

# Oracle Database 12cとOracle FS1-2フラッシュ・ストレージ・システムを使用したストレージ階層化テクノロジー

Oracle ホワイト・ペーパー | 2015年4月



## 目次

|  |    |
|--|----|
| はじめに .....                               | 2  |
| 表、表領域、およびストレージ .....                     | 3  |
| Oracle Database 12c でデータを移行するための技術 ..... | 5  |
| DBA 開始/管理 .....                          | 5  |
| ILM ヒート・マップおよびデータ自動最適化ポリシー .....         | 6  |
| Oracle FS1-2 のヒート・マップおよび自動階層化 .....      | 9  |
| まとめと推奨事項 .....                           | 12 |

## はじめに

ストレージ階層化、つまりデータとストレージのI/O特性を一致させるように、異なるタイプのストレージ間でデータを移行する機能は、管理者が長年（何十年もの間）使用してきた技術です。この技術を使用するもっとも単純で一般的な理由は、データの経年劣化です。つまり、データ・セクションの存続期間が長くなれば、そのアクセス・パターンが変化し、アクセス頻度が低下する可能性が高くなります。つまり、データはクール状態になり、最終的にはコールド状態になります。コールド・データのI/Oパターンは、ウォーム・データおよびホット・データとは非常に異なっているため、異なるストレージが必要になります。ホット・データの場合でも、異なるストレージが必要になることがあります。OLTPデータベース・ワークロードのデータは、多くの場合I/Oパターンがランダムであるため、パフォーマンス・ソリッド・ステート・ドライブ（SSD）が適しています。一方、意思決定支援のシーケンシャル・ワークロードまたは分析ワークロードは、容量SSDまたは容量ハード・ディスク・ドライブ（HDD）の方が適しています。

長年にわたり管理者が使用できるツールや技術は変化してきましたが、1つだけ変わらないことがあります。それは異なる特徴（価格、パフォーマンスなど）を持つストレージ・メディアの**範囲**です。これは、これまでも現在も同じです。ストレージ・タイプの仕様は変化していますが、範囲があるという事実は変わりません。図1は、Oracle FS1-2で使用可能なメディアの範囲を示しています。

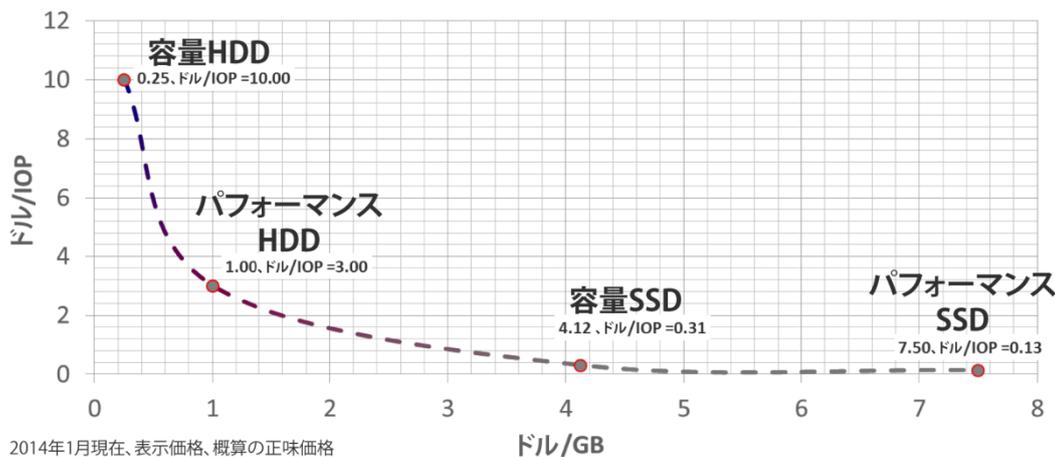


図1：異なるストレージ・タイプのプライス・パフォーマンスの特徴

Oracle Database 12c および Oracle FS1-2 ストレージ・システムが導入されたことにより、管理者のツールに新しい技術が追加されました。このホワイト・ペーパーの最終目標は、これらの技術を比較・対比することです。これらの技術はすべて、有効で強力です。管理者は、このホワイト・ペーパーを参照することにより、それぞれの技術を適切に使用するタイミングを特定できます。

このホワイト・ペーパーでは、最初にデータベースに関する用語とストレージへのマッピング方法について紹介し、説明します。本文では、ストレージ階層化に関する3つの技術について説明します。最後の項では、各技術を使用するタイミングに関する推奨事項を示します。

## 表、表領域、およびストレージ

ストレージ層間でデータベース内のデータを移行する場合、さまざまな技術を掘り下げて考える前に、ストレージ・スタック内の層について理解する必要があります。図 2 に、これらの層を示します。この層は、まず Oracle FS1-2 フラッシュ・ストレージ・システム内に物理ストレージ（容量ハード・ドライブ、パフォーマンス・ハード・ドライブ、容量 SSD、およびパフォーマンス SSD）があり、その上に Oracle FS1-2 が示すボリューム（仮想ディスクまたは LUN）があります。Oracle Automatic Storage Management は、Oracle DataBase の 1 つの機能であり、論理ボリューム・マネージャ機能を実行します。つまり、Oracle FS1-2 が示すボリュームを消費し、データベースに対してディスク・グループとして示します。

データベース・レベルでストレージを抽象化したものが表領域です。特定のデータベースは 1 つ以上の表領域を所有しており、各表領域は基盤のオペレーティング・システムにマップされる 1 つ以上のデータファイル<sup>1</sup>を所有しています。データベースの表は表領域内に存在しており、各表領域は特定の Oracle Automatic Storage Management ディスク・グループにマップされています。

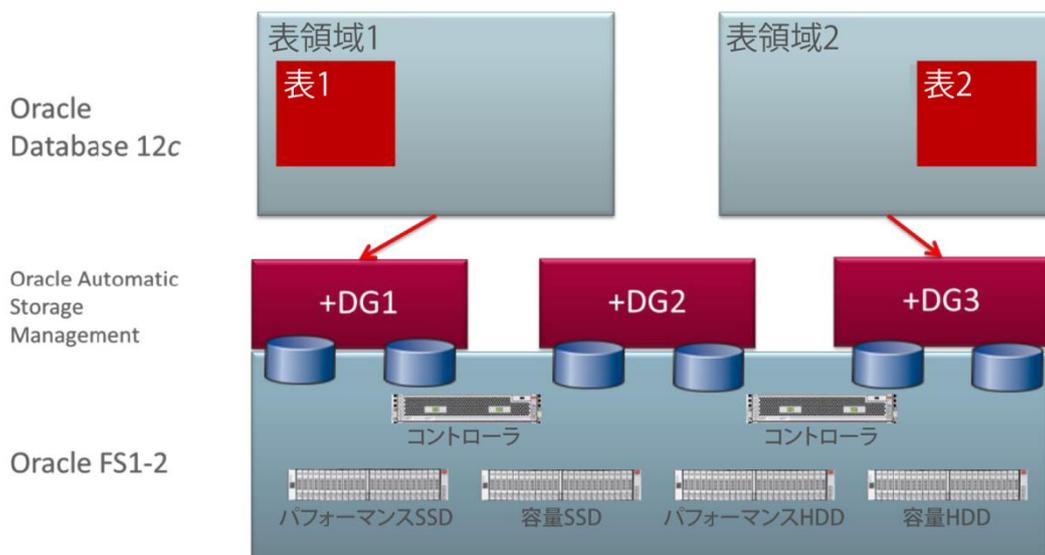
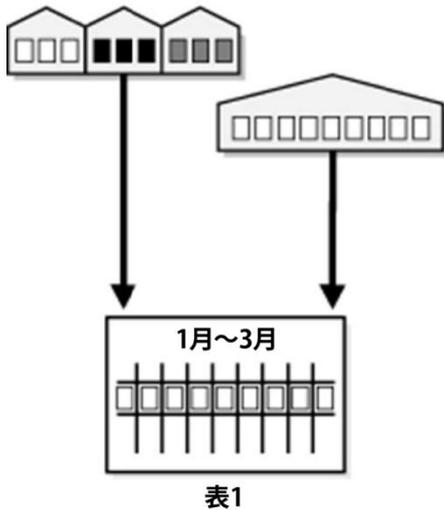


図2：表、表領域、Oracle Automatic Storage Managementディスク・グループ、およびストレージ・ボリューム間の関係

次に、パーティション化の概念について説明します。図2では、表内のすべてのデータがコールド状態になっていれば、DBAは表全体を別のストレージ層に移行させる必要がありますが、データが部分的にコールド状態になっている可能性が高く、このような場合、パーティション化が重要な役割を果たします。

<sup>1</sup> オラクルの管理対象ファイルでは、データベース管理者（DBA）がオペレーティング・システム・ファイルを直接管理する必要はありません。すべての操作は、ファイル名ではなく、データベース・オブジェクトの観点から指定されます。したがって、このホワイト・ペーパーでは、データファイルについては詳しく説明しません。

パーティション化されていない表  
パーティション化されている、  
またはパーティション化されていない  
索引を所有することが可能



パーティション化されている表  
パーティション化されている、  
またはパーティション化されていない  
索引を所有することが可能

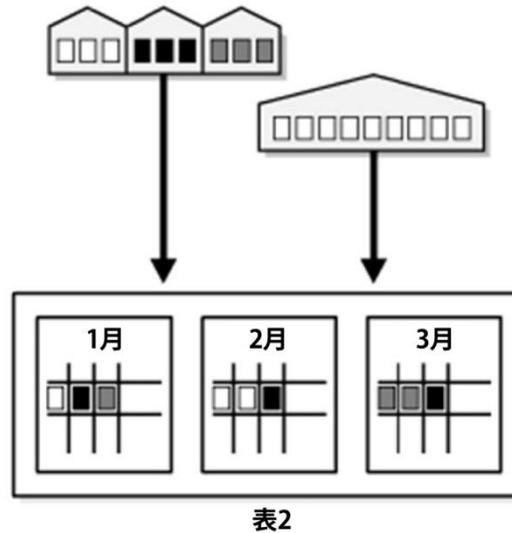


図3：パーティション化された表のイメージ

パーティション化すると、表、インデックス、またはインデックス構成表がさらに小さいピースに分割されますが、このようなデータベース・オブジェクトの各ピースをパーティションといいます。各パーティションは固有の名前を持ち、オプションとして固有のストレージ特性を指定することもできます。

データベース管理者の観点から見た場合、パーティション化されたオブジェクトは一括管理も個別管理も可能な複数のピースを所有しています。これにより、管理者は、パーティション化されたオブジェクトの管理がかなり柔軟に行えます。しかし、アプリケーションの観点からすると、パーティション化された表は、パーティション化されていない表と全く同じものです。SQL 問合せと DML 文を使用して、パーティション化された表にアクセスする際に、変更を加える必要はありません。

パーティション化された表の各行は、一義的に 1 つのパーティションに割り当てられます。パーティション化キーは、各行が格納されるパーティションを決定する 1 つ以上の列から構成されています。Oracle Database 12c はパーティション化キーを使用して、挿入、更新、および削除の各操作に対応するパーティションに自動的に割り当てます。管理者はかなり柔軟にパーティション化キーを選択できますが、図 3 に示すように、もっとも一般的な選択は日付や時間に基づき行います。たとえば、Order Entry (OLTP) データベースでは、管理者は、今四半期の注文を高パフォーマンス・ストレージに保持すると同時に、前四半期の注文（現在はコールド状態）をより低価格でより低パフォーマンスのストレージに移行します。

図 4 は、図 2 に少し変更を加えたもので、パーティション化された表が追加されています。表 3 には 4 つのパーティションが存在し、その中の 2 つはディスク・グループ+DG1 の表領域 1 に配置され、残りの 2 つはディスク・グループ+DG3 の表領域 2 に移行されています。

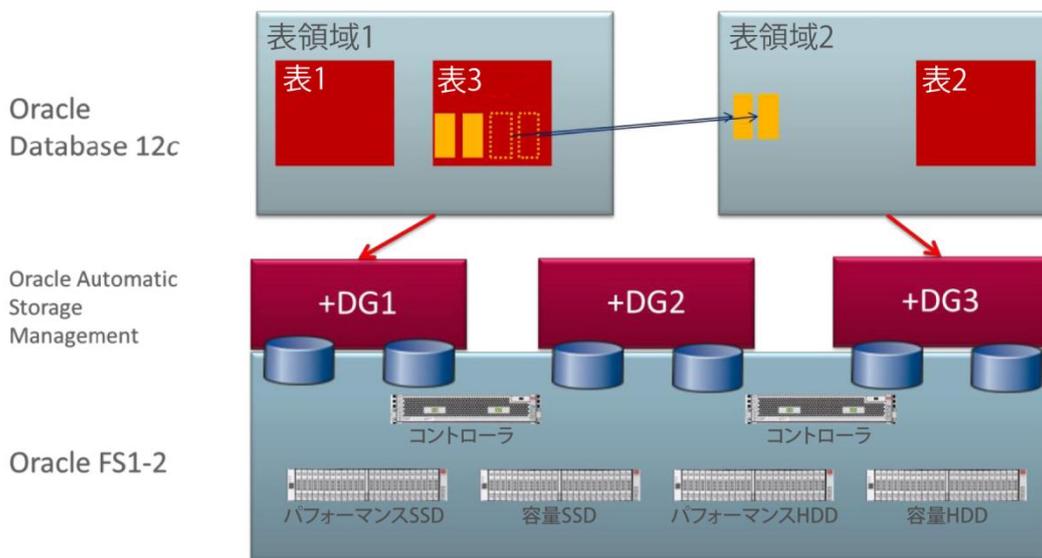


図4：表、パーティション、表領域、およびストレージ間の関係

## Oracle Database 12cでデータを移行するための技術

以下の項では、3つの補完的な技術について説明します。

1. DBA 開始/管理（手動）
2. 情報ライフサイクル管理（ILM）ヒート・マップおよびデータ自動最適化ポリシー
3. Oracle FS1-2 のヒート・マップおよび自動階層化

### DBA開始/管理

これまでの多くの Oracle Database リリースでは、管理者は、表領域間の表の移行を手動で開始できるようにしていました。図5は、その簡単な例を示しています。

図6は、表全体ではなく、表のパーティションを移行している例です。表を移行すると、その表の行の行 ID が変更されます。これにより、表のインデックスに unusable というマークが付きます。この索引を使用して表に DML でアクセスするとエラーが返されるため、図6に示されているように UPDATE INDEXES 句を使用する必要があります。

```
SQL> ALTER TABLE table1
      MOVE
      TABLESPACE new_tablespace;
```

図5：表全体を新しい表領域に移行する例

```
SQL> ALTER TABLE table1
      MOVE PARTITION p1
      TABLESPACE new_tablespace
      UPDATE INDEXES;
```

図6：表のパーティションを新しい表領域に移行する例

表全体または表のパーティションを移行することは単純かつ簡単な操作であり、DBA による直接実行や（図 5 と図 6 の簡単な例を参照）、スケジュール済みジョブの実行が可能です。ストレージ階層化の場合、データがコールド状態になるタイミングを管理者が正確に認識していれば、この操作は明らかに非常に有効な技術です。このケースでは、多数の実例が存在します。たとえば、管理者はご存じかもしれませんが、新しい四半期の最初の日が来ると、前の四半期の注文情報にはほぼ（または、まったく）アクセスしなくなるため、前の情報をより低価格のストレージに移行できます。

#### ILM ヒート・マップおよびデータ自動最適化ポリシー

前の項では、表およびパーティションを手動移行する方法について紹介しました。これは、パーティションの移行準備が完了したタイミングを正確に認識している場合には、有効な手段です。しかし、パーティションの移行準備が完了したタイミングを正確に認識していない場合はどうでしょうか。

この場合は、前項の例が役に立ちます。つまり、新しい四半期が開始されたときに、前の四半期のパーティションを移行するということです。しかし、注文は完了するまでに数日を要することがあります。このため、前の四半期の最終日付近に発行された注文は、新しい四半期で数日が経過するまでは、実際には完了していません。新しい四半期に入って 2 日後である可能性も、3 日後である可能性もあります。パーティションのアクティビティを実際に監視しておらず、注文のすべてがコールド状態になったことを確認していない場合、DBA はそのパーティションを移行する適切なタイミングを判断できません。この場合、Oracle Database 12c の新しい機能である、ヒート・マップおよびデータ自動最適化ポリシーが役に立ちます。

ILM ヒート・マップは、行およびセグメントのレベルで使用状況の情報を自動的に追跡します。データの変更時間は行レベルで追跡され、ブロックレベルに集約されます。また、変更時間、全表スキャン時間、およびインデックス参照時間も、セグメント・レベルで追跡されます。ヒート・マップでは、データのアクセス状況、および時間とともにアクセス・パターンが変化している様子を示すサマリー・ビューが使用できます。

ヒート・マップは、データベースで明示的に有効にする必要があります（図 7 を参照）。

```
SQL> ALTER SYSTEM SET HEAT_MAP = ON;
```

図7：ヒート・マップの有効化

プログラムによる ILM ヒート・マップ・データへのアクセスは、PL/SQL テーブル・ファンクションおよびデータ・ディクショナリ・ビューを通して可能になります（図 8 の例を参照）。

```
SQL> SELECT * FROM USER_HEAT_MAP_SEGMENT WHERE OBJECT_NAME='ORDERS' ;
```

| OBJECT_NAME | SUBOBJECT_NAME | SEGMENT_WRITE_TIME | SEGMENT_READ_TIME | FULL_SCAN         | LOOKUP_SCAN       |
|-------------|----------------|--------------------|-------------------|-------------------|-------------------|
| 1 ORDERS    | O_FIRST        | 01/29/15 10:47:44  | (null)            | 02/01/15 13:41:48 | 01/29/15 10:47:44 |
| 2 ORDERS    | SYS_P1584      | 01/31/15 00:31:23  | (null)            | (null)            | 01/30/15 00:48:08 |
| 3 ORDERS    | SYS_P1605      | 02/01/15 00:41:32  | (null)            | (null)            | 01/31/15 00:31:23 |
| 4 ORDERS    | SYS_P1614      | 02/02/15 13:27:15  | (null)            | (null)            | 02/02/15 00:42:03 |
| 5 ORDERS    | SYS_P1633      | 02/02/15 00:42:03  | (null)            | (null)            | (null)            |
| 6 ORDERS    | SYS_P1641      | 02/02/15 13:27:15  | (null)            | (null)            | (null)            |

図8：データベース・ビューを使用した、ILMヒート・マップ・データへのアクセス

データの自動最適化により、データ圧縮およびデータ移行用のポリシー作成が可能になり、圧縮の有無に関係なくストレージの階層化が実装できます。Oracle Database は、データ自動最適化ポリシーを定期的に評価し、ILM ヒート・マップが収集した情報を使用して、実行する操作を決定します。データ自動最適化操作はすべて、バックグラウンドで自動的に実行され、ユーザー側の操作は必要ありません（注：このホワイト・ペーパーでは、ストレージ階層化を対象にしているため、ヒート・マップおよびデータ自動最適化の圧縮機能については、これ以上説明しません）。管理者は、この非常に有効な機能について、個別に調査および評価されることを推奨します。

データ自動最適化ストレージ・ポリシーは、表および表パーティションについてセグメント・レベルで指定できます。ポリシーは、メンテナンス・ウィンドウの期間にバックグラウンドで自動的に評価および実行されます。また、データ自動最適化ポリシーは、DBA、手動、またはスクリプトを介して常時、評価および実行できます。

データ自動最適化ポリシーは、データ自動最適化操作を開始する（データ・アクセスの）条件（アクセスなし、変更なし、作成時間など）および、ポリシーが有効になるタイミング（ $n$  日後、 $n$  か月後、 $n$  年後など）を指定します。

データ自動最適化ポリシーのもっとも単純な形式が表領域の使用状況によって引き起こされます。図 9 のように、上限（“Tablespace Used (%)”）を設定します。この上限を超えると、ポリシーは領域を解放しようとして表領域からデータを移行します。ポリシーの実行（および、データの移行）は、下限（“Tablespace Free (%)”）に達するまで継続して行われます。図 10 は、管理者が Oracle Enterprise Manager を使用して、このようなポリシーを追加する方法を示しています。このケースでは、DBA はセグメント・レベルのヒート・マップを使用することを選択しており（ストレージ階層化ポリシーでは、行レベルは使用不可）、これに“Automatic tablespace fullness detection”を実行基準として指定したストレージ階層化ポリシーを組み合わせています。

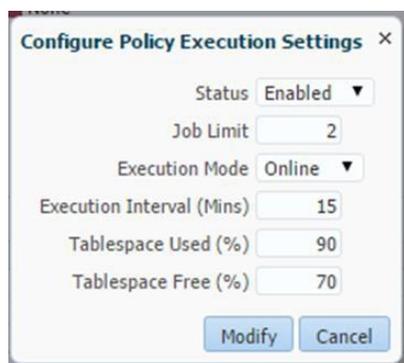


図9：ポリシー実行の設定

## Add Automatic Data Optimization Policy

Automatic Data Optimization (ADO) allows you to define compression or storage tiering policies to automatically compress or move data to opti

図10: Oracle Enterprise Managerを使用したデータ自動最適化ポリシーの追加

図 10 に示されているようなポリシーは、簡単に設定でき（特に、Oracle Enterprise Manager を使用した場合）、有効に機能します。ただし、ILM ヒート・マップおよびデータ自動最適化の真の能力は、データ移行関数をカスタムで使用することにより引き出すことができます。

図 11 は、このようなデータ移行関数の例を示しています。関数は、PL/SQL でコード化されており、引数が 1 つあります（オブジェクト ID）。関数は単一の値を返します。True の場合、データの移行が行われますが、False の場合は行われません。この例では、関数は、USER\_HEAT\_MAP\_SEGMENT（特定ユーザーのオブジェクトのみを表示している、ILM ヒート・マップのデータベース・ビュー）から SEGMENT\_WRITE\_TIME（最後にオブジェクトが書き込まれた時間を記録）を調べ、この 3 日間に書き込まれていないオブジェクトを探します。

図 12 は、move3d 関数（図 11 で定義）を使用して、データ自動最適化ストレージ階層化ポリシーを追加する（このケースでは、orders 表に対して追加）SQL コマンドを示しています。この関数は（指定されたメンテナンス・ウィンドウ期間中に）1 日に一度評価されます。関数が True を返した場合、その対象となった表またはパーティションはすべて、tier\_1c 表領域に移行されます。

```
SQL> CREATE OR REPLACE FUNCTION MOVE3D
( OBJ_ID IN NUMBER )
RETURN BOOLEAN AS
  w_time date;
  now date;
BEGIN
  select SEGMENT_WRITE_TIME into w_time
  from user_objects u join user_heat_map_segment h
  on u.object_name=h.object_name and u.subobject_name=h.subobject_name
  where u.object_name=obj_id;
  now := systimestamp;
  if (trunc(now - w_time) > 2) then RETURN TRUE; else RETURN FALSE; end if;
END MOVE3D;
```

図11: カスタムのデータ移行関数

```
SQL> ALTER TABLE orders ILM ADD POLICY TIER TO tier_1c ON move3d;
```

図12: カスタム関数によるデータ自動最適化ポリシーの追加

## Oracle FS1-2のヒート・マップおよび自動階層化

ILM およびデータ自動最適化は、データベース・レベルで動作します。つまり、ILM ヒート・マップはデータベース・オブジェクト（表領域、表、パーティションなど）へのアクセスを測定して記録し、データ自動最適化はデータベース表およびパーティションに関するポリシーです。この機能は強力ですが、データ移行の粒度は通常、表全体またはパーティション全体に対して数 GB または数十 GB になります。しかし、表が適切にパーティション化された場合でも、ホット・データおよびコールド・データが混在しているパーティションが依然として存在するとしたらどうでしょうか。つまり、ヒート・クラスタの粒度が、表全体またはパーティション全体と比較してかなり小さいということです。この場合は、Oracle FS1-2 のヒート・マップおよび自動階層化が、重要な役割を果たすことができます。

Oracle Database 12c と Oracle FS1-2 は両方とも、データ使用状況を追跡してヒート・マップを作成していますが、追跡する内容は非常に異なっており、データの使用方法も異なります。表 1 では 2 つのヒート・マップを比較していますが、おもな相違点は、ILM は表またはパーティション・レベルでアクセス属性を追跡するのに対して、Oracle FS1-2 は 640KB のサブ LUN 粒度レベルでアクセスを追跡するという点です。通常、これはパーティションよりかなり小さいため、データをより小さなまとまりで移行できます。

表1：ILMヒート・マップとOracle FS1-2ヒート・マップの比較

|                   | ILM ヒート・マップ                                   | Oracle FS1-2 ヒート・マップ  |
|-------------------|---|---|
| 粒度                | 表またはパーティション (GB)                              | 640KB   |
| データベース・オブジェクトでの動作 | はい (表およびパーティション単位)                            | いいえ (ディスクまたは SSD ストレージでのデータのブロック単位)   |
| 収集される属性           | 最後の読み込み/書き込み、最後の参照、最後のスキャン                    | 長期間にわたるアクセス密度、読み込み/書き込み比率、ランダム操作対シーケンシャル操作など  |
| 収集したデータの用途        | DBA 開発のポリシーは、データベース・ビューを介してヒート・マップ・データにアクセス可能 | 管理者は、LUN が単一層であるか自動階層であるかを定義。自動階層 LUN の場合、FS1 システムは、内部ヒート・マップ・データを使用して、ストレージ層間でデータを移行 |

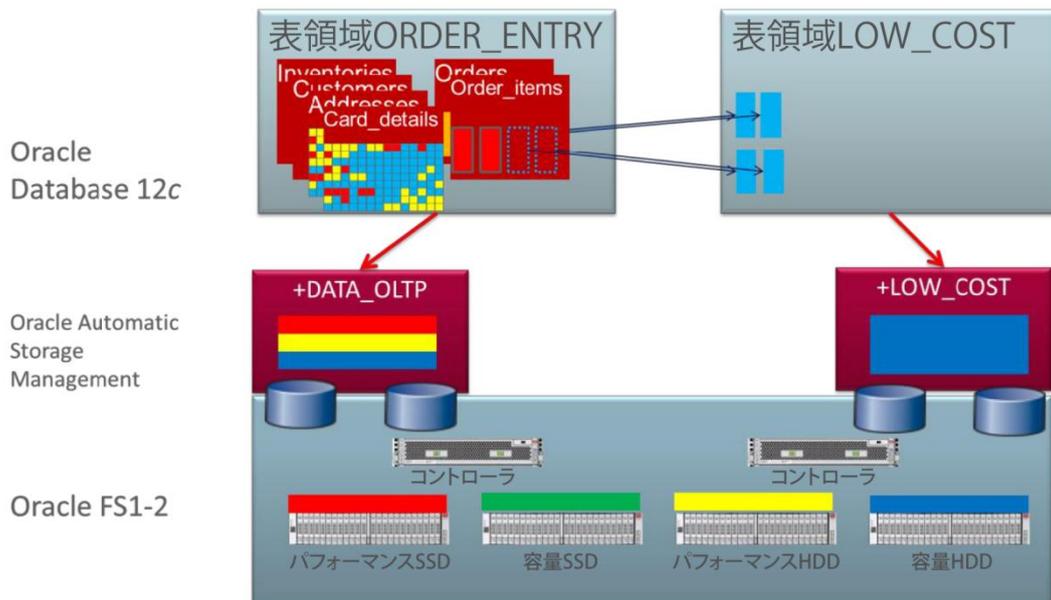


図13 : Oracle Database 12cでのOracle FS1-2自動階層化の使用

図 13 は、図 4 で示した例を拡張して、Oracle FS1-2 自動階層化を追加しています。異なるタイプのストレージを識別するために、次の色分けを使用しています。パフォーマンス SSD、容量 SSD、パフォーマンス HDD、および容量 HDD。また、以下の 2 つの Oracle Automatic Storage Management ディスク・グループが存在しています。

1. +LOW\_COST は、容量ハード・ドライブの単一層 LUN で構成されています。
2. +DATA\_OLTP は、パフォーマンス SSD、パフォーマンス HDD、および容量 HDD から成る、自動階層化 LUN で構成されています。容量 SSD が構成に含まれる場合もありますが、この例では含まれていません。

データベース・オブジェクトでは、所有しているヒートを示すために同じ色分けが使用されていません。OrdersとOrder\_itemsは、パーティション化された表で、(最近アクセスされた)ホット・パーティションは、ORDER\_ENTRY表領域内に存在し (+DATA\_OLTPにマッピング)、一方コールド・パーティションは、(データ自動最適化ポリシーにより) LOW\_COST表領域 (+LOW\_COSTにマッピング) に移行されています。その他の表 (たとえば、Card\_details) 内では、Oracle FS1-2自動階層化により) +DATA\_OLTP内の別のストレージ層に移行された (ホット、ウォーム、およびコールド・データの領域がOracle FS1-2ヒート・マップにより示されています。これらの層は、ホストまたはData Baseには表示されません。Data Baseは、LUNを参照するだけであり、DBまたはアプリケーション・レベルで変更する必要はありません。

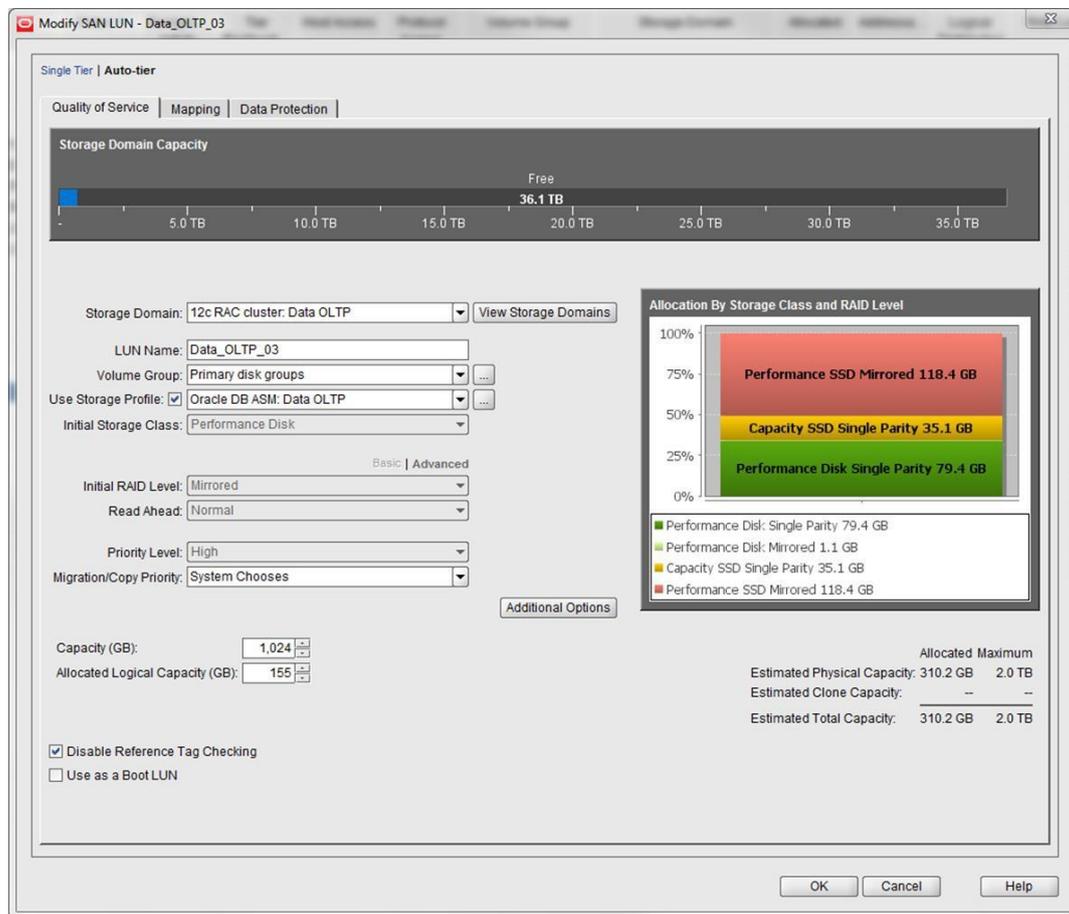


図14：自動階層化の実際の結果を示すOracle FS1-2のスクリーン・ショット

Oracle FS1-2 自動階層化機能は、コールド・データを“低”価格のストレージに移行するのに使用できますが、実際にはもっと多くの処理をします。つまり、収集したアクセス・データを使用して、LUN を構成する 640KB サイズのチャンクすべてを最適なストレージ層に一致させるような、高度な最適化アルゴリズムを実装しています。たとえば、ランダムな書き込み動作を行うチャンク（OLTP データベース）をパフォーマンス SSD（I/O 特性にもっとも適した価格/パフォーマンスを備えたストレージ）と一致させます。また、適切な RAID レベル（ミラー化、シングル・パリティ、またはダブル・パリティ）も動的に選択します。

この処理は、DBA またはストレージ管理者が明示的にデータ移行のリクエストを実行しなくても、すべて実行されます。システムは、自動的かつ透過的に（データベース、ストレージ、またはアプリケーションのパフォーマンスに影響することなく）、ストレージ層間でチャンクを移行します。管理者は装置を明示的にオン/オフするか、または、さまざまな層内のストレージの容量とタイプをチューニングすることにより制御できます。

図 14 は、Oracle Database 12c で実行されている OLTP ワークロードにストレージを割り当てるようにプロビジョニングされた、Oracle FS1-2 による自動階層化の実際の結果です。ここで使用されている自動階層 LUN は、パフォーマンス SSD、容量 SSD、およびパフォーマンス HDD から構成されています。この図は、ワークロードが長時間実行された後にデータが階層間にどのように割り当てられたかを示しています。すべての結果において、パフォーマンスを犠牲にせずに全体のコストが低く抑えられています。

## まとめと推奨事項

データベース内のストレージ層間でデータを移行する手段として、長い間手動による方法が使用されてきました。しかし、最近開発された 2 つの新技术—ILM ヒート・マップ/データ自動最適化ポリシーおよび Oracle FS1-2 ヒート・マップと自動階層化—により、管理者ツールのポートフォリオに大幅な柔軟性と機能が追加されました。現実のデータベースでは、これら 3 つの手法はすべて有効に使用されており、いずれかを選択する必要はありません。むしろ、それぞれの技術が適切に使用される状況を理解しておくことを推奨します。

以下に、おもな推奨事項を示します。

- 》 指定したスケジュールに従いコールド状態に移行する表およびパーティションの場合、DBA は手動でデータを移行できます（または、スケジュール済みジョブを作成できます）。
- 》 パーティション内のすべてのデータが同時にコールド状態に移行するが、そのタイミングを正確に認識できていない場合は、ILM ヒート・マップにより、データのアクセス・パターンを測定し、カスタムのデータ自動最適化ポリシーを作成し、データの移行を開始することを推奨します。
- 》 ホット・データ/コールド・データの粒度が、パーティション全体より小さいという一般的なケースを処理するには、データのアクセス・パターンを測定する Oracle FS1-2 ヒート・マップや、ストレージ層間でデータを移行する Oracle FS1-2 自動階層化を使用します。これは、データ自動最適化ポリシーとは独立して、または連携して実行できます。



CONNECT WITH US

 [blogs.oracle.com/oracle](https://blogs.oracle.com/oracle)  
 [facebook.com/oracle](https://facebook.com/oracle)  
 [twitter.com/oracle](https://twitter.com/oracle)  
 [oracle.com](https://oracle.com)

お問い合わせ窓口

Oracle Direct

 0120-155-096

 [oracle.com/jp/direct](https://oracle.com/jp/direct)

#### Hardware and Software, Engineered to Work Together

Copyright © 2015, Oracle and/or its affiliates. All rights reserved. 本文書は情報提供のみを目的として提供されており、記載内容は予告なく変更されることがあります。本文書は、その内容に誤りがないことを保証するものではなく、また、口頭による明示または法律による黙示を問わず、特定の目的に対する商品性もしくは適合性についての黙示的な保証を含め、いかなる他の保証や条件も提供するものではありません。オラクル社は本文書に関するいかなる法的責任も明確に否認し、本文書によって直接的または間接的に確立される契約義務はないものとします。本文書はオラクル社の書面による許可を前もって得ることなく、いかなる目的のためにも、電子または印刷を含むいかなる形式や手段によっても再作成または送信することはできません。

Oracle および Java は Oracle およびその子会社、関連会社の登録商標です。その他の名称はそれぞれの会社の商標です。

Intel および Intel Xeon は Intel Corporation の商標または登録商標です。すべての SPARC 商標はライセンスに基づいて使用される SPARC International, Inc. の商標または登録商標です。AMD、Opteron、AMD ロゴおよび AMD Opteron ロゴは、Advanced Micro Devices の商標または登録商標です。UNIX は、The Open Group の登録商標です。0115

Oracle Database 12c と Oracle FS1-2 フラッシュ・ストレージ・システムを使用したストレージ階層化テクノロジー 2015 年 4 月



Oracle is committed to developing practices and products that help protect the environment