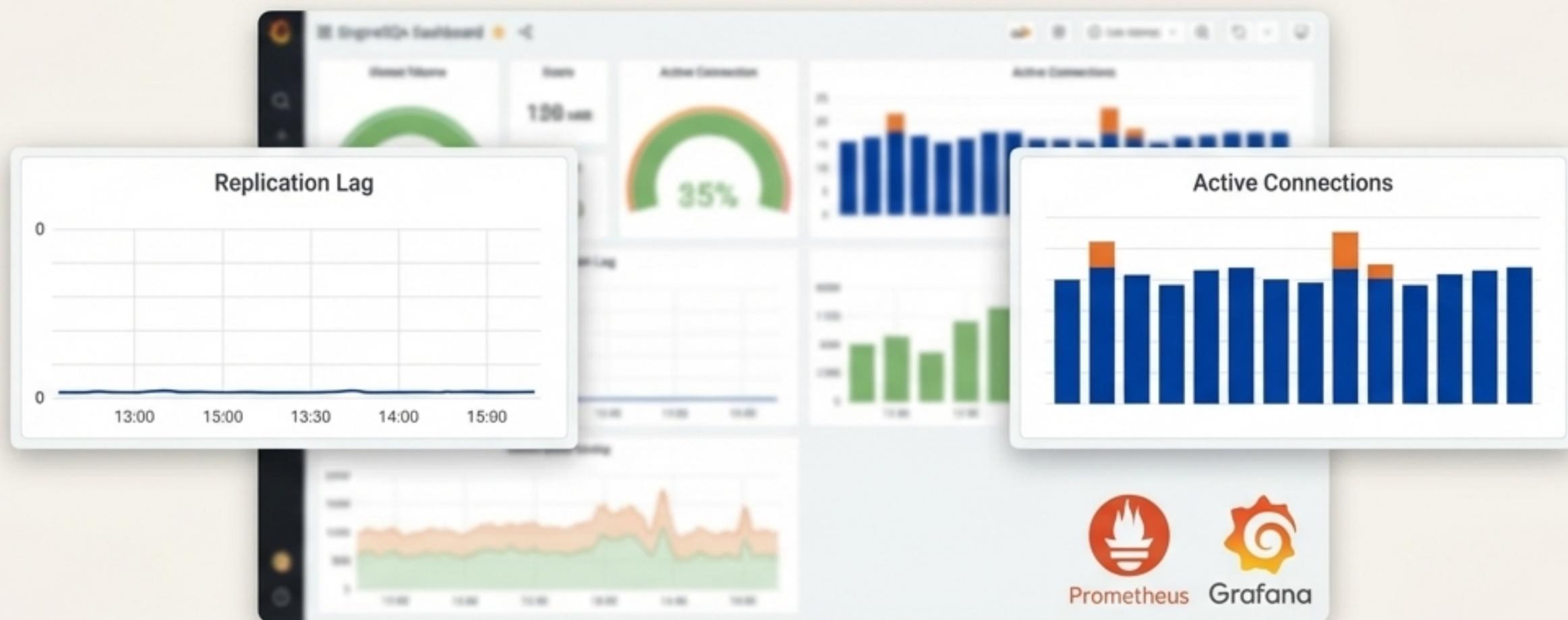
ノード障害やプライマリインスタンスの異常をCNPGが自動検知し、数秒で最適なレプリカを新プライマリへ昇格させる。



- CNPGは各インスタンスの状態を常時監視。
- プライマリ障害時、最も同期が進んでいるレプリカを自動で選定し、フェイルオーバーを実行。
- サービスエンドポイントも自動で更新され、アプリケーションからの接続はシームレスに維持される。

特徴③: 問題を予見し、先手を打つ統合された監視

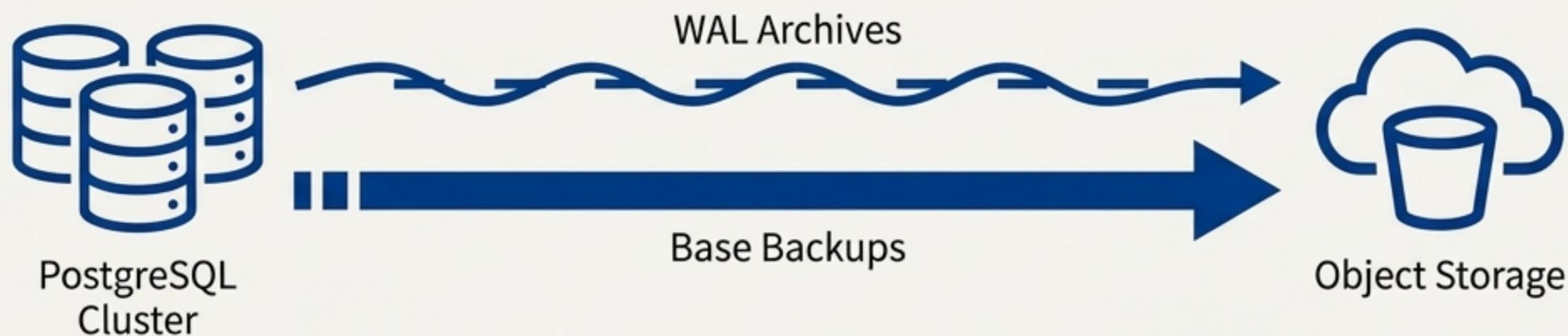
追加設定不要でPrometheusエクスポーターが標準搭載。PostgreSQLの内部状態を詳細に可視化し、プロアクティブな運用を可能にする。



- `PodMonitor` リソースを自動生成し、Prometheusが即座にメトリクスを収集開始。
- トランザクション数、レプリケーションラグ、コネクション数など、DBAが必要とするメトリクスを網羅。
- 公式のGrafanaダッシュボードも提供。インポートするだけで高度な可視化を実現。

特徴④: "その時"に備える、柔軟なバックアップ&リカバリー

オブジェクトストレージへの継続的なWALアーカイビングと定期的なベースバックアップを宣言的に設定。PITR (ポイントインタイムリカバリ) も容易に実現。



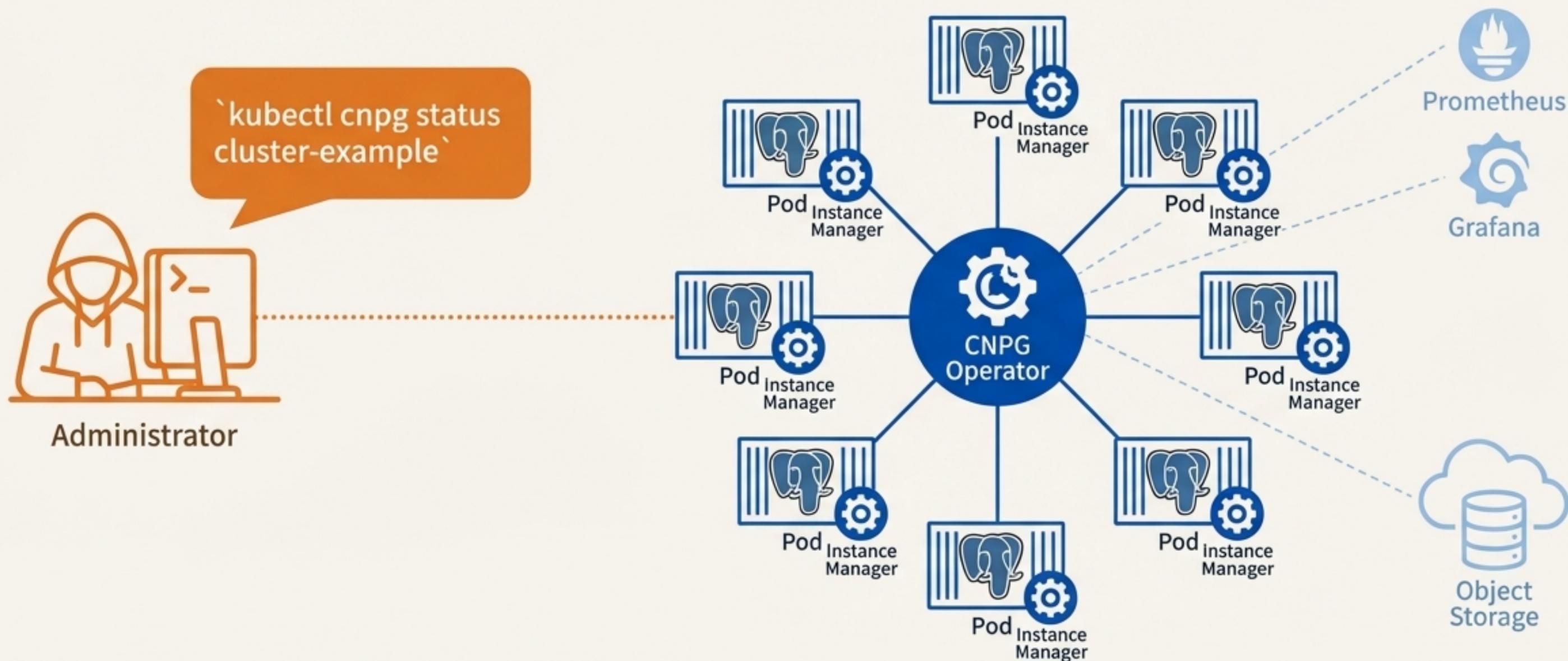
Hacker Newsの声:

"...recovery just works (unlike zalando...)"

- S3互換のオブジェクトストレージに対応。
- バックアップスケジュール、リテンションポリシーもYAMLで管理。
- ボリュームスナップショットを利用した高速なリカバリーもサポート。

全体像: CNPGアーキテクチャとエコシステム

オペレーター、インスタスマネージャー、そして豊富な`kubectl`プラグインが連携し、PostgreSQLの運用をエンドツーエンドでサポートする。



まとめ: CloudNativePGがもたらす価値

CNPGは、Kubernetes上でのPostgreSQL運用を「苦行」から「標準的なプラクティス」へと変革する。



**Statelessな思想で
Statefulを管理:**
StatefulSetを回避し、
DBに最適化された制御
を実現。



運用の劇的な簡素化:
宣言的設定、自動HA、
統合監視、柔軟なバック
アップを全てYAMLで。



真のCloud Native:
DevOps、GitOpsのワークフローに
完璧にフィット。

