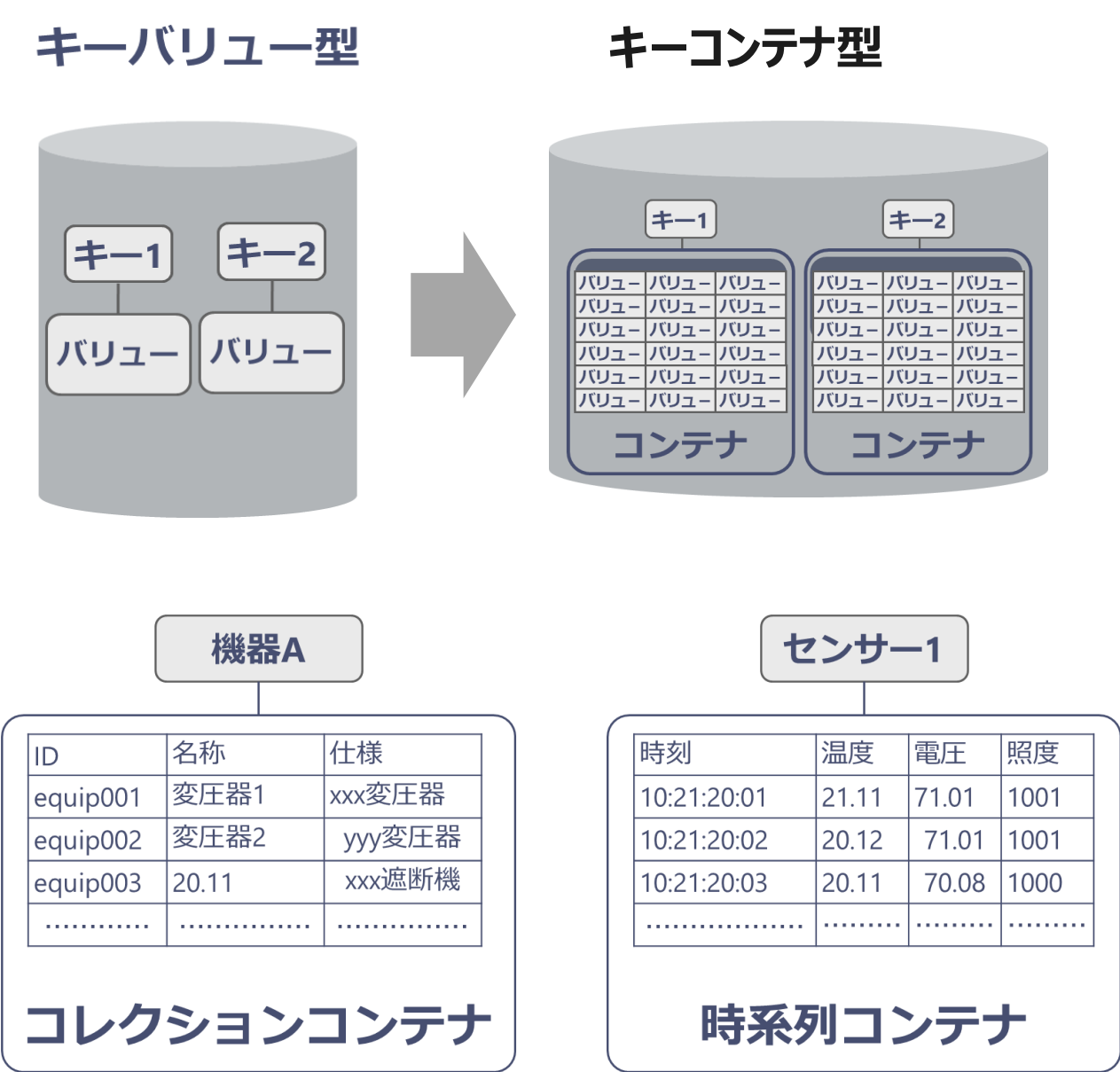
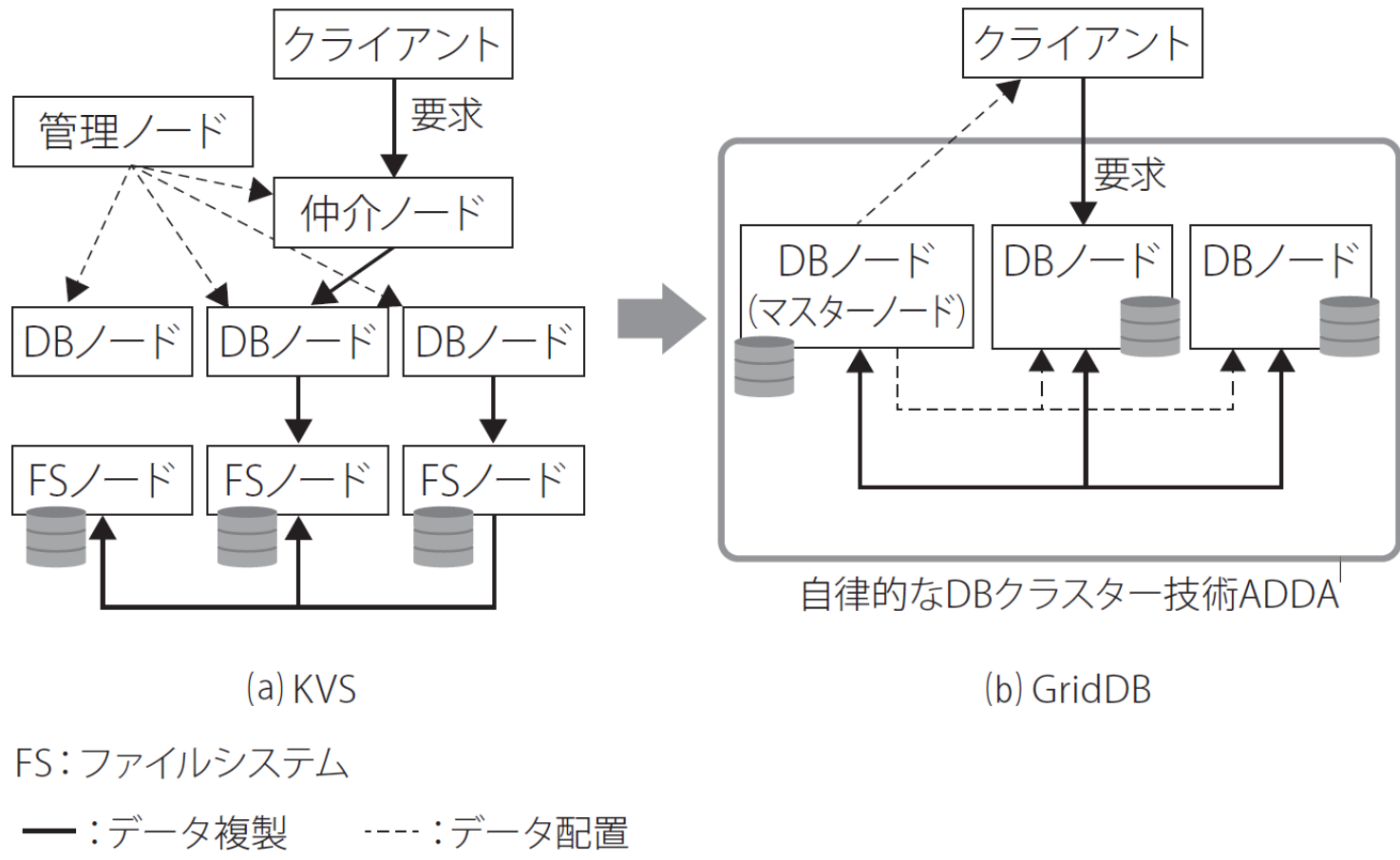
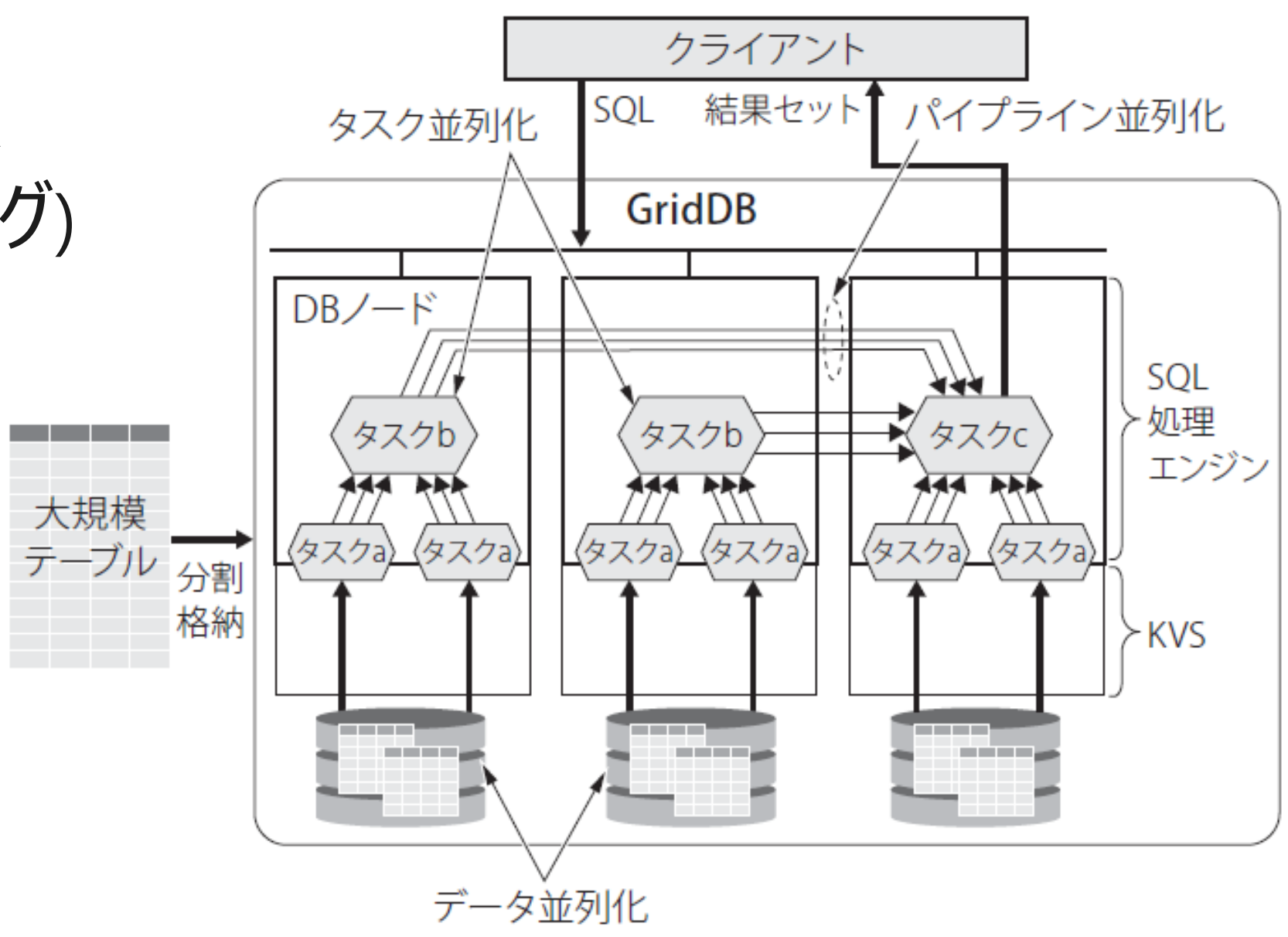


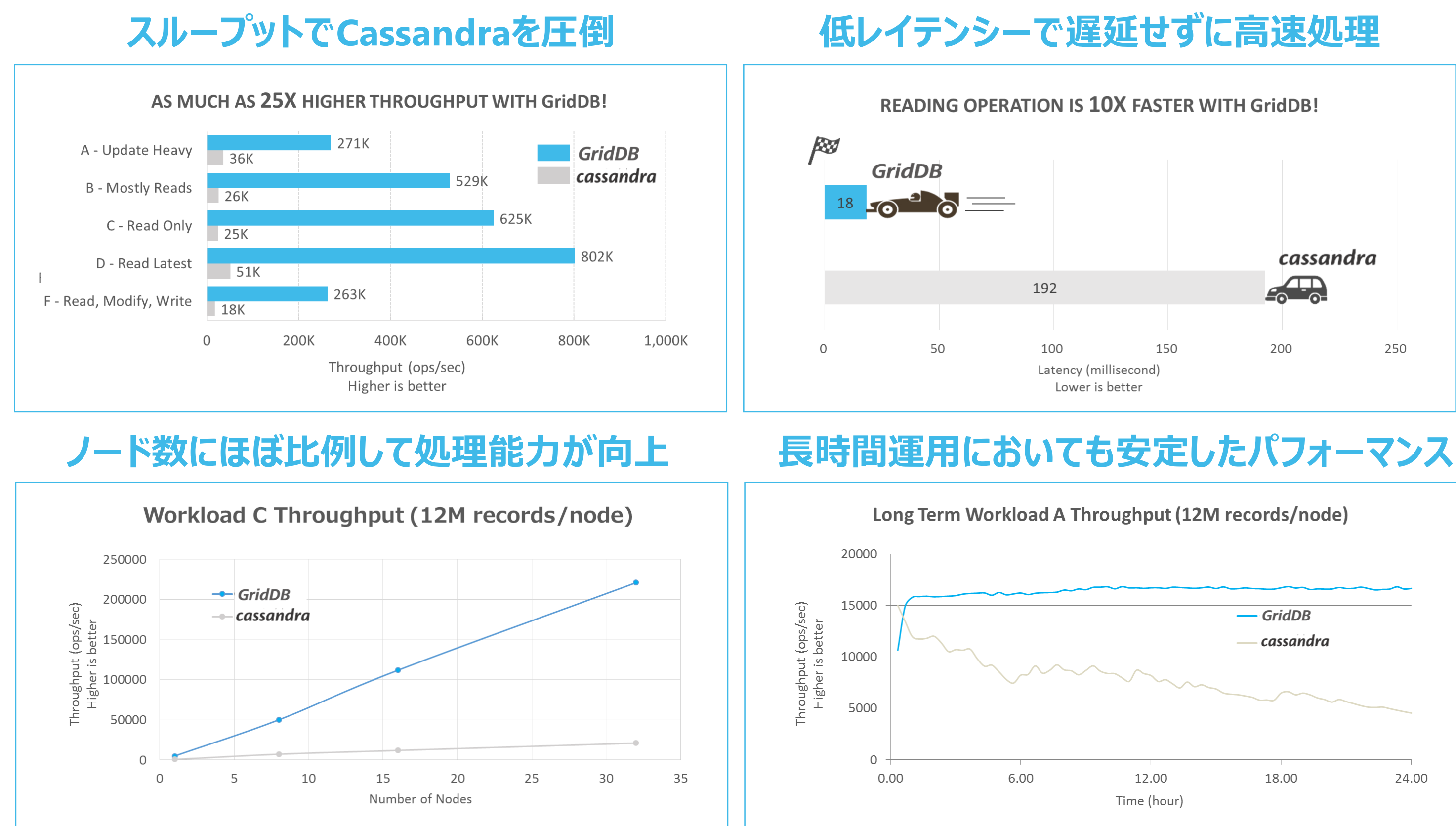
もうSQLとNoSQLを選ぶ必要はない！？

～ 両者を備えたスケールアウトデータベース **GridDB** ～

データ登録などには高速な NoSQL (KVS) インターフェース、分析や他システムとの連携には
有用な機能が多い SQLインターフェース を使うことで、双方の長所を有効活用可能

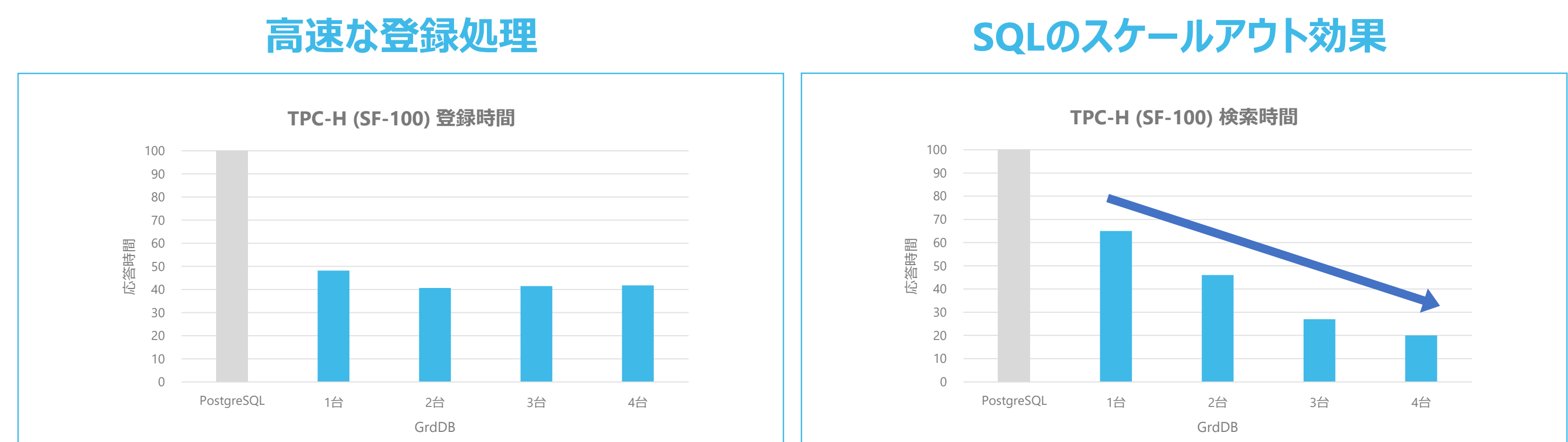
特長		代表的な実現技術	
IoT 指向モデル 高い信頼性と 可用性 高い拡張性 高性能なNewSQL (=NoSQL+SQL)	- データモデルはキー・コンテナ。 コンテナ内でのデータ一貫性を保証 - 時系列データ管理する特別な機能 - 過去データをコールド保存する長期アーカイブ機能	キーコンテナ型データモデル - NoSQL型でよく採用されているキー・バリューを拡張 - 時間順にレコードが格納される時系列コンテナ 時系列レコードを圧縮する機能や期限解放する機能を装備 - 順序に関係無くレコードが格納されるコレクションコンテナ - コンテナ内でのデータ一貫性を保証 長期アーカイブ技術 - 適切に容量を抑えつつ、長期間データを保存効率の良いアーカイブ処理	
	- データの複製をノード間で自動的に実行 - ノードに障害があっても縮退してサービス継続 - 数秒から数十秒の切替時間	自律データ再配置技術 (ADDA : Autonomous Data Distribution Algorithm) - 明示的なマスターノードを必要としない自律制御によるクラスタ管理 - ノード間でデータ配置を自動決定。クライアントとノードの間に位置する管理ノードや仲介ノードを排除 → 通信やデータ変換処理などのコストが削減され、高いパフォーマンスとデータの一貫性を同時に実現 → 単一障害点になりやすい管理ノードがクラスタ内に存在しないために、さらに耐障害性が向上 → ノード過半数を占めたサブクラスタのみがサービス可能となるクォーラムポリシーによりスプリットブレインを完全排除	
	- 少ないサーバ台数で初期投資を抑制 - 負荷や容量の増大に合わせたサーバ増設が可能 - 自律データ再配置により、高い拡張性を実現	デュアルインターフェース - SQL (ANSI-92) インタフェース - コンテナ(テーブル)をRDBのリレーションとみなしてアクセス - 巨大コンテナをノード間に分散 (テーブルパーティショニング) - ジョインなど複数テーブルに対するSQL - JDBC/ODBCドライバー - NoSQL (キー・バリュー型) インターフェイスSQL - 高可用、高スループット指向のKVS - キーコンテナに対するCRUD - Java/C/Python/Rubyドライバー	

NoSQLベンチマーク YSCB (Yahoo! Cloud Serving Benchmark)



出典：株式会社フィクスターズ「GridDBとCassandraのパフォーマンスとスケーラビリティ」（2017年8月23日）

NewSQLベンチマーク TCP-H



- PostgreSQL 9.6
- GridDB AE 4.0
- CPU : 8-core Intel® Xeon® E5-2620 v4 2.10GHz
- Memory : 64GB
- HDD : SAS 12TB
- OS : CentOS 7 with kernel 3.10.0-514.el7.x86_64
- Network : 1Gb Ethernet
- Dataset : TPC-H (SF 100), Q1-Q8

オープンソース・コミュニティ

オープンソースソフトウェアとして公開
<https://github.com/griddb/>

デベロッパーサイトを用意
<https://griddb.net/>

