

プライベートクラウド構築に向けた IT基盤のあり方と今後

～なぜ、Oracle Database 12cを導入したのか、
導入事例から学ぶその効果とは～

日本オラクル株式会社
クラウド・テクノロジー事業統括

2015年8月19日

ORACLE®

Copyright © 2015 Oracle and/or its affiliates. All rights reserved. |

SAFE HARBOR STATEMENT

以下の事項は、弊社の一般的な製品の方向性に関する概要を説明するものです。また、情報提供を唯一の目的とするものであり、いかなる契約にも組み込むことはできません。以下の事項は、マテリアルやコード、機能を提供することをコミットメント（確約）するものではないため、購買決定を行う際の判断材料になさらないで下さい。オラクル製品に関して記載されている機能の開発、リリースおよび時期については、弊社の裁量により決定されます。

Oracleは、米国オラクル・コーポレーション及びその子会社、関連会社の米国及びその他の国における登録商標または商標です。他社名又は製品名は、それぞれ各社の商標である場合があります。

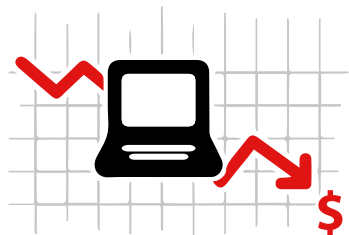
IT部門の代表的な課題

これらの課題を解決するために、システム統合を進めてきたが…

IT コスト削減

ビジネスへの貢献

事業継続リスクの最小化



- ✓ システム統合
(ITリソース利用率最大化)
- ✓ 保守・運用作業の軽減

- ✓ Time-to-Marketの迅速化
- ✓ 業務部門における
情報活用の促進
(感覚ベース->事実ベース)

- ✓ 可用性の向上
- ✓ セキュリティレベルの向上

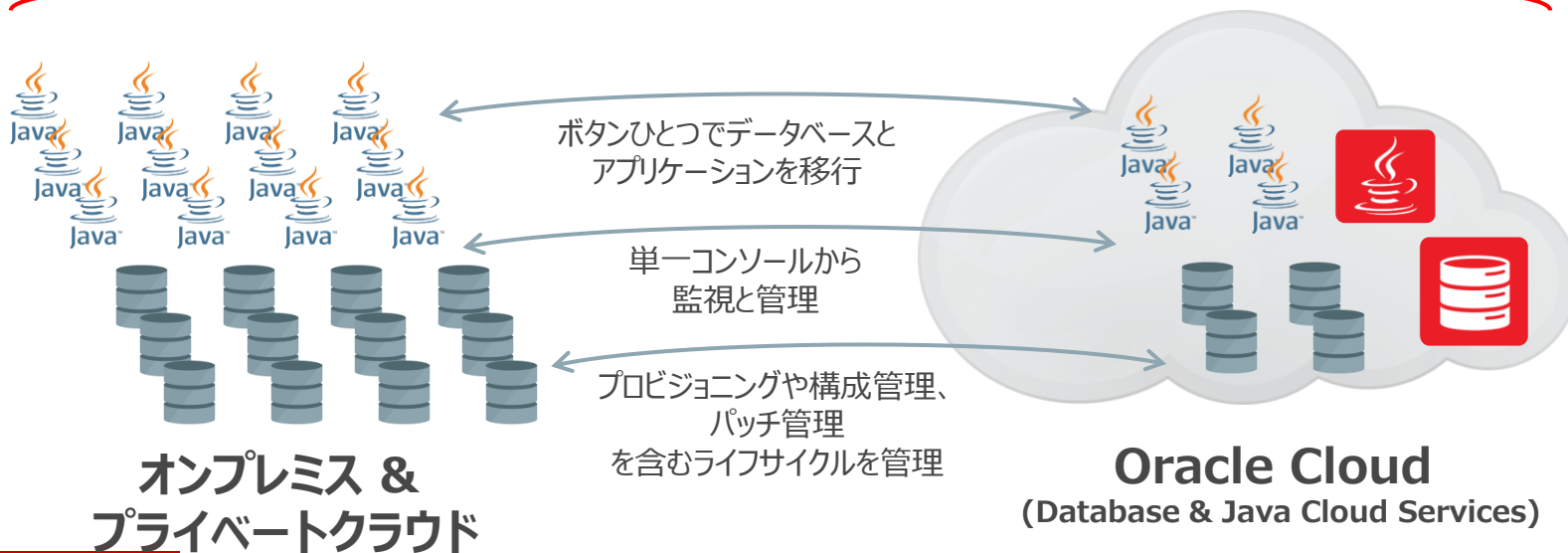
オラクルのクラウド戦略：オンプレとクラウド間の優れた移行性

ハイブリッドクラウドの管理：単一コンソールからアップグレード、移行、管理を実現

ORACLE
ENTERPRISE MANAGER

同じアーキテクチャ、同じ標準技術、同じスキルセット

より速く、より簡単に、より少ないコストとリスクで



データベース・クラウド実現の歩み

オラクルによるイノベーションの継続 (i → g → c)

ORACLE
DATABASE 12^c

クラウド対応

ORACLE
EXADATA

Extreme Performance

ORACLE
DATABASE 11^g

リソース仮想化

ORACLE
DATABASE 10^g

エンタープライズ・グリッド

ORACLE
9ⁱ
INTERNET

リニア・スケーラビリティ

ORACLE
8ⁱ
INTERNET

インターネット対応

Oracle
8

大規模 Database

Oracle 7

エンタープライズ・システム

Oracle 6 行レベルロック

Oracle 5 平行・サーバ

ORACLE
DATABASE 12^c

In-Memory
Data Guard Far Sync
Data Redaction
Multitenant Architecture

ORACLE
EXADATA

Exadata
Grid Infrastructure
Advanced Compression
Active Data Guard

ORACLE
DATABASE 11^g

自動データベース診断モニター(ADDM)
Automatic Storage Management(ASM)
Grid Control

ORACLE
DATABASE 10^g

ORACLE
9ⁱ
INTERNET

Data Guard
Real Application Clusters(RAC)

ORACLE
8ⁱ
INTERNET

Java/XML対応
マテリアライズド・ビュー
ハッシュ/コンポジット・パーティション

Oracle
8

パラレルDML
パーティション表

Oracle 7 平行クエリ
スタンバイ・データベース

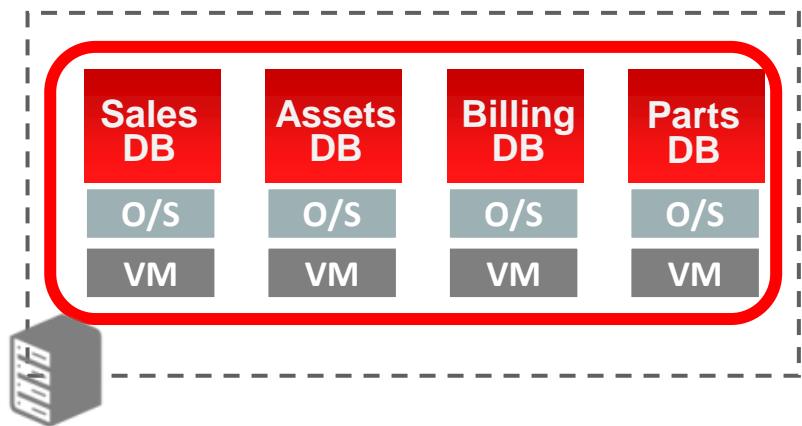
次世代のプライベートクラウド基盤 構築のポイントとは

システム構成と運用は、より**シンプル**に

サーバー仮想化の適用範囲を見極める

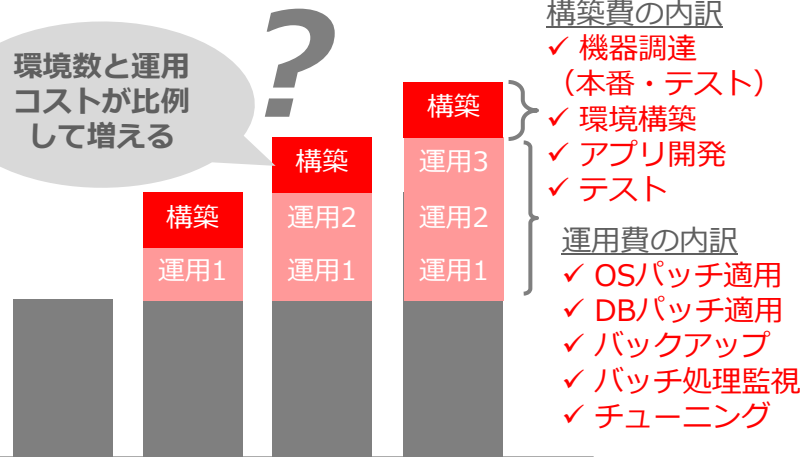
IaaS（サーバ仮想化）による効果と解決しなければならないことは

IaaS: VMによるサーバー仮想化



- OSレベルのITインフラ(仮想化)
- 柔軟に技術が適用できる
- H/Wの集約が可能&システムの延命

環境数と運用コストが比例して増える

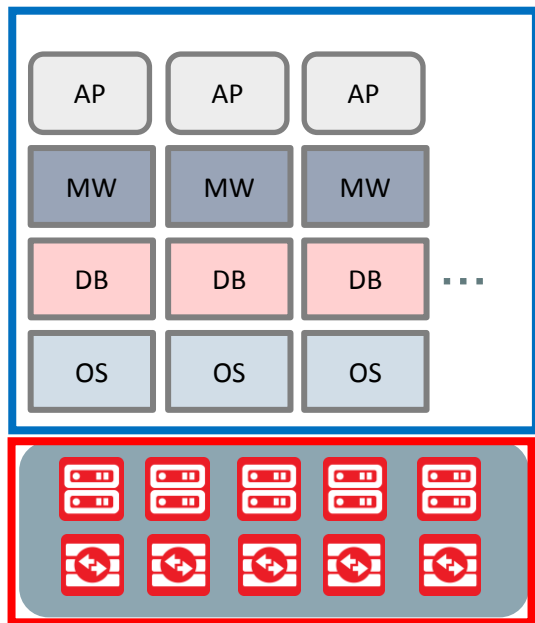


- システムがより複雑に
- 運用管理コスト増につながる
- システムのSLA (Security, 業務継続性) に課題

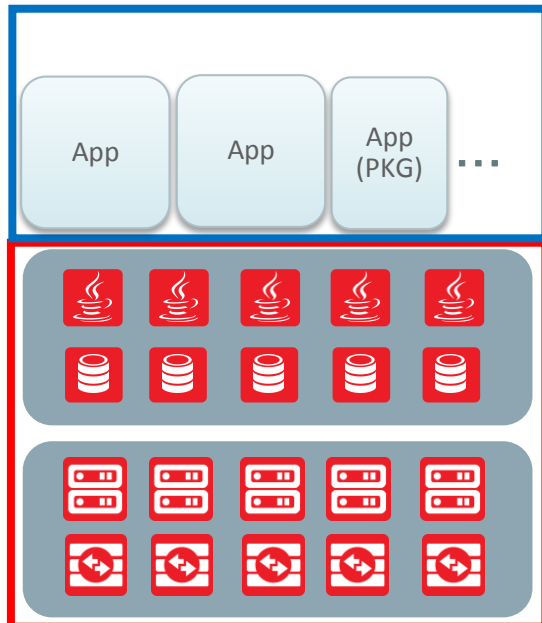
IT基盤の標準化のレベルを引き上げる

PaaS基盤(プライベートクラウド)によるサービスレベルの向上

IaaS



PaaS



アプリ開発側の管理範囲と効果

- アプリ開発に注力できる
- DB・AP環境を瞬時入手、拡張性
- 標準技術(SQL、Java)の採用
- パブリッククラウド(PaaS)との連携

IT基盤提供側の管理範囲と効果

- 社内ガバナンスの強化
- IT基盤の標準化レベルの促進
- リソースの有効活用
- システム俊敏性

- 業務継続性(高可用性)の強化
- セキュリティ対策の強化
- 運用の標準化(自動化)
- バックアップ・パッチ適用の標準化

低

標準化と共有化の度合い

高

プライベートクラウド基盤の構築のポイント（勘所）

データベースとアプリは分離して

（アーキテクチャと費用の観点から考える）

データベースをSLA（サービスレベル）と依存関係を整理

（移行が容易なシステムから順次統合）

インフラ視点 vs サービス視点のアプローチ

（サービス視点からアーキテクチャを考える）

標準化されたプラットフォームとサービスカタログの導入

（プラットフォームは2世代で）

バックアップやパッチ適用の運用はシンプルに

（構成パターンはシンプルに）

ストレージの初期・運用費用が懸案のとき

（高機能なストレージは本当に必要か？）

リソース（CPU、I/O）のコントロールとキャパシティプランニング

（ピークと縮退を想定しておくこと）

インフラ担当部門がPaaS領域もカバーするように

（DBAの専門チームを作る）

プライベートクラウド構築に向けたアプローチ

システムの現状を把握し、段階的な取り組み

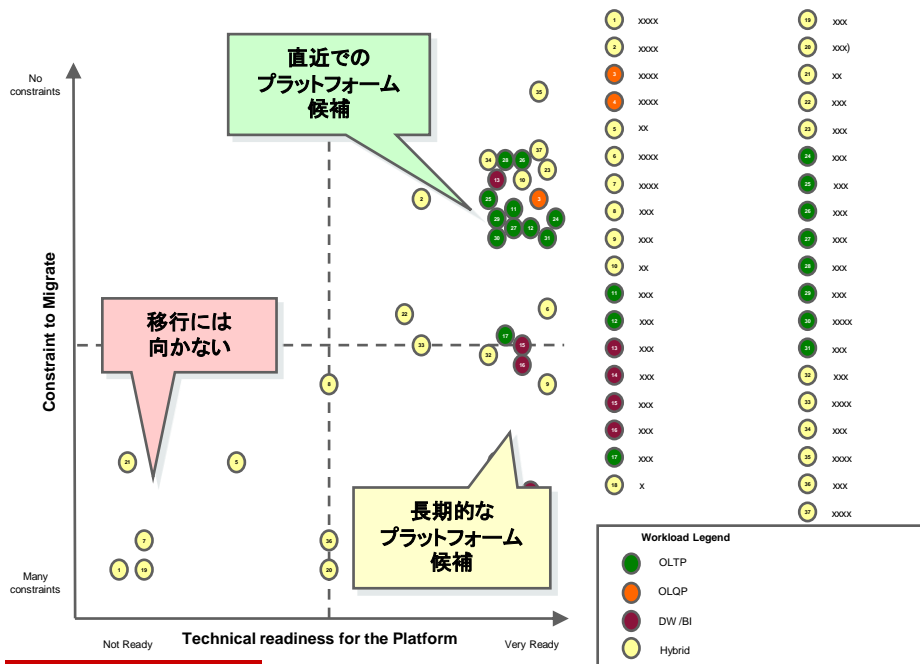
最も敷居が低い

新たに開発

本番

混合ワークロード

ミッションクリティカル



- システムの現状把握: 最も敷居が低いところはどこなのか、明確な基準を定義
- 早期に成功できるように計画を立てる
- 段階的な移行により、リスクを低減する
- 早期に実施した移行プロジェクトで学んだ点を、次期の移行に活かす
- サービスレベル、ワークロードの混合にも耐えうる、柔軟なアーキテクチャを

プライベートクラウド基盤のサービスレベル (例)

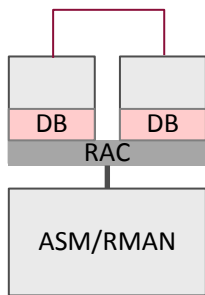
		サービスレベル指向		
		Silver	Gold	Platinum
システム の 特徴	①システム基盤の重要度	低	中	高
	②復旧時間 (RTO)	1日	2時間～できるだけ早く	2時間以内
	③リカバリ・ポイント (RPO)	数時間前	データ損失ゼロ	データ損失ゼロ
	④セキュリティレベル (個人情報、機密情報)	低	中	高
システム の 主要要件	①バックアップ方式、取得 間隔	日次	数回/日 (4～6時間毎)	リアルタイム
	②機器などの冗長化	あり	あり	あり
	③バックアップ (DRサイト)	なし	同一サイト内 (同期/非同期)	遠隔地 (同期/非同期)
	④パッチ適用	計画停止 (休日日中メンテナンス)	計画停止 (夜間メンテナンス)	計画停止 / 無停止
想定システム		Aシステム Bシステム	Cシステム、 Dシステム、Eシステム	Xシステム Yシステム

プライベートクラウド基盤のサービスレベル(例)

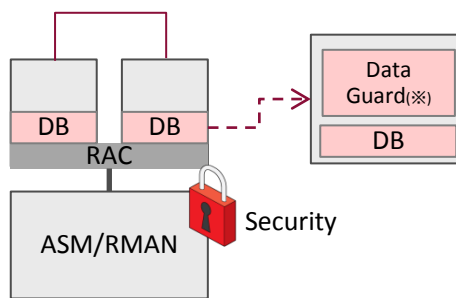
SLA毎(サービスモデル)に具体的なアーキテクチャへのマッピング

SLA毎の
アーキテクチャ

Silver

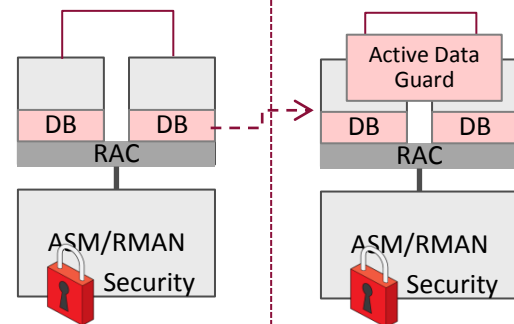


Gold



※同一サイト内でのバックアップ

Platinum



※遠隔地サイト内でのバックアップ

最終的なシステムとSLAのマッピング

Aシステム

Bシステム

Cシステム

Dシステム

Eシステム

Xシステム

Yシステム

次世代IT基盤(プライベートクラウド)

顧客事例:テレコム・イタリア(欧州/通信事業)

イタリア国内の複数拠点にデータセンターやサービスセンターを展開

レガシーシステム (AS - IS)










- 7 データセンター (5,000 データベース)
- 年間約36,000件のインフラの変更依頼に対応
- 1 アプリケーション毎にDBを設計する場合 (これまでの開発アプローチ)
- 1DBのセットアップ時間 : **週単位**
- 異なる製品・技術の組み合わせ (約700パターン) (DBバージョン/ OSバージョン / サーバ・ストレージ等)



- アプリケーション毎に割当てたHWリソースは低い利用率

データベースクラウド (DBaaS)

- 標準化された DB サービス 提供する場合 (DB のサービスカタログ)
- 1DBのセットアップ時間 : **時間単位**
- 5パターンの標準構成に3つのサービスレベル

	方式A	方式B	方式C
Gold			1 
Silver		2 	3 
Bronze	4 	5 	

- HWリソースを共有することでリソースの有効活用が可能

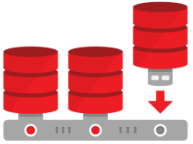
DBaaS導入による効果

効果指標

導入効果

EM12cの役割

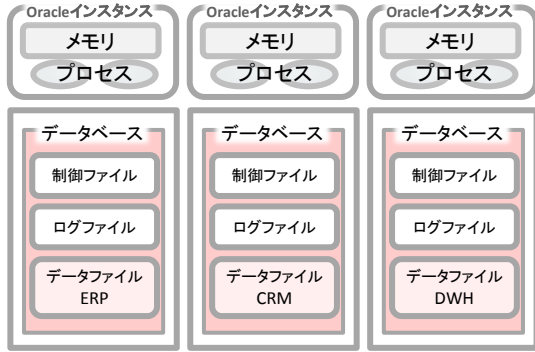
	効果指標	導入効果	EM12cの役割
作業効率	HWリソースとファシリティの省力化 (サーバ, ストレージ, 設置場所, 消費電力)	保守コストを 80%削減	<ul style="list-style-type: none">DBaaS環境に割り当てたリソース (DB, OS, HW) 利用状況のモニタリング
	DBセットアップ作業の省力化	DBセットアップ毎の工数を 60%削減 操作工数を 20%省力化	<ul style="list-style-type: none">GUI操作によるDBセットアップ (これまでのコマンドラインによる複雑な操作より手順が大幅にシンプルに)
アジリティ	利用者へのDBの納期 (時間)	納期を 80%短縮	<ul style="list-style-type: none">DBプロビジョニングの自動化 (テレコム・イタリアで導入しているプロビジョニングツールとも連携しセットアップ実行)
	サービスレベルの改善 (DBの信頼性と性能/システムの操作性)	サービスレベルの 事前定義化 (システム毎に定義する 必要がなくなる)	<ul style="list-style-type: none">テレコム・イタリアで導入している監視ツールと連携し、DBの性能、キャパシティ、インシデントを集中管理



マルチテナント・アーキテクチャ Oracle Multitenant

アプリケーションは仮想的なDB単位で独立、メモリとプロセスはDBレベルで共有

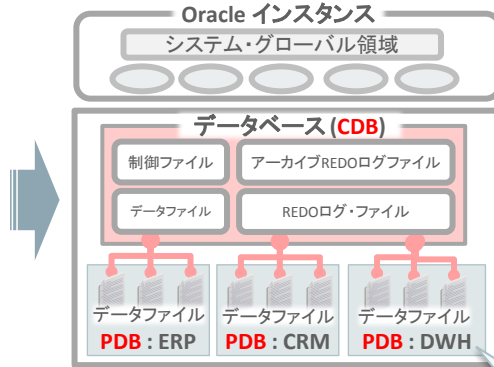
通常のデータベース



従来のデータベース構成
データベースごとに独立したインスタンスを立ち上げメモリ領域を確保

マルチテナントのデータベース

ORACLE DATABASE 12c



マルチテナント・アーキテクチャ
複数のPDB間でデータベース・インスタンスとメモリ領域、プロセスなどを共有

システムはPluggable Database (PDB) ごとに稼働

- 1つのデータベースに複数作ることができる
- アプリケーション&DBに変更は不要
- 移動・コピー・環境作成等は簡単なコマンドで実施可

メモリとバックグラウンドプロセスは共有

- 1つのDB(CDB)に、複数の仮想DB(PDB)を構築
- 制御ファイル、REDOなどの管理要素も共有
- リソースを効率的に共有

運用管理はデータベース(CDB)単位で実施可能

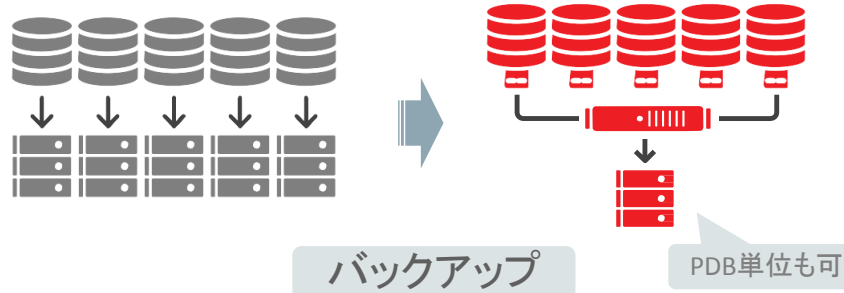
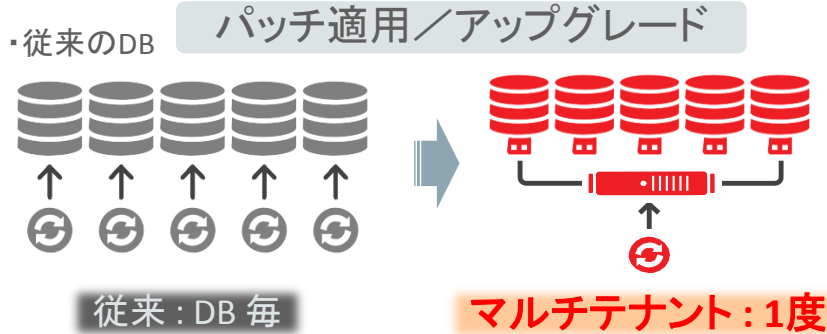
- アップグレード、バックアップなどをCDB単位で実行
- 必要に応じてPDB毎に管理することも可能

プラグブル・データベース(PDB)

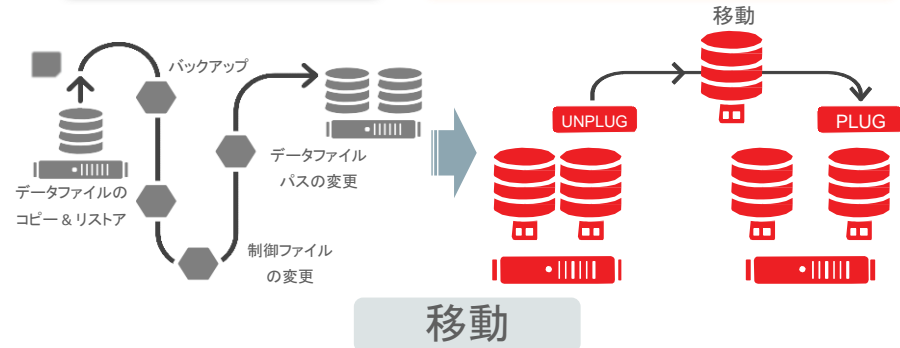
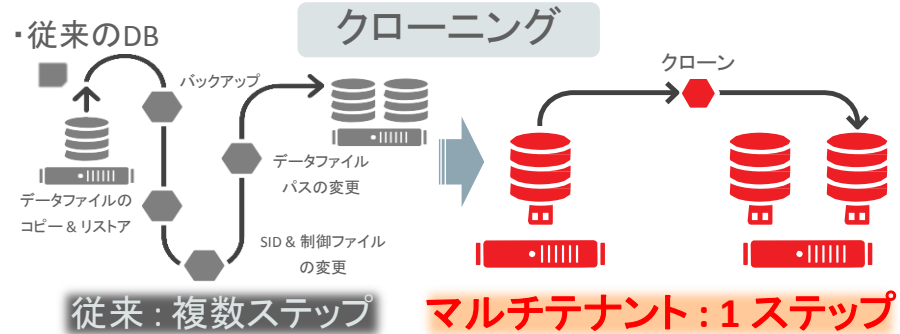
- スキーマや表領域を含む論理的なセット
- PDB間は排他的な関係。データは論理的に分離される
- 同一CDB内に複数作成可能(シードを除き最大252まで)
- CDBから取り外し・取り付け、異なるCDBへ取り付け可能

Oracle Multitenant Database Cloudを実現するシンプルな管理

多数のDBを1つのDBとして管理



迅速なDBの展開

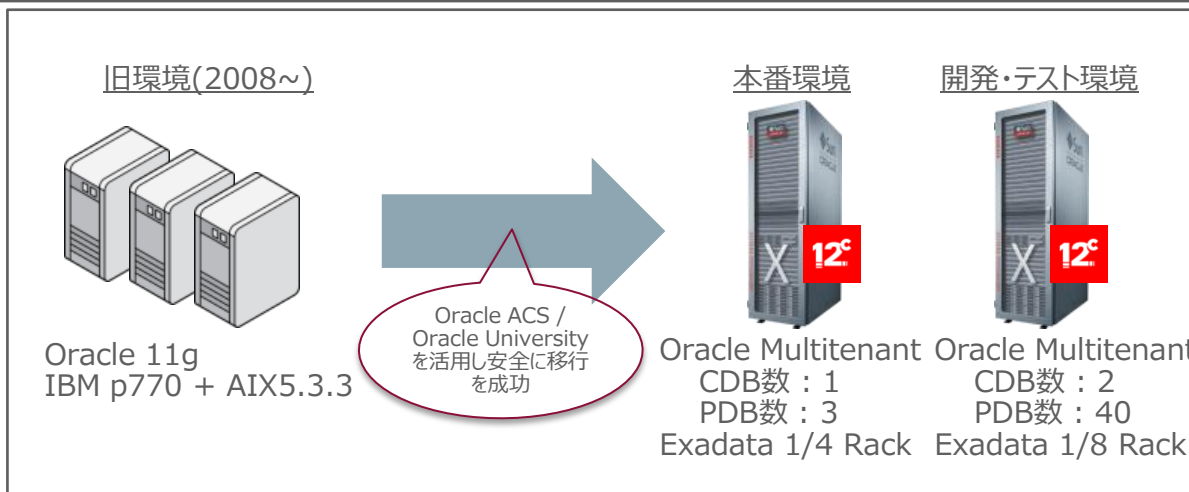


Oracle Multitenant 事例 Pulte Homes - 米国大手住宅建設会社



導入効果	導入コスト削減	運用コスト削減	性能向上
<p>“This was an exceptionally smooth migration for us from Oracle Database 11g to Oracle Database 12c and Oracle Multitenant on Oracle Exadata.” -Brian Pawlik, Pulte Group Inc.</p> <p>Play the Podcast! -> </p>	<p>170万USDの 導入費用減</p>	<p>年間3~4人月の運用工数減 DB複製：16時間→1.5時間 パッチ適用：43回→3回</p>	<p>ERPの処理性能が 45~65%向上</p>

- ### ビジネス課題
- ERPアップグレードに伴うインフラ更新を安価に実現
 - 運用工数の削減
 - 安全なインフラ移行の実現
- ### 採用製品とサービス
- IBM p770サーバーをOracle Exadata (Quarter Rack+Eighth Rack)に更新
 - Oracle Multitenant の採用
 - Oracle ACS と Oracle University サービスの活用



Oracle Engineered Systemsがもたらすお客様価値 「ガラスボックス」化が、行政システムのコスト構造を変革する

札幌市 様

□ 課題

- **大型汎用機システム**の複雑化が招いた、**IT環境の高コスト体質化**
 - 制度改正の度に発生する追加開発、修正
 - **随意契約の長期化** -> **ブラックボックス化**
 - 汎用機稼働年数:**25年以上**
 - **公正なコスト比較&システム変更の困難化**
 - **発注者主体、マルチベンダ分割発注へ**

□ 課題解決に向けて

- **ガラスボックス化**
 - **発注者主体、マルチベンダ分割発注**
 - **発注の透明性:地場産業活性化(0→16社)**
 - **費用削減効果:年間費用16億から25%削減**



Oracle Engineered Systemsがもたらすお客様価値 「ガラスボックス」化が、行政システムのコスト構造を変革する

札幌市 様

□ 解決策

- 新基幹系情報システムを構築
 - **オープンな技術の採用**: SQL、Java
 - 競争入札の結果、Engineered Systemを導入
ExadataとExalogicによる最適な組み合わせ
- テスト・開発環境の充実
 - **独立性のあるDB・APのテスト環境**を用意
 - デプロイメントツールによるスムーズな環境構築
 - **マルチテナント技術による複数DB環境構築**
- 大切な個人情報等の重要なデータを守る
 - **本番データを伏字化(マスキング)**
- 柔軟なデータベース間のデータ連携
 - **本番システムのデータ保全、開発へのデータ活用**



お客様事例：札幌市

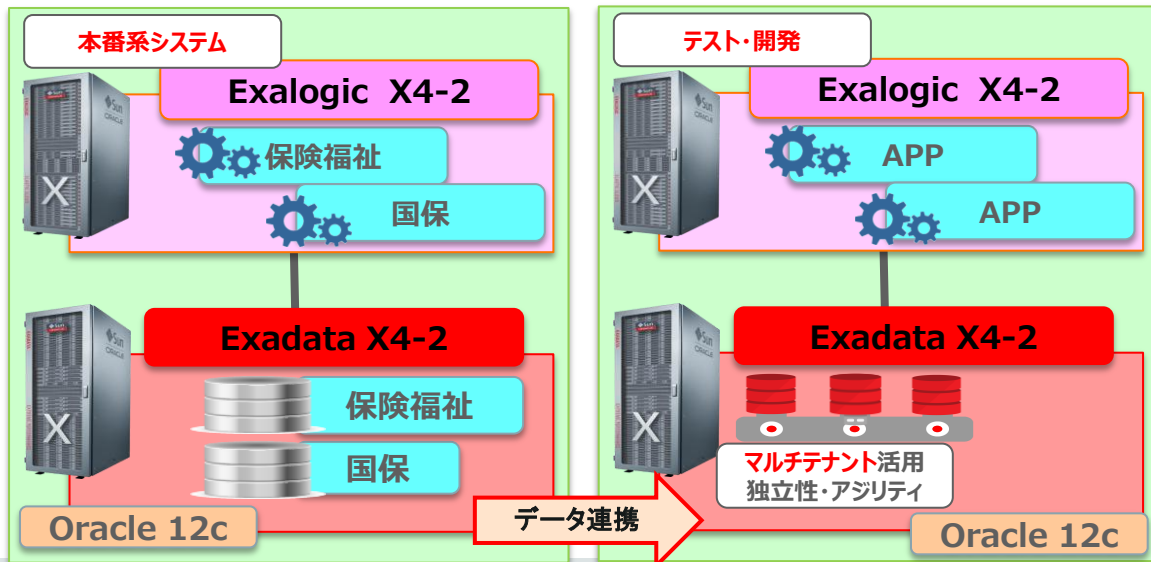
Oracle Multitenantを活用し、ベンダー毎、システム開発毎に潤沢なテスト環境を用意

● 100以上のドメイン管理必要

- 独立性のあるDB、APのテスト環境を用意（※1, ※2）
- 環境デプロイメント作業をツール化して作業軽減（※1）

● テストデータ（マスキング済み）用意

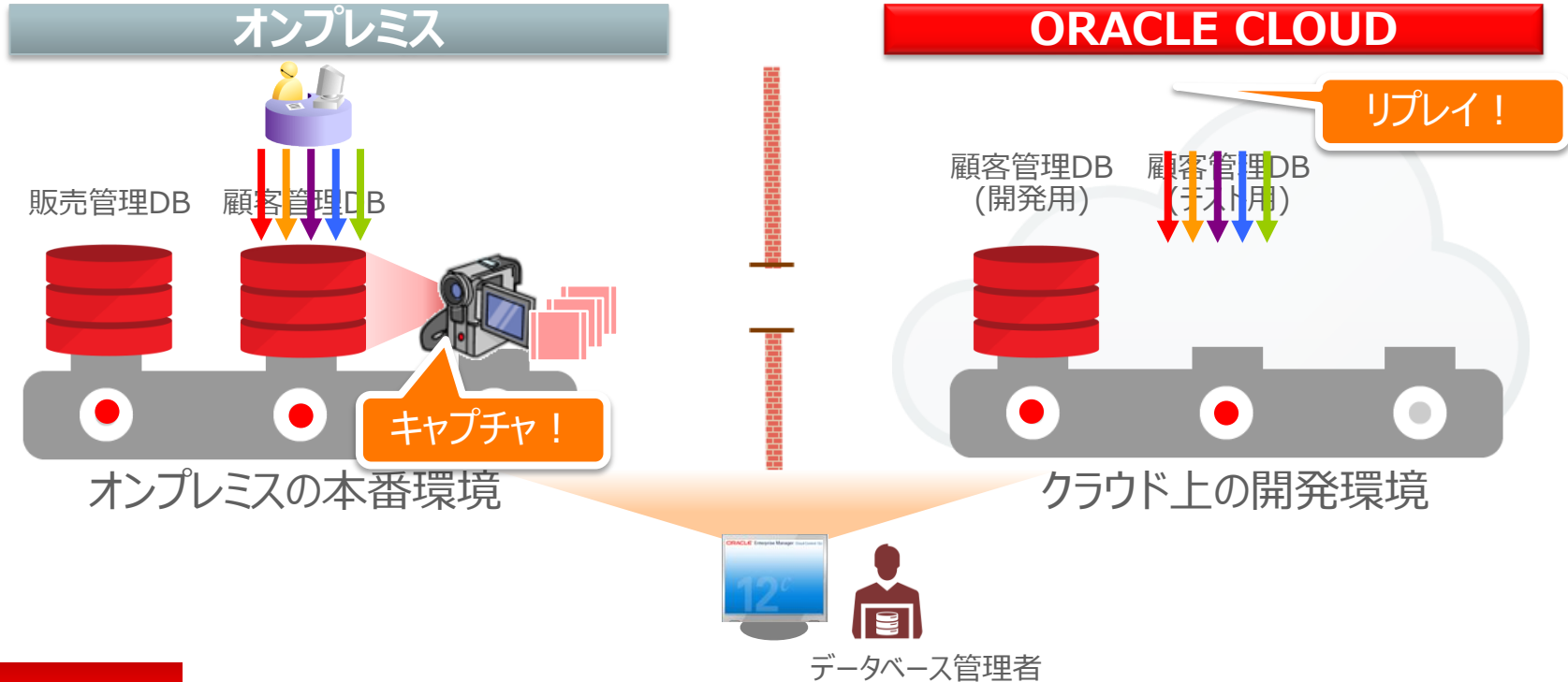
- 本番データを伏字化して、開発・テスト向けに提供（※3）
- 本番システムのデータ保全とテスト・開発環境との柔軟なデータ連携（※4）



- (※1) **オラクルコンサルティングサービス**による支援
- (※2) **Oracle Multitenant**の機能利用、H27実装
- (※3) **Data Masking Pack**の機能利用、H27実装
- (※4) **Oracle GoldenGate**の機能利用、H27実装

PaaSを有効活用してもらうためのシナリオ (開発&テスト)

オンプレミス: **Diag&Tuning, RAT, DB LCM** + Oracle Cloud: "EE High Performance "



高度分析をリアルタイム化する 新たなテクノロジー

Database In-Memory技術が与えるインパクト

高度分析をリアルタイム化する新たな技術

リアルタイム
アナリティクス



100x

OLTPの
高速化

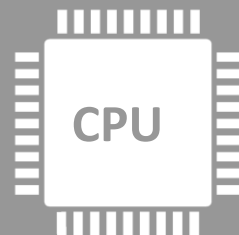


2x

アプリケーション
の変更なし



最新世代の
H/Wを有効活用



■ Oracle Database In-Memory Option

- お客様の**既存資産**（データとアプリケーション）の保護と継承
- 実績のあるOracle Databaseの**高可用性とセキュリティ機能**を利用可能
- 業界初のデュアルフォーマットの提供により、**OLTPとDWHを統合可能**
- DWHからインデックスを排除し、**アプリ開発・DB運用コストを削減**

OLTP処理と分析処理の両立

ロー vs. カラム - いままでのデータベースは、いずれかの特性“のみ”を実装

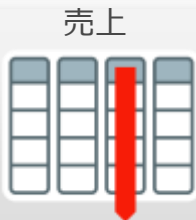
ロー(行)
型



■ OLTP処理を得意とするロー型

- 例：注文データの挿入と検索
- 少数の行(ロー)と多数の列(カラム)を高速処理

カラム(列)
型



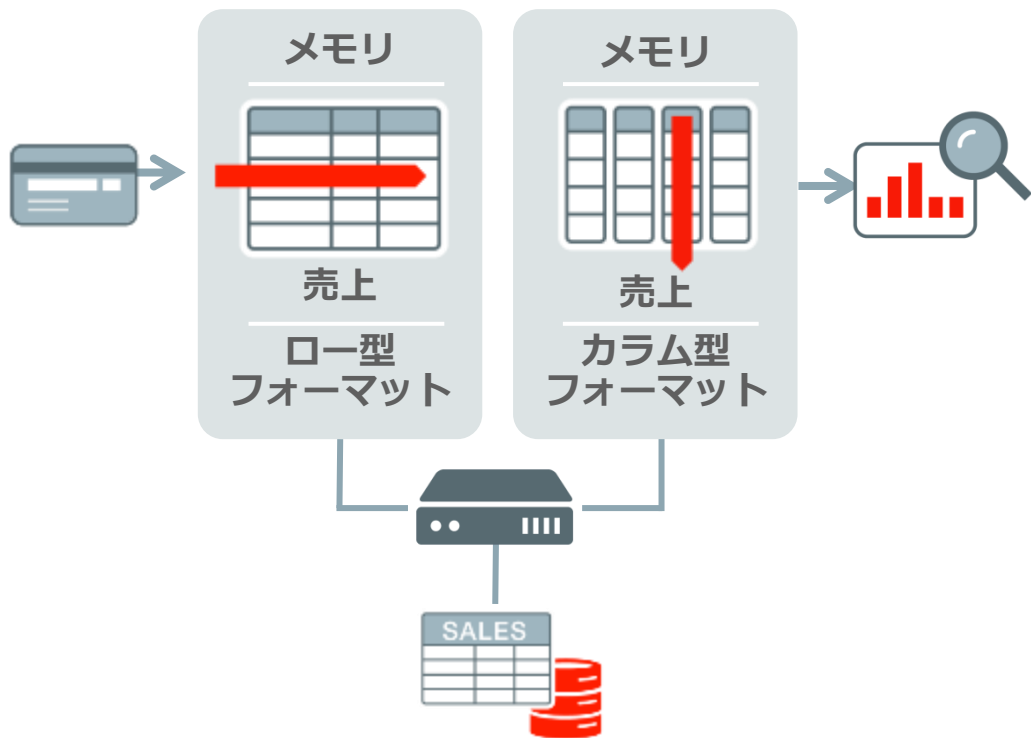
■ 集計、分析処理を高速化するカラム型

- 例：都道府県毎の売上合計のレポート
- 少数の列(カラム)と多数の行(ロー)を高速処理

Oracle Database In-Memory テクノロジーは各特性を持つ、
2つのフォーマットを“**両方同時に**”メモリ上にロードし利用可能

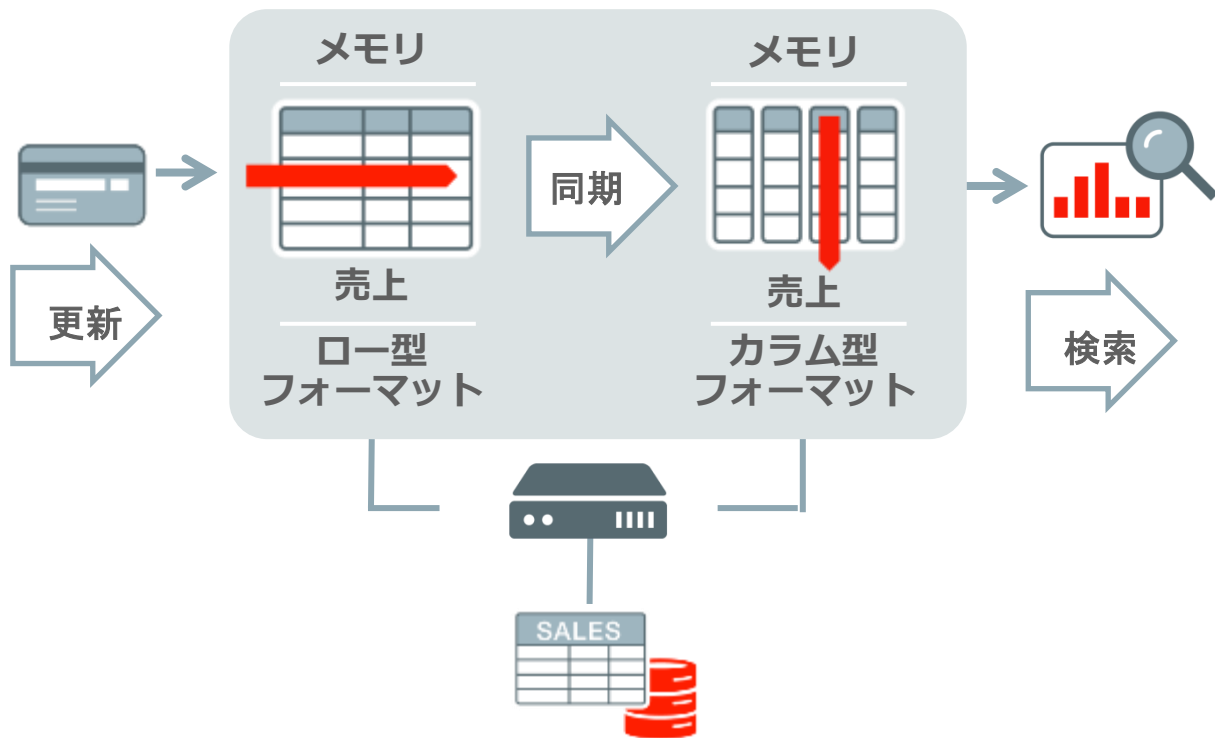
高速な分析をリアルタイム化する新たな技術革新

インメモリ・デュアル・フォーマット



- 同一のデータをロー型、コラム型両方のフォーマットでメモリ上に保持
- 両方のフォーマットを同時に利用可能、トランザクションの一貫性も担保
- 2種類のフォーマットをオプティマイザが自動判別
 - 集計、レポート処理はコラム型フォーマットに対して実行
 - OLTP処理はロー型フォーマットに対して実行

OLTPと分析処理のトランザクション一貫性を担保 インメモリ・デュアル・フォーマット



- ロー型フォーマットで更新した情報をコラム型フォーマットへ同期
- コラム型フォーマットは変更データのみを定期的にリポピュレート
- ユーザは最新の更新データを元に検索が可能
ユーザからは透過的

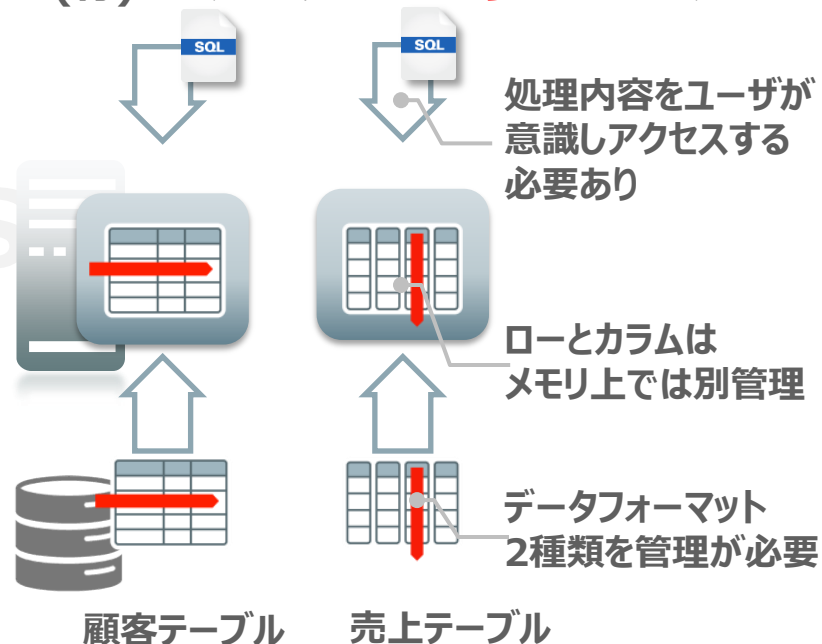
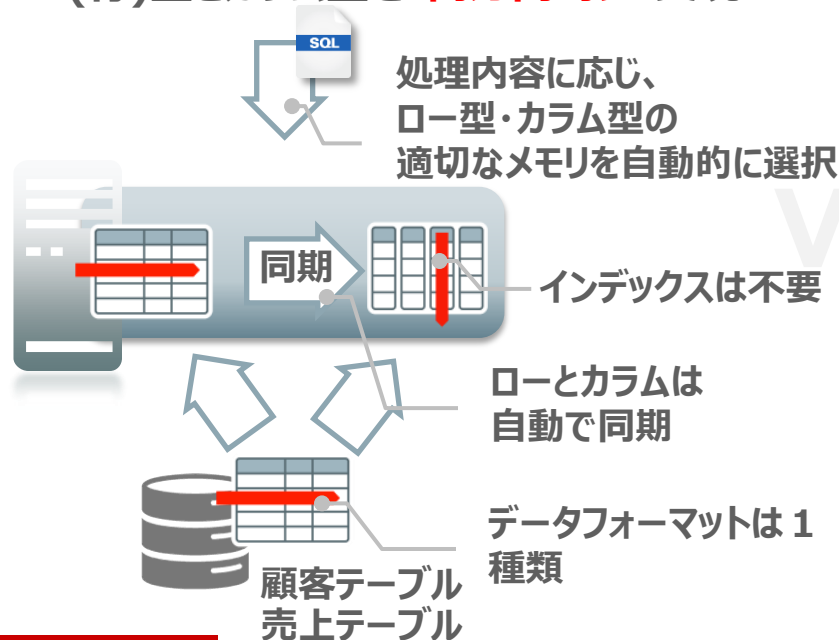
OLTPと分析処理をリアルタイムに融合可能な“唯一の”仕組み インメモリ・デュアル・フォーマット

Oracle Database In-Memory

一般的なインメモリの仕組み

■ロー(行)型とカラム型を“両方同時に”実現

■ロー(行)型とカラム型の“どちらか一方を”選択



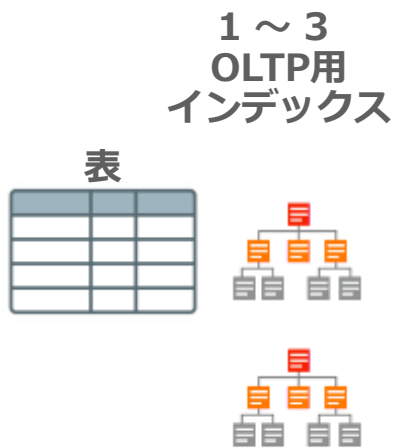
分析基盤におけるインデックスが不要

■ 既存Oracle Databaseによる分析基盤

- インデックスは事前予測可能なパターンのみ高速化
- 更新処理は10~20個のインデックスの更新が必要
 - パフォーマンス低下の原因

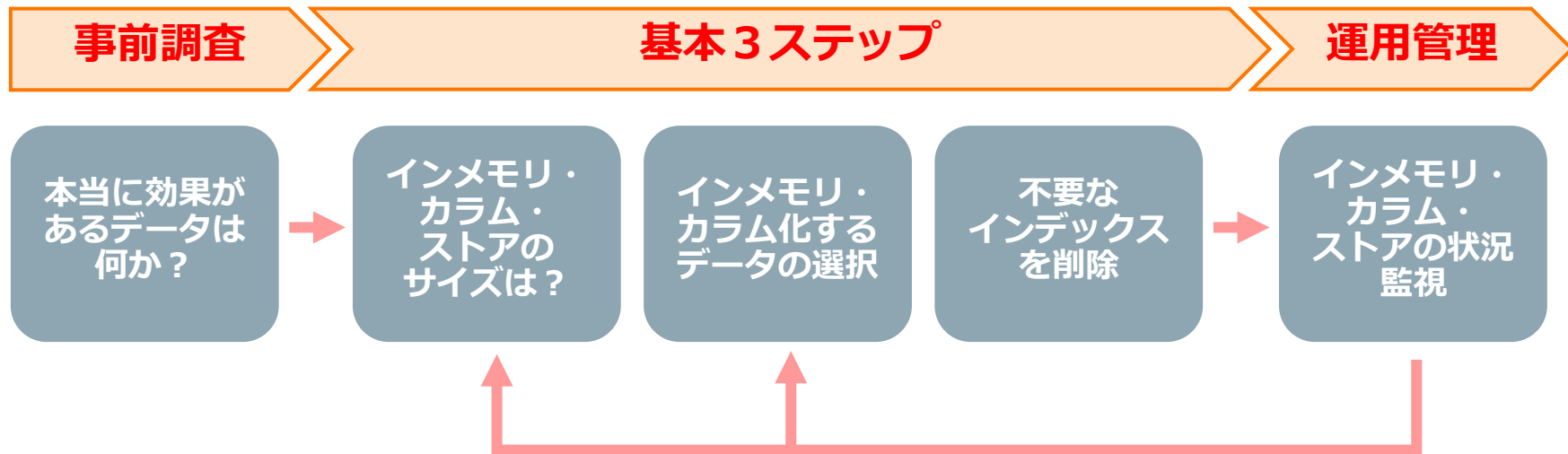
■ Oracle Database In-Memoryの適用

- インデックスなしですべてのカラム(列)の処理を高速化
- カラム型ストアはディスクI/Oなし



Oracle Database In-Memoryの基本導入手順

設定・運用管理



Enterprise Managerによるインメモリ管理 インメモリ・セントラル

メニュー・パス
データベース・ホーム → 管理 → インメモリ・セントラル

Configuration

Configuration

- Total SGA (GB) 3.00
- In Memory Area (GB) 2.00
- In Memory Query Enabled
- In Memory Force Default
- Default In Memory Clause Not Specified

Performance

- Active Sessions (CPU) 0.02

Objects Summary

In Memory Loaded Object Statistics

- Size (GB) 0.05
- Compression Factor 9.82
- Loading (%) 91.64

In Memory Enabled Object Statistics

- Total Size of all Objects (GB) 2.38
- In Memory Enabled Object Size (GB) 0.57
- In Memory Enabled (%) 24.11

In Memory Objects Distribution

- Non-partitioned Tables (GB) 0.03
- Sub-partitions (GB) 0.01
- Partitions (GB) 0.01
- Non-partitioned Materialized Views (GB) 0

Heat Map

Objects without access data

Use the date selector to pick the date range for objects displayed in the heat map. Use the slider to control the granularity of the color.

In Memory Objects Search

Search

Name Segment Type

Schema Size (GB) >

Tablespace In-Memory Size (GB) >

Compression Priority Distribution

Name	Segment Type	Size (GB)	In Memory Size (GB)	Compression Factor	Loading (%)	Population Status	Cor
SALES	Non-partitioned T...	0.1797	0.0178	10.08	100	Completed	FOR
LOCATIONS	Non-partitioned T...	0.1797	0.0081	22.14	100	Completed	FOR
INVENTORY.DEPT_100	Partitions	0.0332	0.0052	6.4	100	Completed	FOR
REGIONS	Non-partitioned T...	0.0107	0.0032	3.38	100	Completed	FOR
NON_MARKETING_DEPARTMENT	Non-partitioned M...	0.0098	0.0032	3.08	100	Completed	FOR
EMPLOYEES.EMPLOYEE_2006.SYS_SUBP222	Sub-partitions	0.0332	0.0021	15.54	100	Completed	FOR
EMPLOYEES.EMPLOYEE_CURRENT.EMPLOYEE...	Sub-partitions	0.0225	0.0021	10.51	100	Completed	FOR
LINE_ORDER	Non-partitioned T...	0.0001	0.0021	0.06	100	Completed	FOR
DEPARTMENTS.DEPT_OTHERS	Partitions	0.0039	0.0021	1.83	100	Completed	FOR
EMPLOYEES.EMPLOYEE_2003.SYS_SUBP221	Sub-partitions	0.0068	0.0021	3.2	100	Completed	FOR

Enterprise Managerによるインメモリ管理

インメモリ・セントラル

The screenshot displays the Oracle Enterprise Manager In-Memory Central interface. It features several panels for monitoring and configuring memory usage. A red box highlights the configuration section, which includes a bar chart showing total SGA (2.53 GB) and memory-allocated objects (1.00 GB). Another red box points to a callout text explaining that the total SGA area is occupied by memory-allocated objects (1.00 GB in the example). A third red box highlights the object statistics table, with a callout explaining that 0.58 GB of the 1 GB memory-allocated area is occupied by objects.

全体のSGA領域に占めるインメモリ領域(GB) (画面の例では1.00 GB)

インメモリ領域1GB中オブジェクトへ割当て済のインメモリ領域(GB) (画面の例では0.58GB)

名前	セグメント・タイプ	メモリ内圧縮率	メモリ内ロード(%)	メモリ内サイズ(GB)	Population	FOR QUERY LOW	CRITICAL	AUTO	0.0206	なし
SSB.LINEORDER	パーティション化されていない表	1.2315x	100	0.5202	Completed					
SSB.CUSTOMER3	パーティション化されていない表	1.5504x	100	0.0157	Completed					
SSB.CUSTOMER	パーティション化されていない表	1.8876x	100	0.0109	Completed					
SH.SALES2.SALES_2000	パーティション	1.9248x	100	0.0081	Completed					0.0156
SH.SALES2.SALES_2001	パーティション	1.9248x	100	0.0081	Completed					0.0156

Oracle Database In-Memoryがもたらすお客様価値（採用事例） 「非定型的な分析」の迅速化が、制度対応や新商品追加に貢献

松井証券 様

□ 現状

- 膨大な取引データを集計した情報系システムを運用
- 定形業務や非定型分析:複数部門で利用
 - 営業支援情報、売買動向、システム稼働の状況など多種多様なデータ活用:顧客属性や時間属性(日次/週次/月次)、商品属性を分析
- 過去10年以上のデータを蓄積
 - 直近データ保管用、長期保管用を運用

□ お客様の現状課題

- 検索結果が遅延:分析する軸に依存
- 運用負荷増:DB間のデータ移行等



Oracle Database In-Memoryがもたらすお客様価値（採用事例） 「非定型的な分析」の迅速化が、制度対応や新商品追加に貢献

松井証券 様

- お客様の課題解決に向けて
 - Oracle Database 12cへの統合
 - 分散されていた既存の2つのDBを1つに統合
 - **非定型分析やレポート作成の迅速化**
 - Database In-Memoryの採用決定
 - 既存の**アプリケーションの改修不要**
- お客様が期待しているビジネス価値（効果）
 - 分析やレポート作成の高速化による、複数部門の**業務（迅速な意思決定）への貢献**
 - NISA（少額投資非課税制度）などの**制度対応や、新商品の追加などへの柔軟な対応**
 - データベース統合による運用負荷の軽減



Oracle Database In-Memory アドバンテージ

既存資産の保護とインメモリ化による高速性の両立

インメモリ導入の容易性

インメモリ導入事前調査

設定の基本3ステップ

1. 使用するメモリ容量を設定

```
inmemory_size = XXX GB
```

2. メモリー上に格納するテーブル、
パーティションを選択

```
alter table | partition ... inmemory;
```

3. インデックスを削除

+

既存の資産を完全に保護

アプリケーションの変更なし

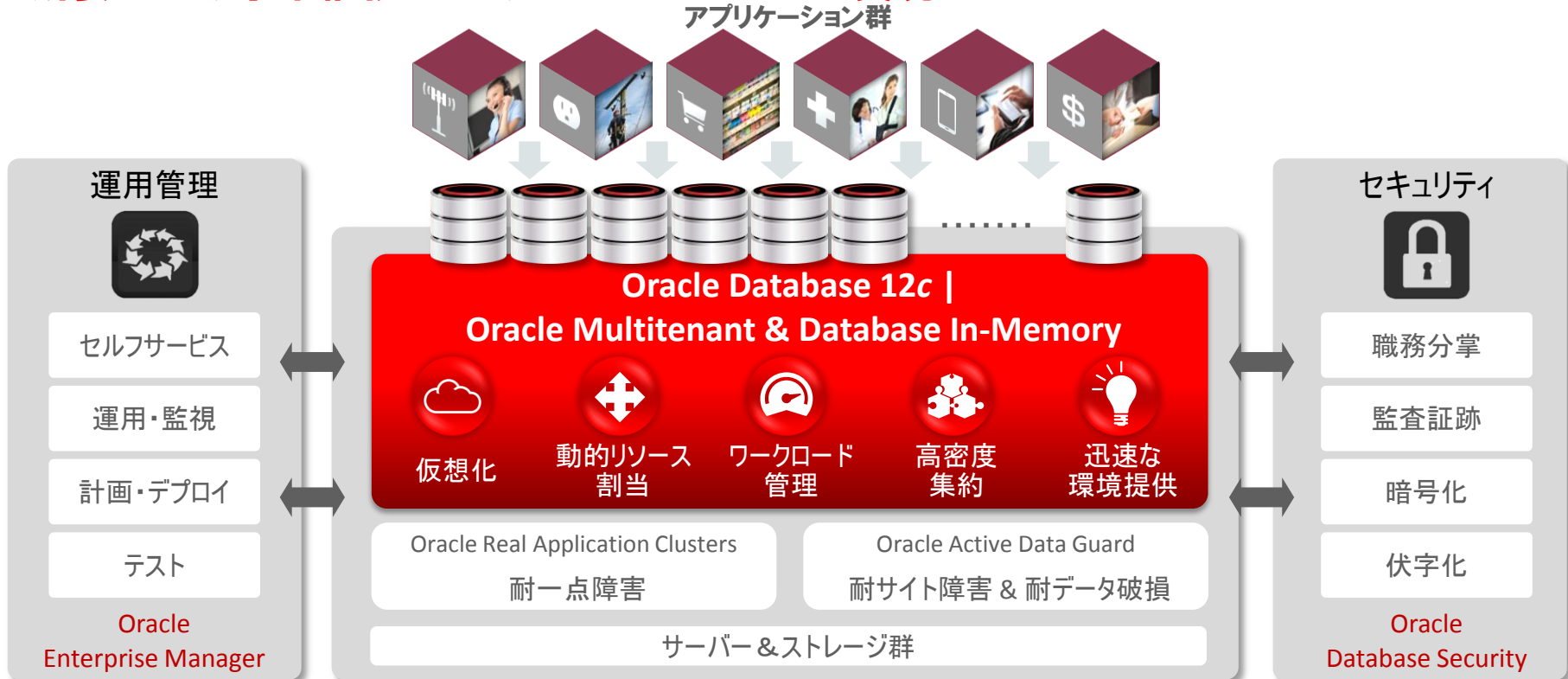
データ・フォーマットの変更なし

既存の柔軟な拡張性と融合

既存の堅牢なセキュリティと融合

プライベートデータベースクラウド基盤

所要コスト、事業継続リスク、Time-to-Marketを実現するデータベースプラットフォーム



Hardware and Software Engineered to Work Together

ORACLE®