

上下水道監視システムにおける データベース事例

小松電機産業(株) 廣江 深

目次

- 自己紹介
- 企業紹介・上下水道概要・サービス紹介
- PostgreSQL冗長化の失敗談と現状
- 分散キーバリューストアRiakの紹介

自己紹介

名前：廣江 深

勤務地：島根県松江市

勤務先：小松電機産業株式会社

github : hiroeorz



企業紹介

社名 : 小松電機産業株式会社

本社 : 島根県松江市 (東京 大阪 ソウル支社)

社員数: 90人くらい(うちソフトウェア関係10人くらい)

創立 : 1973年



水インフラ監視・制御システム
「やくも水神ネットワーク」



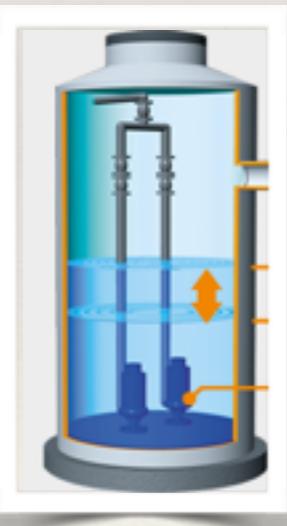
上下水道概要



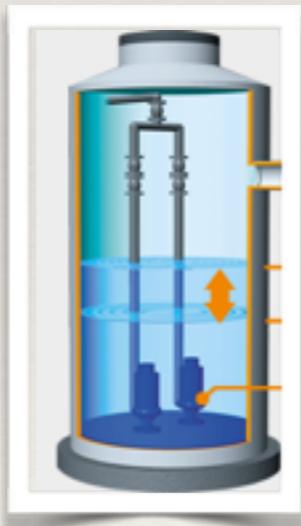
浄水場



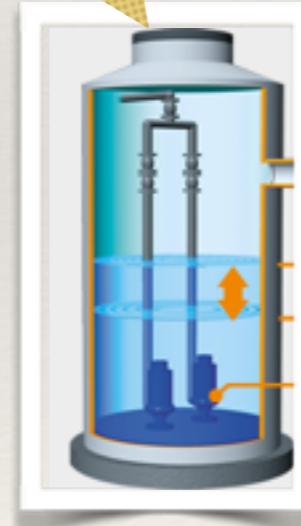
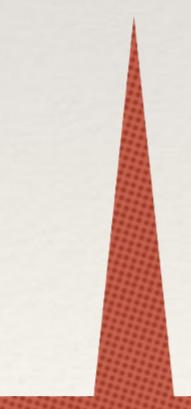
下水道



マンホールポンプ場



次々バケツリレーして



下水処理場

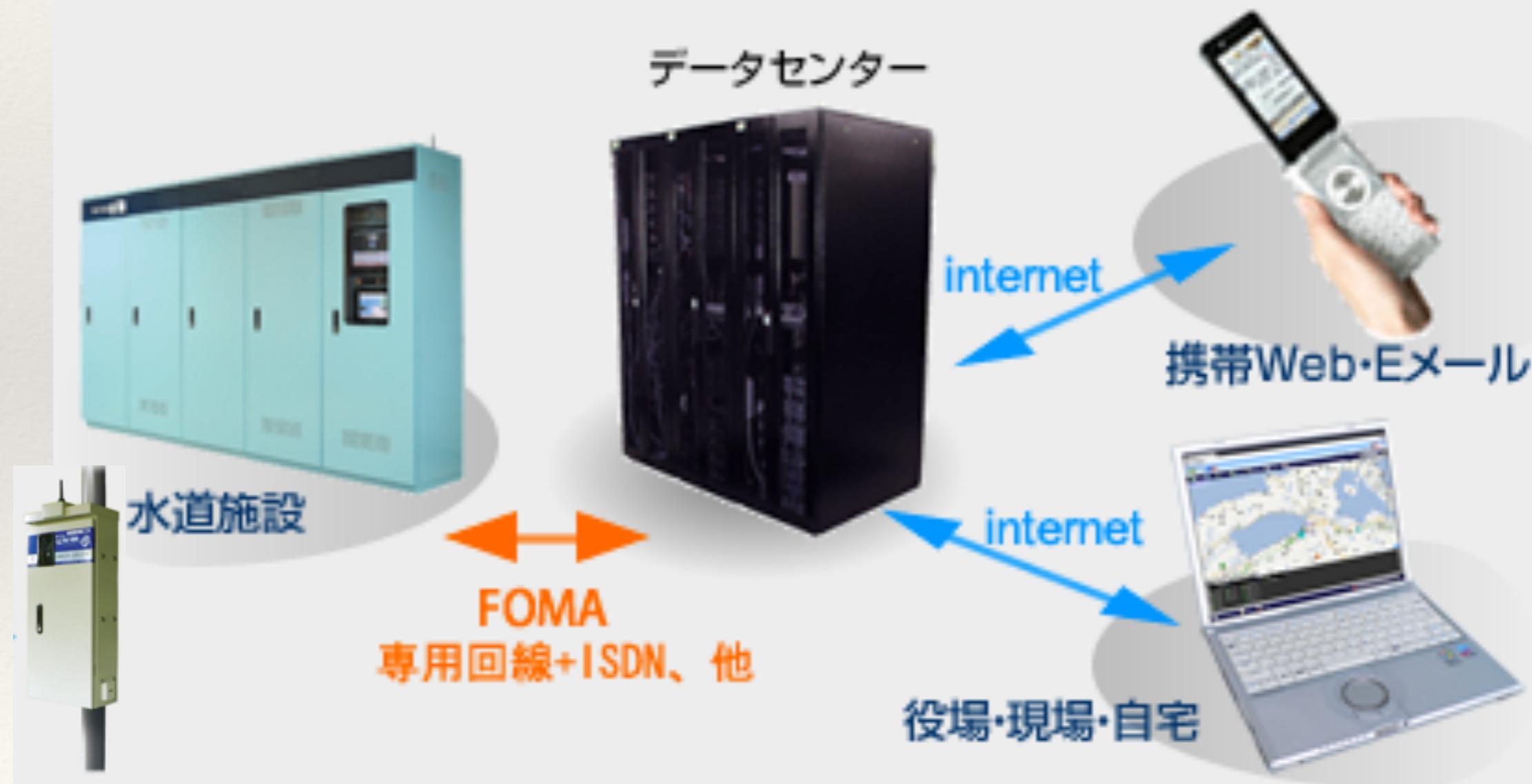


綺麗な水にして川に流します

サービス紹介

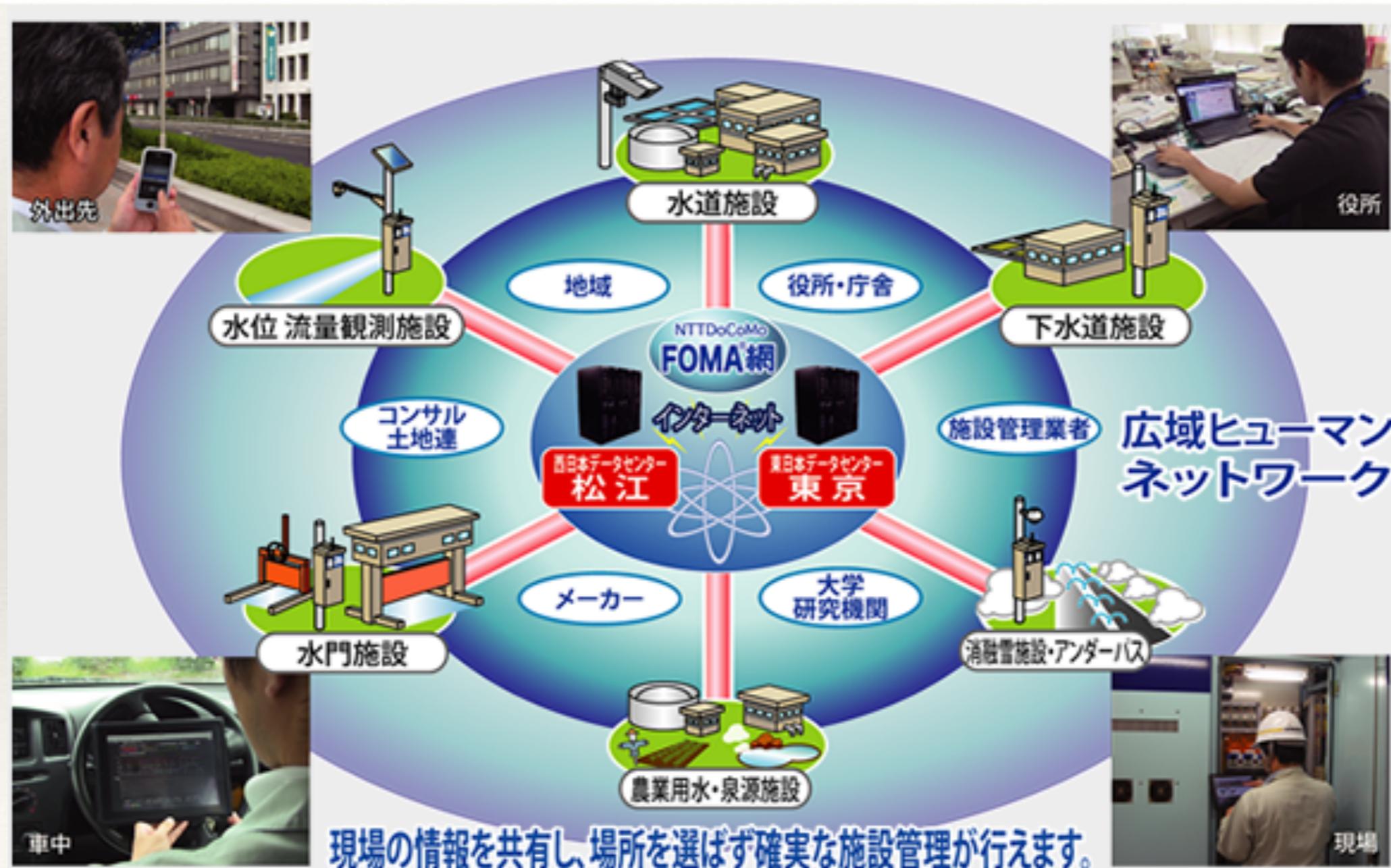
各施設をデータセンターのサーバーを介して、スマートフォン・パソコン・携帯電話から状態の確認や制御を行えます。

また、現場で異常が発生するとメールやプッシュ通知で担当者に知らせます。



サービス紹介

上水道・下水道・消融雪装置・水門装置・農業用水等
水関連のインフラでご利用いただいているネットワークサービスです。



監視インターフェイスの例

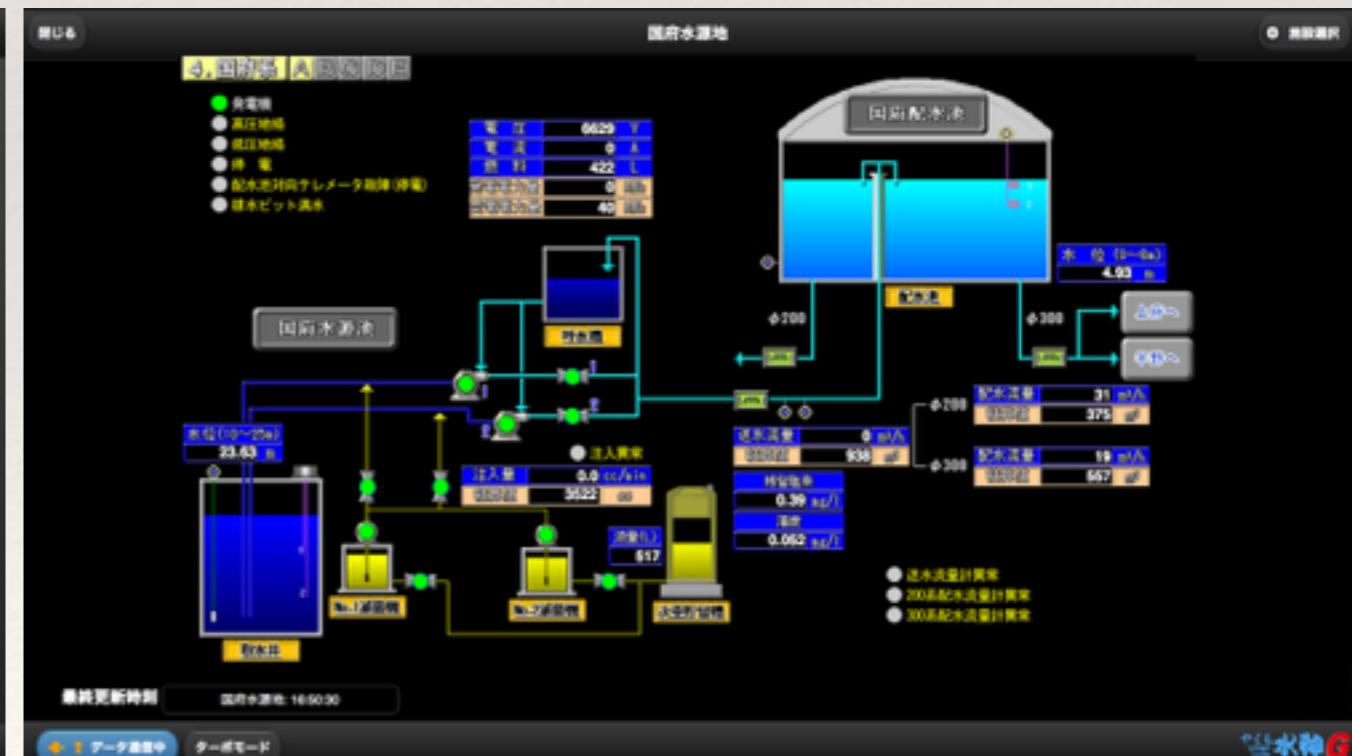
The figure displays two screenshots of a water supply system monitoring software. The left screenshot shows a grid of eight data panels with various parameters and their current values. The right screenshot shows a map of the water network with specific data points highlighted.

Left Screenshot (Monitoring Values):

- コントローラ内部温度: 国府水源地 (27.3 °C)
- 送水流量: 国府水源地 (300 m³/h)
- 取水井水位: 国府水源地 (24.99 m)
- 次亜鉛蓄槽液位: 国府水源地 (518 L)
- 次亜注入量: 国府水源地 (0.0 cc/min)
- 残留塩素: 国府水源地 (0.39 mg/l)
- 配水池水位: 国府水源地 (4.92 m)
- 燃料レベル: 国府水源地 (422 L)

Right Screenshot (Map View):

- 送水流量: 国府水源地 (0 m³/h)
- 300系配水流量: 国府水源地 (24 m³/h)
- 200系配水流量: 国府水源地 (30 m³/h)
- 小国系 送水流量: 波名淨水場 (0.1 m³/h)
- 第2取水井取水流量: 波名淨水場 (18.1 m³/h)
- 二子山配水池 水位: 波名淨水場 (2.35 m)
- 上來原第2配水池 z: 波名淨水場 (6.04 m)
- 上來原系 配水流量: 波名淨水場 (38.4 m³/h)
- 二子山配水池 水位: 波名淨水場 (2.35 m)
- 小国系 送水流量: 波名淨水場 (6.04 m³/h)



採用実績

2000年発売以来

314自治体 7050施設

でご採用いただいております。 2013年12月現在

上下水道をはじめ様々な施設に採用!!

北海道・青森県・秋田県・岩手県・宮城県・山形県・福島県・新潟県・長野県・群馬県・茨城県・埼玉県・栃木県・千葉県・東京都・神奈川県・山梨県・静岡県・富山県・石川県・福井県・岐阜県・愛知県・三重県・滋賀県・京都府・大阪府・奈良県・和歌山県・兵庫県・岡山県・広島県・鳥取県・島根県・山口県・愛媛県・高知県・徳島県・福岡県・佐賀県・長崎県・熊本県・大分県・宮崎県・鹿児島県・沖縄県

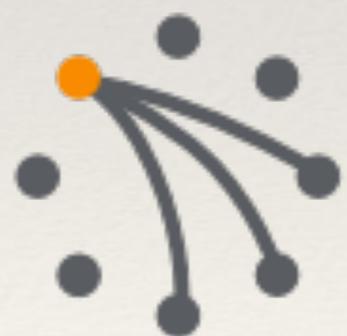


使用しているデータベース

RDB



KVS

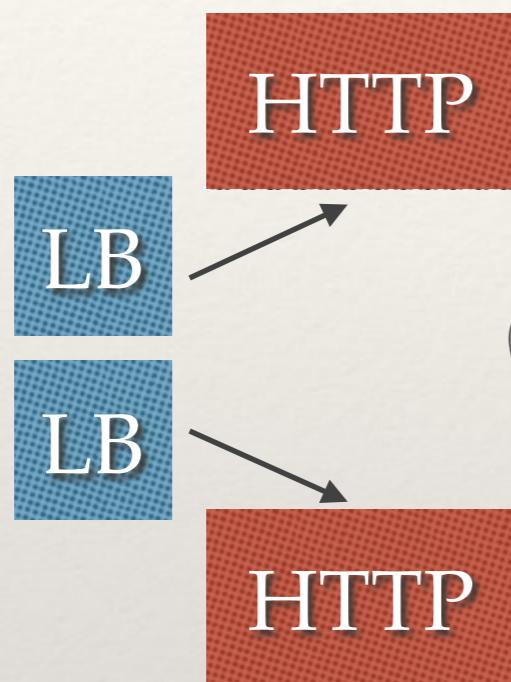


riak



システム構成

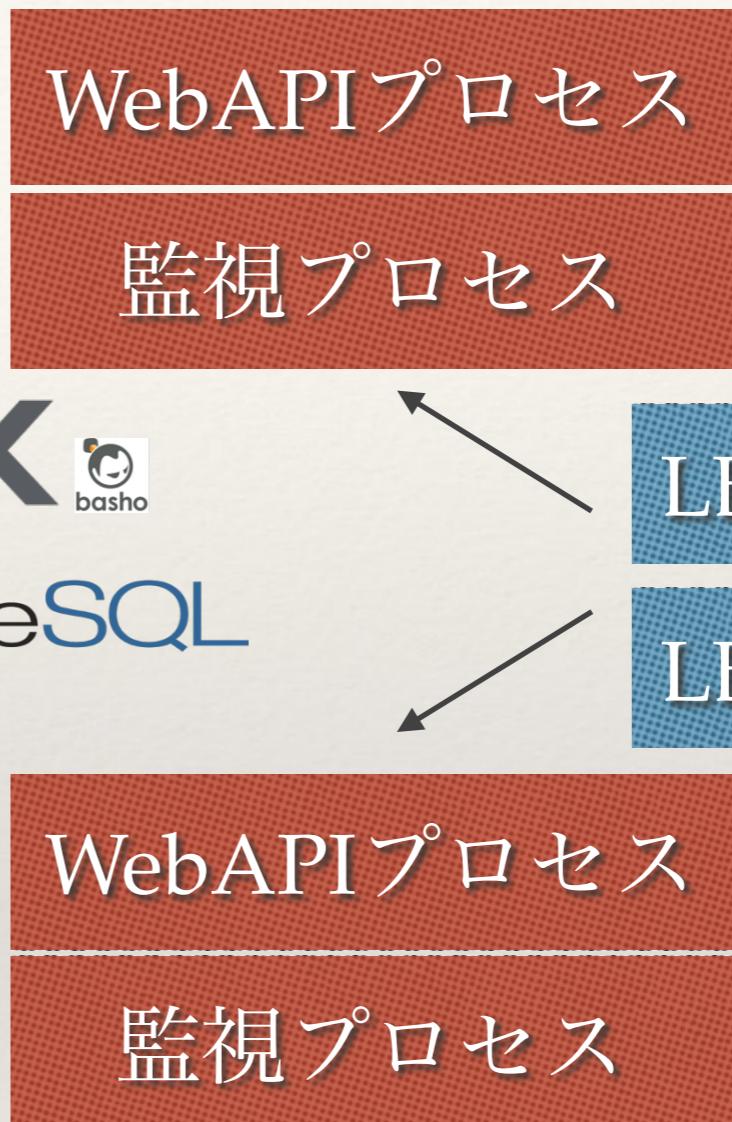
インターネット



PITR+rsyncで差分バックアップ

マスター交代の際には最大5分間のデータ欠損

スレーブ
PostgreSQL

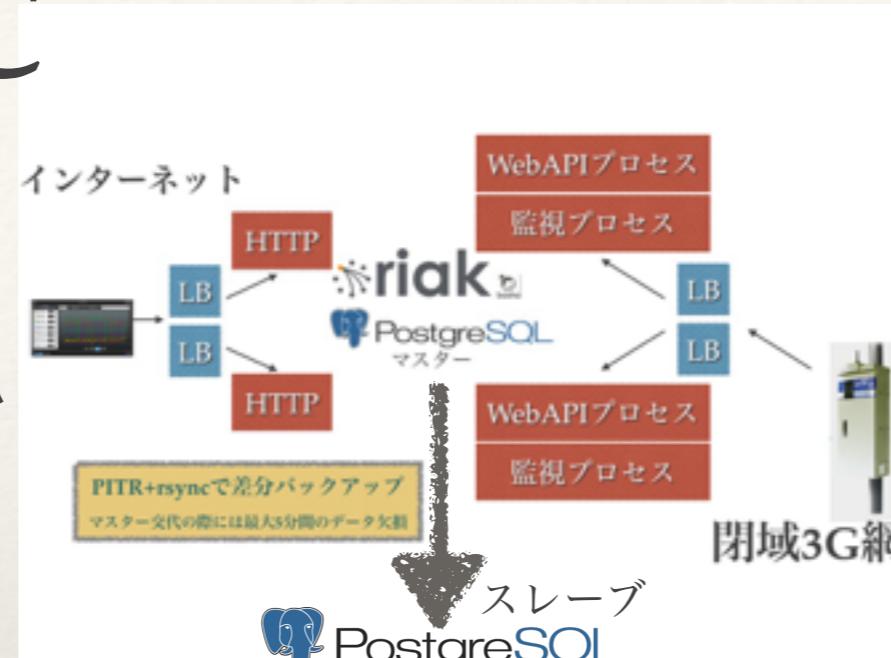


閉域3G網

システム構成

東京DC

正常時



IP-VPN

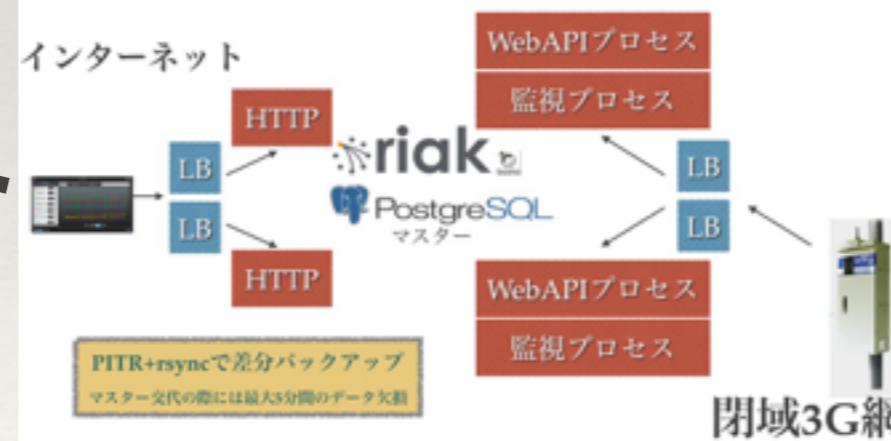
異常時



島根DC

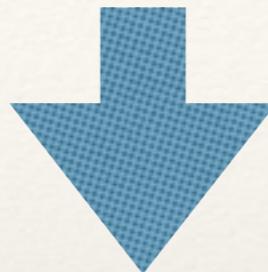
正常時

異常時



RDBとKVS使い分けの考え方

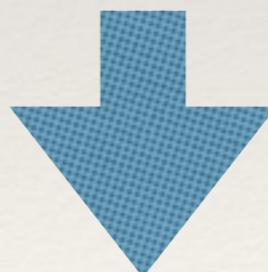
各種設定データ・集計結果データ



リレーショナルデータベース



未集計のログデータ・画像・システムログ



分散キーバリューストア





- バージョン：8.3
- データサイズ：400GBくらい
- 総データベース数：380個くらい
- 総テーブル数：38,000個くらい



保存しているデータ種別

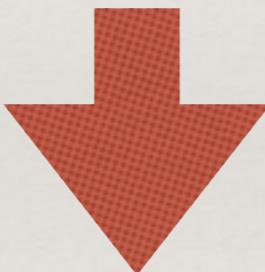
- ・ユーザー設定
- ・端末設定（通信デバイスのIPアドレス、ID、ソフトバージョンなど）
- ・表示設定（どの項目をグラフに表示するかなど）
- ・毎日の集計データ（機器運転時間、センサー計測値、積算値等）
- ・月毎の集計データ
- ・機器故障等の履歴

主に設定データ、集計済みデータの保存に使用

RDBの冗長化について

一応、上下水道などの生活インフラを監視しているので、365日24時間、サービスは基本的には落とせない。

そういうわけでシステム全体にわたってマルチマスタな冗長構成をとっているけどRDBのところだけは、どうしても单一障害点となりやすい。



安定運用の為にはRDBの冗長化が大事

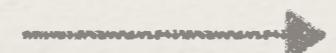


PostgreSQL

RDBの冗長化について

やったことその1（2003年～2005年くらい）

Usogresによる入口での冗長化

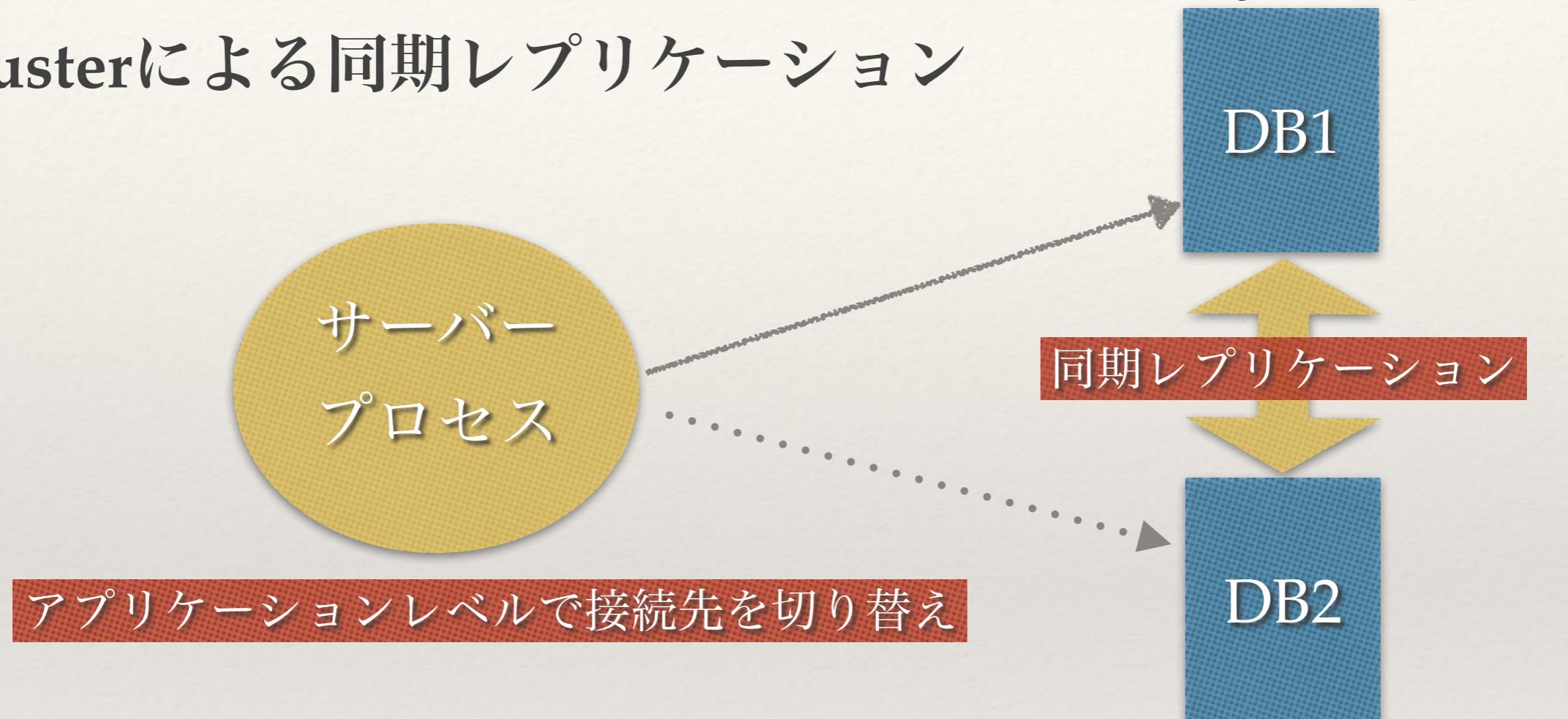


- 同期マスタースレーブ方式
- 最初の数年はうまくいっていた
- どちらかのDBが停止した場合は、しばらく片方のDBで動かしておいて、タイミングを見てシステム全体を止めて再同期の必要あり
- 数年経過したあたりから再同期にかかる時間が無視できなくなってきたため、この方法は現在は使っていない。
- こちらのツールは2003年頃からメンテナンスが行われていない模様

RDBの冗長化について

やったことその2 (2005年～2006年くらい)

pgclusterによる同期レプリケーション

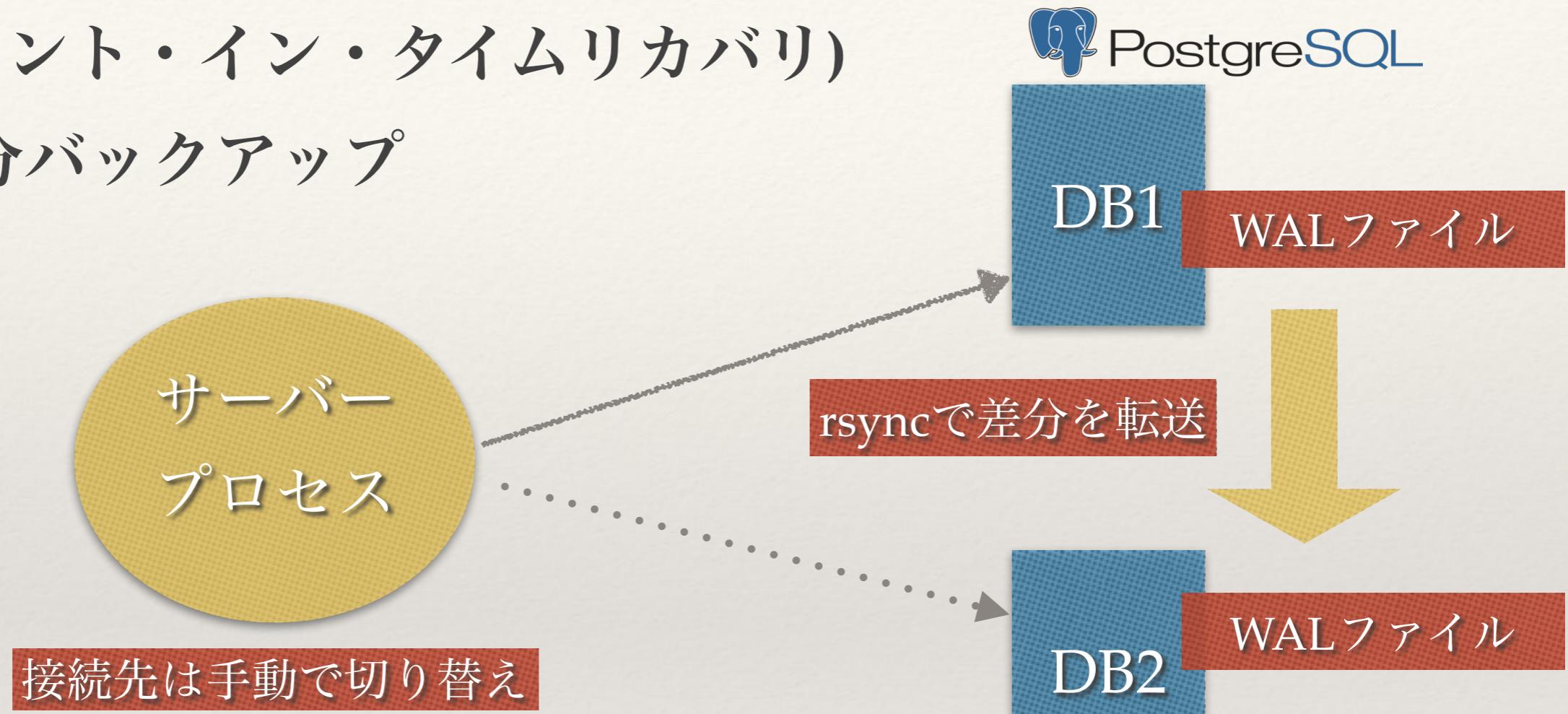


- 原因不明のデッドロックが時折発生する
- どうやら毎日午前1時ごろから実施している全端末の集計データ収集・保存のタイミングで発生しているらしいけど原因がつかめなかつたので、現在はこの方法はやってない。
- こちらのツールも2005年頃を最後にメンテナンスが行われていない模様

RDBの冗長化について

やったことその3 (2007年～現在)

PITR(ポイント・イン・タイムリカバリ)
による差分バックアップ



- 1日2回、DB1からDB2へベースバックアップを実行
- その時点からのWALを5分間隔でDB2に転送
- DB1障害時にはDB2をリカバリー モードで起動、アプリ接続先をDB2に切り替える
- 現在はこの方法で安定して運用できています

RDBの冗長化について

東京DC



接続先は手動で切り替え

島根DC

 PostgreSQL

DB1

DB2

 PostgreSQL

WALファイル

WALファイル

WALファイル

rsyncで差分を転送





- Basho社が開発しているオープンソースの分散キーバリューストア
- バージョン：1.4（2.0から検索が強力になってるけどまだバージョンアップしてない...）
- クラスタ数3で運用（本当は5以上が望ましいらしい）
- 運用のしやすさが特徴
- マルチマスター・单一障害点がない
- 実装言語はErlang/OTP
- 2つ前のバージョンまで後方互換性を確保



保存しているデータ種別

- ・機器運転の詳細ログ
- ・センサー計測値データの詳細ログ（24時間 1分間隔）
- ・各施設の写真データ（ユーザーがスマートフォン等からアップロードしたもの）
- ・画像データ
- ・端末が10秒間隔でサーバーに送信してくる機器状態・センサー計測値データ
- ・ユーザーの閲覧ログ（fluent経由で保存）
 - ・必要な時に自作CUIツールで集計



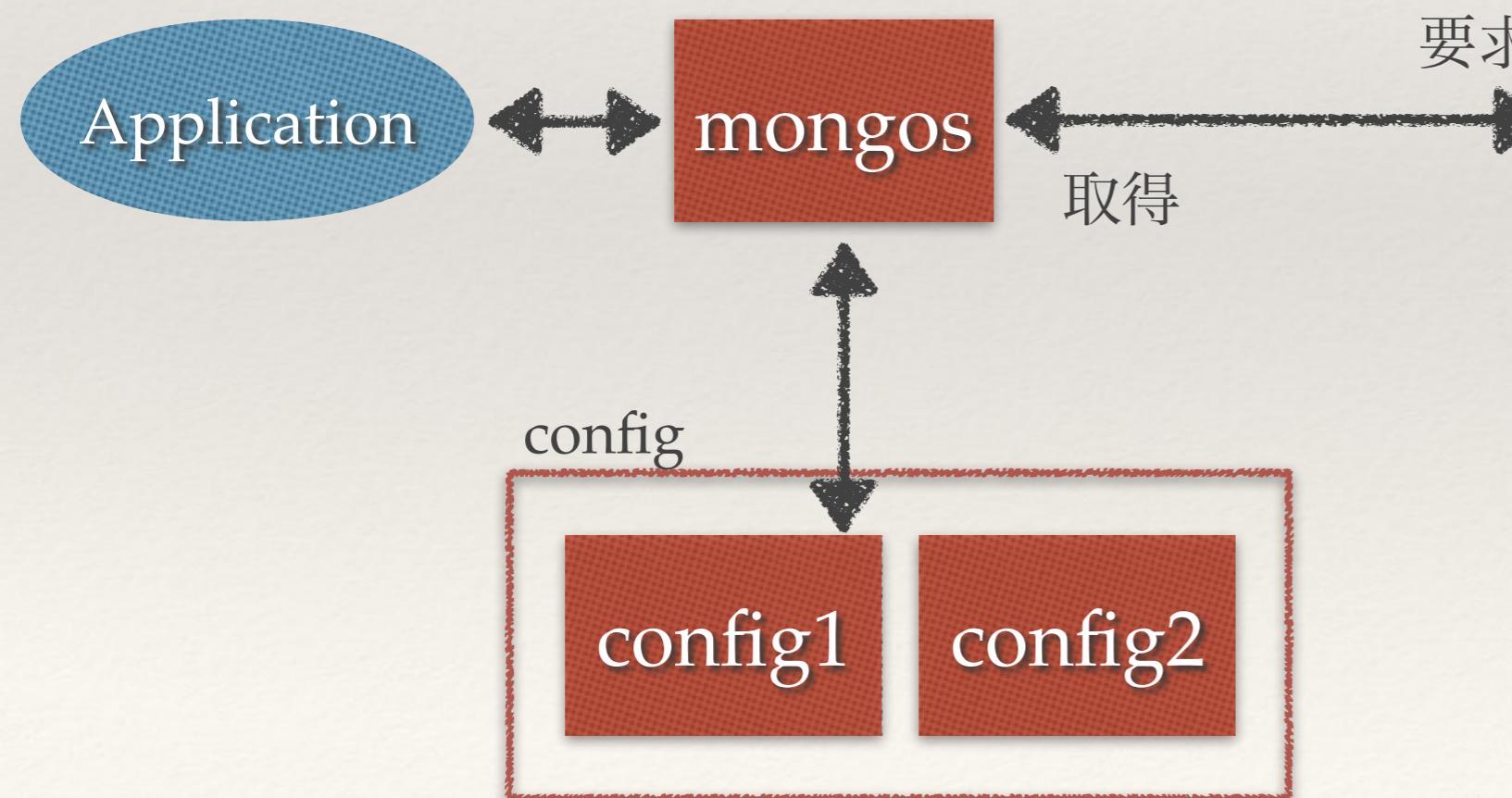
主に未集計のログデータ、画像データ等の保存に使用

分散キーバリューストアRiakの紹介

mongoDBとの構成比較



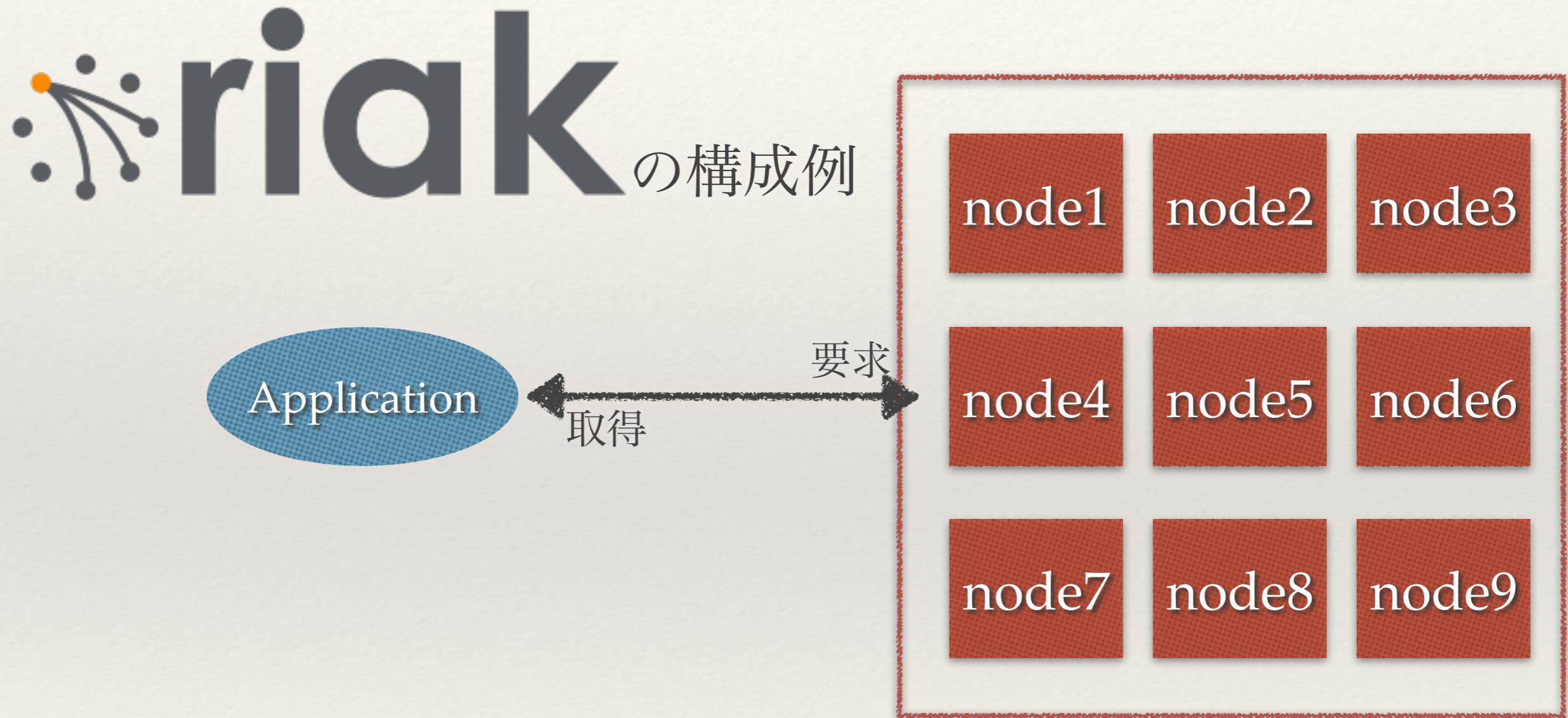
の構成例



赤：プライマリ
青：セカンダリ
黄：仲介サーバ

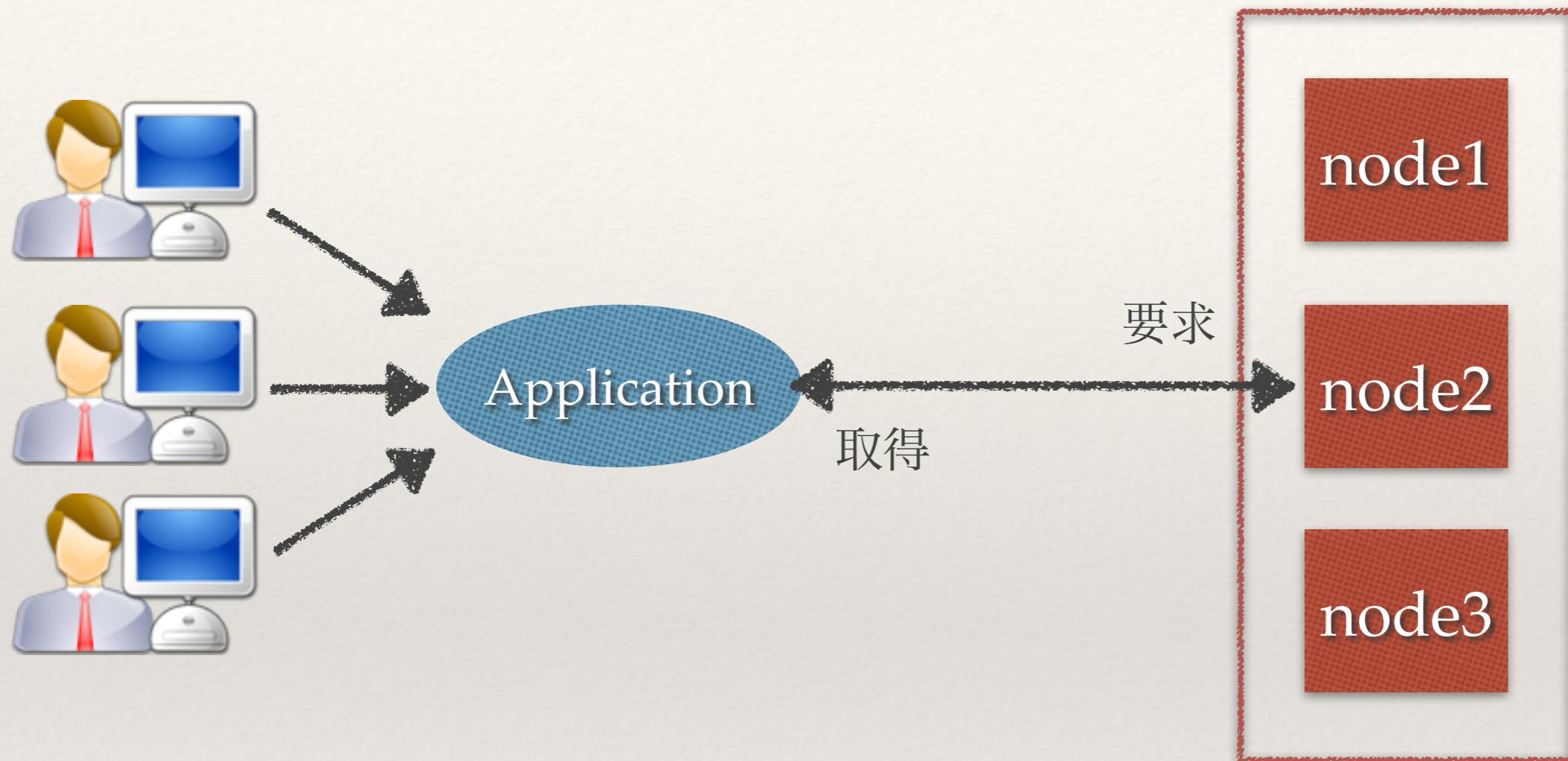
分散キーバリューストアRiakの紹介

mongoDBとの構成比較



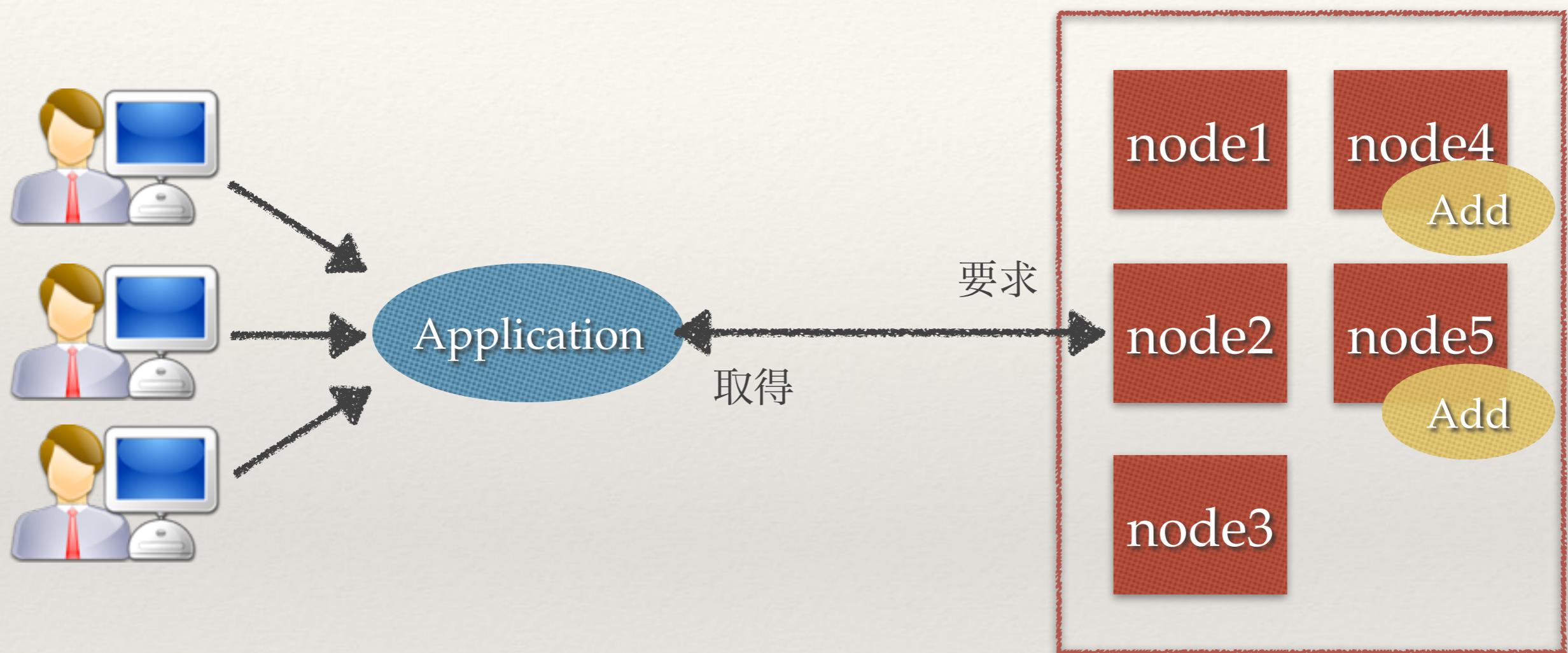
- ・特定機能を提供するノードがないので構成がシンプルになる
- ・マルチマスタ。プライマリ・セカンダリなどの概念がない

分散キーバリューストアRiakの紹介



- 稼働したままノードの追加・削除ができる

分散キーバリューストアRiakの紹介



- 稼働したままノードの追加・削除ができる

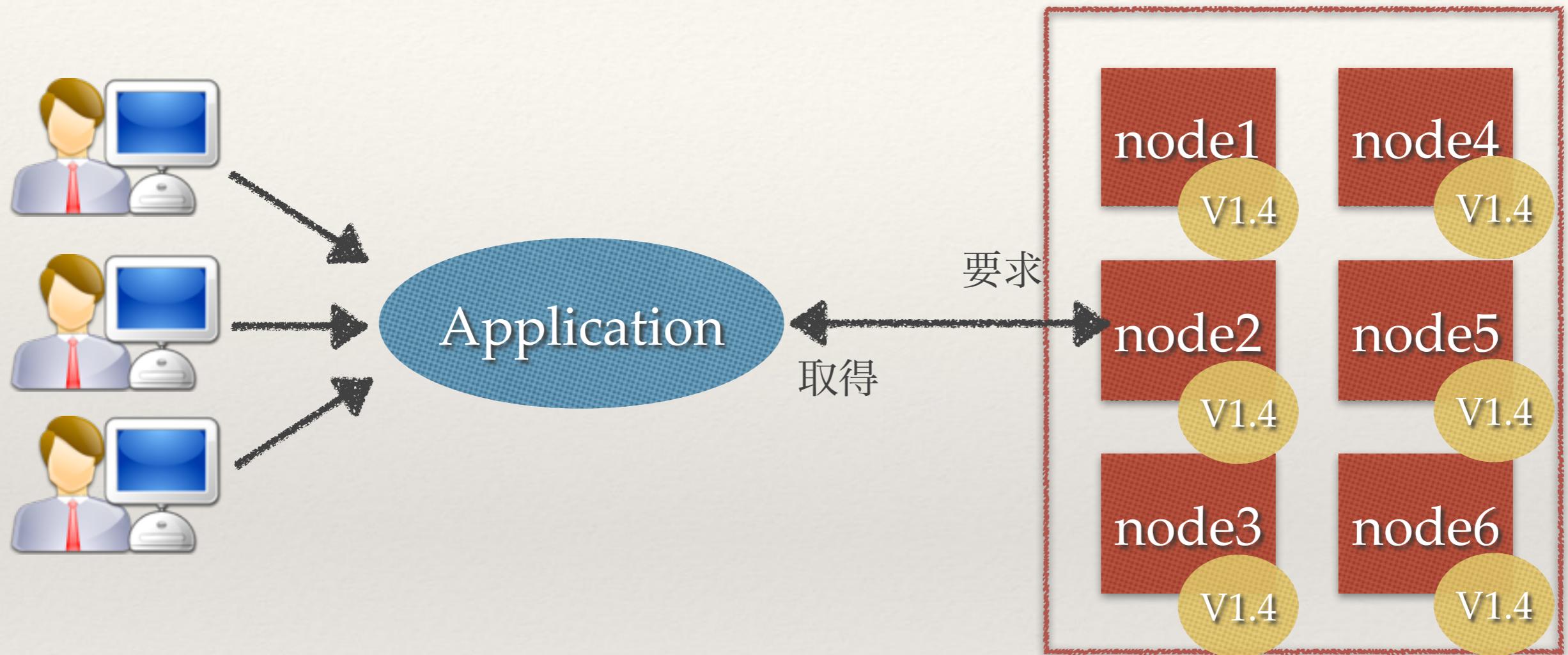
分散キーバリューストアRiakの紹介

The screenshot shows the Riak Control interface for Cluster Management. At the top, there's a toolbar with icons for camera, refresh, and other controls. The main title "Cluster Management" is centered above a section for adding nodes. A text input field is labeled "Type the name of a node to add to this cluster." and an "ADD NODE" button is on the right. Below this, the "Current Cluster" section lists three nodes: riak@192.168.0.1, riak@192.168.0.2, and riak@192.168.0.3. Each node entry includes an "Actions" column with a "HIDE" button, a green circular status indicator, and the node's IP address. To the right, columns show "Partitions" (34%, 33%, 33%) and "RAM Usage" (19%, 27%, 17%) with corresponding progress bars.

Actions	Name & Status	Partitions	RAM Usage
HIDE	riak@192.168.0.1	34%	19%
HIDE	riak@192.168.0.2	33%	27%
HIDE	riak@192.168.0.3	33%	17%

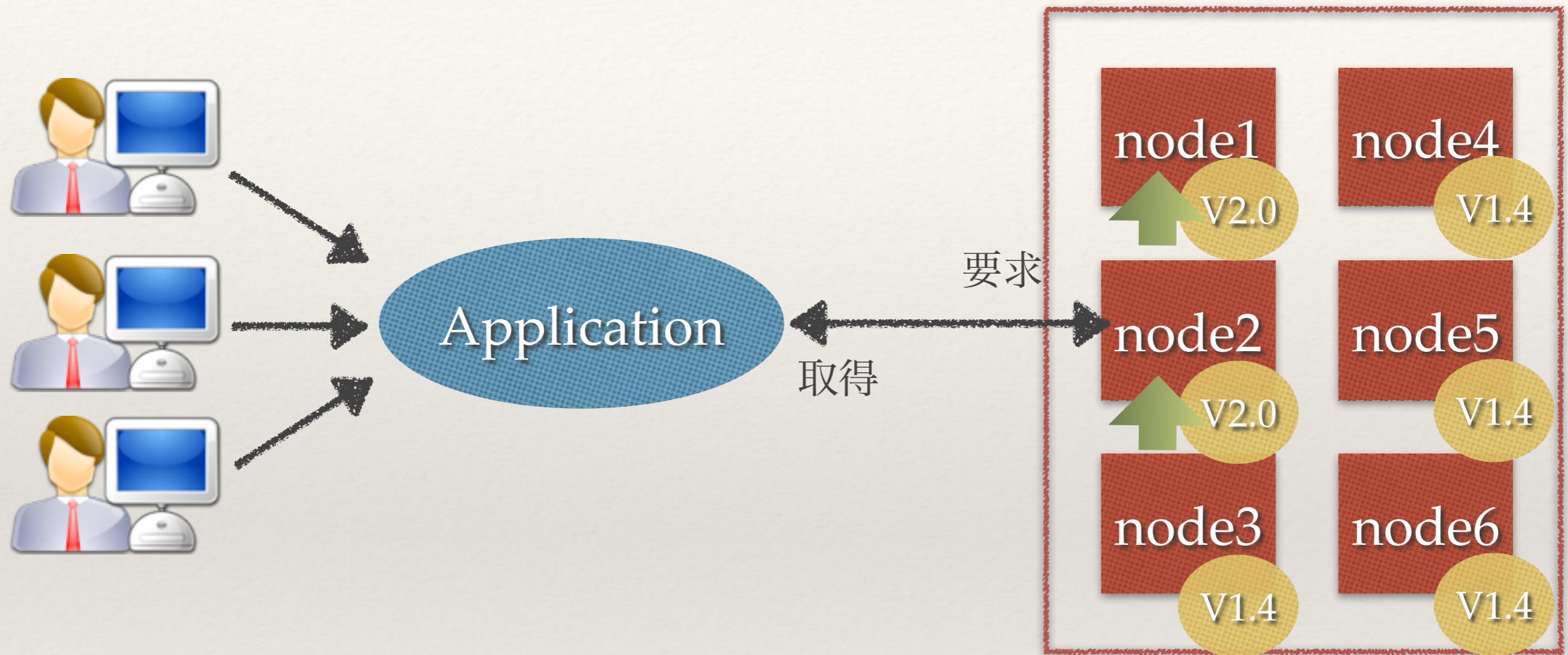
- 管理画面またはコマンドラインから簡単にクラスタを追加できる

分散キーバリューストアRiakの紹介



- 2つ前のバージョンまで互換性があるため、稼働させたままクラスタを順次バージョンアップできる

分散キーバリューストアRiakの紹介



- 2つ前のバージョンまで互換性があるため、稼働させたままクラスタを順次バージョンアップできる

まとめ

- RDBとKVSの使い分け

- 設定・集計済みデータはRDB

- 未集計ログデータや画像データはKVS

- RDBの同期レプリケーションは本当に必要な時だけにしたほうがいいかも

- かえって運用が難しくなりがち

- 開発しやすさだけでなく、運用のしやすさも大事！

- 運用開始後のバージョンアップ（できればサービスを止めずにやりたい）

- オペレーションの簡単さなど（特定の人しか触れないものにならないように）

社員募集のお知らせ

小松電機産業では水関連インフラの監視・制御システムを
一緒に作ってくれる仲間を募集中です！

