

## 何を基準に選定すべきなのか！？ ～ビッグデータ×IoT×AI時代のデータベースの アーキテクチャとメカニズムの比較～

2017年9月7日

東芝デジタルソリューションズ株式会社

ソフトウェア&AIテクノロジーセンター 知識・メディア処理技術開発部

野々村 克彦

# 発表内容

---

## 1.はじめに

- ビッグデータ x IoT x AIで今起きていること
- NoSQL
- スケールアウト型データベースGridDB

## 2.アーキテクチャとメカニズムについて

- GridDBと代表的なNoSQLとの比較

## 3.ビッグデータ x IoT x AI向けのデータベースについて

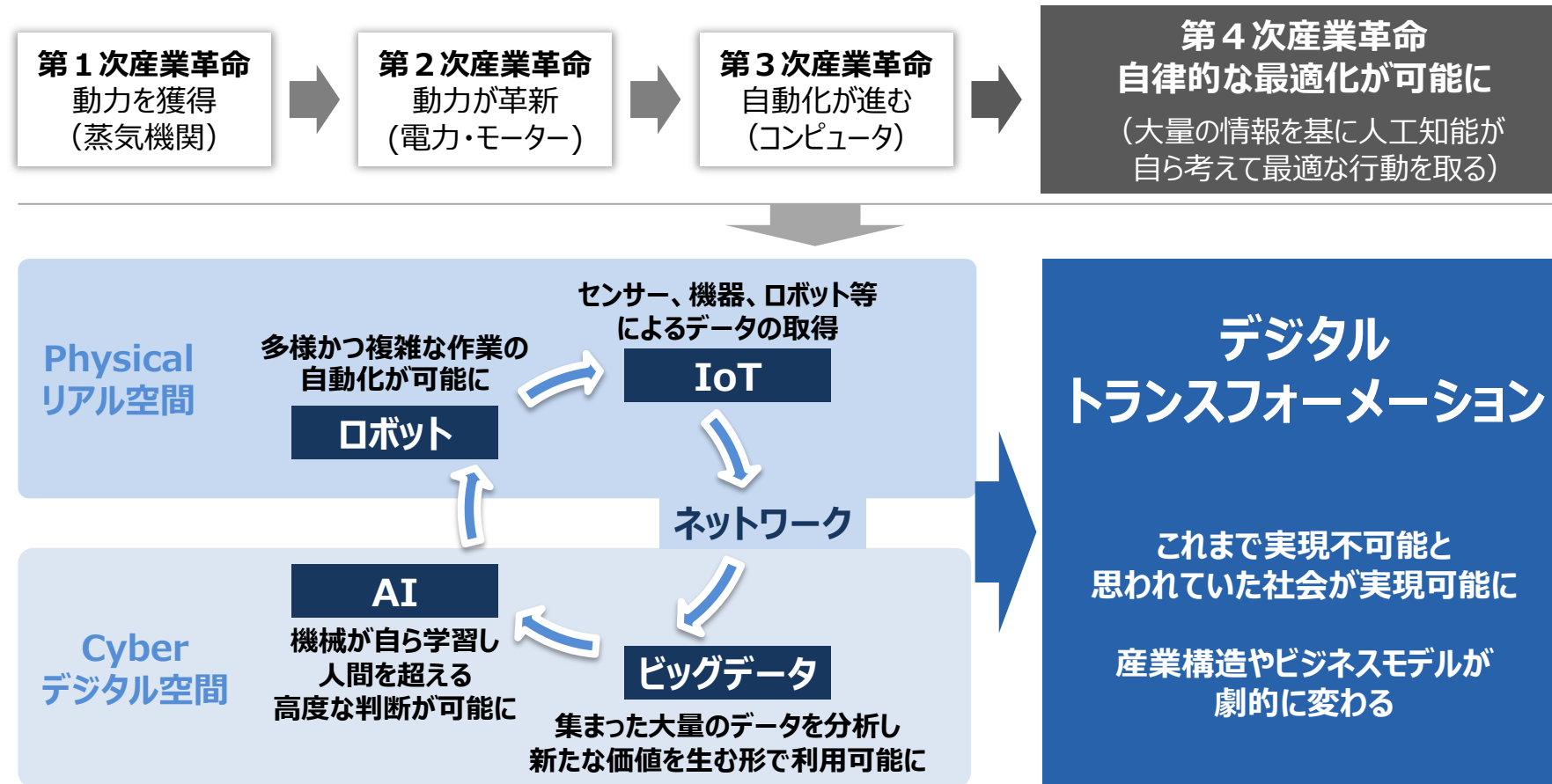
- 選定基準
- GridDBの性能、適用事例

## 4.まとめ

# 今起こっていること デジタルトランスフォーメーション

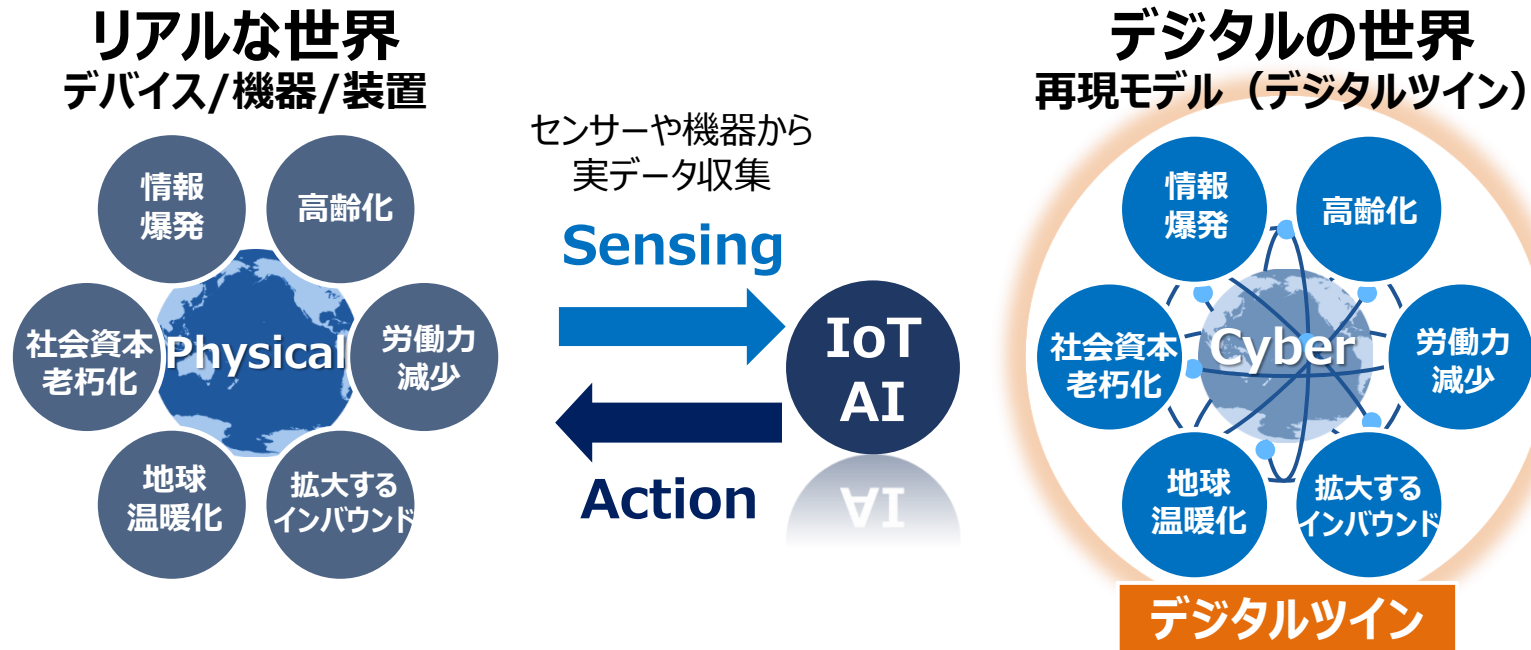
ヒトやモノがデジタル情報やネットワークでつながり  
産業構造やビジネスモデルが劇的に変わる

出典： 経済産業省：IoT、AI、ロボットに関する経済産業省の施策について（2016年3月）  
経済産業省：「新産業構造ビジョン」～第4次産業革命をリードする日本の戦略～（2016年4月）



# ICTによる「見える化」でデジタルトランスフォーメーションを加速

## IoT/AI技術による再現モデル（デジタルツイン）でリアルな世界をシミュレーション



大量のデータを高速、かつ拡張性をもって処理できるデータベースが求められている

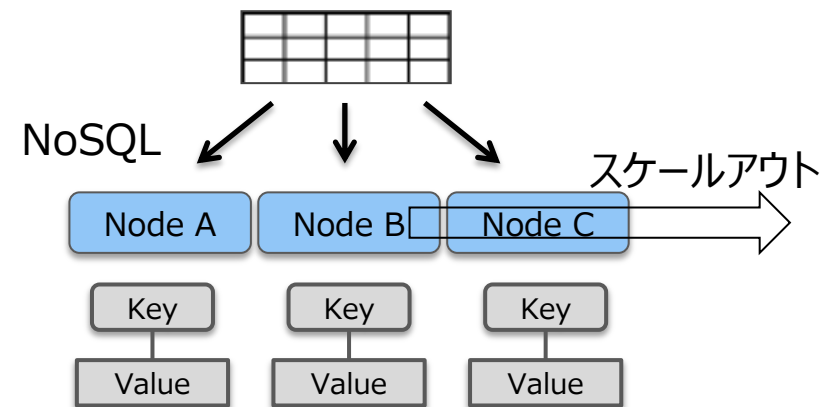
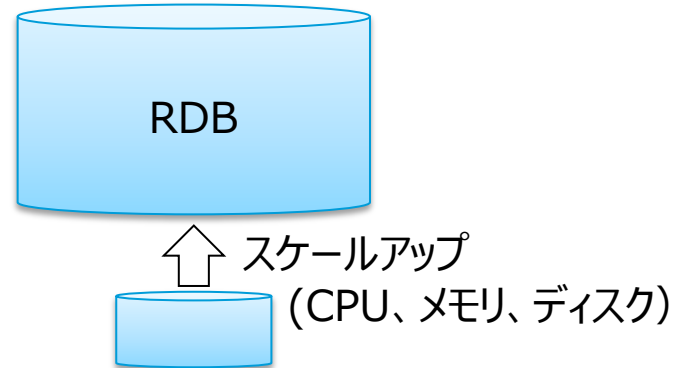
# NoSQLとは

- **RDB**

- スケールアップでは限界がある。ビッグデータを管理するのに適していない
- 一貫性を重視するため、スケールアウトは困難である

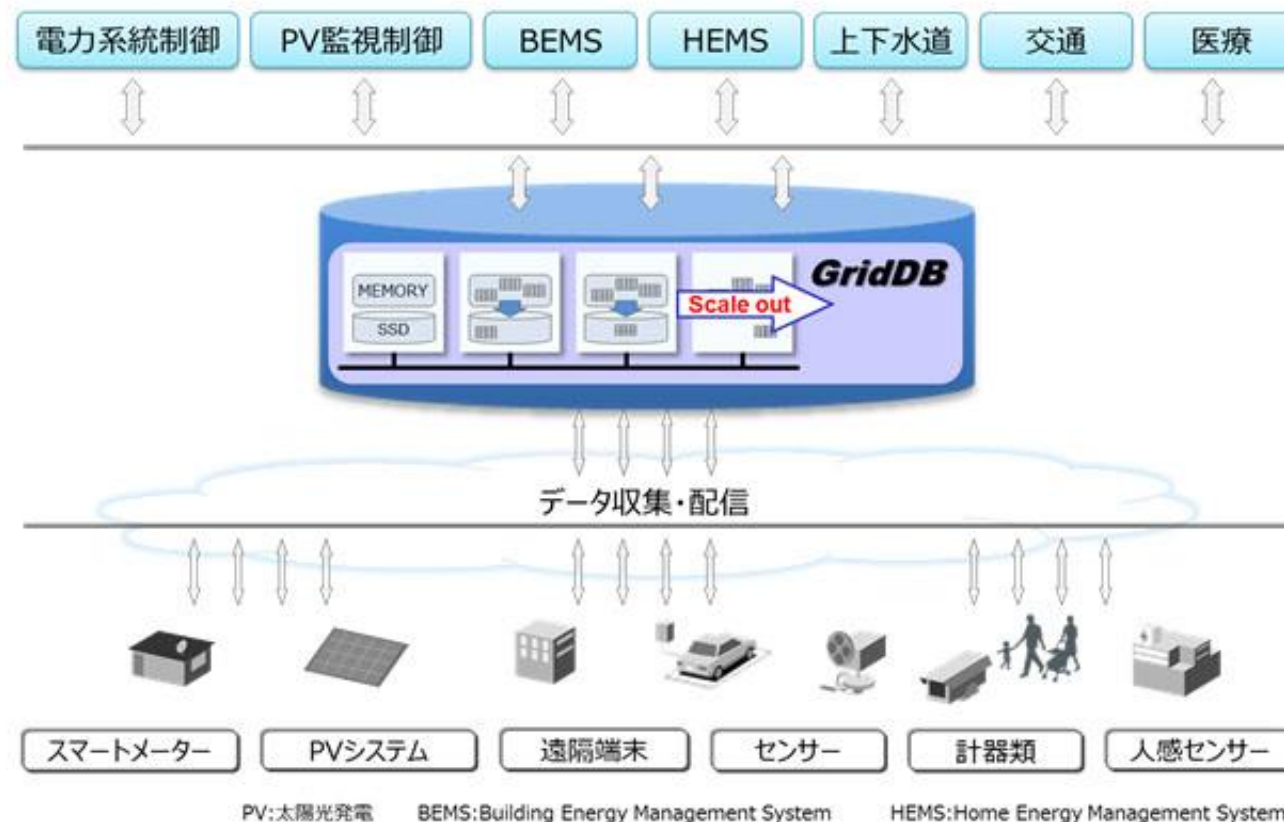
- **NoSQL(Not Only SQL)**

- キーによる更新/参照が基本的なインタフェース（キーバリュー型）
- スケールアウトによる性能向上で近年注目されている
- 一貫性を緩和する代わりにRDBでは対応できない規模の大容量データを管理可能である



# GridDBとは

- ビッグデータ/IoT向けのスケールアウト型データベース
- 開発（2011年～）、製品化（2013年）
- 社会インフラを中心に、高い信頼性・可用性が求められるシステムで使われている

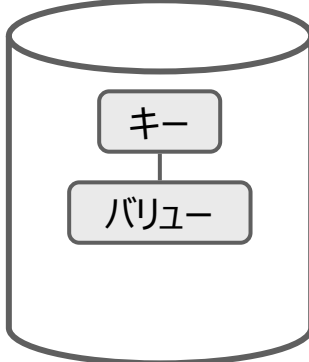
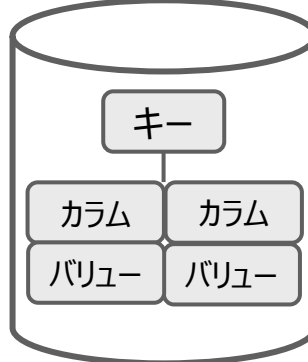
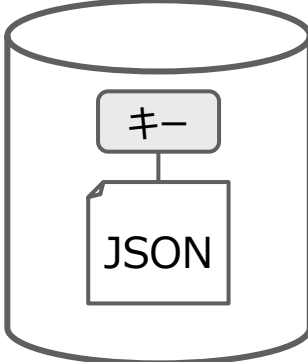
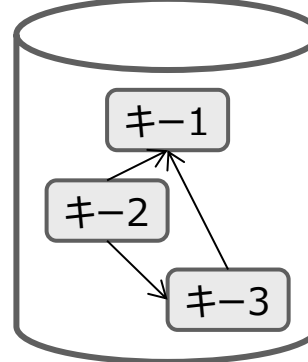
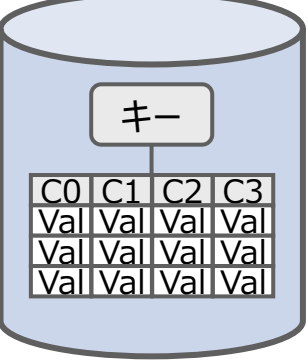


# アーキテクチャとメカニズムの比較

- **GridDBと代表的なNoSQL DBとの比較**
  - Apache Cassandra
  - MongoDB
- **まず、①データモデル、②クラスタ管理、③一貫性と可用性について順に説明します。**
- **続いて、(A)Cassandra、(B)MongoDB、(C)GridDBの順に、クラスタ管理のメカニズムについて説明します。**
  - データ分散方法
  - レプリケーション
  - フェイルオーバー

# ①データモデル

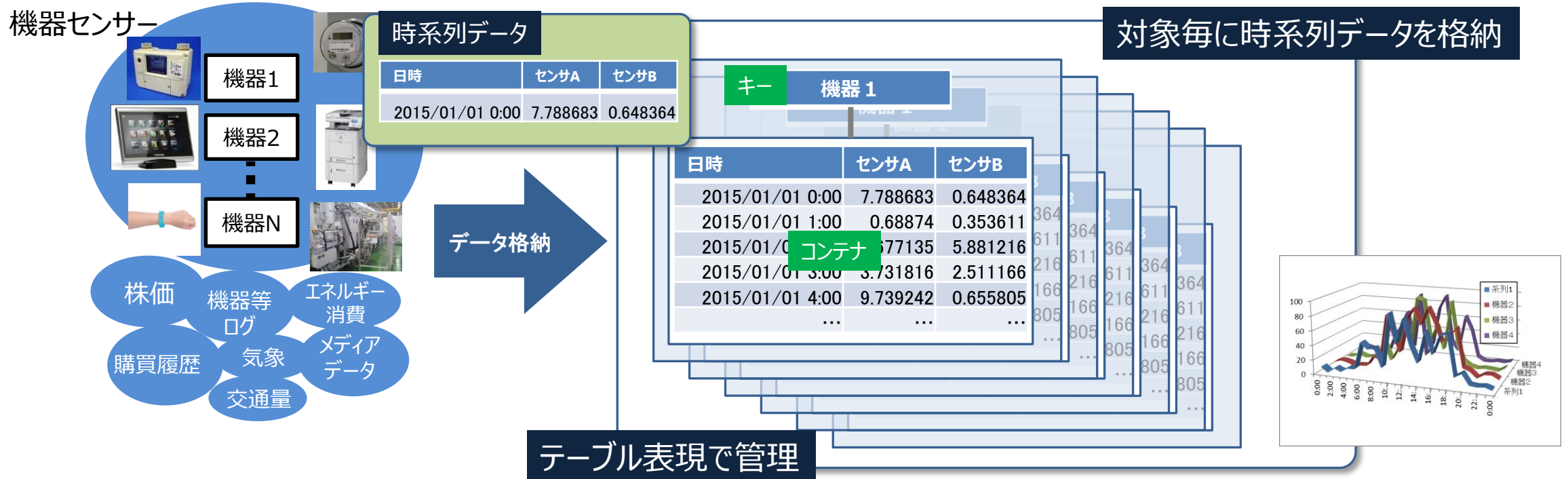
## GridDBはキーコンテナ型

	キーバリュー型	ワイドカラム型	ドキュメント型	グラフ型	キーコンテナ型
データモデル					
NoSQLの例	Riak, Redis	<b>Cassandra</b>	<b>MongoDB</b>	Neo4j	<b>GridDB</b>

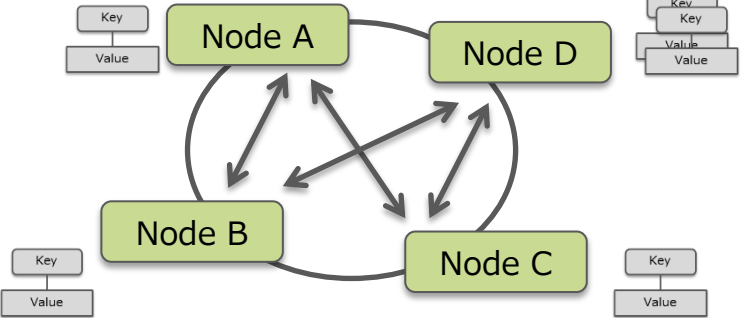
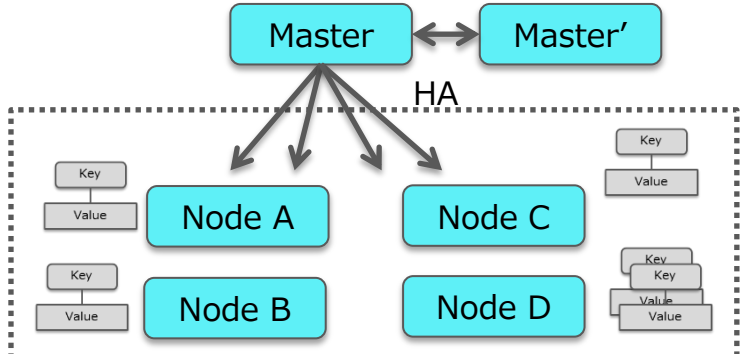


# キーコンテナ型のデータモデル

- データをグループ化するコンテナ（テーブル）
  - ✓ コレクションコンテナ：レコードデータ管理用
  - ✓ 時系列コンテナ：時系列データ管理用。サンプリング、時系列圧縮、期限解放など時系列特有の機能がある
- コンテナ単位でACID保証



## ② クラスタ管理

P2P(Peer to Peer)方式	マスタスレーブ(Master Slave)方式
	
<p>○ノード追加でのデータ再配置が容易 ×一貫性維持のためのノード間通信のオーバーヘッドが大⇒一貫性と処理速度がトレードオフ</p>	<p>○一貫性の維持は容易 ×マスタノードが単一障害点(SPOF) ×ノード追加でのデータ再配置が難しい</p>

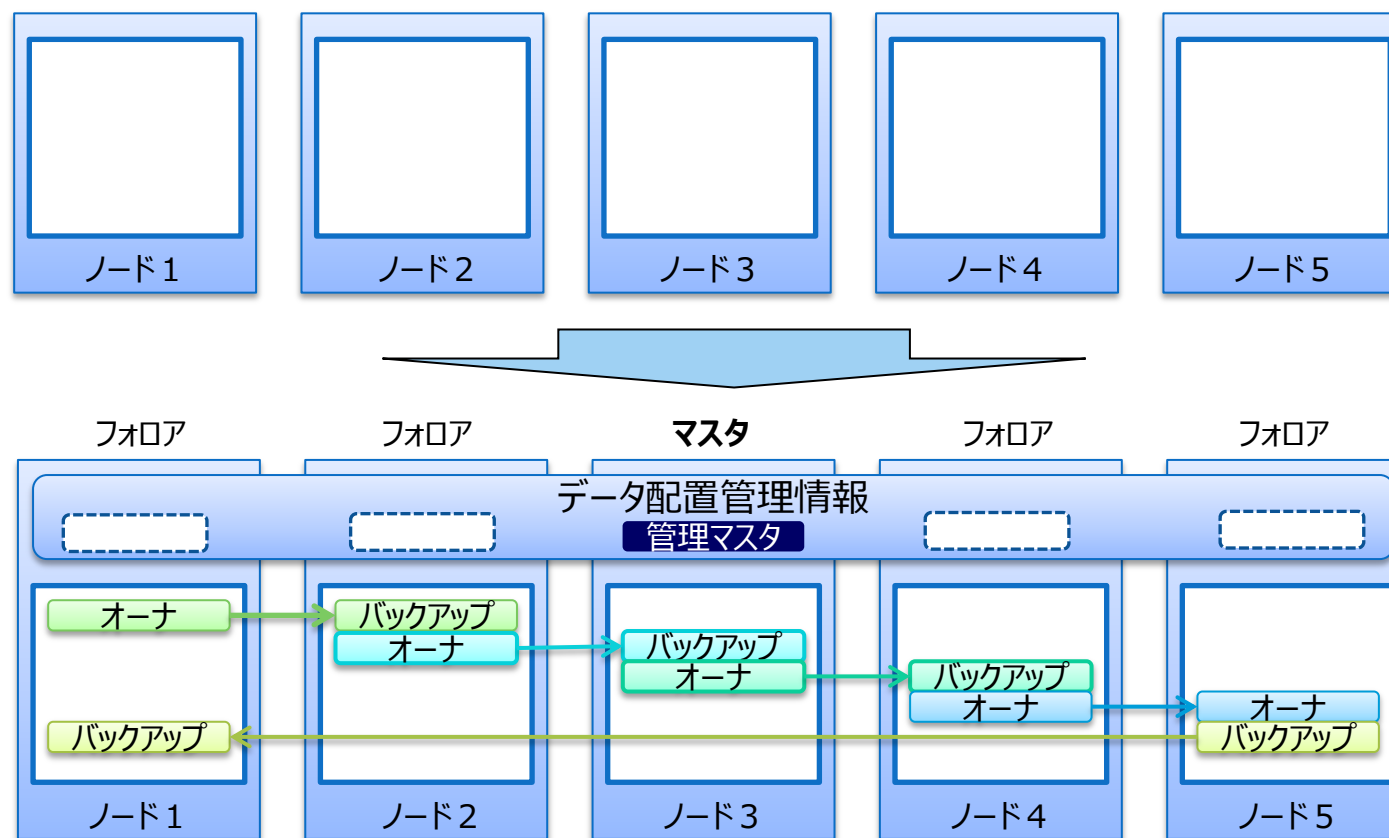
例 : Cassandra

例 : MongoDB

## ② クラスタ管理

### GridDBはハイブリッド型

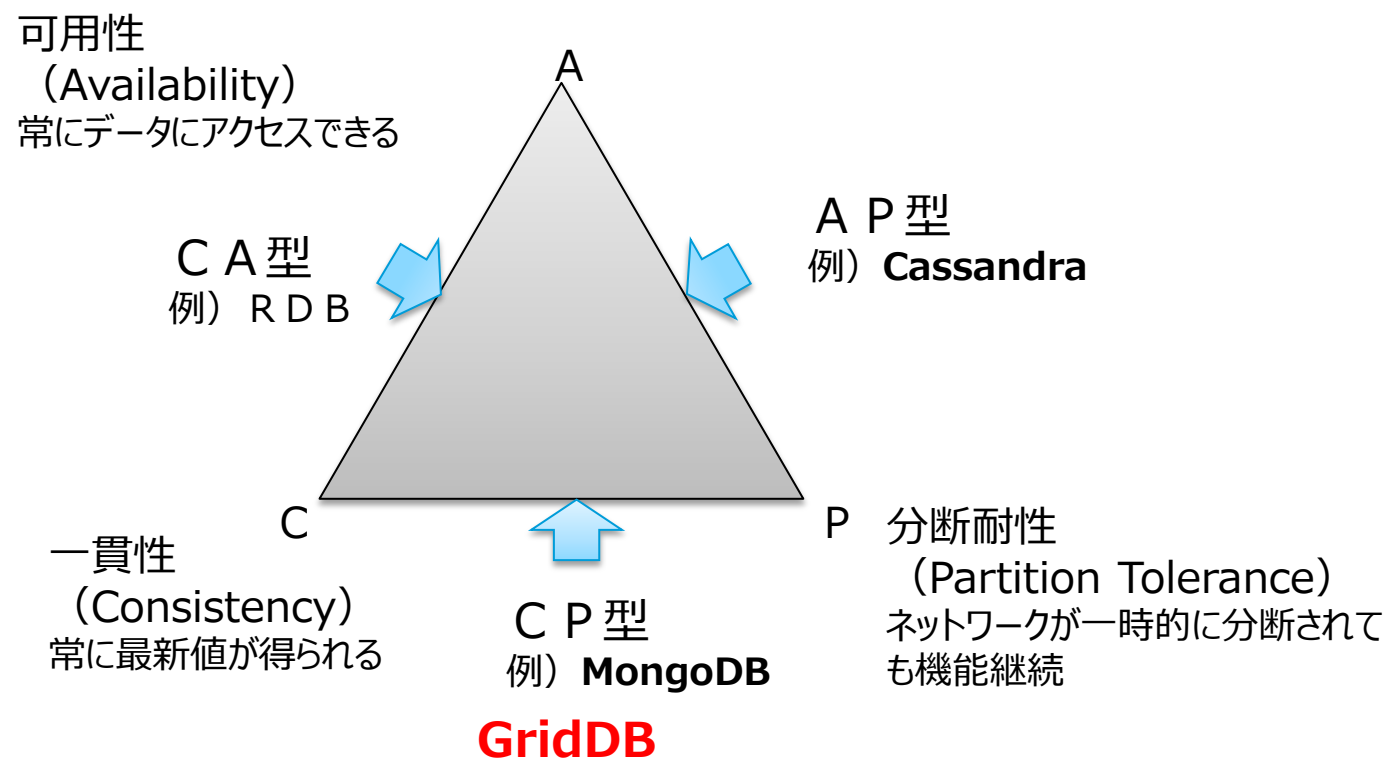
- ノード間で自律的、動的にマスタノードを決定。単一故障点（SPOF）を排除
- マスタがデータ配置（オーナ/バックアップ）を決定



### ③一貫性と可用性

## GridDBはCP型

- **CAP定理** : E. Brewer, "Towards Robust Distributed Systems"[1]



[1] Proc. 19th Ann. ACM Symp. Principles of Distributed Computing (PODC 00), ACM, 2000, pp. 7-10;

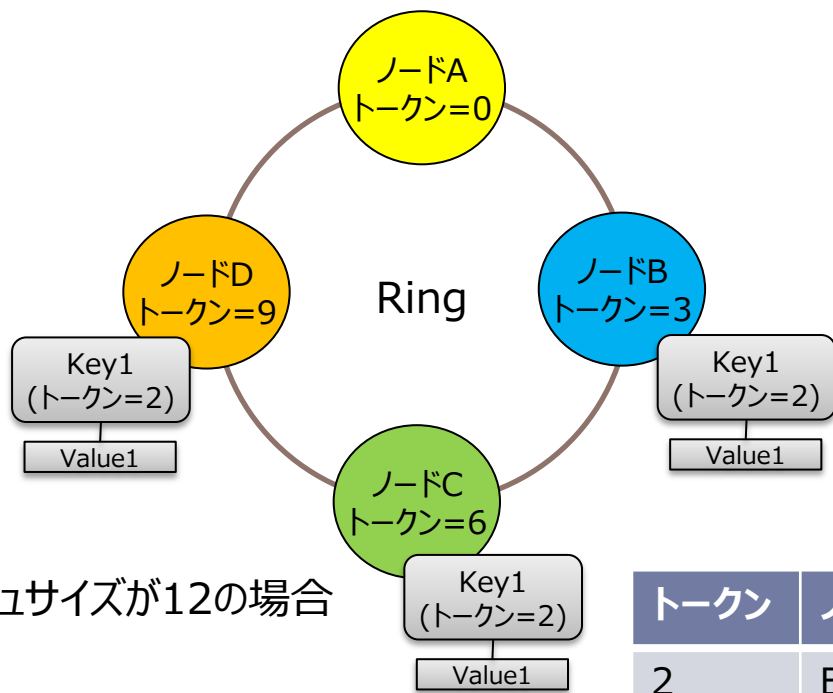
# アーキテクチャとメカニズムの比較

- **GridDBと代表的なNoSQL DBとの比較**
  - Apache Cassandra
  - MongoDB
- **まず、①データモデル、②クラスタ管理、③一貫性と可用性について順に説明します。**
- **続いて、(A)Cassandra、(B)MongoDB、(C)GridDBの順に、クラスタ管理のメカニズムについて説明します。**
  - データ分散方法
  - レプリケーション
  - フェイルオーバー

# (A) Cassandra

## ● 単一トークンによるコンシステントハッシュ法

データ分散の例：SingleStrategy  
まずハッシュ関数で決められたノードに配置。  
続いて時計回りでリング状の次のノードに配置。



トークン	ノード
2	B, C, D

データモデル	ワイドカラム型
クラスタ管理	P2P方式
CAP	AP型
データ分散方法	コンシステント・ハッシュ
レプリケーション	プライマリ/セカンダリ の区別なし
フェイルオーバ	P2P方式

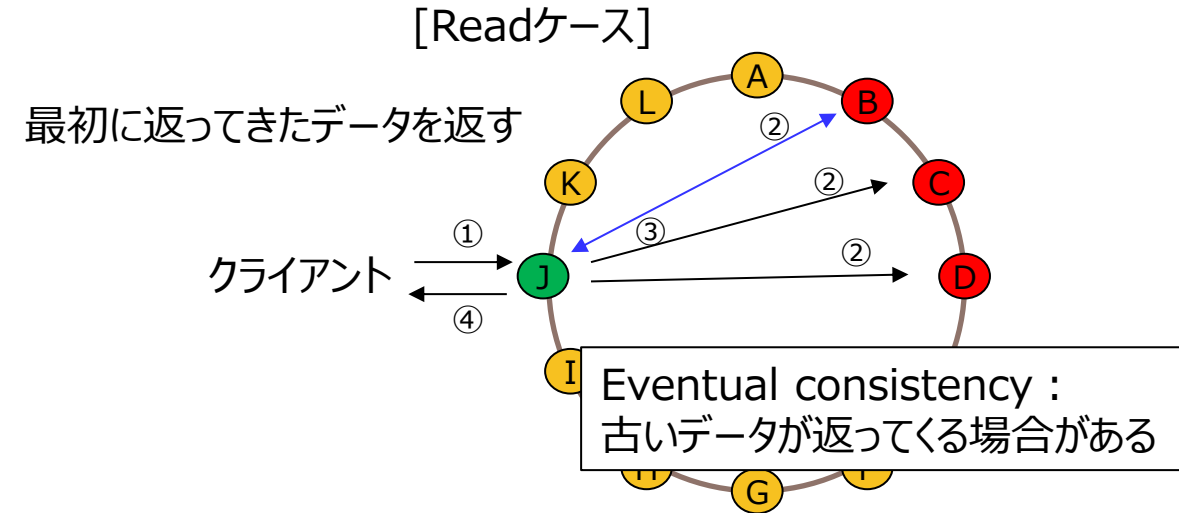
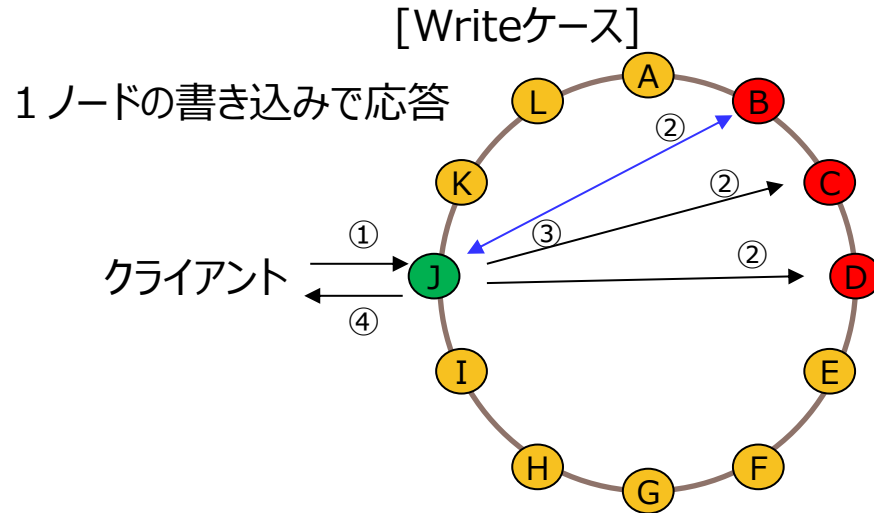
最新バージョン：3.11

# 一貫性の調整 (ONE、ALL)

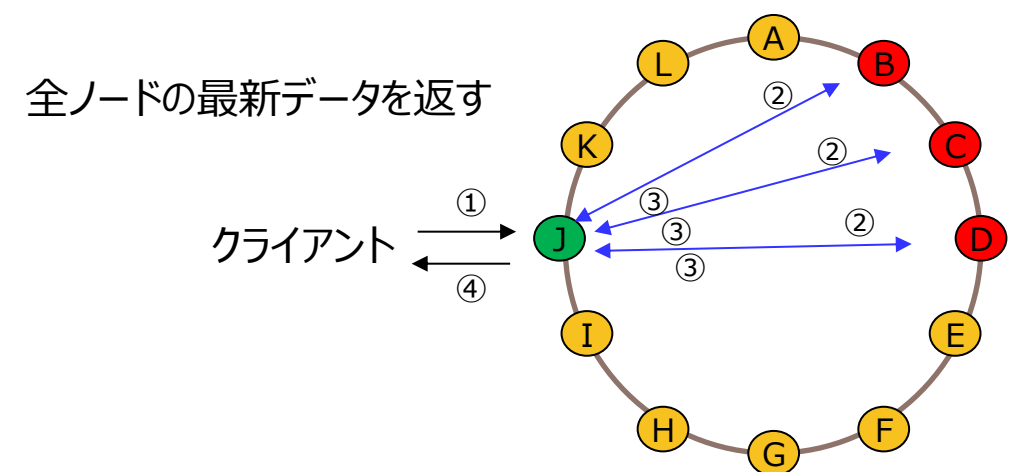
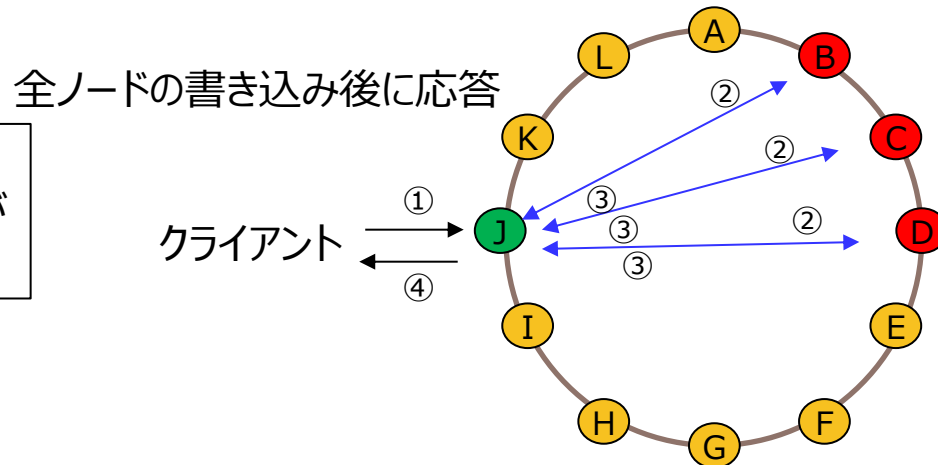
## 一貫性と可用性・性能はトレードオフ

※レプリカ数が3の場合

(ONE)



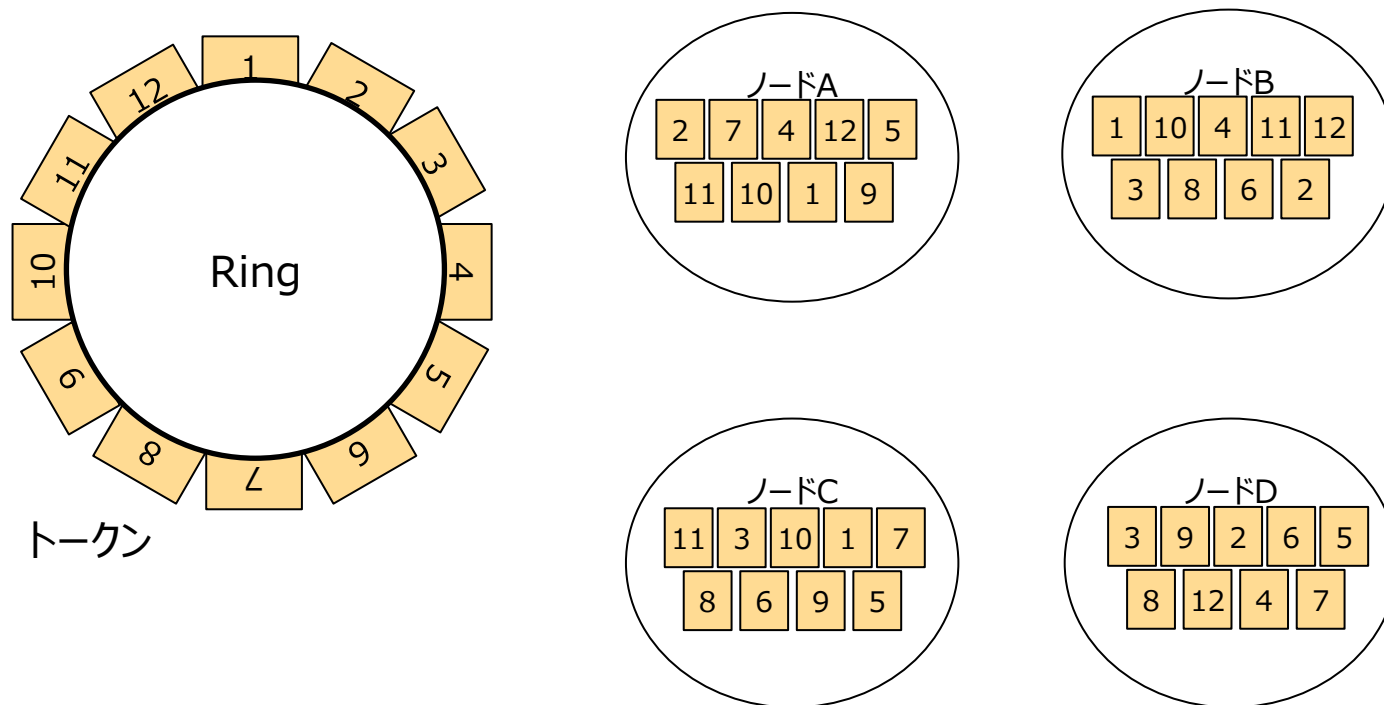
(ALL)  
一貫性は高くなるが  
可用性は低くなる



# ①稼働状態

## ● 仮想ノード(V1.2～)によるコンシステントハッシュ法

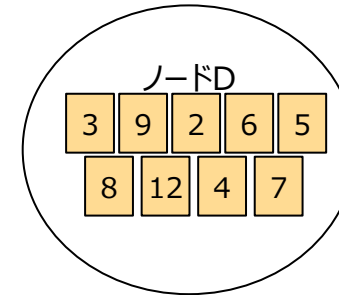
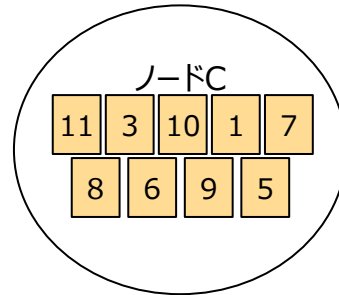
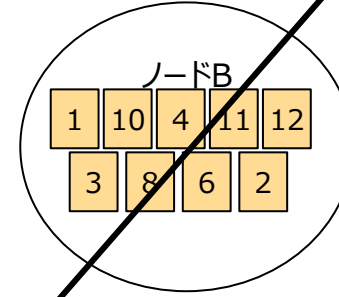
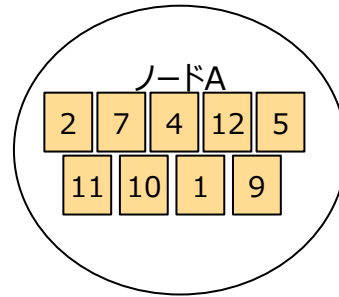
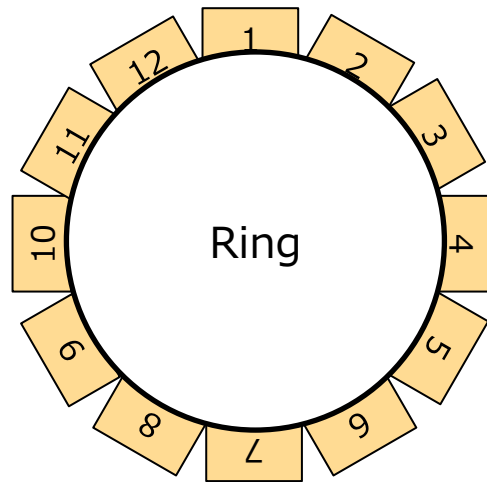
※レプリカ数が3の場合



1つのノードに複数のランダムなトークン値を割り当。担当範囲を複数に分割

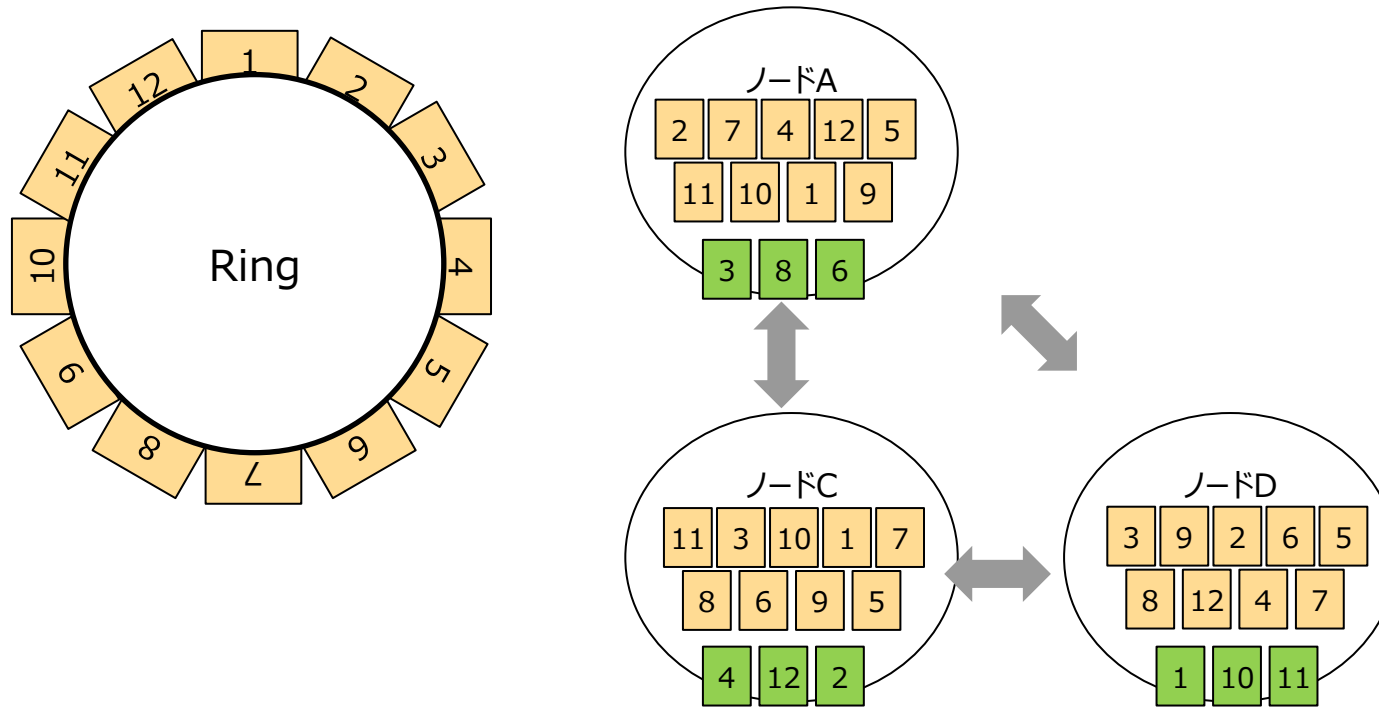


## ② ノードBがノードダウン



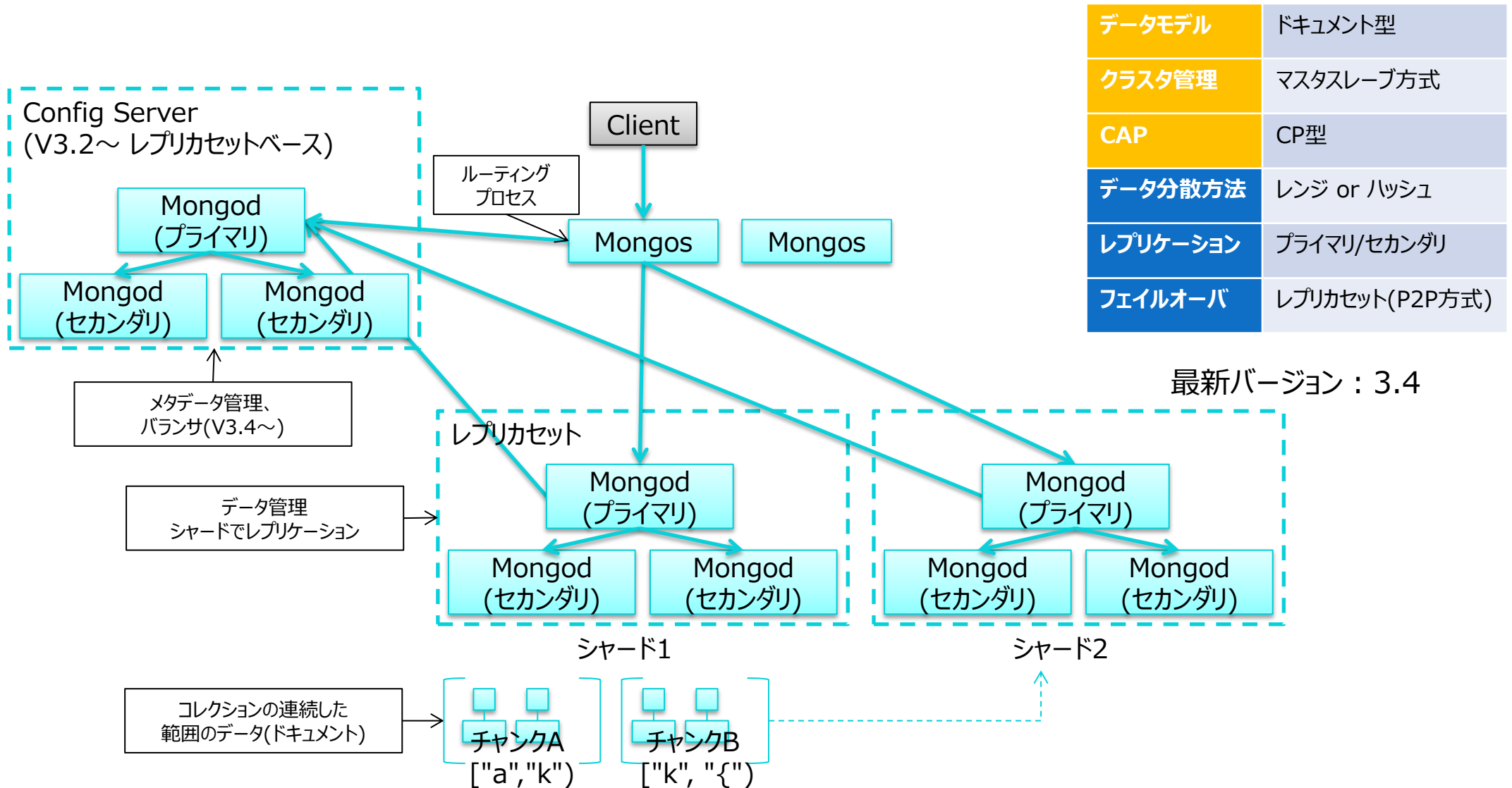
### ③安定状態

## P2Pにより自動バランシング



P2Pでトークンを再配置してバランシング

# (B) MongoDB



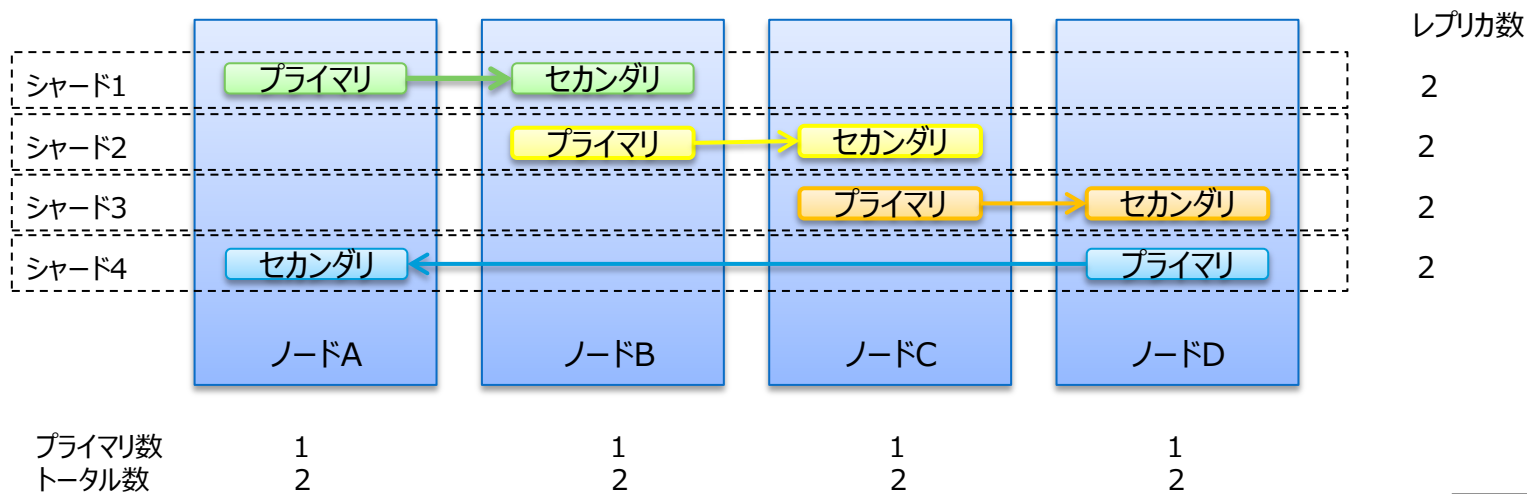
# MongoDBの歴史（アーキテクチャ関連）

- **V1.0(2009)**
  - Master/Slave、replica pair
- **V1.6(2010/8)**
  - シャーディング機能とレプリカセット機能を追加
- **V1.8(2011/3)**
  - ジャーナリング機能
  
- **V3.0(2015/3)**
  - config serverがミラーリングから3台2PC方式に変更
- **V3.2(2015/12)**
  - config serverが3台2PC方式からレプリカセットに変更
- **V3.4(2016/11)**
  - バランサがmongosからconfig serverに変更
  - マイグレーション同時実行

# ①稼働状態

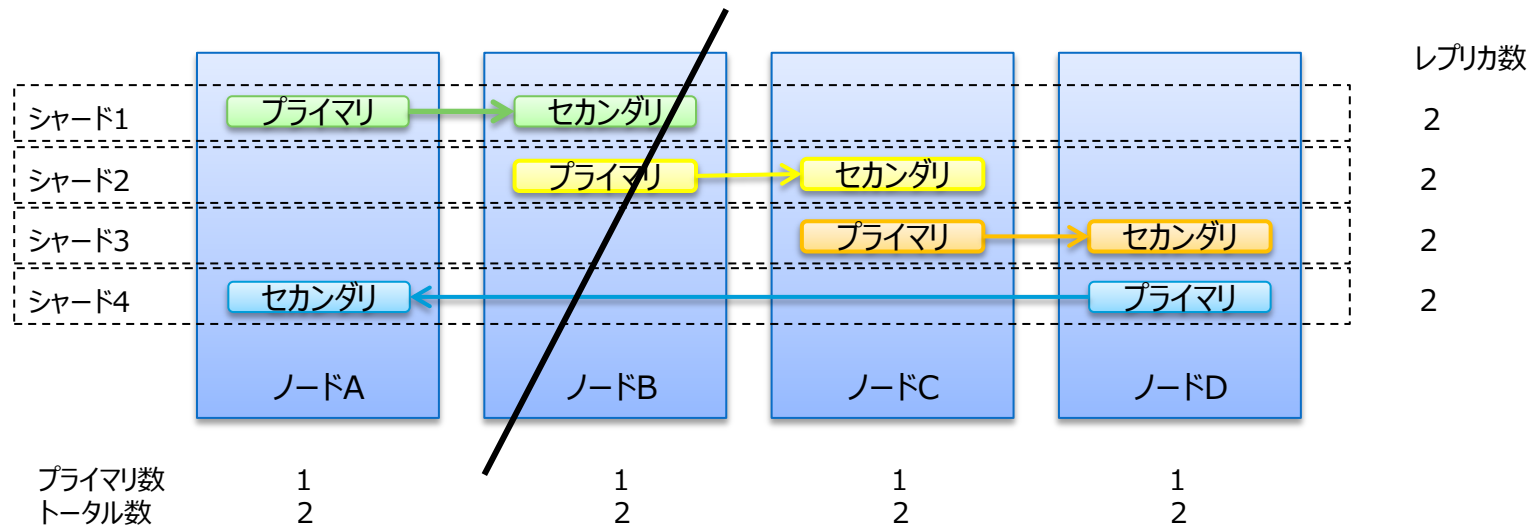
サーバ(プロセス)数=シャード数×レプリカ数

※Config Server以外のMongodサーバを同一ノード内に混在させた場合  
※レプリカ数が2の場合



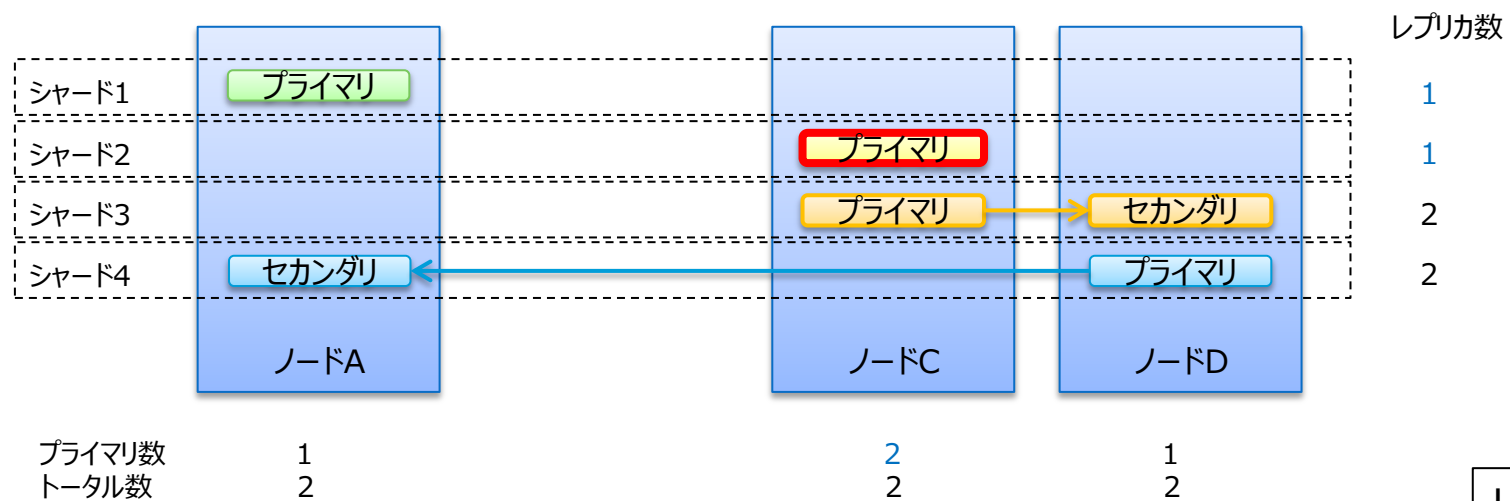
4ノード、8サーバ(プロセス)、4シャード  
各ノードのプライマリは1個、セカンダリは1個  
レプリカ数はすべて2個

## ② ノードBがノードダウン



### ③フェイルオーバ

## P2Pでセカンダリがプライマリに自動切り替え



レプリカ数の少ないものが発生  
...サーバの手動追加が必要  
プライマリ数の偏りが発生  
...プライマリの手動切替が必要

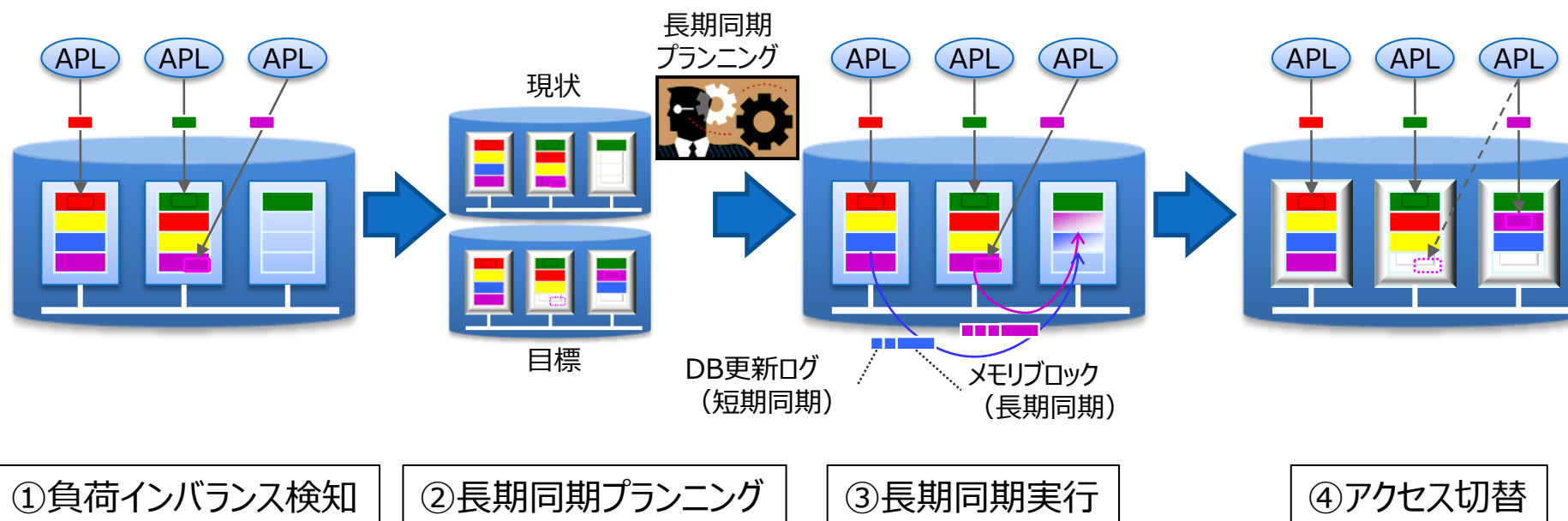




# 自律データ再配置技術 (ADDA)

## ADDA : Autonomous Data Distribution Algorithm

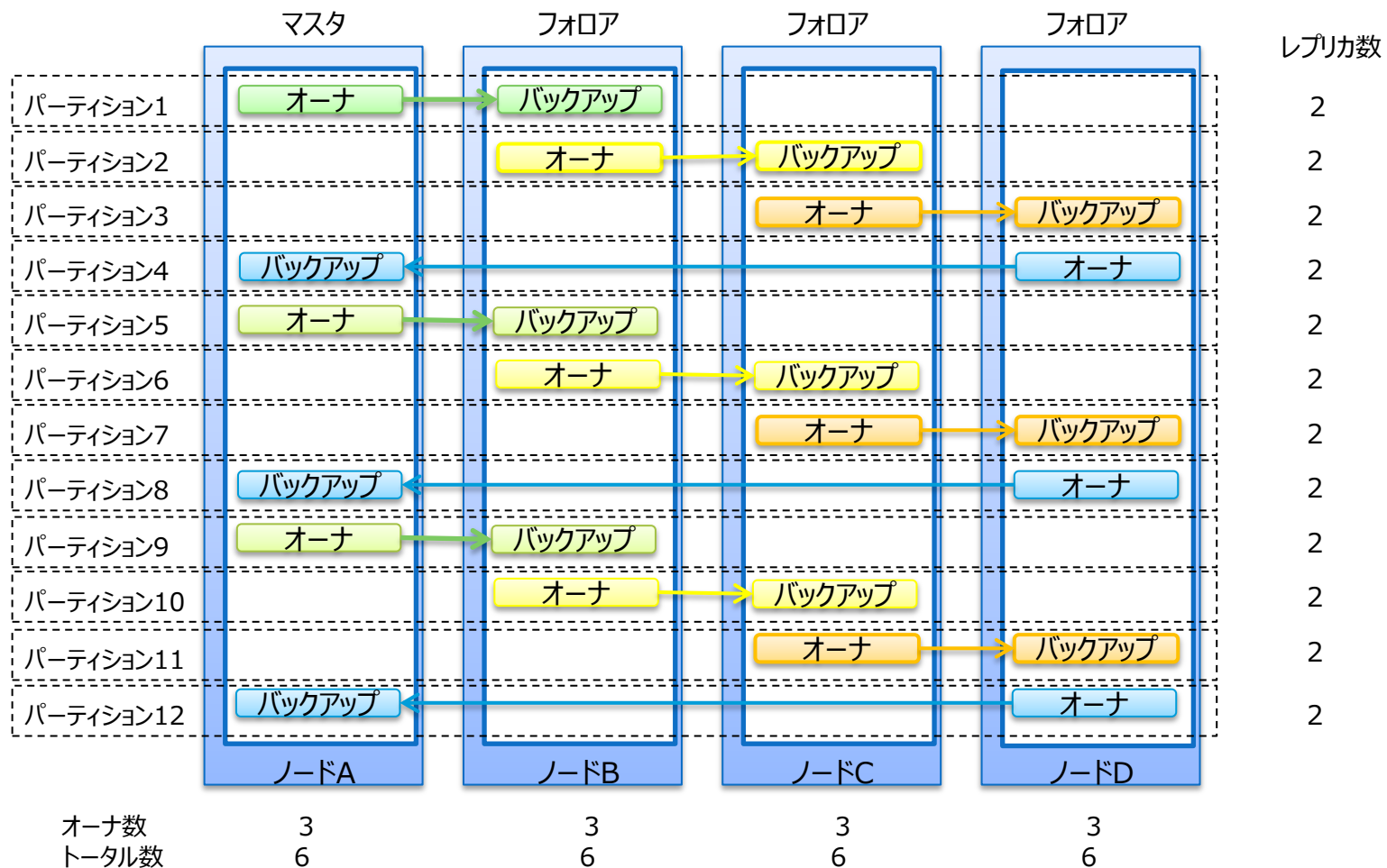
- インバランス状態を検知、長期同期プランニング
- 2種類のデータを使ってバックグラウンド高速同期、完了後切替
  - ✓ DB更新ログ、メモリブロック



# ①稼働状態

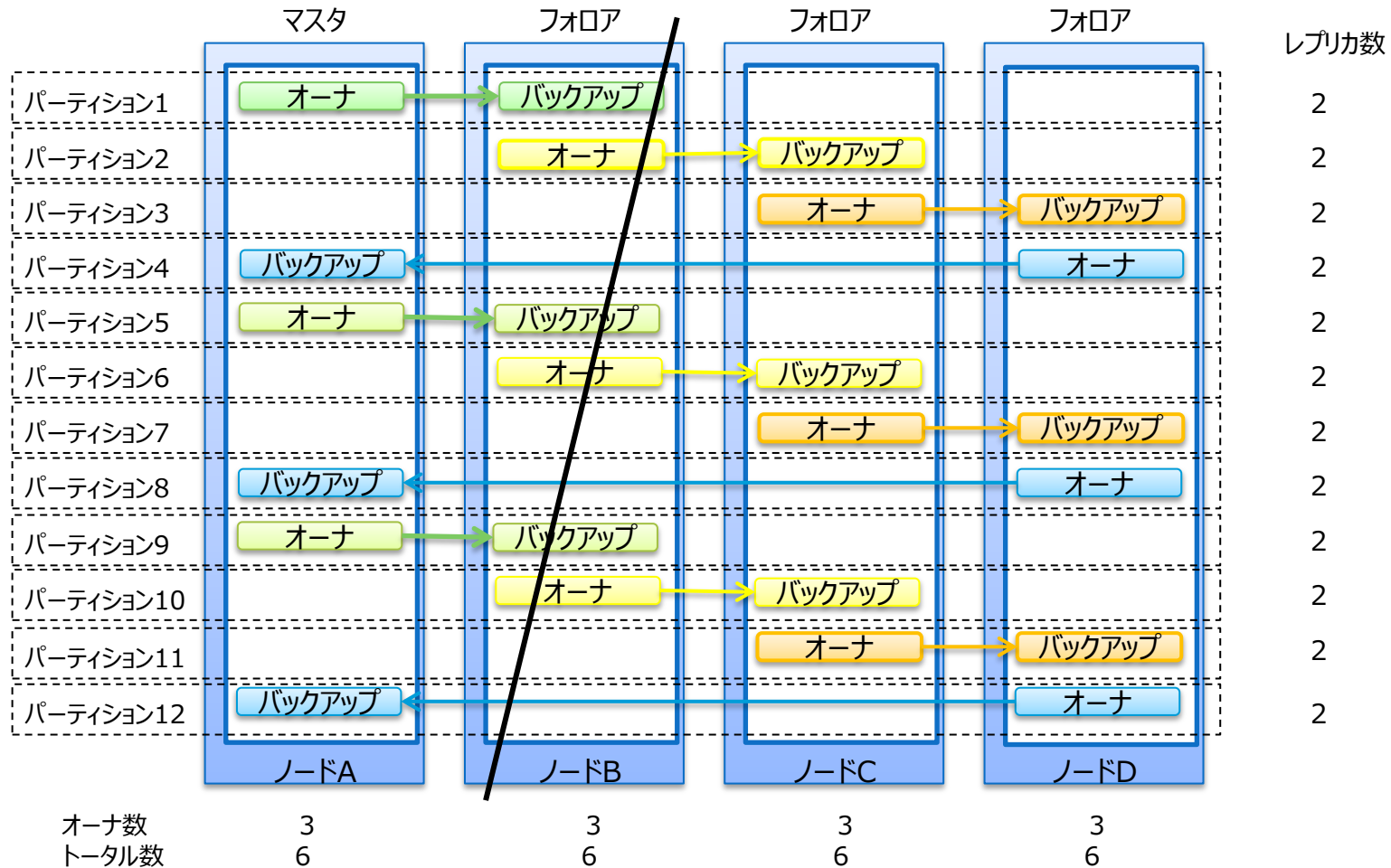
## ノード、パーティション単位にデータ分散配置

※レプリカ数が2の場合



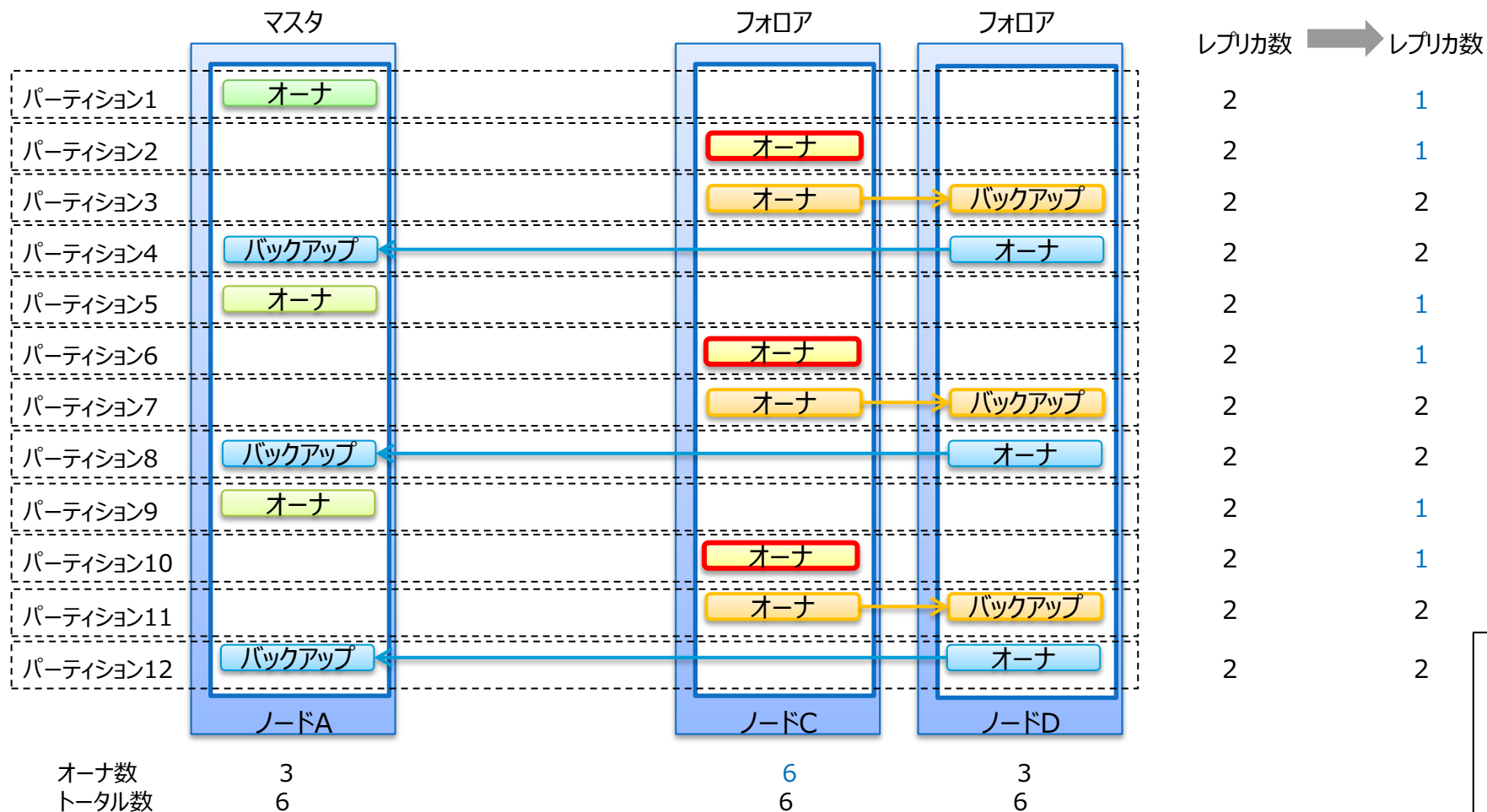
4ノード、4サーバ、12パーティション  
各ノードのオーナーは3個、バックアップは3個  
レプリカ数はすべて2個

## ② ノードBがノードダウン



# ③フェイルオーバ

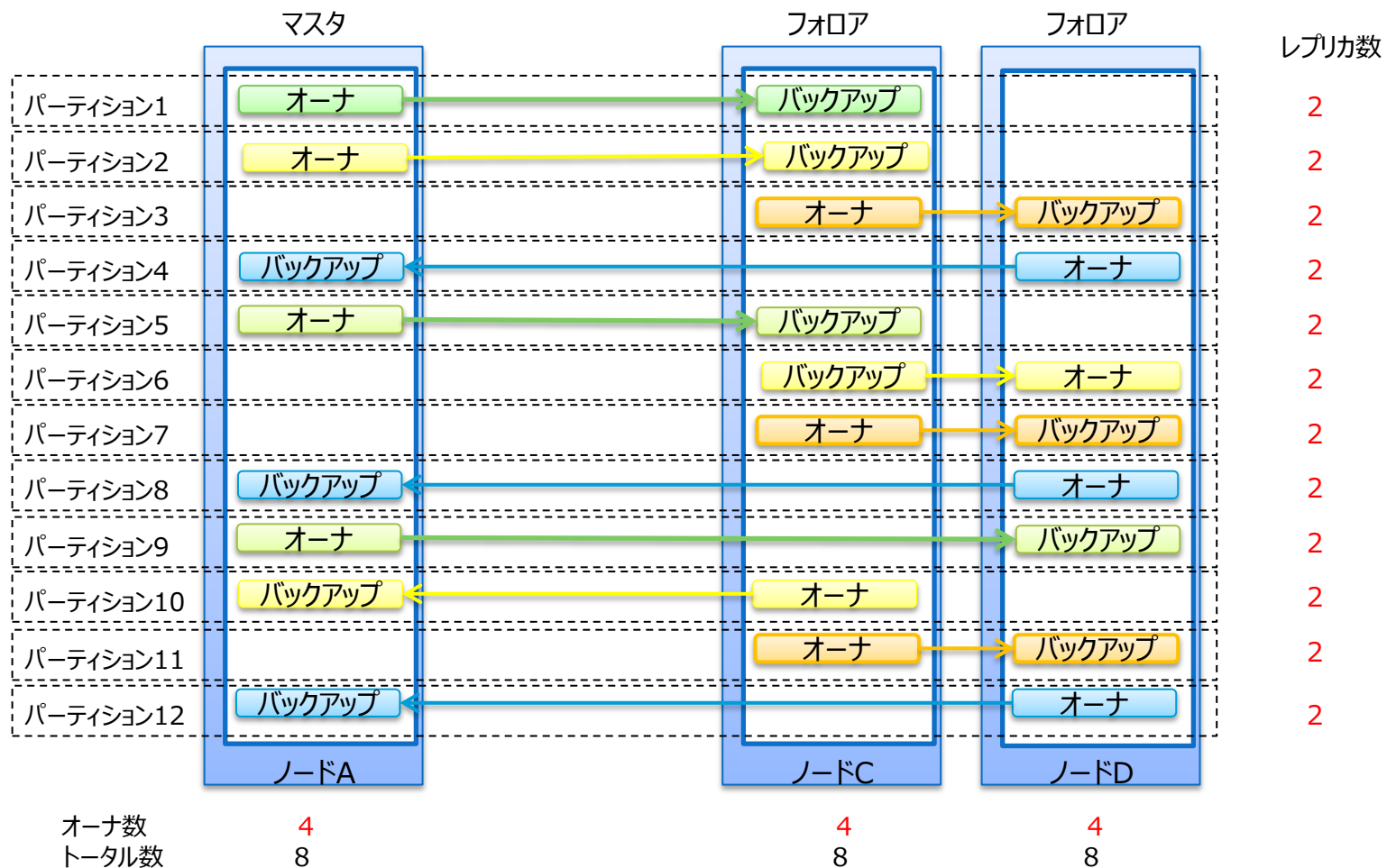
## マスタが即座にバックアップをオーナに自動切り替え



レプリカ数の少ないものが発生  
 オーナ数の偏りが発生

# ④安定状態

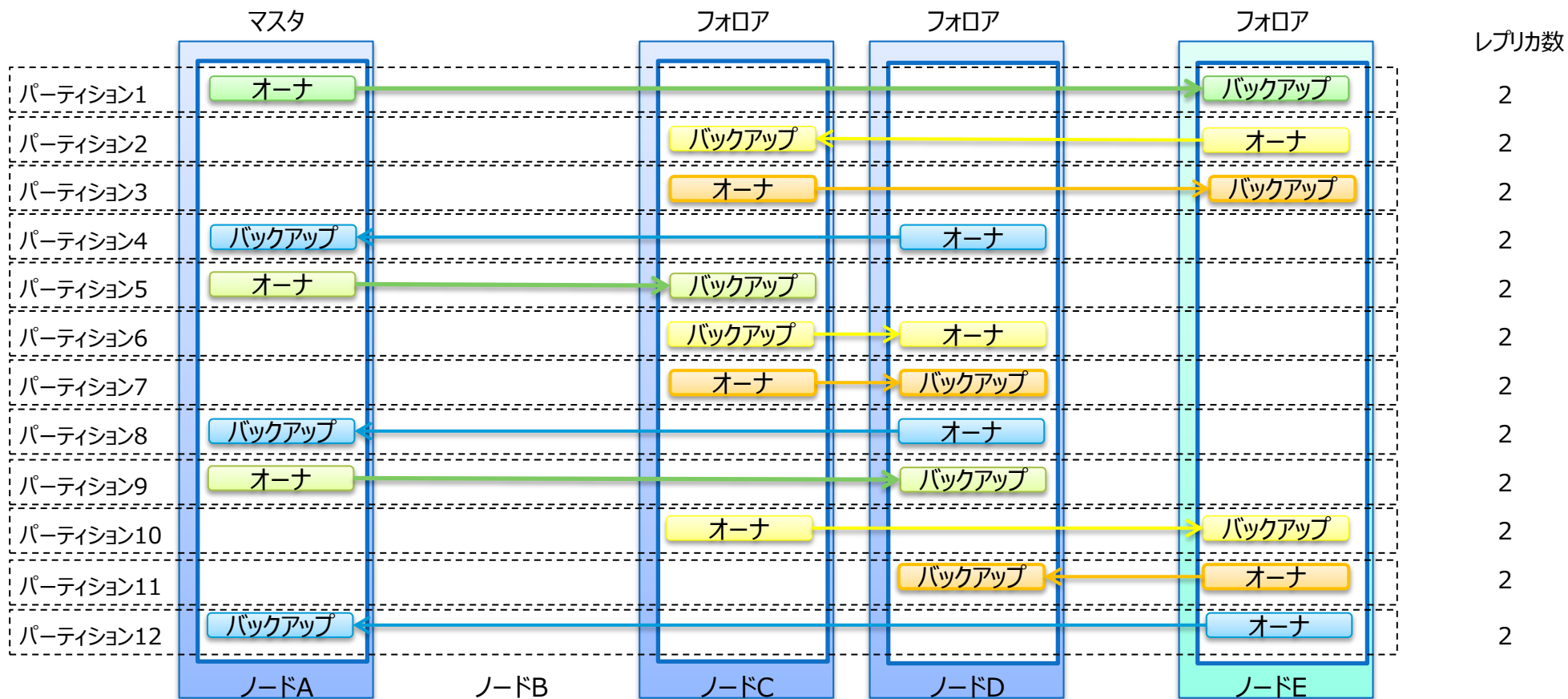
## ADDAにより自動バランス



レプリカ数の少ないものが発生  
 ⇒レプリカ数はすべて2個に  
 オーナ数の偏りが発生  
 ⇒オーナー数はすべて4個に

# ⑤ ノード追加⇒安定状態

## ADDAによりオンラインスケールアウト。自動バランシング



オーナー数 3  
トータル数 6

3  
6

3  
6

3  
6

各ノードのオーナーは4個から3個に

# アーキテクチャ・メカニズムの比較（まとめ）

	Cassandra	MongoDB	GridDB
データモデル	ワイドカラム型	ドキュメント型	<b>キー・コンテナ型</b>
クラスタ管理	P2P方式	マスタスレーブ方式	<b>ハイブリッド型</b> (P2Pによるマスタ自動選出)
CAP	AP型	CP型	CP型
シャーディング	コンシステント・ハッシュ	ハッシュ or レンジ	ハッシュ
レプリケーション	プライマリ/セカンダリ の区別なし	プライマリ/セカンダリ (サーバ単位)	オーナ/バックアップ (パーティション単位)
フェイルオーバ	P2P方式	P2P方式	ハイブリッド型

一貫性(C)/可用性(A)を 高める仕掛け	調整可能な一貫性レベル (一貫性アップ)	P2Pによるレプリカセット (可用性アップ)	<b>ADDA</b> (可用性アップ)
一貫性	性能とトレードオフ	常に最新データ	常に最新データ
可用性、スケーラビリティ	P2Pによる自動バランシング	レプリカ数は減ったまま ⇒レプリカ追加操作が必要	マスタによる自動バランシング (ADDA)

---

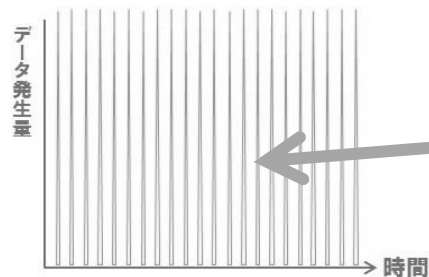
# ビッグデータ x IoT x AI向けのデータベースについて



# ビッグデータ×IoT×AI向けのデータベースの選定基準

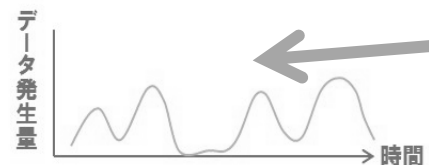
秒、ミリ秒オーダーで  
高頻度に発生

・高頻度になっても大丈夫か？



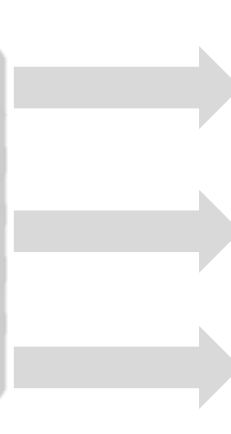
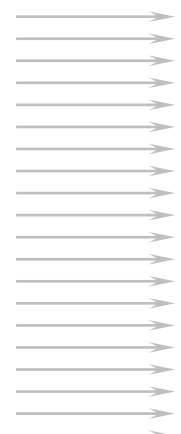
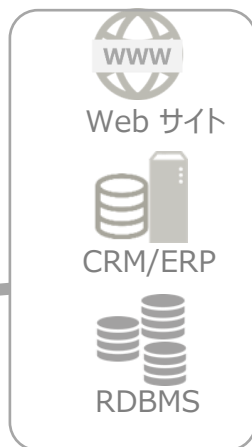
24H365D  
絶え間なく発生

・24H365D稼働でも大丈夫か？  
・復旧が容易か？



大量データが  
単調増加

・膨大な機器数でも大丈夫か？  
・ノード追加が容易か？



見える化  
各種集計  
異常検出

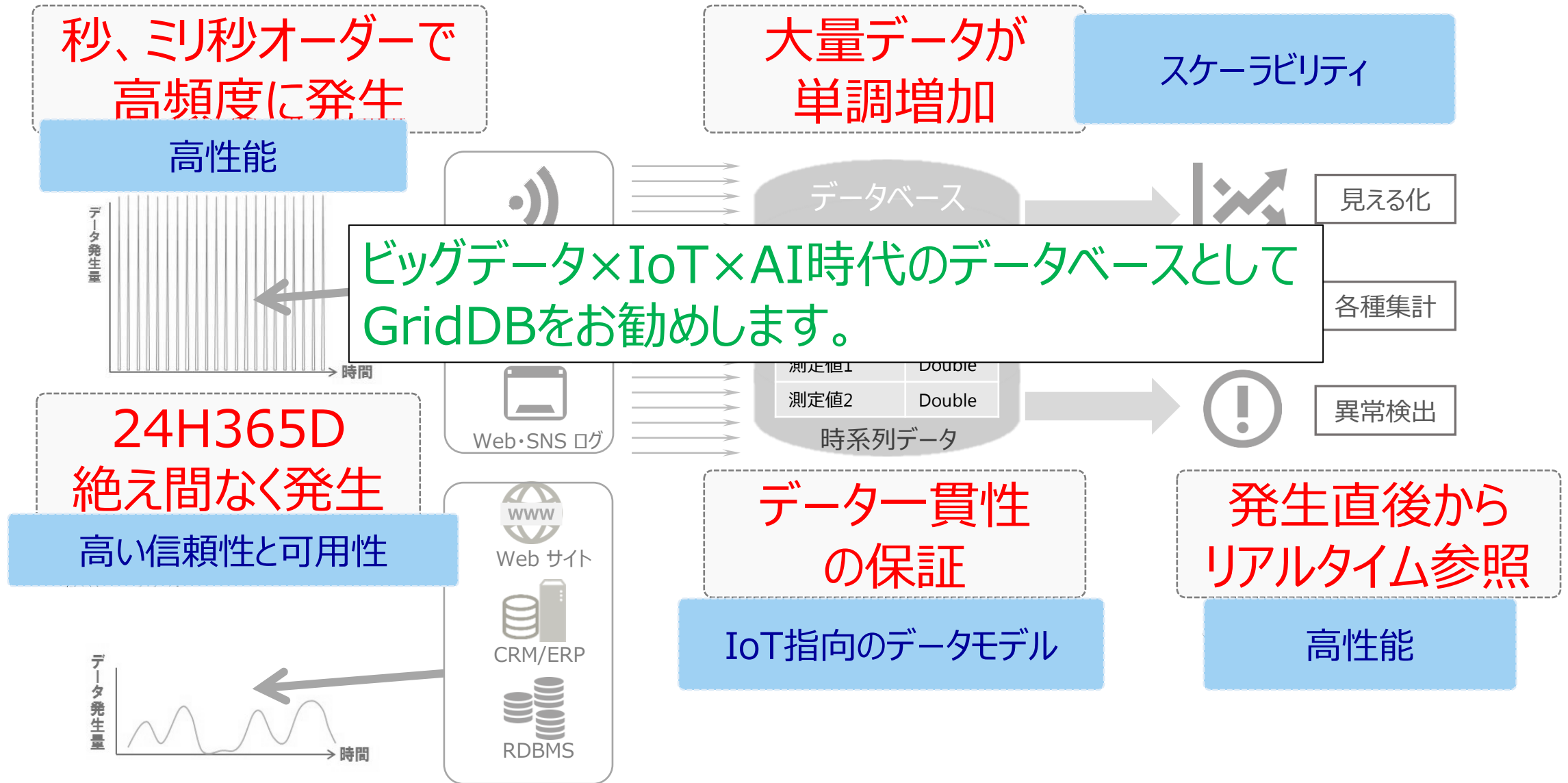
データ一貫性  
の保証

・常に最新データを取得できるか？  
・機器ごとに一貫性を保てるか？

発生直後から  
リアルタイム参照

・即座に処理できるか？

# ビッグデータ×IoT×AI向けのデータベースの選定基準



# GridDB 4つの特長

## IoT指向の データモデル

- データ集計やサンプリング、期限解放、データ圧縮など、時系列データを効率よく処理・管理するための機能を用意

**キーコンテナ型**

- データモデルはユニークなキーコンテナ型。コンテナ内でのデータ一貫性を保証

## 高性能

- メモリを主、ストレージを従としたハイブリッド型インメモリーDB
- メモリやディスクの排他処理や同期待ちを極力排除したオーバヘッドの少ないデータ処理により高性能を実現

## スケーラビリティ

- データの少ない初期は少ないサーバで初期投資を抑え、データが増えるにしたがってサーバを増やし性能・容量を高めるスケールアウト型アーキテクチャ
- コンテナによりサーバ間通信を少なくし、高い

**ハイブリッド型のクラスタ管理  
ADDA**

## 高い信頼性と 可用性

- データ複製をサーバ間で自動的に実行し、サーバに障害が発生しても、システムを止めることなく運用を継続することが可能

# Cassandraとの性能比較 (YCSB)

- **YCSB(Yahoo! Cloud Serving Benchmark)**

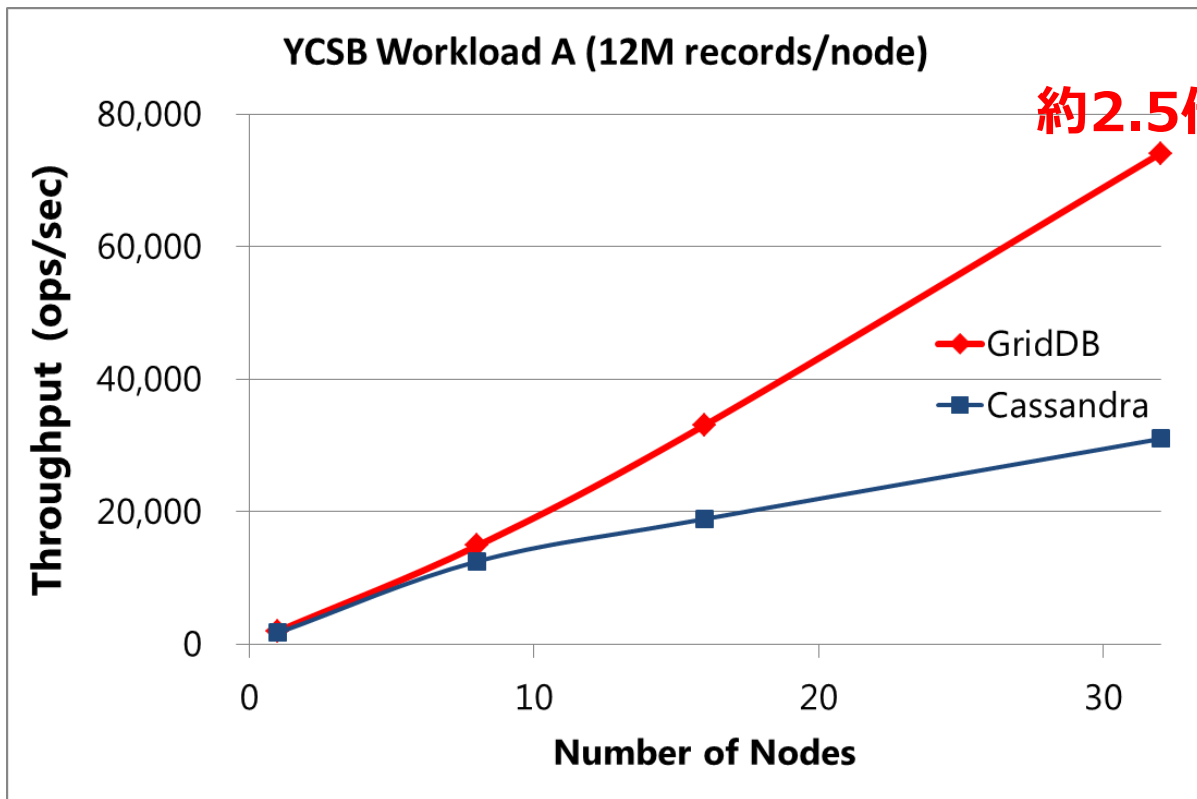
<https://github.com/brianfrankcooper/YCSB/wiki>

- NoSQLの代表的なベンチマーク
- Load/Runの2フェーズ、Runは6種のworkloadから成る

work load	type	insert	read	update	scan
A	Update heavy		50%	50%	
B	Read mostly		95%	5%	
C	Read only		100%		
D	Read latest	5%	95%		
E	Short ranges	5%			95%
F	Read-modify-write		50%	50% ※read-modify	

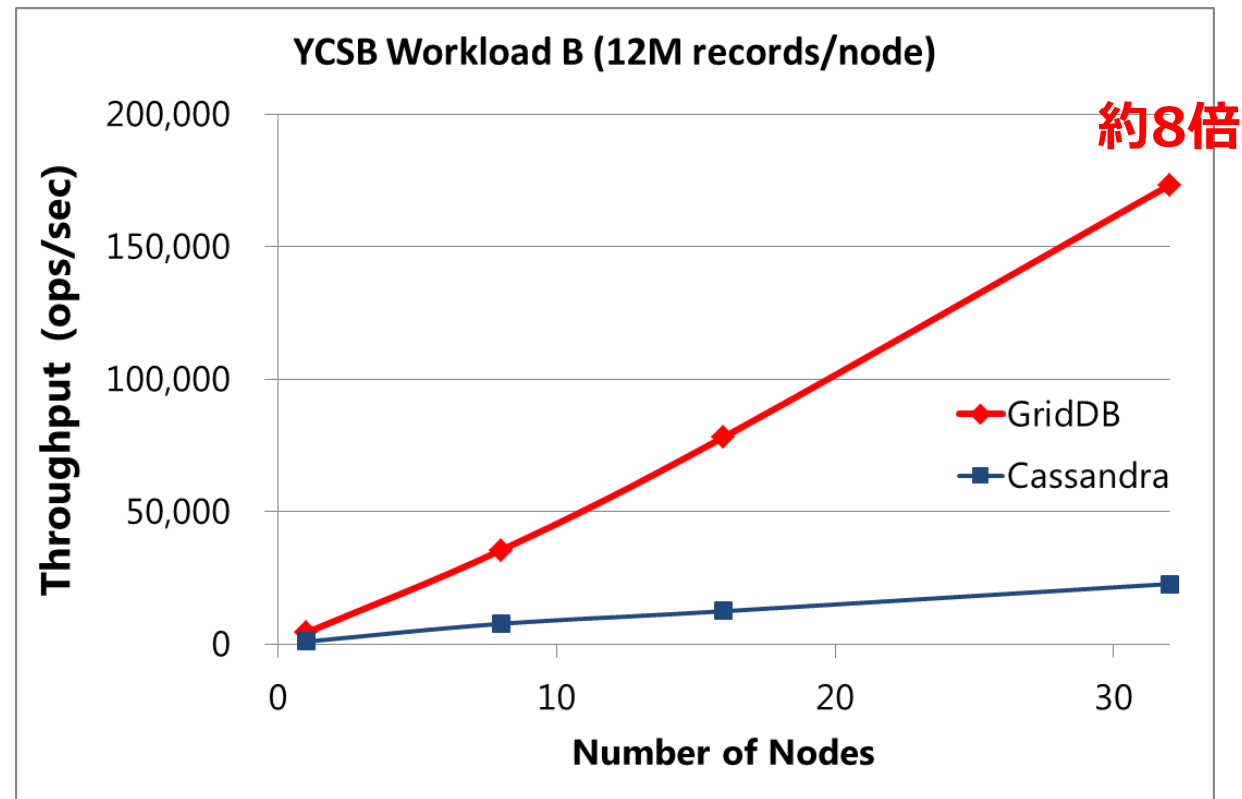
# Cassandraとの性能比較 (YCSB)

高性能を売りにするCassandraと比較しても、GridDBの方が圧倒的に高性能



Read 50% + Write 50%

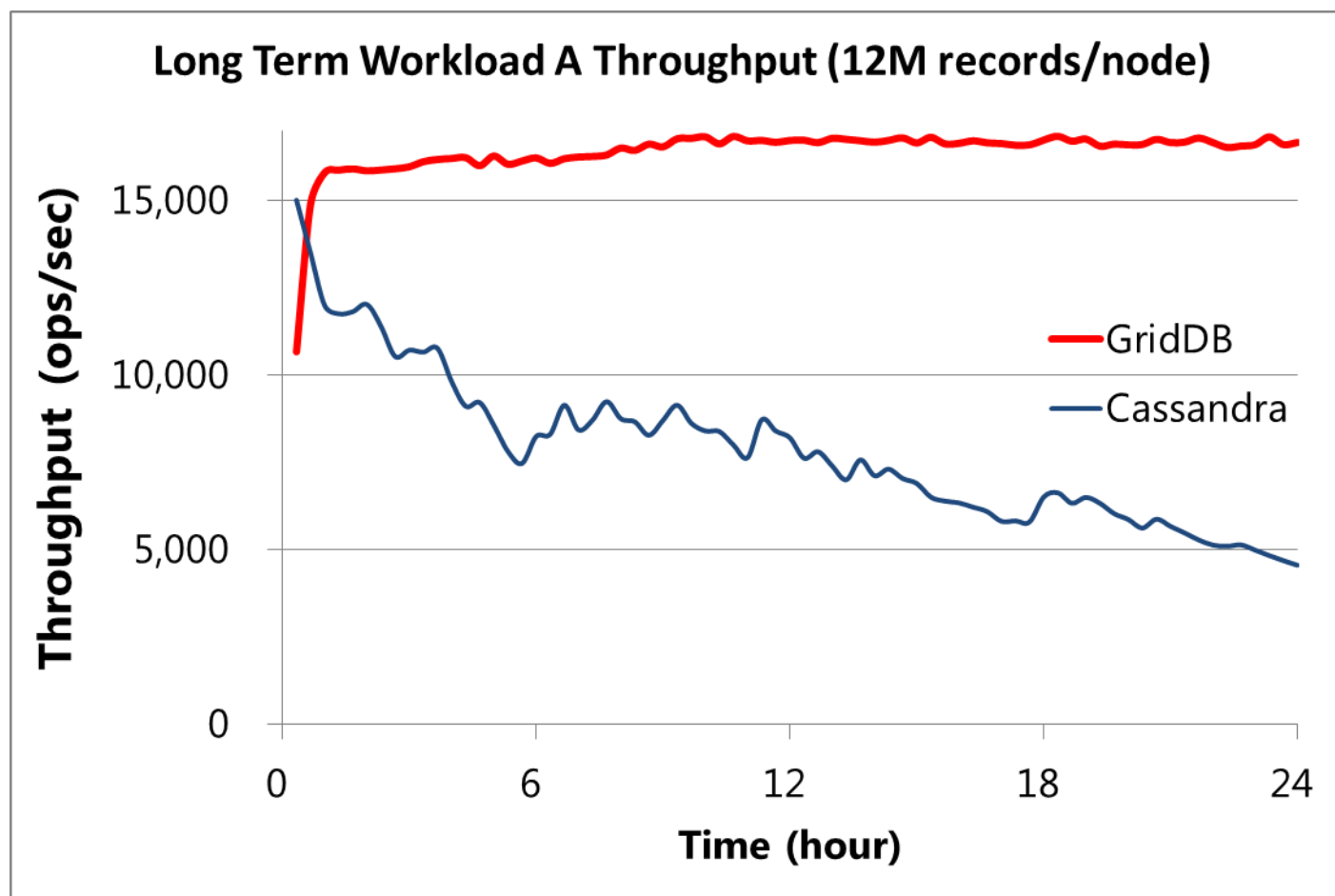
※フィックスターズ社によるYCSBベンチマーク結果



Read 95% + Write 5%

# Cassandraとの性能比較 (YCSB)

長時間実行してもGridDBは性能劣化が少ない



※フィックスターズ社によるYCSBベンチマーク結果

# GridDB適用事例

## エネルギーIoT

- **フランス リヨン 太陽光発電 監視・診断システム**

- 発電量の遠隔監視、発電パネルの性能劣化を診断

- **石巻スマートコミュニティ プロジェクト**

- 地域全体のエネルギーのメータ情報の収集、蓄積、分析

- **電力会社 低圧託送業務システム**

- スマートメータから収集される電力使用量を集計し、需要量と発電量のバランスを調整

## ビル・施設IoT

- **クラウドBEMS**

- ビルに設置された各種メータの情報の収集、蓄積、分析

高い信頼性・可用性が求められる  
システムで使われている

# 適用事例：電力会社

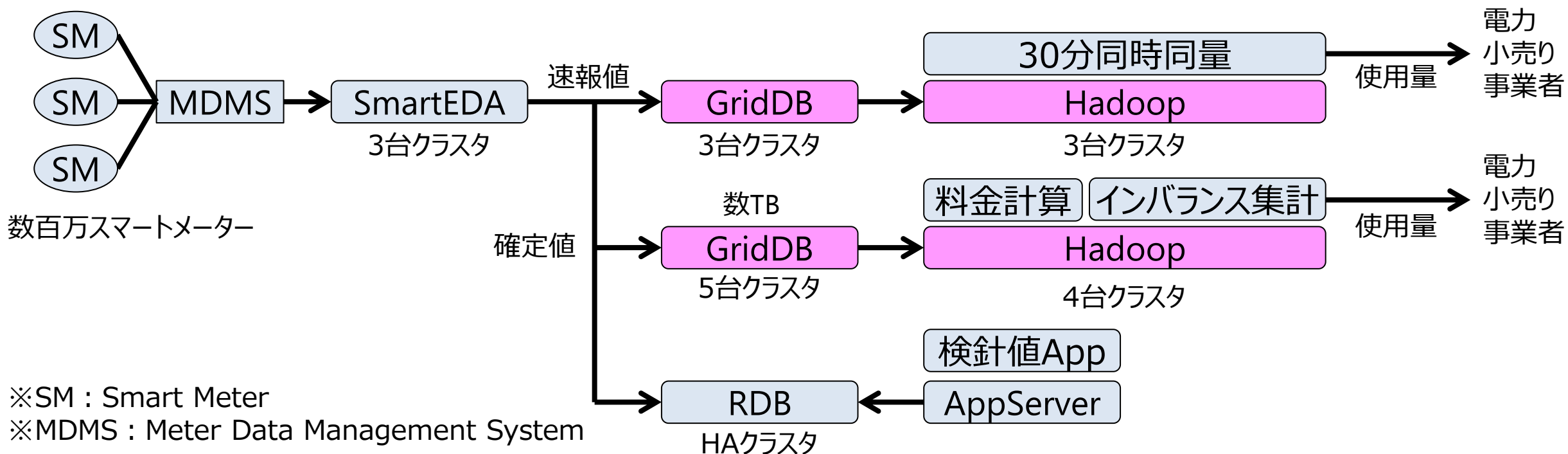
- 電力会社が電力小売り事業者に対し、電力送配電網の使用料を請求するシステム
- 電力の自由化に伴い、多数の電力小売り事業者が参入
- 数千契約 → 数百万契約
- ビッグデータ技術を適用





# 適用事例：電力会社（システム構成）

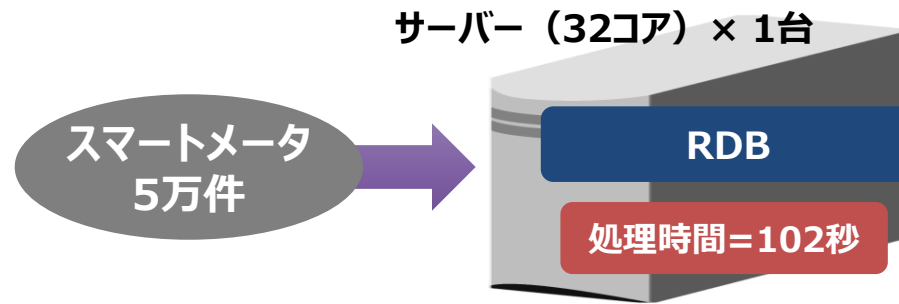
- 数百万台のスマートメータから30分おきに送られてくるメータデータ3ヶ月分をGridDBに蓄積（データサイズ：数百億レコード、数TB）
- HadoopのMapReduceを使って使用量を計算
- 2016年4月の運用開始以来、安定稼働



# 適用事例：電力会社

RDBを使った従来システムに比べ、GridDBとHadoopを使った新システムは、処理性能が35倍に

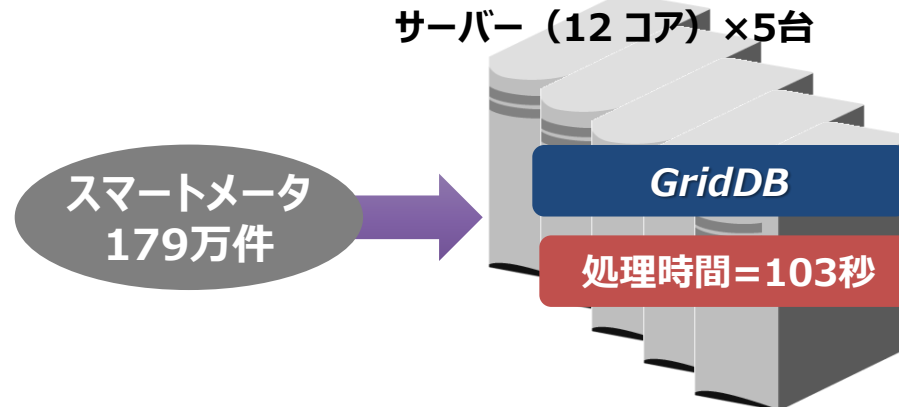
## 旧システム



1万メータあたり 20.4秒

処理能力 35倍

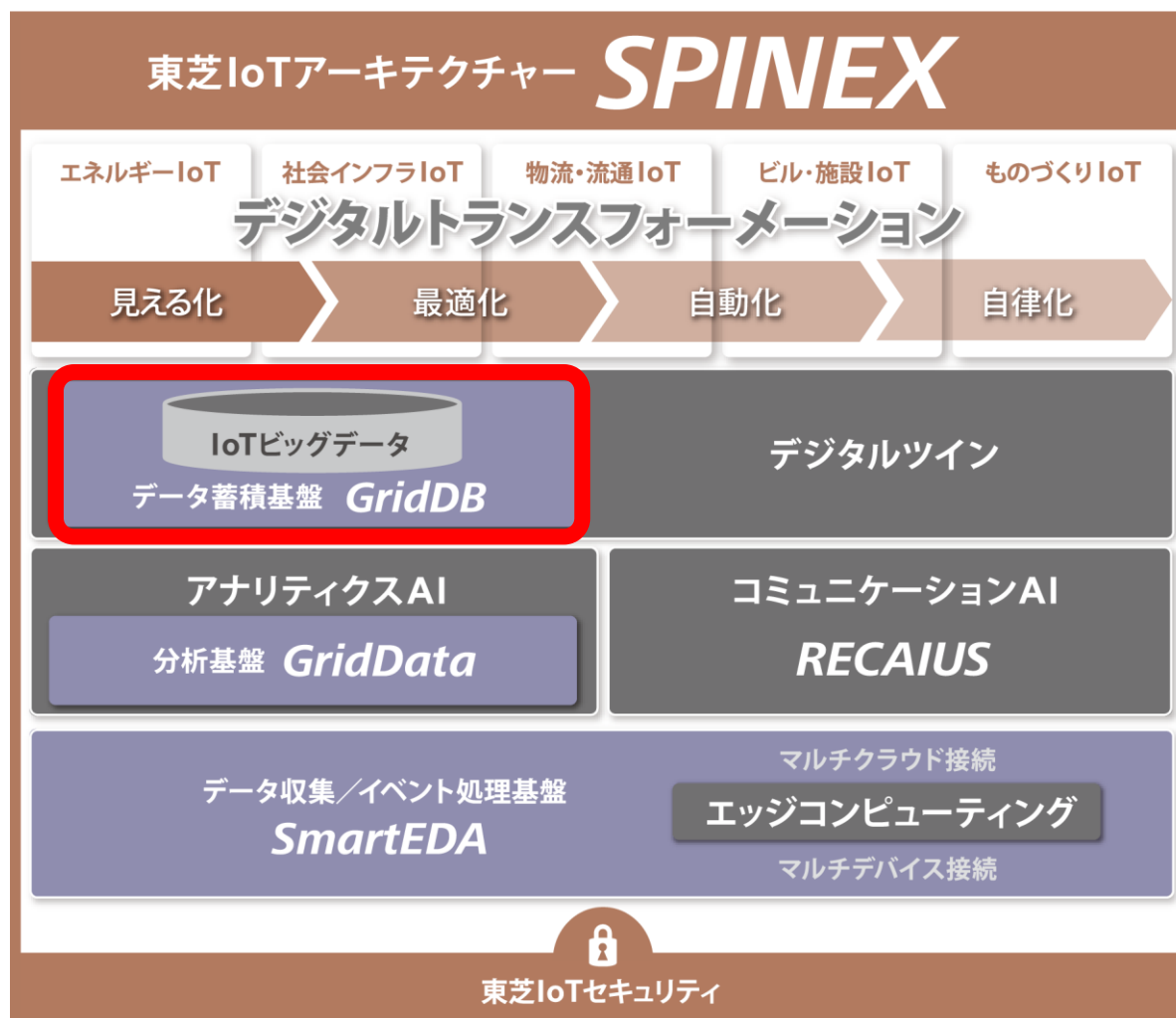
## 新システム



1万メータあたり 0.57秒

# 適用事例：東芝のIoTアーキテクチャーSPINEX

SPINEX のコアコンポーネントとして使われています。



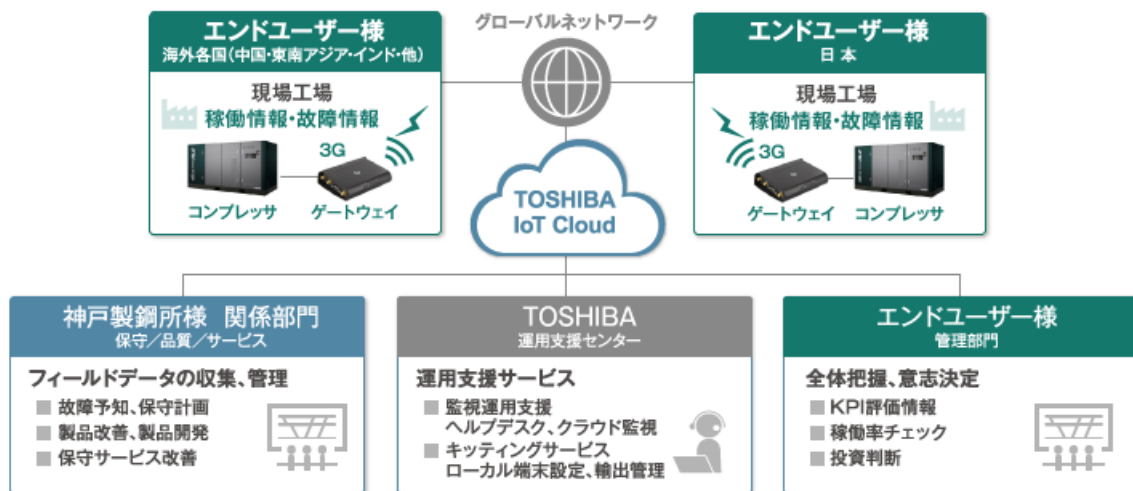
# 適用事例：見える化・遠隔監視サービス「IoTスタンダードパック」

## フィールドIoT

### 神戸製鋼所様 「コンプレッサM2Mクラウドサービス」

図1 コンプレッサM2Mクラウドサービスの概要

世界各地で稼働する汎用コンプレッサからデータを収集・蓄積し、機器の状態を見える化することで、安定稼働に貢献しています。



KPI : Key Performance Indicators (重要業績評価指標)

コンプレッサは、株式会社神戸製鋼所様の製品「Emeraude(エメロード)-ALE」の写真です  
ゲートウェイは、東芝のIoTスタンダードパックにラインナップしたゲートウェイ製品の写真です

<http://www.toshiba.co.jp/cl/articles/tsoul/21/004.htm>

### 東芝機械様 「IoT+mプラットフォーム」

#### お客さまインタビュー

#### 東芝グループのIoTへの取り組み

#### モノづくりの現場を支える東芝機械の「IoT+mプラットフォーム」

カンパニー： 東芝機械株式会社 × ソリューション： 「IoTスタンダードパック」

※「IoT+mプラットフォーム」からのリンクは、東芝機械株式会社様のウェブサイトが別ウィンドウで開きます。

東芝機械は、2016年11月に製造設備向けのIoT活用ソリューションである「IoT+mプラットフォーム」を発表。「集める・分析する・つながる」をキーワードに、モノづくりに携わるお客さまが使用する製造設備の付加価値向上と生産性向上を目指し、本ソリューションの基本機能のひとつとして東芝の「IoTスタンダードパック」を導入した。

東芝グループのIoTへの取り組みについて、東芝機械の事例をもとに紹介していく。



#### Before

東芝機械は2010年頃から製造設備のリモートメンテナンス、遠隔監視、省エネなどの分野で独自にIoT活用に取り組み、工場内の効率化に貢献。並行して、高精度な機械の予防保全や、生産性向上につながる新たなIoT活用ソリューションを模索していた。

#### After

「IoT+mプラットフォーム」に、エッジからクラウドまでのトータルソリューションとして提供する、東芝の「IoTスタンダードパック」を採用。高速処理と高い信頼性を両立するデータベース「GridDB」などのアドバンテージや、東芝のモノづくりの知識・知見を結集したIoT活用ソリューションを、今後、お客さまへ積極的に提案していく。

<http://www.toshiba.co.jp/cl/case/case2017/tsm.htm>

# 適用事例：次世代ものづくりソリューション「Meisterシリーズ」

## ものづくりIoT

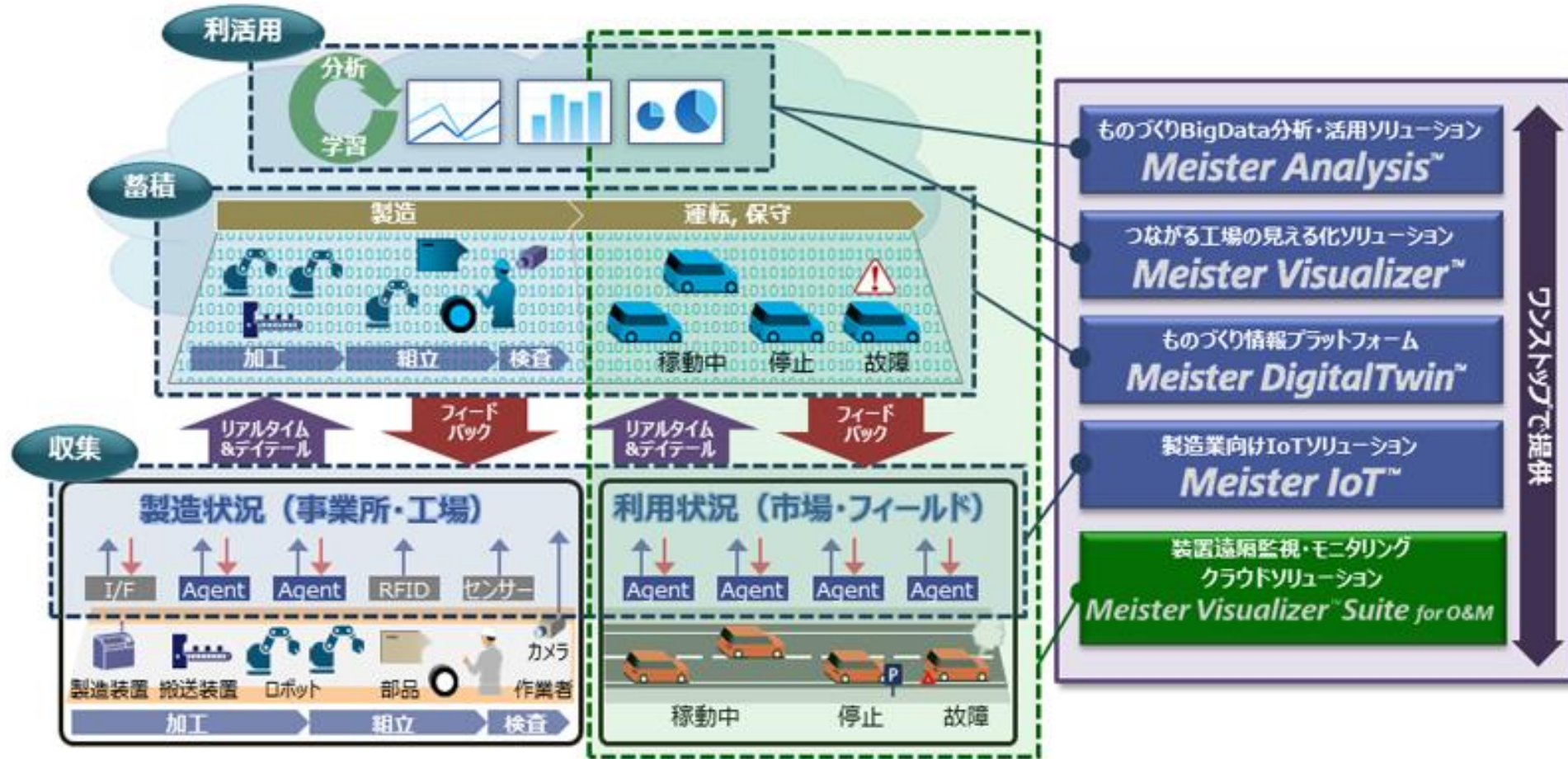


図1：次世代ものづくりソリューション『Meisterシリーズ』概要図

<http://www.toshiba.co.jp/cl/news/news20170308.htm>

# GridDBのラインアップ

## ハイパフォーマンスでかつスケラビリティとリライアビリティを備えたオープンソース NoSQL型データベース



分散データベース関連のエンジニア、アーキテクトや研究者向け  
オープンソースコミュニティで使用するためにAGPLv3 ライセンスの下で利用  
(ソフトウェアサポートなし)

## ハイパフォーマンスでかつスケラビリティとリライアビリティを備えた商用NoSQL型データベース



ミッションクリティカルなビッグデータのアプリケーションを対象に、それに必要な  
ツールや機能およびソフトウェアサポートを提供する商用ライセンス製品

## NoSQLの強みを生かしながらSQLサポートなどRDBの使い勝手を加えた商用NewSQL型データベース



ミッションクリティカルなビッグデータのアプリケーションを対象に、それに必要な  
ツールや機能およびソフトウェアサポートを提供する商用ライセンス製品

## 超高速ベクトル照合機能を備えたビッグデータ解析・メディア解析のためのベクトル高速照合データベース



ベクトル照合技術+分散データベース技術により、10億超規模のベクトル  
高速照合に対応した商用ライセンス製品

# OSSサイト

- **GitHub上にNoSQL機能をソース公開 (2016/2/25)**

- [https://github.com/griddb/griddb\\_nosql/](https://github.com/griddb/griddb_nosql/)

- **目的**

- ビッグデータ技術の普及促進
    - 多くの人に知ってもらいたい、使ってもらいたい。
    - いろんなニーズをつかみたい。
  - 他のオープンソースソフトウェア、システムとの連携強化

- **各種クライアント、コネクタを公開中**



The screenshot shows the GitHub profile for GridDB, a high-performance, high-scalability, and high-reliability database for big data. The profile includes the GridDB logo, a description, location (Japan), website (https://griddb.net), and contact email (contact@griddb.org). Below the profile, there are tabs for 'Repositories' and 'People' (5). A search bar is present. The main content area lists several repositories:

- griddb\_nosql**: high performance, high scalability and high reliability database for big data. C++ language, 281 stars, 24 forks, updated 17 days ago.
- griddb\_kairosdb**: GridDB connector for KairosDB. Java language, updated 21 days ago.
- griddb\_ycsb**: GridDB connector for YCSB. Java language, updated on Oct 28 2016.
- griddb\_hadoop\_mapreduce**: GridDB connector for Hadoop MapReduce. Java language, updated on Aug 3 2016.

# デベロッパーズサイト

- **アプリケーション開発者向けのサイト**

<https://griddb.net/>

- **コミュニケーションの場(フォーラム)を提供**

- **様々なコンテンツを公開**

- ホワイトペーパー
- マニュアル
- サンプルコード

など

Japanese | English [DOWNLOAD](#)

**GridDB** Developers Documentation Community Blog Forum FAQ Resources

NoSQL Database Architecture Comparison Webinar Coming Sept. 6th, 2017 @ 10am PDT

## Is your IoT data getting too BIG to manage?

GridDB is an In-Memory NoSQL Database for highly scalable IoT applications

Learn how GridDB can help you scale

[Yes, tell me more!](#)

### Optimized for IoT

ACID-compliance is guaranteed at the **container level** and time-series functionalities such as term release, data aggregation and sampling are supported

### High Performance

An in-memory data architecture, along with superb parallel processing and minimal overhead, grants excellent performance for various types of data structures

### High Scalability

Extremely easy to scale out/down in response to changing capacity and performance

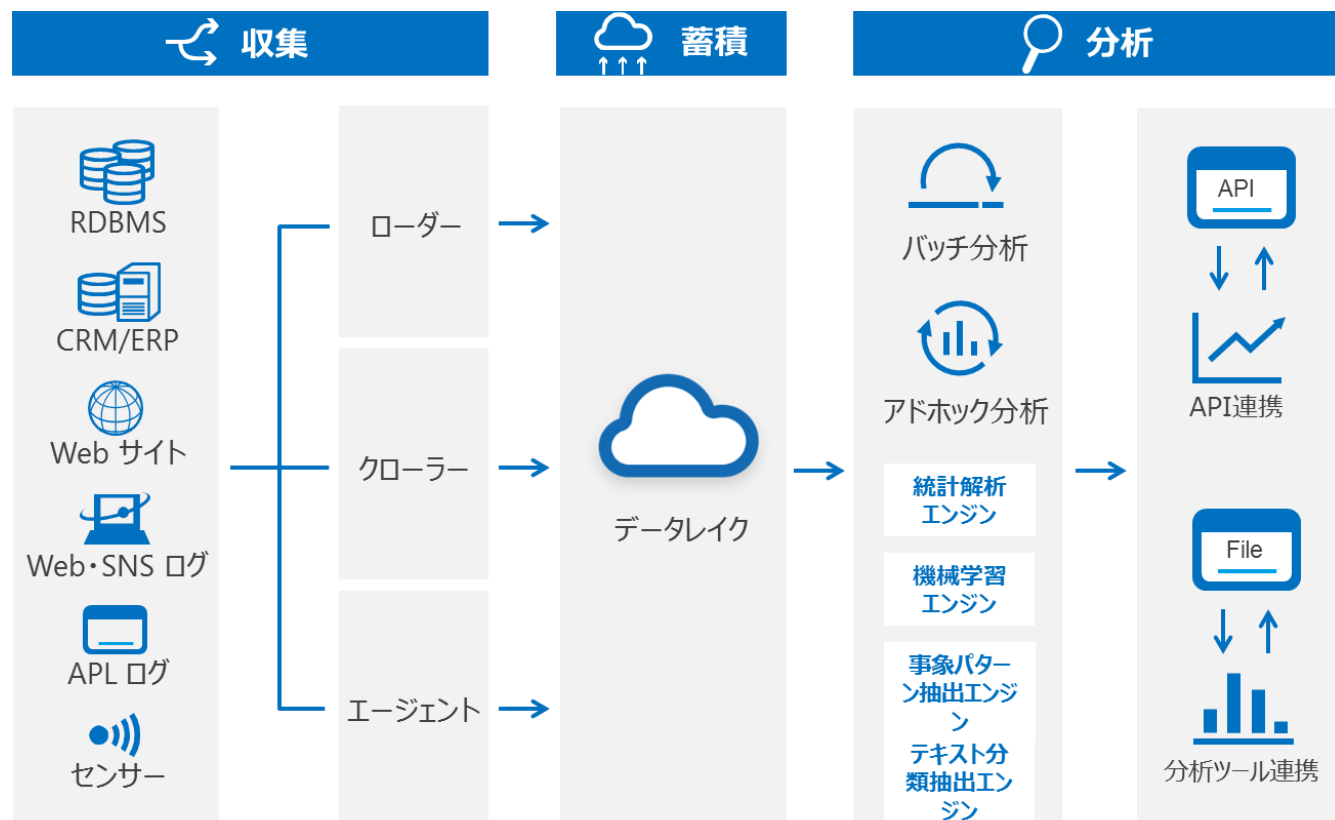
### High Reliability

Non-stop operations: even under inevitable node failures

[Learn More](#)



## ビッグデータの収集・蓄積・分析（AI / 機械学習）をクラウドで簡単、かつ賢く実現



- クラウドだから初期コストを抑えて、気軽にスタート
- データコレクターを活用して効率よく収集
- 機械学習ライブラリを使用して賢く分析
- インタラクティブな環境で、分析の試行錯誤も簡単に

# AWS Marketplaceで、すぐにGridDBを使用可能

**GridDB Community Edition (CE)**  
Sold by: Toshiba Corporation

GridDB Community Edition (CE) is an open source In-Memory NoSQL database best suited for mission critical IoT applications. Toshiba's GridDB offers high performance, high scalability and high reliability that are essential for IoT systems. GridDB supports Time Series data and numerous functions that operate on data associated with time-stamps. In-Memory architecture of GridDB lets primary data to be stored and processed in memory while simultaneously offering disk persistence (SSDs and HDDs) and thus significantly enhancing the performance of the entire system. GridDB's massive scale-out... [Read more](#)

**Customer Rating** ★★★★★ (0 Customer Reviews)

**Latest Version** 1.1

**Operating System** Linux/Unix, CentOS 7.2.1511

**Delivery Method** 64-bit Amazon Machine Image (AMI) ([Read more](#))

**Support** [See details below](#)

**AWS Services Required** Amazon EC2, Amazon EBS

**Highlights**

- High Performance - GridDB Memory first, Storage second structure is a hybrid composition of In-Memory and Disk architecture designed for maximum performance.
- High Scalability - GridDB maintains excellent performance by adopting scale-out architecture which scales linearly and horizontally on commodity hardware.
- High Reliability - Hybrid cluster management and high-fault

**Continue** You will have an opportunity to review your order before launching or being charged.

**Pricing Details**

For Region: US East (N. Virginia)

**Hourly Fees**  
Total hourly fees will vary by instance type and EC2 region.

EC2 Instance Type	Software	EC2	Total
r3.large	\$0.00/hr	\$0.166/hr	\$0.166/hr
r3.xlarge	\$0.00/hr	\$0.333/hr	\$0.333/hr
r3.2xlarge	\$0.00/hr	\$0.665/hr	\$0.665/hr
r3.4xlarge	\$0.00/hr	\$1.33/hr	\$1.33/hr
r3.8xlarge	\$0.00/hr	\$2.66/hr	\$2.66/hr
x1.32xlarge	\$0.00/hr	\$13.338/hr	\$13.338/hr

<https://aws.amazon.com/marketplace/pp/B01N5ASG2S>

Marketplace : パブリックIaaSの上で、各社のソフトウェアが時間単位で使えるようになっている

## まとめ

- 代表的なNoSQLとGridDBをアーキテクチャとメカニズムの観点で比較しました。
- GridDBはビッグデータ×IoT×AI向けのスケールアウト型データベースです。
- OSSサイト、デベロッパーズサイト、AWS Marketplace上のサービス、などを公開・提供しています。

GridDBのオープンソース版がありますので、是非とも使ってみてください。

- 本資料に掲載の製品名、サービス名には、各社の登録商標または商標が含まれています。

# GridDBに関する情報

- **GridDB 製品情報**
  - <http://www.toshiba.co.jp/cl/pro/bigdatapf/>
- **GridDB デベロッパーズサイト**
  - <https://griddb.net/>
- **GridDB OSSサイト**
  - [https://github.com/griddb/griddb\\_nosql/](https://github.com/griddb/griddb_nosql/)
- **OSSを利用したビッグデータ分析環境 *GridData Analytics Cloud***
  - <https://www.griddata-analytics.net/>
- **AWS Marketplace: GridDB Community Edition (CE)**
  - <https://aws.amazon.com/marketplace/pp/B01N5ASG2S>
- **AWS Marketplace: GridDB Standard Edition (SE)**
  - <https://aws.amazon.com/marketplace/pp/B01N9QMCMF/>
- **Twitter: GridDB Community**
  - <http://twitter.com/GridDBCommunity/>
- **Facebook: GridDB Community**
  - <http://fb.me/griddbcommunity/>



# デベロッパーズサイト(<https://griddb.net/>)の主なコンテンツ一覧

## ホワイトペーパー :

- **GridDB®とは**
- **GridDB と Cassandra のパフォーマンスとスケーラビリティ – Microsoft Azure 環境における YCSB パフォーマンス比較**
- **GridDB Reliability and Robustness**

など

## ブログ :

- **GridDB クラスタのハードウェア要件の見積もり**
- **CAP 定理と GridDB**
- **Raspberry Piチュートリアル : KairosDBコネクタを介してGridDBに温度データを送信する**
- **Docker上でGridDBを実行する**
- **IoT産業におけるGridDB導入事例**
- **GridDB Azureクラスタの構築**
- **YCSB向けGridDBコネクタを使ってみよう**

など

- **IoT/ビッグデータ向けデータベース「GridDB」 増え続ける時系列データを高速かつスケーラブルに処理**  
<http://www.toshiba.co.jp/cl/articles/tsoul/22/003.htm>
- **GridDB導入事例 電力小売自由化に対応した大規模なスマートメーターデータの高速処理**  
<http://www.toshiba.co.jp/cl/articles/tsoul/22/004.htm>
- **日本発、デジタルトランスフォーメーションを支える 東芝のIoTアーキテクチャー「SPINEX」のすべて**  
<http://www.toshiba.co.jp/cl/articles/tsoul/21/001.htm>
- **デジタルトランスフォーメーションの実現へ 「SPINEX」の設計思想を現場で具現化するテクノロジー**  
<http://www.toshiba.co.jp/cl/articles/tsoul/22/001.htm>
- **SPINEXの構成要素を活用したソリューション お客様のビジネス革新を加速する「IoTスタンダードパック」**  
<http://www.toshiba.co.jp/cl/articles/tsoul/21/002.htm>
- **価値創出を加速する、IoTスタンダードパック導入事例 神戸製鋼所様「コンプレッサM2Mクラウドサービス」**  
<http://www.toshiba.co.jp/cl/articles/tsoul/21/004.htm>

---

**TOSHIBA**  
**Leading Innovation >>>**