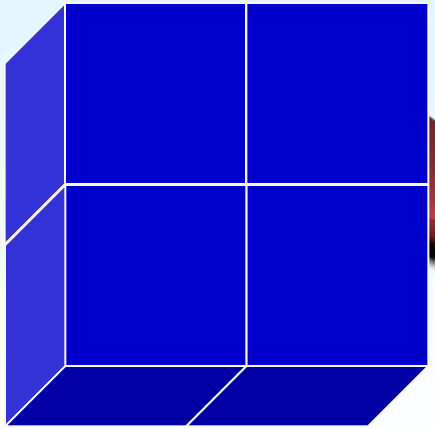
HPCシステムでのストレージシステムの動向

スケーラブルシステムズ株式会社

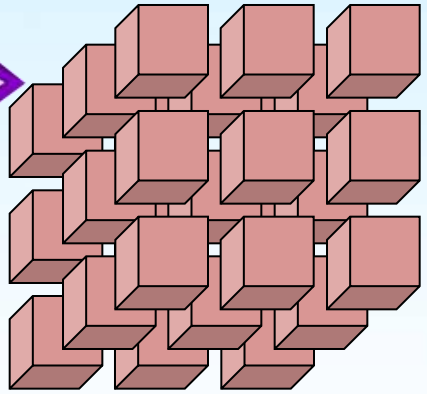
コンピュータシステムの変遷

メインフレーム
スーパーコンピュータ

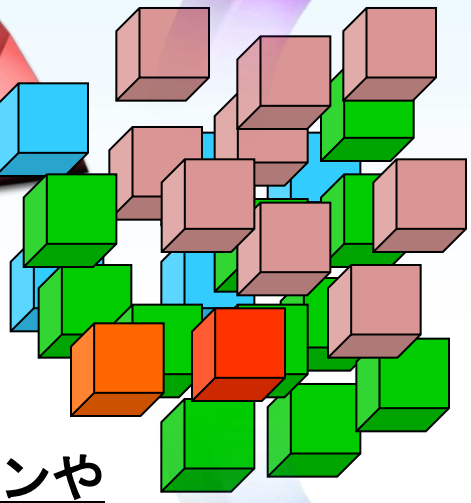


（リソースの集中と管理）

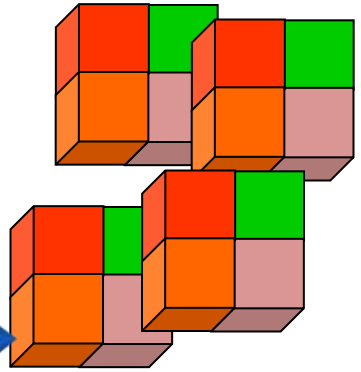
■クラスタによる仮想コンピュータ



分散したリソースの管理



■ワークステーションや
サーバによる分散処理

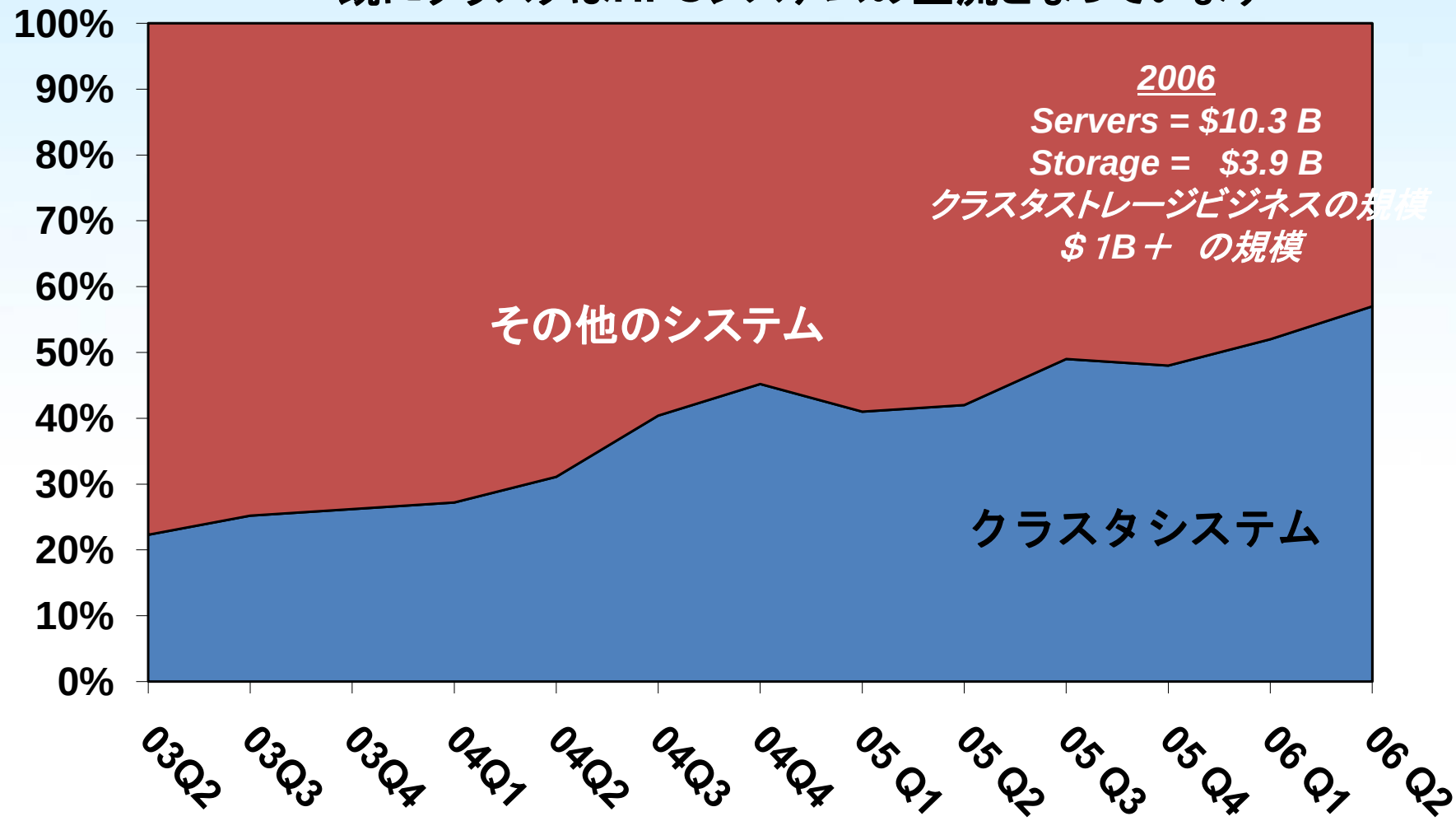


■仮想化によるサーバ・コンソリデーション

システムの課題

- 代表的なプラットフォームアーキテクチャ
  - クラスタシステム　　（1－2pノード）
  - SMPシステム　　（＞2pノード）
  - SMPシステムをベースとしたクラスタシステム
- ハイパフォーマンスシステムの利用の現状
  - クラスタシステムがより一般化している
  - SMPシステムは、より高い付加価値を必要とするユーザやアプリケーションで利用

HPCクラスタがもたらすビジネス



IDC - 2006

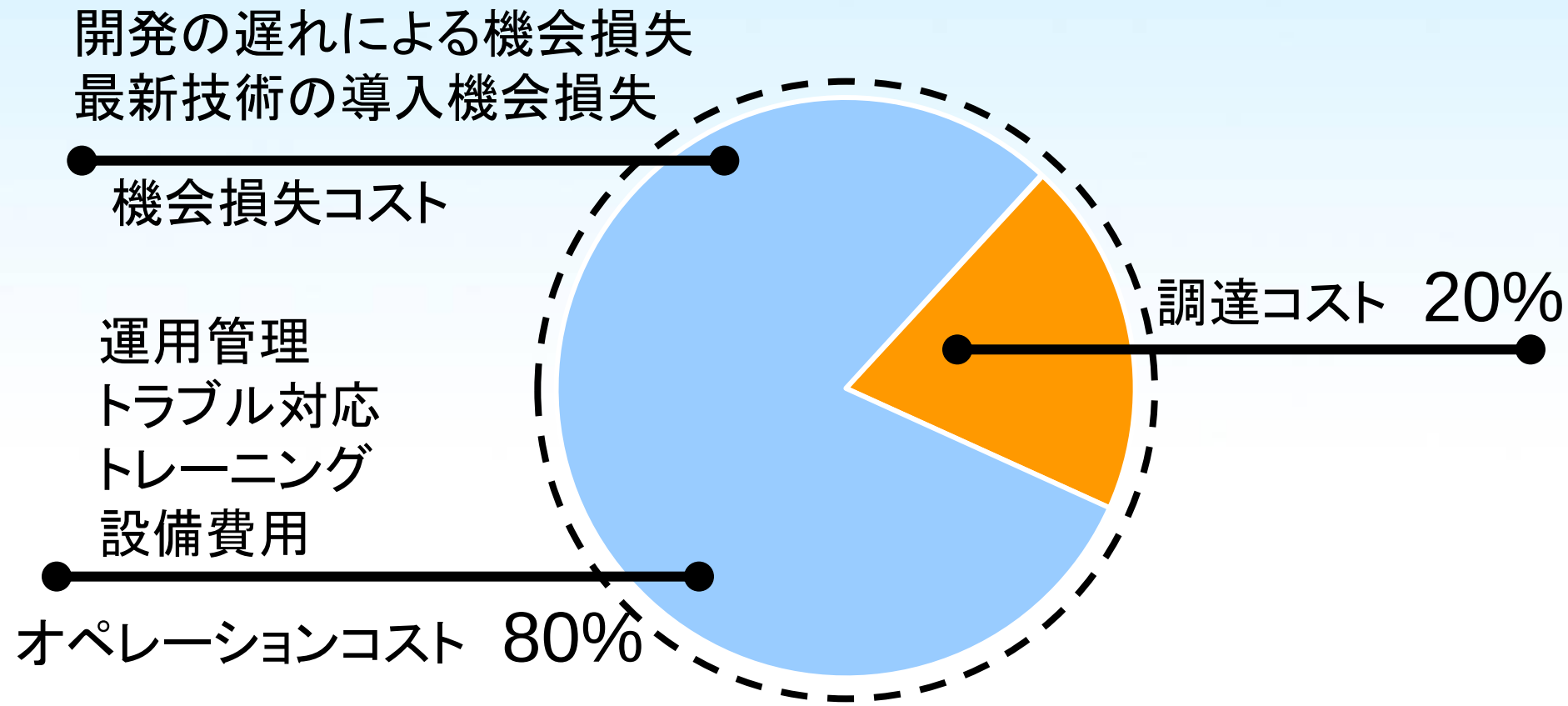
クラスタシステムの利点

- ハードウエアコストの劇的な低下
- 非常に高いピーク性能のシステムの導入が可能
- 増設が容易で、必要に応じて、システムの規模の拡大が容易
- 標準コンポーネントの技術革新と性能向上
  - プロセッサの性能向上（‘マルチコア’による省電力での性能向上）
  - 高性能なスケーラブルファイルシステム（オープンソース）
  - 高速な商用インターコネクトスイッチ

ハイパフォーマンスシステム

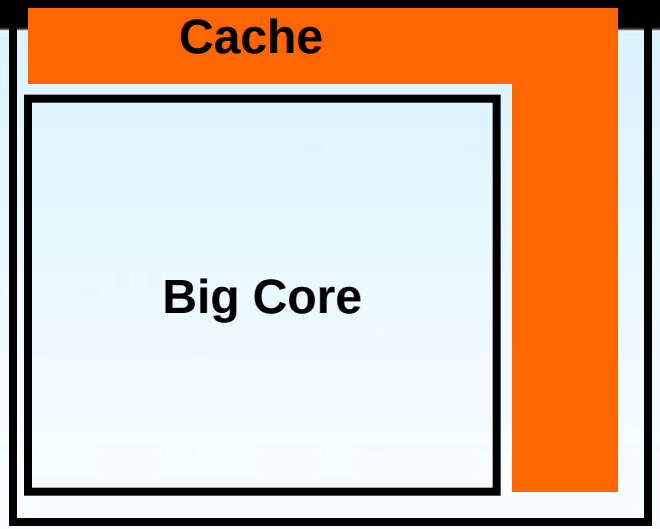
- ハイパフォーマンスシステムの増強のニーズは高い
  - より大規模な解析
  - より多くのシュミレーション
  - より短い時間でのシュミレーションの完了
- 同時にシステムに対するコスト・パフォーマンスの要求も厳しい
  - ベンダー間での競合
  - アプリケーションのスケーラビリティ
  - より大規模なシステムの導入の希望
- 「コスト・パフォーマンス」に対する要求も強い

TCO : Total Cost of Ownership

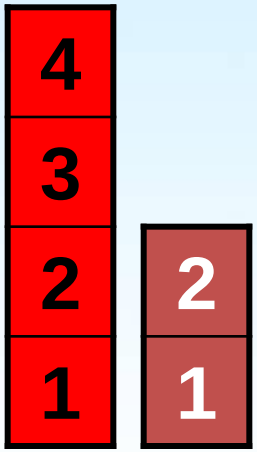


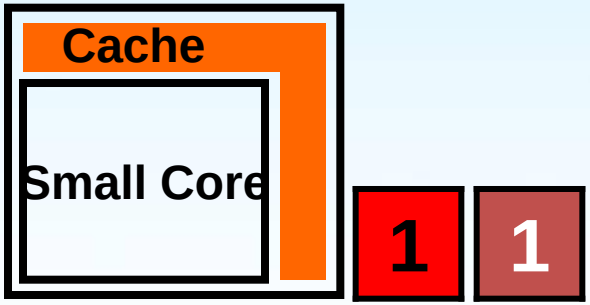
Source: Gartner Group 2005

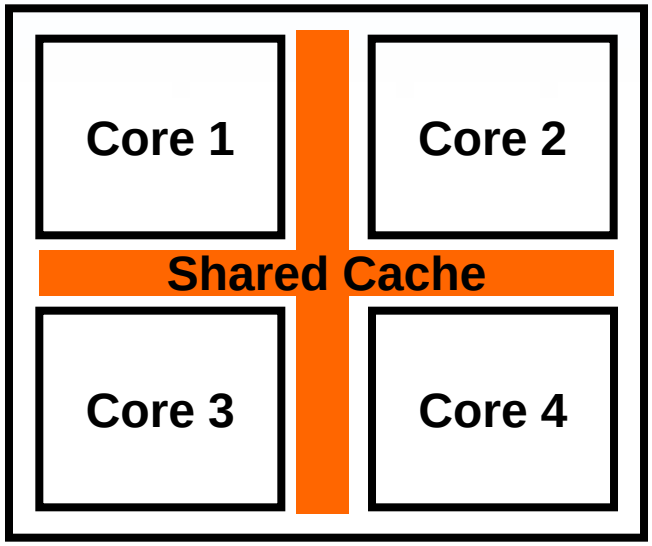
マルチコア：‘性能/消費電力’を改善

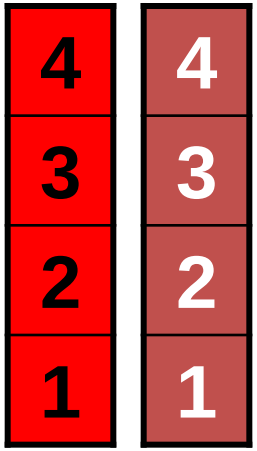


消費電力，性能









Power ~ コアサイズ
PERFORMANCE ~ √コアサイズ

顧客が直面する問題

- セットアップは非常に苦痛
  - クラスタの設定に時間がかかり、実際にシステムの利用が可能になるまでの時間が無駄になる
- システムのアップデートが煩雑
  - 既存のITインフラとHPCシステムの互換性の問題
  - Windows環境とLinux環境（セキュリティ、ユーザ管理）
- ジョブ管理
- アプリケーションの運用
  - 並列アプリケーションの効率的な運用
  - プリ・ポストとの連携
  - ストレージシステム

SMPとクラスタのギャップ

SMP (Shared Memory Systems)

ワークステーションやサーバ
PA-RISC,POWER5,
Itaniumなどのプロセッサ
によるSMPサーバ

クラスタシステム

システムの構築には、
高いITスキルが要求される
運用管理コストが高い
複雑なオペレーション環境
複数のOS
クラスタファイルシステム
ソフトウエア、インストールや
アップグレードなど

?

ワークステーション
サーバ

クラスタ

#Processors 2 4 8 16 32 64 128

顧客の要求

- エンドユーザ：
  - シンプルにセットアップが可能
  - アプリケーションが豊富で、簡単に導入出来る
  - 簡単なジョブの投入とモニターリング
  - 高い実行性能とスケーラビリティ
- 運用管理者：
  - IT環境をシンプルに
  - シンプルにクラスタを導入し、シンプルな運用で、高い生産性を実現できる環境の構築
- アプリケーション開発者：
  - 生産性の高い開発環境
  - シンプルなシステムデザイン
  - 標準のライブラリやAPI環境

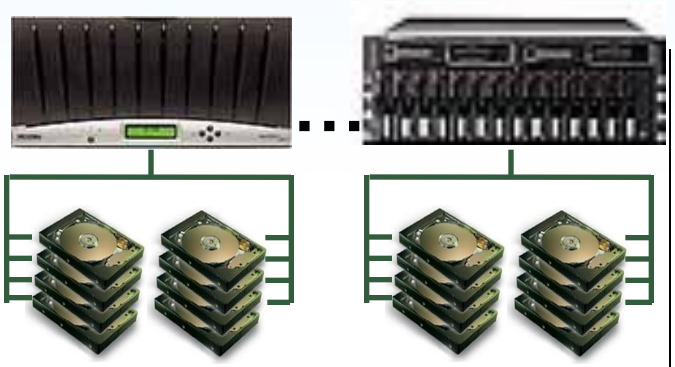
ストレージアーキテクチャ

Linux計算クラスタ

Direct Attached Storage

課題: 運用管理

Networked Attached Storage



課題: 性能

Storage Area Network

課題: 運用管理とコスト

共通の課題と問題

性能、運用管理の容易さ、スケーラビリティ、コスト

ストレージに関する課題

クライアント（エンドユーザ）

クラスタ
- 計算クラスタはI/O処理の終了まで計算を中断
- I/O処理は、クラスタの利用率の低下を引き起こす
- ノード数を増やした場合のスケーラビリティの維持の問題

クライアント
- ジョブの実行終了を待つ
- ユーザ数が増えた場合のスケーラビリティの問題
- ユーザ間でのコラボレーションやデータの共有の問題

BOTTLENECK

BOTTLENECK

従来のネットワークストレージ

BOTTLENECK

クラスタ

バックアップ/リストア
- バックアップ処理のためのストレージシステムの負担
- バックアップ実施のタイミング
- 高速でのバックアップの問題

バックアップ/リストア

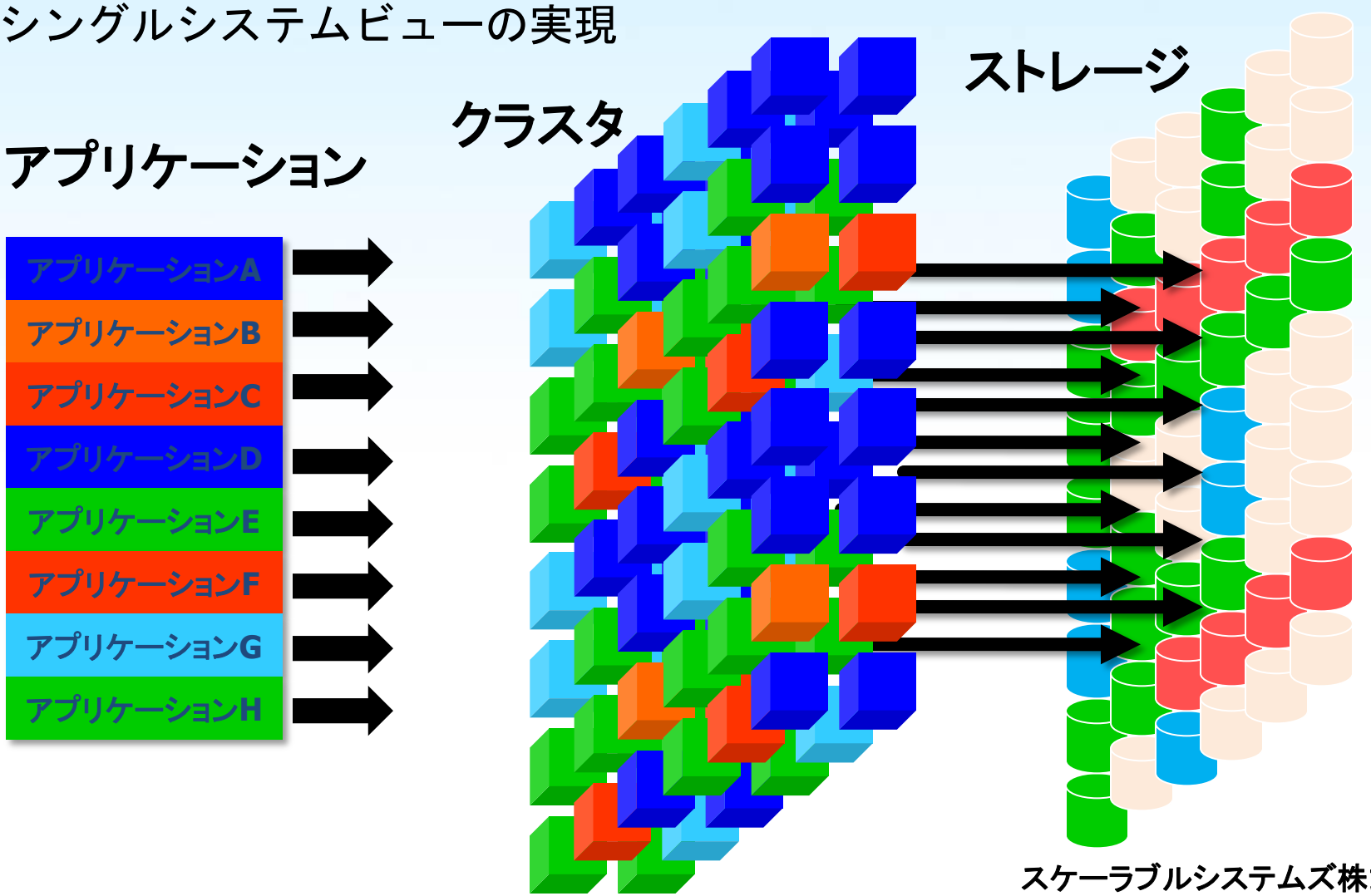
クラスタ環境での従来型ストレージの問題（課題）

- ファイルサーバ(NFS, NetApp, Linux roll-your-own)
  - 性能向上には、高機能なサーバの追加が必要
  - ファイルサーバ間でのデータ共有やロードバランスが課題
- SAN ファイルシステム(ADIC StorNext, GPFS, CXFS, QFS)
  - 数百、数千台のクライアントやノードの処理
  - 高価なスイッチやクライアント側のインフラの準備

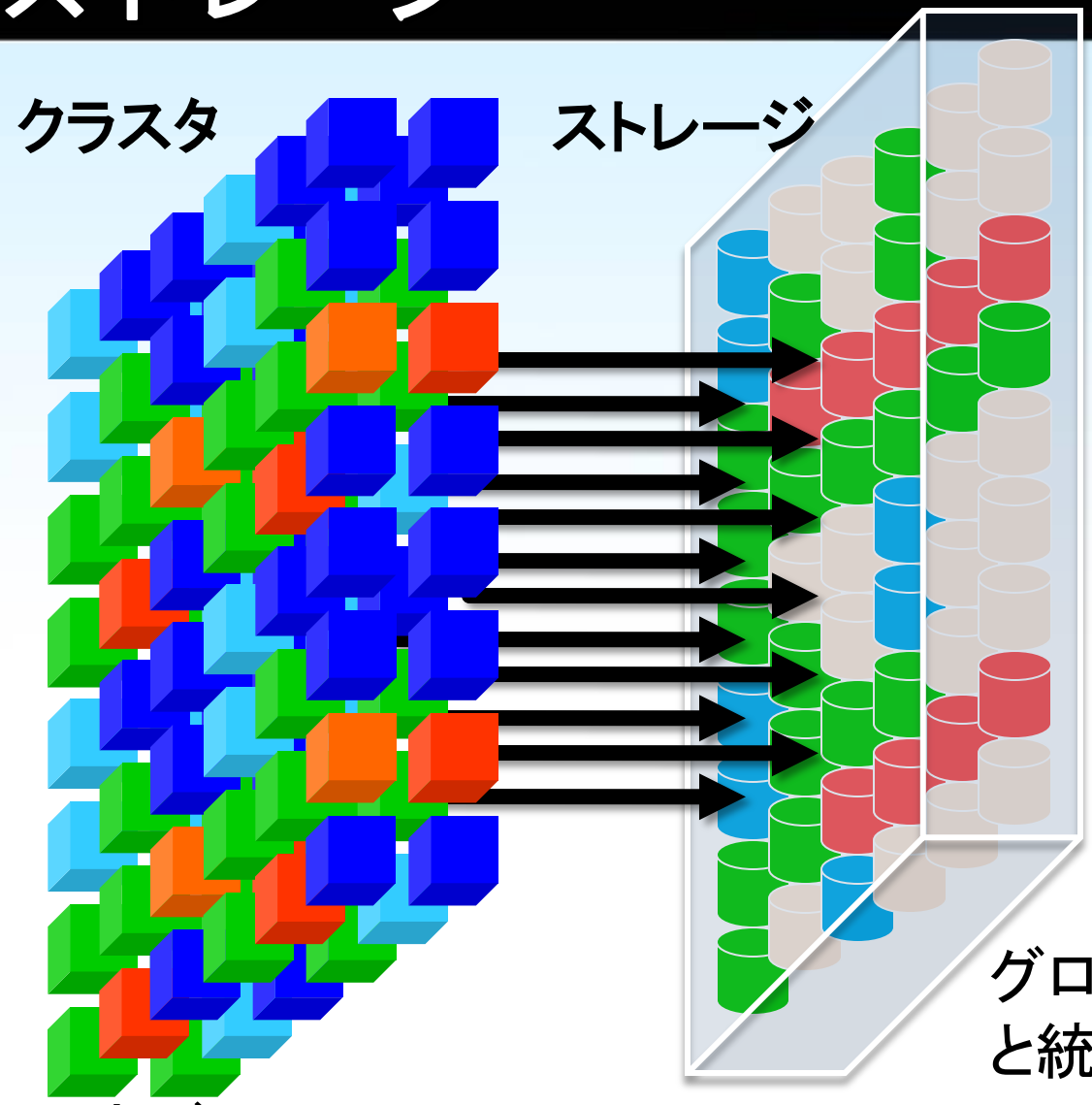
- ブロックベースRAIDシステムの課題
  - ドライブの容量の急激な増大
  - RAIDの再構成や障害時の復旧により長い時間が必要

クラスタ環境でストレージ

- すべての点でスケーラブル：パフォーマンス、容量
- シングルシステムビューの実現



求められるクラスタ環境でのストレージ



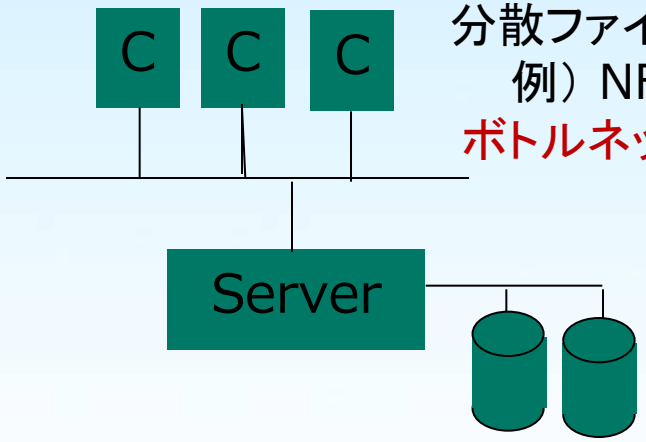
ストレージプール

- 単一の仮想リソース
- 透過的なデータアクセス
- システムの再構築が容易
- 様々なデータの格納が可能
- 可用性

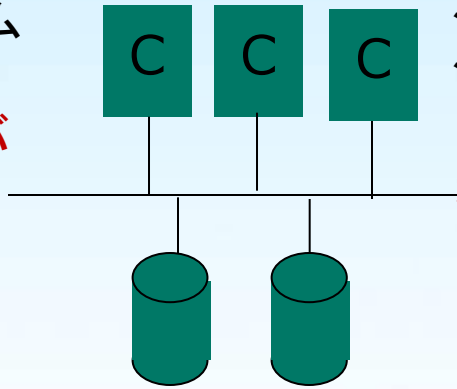
グローバルネームスペースと統合された運用管理環境

OSとバッチシステム

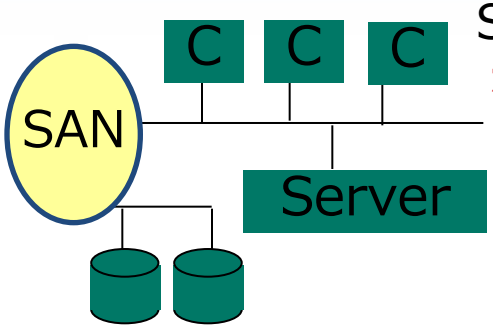
クラスタでのファイルシステム



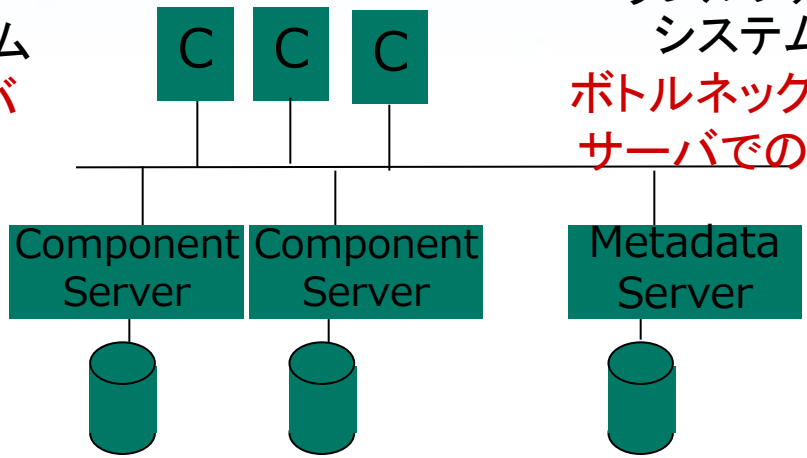
分散ファイルシステム
例）NFS/CIFS
ボトルネック：サーバ



シンメトリックスファイル
システム
例） IBM GPFS
ボトルネック：ファイルア
クセスでのロック処理



SANファイルシステム
ボトルネック：サーバ
限定された
スケーラビリティ



パラレルファイル
システム
ボトルネック：MD
サーバでの処理

マーケットの動向

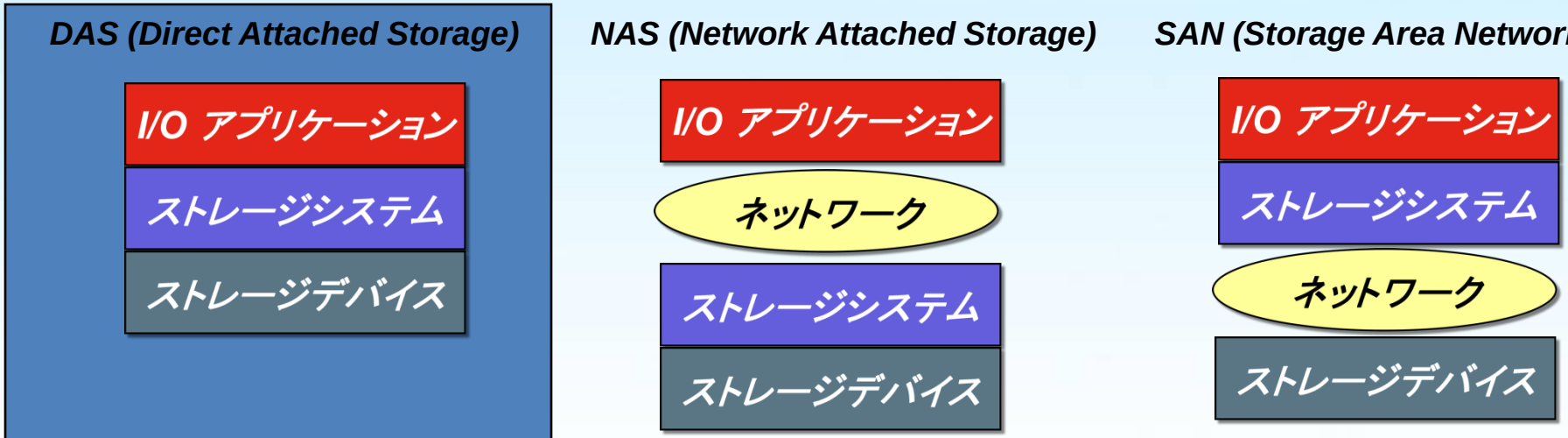
- **SANファイルシステム**
  - シングルネームスペース
  - 高いスループットI/O
  - スケーラビリティは限定的
  - 高価なSANシステムが必要
    - RedHAT GFS
    - SGI CXFS
    - IBM GPFS

- **パラレルファイルシステム（SAN)**
  - 高いスケーラビリティ
  - 高価なSANのインフラが必須
  - 複雑なSANのマネージメントが必要
    - LUN、ファイルシステム、ボリューム
    - Lustre （HPやDataDirect)
    - IBM Storage Tank

- **NAS/NFS**
  - シンプルなシステムマネージメント
  - 限定的な性能
  - スケーラビリティの限界
    - NetApp
    - EMC
    - SUN FC Array

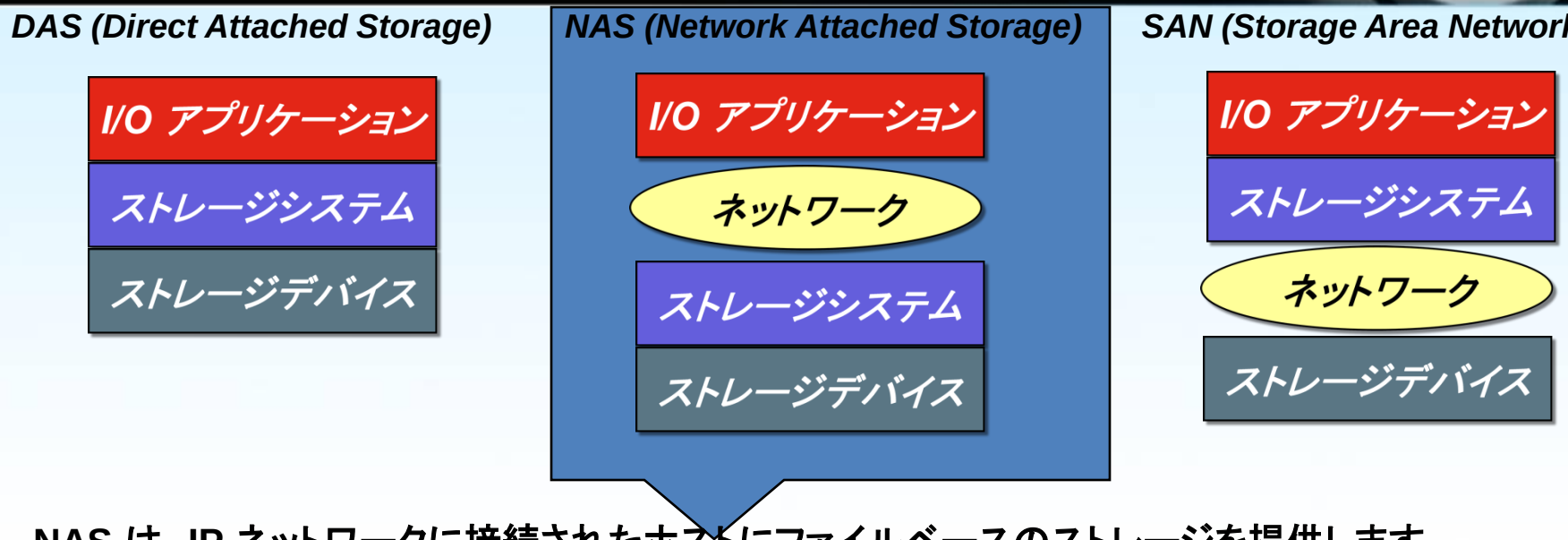
- **クラスタNAS**
  - 高いスケーラビリティ
  - 高いスループット
    - 高いバンド幅の実現は困難
  - 容易な管理
    - ロードバランスの問題

ストレージアーキテクチャ



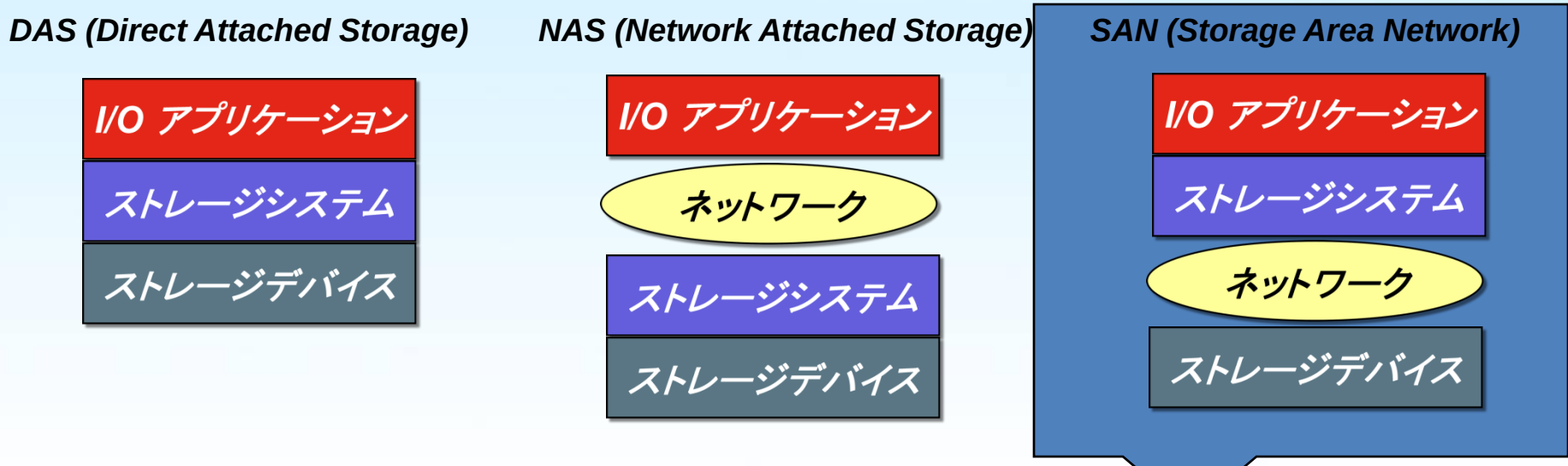
- DAS は、ホストマシンに直接接続されたブロックベースのストレージデバイス (SCSI パラレルバスに接続されたディスクドライブなど) で構成される
- DAS は高いパフォーマンスが求められるアプリケーションで広く使用されていますが、サーバー間でのデータ共有に制約があります。
- DAS の代表例には、小規模なデータベースやファイルサーバーがあります。

ストレージアーキテクチャ



- NAS は、IP ネットワークに接続されたホストにファイルベースのストレージを提供します。
  - NAS はホストからファイルサーバーへファイルシステムを完全にオフロードします。
- NAS アーキテクチャーでは、複数のホストがファイルを共有できます。
  - CIFS (Common Internet File System) や NFS (Network File System) など、ファイル要求に関するインターフェイスが標準化されているため、異なるプラットフォーム間でもファイルを共有できます。
  - NAS は異種プラットフォーム間でのストレージ共有が求められる用途で多く採用されます。
- NAS の代表例には、エンタープライズにおける多数の Web サーバー、ファイルとして保存された HTML コンテンツへのアクセス、多数のワークステーションで構成された部門ネットワークがあります。

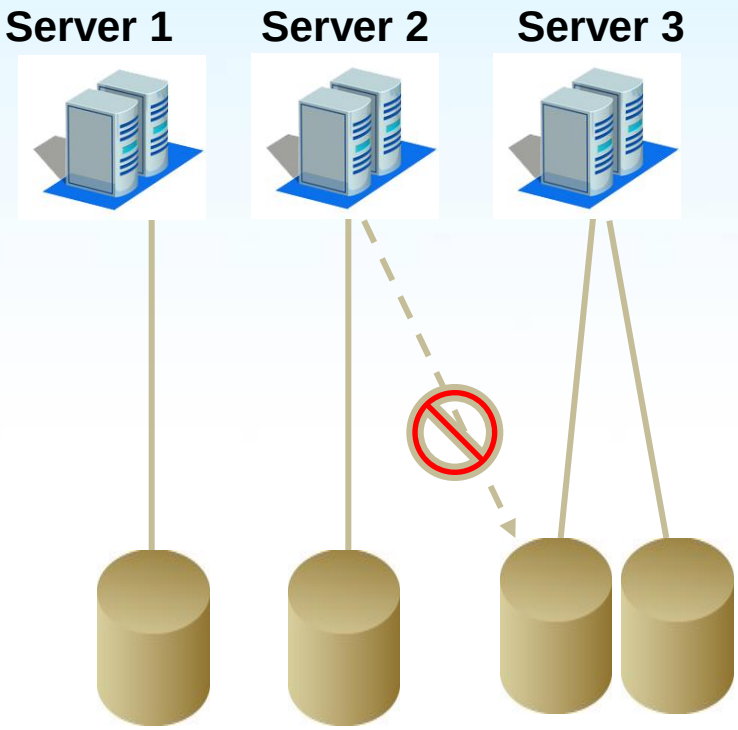
現在のストレージアーキテクチャ



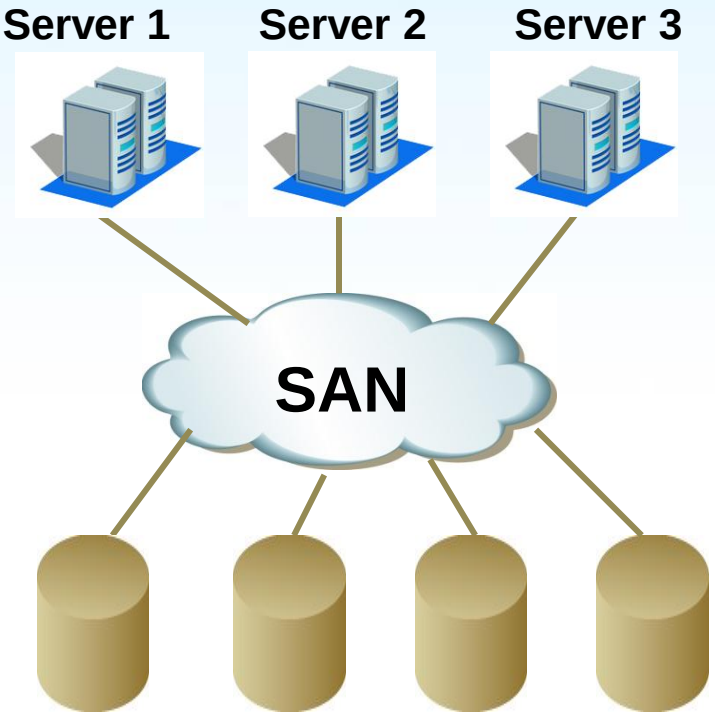
- SAN は、DAS のバスベースのアーキテクチャーをスイッチド・ファブリックに置き換えたものです。
- SAN ではホストとデバイスがいずれもファブリックに接続され、スケーラブルなパフォーマンスと容量を実現するとともに、複数のホスト間でデバイスを共有することもできます。
- SAN のアーキテクチャーは、ストレージデバイスに極めてスケーラブルなパフォーマンスが要求されるアプリケーションで広く採用されています。
- SAN の代表例には、ワークステーション・クラスター上で稼働する分散型データベースがあります。

現在のストレージアーキテクチャ

Direct Attach
サーバ間でデータ共有は出来ない

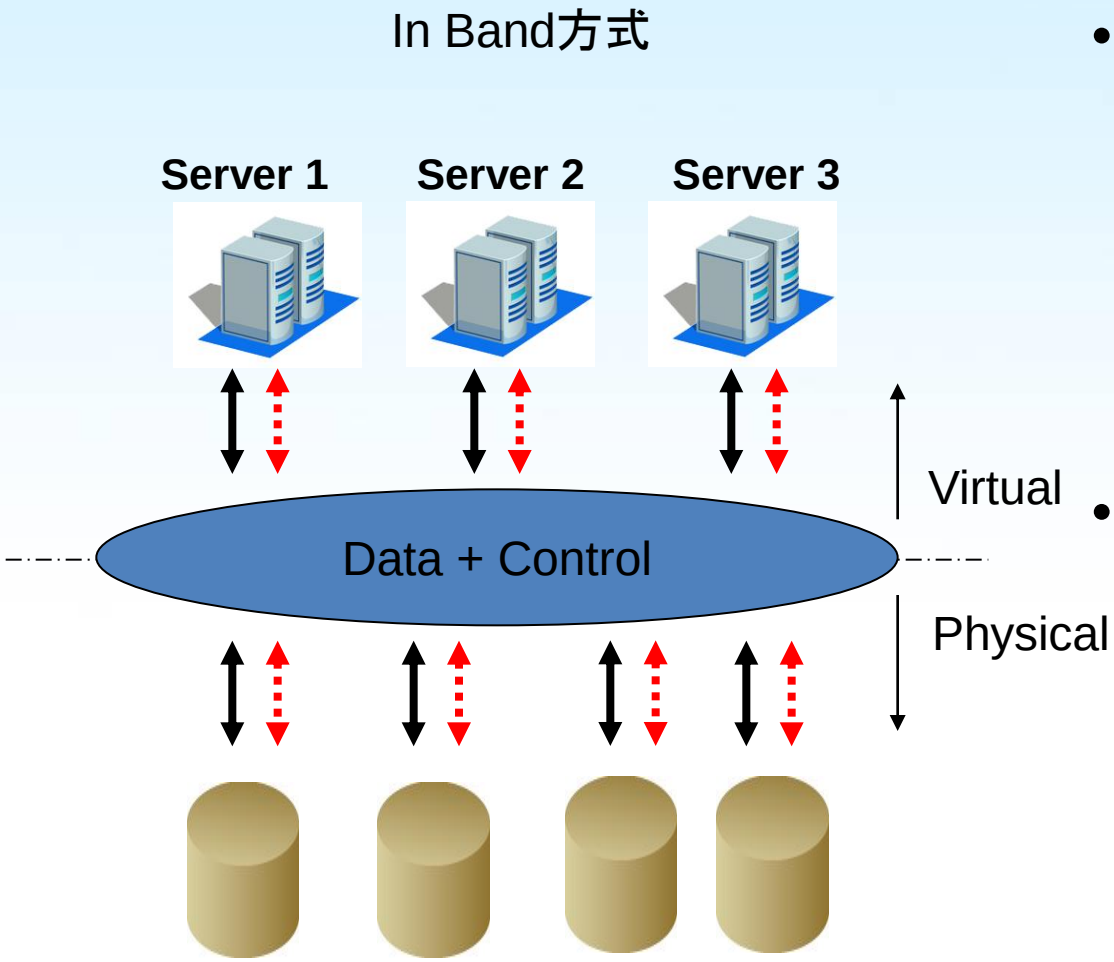


Storage Area Networking
SANはデータ・パスの共有をしている
に過ぎない



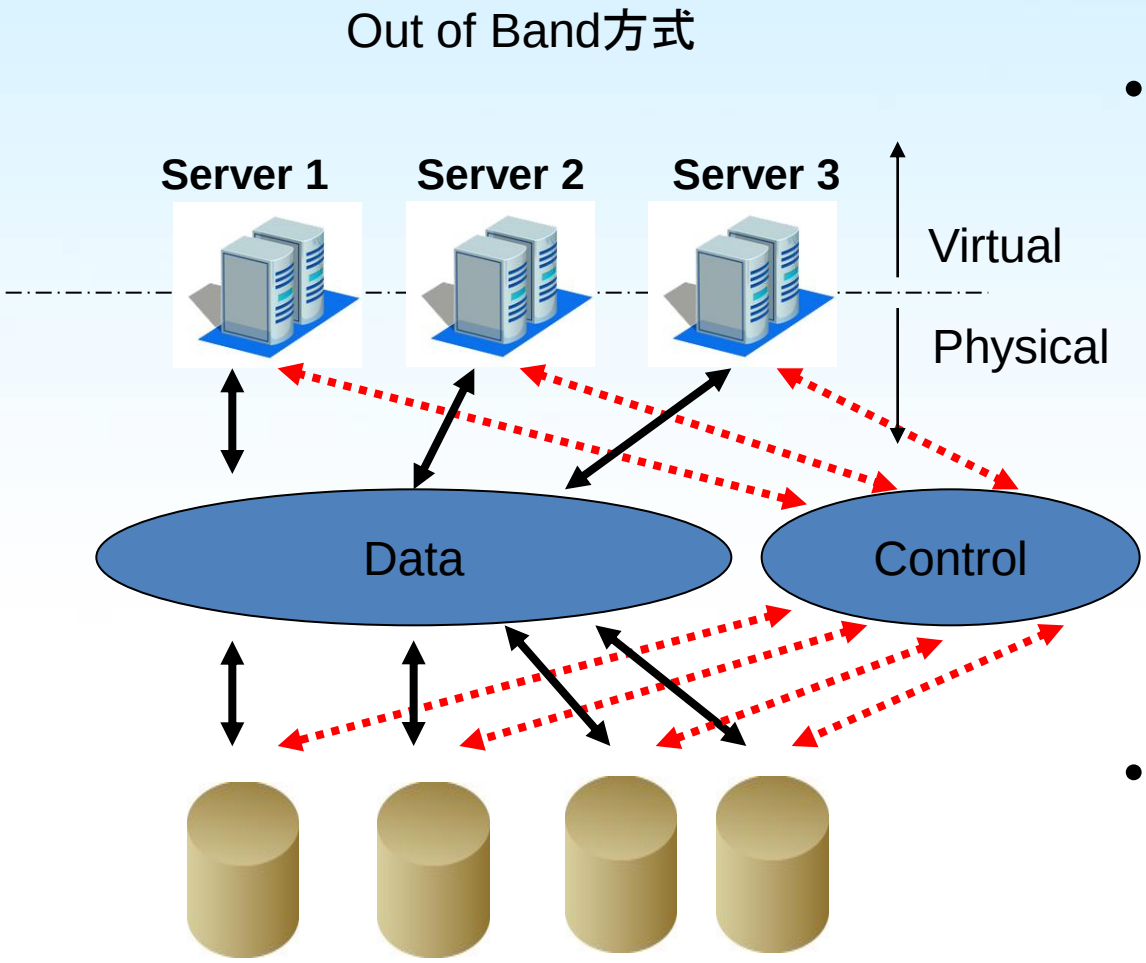
実質的には、個別利用と同じ

アーキテクチャ:In Band方式



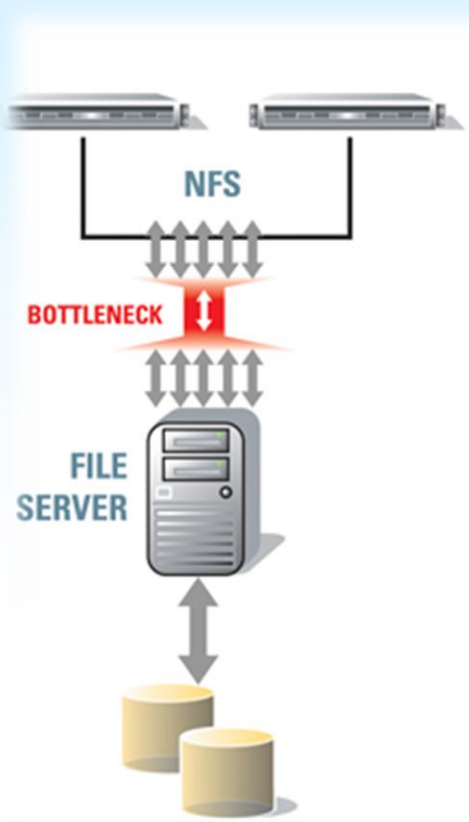
- 利点
  - 複数のストレージ装置を利用可能
  - プール化によるリソースの活用
  - 機能拡張の共通化
- 欠点
  - 高速化のためにキャッシュ利用が必須
  - キャッシュミス発生時のアクセスの集中

アーキテクチャ：Out of Band方式



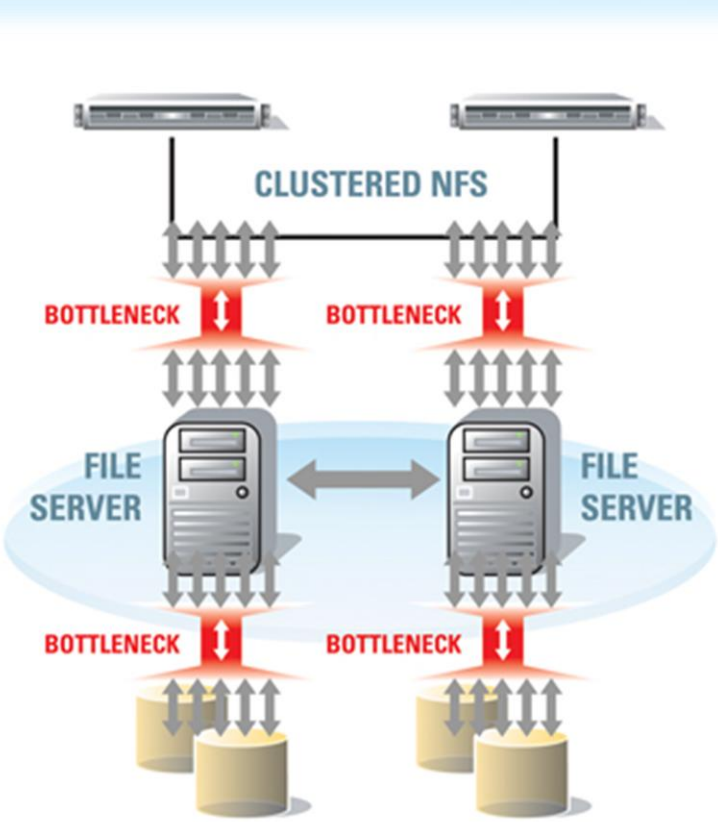
- 利点
  - サーバが直接ストレージにアクセス可能
  - プール化によるリソースの有効活用
  - データパスとコントロールパスの選択の柔軟性
- 欠点
  - サーバの負担
  - 制御パスのアクセス

ストレージアーキテクチャ
パラレルストレージの価値の検証



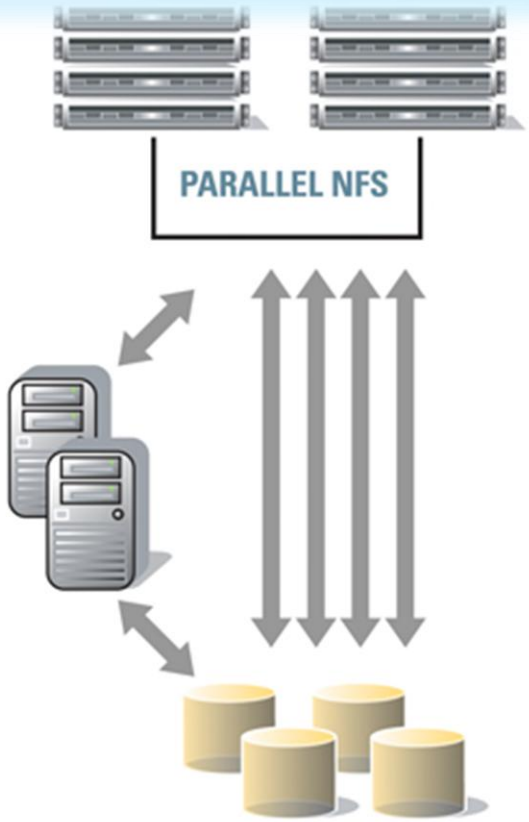
NAS

Network Attached Storage
シリアルI/Oがボトルネック



CLUSTERED STORAGE

複数のNASを統合的に運用管理
個々のNASサーバでのシリアルI/O
がボトルネック



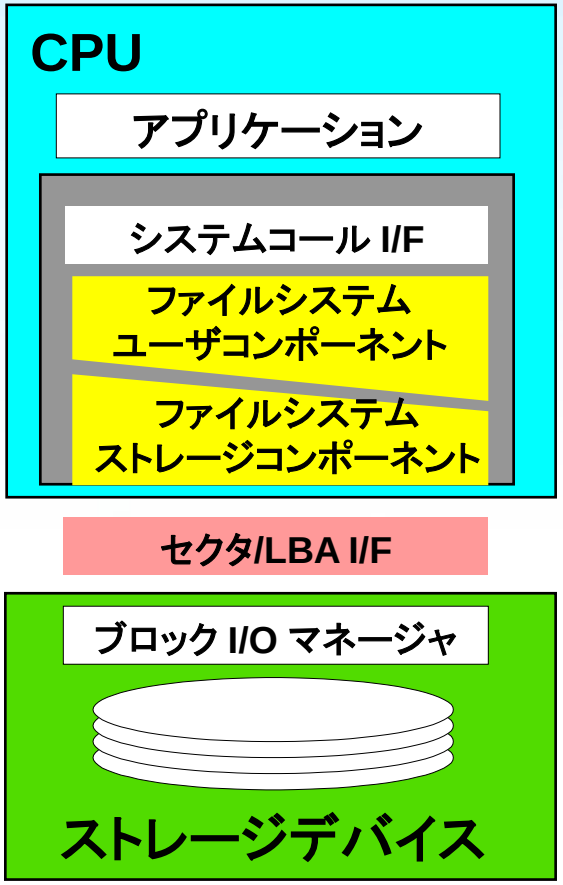
PARALLEL STORAGE

ファイルサーバを経由しないデータ
転送パス
シリアルI/Oのボトルネックの解消と
容易なシステム全体の運用管理

スケーラブルシステムズ株式会社

ファイルシステム
ストレージプロトコルスタック

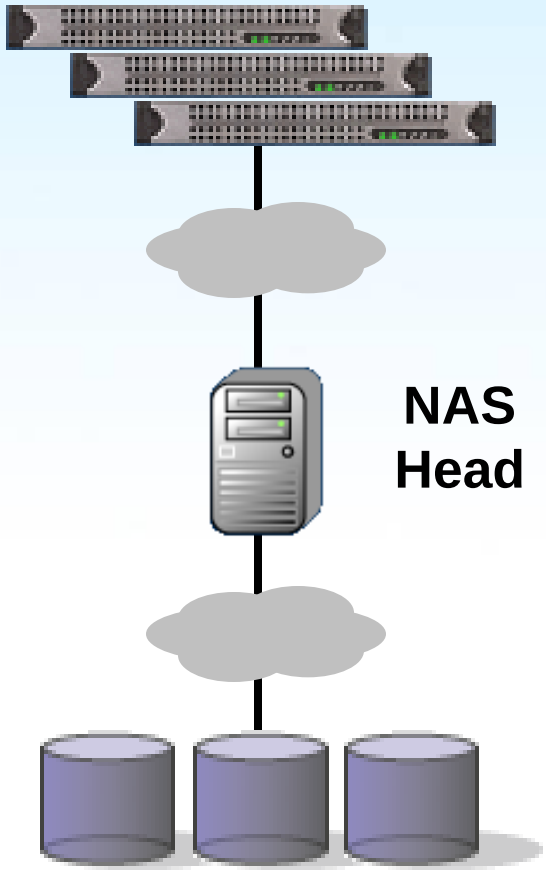
- OS:ファイルシステムのポリシーチェック
  - ユーザ認証
  - アクセスパーミッション(read, write, quota)
  - ファイルレベル属性の管理
  - 物理的なサイズ、論理的なレコード長、タイムスタンプ（最終アクセス、最終修正日時）
- ファイルシステム：データをブロックに転送
  - OSがストレージデバイス上での処理を行う
  - レイアウト、ストレージに対するアクセス要求の処理
  - データのプリフェッチやバッファキャッシュの処理
  - ディスクは、キャッシュやプリフェッチの機能を持つ（非常に限定的な機能）
  - 物理的なレイアウトに依存
  - アプリケーションの動作に関する情報などは活用出来ない



LBAとは、ハードディスク内のすべてのセクタに対してゼロからの通し番号を振ることで、その通し番号によってセクタを指定する方式のことである。論理ブロックアドレスと呼ばれることもある。

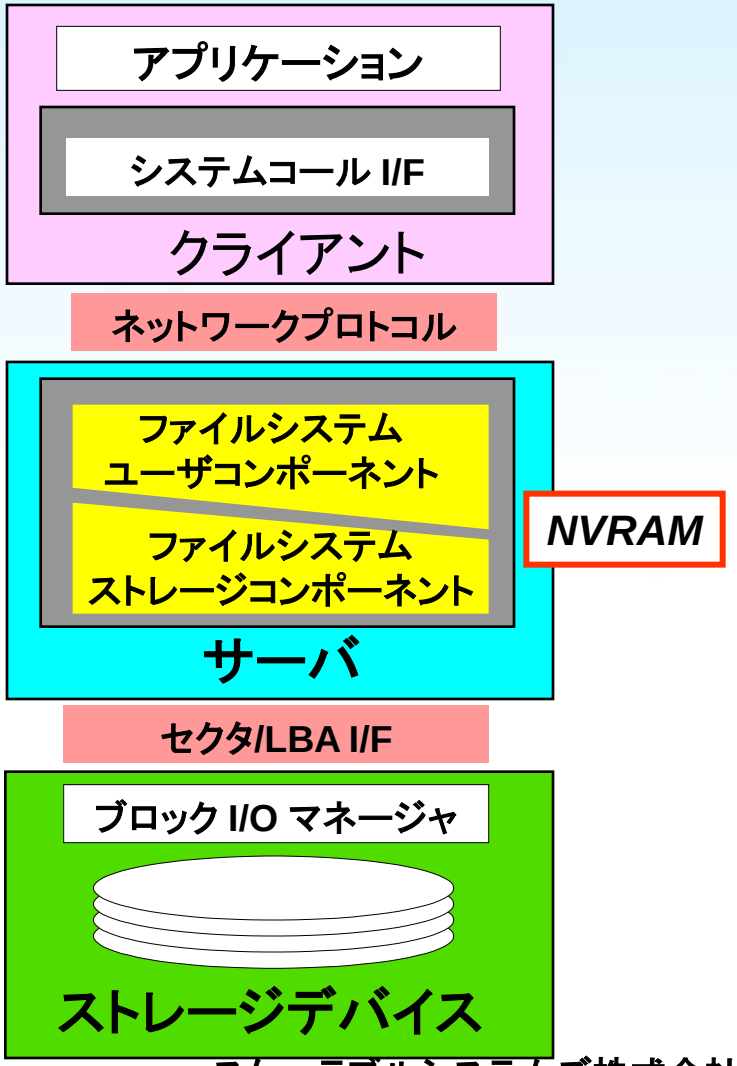
Network-Attached Storage：NAS
ネットワークアタッチトストレージ

- ファイルサーバは、ファイルレベルで、ストレージをエクスポート
  - NFS/CIFは広く利用されている
  - NFSは、唯一の業界標準のファイルシステム
- スケーラビリティは、サーバのハードウエアによって制限される
  - 中規模のクライアント数（数十から100程度）
  - 中規模のストレージ容量（数テラバイト）
- データ転送能力の限界以内であれば、非常に優れたモデル
  - 複数のストレージは個別に配置
  - サーバの能力によって、ファイルアクセスのバンド幅が制限される
- NetApp (ONTAP 7.x), Sun/HP/IBM NAS, SnapServer, EMC Celerra, whitebox Linux



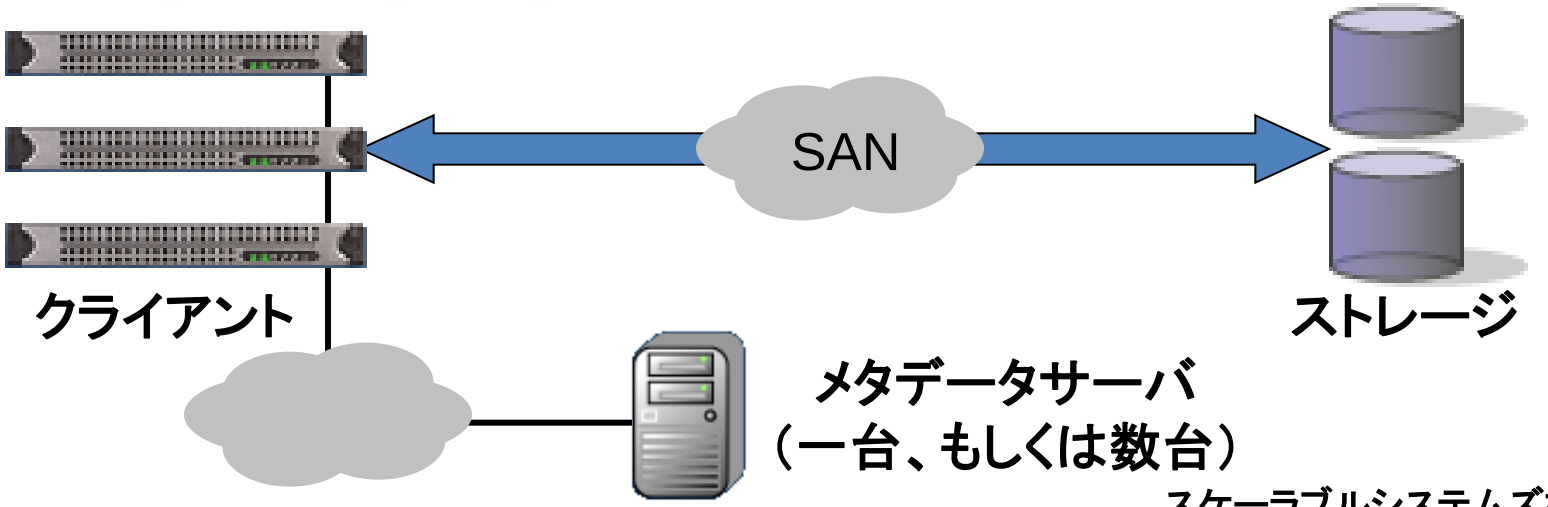
Network-Attached Storage：NAS
NASファイルシステム

- ハイレベルのインターフェイスによって、ファイルシステムの違いは隠蔽される
  - クライアントのGenericレイアは、VFS (vnode) の導入によって修正される
    - どのようなサーバのファイルシステムもエクスポート可能
  - NVRAM(Non-volatile write buffer)は、性能面からも、また障害時対応のためにも必要
- NASファイルシステムでは、データとメタデータは、NASサーバを介してストレージデバイスに転送される



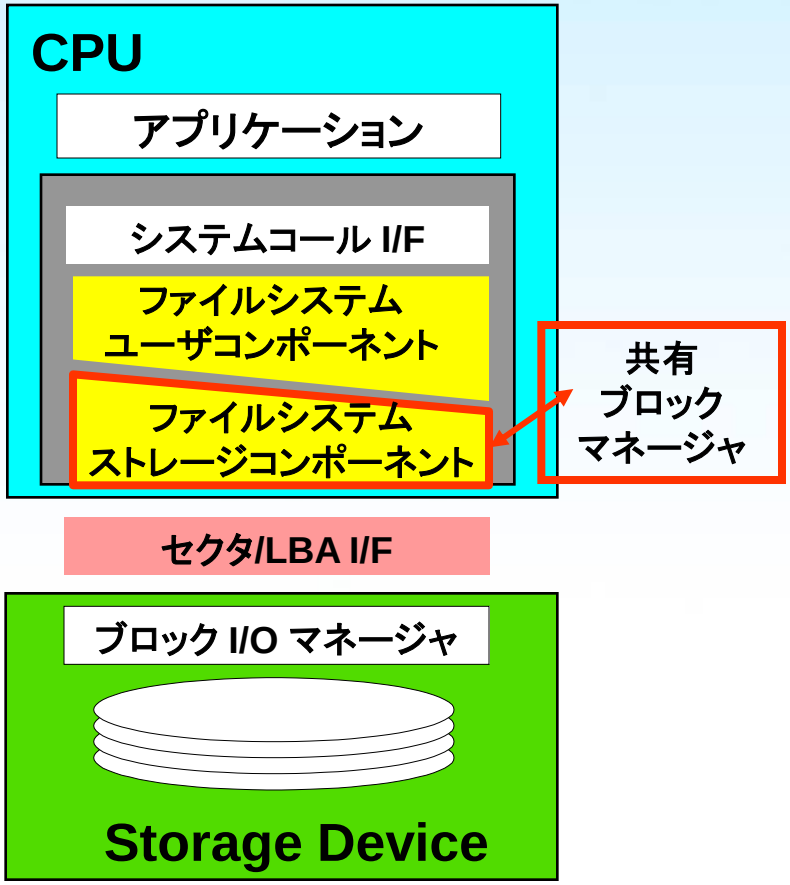
Storage Area Network：SAN
ストレージエリアネットワーク

- ホストシステムのストレージと同様に管理運用とプロビジョニングが可能
  - ブロックデバイス (JBOD 又は RAID) は、iSCSI やFC ネットワークで接続
  - 接続速度/RAIDの速度が性能を決定する
- 共有ファイルシステム構築のために特別なシステムが必要
  - スケーラビリティは、メタデータサーバ上のブロック管理によって制限される（一般には、32ノード程度）
  - SANファイルシステムを‘ファイルヘッド’として、再エクスポートすることで、NASとしても利用可能
- 非対称構成（下図）と対称構成が可能



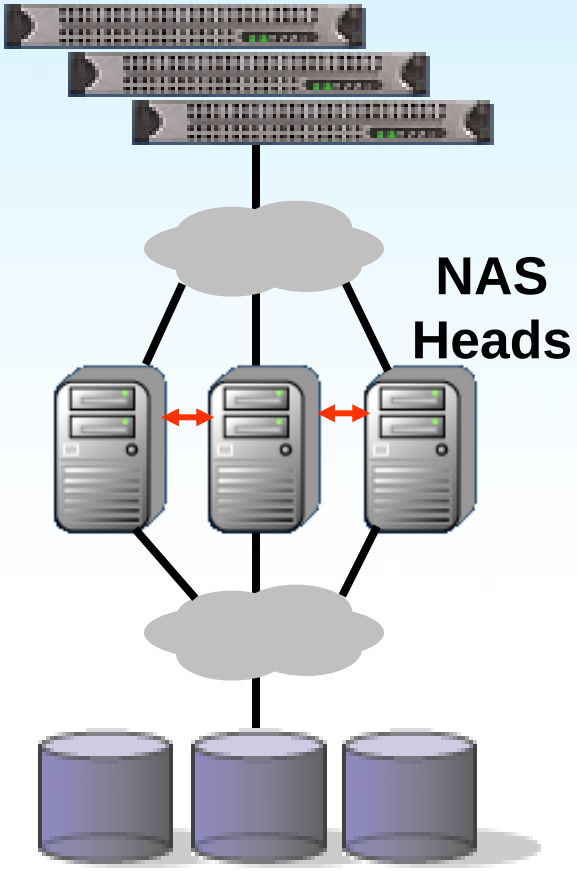
Storage Area Network：SAN
SAN ファイルシステム

- 共有 SAN ファイルシステム
- ストレージレイヤーは、ファイルシステムを共有するホスト間で、ブロックレベルで同期を取る必要がある。
- ブロックの所有権とアロケーションに関する限定されたインターフェイスは、既存のファイルシステムの‘容易な’変更を可能とする。
  - 例えば XFS -> cXFS, Terrascale
- 低レベルのインターフェイスは、分散システムでは、より大きなオーバヘッドを引き起こす。
  - write() システムコールは、幾つかのブロックレベルのIOオペレーションを含む
  - 様々な構成ブロック(inodes, indirect blocks, data, allocation bitmaps)
- 数億ブロックの管理
  - 500GBデバイスでは、10億個の512-byte セクターを管理し、50TBでは、1000億のセクターの管理が必要



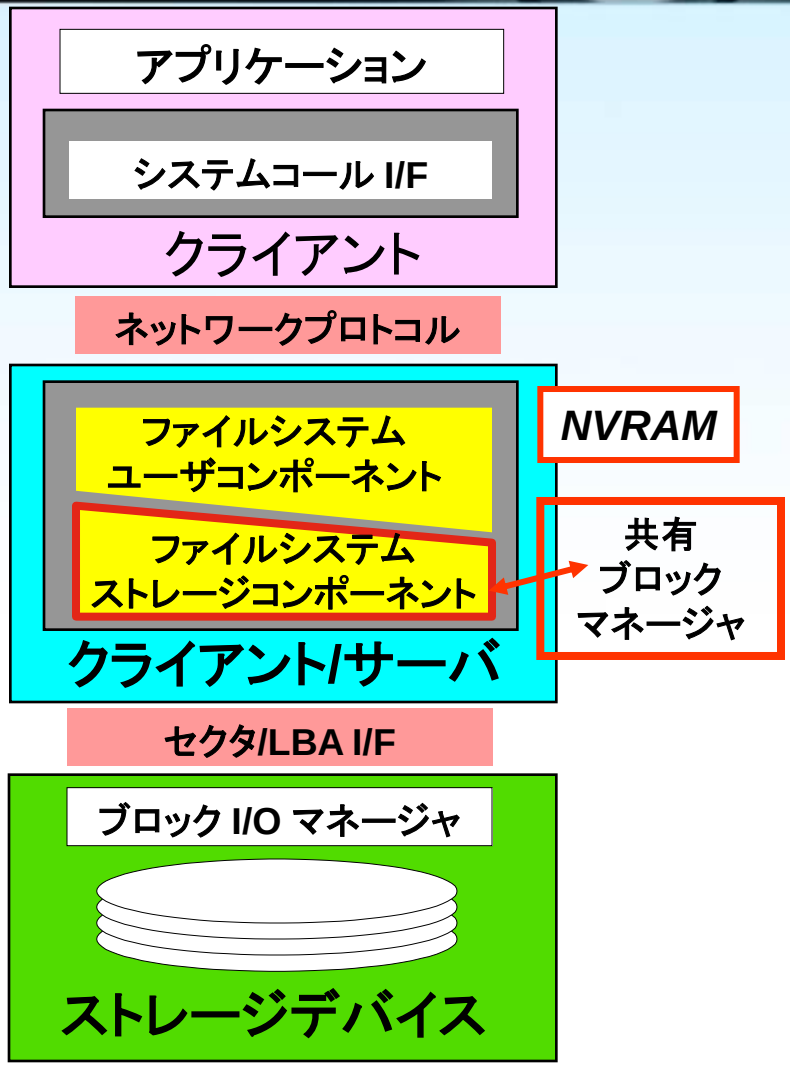
クラスタNAS

- シングルヘッドのNASよりもよりスケーラブル
  - 複数のNASヘッドがバックエンドのストレージを共有
  - 個々のNASヘッドの性能がボトルネックとなる。また、NASヘッドを追加することでコストは高くなる。
- 2つの主要アーキテクチャ
  - データを‘オーナー’ヘッドへ転送
  - クラスタSANファイルシステムからNASとしてエクスポート
- NFSは動的なロードバランス機構を提供出来ない
  - クライアントは、いずれかのノードに恒久的にマウントされる
- GPFS、Isilon OneFS、IBRIX、Polyserve、NetApp GX、BlueArc、Exanet ExaStore、ONStor、Pillar Data、IBM/Transarc AFS、IBM DFS



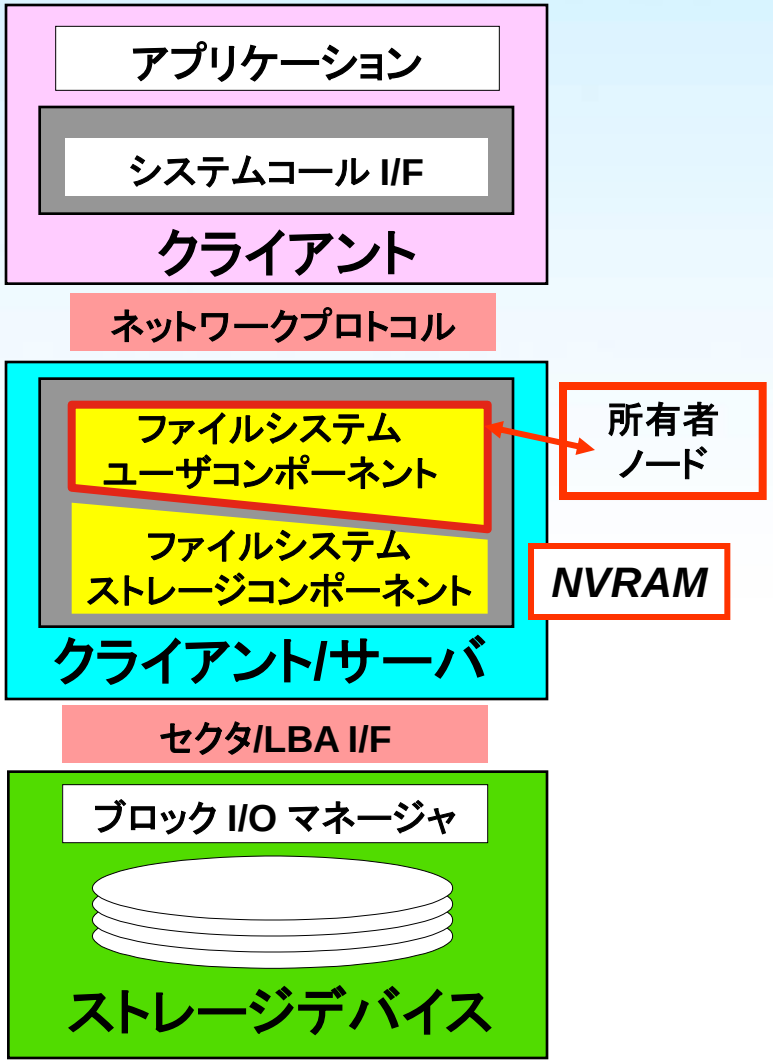
クラスタNAS+SANファイルシステム

- ハイブリッドモデル
- SANファイルシステムをクライアントにネットワークプロトコルを介し再エクスポート
- SANファイルシステムが持つ、ブロックアロケーションとwriteオペレーションの制限を持つ
- GPFS は、このモデルであり、非標準のネットワークプロトコルを利用している

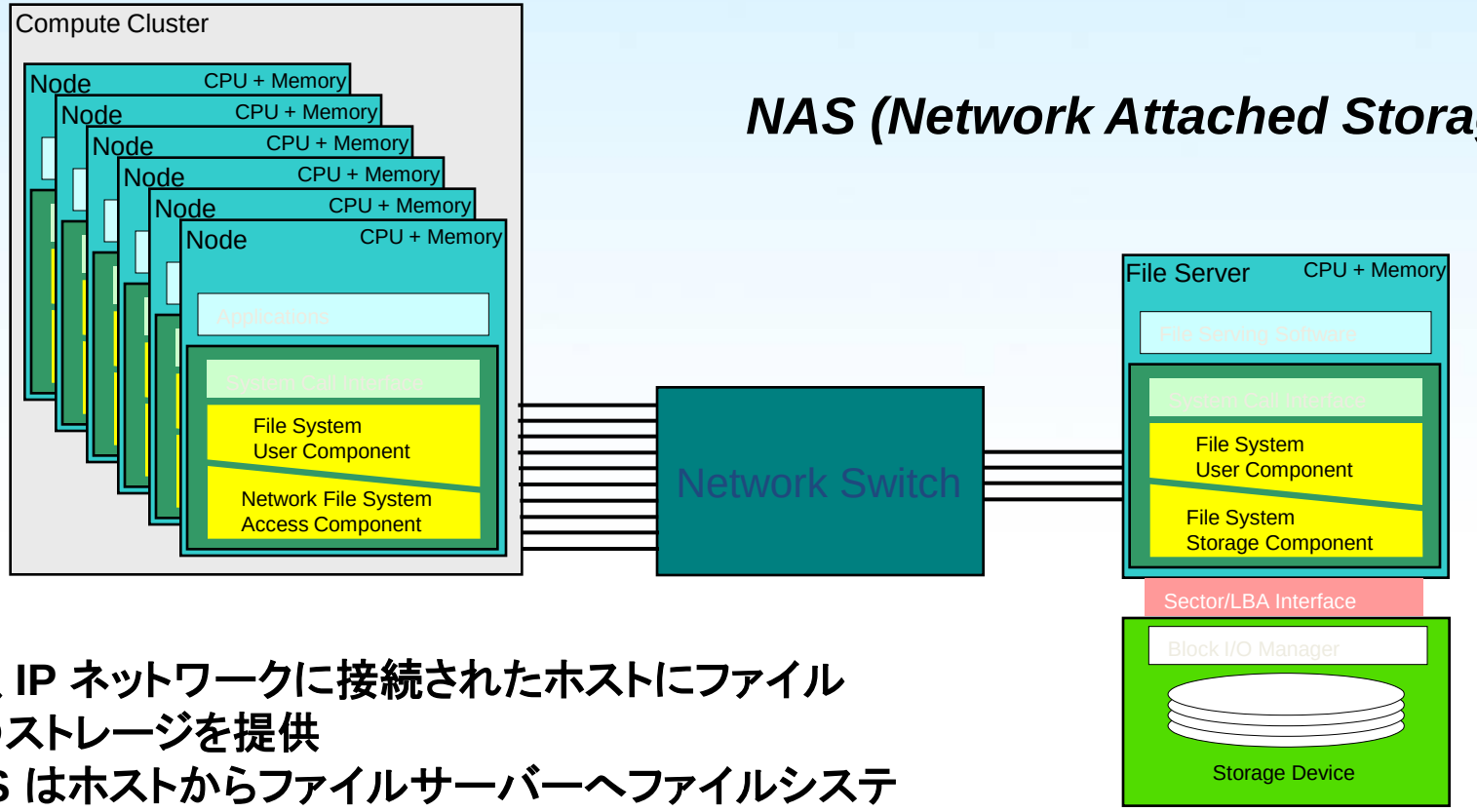


クラスタNAS+SANファイルシステム

- ノード自身がストレージのサブセットを持つ（ブロックアロケーションのために）
- NASヘッドは、大容量のキャッシュを持ち、自身のキャッシュミスの場合には、オペレーションを転送する
- キャッシュミスが発生する場合には、通常のファイルサーバと同じボトルネックが発生する
- NetApp-GX, Isilon, IBRIX



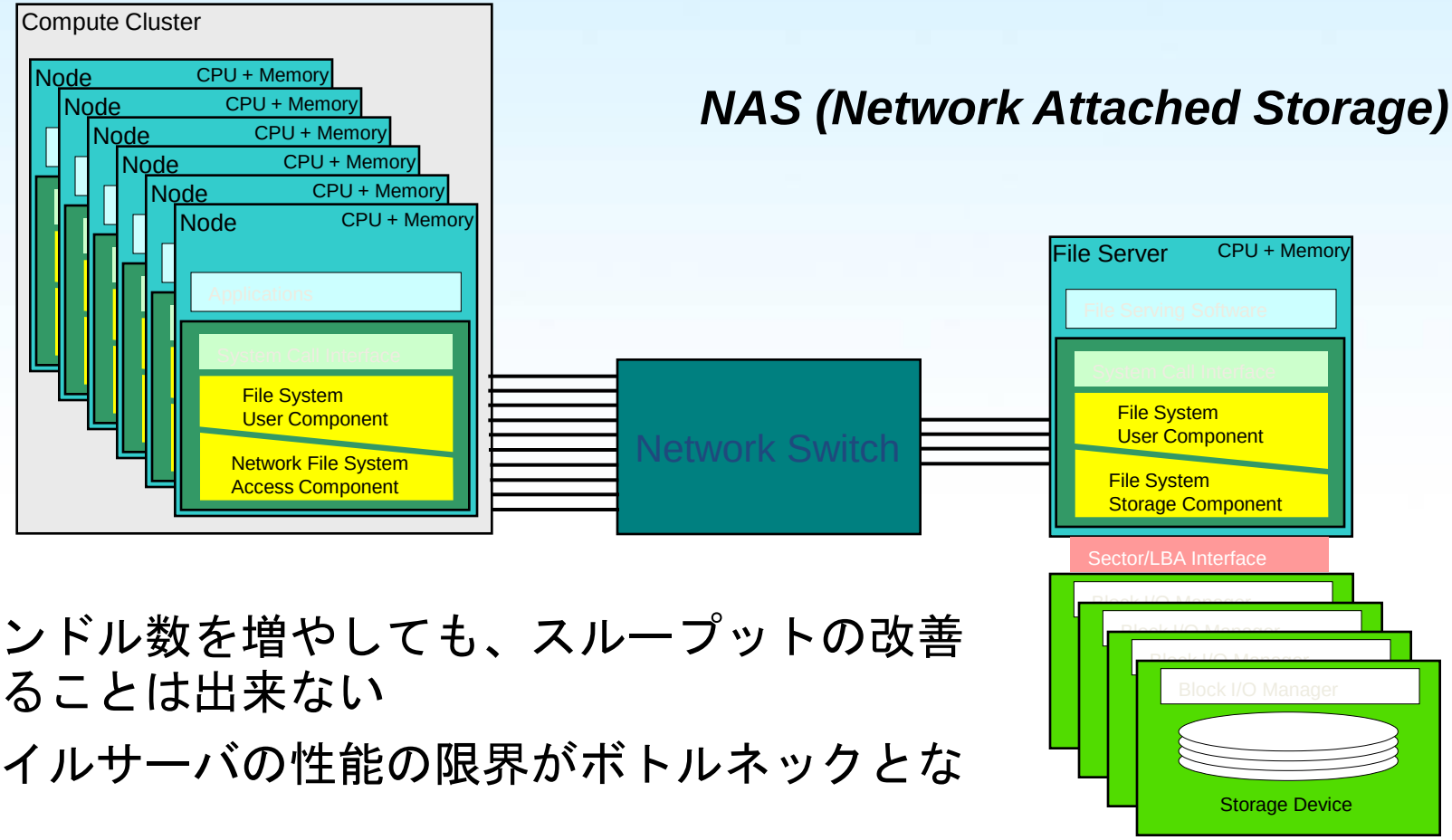
古典的なファイルサーバ



NAS (Network Attached Storage)

- **NAS は、IP ネットワークに接続されたホストにファイルベースのストレージを提供**
  - **NAS はホストからファイルサーバーへファイルシステムを完全にオフロード**
- **NAS アーキテクチャーでは、複数のホストがファイルを共有可能**

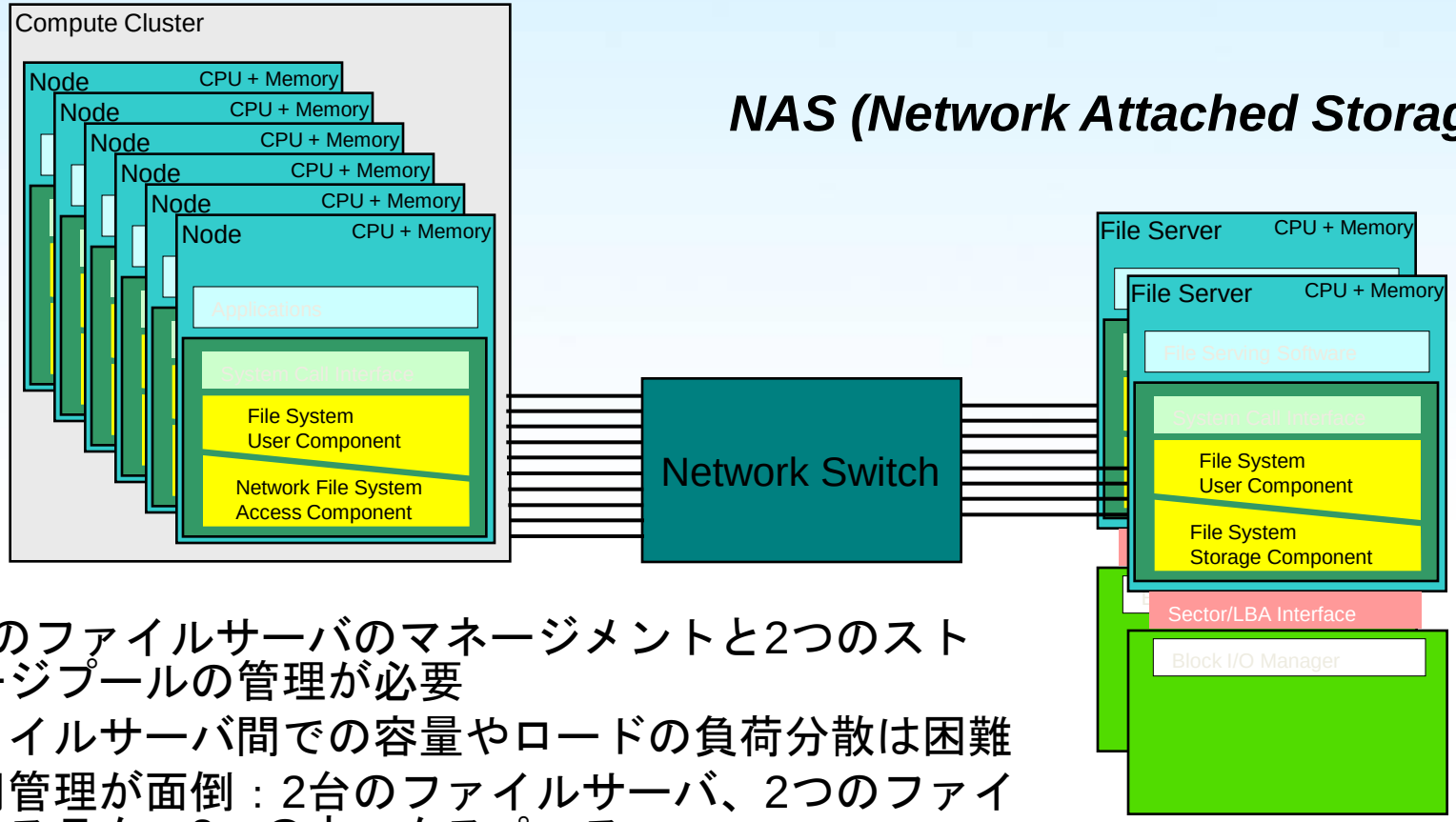
スループットの改善
ディスクのスピンドルを増やす?



NAS (Network Attached Storage)

- スピンドル数を増やしても、スループットの改善を図ることは出来ない
- ファイルサーバの性能の限界がボトルネックとなる

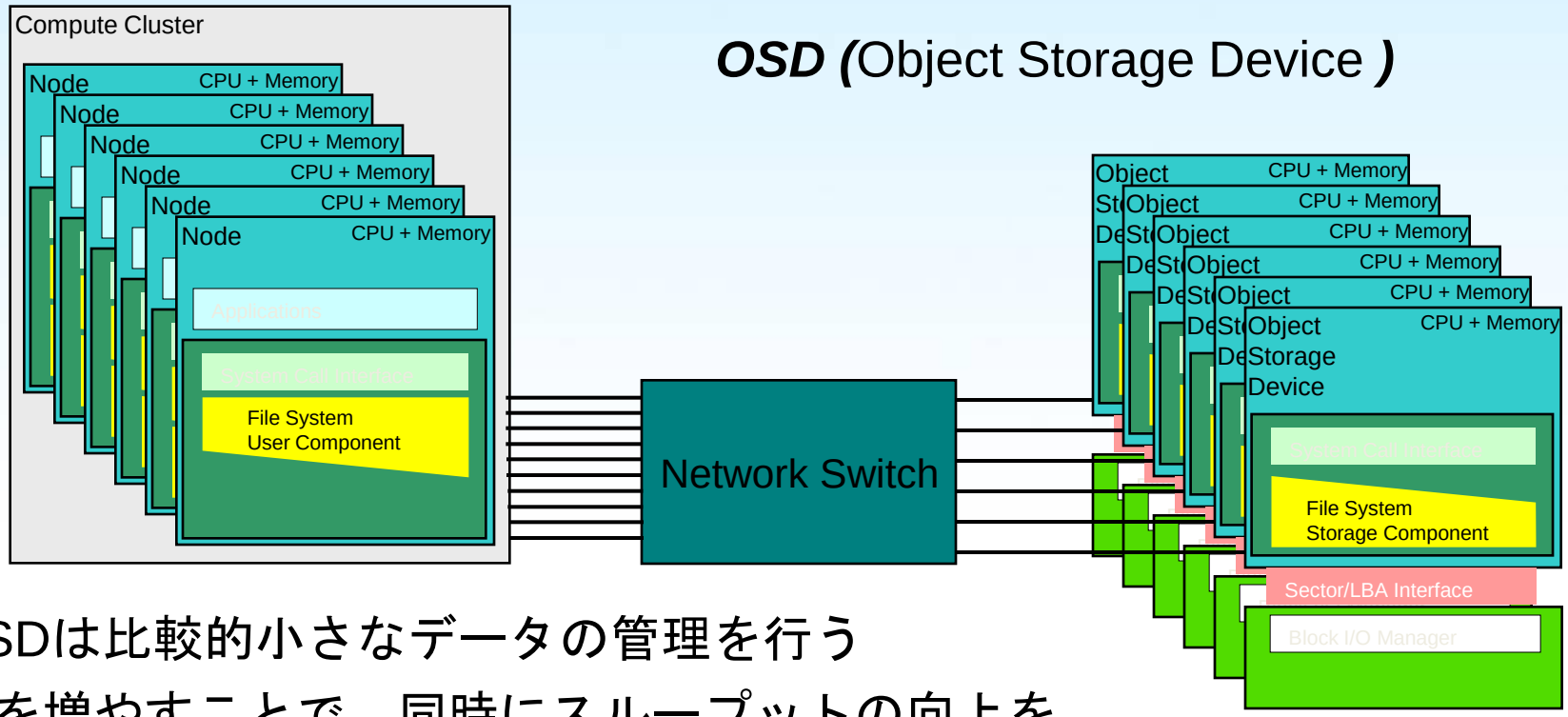
スループットの改善
ファイルサーバの増設?



NAS (Network Attached Storage)

- 2つのファイルサーバのマネージメントと2つのストレージプールの管理が必要
- ファイルサーバ間での容量やロードの負荷分散は困難
- 運用管理が面倒：2台のファイルサーバ、2つのファイルシステム、2つのネームスペース

オブジェクトストレージデバイス



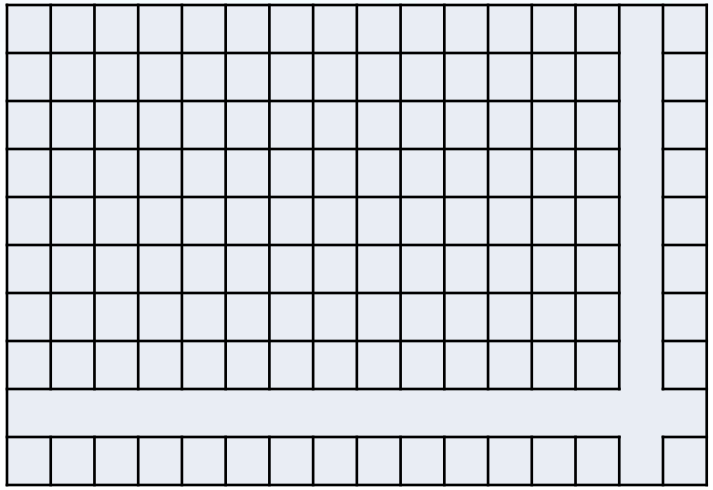
- 各OSDは比較的小さなデータの管理を行う
- 容量を増やすことで、同時にスループットの向上を図ることが可能となる
- シングルネームスペースの提供による柔軟で容易なシステムの運用管理

ブロック・ベースと オブジェクトベースの違い

個々のブロックと通信するプロトコルを利用（SCSI,ATA)

ブロックサイズは固定

データとメタデータの両方が含まれるブロックのコレクション

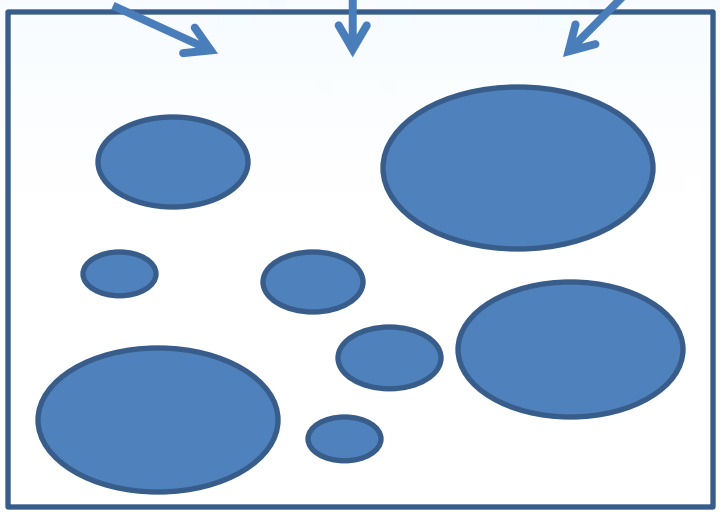


ブロックベースストレージシステム

個々のオブジェクトと通信するプロトコルを利用（OSDなど)

オブジェクトサイズは固定

オブジェクトとそのオブジェクトに関するメタデータ



オブジェクトベースストレージシステム

オブジェクト・ベースの利点

- マルチテナント性を持ち、高度なセキュリティの実装が可能
- 極めてスケーラブルなシステムの構築が可能
  - ファイルシステムを部分的にシステムから取り除き、その部分をストレージシステムに組み込む
  - ファイルシステムの構成部分を複数のエンドポイントに分散し、作業負荷を低減することでストレージシステムの規模を大幅に拡大
  - メタデータ参照を分散させることで、検索機能などの強化も可能

参考資料：次世代 Linux ファイルシステム: NiLFS(2) と exofs
http://www.ibm.com/developerworks/jp/linux/library/l-nilfs-exofs/

ファイルシステム

Shannon Filing Cabinet, Schlicht & Field Co.,
Rochester, NY, 1886
http://www.officemuseum.com

Unified Heterogeneous File Systems

ファイルシステム

- 分散ファイルシステム
  - システムに直接接続していないファイルシステムの総称
  - NFSとCIFSが最も一般的な分散ファイルシステム
- グローバルファイルシステム
  - ネームスペースを参照
  - すべてのホストから、すべてのファイルのファイル名とパスが同じに見える
- SANファイルシステム
  - FC(Fibre Channel)ストレージへのホストからのアクセス
  - ブロックレベルでのメタデータのマネージメントが異なったSANデバイスへのアクセスでは必要
  - スケーラビリティは、メタデータの処理にも大きく依存する
  - 例：SGI CXFS、IBM GPFS、RedHat Sistina GFS　など

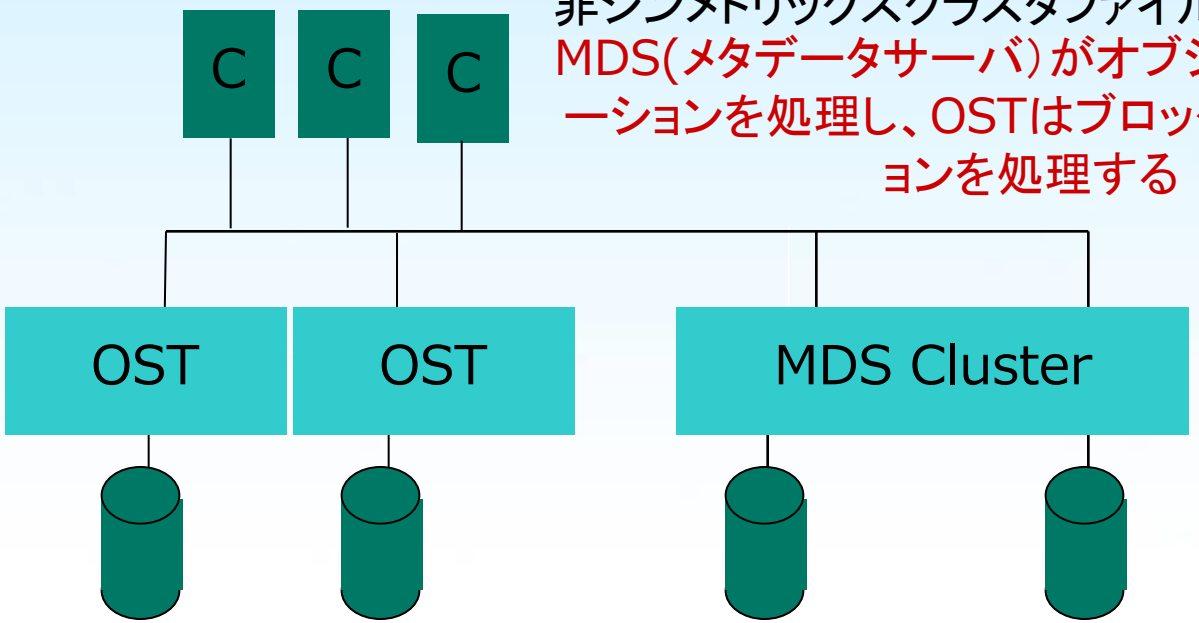
ファイルシステム

- シンメトリックスファイルシステム
  - クライアント上でメタデータ処理を行う
  - すべてのノードがディスク構成の情報を持つ必要がある
  - 問題点
    - クライアントの負荷（クライアント自身の処理と他のノードへのデータ配信）
    - 計算ジョブとメタデータ処理の双方の実行
  - 例：IBM GPFS、RedHat Sistina GFS、Vetitas CFS
- 非シンメトリックスファイルシステム
  - 一つ以上の専用のメタデータ処理用のサーバがファイルシステムとディスク構造の管理を行う
  - 例： SGI CXFS 、IBM SanFS、Panasas PanFS、Lustre

ファイルシステム

- クラスタファイルシステム
  - クライアントの高速のファイルアクセス処理を提供するために、ファイルサーバをクラスタとして、利用
- パラレルファイルシステム
  - アプリケーションの並列実行時のI/O処理をサポートできるファイルシステム
  - 全ノードから同一ファイルへの同時アクセスも可能
  - 例： SGI CXFS、IBM GPFS、RedHat Sistina GFS
- これらのファイルシステムの定義は、オーバラップする

Lustreファイルシステム

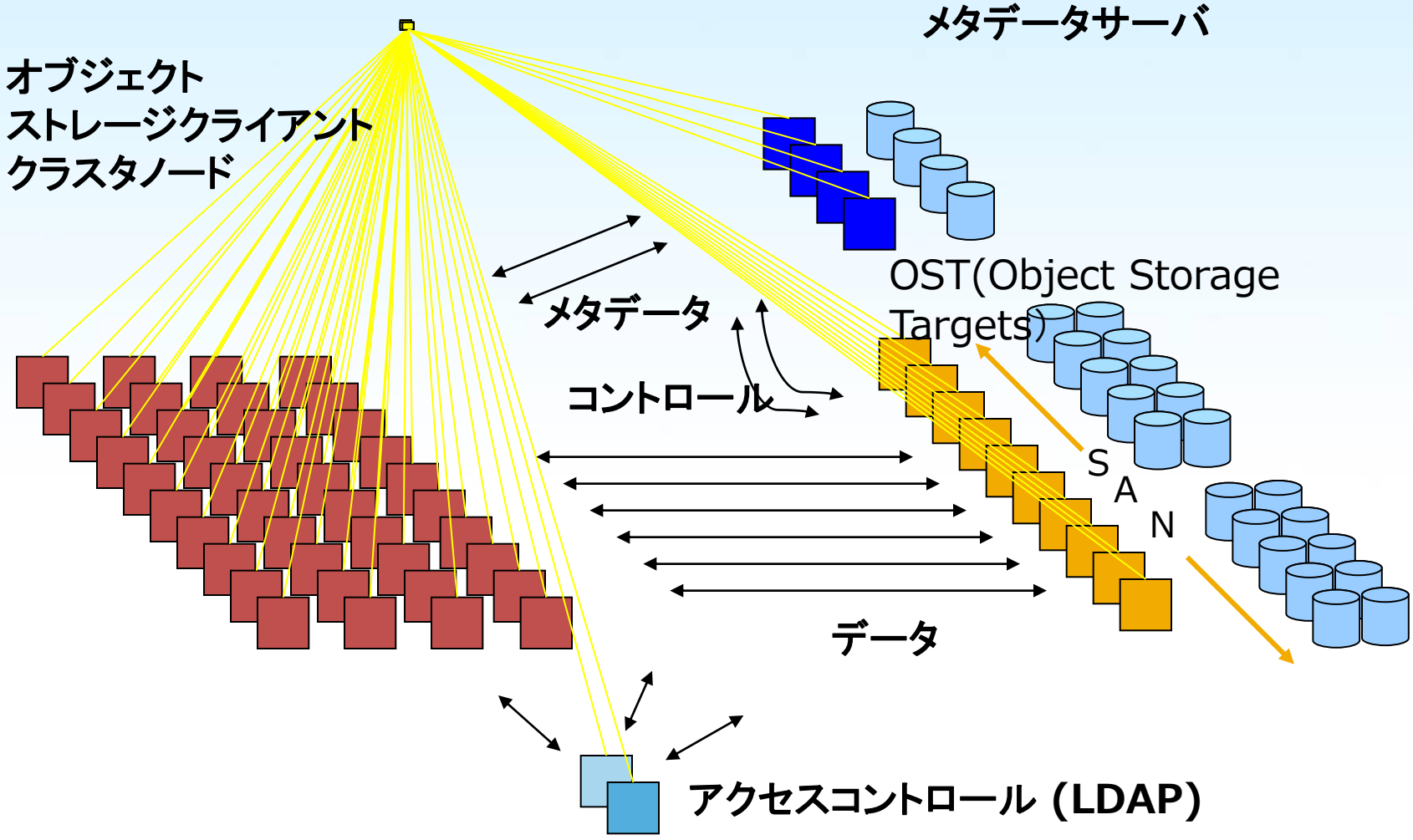


非シンメトリックスクラスタファイルシステム
MDS(メタデータサーバ)がオブジェクトのアロケーションを処理し、OSTはブロックのアロケーションを処理する

On deploying Lustre: "It's not like backing your car out of the driveway. Installing Lustre is more like launching the space shuttle, with pieces of foam falling off."*

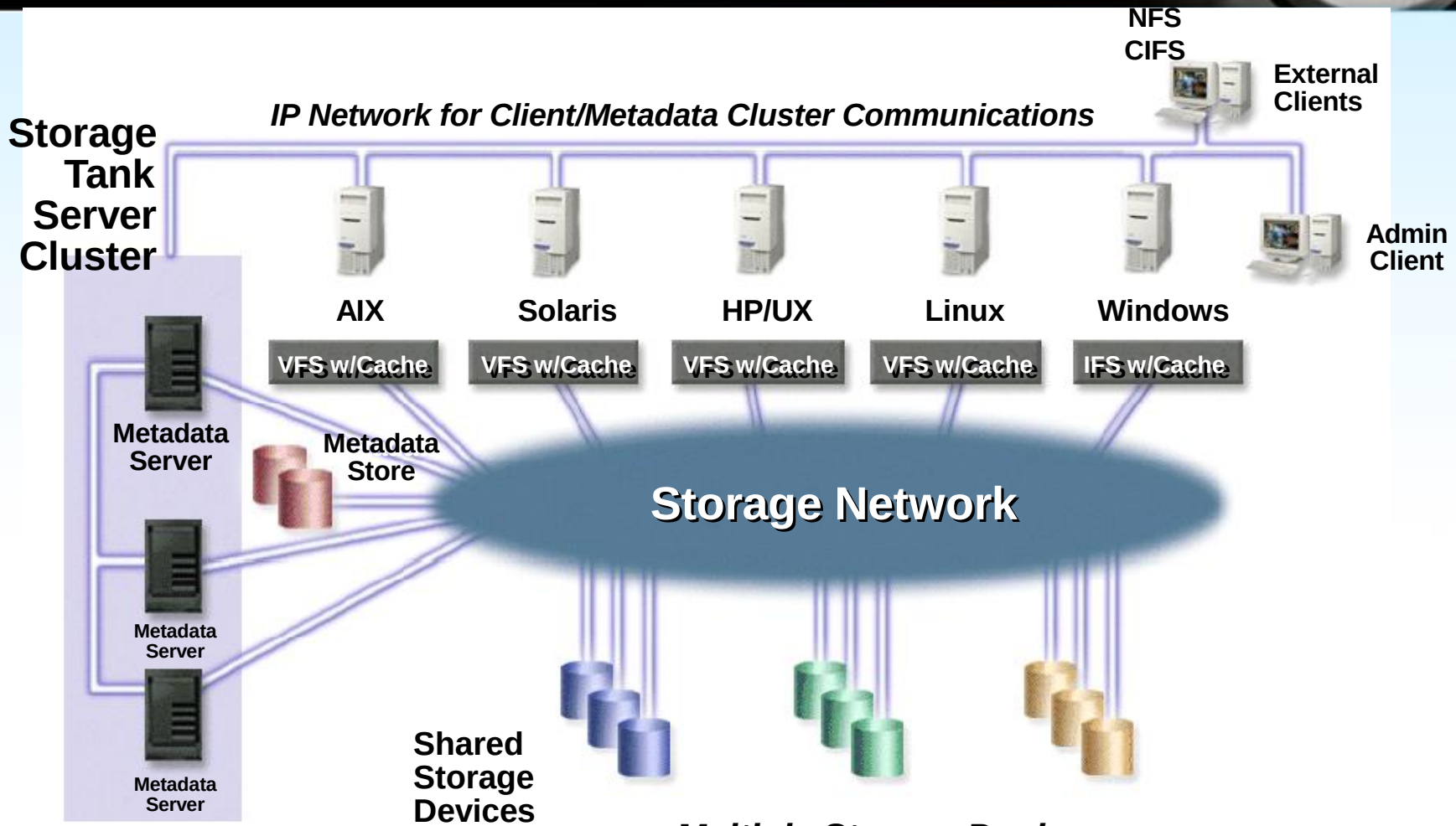
Quote from Peter Braam – CTO of CFS – at LLNL conference - 8/17/05

Lustreオブジェクトストレージモデル



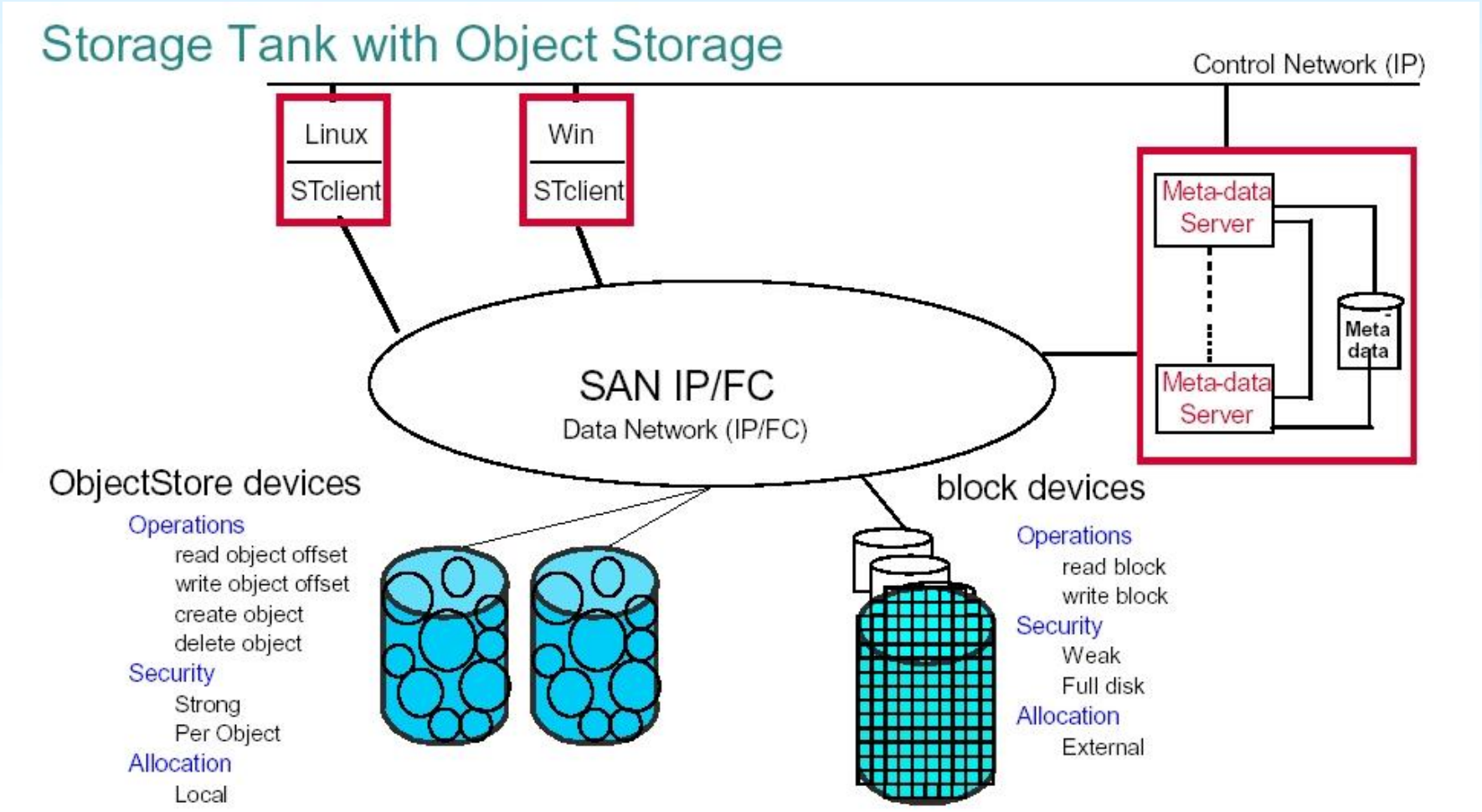
（source:HP)

IBM Storage Tank概要



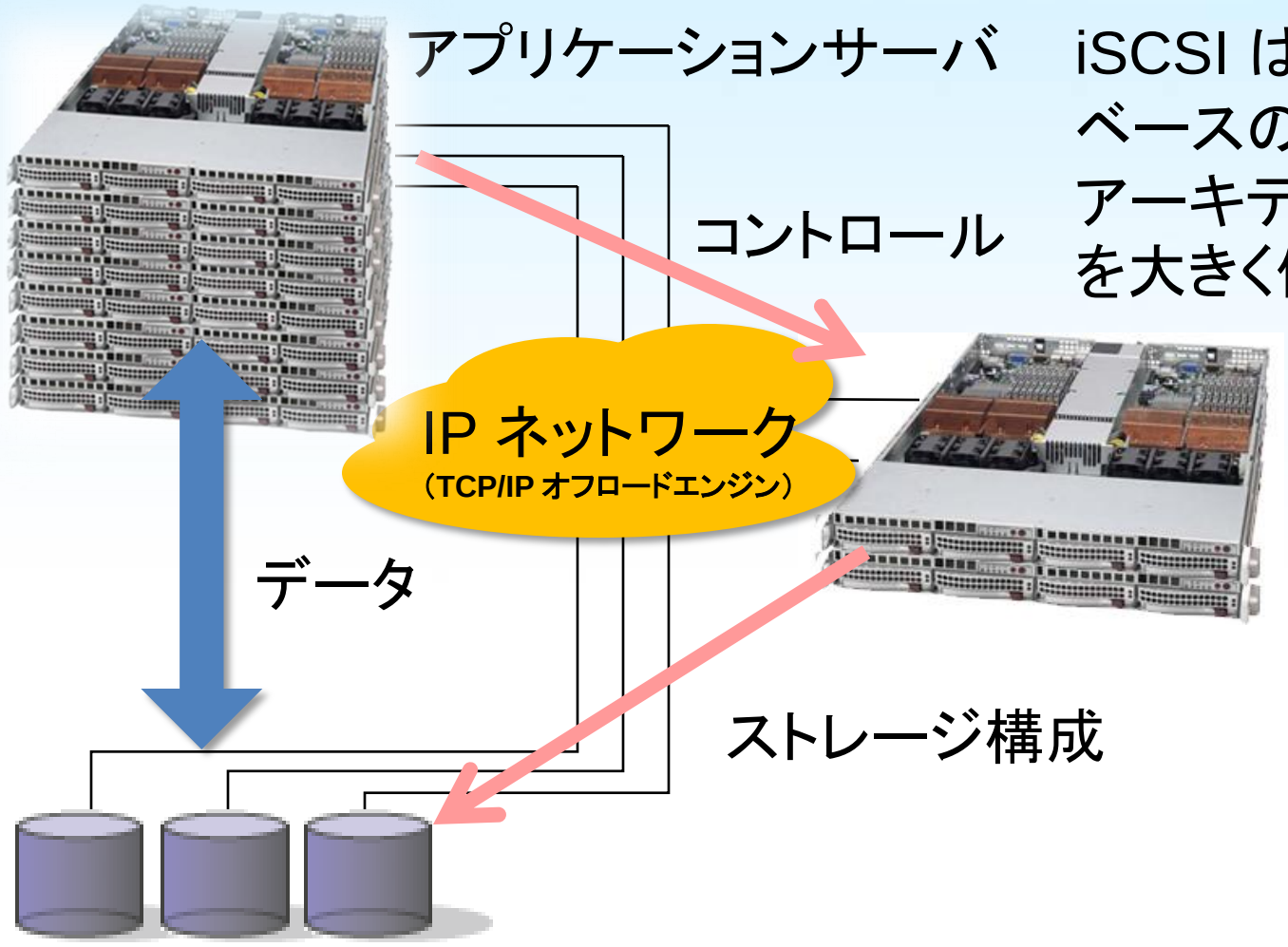
（source:IBM)

IBMオブジェクトストレージ概要



（source:IBM)

オブジェクトベースストレージ
iSCSI の登場とその影響



iSCSI は、オブジェクトベースのストレージアーキテクチャの発展を大きく促す

オブジェクトベースストレージ
Panasasストレージクラスタ

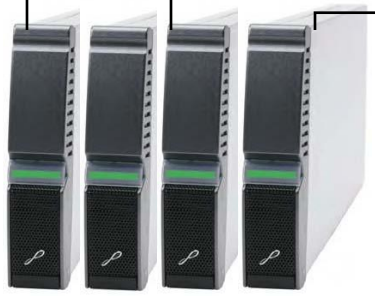
アプリケーションサーバ
DirectFLOWクライアント

IP ネットワーク
（TCP/IP オフロードエンジン）

コントロール
ActiveScale
オペレーティング
システム

パラレルな
データパス

ストレージ構成
ActiveScaleオペレーティング
システム



オブジェクトベースストレージ
StorageBladeクラスタ

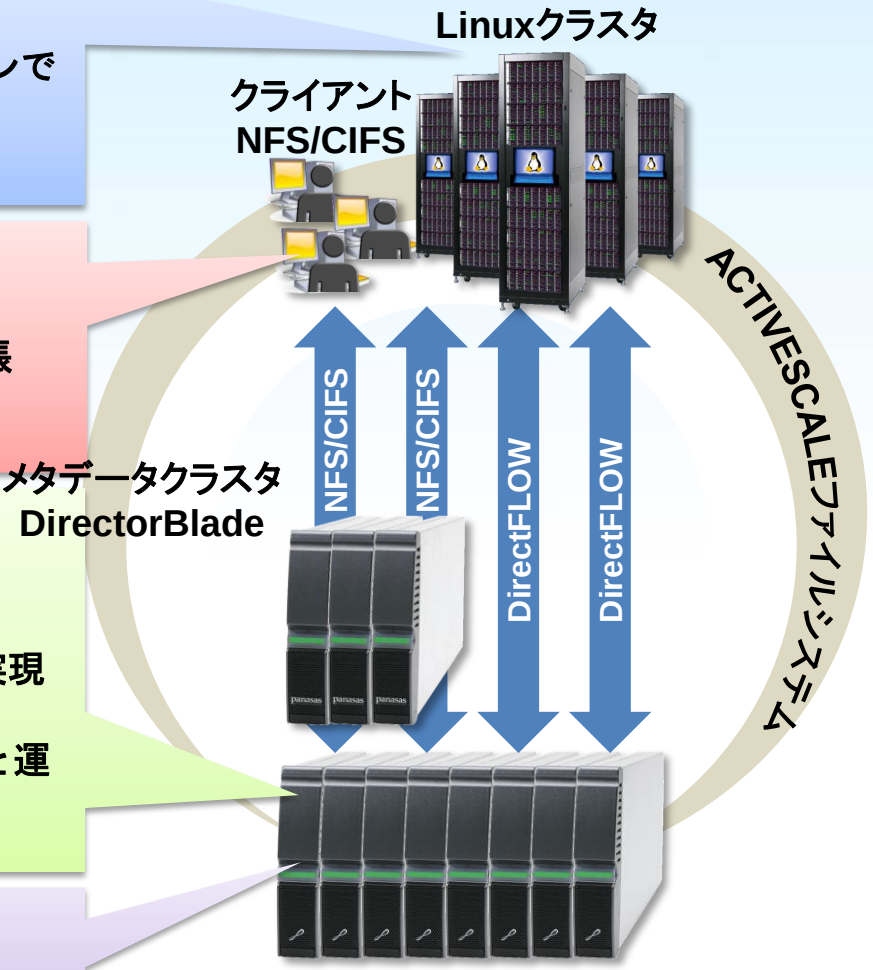
Panasasストレージクラスタ

DirectFLOW クライアントS/W
- クライアントからの同時アクセスを並列に処理可能
- RedHat,SUSEなどの主要なLinuxディストリビューションで利用可能
- pNFSにも対応可能

スケーラブルな NFS/CIFS/NDMPサーバ
- 負荷を自動的にストレージクラスタ全体に分散
- クライアント数の増加に合わせてスケーラブルな性能拡張
- 全てのDirectorBladeが全てのファイルにアクセス可能

シングルネームスペース
- 同一データへのいずれのプロトコルでのアクセスも可能
- シングルファイルシステム
- DirectFLOW/NFS/CIFS/NDMP間の完全なコヒレンシの実現
- 非Linuxのデバイスをシステムに統合
- グローバルネームスペースによるシステムの容易な拡張と運用の容易さ

オブジェクトベース
- 優れたスケーラビリティ、信頼性、運用管理
- Panasas Tiered Parityによるデータ保護の強化



スケーラブルシステムズ株式会社

SMPとクラスタのギャップを埋める

SMP (Shared Memory Systems)

ワークステーションやサーバ
PA-RISC,POWER5,
Itaniumなどのプロセッサ
によるSMPサーバ

クラスタシステム

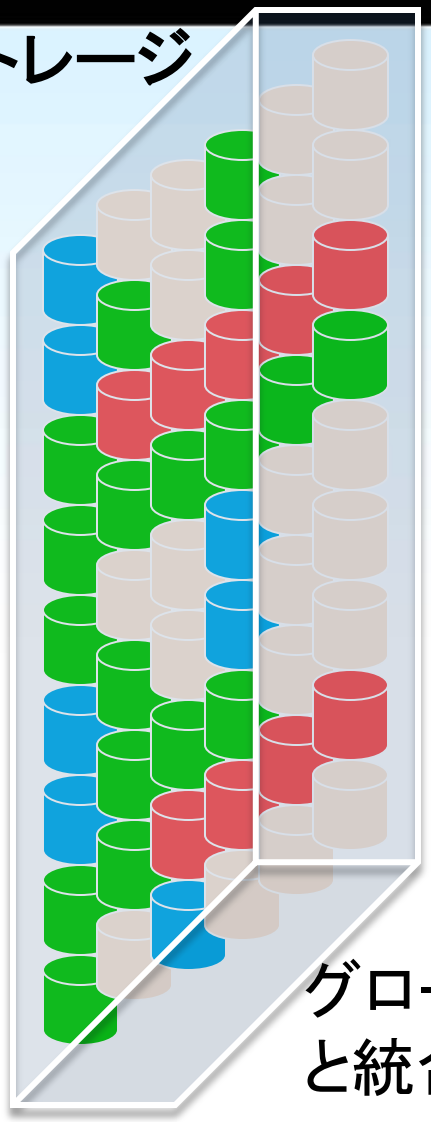
システムの構築には、
高いITスキルが要求される
運用管理コストが高い
複雑なオペレーション環境
複数のOS
クラスタファイルシステム
ソフトウエア、インストールや
アップグレードなど

Panasas Storage Cluster

ワークステーション
サーバ
クラスタ

#Processors 2 4 8 16 32 64 128

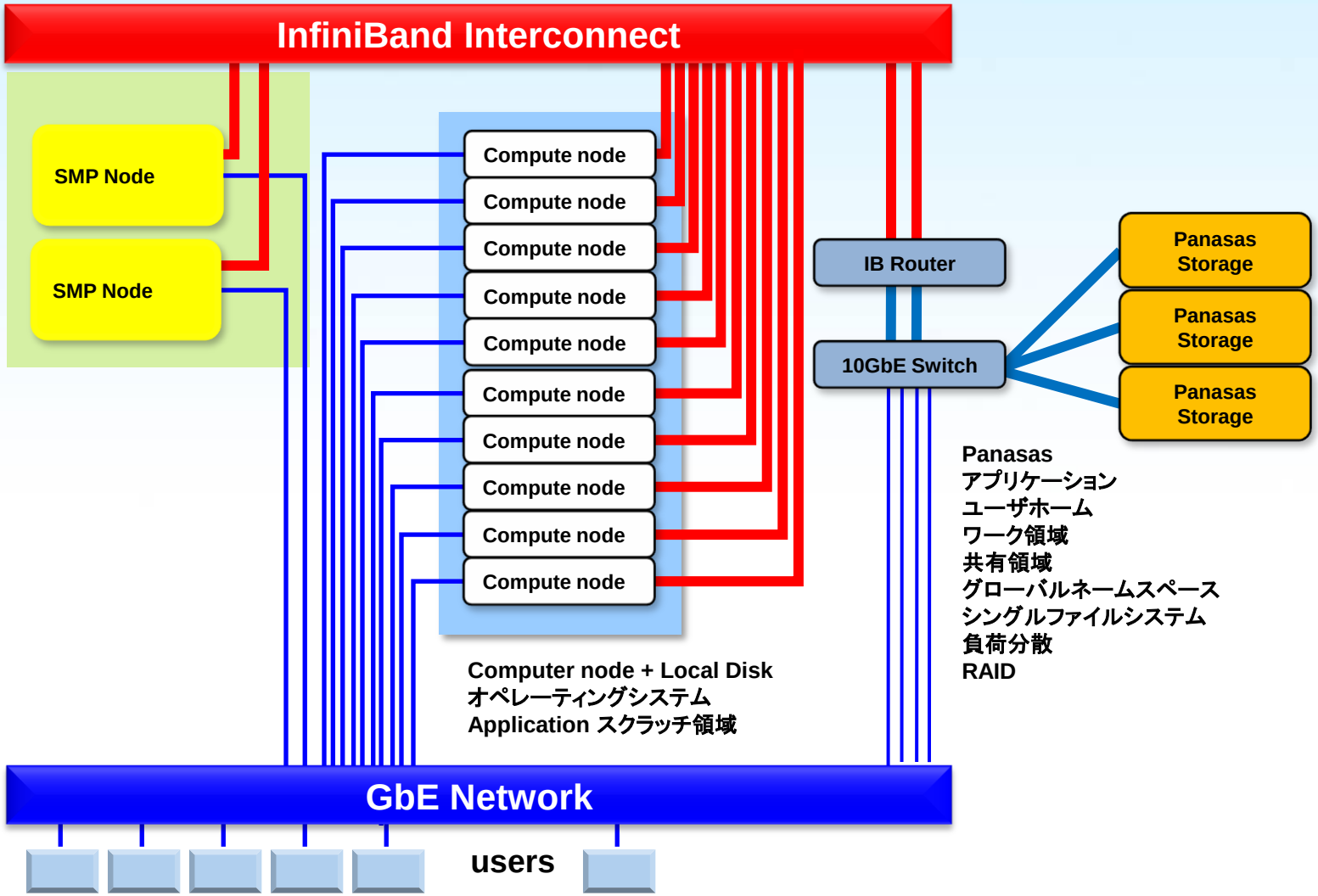
グローバルネームスペース



- 利用の簡便さ
  - 全クライアントが全データを見ることが可能
  - マウント・ポイント管理が不要
  - クライアント側の変更が不要
- 透過性
  - 容易な拡張
  - Failover
- スケーラビリティ
  - ネームスペースをペタバイトにまで拡張可能
  - 大規模ボリュームの容易な管理

グローバルネームスペース
と統合された運用管理環境

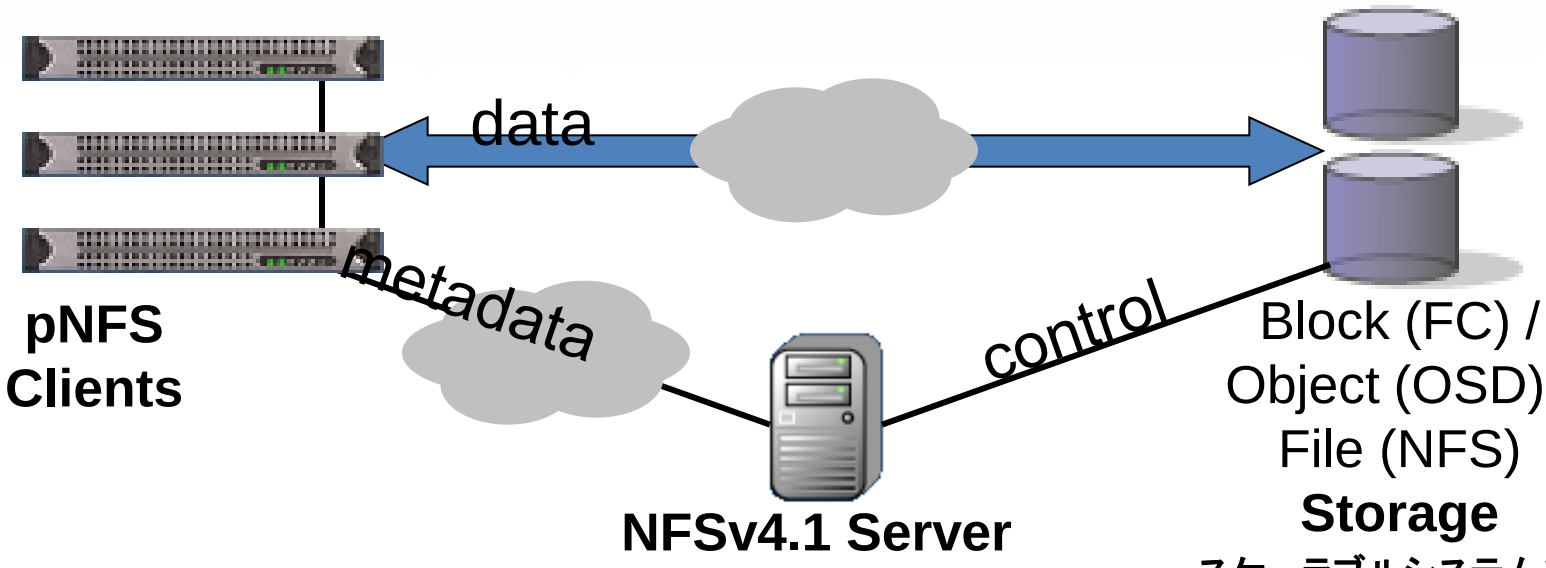
Panasasストレージクラスタ
シングルグローバルネームスペース

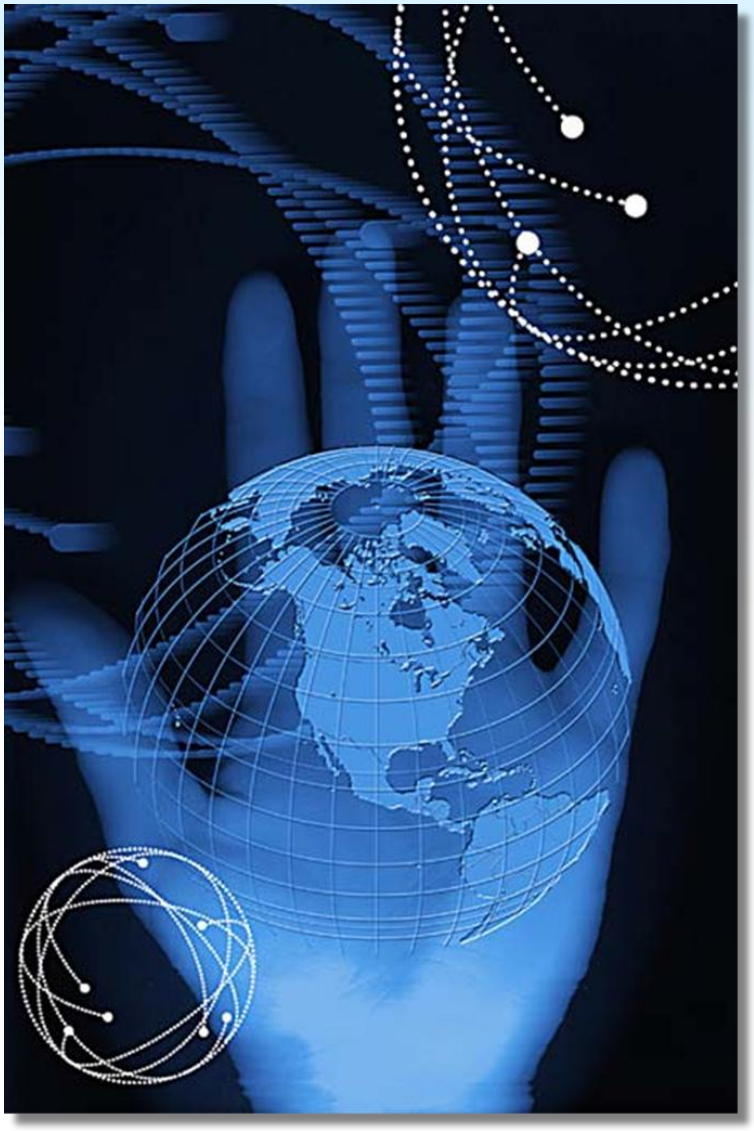


スケーラブルシステムズ株式会社

pNFS: ストレージクラスタの標準化

- pNFSはNetwork File System v4 標準プロトコルの拡張
- パラレルでの直接データアクセスが可能
  - パラレルネットワークファイルシステムクライアント
  - ストレージデバイスへの複数のストレージプロトコルでのアクセス
  - NFSをデータパスから分離





お問い合わせ

0120-090715

携帯電話・PHSからは（有料）

03-5875-4718

9:00-18:00 (土日・祝日を除く)

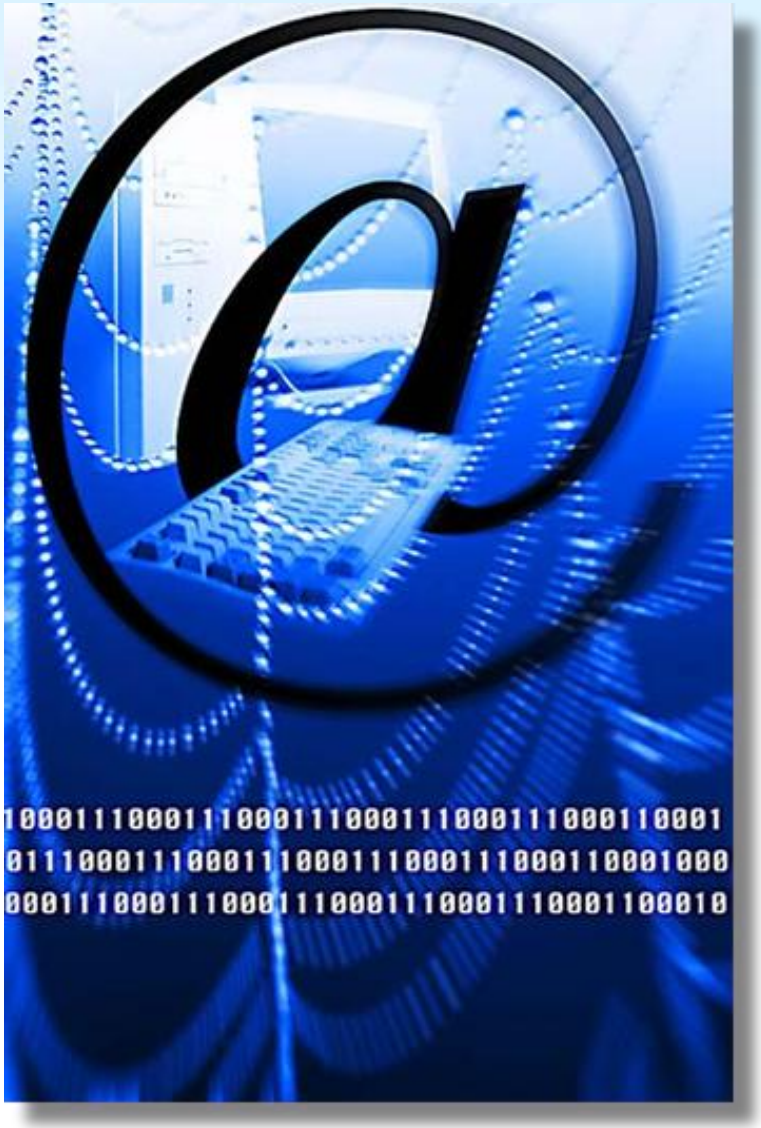
WEBでのお問い合わせ

www.sstc.co.jp/contact

1/24/2010

スケーラブルシステムズ株式会社

さらに詳しい情報や最新情報は……



ホームページにて公開しています。ホームページには、お問い合わせ窓口も開設してありますので、ご利用ください。

コンサルテーション

http://www.sstc.co.jp/biz

製品技術

http://www.hp2c.biz

1/24/2010

スケーラブルシステムズ株式会社