
ORACLE®

Oracle **Database** Technology Night
～集え！オラクルの力（チカラ）～ in September

データベースに最適なストレージ構成の極意

ORACLE DATABASE 12c

Plug into the Cloud

日本オラクル株式会社
クラウド・テクノロジー事業統括
Database & Exadata プロダクトマネジメント本部
応用技術部 ディレクター
柴田 長

ORACLE

Safe Harbor Statement

The following is intended to outline our general product direction. It is intended for information purposes only, and may not be incorporated into any contract. It is not a commitment to deliver any material, code, or functionality, and should not be relied upon in making purchasing decisions. The development, release, and timing of any features or functionality described for Oracle's products remains at the sole discretion of Oracle.

自己紹介

“しばちょう“こと、柴田 長(しばた つかさ)と申します

日本オラクル株式会社
クラウド・テクノロジー事業統括
Database & Exadata プロダクトマネジメント本部
応用技術部 ディレクター
柴田 長

Oracle Technology Networkで、ほぼ毎月連載中
「しばちょう先生の試して納得！DBAへの道」

http://www.oracle.com/technetwork/jp/database/articles/shibacho/index.html

本日のアジェンダ

- なぜ、ストレージ設計が大切なのか？
- 従来のRAWデバイス構成の課題
- Oracle Automatic Storage Management
  - 概要
  - 12.1の主な新機能
  - 12.1で実現できないこと
- ストレージからのASM Diskの切り出し方法ガイド
- まとめ

なぜ、ストレージ設計が大切なのか？

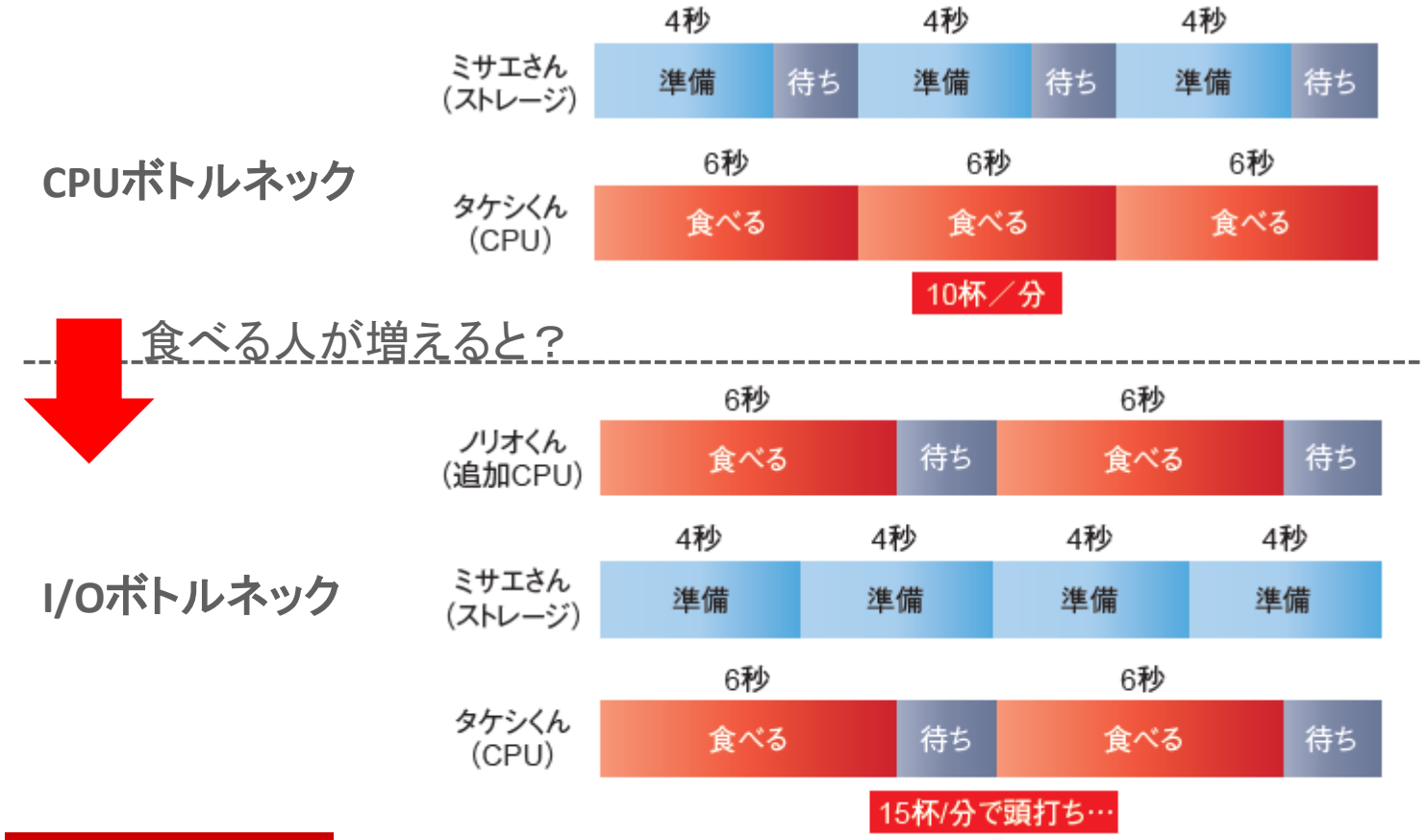
データベース時間（処理時間）とは？

ほぼ、CPU時間とI/O時間の合計で決まる

わんこそば理論

http://www.atmarkit.co.jp/ait/articles/1606/01/news007.html

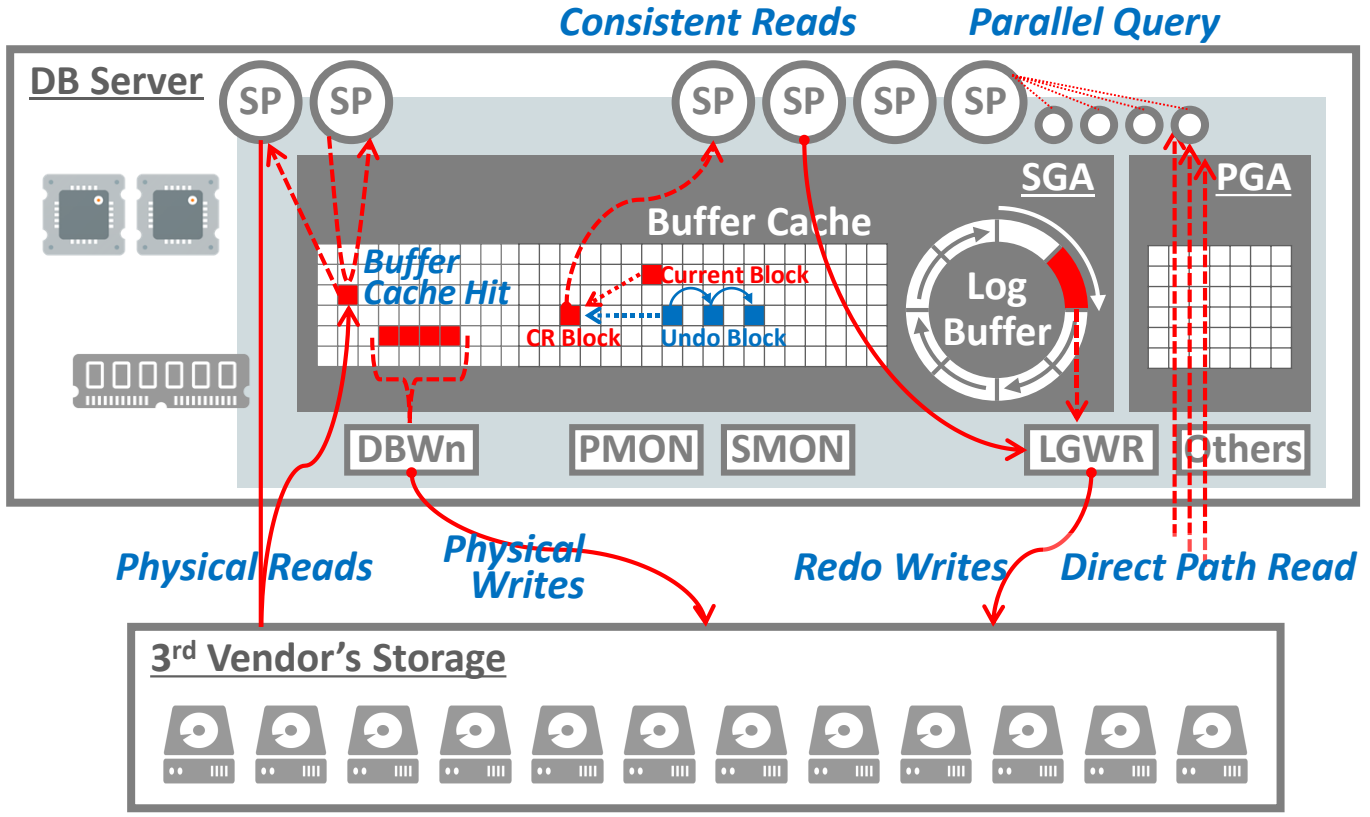
そばがお椀に盛られなければ食べられない
= データが無ければ、CPU処理はできない



Oracle Databaseのアーキテクチャ

ストレージ・アクセスが必要な主な処理

- SQL実行時に、Server Processによる Data Blockの読込み
  - Buffer Cache上にBlockが存在しない時
  - Direct Path Read（Table Full Scan）
- DBWRによるDirty Blockの書出し
- LGWRによるRedo Recordの書出し
- その他の処理
  - アーカイブ・ログ生成
  - Flashback Logの書出し
  - RMANバックアップ



※ 大容量メモリでは解決できない処理もある

ストレージ性能が充分ではない場合

性能劣化の例は?

#	Case	待機イベントの例	性能劣化の事象
1	Data Blockの読込み遅延	db file sequential read db file scattered read direct path read	SQL処理時間の劣化する データが届かない為、CPUリソースも活用出来ない状況
2	Dirty Blockの書出し遅延	free buffer waits	DBWRによるDirty Blockの書出しは、ユーザー・トランザクションとは非同期のため、この書出しが遅延すること自体は、直接的にSQL処理時間へは影響しない。しかし、Dirty BlockがBuffer Cache上に増加することにより、他のSQL処理で使用したいBlockをキャッシュし切れなくなるため、free buffer waits待機イベントが発生し、最終的にはユーザーのSQL処理時間の劣化につながる可能性有り
3	Redo Recordの書出し遅延	log file sync log buffer space	Commitコマンドのレスポンス・タイムが遅延する。最悪、書き出されていないRedo RecordがLog Buffer内に滞留することで、Log Bufferの空きスペースが枯渇し、新たなRedo Recordを生成する処理を実行できず待機する可能性有り

【注意ポイント！】
待機イベントの発生自体は100%悪ではありません。
その待機イベントの平均待機時間を、負荷が平常時とピーク時とで比較することが大切ですね。

Oracle® Databaseパフォーマンス・チューニング・ガイド 12cリリース1 (12.1)

10.2.3 待機イベントおよび潜在的な原因の表

表10-1に、待機イベントと考えられる原因との関連付けの他、次に検討するのに最も有益と思われるOracleデータの概要を示します。

表10-1 待機イベントおよび潜在的な原因

待機イベント	一般的な領域	考えられる原因	検索/調査
buffer busy waits	バッファ・キャッシュ、DBWR	バッファ・タイプによって異なります。たとえば、索引ブロックの待機は、昇順に基づく主キーが原因である場合があります。	問題が発生している間にV$SESSIONを調べ、競合したブロックのタイプを判別します。
free buffer waits	バッファ・キャッシュ、DBWR、I/O	低速なDBWR(おそらくI/Oに起因) 小さすぎるキャッシュ	オペレーティング・システム統計を使用して書込み時間を調べます。キャッシュが小さすぎることの証拠があるかどうかについてバッファ・キャッシュ統計をチェックします。
db file scattered read	I/O、SQL文のチューニング	チューニングが適切ではないSQL 低速なI/Oシステム	V$SQLAREAを調べて、多数のディスク読取りを実行するSQL文があるかどうかを確認します。I/OシステムとV$FILESTATをクロスチェックして、読取り時間に問題がないかをチェックします。
db file sequential read	I/O、SQL文のチューニング	チューニングが適切ではないSQL 低速なI/Oシステム	V$SQLAREAを調べて、多数のディスク読取りを実行するSQL文があるかどうかを確認します。I/OシステムとV$FILESTATをクロスチェックして、読取り時間に問題がないかをチェックします。
enqueue待機(enq:で始まる待機)	ロック	エンキューのタイプにより異なる	V$ENQUEUE_STATを参照します。
ライブラリ・キャッシュ・ラッチ待機: library cache、library cache pinおよびlibrary cache lock	ラッチの競合	SQLの解析または共有	V$SQLAREAを調べて、比較的多数の解析コールまたは多数の子カーソルを使用するSQL文があるかどうかを確認します(VERSION_COUNT列)。V$SYSSTATの解析統計と毎秒の対応する割合を調べます。
log buffer space	ログ・バッファのI/O	小さいログ・バッファ 低速なI/Oシステム	V$SYSSTATの統計redo buffer allocation retriesをチェックします。メモリーの構成の章の、ログ・バッファの構成の項をチェックしてください。オンラインREDOログを格納するディスクをチェックして、リソースの競合の有無をチェックします。
log file sync	I/O、コミット過剰	オンライン・ログを格納する低速なディスク バッチされないコミット	オンラインREDOログを格納するディスクをチェックして、リソースの競合の有無をチェックします。V$SYSSTATから毎秒のトランザクション数(コミット数+ロールバック数)をチェックします。

http://docs.oracle.com/cd/E57425_01/121/TGDBA/pfgrf_instance_tune.htm#i22670

パフォーマンス・チューニング

基本的な考え方

時間↓ = 処理量↓ / (速度 * 並列度)↑

じかん = みちのり ÷ はやさ

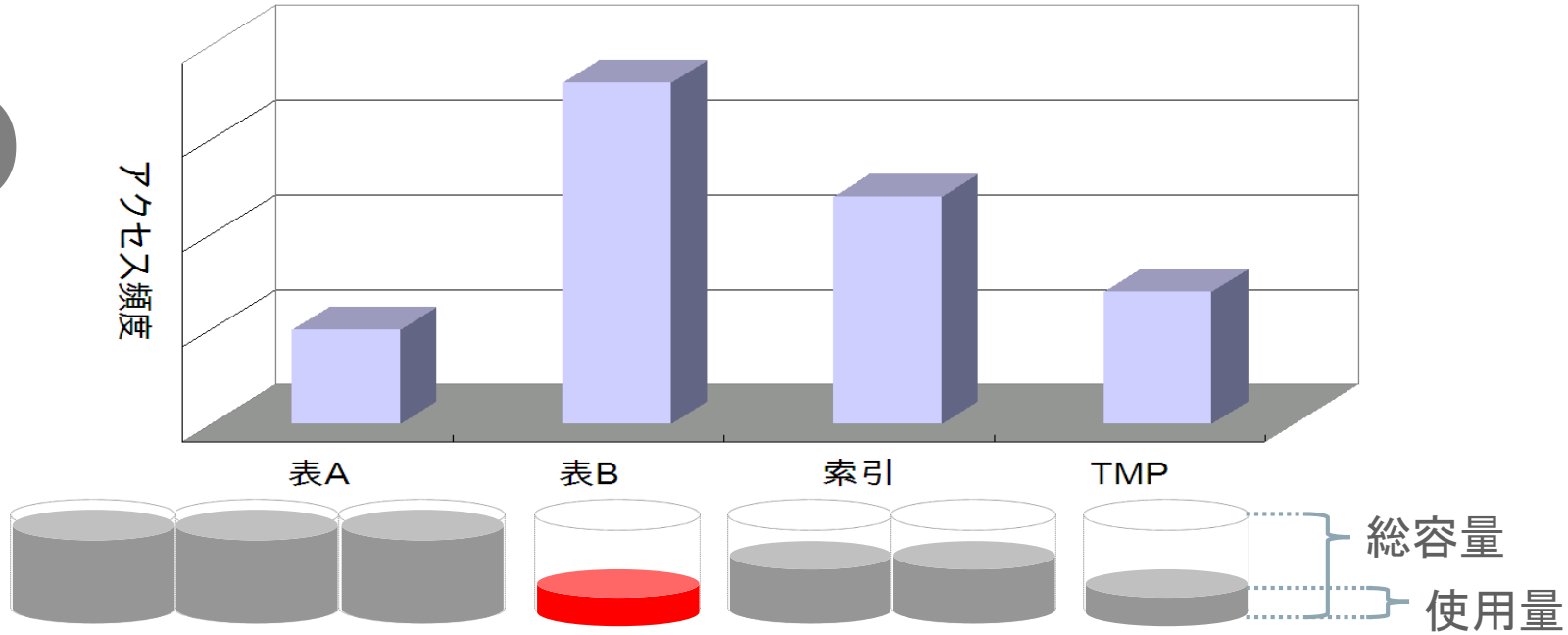
- 処理量を減らす
  - Index, Partitioning, Compression, Exadata Smart Scan/Storage Index …
- 高速化
  - Database In-Memory, Flash Device, InfiniBand, Exafusion, …
- 並列化
  - Parallel Query, Multi-Core, RAC, **ASM**, …

従来のRAWデバイス構成の課題

複数ディスク上へのデータベースの配置の課題

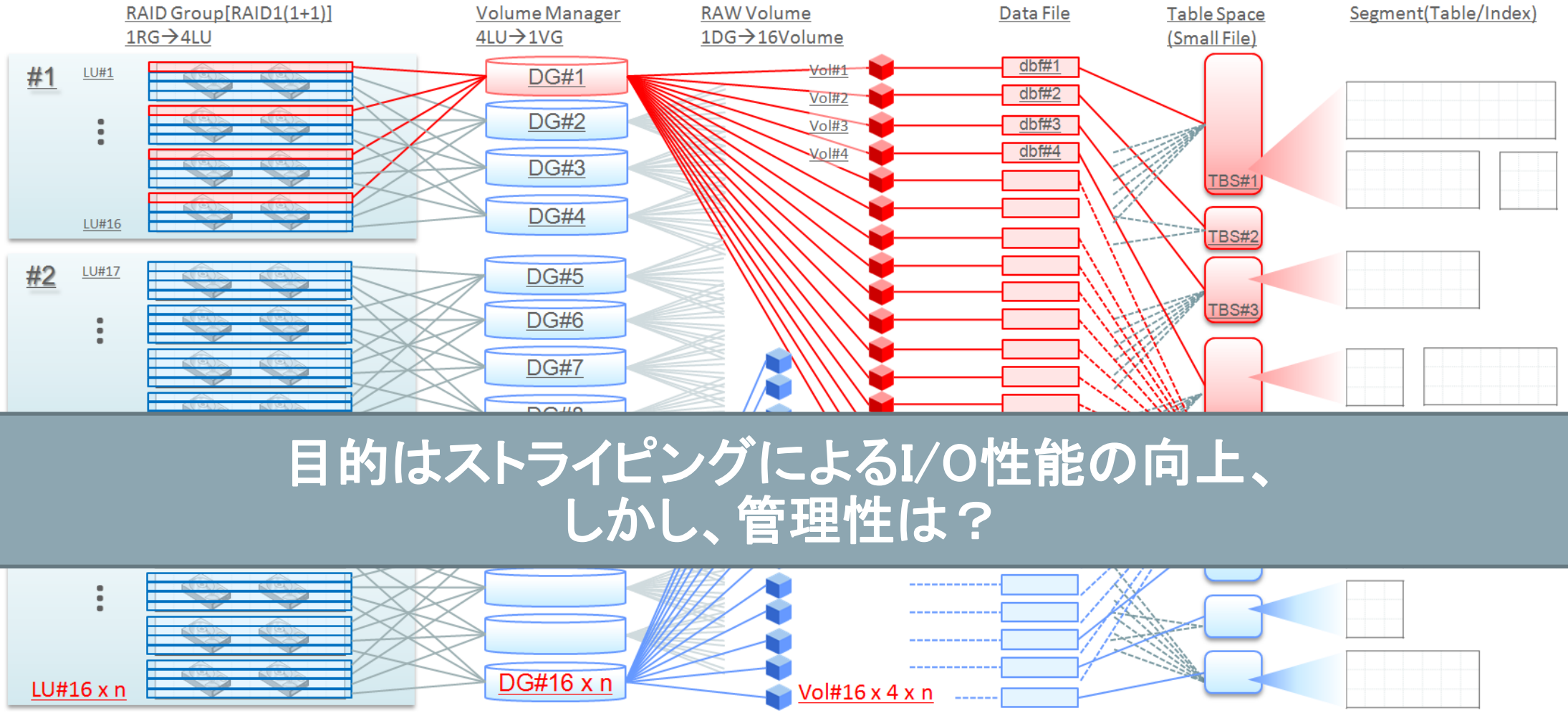
容量ベースで設計すると、Hot Spotが生まれる？

- 業務要件の複雑化やアプリ改修の短期化により、
  データベースのオブジェクト単位で個別最適化を目指す運用は難しい



Hot Sport?

Hot Sportを回避する為のRAWデバイス構成例



目的はストライピングによるI/O性能の向上、
しかし、管理性は？

従来のRAWデバイス構成の課題

運用の複雑化

- 表領域が非常に細かく分割されている
  - 空き領域が表領域毎に独立している為、無駄な空き領域が増大
  - 監視対象（表領域）が多く、頻繁に領域不足に陥り、運用工数が増大
  - データ・ファイル数が多く、SQLの性能劣化やミス・オペレーションを誘発
  - 管理レイヤー数が多い為、運用オペレーションの複雑化
    - データベース管理者とストレージ管理者の間での調整作業の難しさ
- データ・ファイル追加時に、既存データをリバランスしていない
  - 空き領域が新規ボリュームにのみ存在する為、新たにINSERTされるレコードがそのボリュームに集中することで、ボトルネックが発生し易い
  - 既存レコードは既存ボリューム内に格納されている為、性能改善効果は無し

従来のRAWデバイス構成の課題

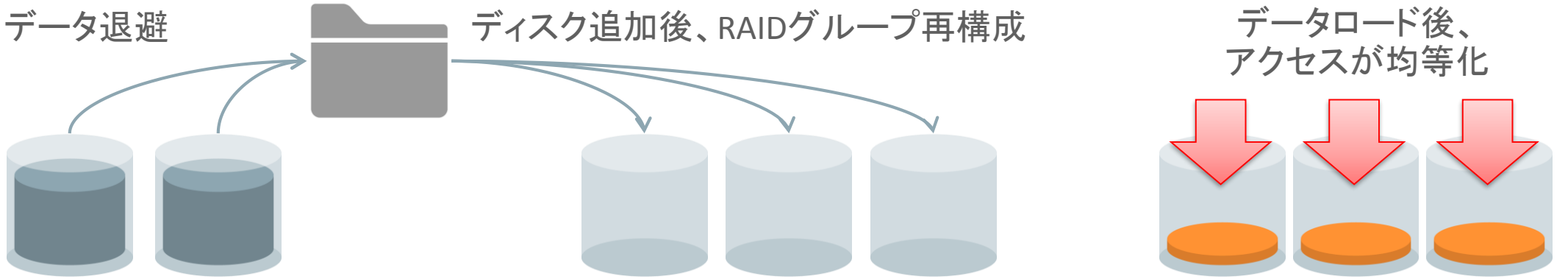
ディスク追加時に必要なオペレーション

- 領域不足/性能劣化の改善の為、ディスク追加

Hot Sport?

最新データが入っている1本のみにアクセスが集中

- Hot Sport回避のためには、既存データの再配置が必要



こんなストレージがあったらいいな。

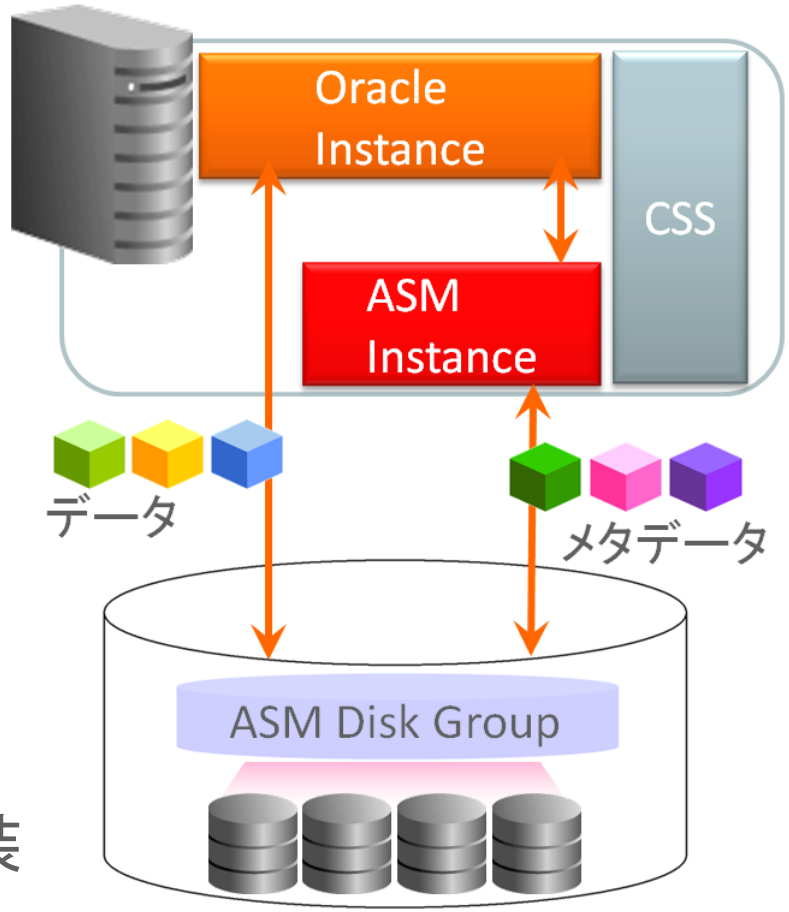
- 全てのデータベースのファイルが、それぞれ特定のデバイスに偏ることなく、全てのデバイスが**均等に配置**されるような仕組みがあったらいいな。
  - オブジェクト単位で容量やIOPS要件を整理する必要がなくなる
  - データベースのパフォーマンス・チューニングで、どのデータファイル（RAWデバイス）がボトルネックなのかを特定する必要が無くなる
- しかも、容量やIOPSが不足した場合、新規デバイスを追加したら、**自動的に再配置**される仕組みがあるといいな。
  - RAIDグループの再構成やデータの入れ直しが必要無くなる
- さらに、**低コスト**で組みたいが、**高い可用性**は維持したいな。

Oracle Automatic Storage Management

Oracle によるストレージ仮想化

Oracle Automatic Storage Management（ASM）

- Oracle Database 10g より提供されている、ディスク構成の仮想化技術
  - Oracleデータベースに対してボリューム・マネージャ兼ファイルシステム
    - Oracle Databaseにフラットなディスク・プールを提供 + ディスク管理工数を大幅削減
    - 複数ディスク・アレイにまたがってディスクを仮想化、ディスク追加/削除時にデータを透過的に再配分
  - エディション(EE/SE)に関係なく、シングル環境、クラスタ環境共に使用可
  - 11g Release2より、ASMクラスタファイルシステムが実装

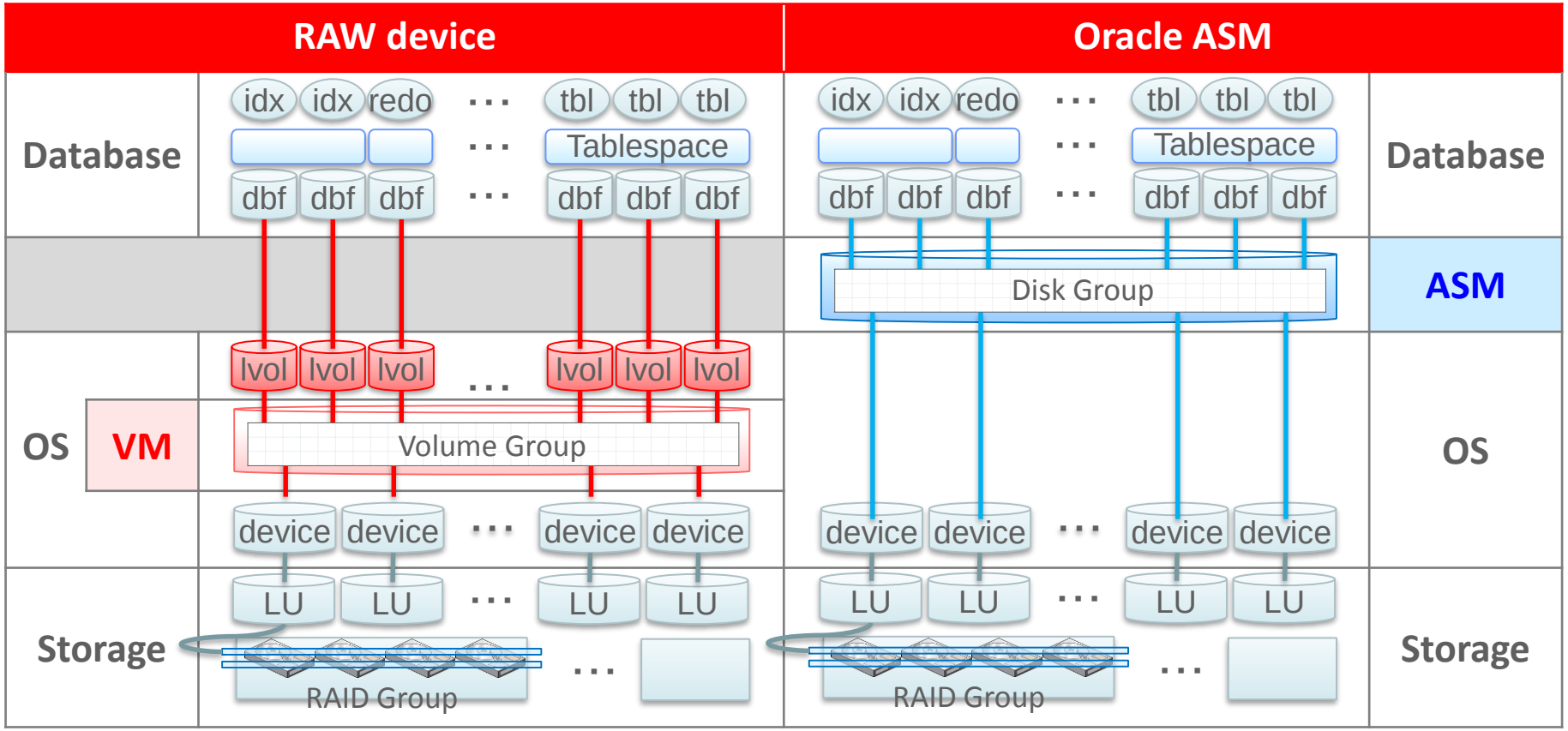


ASMによる全体最適化機能

- ストライピング
  - ASMディスク・グループ内の全てのディスクでストライピング(ホットスポット無し)
  - 性能の維持
- ミラーリング
  - ファイルタイプに応じて、Oracle レベルでミラーリング(2重化/3重化/ミラー無し)
  - 可用性の担保
- 動的リバランシング
  - ディスクの追加/削除時に自動的にデータを再配置
  - 拡張性

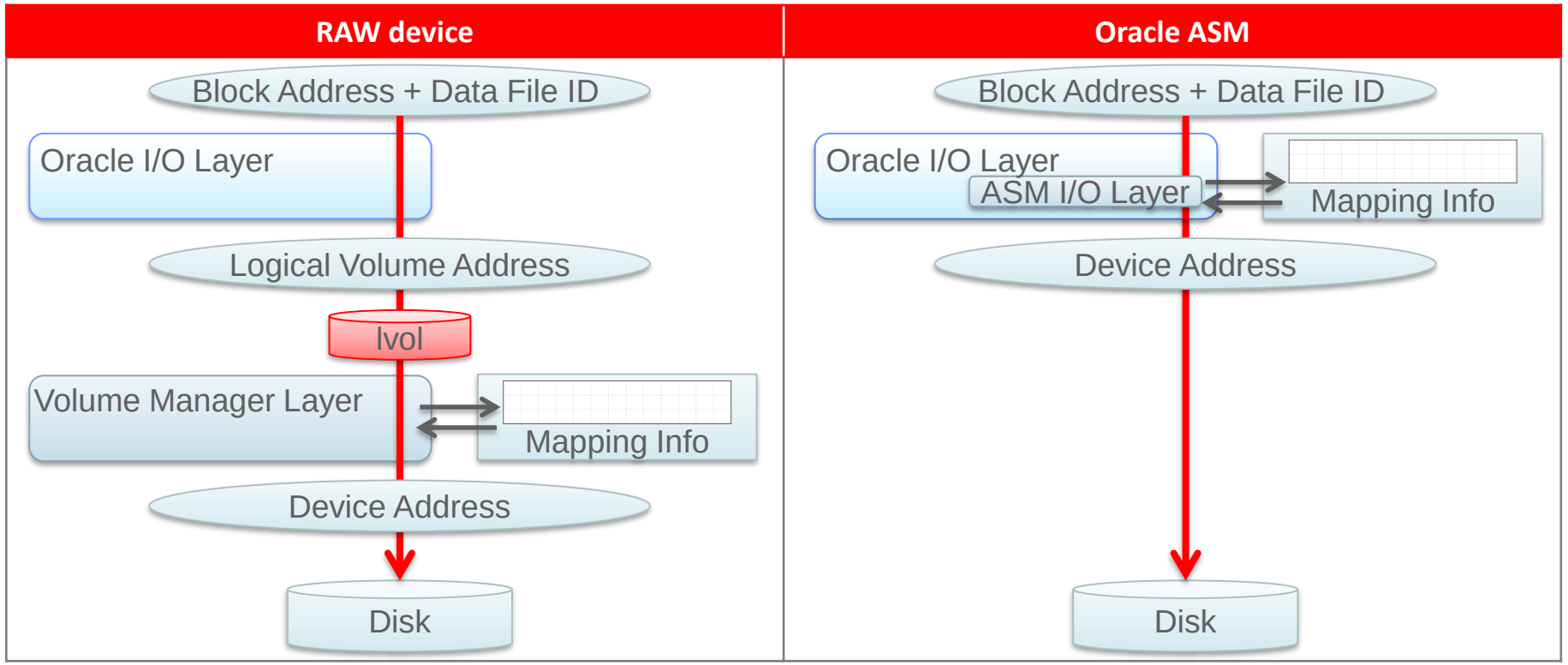
RAWデバイス構成とASM構成の比較

スタック構成イメージ図



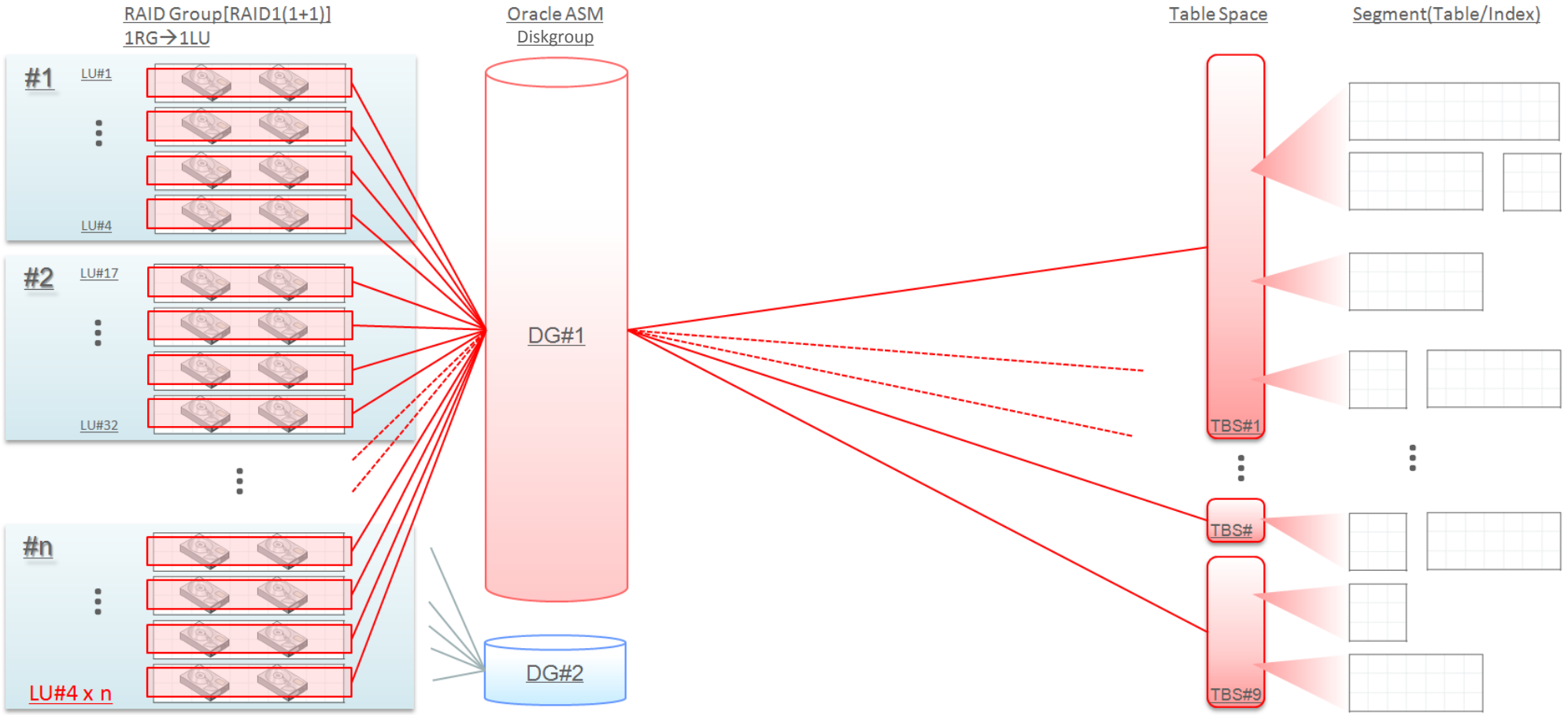
RAWデバイス構成とASM構成の比較

Block書き込み時のフロー・イメージ図



Oracle ASMの構成例

Simple is the BEST



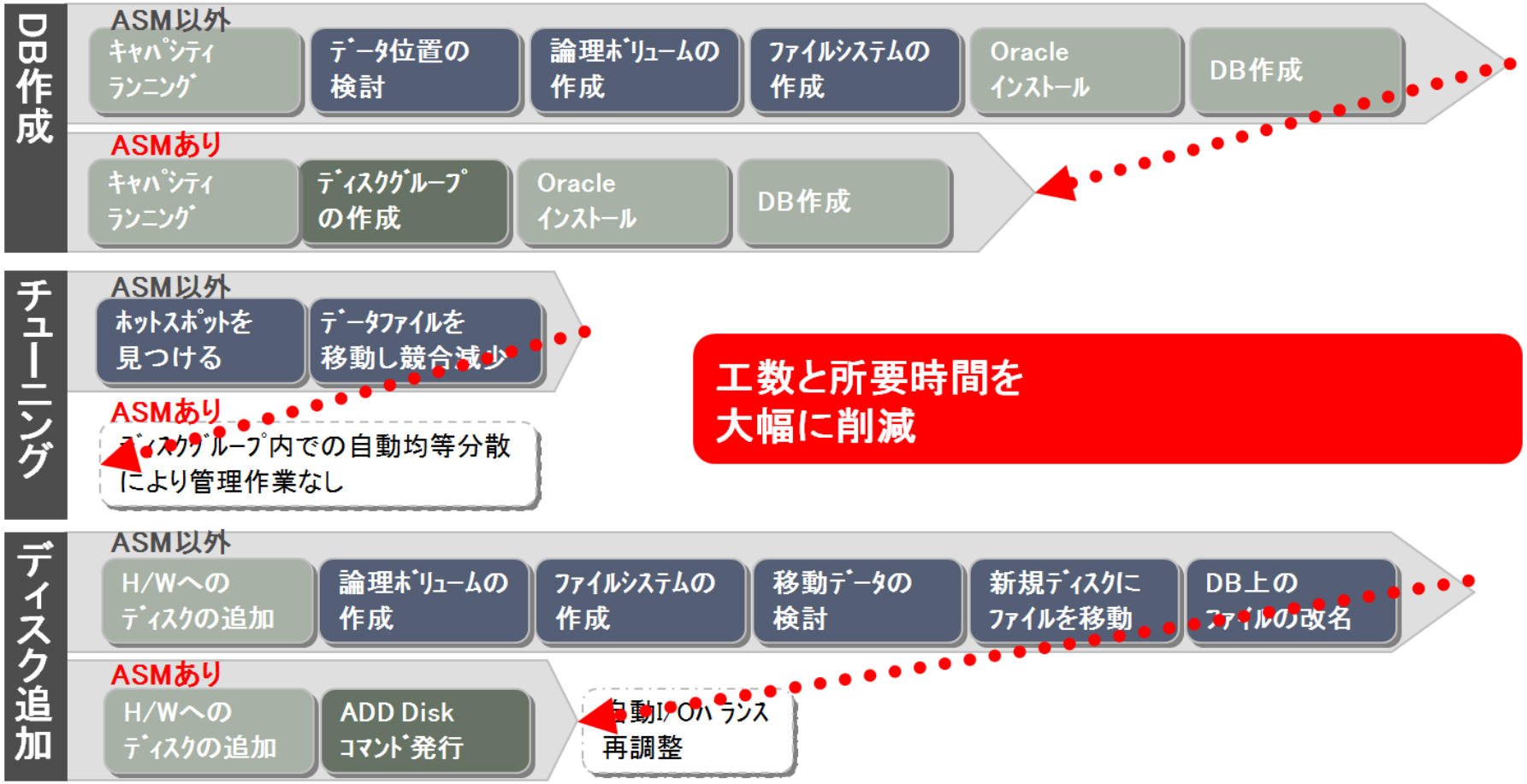
Oracle ASMによる運用管理の簡素化

従来構成の課題を解決

- オペレーションの簡素化
  - 表領域拡張やDisk追加の手順が簡素化し、運用オペミスのリスクが減少
- 管理対象オブジェクトの削減
  - ASM Diskgroupの容量内で表領域を自由に拡張可能であり、従来のVolumeやRAWデバイス（データファイル）を意識する必要なし
  - ストライピングでI/Oが均等化することで、表領域を細かく分割してI/O競合を回避する必要なし。表領域の総数を大幅に削減可能
- データ再配置の工数不要
  - Disk追加時に自動的に既存データの再配置（リバランシング）を実施

ASMによるデータベースの物理設計の簡易化

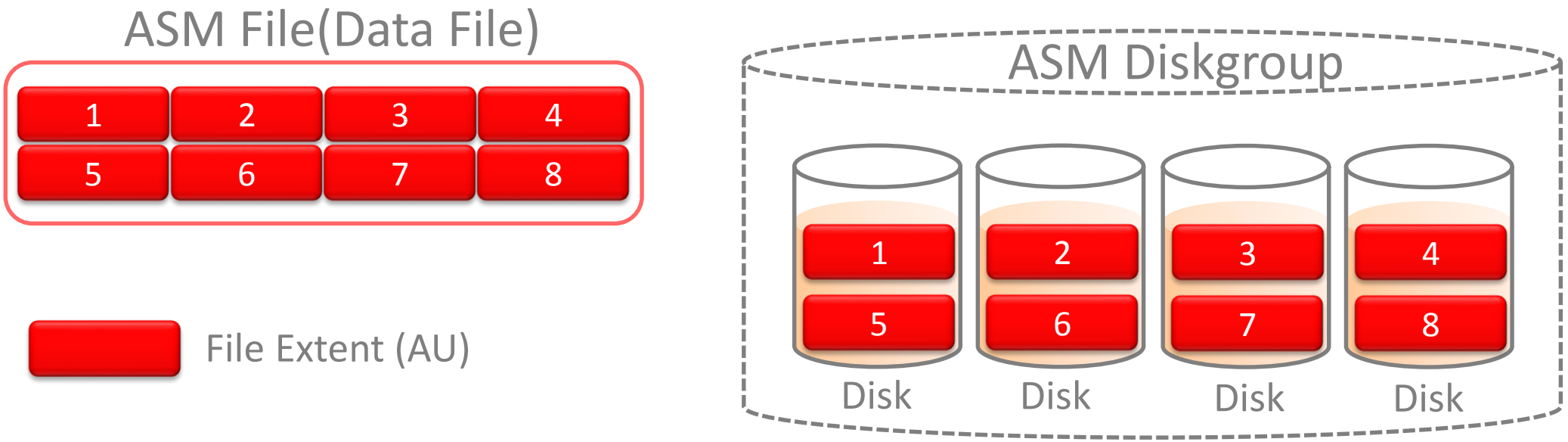
DBA のストレージ管理の効率化



Oracle ASMによるストライピング

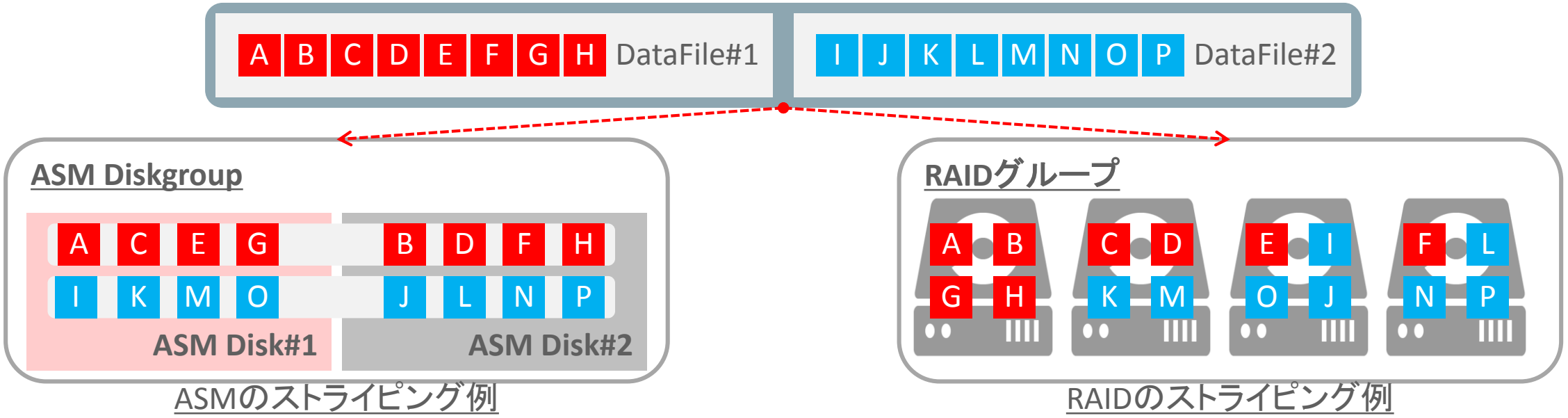
ASM File（データファイル）の分散配置例

- ASM Diskgroupに含まれる全てのASM Diskに対して、
  ASM File（Data File）をFile Extent(Allocation Unit:=AU)単位に分割して配置



RAID0とASMのストライピングの違いは？

- ASMのストライピングは、データファイル（ASM File）単位で均等化
- RAIDのストライピングは、データファイル（ASM File）を認識不可能
  - ハードディスク・レベルで、偏り（Hot Sport）が生まれてしまう可能性有り

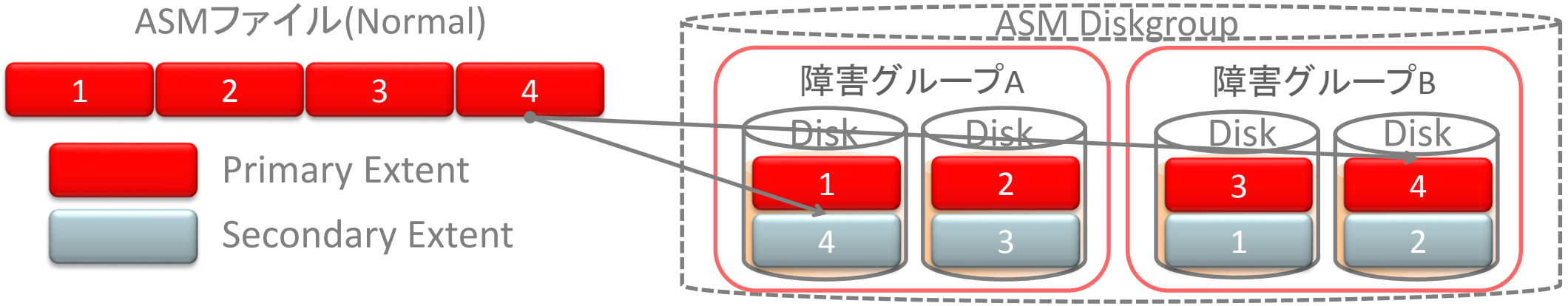


ASMのストライピング例　　RAIDのストライピング例

Oracle ASMによるミラーリング

Normal Redundancy時のミラーリングと障害グループ例

- 異なる障害グループに属するASM Disk間で保持
  - 通常、リソース(電源等)を共有している単位(筐体/コントローラー)で設定
  - 例えば、障害グループBのDisk障害が発生しても、ASMファイルへアクセス可能



【しばちょう先生の試して納得！DBAへの道】

第34回 ASMのミラーリングによるデータ保護(1) ～障害グループと冗長性の回復～

Oracle ASMによるミラーリング

Oracle Clientに透過的、かつ自動的にBlockを修復

- Normal / High Redundancy（2重化、3重化）で構成されている場合
  - **読み取り処理時にI/Oエラーを検知した場合**
    - Oracle Clientに対して透過的（ORAエラーは戻らない）
    - サーバー・プロセスは、ミラー側から読み取ることで処理継続
      - サーバー・プロセスは、不良ブロックの修復をASMへ依頼し、ASMが自動修復
  - **書き込み処理時にI/Oエラーを検知した場合**
    - Oracle Clientに対して透過的（ORAエラーは戻らない）
    - I/Oエラーが発生しても、一つでも成功していればサーバープロセスは処理継続
      - 書込み失敗をASMへ通知し、ASMが障害Diskが自動でオフライン化
    - 一時的な障害の場合、高速ミラー再同期により生存Disk側から必要最小限の差分データを同期
    - 復旧できない場合、ASM Diskgroupから切り離し（自動リバランスが発生）

ディスクの同時二重障害を考慮した構成案

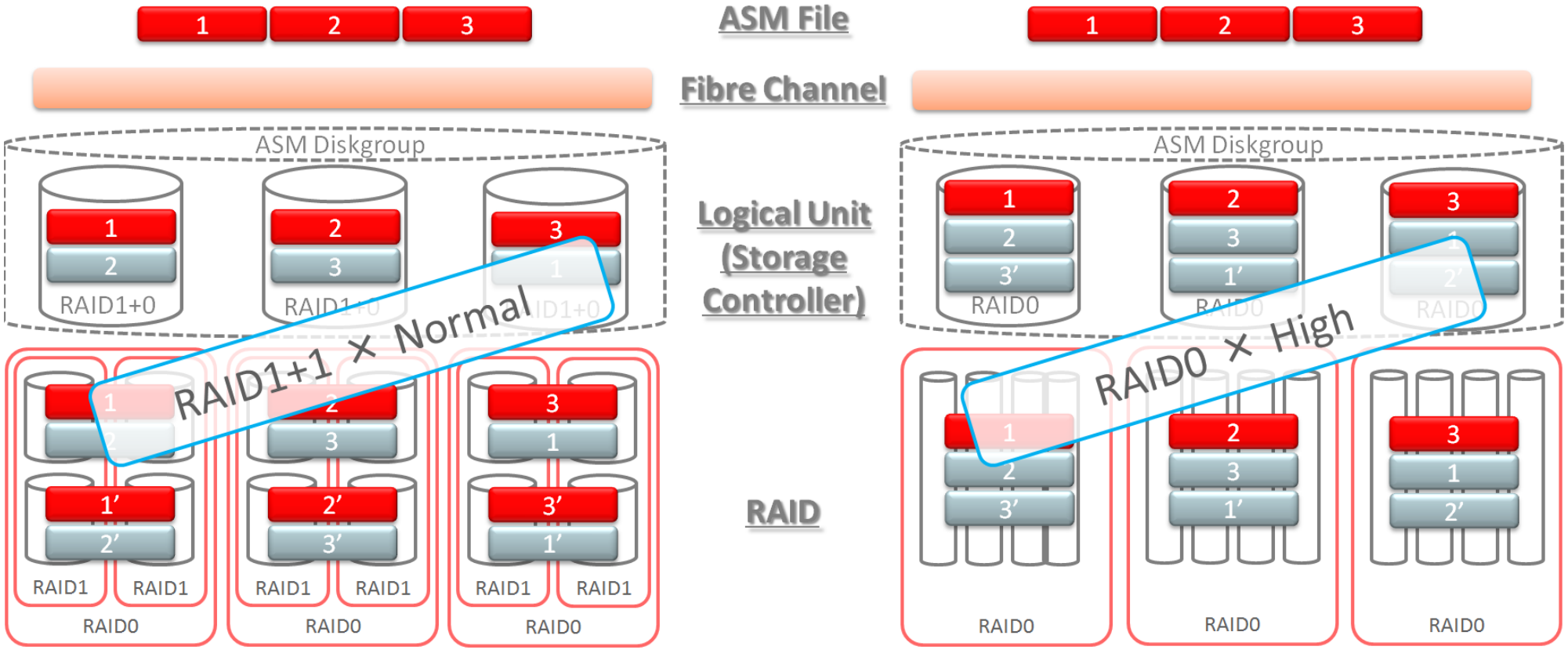
【参考までに】 RAID1+0 × Normal vs. RAID0 × High の比較

- ASMのHigh Redundancy（トリプル・ミラー）が効率的
  - Inifinibandを搭載したExadataであれば、よりメリットが出てくる

	RAID1+0 × Normal	RAID0 × High
Read時にアクセスされるDisk数	6本	**12本**
Write時のFC帯域を流れるデータ量	**2倍**	3倍
Write時のストレージ内のI/O量	4倍	**3倍**
Disk使用量	4倍	**3倍**

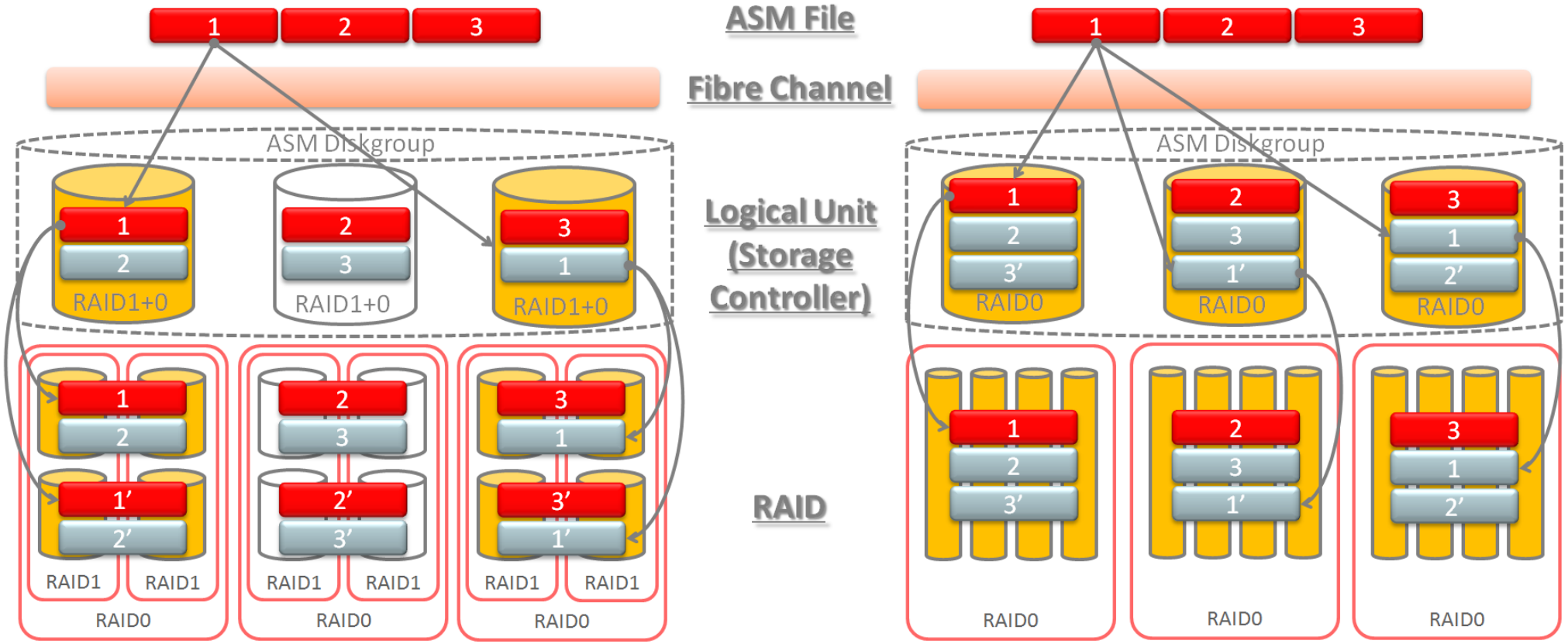
RAID1+0 × Normal vs. RAID0 × High

各構成の可用性を担保する構造図



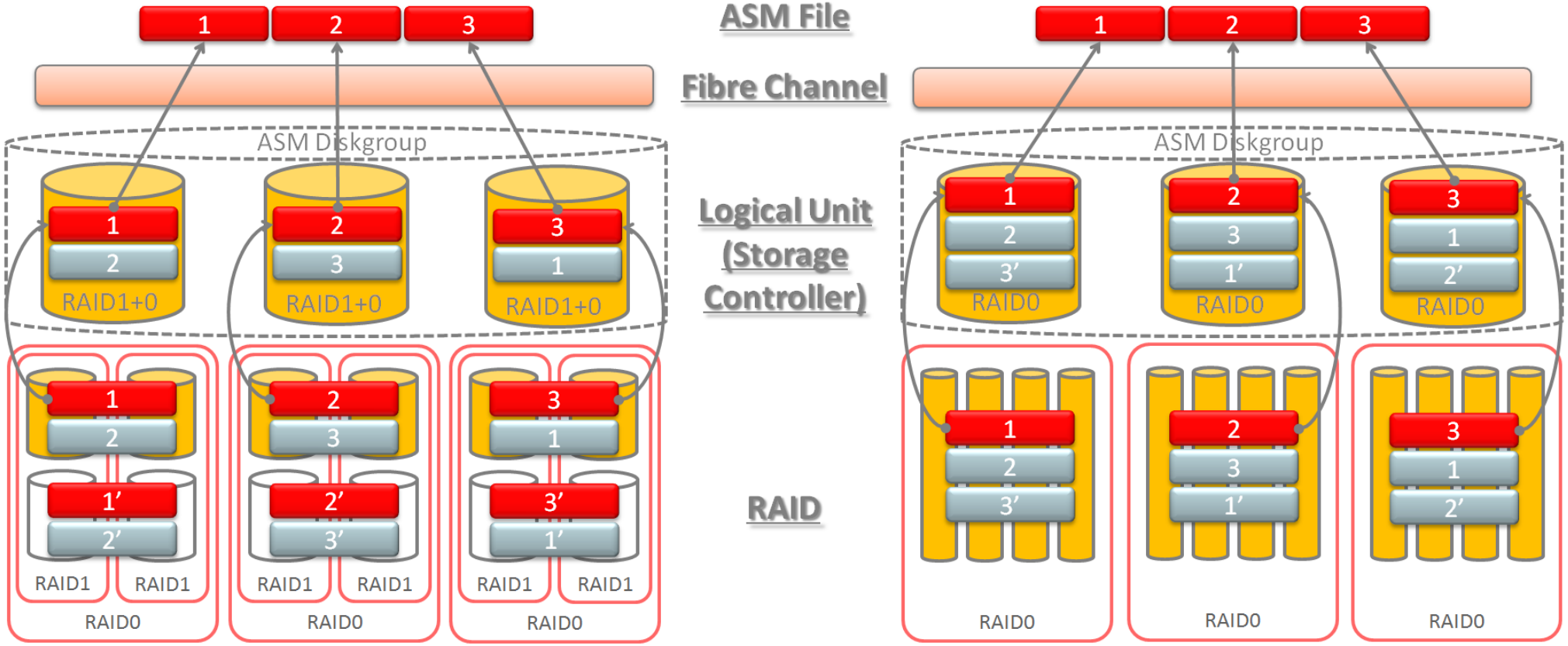
RAID1+0 × Normal vs. RAID0 × High

Write時のI/O量と分散状況



RAID1+0 × Normal vs. RAID0 × High

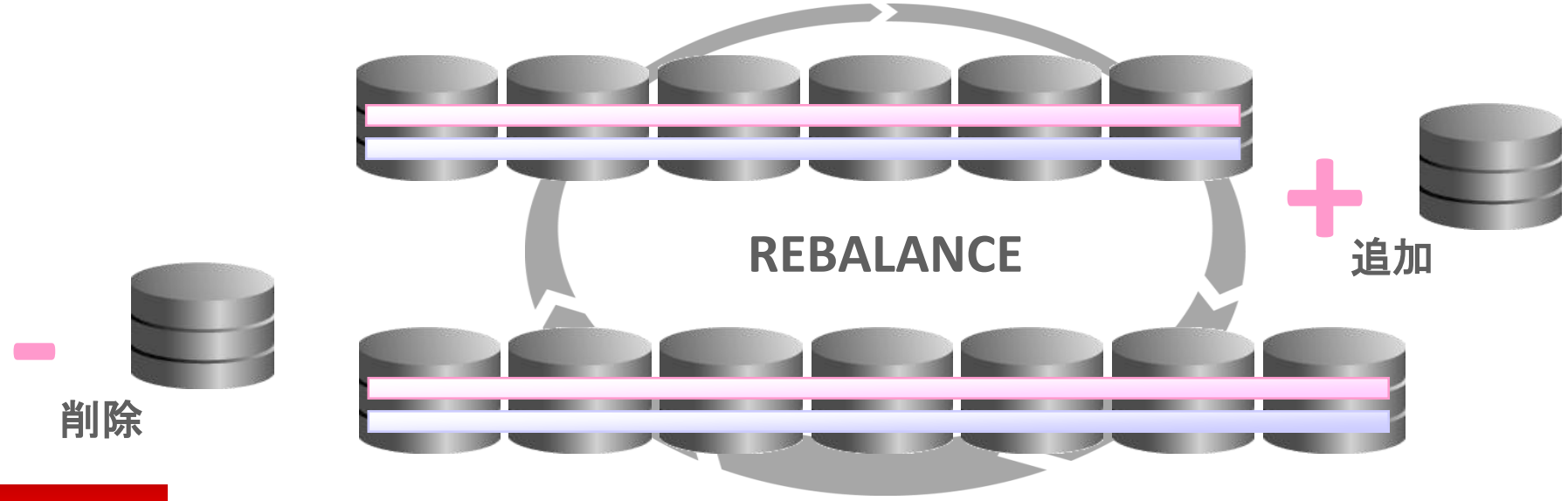
Read時にアクセスされるDisk数



Oracle ASMのリバランス（データ再配置）

データベース無停止でリバランスが可能

- ASM Diskを追加／削除（故障）した際、データの再配置を実施
  - メタデータ（配置状況）を元に、ASM File単位で全てのDiskに均等配置されるように最小限のExtent（AU）の移動で実現
  - 多重度（リバランス強度）の設定や計画実行で、業務影響を制御可能



Fast Mirror Resync（高速ミラー再同期）

一時的なディスク障害の復旧を高速化

- 本機能が実装される以前の課題
  - 一時的な I/O 障害（ディスク・パス障害や電源障害等）にも関わらず、対象ディスクを削除して自動リバランスが行われてしまう動作
    - 復旧後に、ディスクを追加して、再度リバランスを実行する必要有り
- 本機能による改善
  - 指定期間（デフォルト3.6時間）、障害ディスクをOFFLINE状態（自動削除を保留）にする
    - 復旧後にディスクをONLINE化する際に、本来書き込まれるはずであった差分データのみを同期
    - ASM ディスクの手動 ONLINE/OFFLINE 処理が可能

【しばちょう先生の試して納得！DBAへの道】

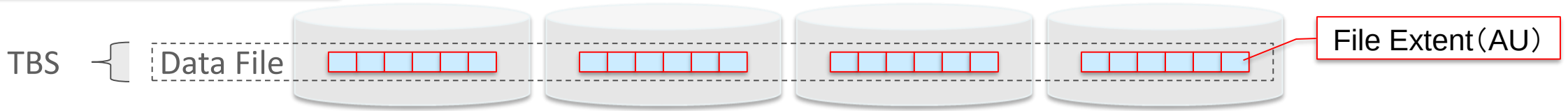
第35回 ASMのミラーリングによるデータ保護(2) ～高速ミラー再同期～

ASMによるデータの分散配置

Data File単位で各ASM Diskに対し均等にFile Extent(AU)を割り当て

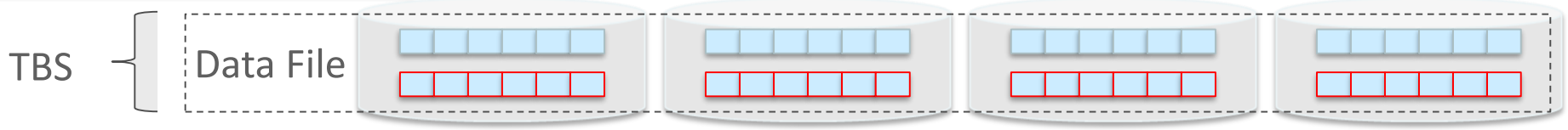
(1) 表領域の作成

CREATE TABLESPACE *TBS* DATAFILE '*+DATA*' SIZE 1G ;



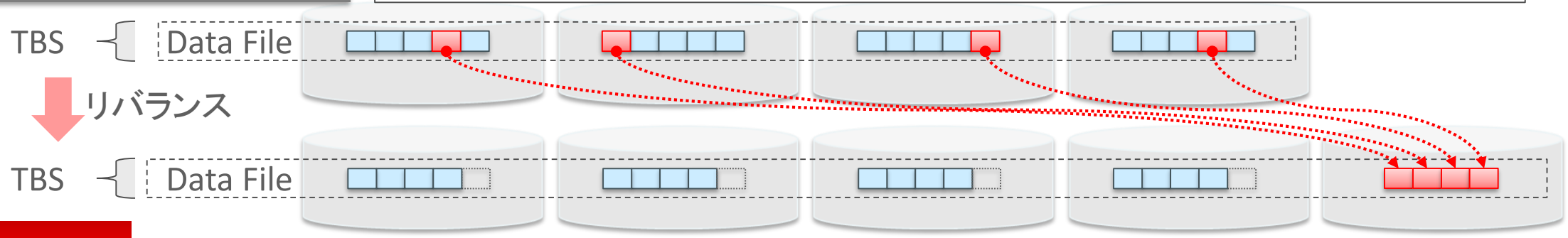
(2) 表領域の拡張

ALTER TABLESPACE *TBS* RESIZE 2G ;



(3) 新規Diskの追加（リバランス）

ALTER DISKGROUP *DATA* ADD DISK '*/dev/sde1*' ;



ASMによるデータの分散配置

リバランス処理の2つのフェーズ

- リバランス・フェーズ
  - ASM File（= Data File）単位で、各ASM Diskの利用率を均等に分散し直す
  - 追加Diskへ移動するFile Extentは、全てのASM Fileが対象
- コンパクション・フェーズ
  - リバランス・フェーズでFile Extentが追加Diskへ移動することで、既存Diskでは歯抜け状態になるが、コンパクション処理でその状態を解消
    - 各ASM Diskにおいて、後方に位置するFile Extent（AU）から順番に前方の空きスペースへ移動

【しばちょう先生の試して納得！DBAへの道】

第33回 ASMのリバランスの動作

リバランス強度の設定

Grid Infrastructure 11.2.0.2以降

- PSR11.2.0.2~ リバランス強度の動作変更有り
  - ASM_POWER_LIMITS初期化パラメータに設定可能な値は、0～1024（以前は、0~11）
  - ARB0プロセス（クラスタで1つのみ起動）が同時に発行する非同期I/Oの数
  - ストレージのI/O性能に応じた適切な値を設定すること
- 事前準備 : *対象のASM DiskgroupのCOMPATIBILITY.ASM属性を11.2.0.2以上*
  - デフォルト値は、ASM Diskgroupを作成する方法に依存するので注意
    - 例えば、Oracle ASM 12*c*では、12.1がASMCA使用時のCOMPATIBLE.ASM属性のデフォルト設定
      SQL CREATE DISKGROUP文とASMCMD mkdgコマンドを使用する場合のデフォルト設定は10.1

```
alter diskgroup <ASM Diskgroup Name>
        set attribute 'compatible.asm'='11.2.0.2.0';
```

ASM DiskgroupのCOMPATIBLE属性と使用可能になる機能

Oracle® Automatic Storage Management管理者ガイド 12cリリース1(12.1)

使用可能なディスク・グループ機能	COMPATIBLE.ASM	COMPATIBLE.RDBMS	COMPATIBLE.ADVM
より大きなAUサイズ(32または64MB)のサポート	>= 11.1	>= 11.1	該当なし
V$ASM_ATTRIBUTEビューに表示される属性	>= 11.1	該当なし	該当なし
高速ミラー再同期	>= 11.1	>= 11.1	該当なし
可変サイズのエクステント	>= 11.1	>= 11.1	該当なし
Exadataストレージ	>= 11.1.0.7	>= 11.1.0.7	該当なし
インテリジェント・データ配置	>= 11.2	>= 11.2	該当なし
ディスク・グループに格納されるOCRおよび投票ファイル	>= 11.2	該当なし	該当なし
デフォルト値以外に設定されるセクター・サイズ	>= 11.2	>= 11.2	該当なし
ディスク・グループに格納されるOracle ASM SPFILE	>= 11.2	該当なし	該当なし
Oracle ASMファイル・アクセス制御	>= 11.2	>= 11.2	該当なし
最大値が1024のASM_POWER_LIMIT	>= 11.2.0.2	該当なし	該当なし
ディスク・グループのコンテンツ・タイプ	>= 11.2.0.3	該当なし	該当なし
ディスク・グループのレプリケーション・ステータス	>= 12.1	該当なし	該当なし
ディスク・グループでの共有パスワード・ファイルの管理	>= 12.1	該当なし	該当なし

主なASM関連機能の拡張

Oracle® Database新機能ガイド 12cリリース1 (12.1)

- 2.7 Oracle RACおよびOracle Grid Infrastructure2.7.1 Oracle ASMの拡張
  - 2.7.1.1 Oracle Flex ASM
  - 2.7.1.2 ディスク・グループでのOracle ASMの共有パスワード・ファイル
  - 2.7.1.3 Oracle ASMリバランスの拡張
  - 2.7.1.4 Oracle ASMディスク再同期化の拡張
  - 2.7.1.5 Oracle ASM chown、chgrp、chmodおよびオープン・ファイルのサポート
  - 2.7.1.6 Oracle ASMでのALTER DISKGROUP REPLACE USERのサポート
  - 2.7.1.7 Enterprise ManagerでのOracle ASM機能のサポート
  - 2.7.1.8 WindowsでのOracle ASMファイル・アクセス制御
  - 2.7.1.9 個別パッチに関するOracle Grid Infrastructureのローリング移行

[12.1] Oracle Flex ASM

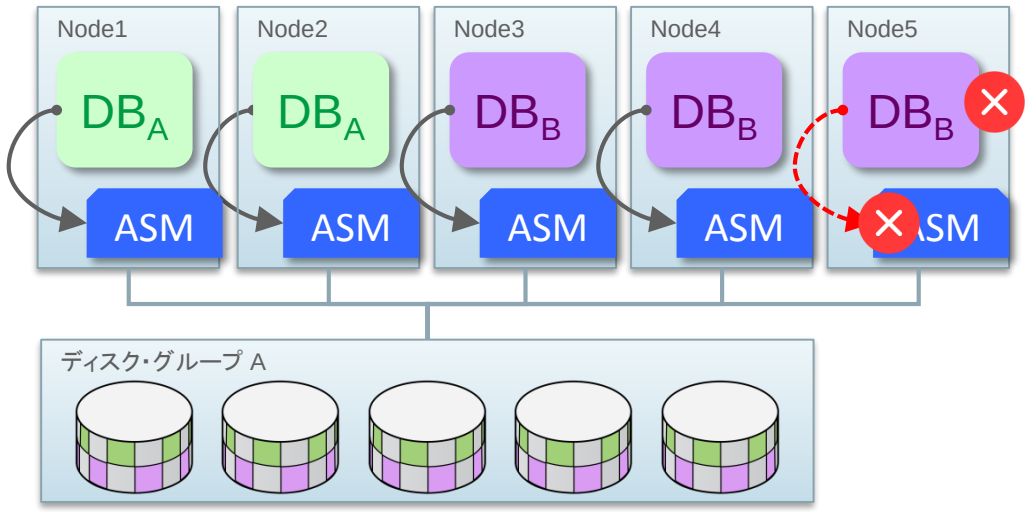
ASM の柔軟な構成による可用性の向上

- ASM インスタンスをデータベース・インスタンスが稼働するサーバーと分離して稼働
  - データベース・インスタンスはネットワーク経由でASM インスタンスにリモート接続
  - クラスタ全体でデフォルトで、3つの ASM インスタンスが起動
  - クラスタ稼働中に ASM インスタンス数を変更することも可能
  - クラスタ全体で ASM によるリソース(メモリー、CPU、ネットワークなど)使用量を低減
  - 障害ポイントの削減
- ASM インスタンスの障害発生時、別の ASM インスタンスへフェイル・オーバー
  - ASM インスタンスへの依存性が緩まり、データベース・サービスの可用性が向上
  - 手動で接続している ASM インスタンスを切り替えることも可能

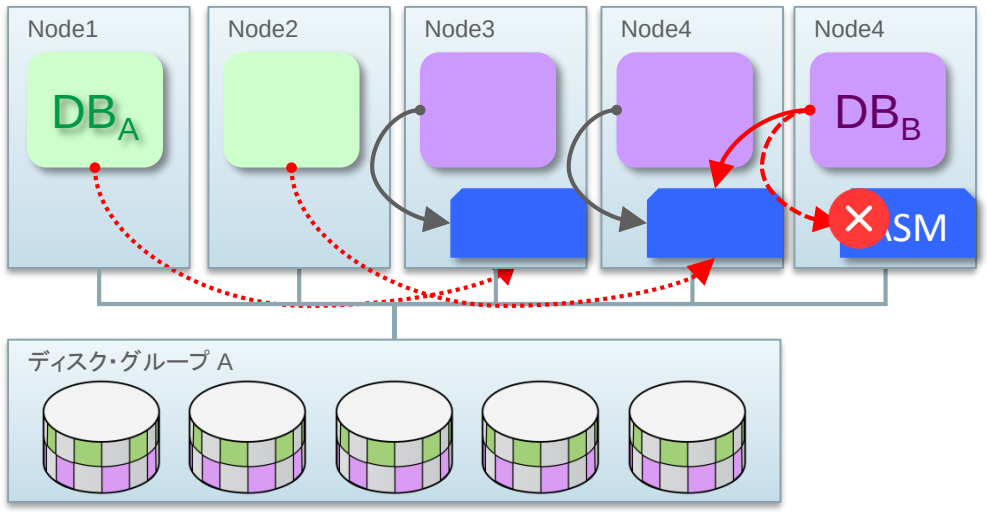
[12.1] Oracle Flex ASM

ASMインスタンス障害発生時の挙動

- 従来のASM構成（左図）では、ASMインスタンスが障害でDownした場合、同一Node上のDBインスタンスもDownする仕様
- Flex ASM構成（右図）では、別Node上のASMインスタンスへ再接続が可能



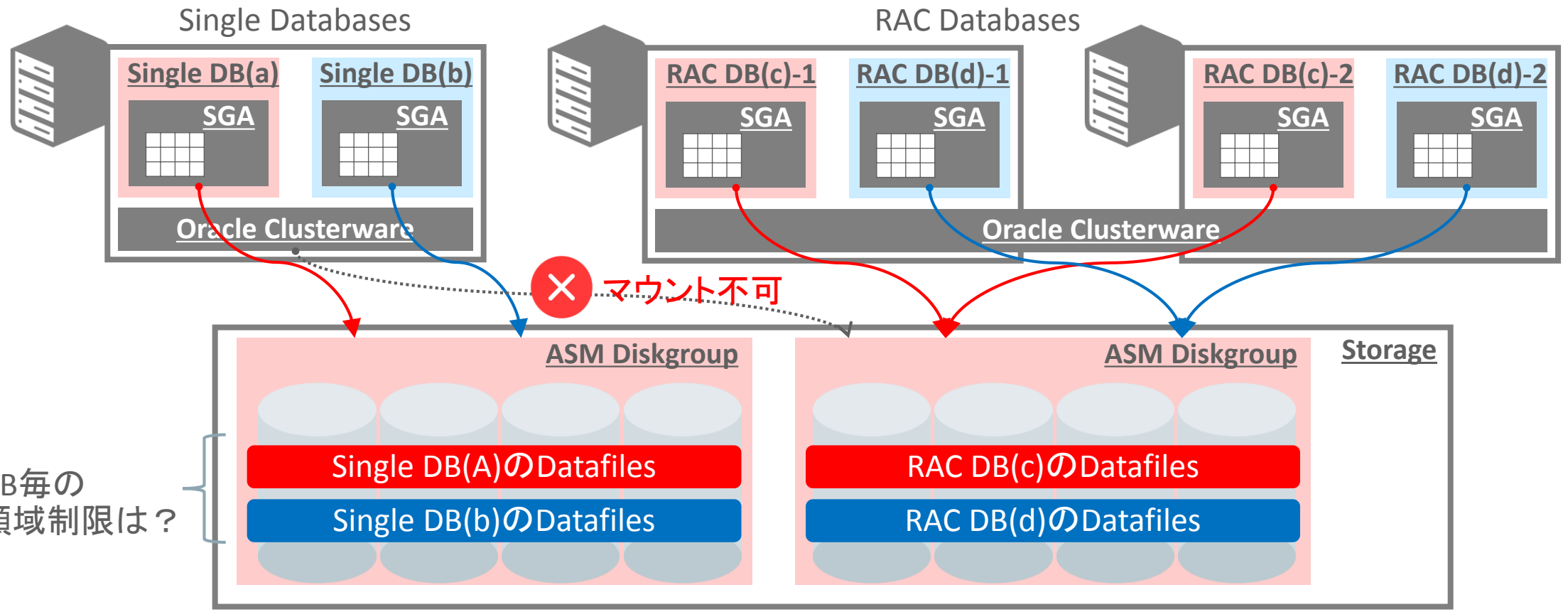
従来のASM構成



Flex ASM構成

【参考】データベース統合とASM Diskgroup

同一Oracle Clusterware上のOracle Databaseは、同一のASM Diskgroupを利用可能



[12.1] Oracle ASMリバランスの拡張

より効率的なディスク交換

- ディスク障害が発生し、ディスク交換後の操作を 1 つのコマンドで実施可能
- ALTER DISKGROUP <disk_group> REPLACE DISK 文が実装
  - 交換するディスクの DROP 操作は不要 (OFFLINE 操作は必要)
    - 従来リリースでは、交換するディスクを DROP した後に新しくディスクを追加する必要があった
  - 交換するディスクには、ミラーされたデータを基にデータが配置される
    - 不要なリバランス処理の実行を回避し、効率よいディスクの交換作業が可能
  - 新しいディスクを元のディスクと同じ名前で追加され、元のディスクと同じ障害グループに割り当てられる

```
SQL> ALTER DISKGROUP DATA REPLACE DISK DATA_0001 with '/dev/sdz';
```

[12.1] Oracle ASMリバランスの拡張

優先順位と同時処理

- リバランス処理を重要なファイルから順に実施
  - 制御ファイルや REDO ログ・ファイルなどを優先してリバランス処理を実施
  - 以前のリリースまでは、file 番号順に実施されていた
- 複数のディスク・グループのリバランス処理の並列実行
  - リバランス処理が完了するまでの時間を短縮
  - 以前のリリースまでは、リバランス処理はシリアルに行われた
  - 複数のディスク・グループに対して同時にリバランス処理がリクエストされた場合は、後からリクエストされた処理はキューで待機
- リバランス処理時に、内部的にエクステントの論理チェックするように設定可能
  - 破損を検知した場合、ミラーされているデータから自動で修正
  - CONTENT.CHECK ディスク・グループ属性で設定

[12.1] Oracle ASMリバランスの拡張

リバランスの詳細な見積もり

- リバランス処理で移動する割当て単位(AU)の数を見積もることが可能
  - EXPLAIN WORK コマンドを使用して work plan を生成
  - work plan は STATEMENT_ID で識別される

```
SQL> EXPLAIN WORK SET STATEMENT_ID='Drop DATA_0001'
  2  FOR ALTER DISKGROUP DATA DROP DISK DATA_0001;
Explained.
```

- 見積もった AU の数を V$ASM_ESTIMATE ビューから確認

```
SQL> SELECT EST_WORK FROM V$ASM_ESTIMATE
  2  WHERE STATEMENT_ID='Drop DATA_0001';
   EST_WORK
-----------
         279
```

[12.1] Oracle ASMリバランスの拡張

再同期、リバランス処理の進行状況と見積もりの確認

- 再同期、リバランス処理の各操作の進行状況と見積もりの詳細は、V$ASM_OPERATION より確認することが可能

```
SQL> SELECT PASS, STATE, SOFAR, EST_WORK, EST_MINUTES
  2  FROM V$ASM_OPERATION;
PASS        STATE                SOFAR   EST_WORK EST_MINUTES
----------- ------------ ---------- ---------- -----------
RESYNC      DONE                  0          0           0
REBALANCE   RUN                1658       4813           2
COMPACT     WAIT                  0          0           0
```

  - 新しく追加された PASS 列から、RESYNC / REBALANCE / COMPACT の各処理の進行状況を確認可能
  - 従来リリースは OPERATION 列を使用
  - 再同期処理の実行中に内部的にチェックポイントが行われ、途中で終了してしまった場合はチェックポイント時点から自動再開

[12.1] Oracle ASMディスク再同期化の拡張

POWER 句の指定による再同期処理の高速化

- ASM Diskgroupの再同期の処理に割り当てるリソース量を POWER 句により任意に設定することで高速化を実現
  - 従来バージョンまでは、常に「1」で固定
  - 12.1以降、POWER 句で 1 から 1024 まで指定可能
    - 指定しない場合は、ASM_POWER_LIMITパラメータの値

  - 対象ASM Diskgroup内でOFFLINEなASM Diskを全てOnline化

```
alter diskgroup <ASM Diskgroup Name> online all power <n>;
```

  - 対象ASM Diskgroup内の特定のASM DiskをOnline化

```
alter diskgroup <ASM Diskgroup Name>
            online disk <Disk Name> power <n>;
```

ストレージからの
ASM Diskの切り出し方法ガイド

ハードディスク･ドライブの性能特性

内周よりも外周の方がI/O性能が高い

- 一般的に、デバイスの先頭が外周
  => ASM Diskの先頭が外周

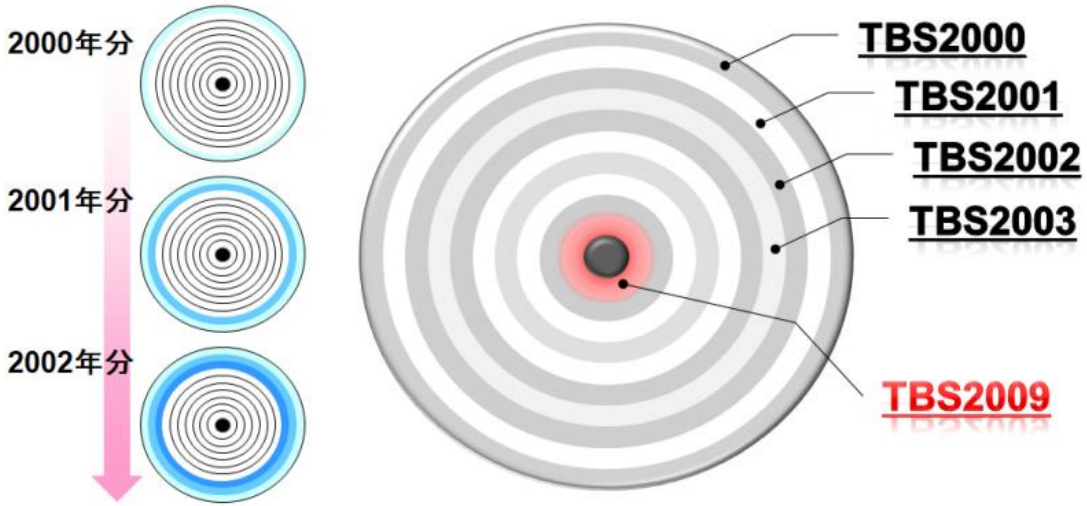


図 7　従来の各表領域とディスク・ドライブ上の物理配置

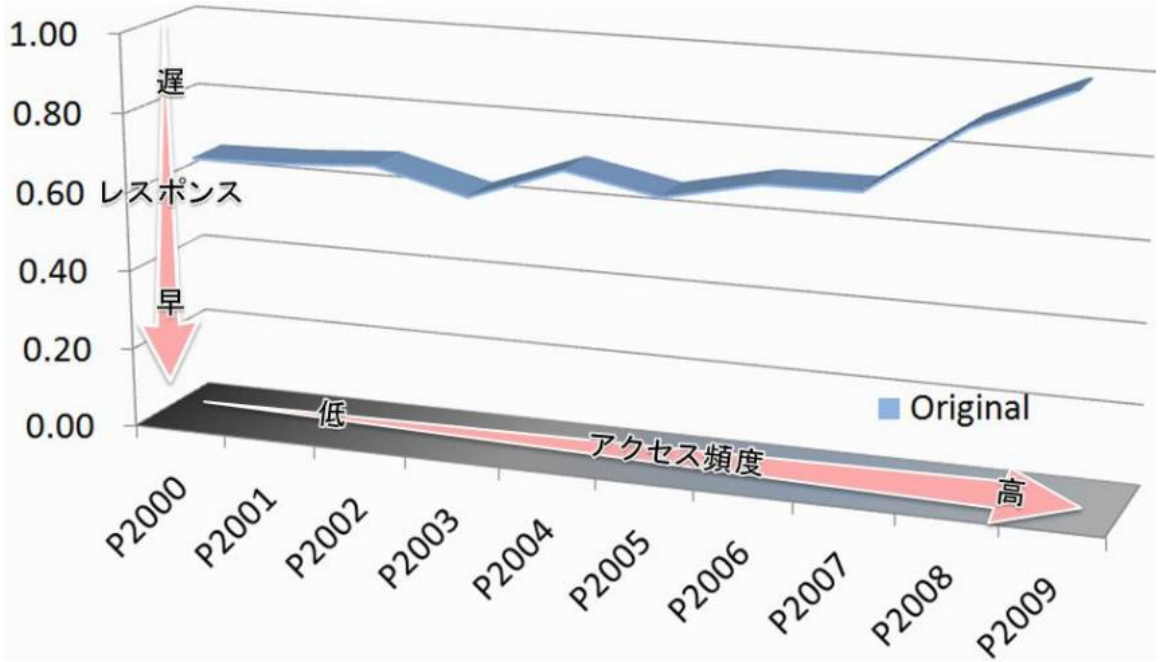
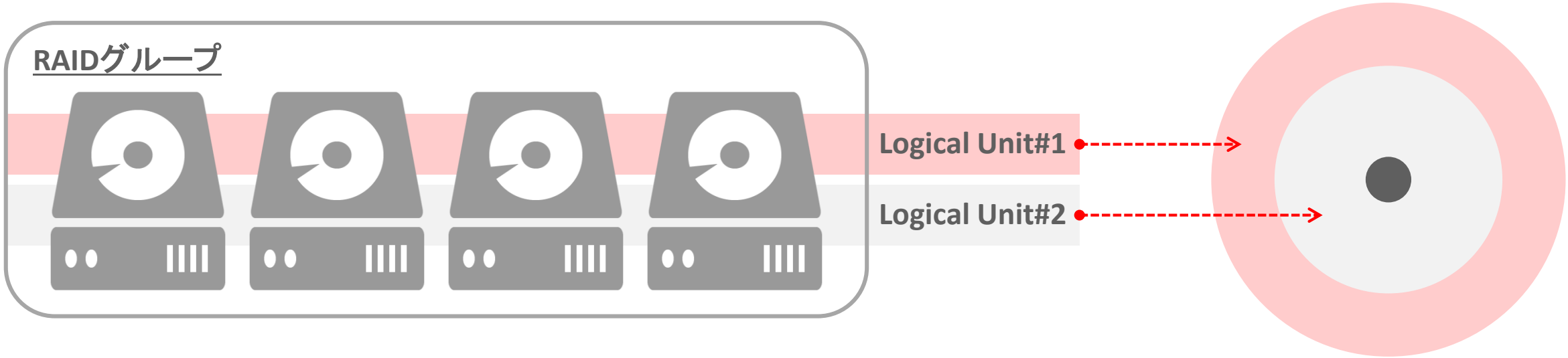


図 8　従来構成時のアクセス頻度と検索性能

【参考】ASM Intelligent Data Placementによるパフォーマンス･チューニング
http://www.oracle.co.jp/solutions/grid_center/nssol/pdf/wp-idp-gridcenter-nssol_v1.0.pdf

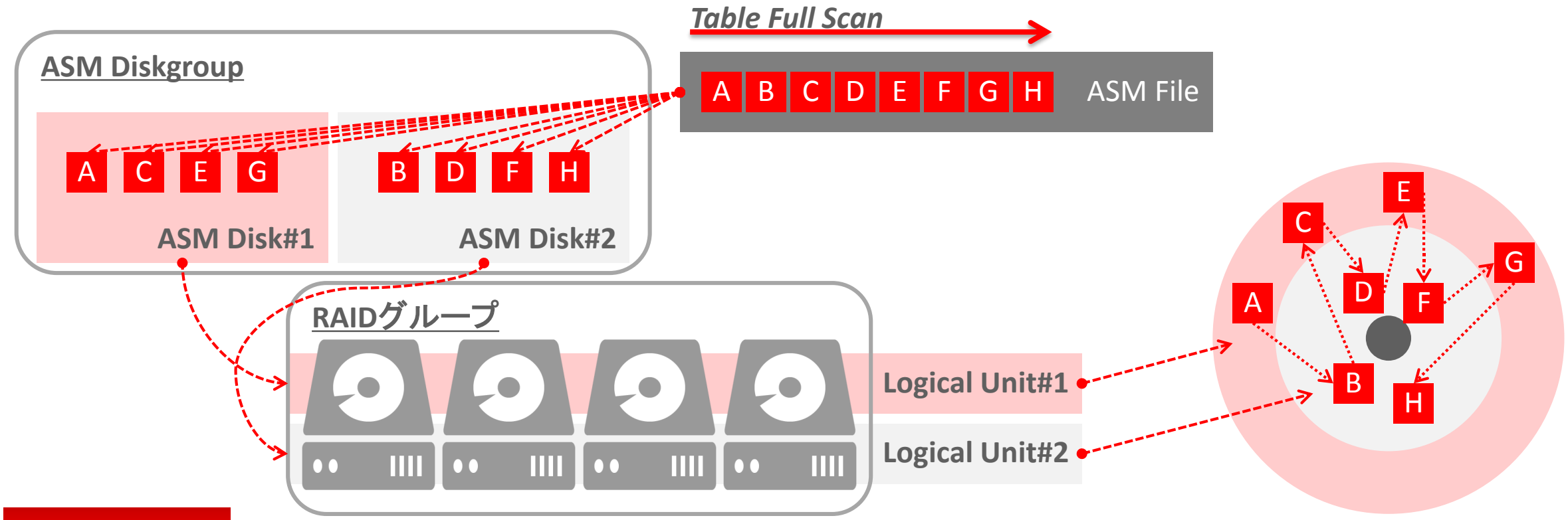
RAIDグループからの切り出す順番と配置

- 一般的に、
  ハードディスクの外周から切り出すと言われているので、
  先に切り出したLogical Unitの方が性能が高い



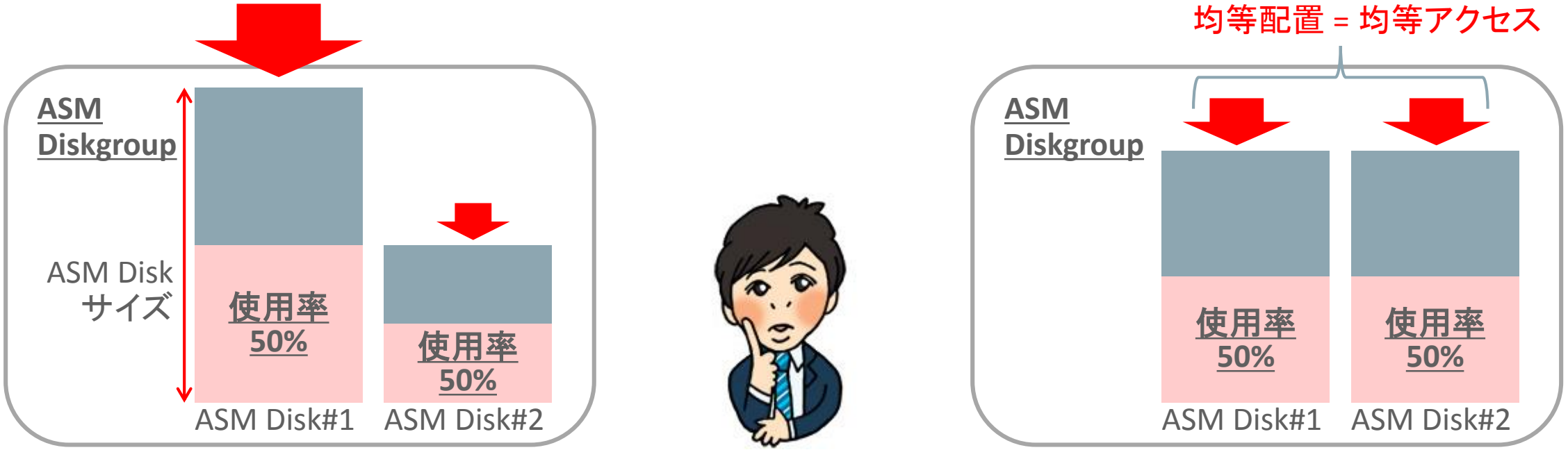
HDDの同一RAIDグループから切り出したLUを同一ASM Diskgroupへ組み込むのは？

- ディスクの外周と内周を行ったり来たりすることで、シークによる待機時間の比率が高まる為に好ましくない



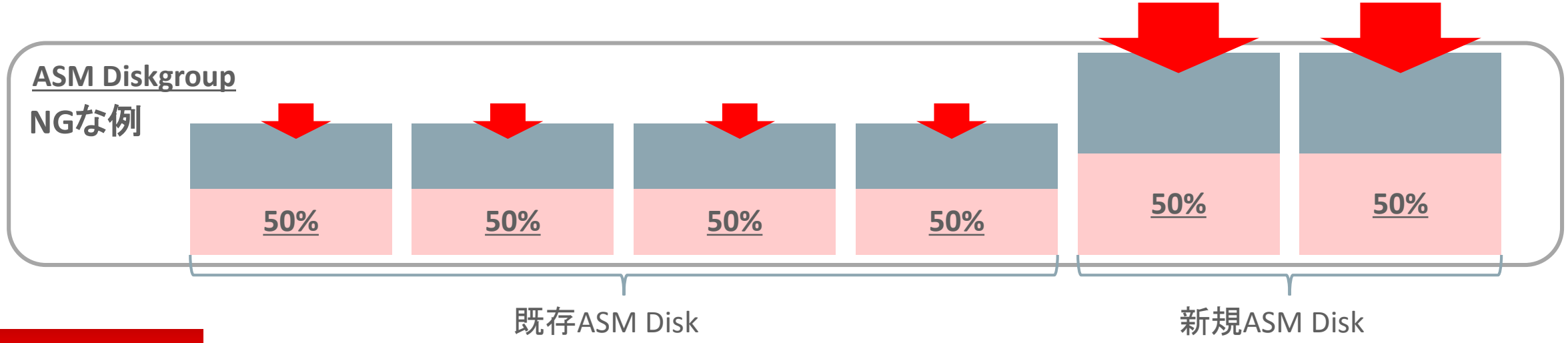
ASM Diskgroupに所属するASM DiskへのI/O特性

- ASMは、各ASM Diskの使用率を「均等化」する
  - サイズが異なるASM DiskでASM Diskgroupを構成した場合、
    サイズが大きなASM Diskへ、より多くのデータが格納される（より多くアクセス）



追加ハードディスク･ドライブの容量が現行より大きい場合

- パフォーマンス最適化の観点から、追加ハードディスクから切り出すLogical Unit（ASM Disk）のサイズは現行のASM Diskと統一すること
  - 容量が勿体ないのは理解できますが、性能劣化（偏り発生）の可能性を避けるべきです
- 余っている領域も、別のLogical Unitとして切り出して、別のASM Diskgroupを構成可能
  - 但し、バックアップや古いデータの格納領域のように、
    日中業務の負荷ピーク時間帯にアクセスが少ない用途として活用すべし



現行HDDで運用、新規でSSDの追加が決定したら

- データベース全体をSSD上に移設できる容量の場合
  - ASM Diskgroupを丸ごと移行

- 一部のデータしか移動できない場合
  - 新規でSSDのみで構成されるASM Diskgroupを作成し、次のどちらかの方法で活用
  1. 特定のオブジェクトや表領域をSSD上に配置

     → AWRレポートを元に、手動でチューニング

     第40回 AWRレポートを読むステップ2:アクセス数が多い表領域とセグメント
  2. Database Smart Flash Cacheの活用

     → Buffer Cacheから追い出されたブロックを自動的にSSD上にキャッシュ

【参考】新規デバイス上へのデータ移行方法例

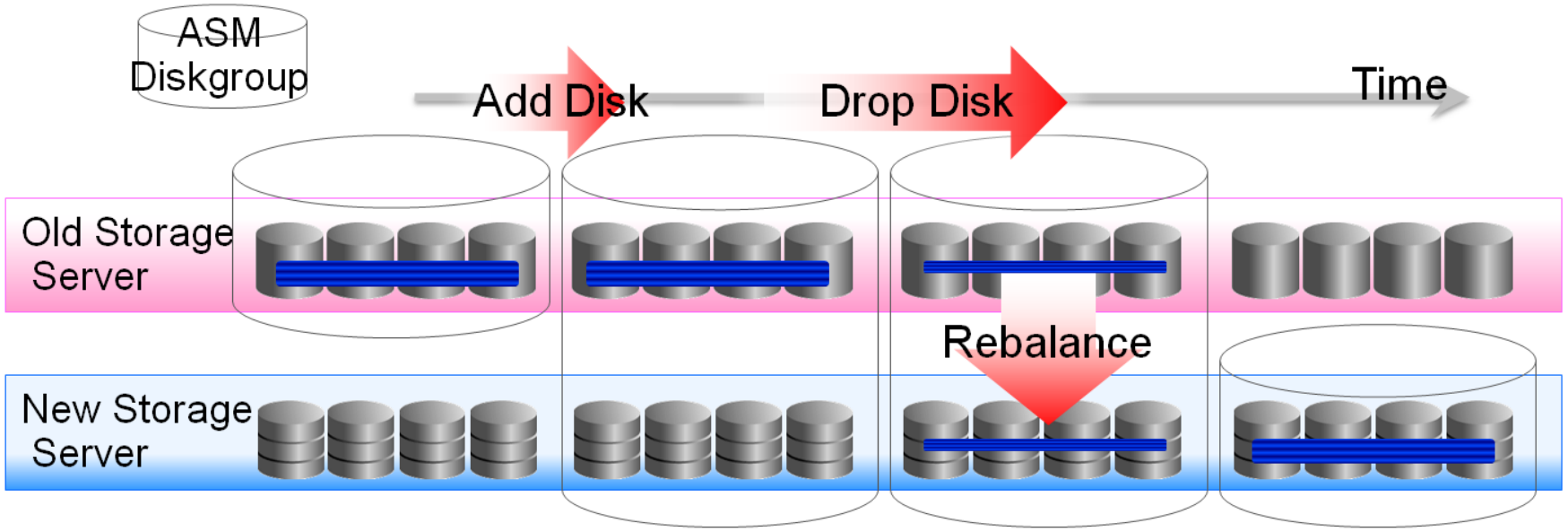
Level	完全無停止での移行方法	一時停止やオフラインを伴う移動方法
Table	表のオンライン再定義 [12.1] オンラインでのパーティション移動	Datapump Export /Import 表、パーティションの移動（alter table move文）
Pluggable Database		Unplug & Plug
Tablespace / Datafile	[12.1] オンライン・データファイルの移動 （alter database move datafile文）	オフライン・データの移動 （alter tablespace rename datafile文） トランスポータブル表領域 RMAN COPY + SWITCH DATAFILEコマンド
Database		オフライン・データファイルの移動 （alter database rename datafile文） RMAN COPY+SWITCH DATABASEコマンド RMAN Duplicateコマンド
ASM Diskgroup	ASM Diskの追加+削除+リバランス	

第47回 [Oracle Database 12c] オンライン・データファイルの移動

【参考】システム無停止で、ストレージ筐体の入替

Automatic Storage Managementの自動リバランスによるストレージ・マイグレーション

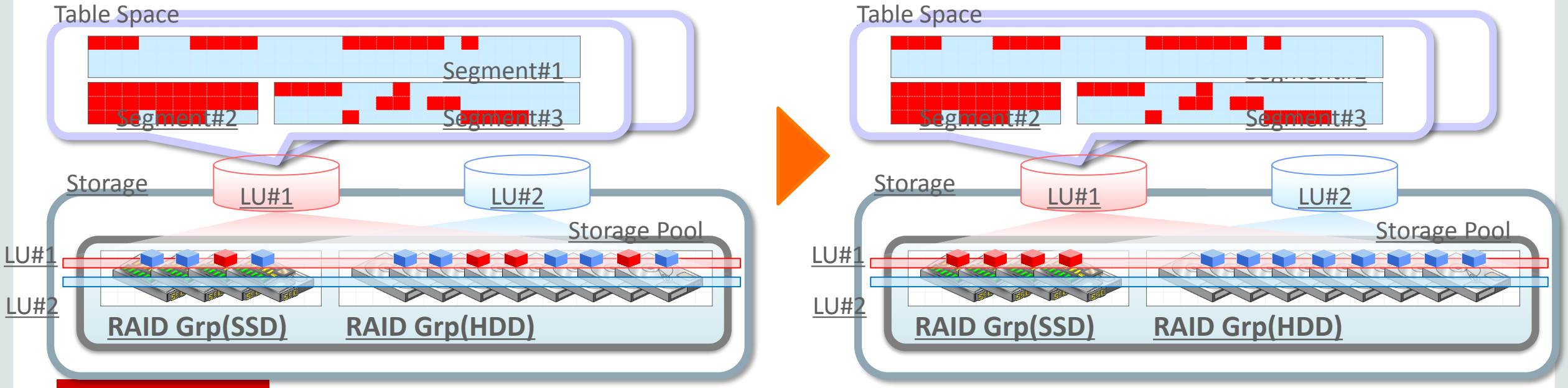
http://www.oracle.com/jp/gridcenter/partner/nssol/wp-storage-mig-grid-nsso-289788-ja.pdf



ストレージ製品の自動階層化機能

Oracle ASMと組合わせて使用することは可能だが・・・

- 過去のI/O実績を分析し、あるタイミングで自動的にデータを再配置
  - アクセス頻度に応じて、適切なデバイス（SSD or HDD）にデータ移動
  - Oracle DatabaseやASMからは透過的



ストレージ製品の自動階層化機能

Oracle Databaseで使用する際の注意点(1)

- 過去のI/O実績を分析し、あるタイミングで自動的にデータを再配置
  - 過去の全てのI/O実績が平常運用時のI/O要求とは限らない
    - アクセス・パターン（日次or月次、日中 or 夜間、一時表）により データへのアクセス時間が偏る場合もある
    - メンテナンスやトラブル時に発行したI/O要求も含まれる
  - ASMのリバランスにより、LU間でデータが移動して上書きされた場合、過去のI/O実績は無意味なものになっている

→ 過去のI/O実績だけに基づく最適化は難しい

ストレージ製品の自動階層化機能

Oracle Databaseで使用する際の注意点(2)

- 過去のI/O実績を分析し、あるタイミングで自動的にデータを再配置
  - データの再配置はデバイスやコントローラーのCPUに負荷が発生する
    - I/O要求のタイミングと再配置のトランザクション影響の考慮した設計が必要となる

→ データ再配置のタイミングや周期の設計が難しい

ストレージ製品の自動階層化機能

Oracle Databaseで使用する際の注意点(3)

- 過去のI/O実績を分析し、あるタイミングで自動的にデータを再配置
  - データベース構造を意識せずにストレージ側で再配置される
    - 一つの表データがSSD, HDDの両方に分かれて配置された場合、セグメント（表／索引）単位、表領域（データファイル）単位のI/O性能は平均値となるため、ボトルネックの特定が困難
  - データベースとストレージの性能情報を突き合わせた複雑な分析
    - データ配置を含めて問題発生時の状況を再現するのは困難
  - ASMのリバランスにより、LU間でデータが移動して上書きされるため、自動階層化機能側での最適化の効果が薄れる可能性有り

→ 性能問題の原因切り分けが長期化

最後に





しばちょう流 Oracle ASMにおけるストレージ設計指針

2016年8月版

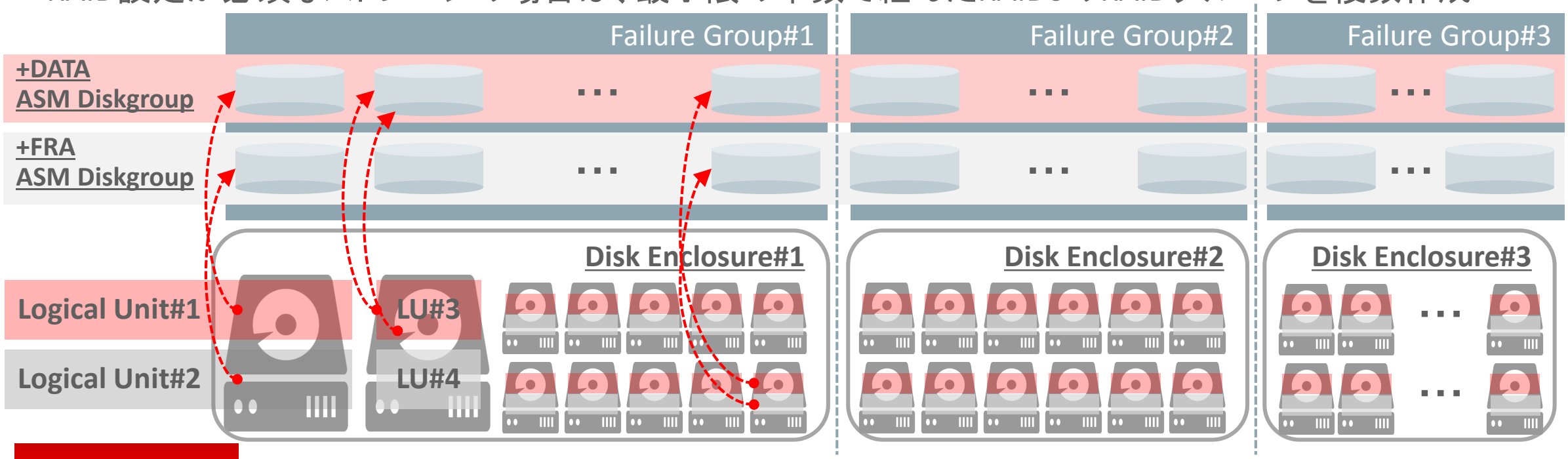
1. 一つのRAID Groupは、二つ+αのLogical Unitを切り出す
   - ✓ 最初のLUのみでオンライン・データファイル用のASM Diskgroupを構成
   - ✓ 残りのLUで、バックアップ用（高速リカバリ領域）のASM Diskgroupを構成
2. 可能な限り多くのLUを一つのASM Diskgroupで束ねる
3. 各ASM Diskgroupに含めるASM Disk(LU)は同一の性能 &サイズ
   - ✓ デバイス・タイプ（HDDやSSD）に応じて、別ASM Diskgroupを構成
4. RAIDグループと冗長構成パターン例
   - ✓ 二重障害対応構成：「RAID無し or RAID0」× High Redundancy
   - ✓ 単一障害対応構成の場合は、RAID or ASMのどちらか一方で担保

ASM Diskgroup構成例

RAIDグループを構成する場合は、後々のHDDの追加購入単位も考慮しておきましょう

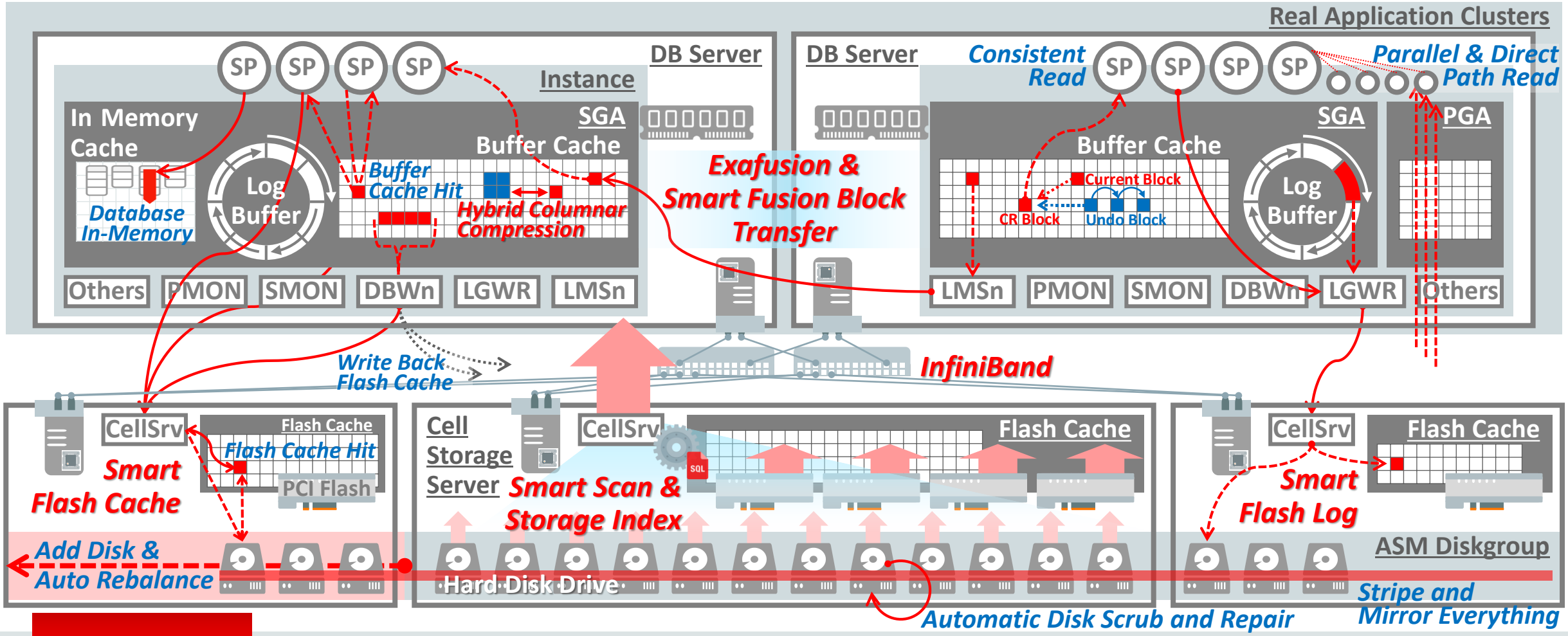
RAIDミラー無しで、ASM三重化が美しい

- 各ASM Diskgroupへ含めるASM Disk（LU）のサイズは統一
- DataFileは+DATAへのみ格納し、負荷ピーク時間帯にシーク時間を極小化（+FRAはバックアップ用）
- RAID設定が必須なストレージの場合は、最小限の本数で組んだRAID0のRAIDグループを複数作成



Oracle Exadata – Database Machine

H/W性能を最大限まで活用可能なアーキテクチャ



しばちょう先生の試して納得！DBAへの道

関連記事一覧

- Automatic Storage Management
  - 第30回 ASMディスク・グループの作成と使用量の確認
  - 第31回 ASMのストライピングとリバランスによるI/O性能の向上
  - 第33回 ASMのリバランスの動作
  - 第34回 ASMのミラーリングによるデータ保護(1) ～障害グループと冗長性の回復～
  - 第35回 ASMのミラーリングによるデータ保護(2) ～高速ミラー再同期～
- I/Oボトルネックのチューニング手法
  - 第39回 AWRレポートを読むステップ1:バッファキャッシュ関連の待機イベントと統計情報
  - 第40回 AWRレポートを読むステップ2:アクセス数が多い表領域とセグメント
- データの移動方法
  - 第41回 [Oracle Database 12c] オンラインでのパーティション移動
  - 第47回 [Oracle Database 12c] オンライン・データファイルの移動
  - 第45回 Recovery ManagerのSWITCHコマンドでリストア時間ゼロ

Safe Harbor Statement

The preceding is intended to outline our general product direction. It is intended for information purposes only, and may not be incorporated into any contract. It is not a commitment to deliver any material, code, or functionality, and should not be relied upon in making purchasing decisions. The development, release, and timing of any features or functionality described for Oracle's products remains at the sole discretion of Oracle.

Integrated Cloud

Applications & Platform Services


ORACLE®