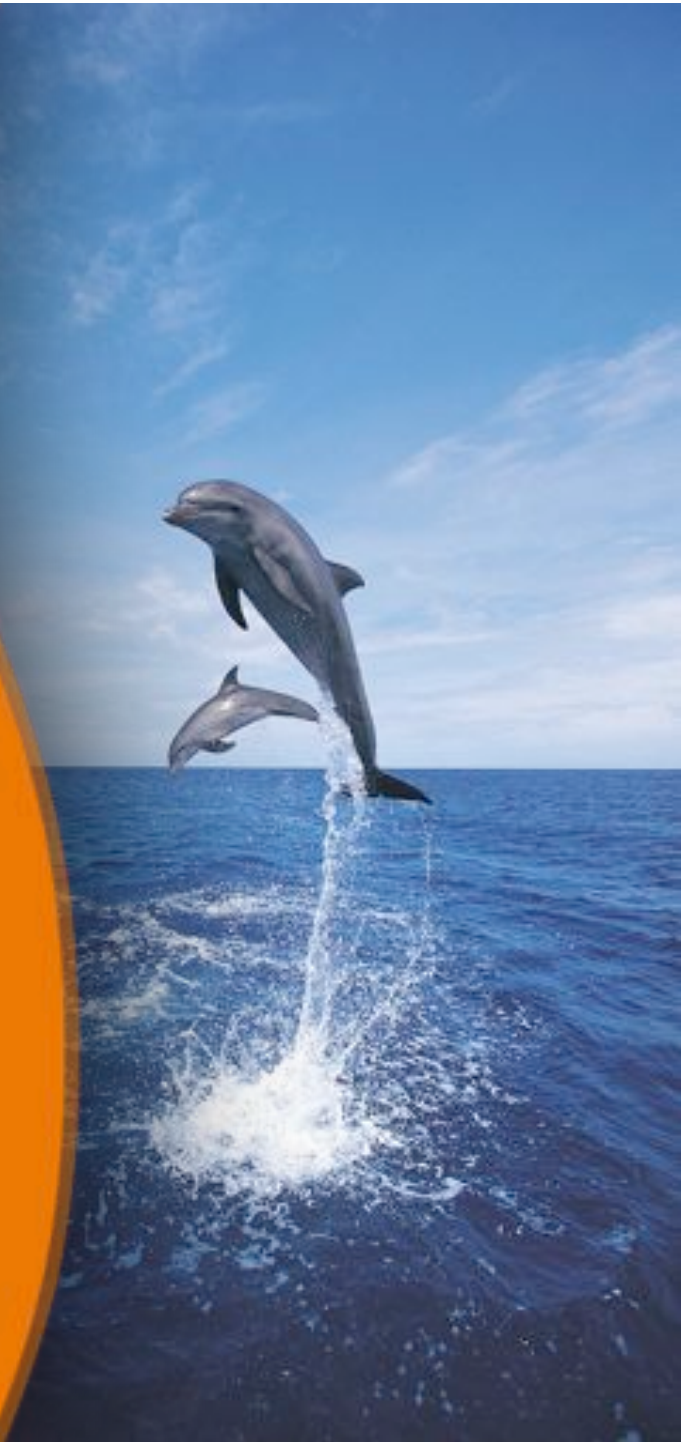




MySQLシステム構成 ～高可用性構成編～

サン・マイクロシステムズ株式会社
MySQLビジネス統括本部
梶山 隆輔



MySQL について

- MySQL は...
 - > LAMP の“M”
(Linux, Apache, MySQL, PHP/Perl/Python)
 - > 業界で最も普及している、急成長中のオープンソースデータベース
 - > 主要なプラットフォームをサポート-
Windows, Linux, Solaris™, Mac OS X
 - > HP, Dell, IBM, Sun, Fujitsuなど
多様なハードウェアに対応
 - > 世界中の大企業が採用



世界で最も普及しているオープンソースデータベース

(旧) MySQL社の企業としての特徴

- ...創業者が **MySQL**を開発
- ...**MySQL**の商標を保持
- ...**MySQL**を開発
- ...**MySQL**の著作権を保持
- ...**MySQL**の商用ライセンスを販売
- ...**MySQL**への技術サポートを提供
- ...**MySQL**のトレーニングを提供
- ...**MySQL**の認定資格を提供
- ...**MySQL**のコンサルティングを提供
- ...バーチャルな組織とパートナーを全世界に展開
- ...Sunとの合併後もオープンソースであることを継続

サンとMySQLが統合



- サンのメリット...
 - > 150億ドル マーケットでの成長
 - > 新規顧客拡大
 - > Web 業界に対する優位な位置づけ
 - > データベースイノベーションの提供
 - > ソフトウェアプラットフォームを完備
- MySQLのメリット
 - > マーケットと収益機会の拡大
 - > グローバルサービスを利用し企業へのアピール度アップ
 - > パフォーマンスと統合性の向上
 - > 成長を加速させるリソースの増加

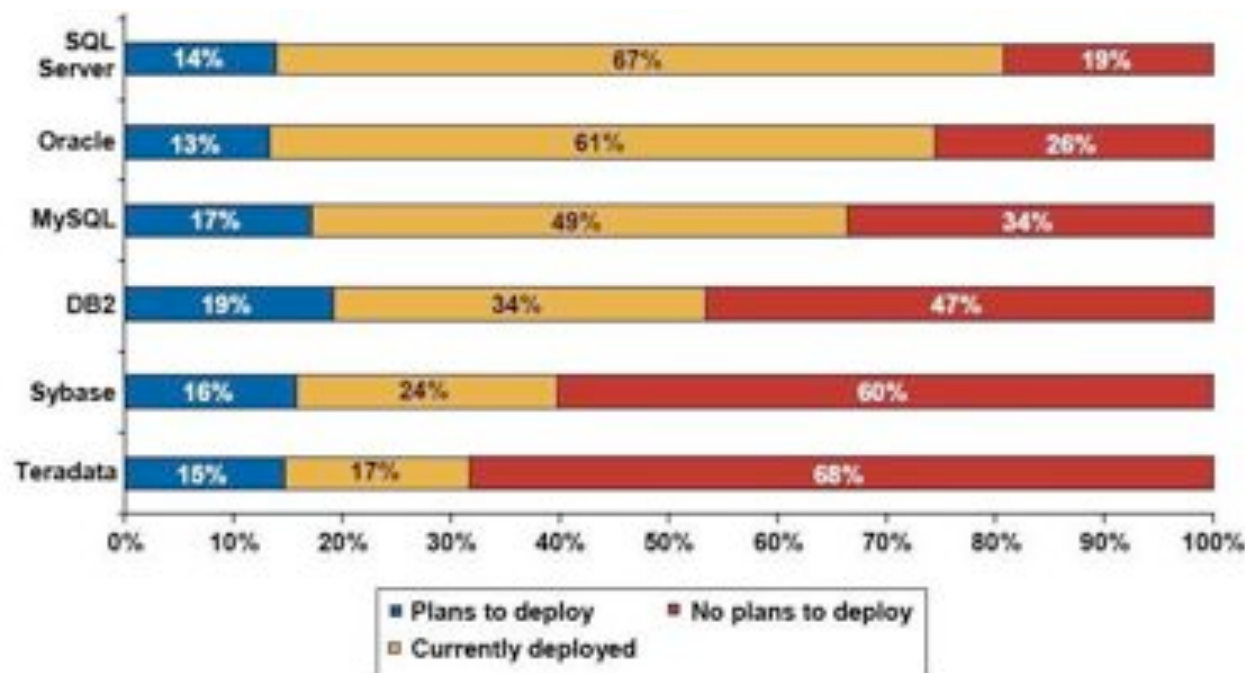


MySQLの業界をリードする企業顧客

<p>Web / Web 2.0</p>	<p>OEM / ISV's</p>	
<p>On Demand, SaaS, Hosting</p>	<p>Telecommunications</p>	<p>Enterprise 2.0</p>

Webにパワーを与えるオープンソース

オープンソース・データベースの採用状況

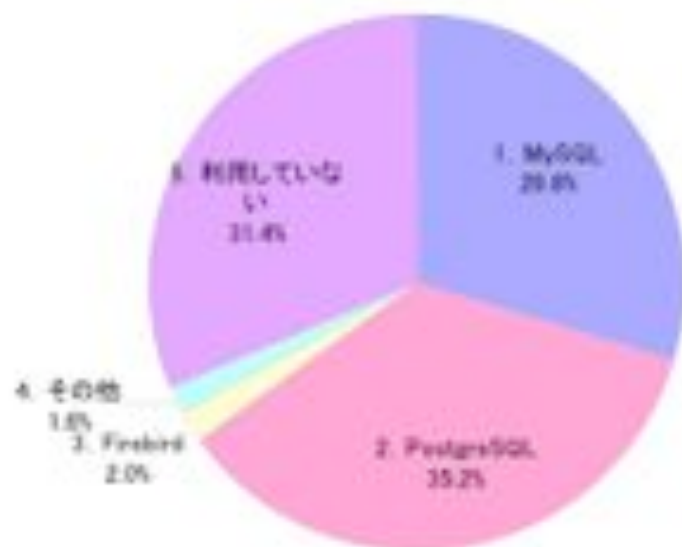


66% の企業が
MySQLを利用中
または利用予定

Gartner

- 2008年までに70%以上のIT部門がオープンソースのデータベースを採用
- 16%が商用データベースからオープンソース・データベースに変更
- 42%が商用データベースからオープンソース・データベースへの変更のために製品調査中

オープンソース・データベースの採用状況(日本)



- 日本ではPostgreSQLがMySQLより普及している
 - > ユーザ会による翻訳などの普及推進活動
 - > 企業ベースでの機能開発

出展：情報処理推進機構

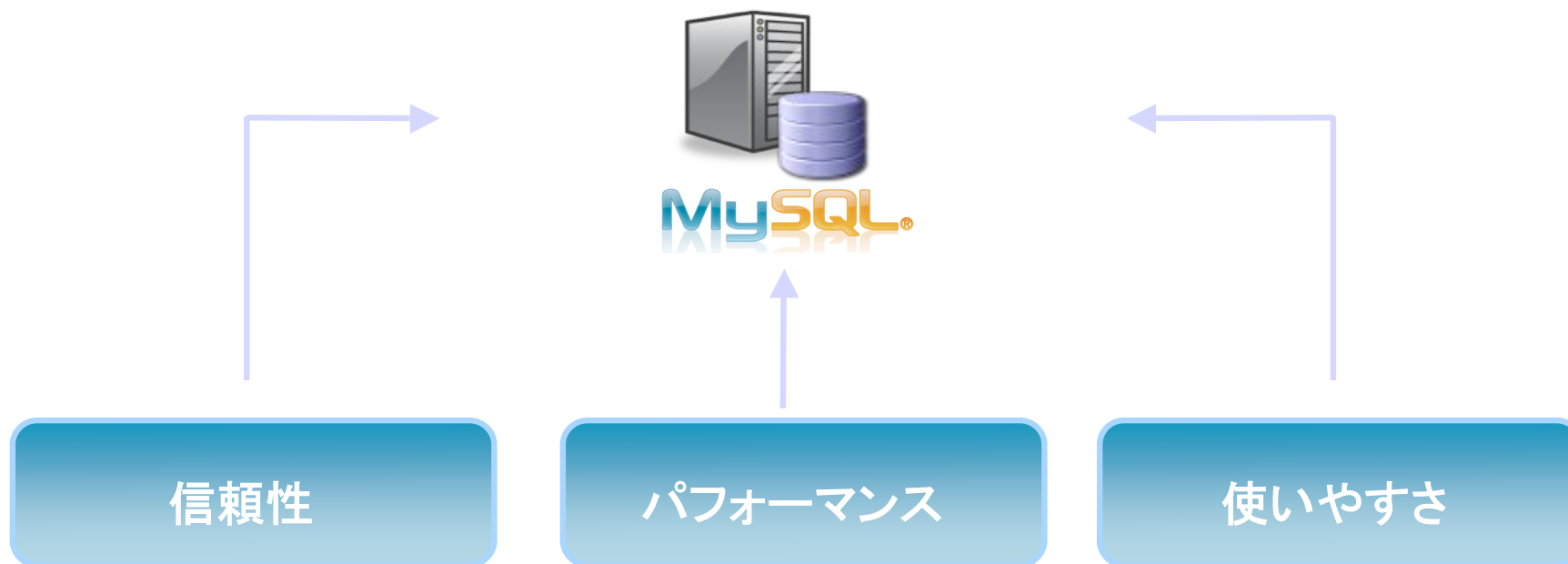
2007年度 オープンソースソフトウェア活用基盤整備事業

「我が国の OSS 活用 IT ソリューション市場の現状と将来展望に関する調査

Sun Microsystems K.K.

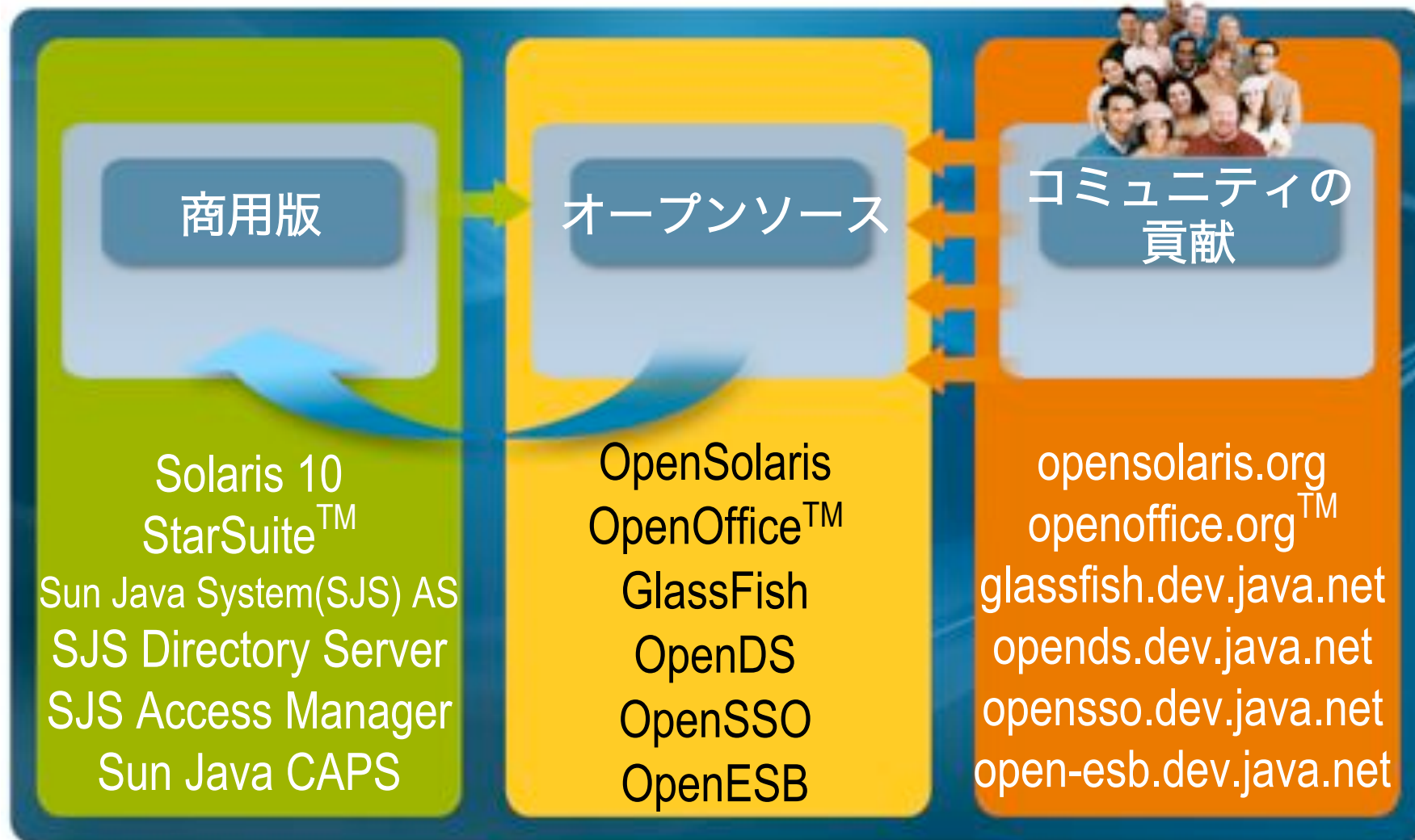
Be the Best Online Database

最高のオンラインデータベース



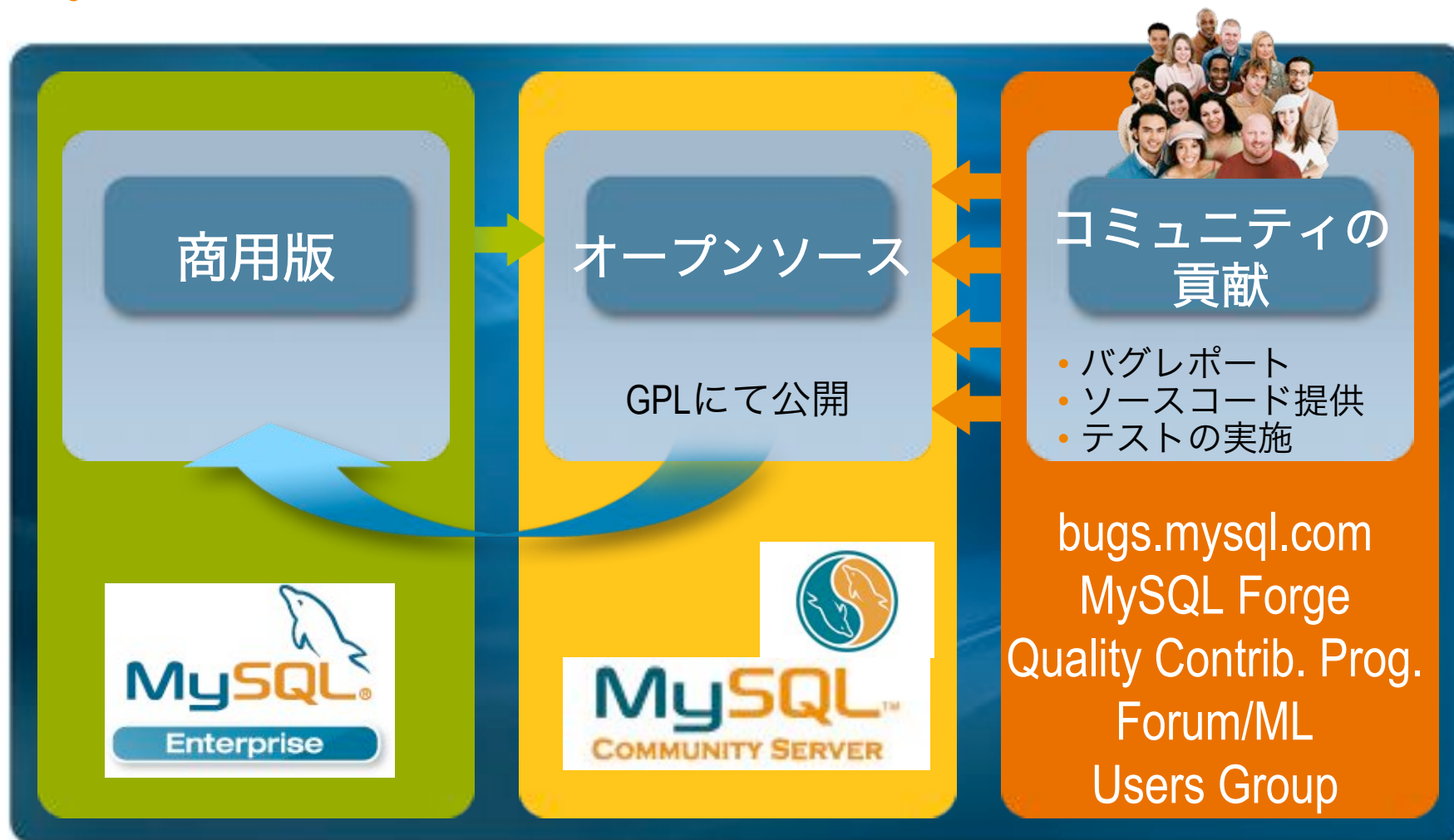
オープンソースと商用配布

サンのソフトウェアビジネス



オープンソースと商用配布

MySQLのビジネスモデル



無償ユーザと有償ユーザ

無償ユーザ

- MySQL Community Server
- MySQL GUI管理
ツール
- MySQL コネクタ
(JDBC, ODBC, etc.)
- ドキュメント
- フォーラム

有償ユーザ

- サブスクリプション:
MySQL Enterprise
MySQL Enterprise Unlimited
- 商用ライセンス(OEM):
組み込みサーバ
サポート
- MySQL Cluster
MySQL Cluster Carrier-Grade
- トレーニング
- コンサルティング &
プロフェッショナル・サービス

データベースインフラ設計

- データベースのインフラ設計で考慮すべき主な項目
 - > 高可用性設計
 - > 拡張方式設計(性能/ストレージ)
 - > セキュリティ設計
 - > バックアップ設計
 - > 監視方式設計
 - > メンテナンス方式設計(データ/ログ/時刻同期)

データベースインフラ設計

- データベースのインフラ設計で考慮すべき主な項目
 - > 高可用性設計
 - > 拡張方式設計(性能/ストレージ)
 - > セキュリティ設計
 - > バックアップ設計
 - > 監視方式設計
 - > メンテナンス方式設計(データ/ログ/時刻同期)

99.999 % の可用性

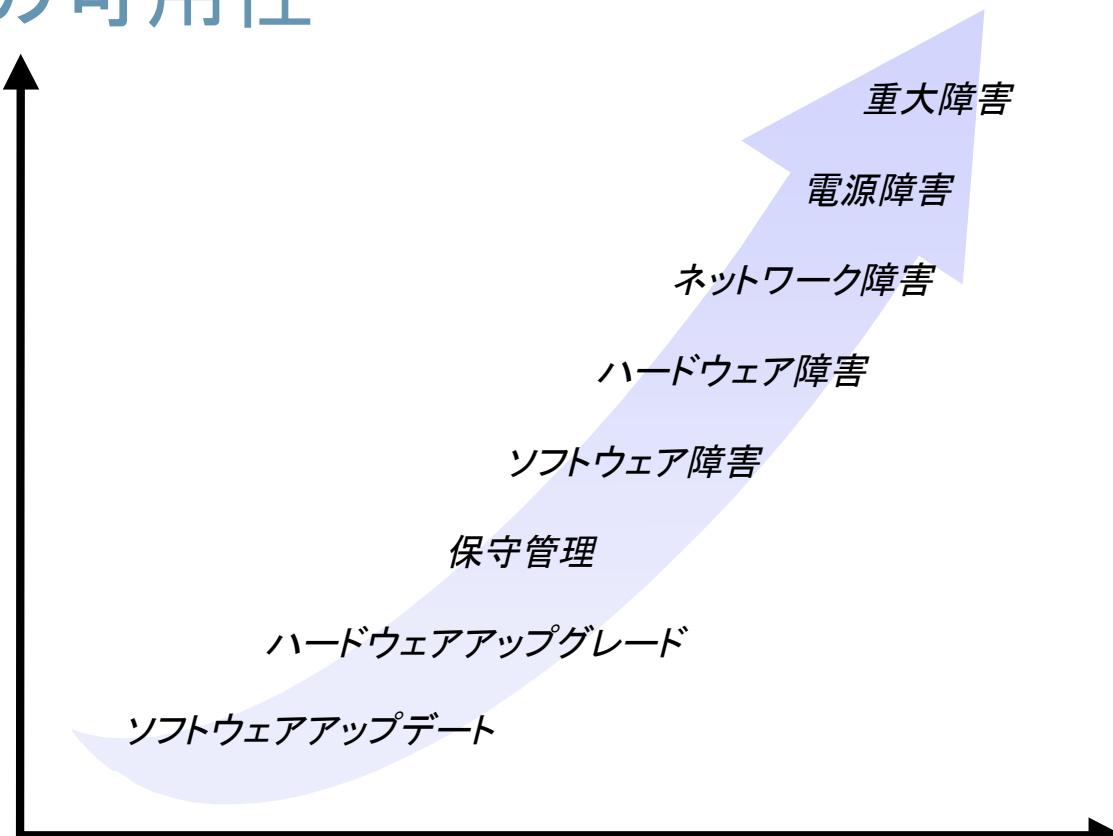
クラスタリング & 地理的冗長性

クラスタリング 技術

レプリケーション 技術

優れた管理性

管理不要



35 日

4 日

8 時間

50 分

5 分

小規模
ビジネス

ISP と
メインストリーム
ビジネス

データ
センター

銀行
医療

テレコム
軍事
防衛

9

9.

9

9

9

%

Sun Microsystems K.K.

MySQLの高可用性ソリューション

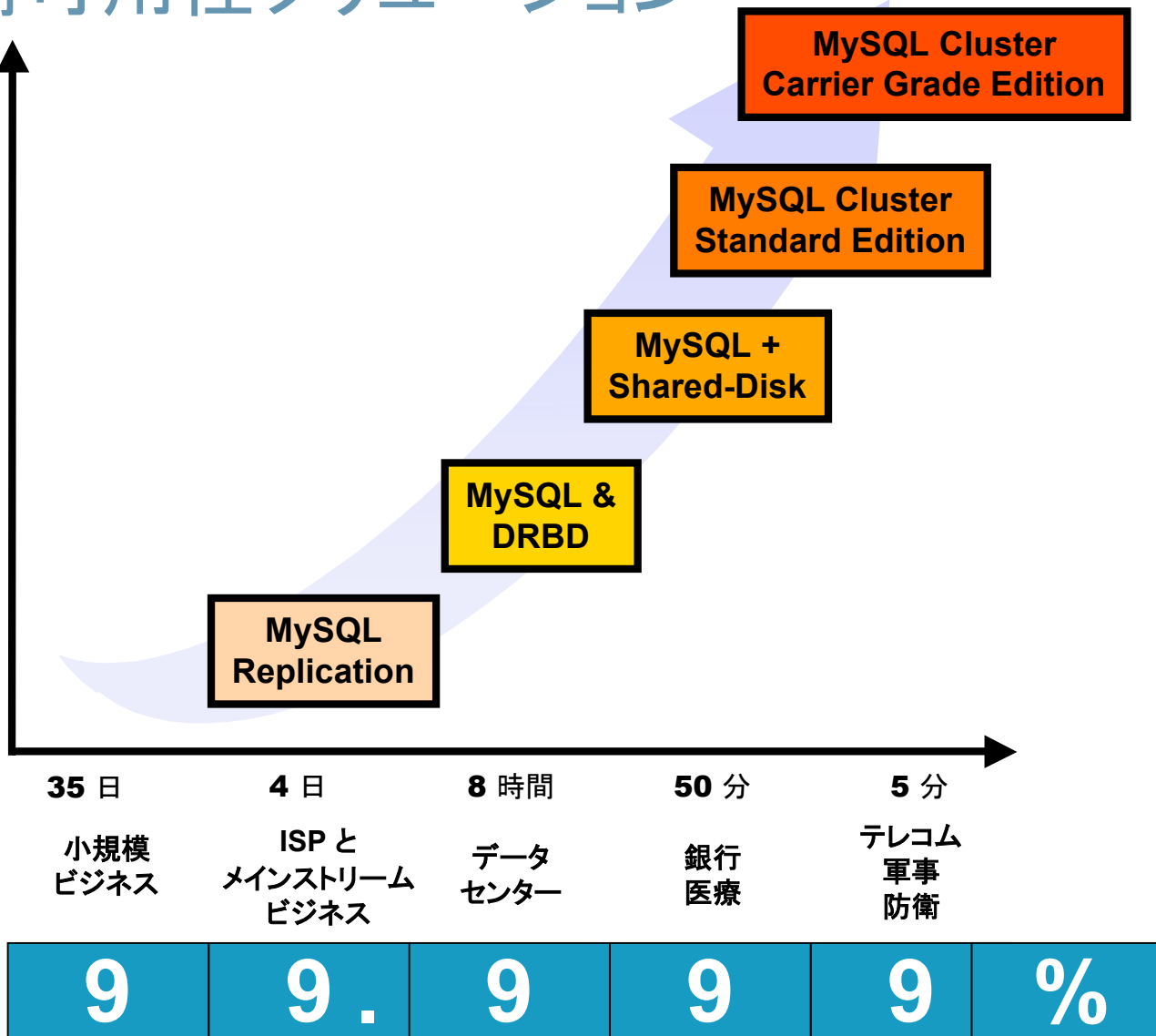
クラスタリング & 地理的冗長性

クラスタリング 技術

レプリケーション 技術

優れた管理性

管理不要



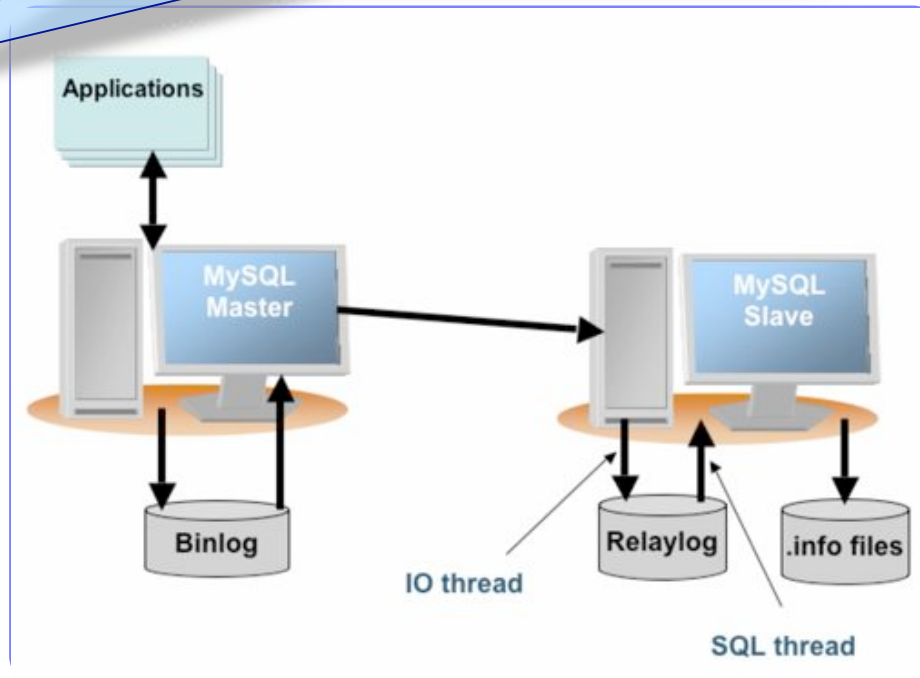
レプリケーション

- MySQLの標準機能
 - シンプルな設定
 - マスター→スレーブ
 - 多数Webでの実績

- 非同期型

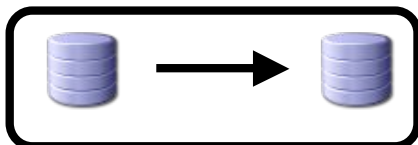
- 特徴
 - 参照性能を向上させる構成
 - バックアップ用途での利用も
 - 基本は一方向でのデータコピーだが、双方向や循環型での利用も可能（データの更新には注意が必要）
 - 利用時には更新ログ(bin-log)を利用

Webアプリケーションでは参照が95%、更新が5%というケースも (Digg.com)
 > シンプルなスケールアウト構成によって簡単に20倍以上の性能向上が図れる

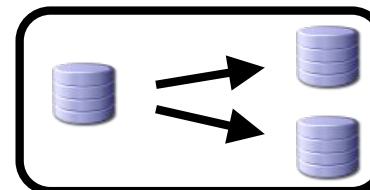


MySQL レプリケーションの構成パターン

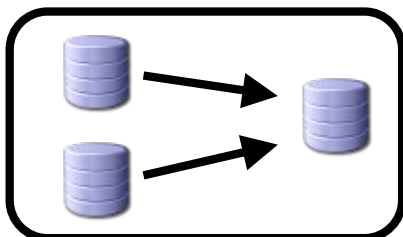
マスタ > スレーブ



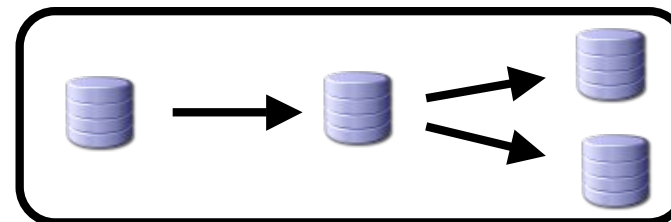
マスタ > マルチスレーブ



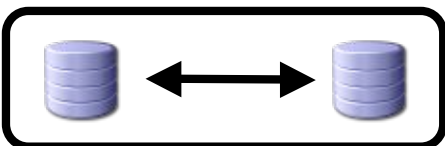
マルチマスタ > スレーブ (マルチソース)



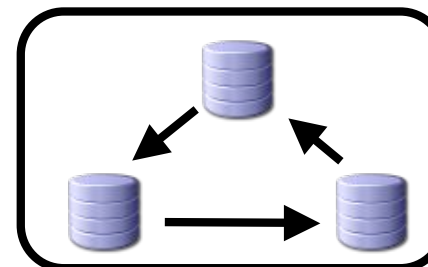
マスタ > スレーブ > マルチスレーブ



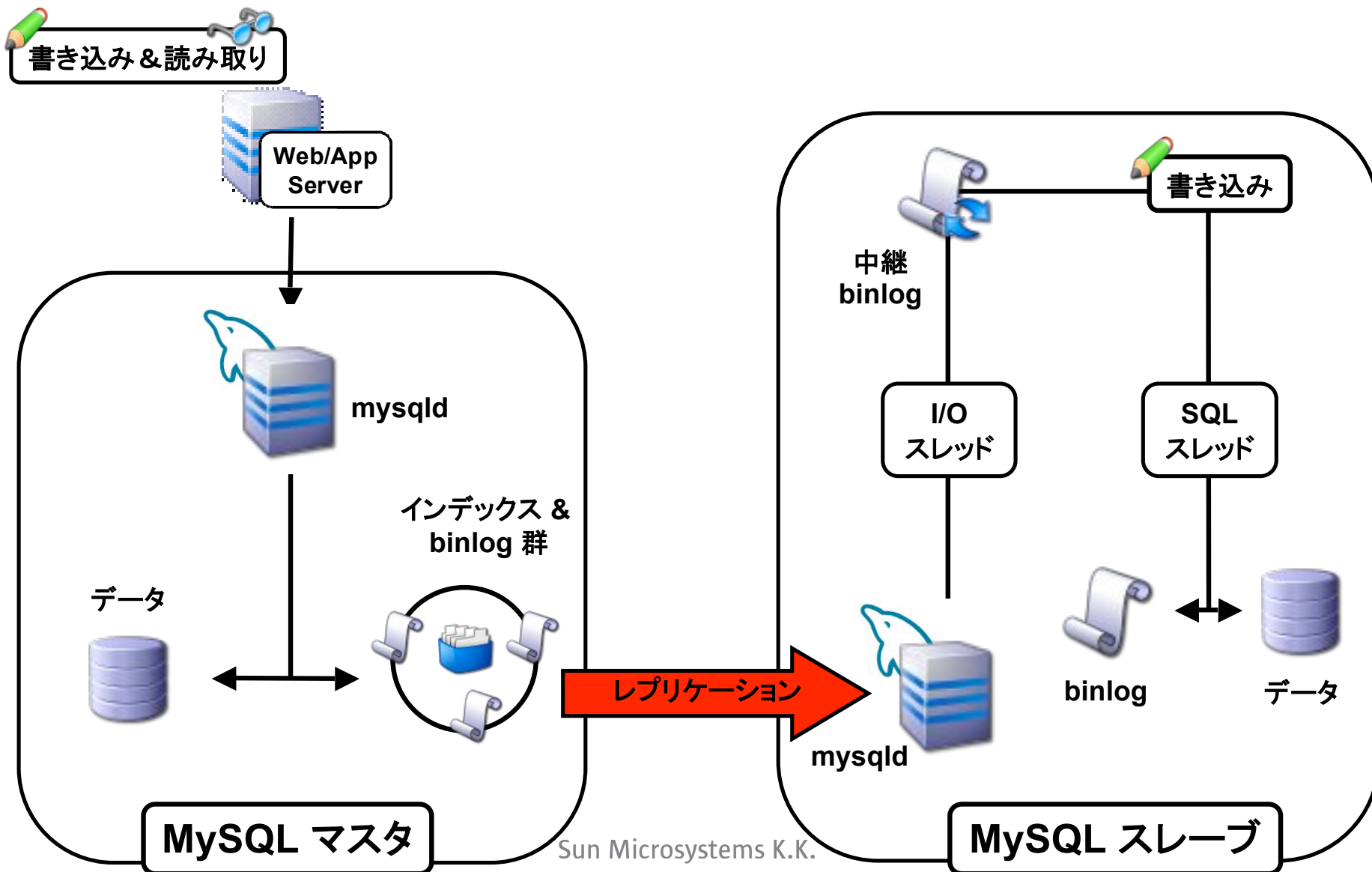
マスタ <> マスタ (マルチマスタ)



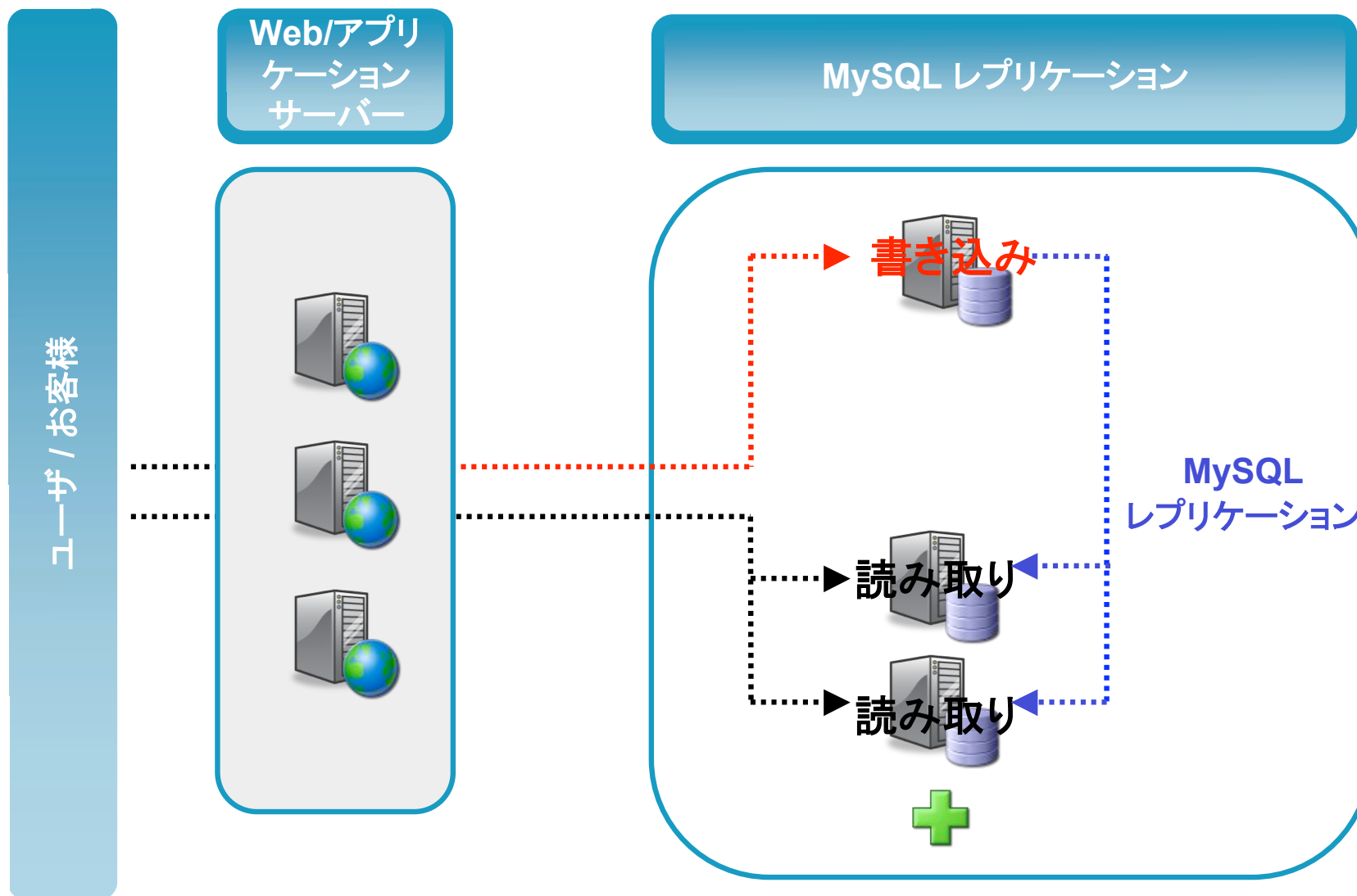
循環型 (マルチマスタ)



MySQL レプリケーションの仕組み



レプリケーションによるスケールアウト構成



DRBD

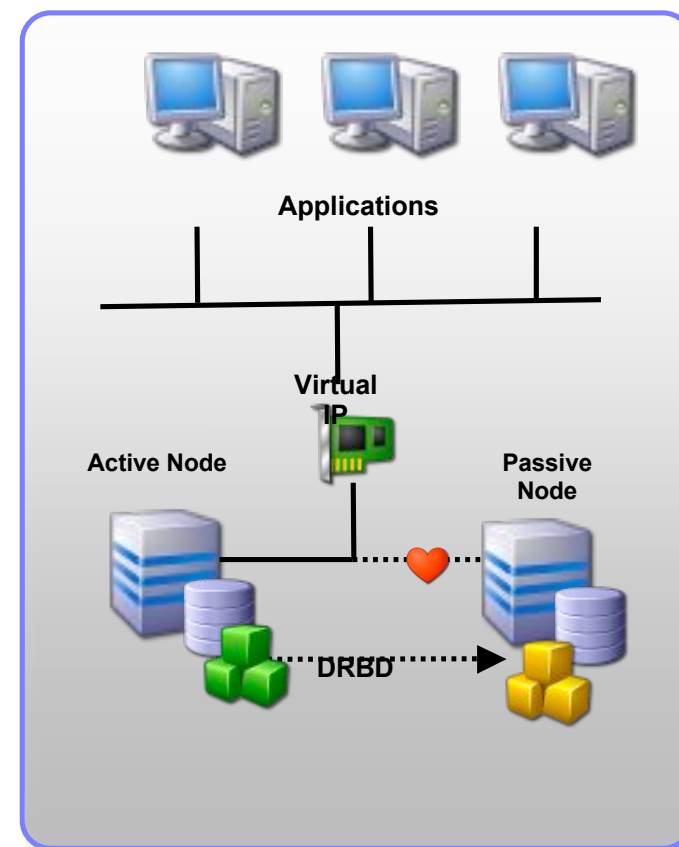
- **Distributed Replicated Block Device**

- 一般的なIPネットワーク上で動作
- 分散ストレージ
- ネットワークRAID

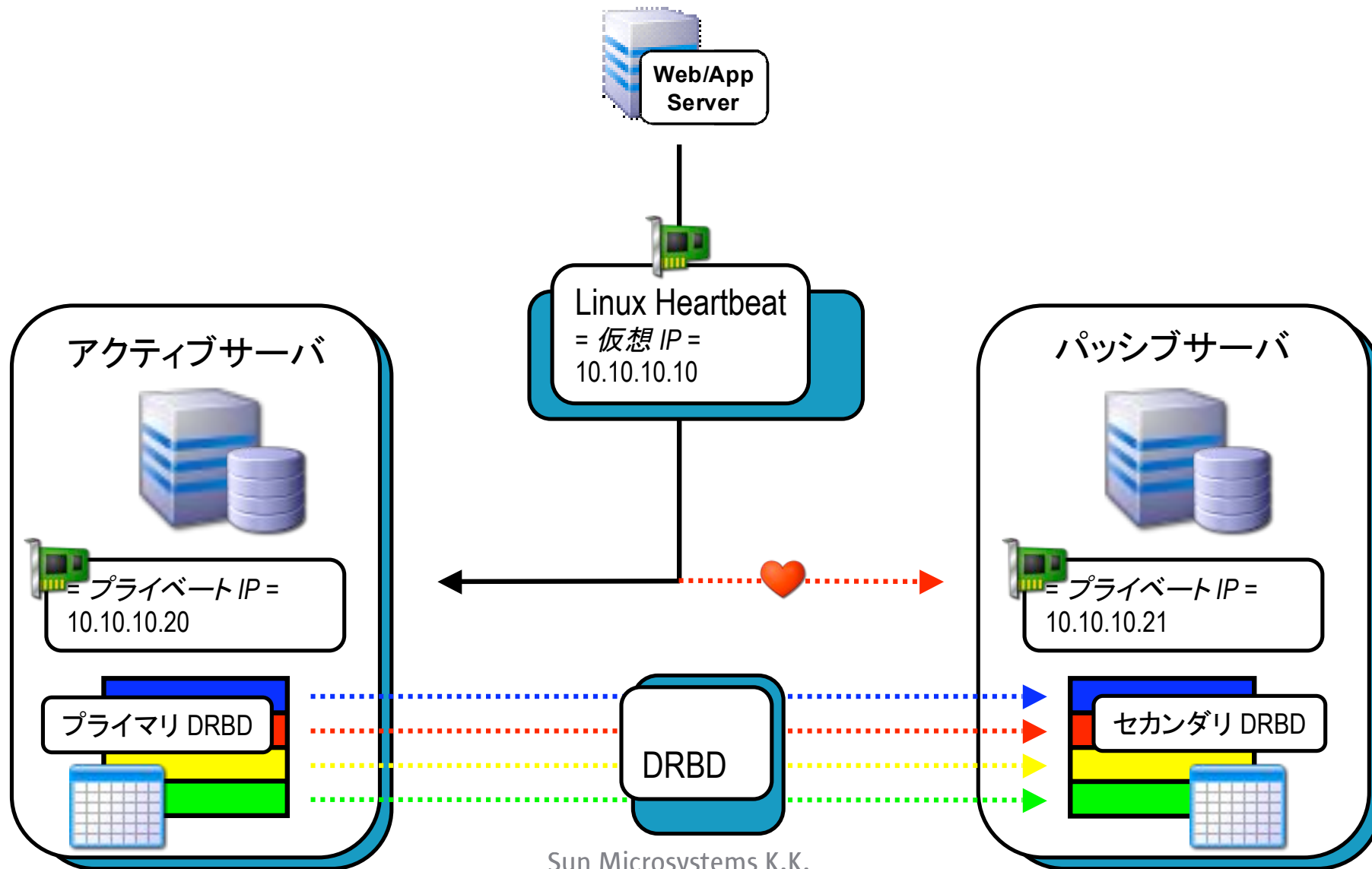
- 同期型

- 特徴

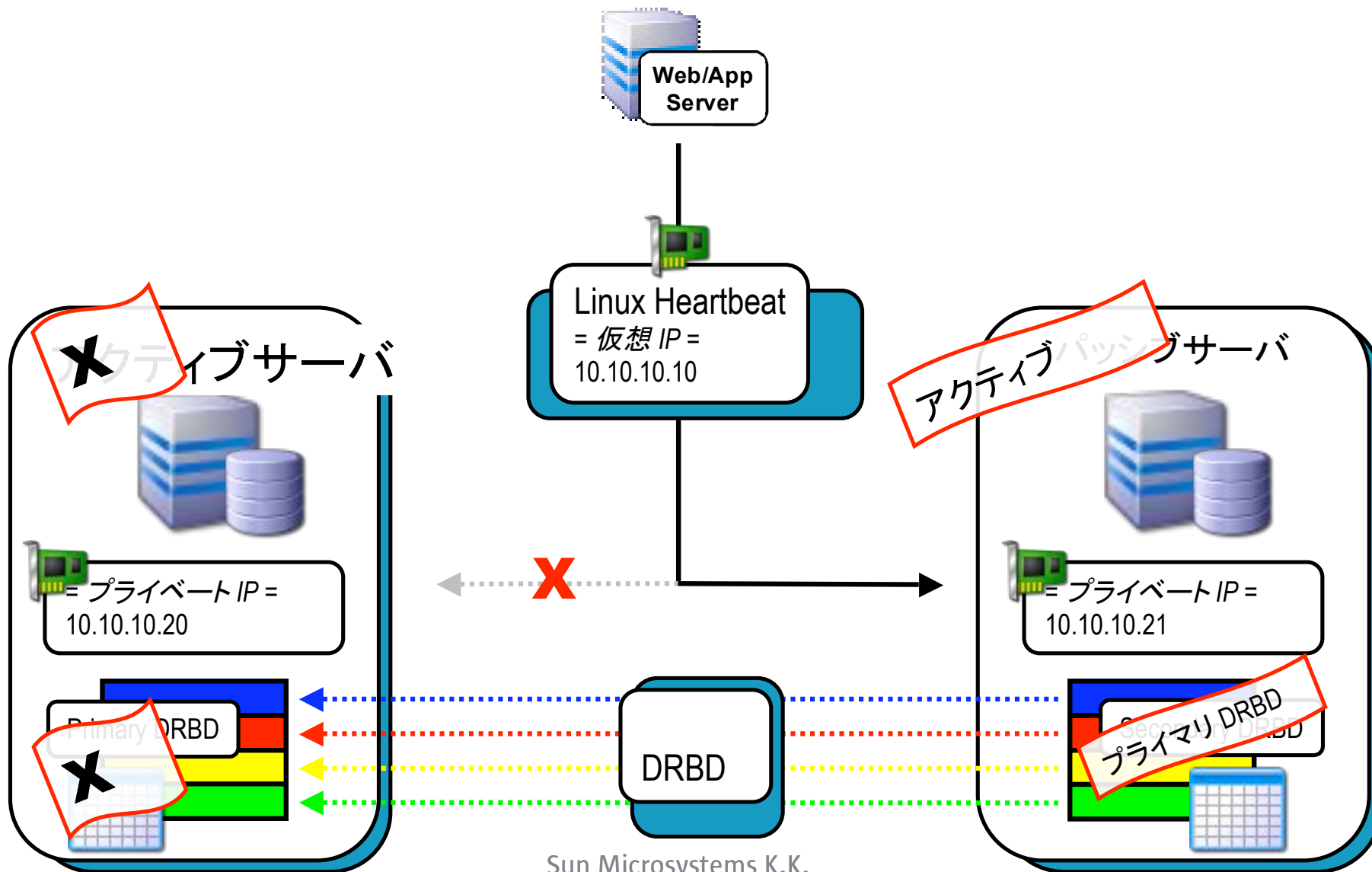
- Linux上のみで利用可能
- 特殊なネットワーク関連コンポーネント不要 (Heartbeatを除く)
- 高いパフォーマンス (blocks vs. rows of data)
- 障害発生時のデータの不整合を管理・解決
- Linux Heartbeat が、フェイルオーバおよび仮想 IP を管理



DRBD + Linux Heartbeat

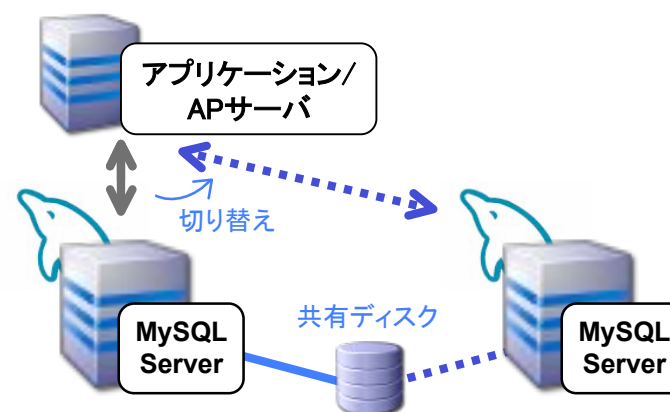


DRBD + Linux Heartbeat (障害発生時)

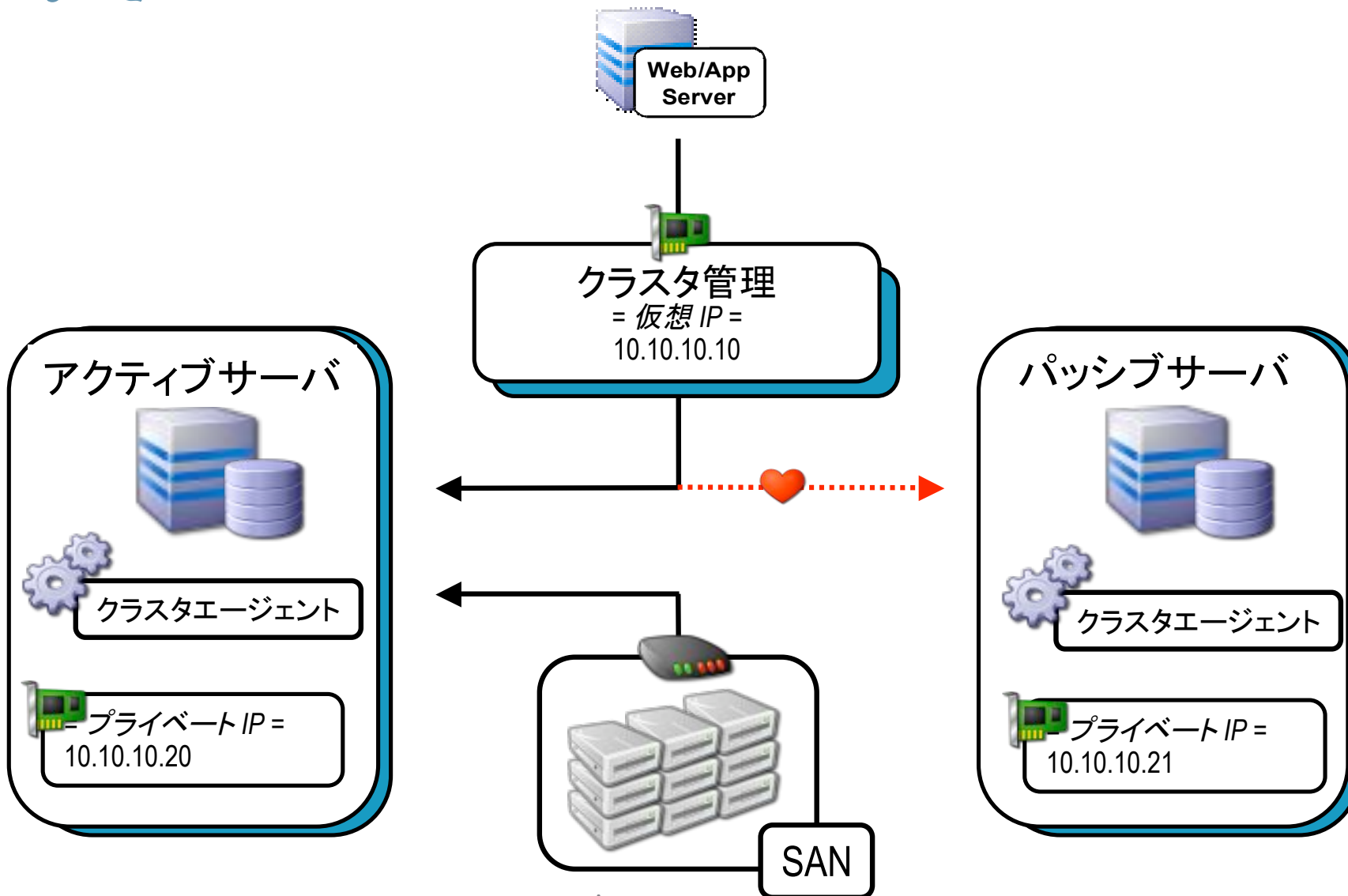


MySQL + 共有ストレージ & クラスタリングエージェント

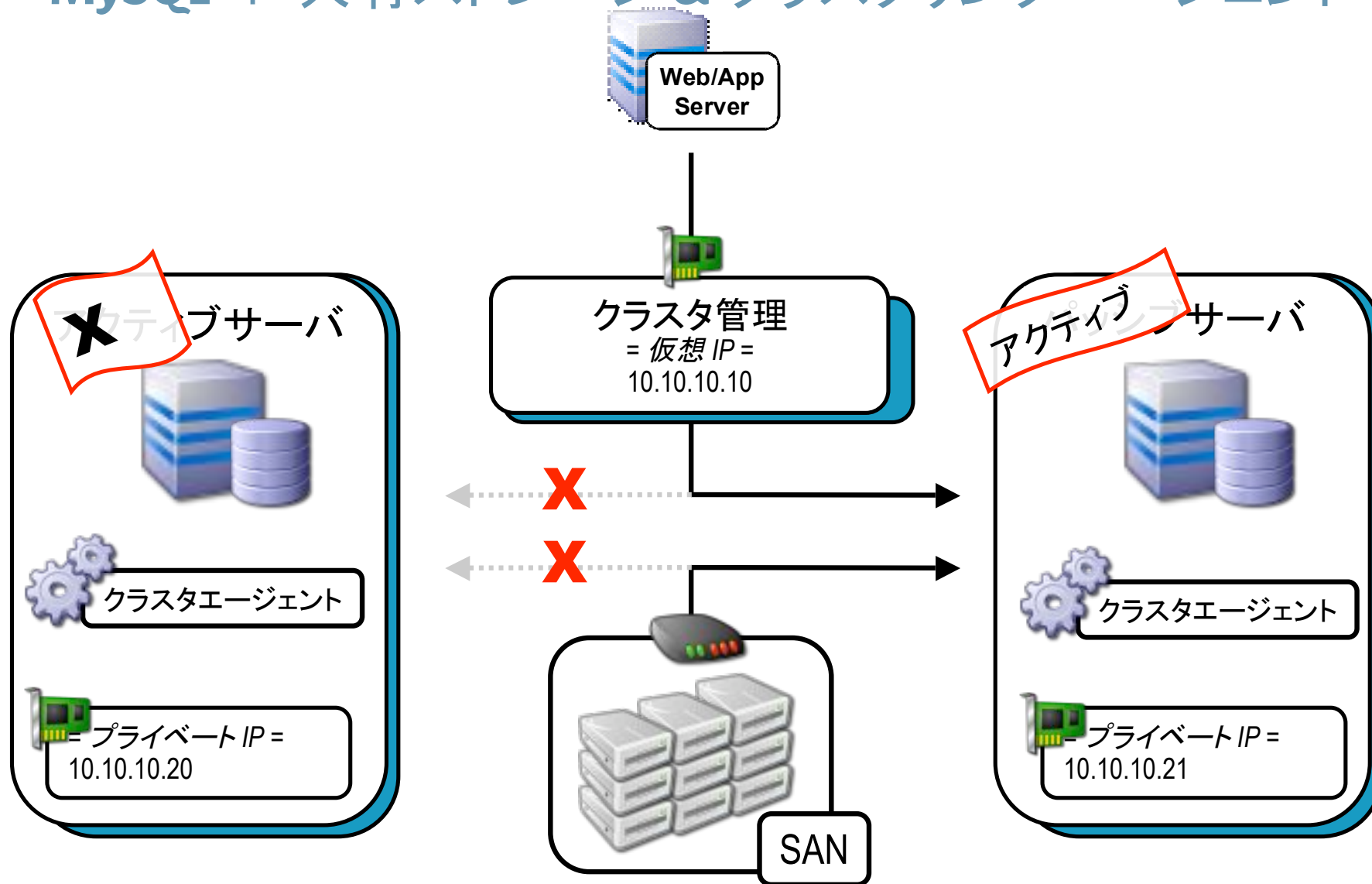
- アクティブ/パッシブ構成
 - クラスタ構成中の1ノードのみがデータファイルにアクセス
 - 共有ディスクを利用
 - クラスタリングソフトウェア
- 同期型
- 特徴
 - 仮想IPによるフェールオーバー
 - クラスタリングソフトウェア (多くの場合有償)
 - 平常時には待機しているだけのリソースが必要
 - MySQLに限らず多くのシステムでの実績



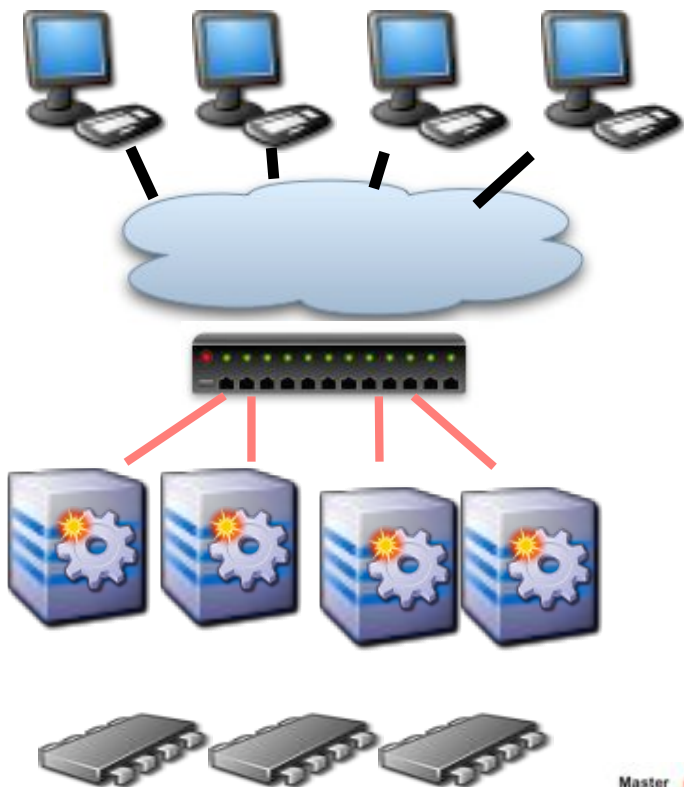
MySQL + 共有ストレージ & クラスタリングエージェント



MySQL + 共有ストレージ & クラスタリングエージェント

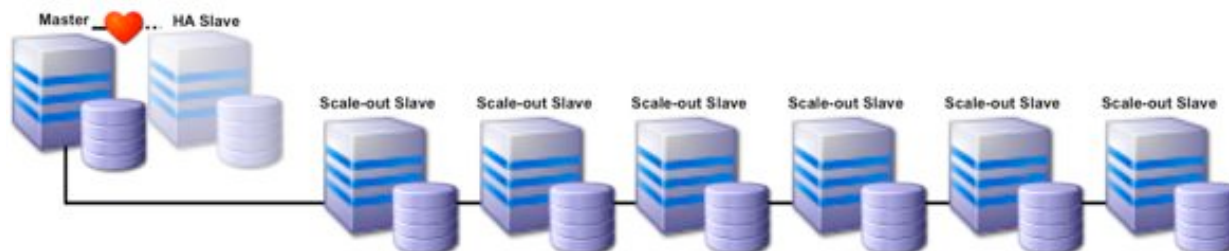


レプリケーションマスターの冗長化



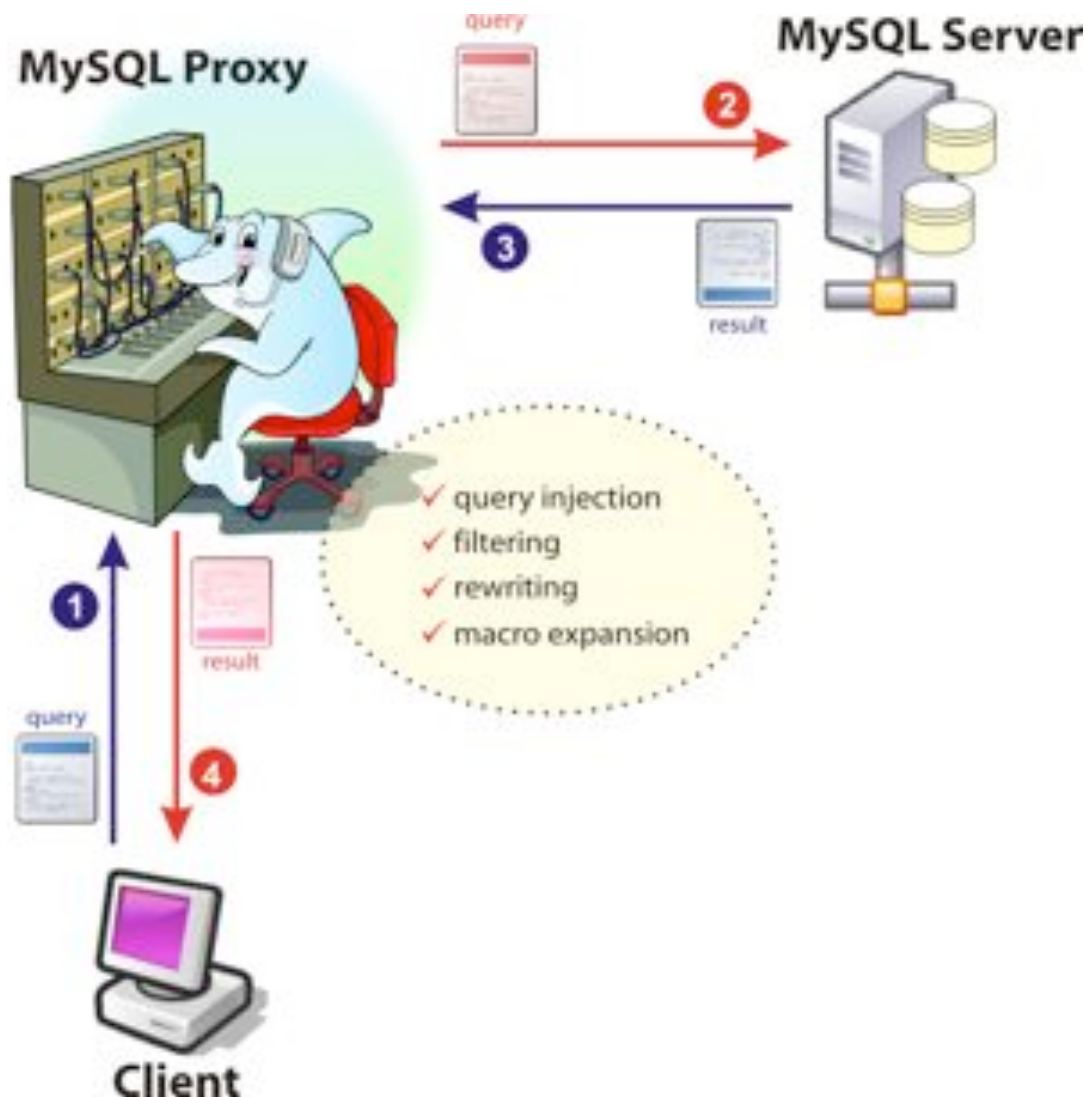
レプリケーション構成のマスターサーバは、DRBDや共有ディスク型のActive/Passiveクラスターで冗長化するケースが多い

スレーブサーバで障害が起きた際には別のサーバに再接続



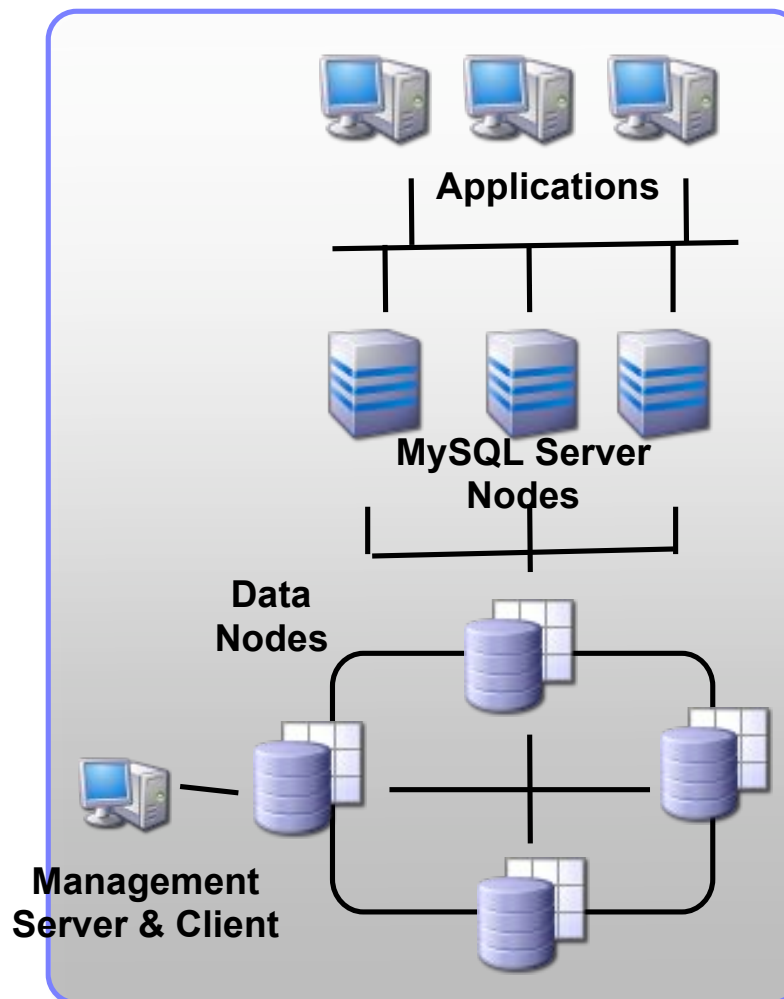
MySQL Proxy

- クライアントとサーバの間で稼働する計量なアプリケーション
- LUA言語のインタプリタを同梱しているため、ユーザが機能拡張可能
 - <http://www.lua.org>
- 用途例
 - ロードバランス
 - フェールオーバー
 - ログイン
 - クエリの書き換え
 - パーティショニング



MySQL Cluster

- シェアードナッシング型クラスタ
- コスト
 - 特別なハードウェアを必要としない
 - 共有ディスクも不要
- 耐障害性
 - 単一障害点(single point of failure)なし
- 高可用性
 - データは複数のノードに記録される
 - 自動フェールオーバー
- スケーラビリティ
 - 参照：コピーされたデータ
 - 更新：分割されたテーブル
 - スケールアウト構成
- 高性能
 - 負荷分散
 - インメモリデータベース
(ディスクへのデータ格納も可能)
 - 100,000/秒のリクエストに対応するように設計されている
- 管理の簡素化
 - 管理用ユーティリティ
 - 一般的な技術スタック




MySQL Cluster

Carrier Grade Database	<ul style="list-style-type: none">• Shared-nothing in-memory parallel database server• ACID compliant relational database
Highly Available	<ul style="list-style-type: none">• Five nines (99.999%) availability• Self-healing, sub-second failover
Real-time Performance	<ul style="list-style-type: none">• High load, real-time performance• Predictable low latency, bounded access times
Linearly Scalable	<ul style="list-style-type: none">• Incrementally grow out with application demands• Linearly scale with distribution awareness
Open Development	<ul style="list-style-type: none">• Open source, multiple data access• High performance APIs (C++ / Java), SQL, LDAP

MySQL Cluster Components

SQL Node
(MySQL)




- Standard SQL Interface
- Scale-out for Performance
- Enables Replication

NDB API
(Application)



- High Performance
- C++ API
- Developer's Guide

Data Node
(NDB Storage Engine)



- Data Storage (Memory/Disk)
- Automatic & User-Defined Partitioning
- Local & Global Checkpoints
- Scale-out for Capacity & Redundancy

Management Node

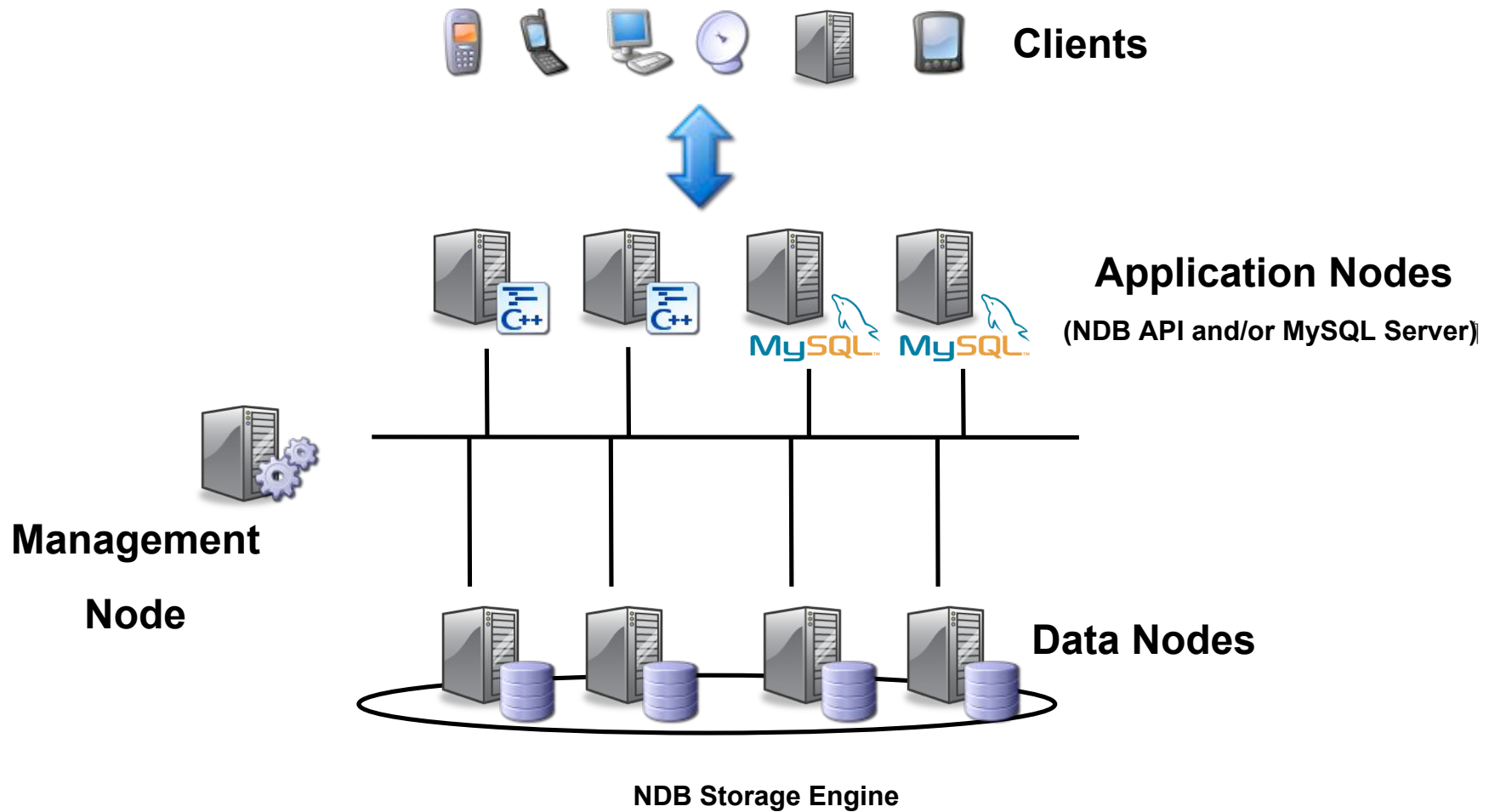


- Administration and Configuration
- Arbitration
- Use Two for Redundancy

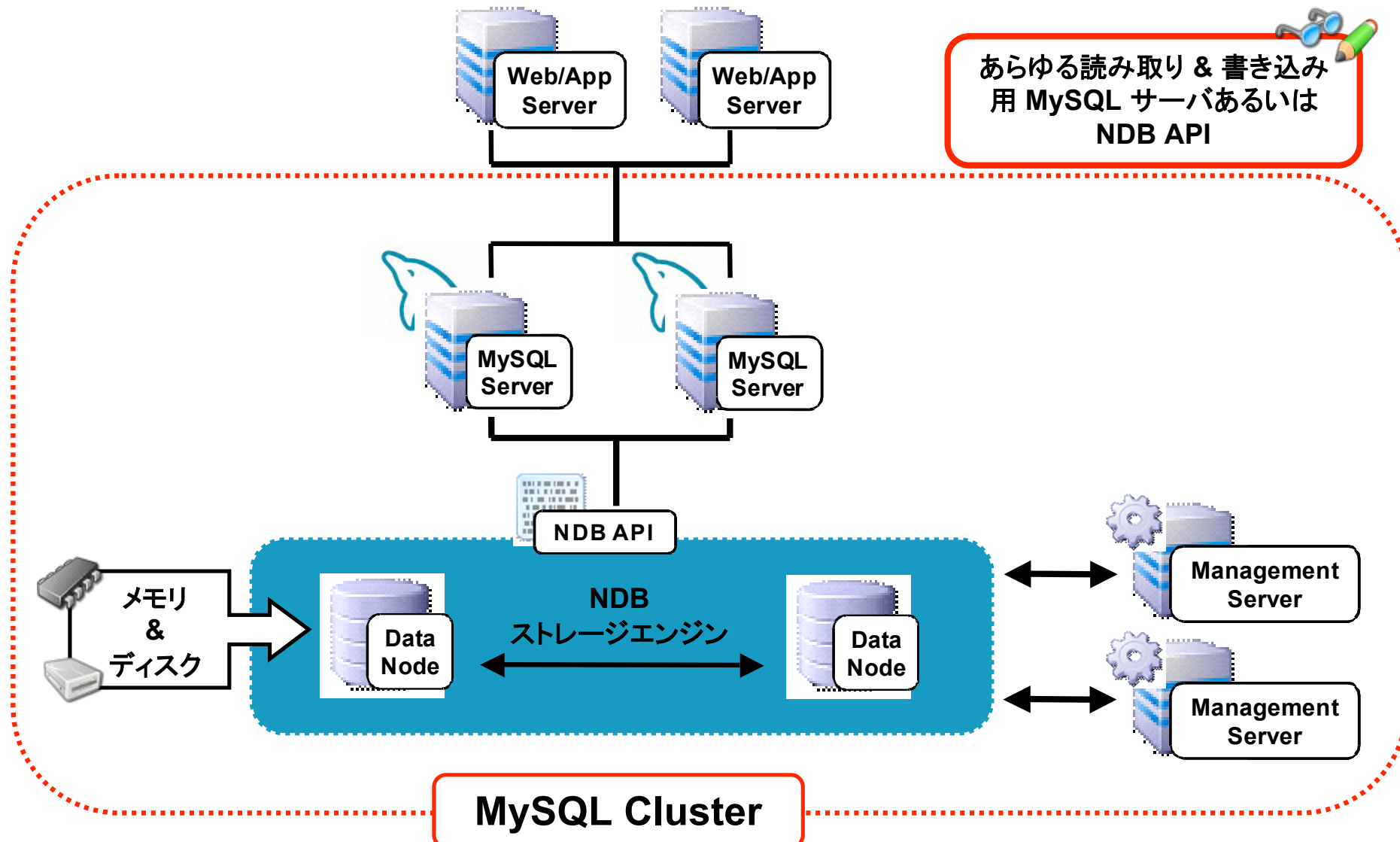
MySQL Cluster 構成要素

- データノード
 - 冗長データフラグメントの保管
 - 他のノードと連結させ、ノードグループを形成
- 管理ノード
 - 起動時に実行
 - 再コンフィギュレーション
- MySQL ノード
 - 標準的な SQL インタフェース
 - スケールアウト
 - NDB API

4 Data Node MySQL Cluster



基本 Cluster アーキテクチャ



MySQL Cluster データノードアーキテクチャ

ID	Capital	Country	UTC
1	Copenhagen	Denmark	2
2	Berlin	Germany	2
3	New York City	USA	-5
4	Tokyo	Japan	9
5	Athens	Greece	2
6	Moscow	Russia	4
7	Oslo	Norway	2
8	Beijing	China	8

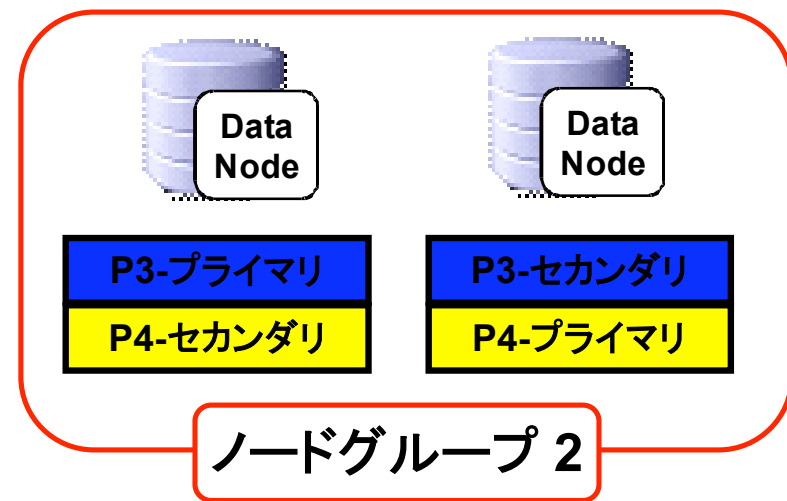
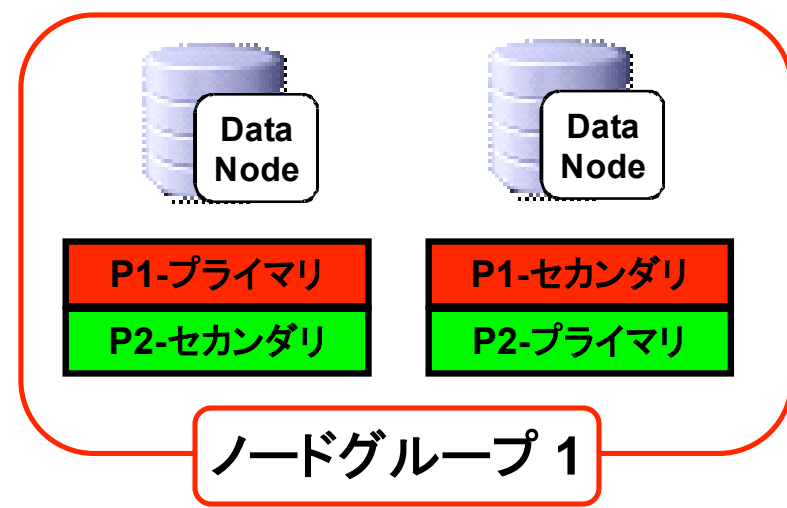
Partition 1

Partition 2

Partition 3

Partition 4

- 4つのデータノード
- 2つのレプリカ
- 2つのノードグループ



MySQL Cluster データノードアーキテクチャ

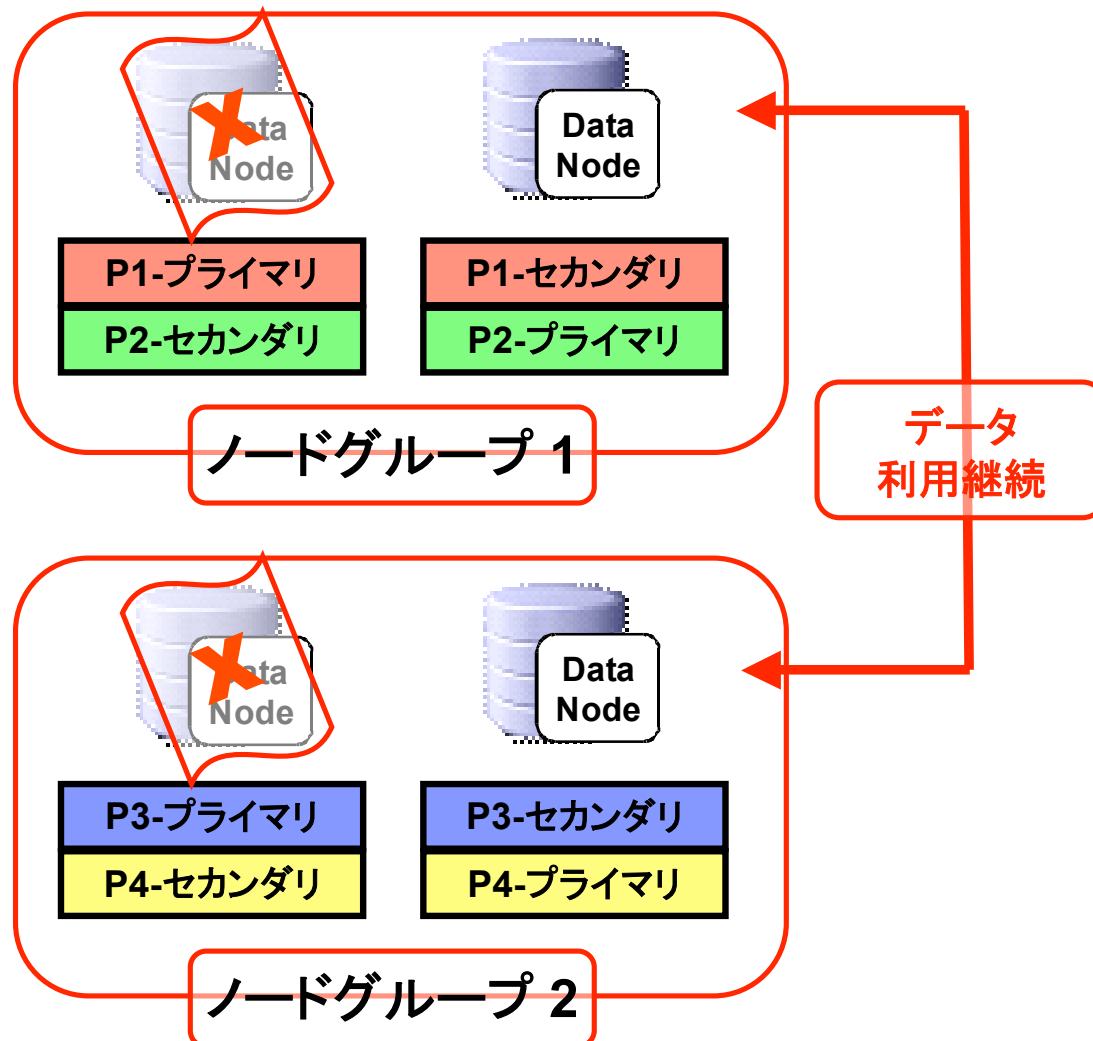
ID	Capital	Country	UTC
1	Copenhagen	Denmark	2
2	Berlin	Germany	2
3	New York City	USA	-5
4	Tokyo	Japan	9
5	Athens	Greece	2
6	Moscow	Russia	4
7	Oslo	Norway	2
8	Beijing	China	8

Partition 1

Partition 2

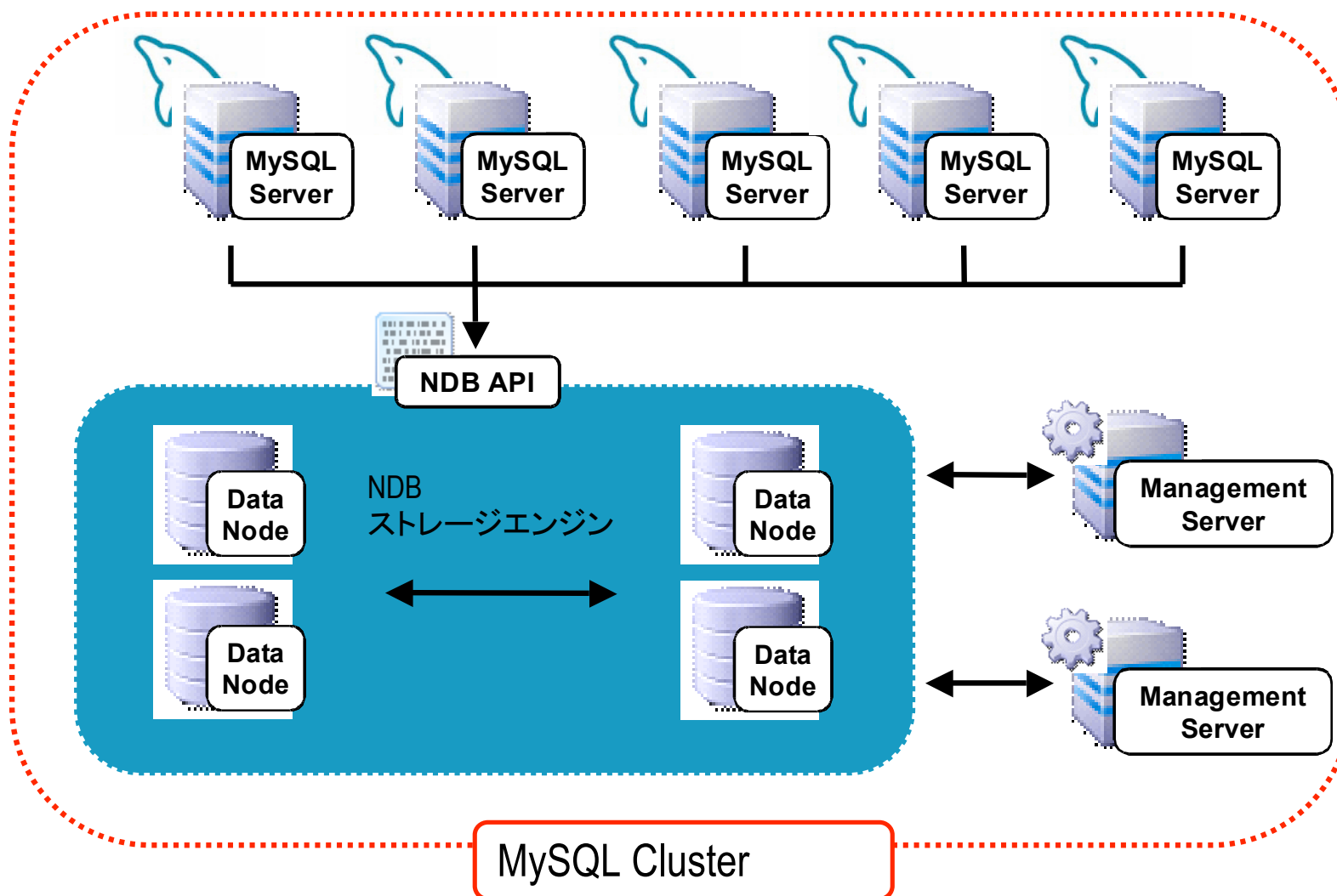
Partition 3

Partition 4



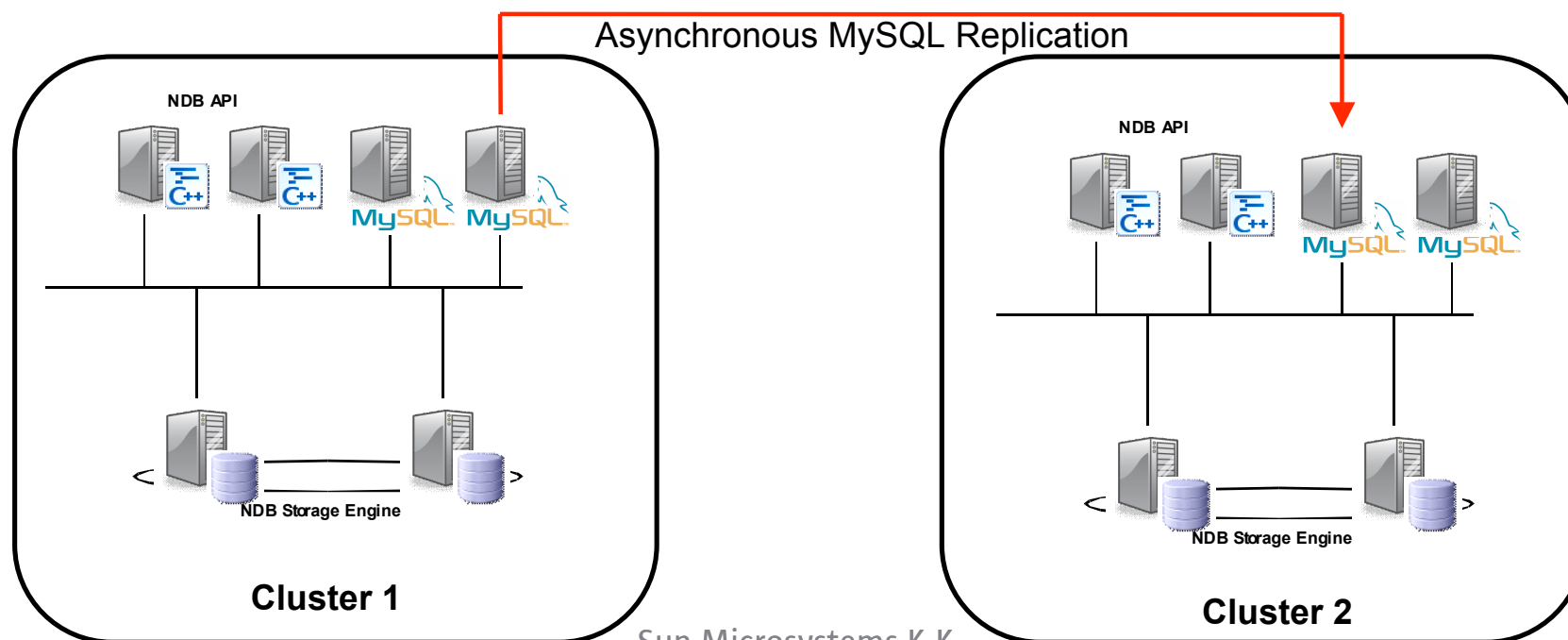
- 4つのデータノード
- 2つのレプリカ
- 2つのノードグループ

スケールアウト Cluster アーキテクチャ

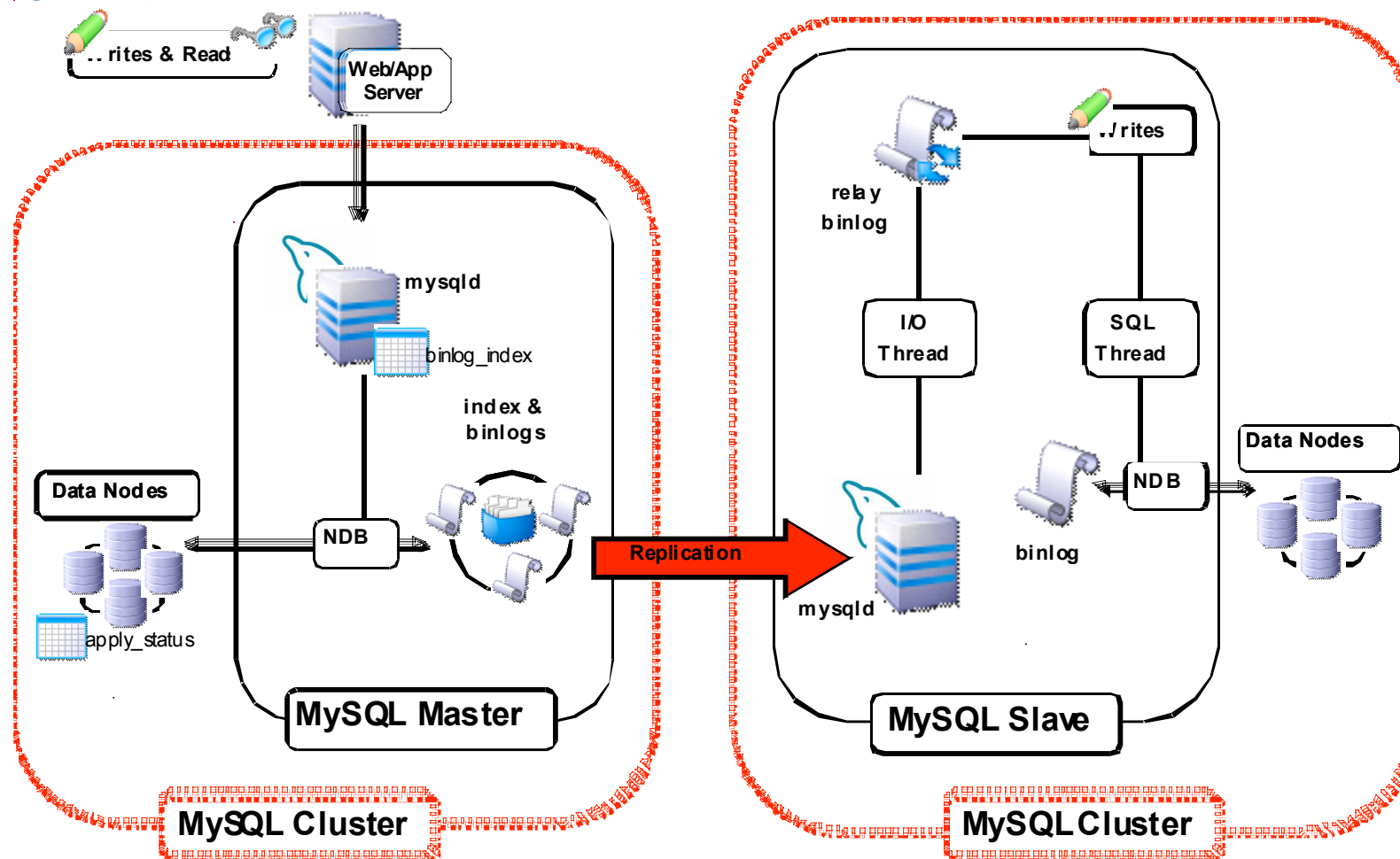


Geographical Redundancy

- Asynchronous replication using MySQL Server
- Various configurations
 - MySQL Cluster to MySQL Cluster
 - MySQL Server to MySQL Cluster
 - MySQL Cluster to MySQL Server



MySQL Cluster とレプリケーション



グローバルな冗長化: ビジネス継続性 & 障害復旧

Sun Microsystems K.K.

MySQL Cluster ライセンス

利用機能によってライセンスが異なる

SE (Standard Edition) MySQL Clusterの基本機能

CGE (Carrier Grade Edition) 先進的な高度機能

MySQL Cluster 機能	SE	CGE
インメモリデータベース	☑	☑
ACID トランザクションサポート	☑	☑
オンラインメンテナンス/アップグレード	☑	☑
シェアードナッシングアーキテクチャ	☑	☑
サーバ間の同期的なデータコピー	☑	☑
自動でのクラッシュリカバリ	☑	☑
オンライン”ホット”バックアップ	☑	☑
Linear Scalability on COTS Hardware	☑	☑
SQLによるアクセス	☑	☑
NDB APIによる(C/C++/Java)		☑
クラスタ間のレプリケーション(矛盾解決機能を含む)		☑ ¹
LDAPインターフェース		☑ ¹

MySQL User Conference Japan 2008

- 2008年10月30日から31日の日程で開催予定
 - > 東京ステーションコンファレンスにて（東京駅八重洲北口）
 - > 日本でのユーザ事例のご紹介や技術系セッションを予定
[予定しているセッションの例]
 - > 金融機関の情報系/基幹系システム事例
 - > パフォーマンスチューニング
 - > 製品ロードマップ
 - > 商用データベースからのマイグレーション
- セミナー修了後MySQLの開発者との交流会を開催
 - > スポンサー様による前夜祭やサテライトイベントも

サンのオープン・スタック

Flexible and Heterogeneous with Zero Barrier to Exit

Database Platform	
Application Infrastructure	
Virtualization	
Operating System	
Partners	
Architecture	



rkajiyama@mysql.com
Ryusuke.Kajiyama@sun.com