



# データベースアップグレードに有効 Real Application Testing再入門

## Oracle Database Technology Night 資料

日本オラクル株式会社  
クラウド事業戦略統括  
データベースソリューション部  
菅原剛

2020/4





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とJavaは、Oracle Corporation 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です。

文中の社名、商品名等は各社の商標または登録商標である場合があります。

# アジェンダ

---

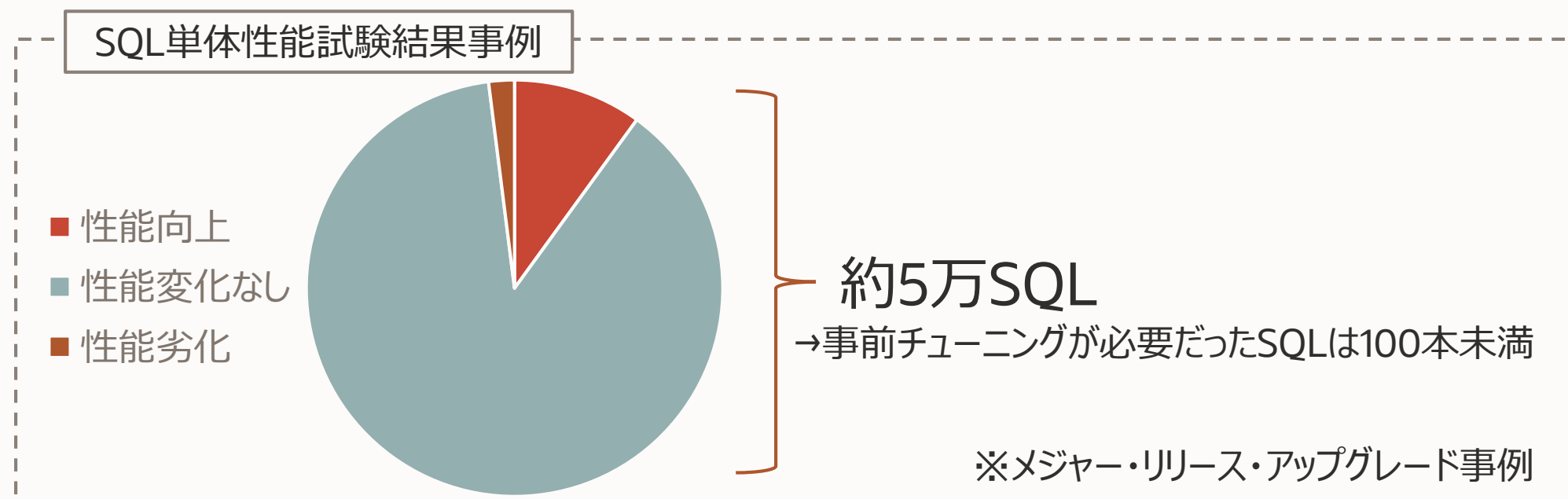
- ➡ • 製品のコンセプトと主な2つの機能
- 必要なライセンスに関して
- STSとSPA
  1. STSによるSQLのキャプチャ
  2. SPAによるテスト
  3. チューニングにSTSを活用する
  4. SPAを自己学習していただくための記事の紹介
- DB Replay
  1. DB Replayの動作概要

## DBアップグレード課題への対処策

性能が劣化するのではないか、という漠然とした不安

適切なテストを実施することで性能劣化リスクは抑えることが可能です

- カットオーバー前後に生じた障害の大半はテスト不足に起因
- 事前に性能が変動するSQLを特定し、対応しておくことが重要



# DBアップグレード課題への対処策

性能テストのハードルが高い

SQL Tuning SetやReal Application Testingを活用することで、  
高い品質のテストを、低コストで実現することが可能です

- テスト対象とするSQLを抽出する方法がわからない  
→ SQL Tuning Set（以下STS）を利用することで、  
実際に本番環境で実行されたSQLを抽出可能
- テストシナリオの策定、テスト環境の構築に工数がかかる  
→ Real Application Testing（以下RAT）を活用すると、  
シナリオ策定やテスト実行用アプリケーションが不要
- SQLの性能を1つ1つ比較するのに工数がかかる  
→ RATのSQL Performance Analyzer（以下SPA）を利用することで、  
性能比較・レポート自動作成が可能

このあとで  
ご紹介します

# システム変更の種類と、それに伴う2種類の「テスト」

## 【インフラストラクチャの変更】

### 変更内容例

- ✓ パラメーター変更
- ✓ パッチ適用
- ✓ アップグレード
- ✓ マイグレーション など

### テストに関わる要望・課題

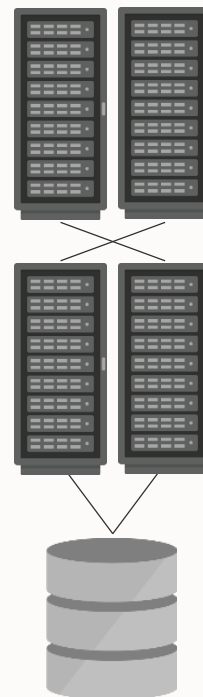
なるべく**本番通りの処理、負荷**を流して、劣化がないかチェックしたいが難しい

画面が100以上、SQLは1000以上あり、**テスト工数が大きすぎる**

インフラの変更なので、**アプリチームの理解・協力**を得るのが難しい

変更後の性能だけでなく、**現行の性能との詳細な比較**を行いたい

### 現行システム



## 【アプリケーションの変更】

### 変更内容例

- ✓ 新規開発
- ✓ 機能追加
- ✓ 機能修正 など

### テストに関わる要望・課題

新しいアプリケーションに合わせて、**テストシナリオを柔軟に簡単に作成**したい

画面への**入力値を自動的に切り替え**ながら、さまざまなパターンでテストしたい

負荷をかけたあと(かけながら)、**性能レポートをすぐさまチェック**したい

多くの開発ベンダーに委託しているので、**データをきちんとマスキング**したい

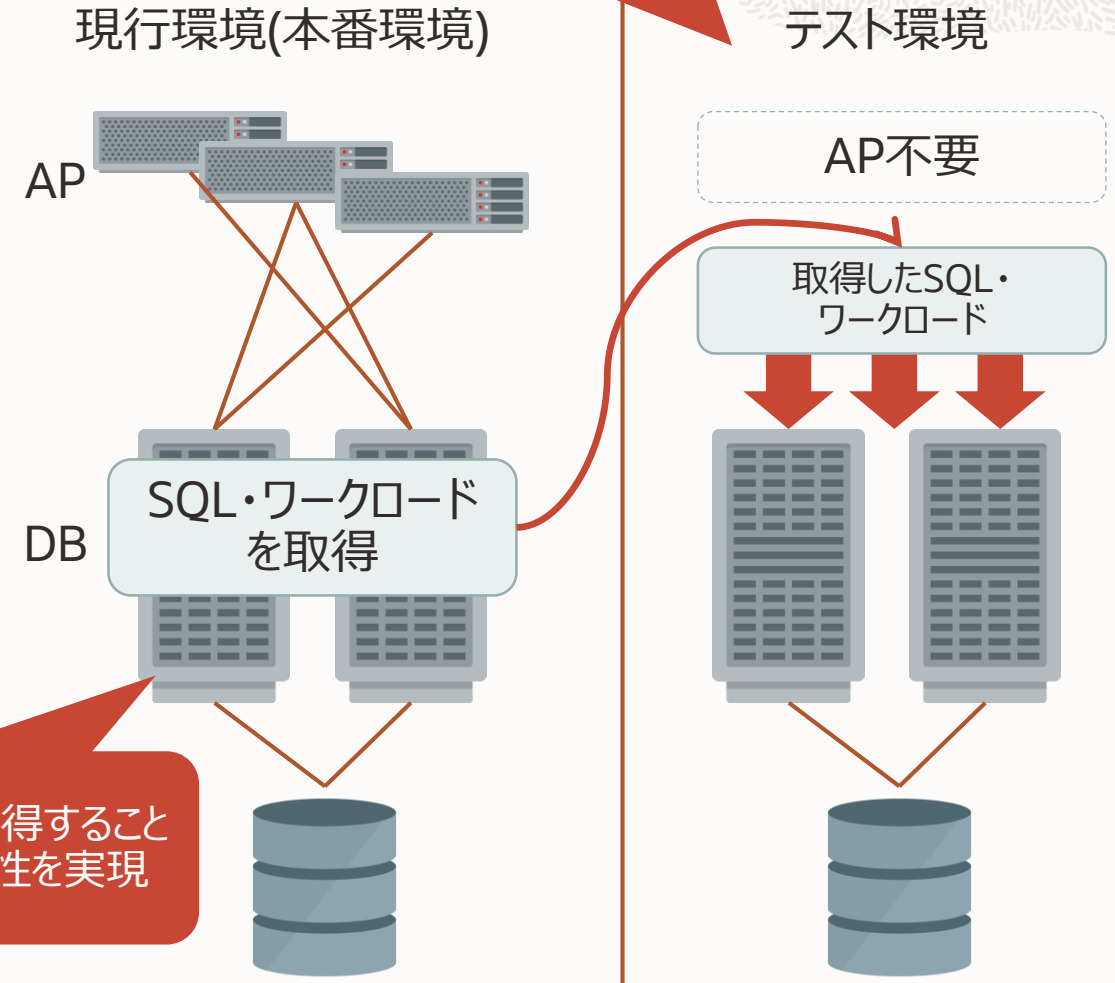
Real Application Testingの範囲

# Oracle Real Application Testing

- Oracle Database自身によるテスト機能(\*)
- 移行・アップグレード・新機能導入など  
インフラ変更に伴うテストを自動化
- 実SQL・ワークロードに基づくテストにより  
高い網羅度とテストの正確性を実現
- アプリケーションチームのテスト作業負荷を  
大幅に軽減

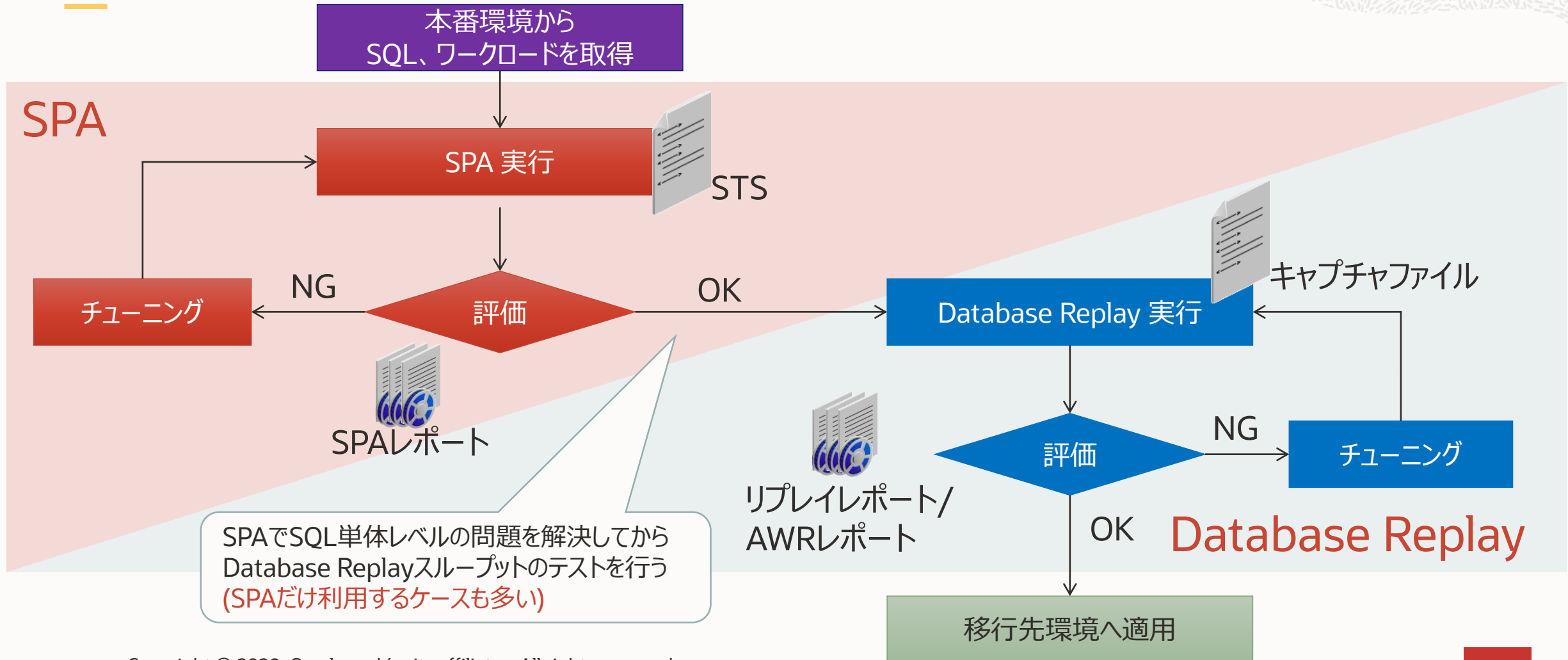
(\*) Enterprise Editionのオプション機能

実SQL・ワークロードを取得することで、高い網羅度と正確性を実現





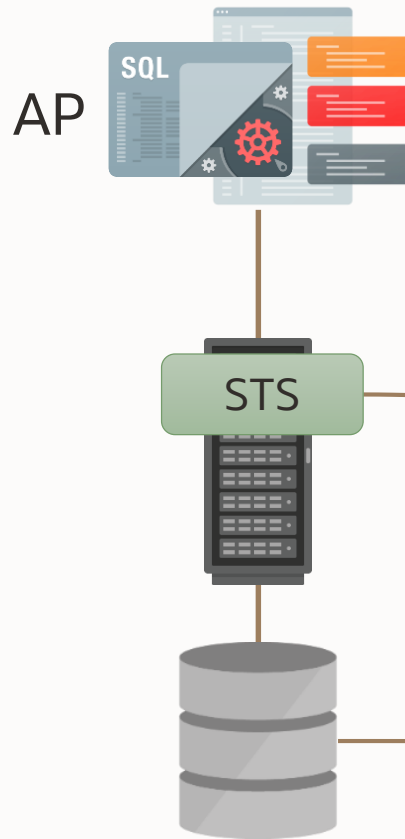
# Oracle Real Application Testing : SQL Performance Analyzer(SPA) と Database Replay





# SPAによるテストの基本形

**本番環境**  
(またはSQL取得が可能な同等の環境)

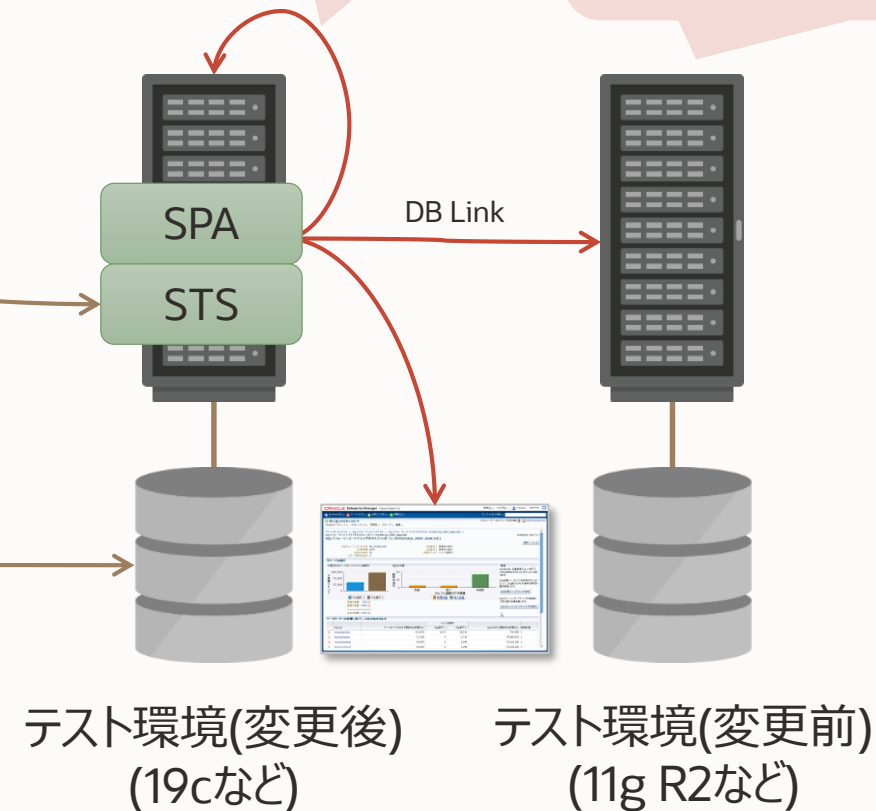


(11g R2など)

**テスト環境**

テスト環境での2つの環境での  
テスト実施結果の比較をする

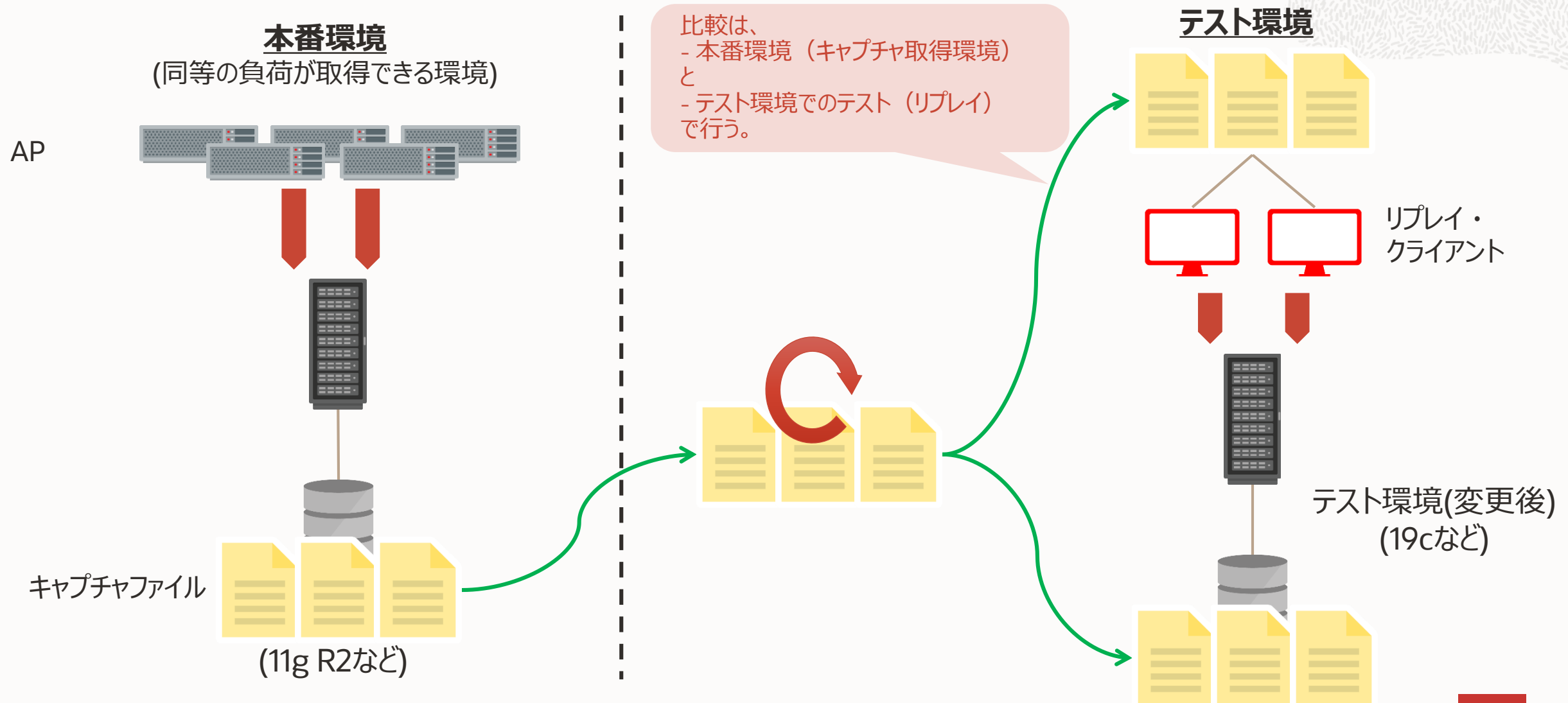
図の「テスト環境(変更前)」をテスト環境に用意  
しないで、  
本番環境での取得結果とテスト環境(変  
更後)の結果で比較することも可能



テスト環境(変更後)  
(19cなど)

テスト環境(変更前)  
(11g R2など)

# Database Replayによるテストの基本形



# アジェンダ

---

- 製品のコンセプトと2つの機能
- ➡ • 必要なライセンスに関して
- STSとSPA
  1. STSによるSQLのキャプチャ
  2. SPAによるテスト
  3. チューニングにSTSを活用する
  4. SPAを自己学習していただくための記事の紹介
- DB Replay
  1. DB Replayの動作概要

# ライセンスについての考え方(STS/SPA)

## 本番環境

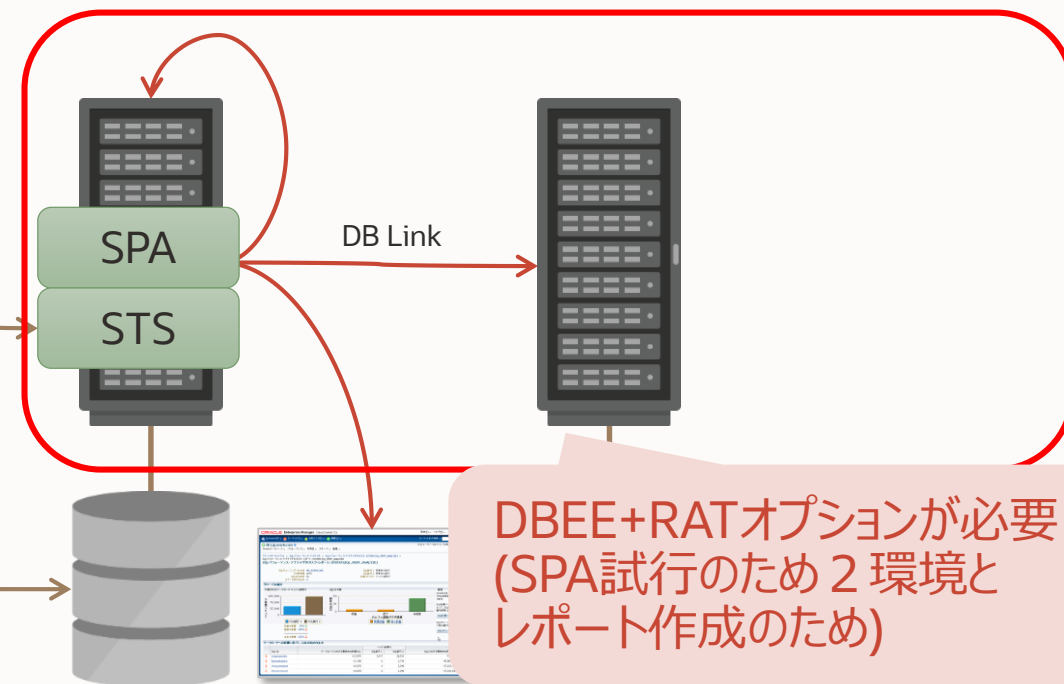
(またはSQL取得が可能な同等の環境)



STS

STSを利用するためには  
DBのEEが必要(オプションは不要)

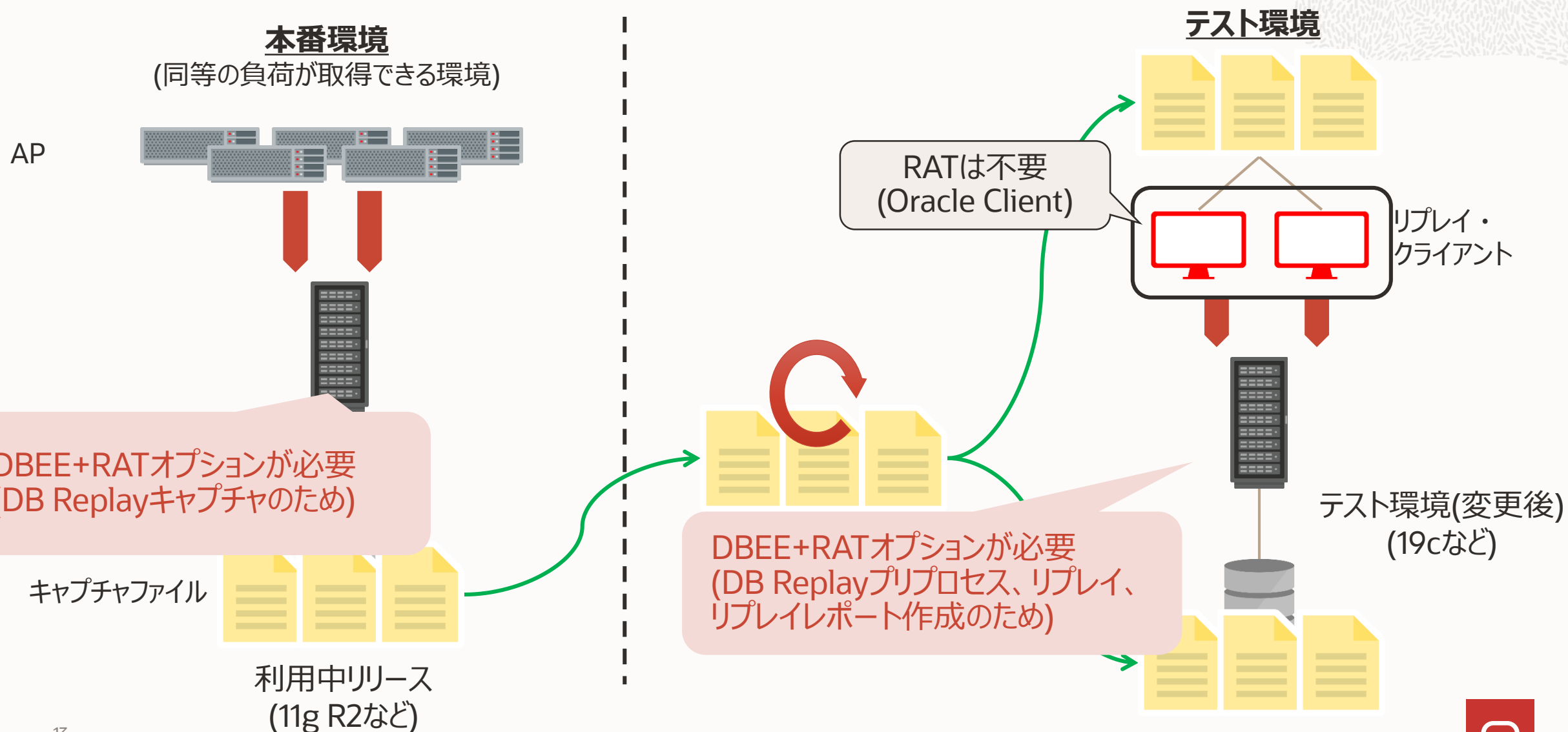
## テスト環境



テスト環境(変更後)  
(19cなど)

テスト環境(変更前)  
(11g R2など)

# ライセンスについての考え方(Database Replay)



# Oracle Cloud Infrastructure – Database Autonomous Database(ADW,ATP)に関して

RATは Oracle Cloud Infrastructure – Database のEnterprise Edition以上(Enterprise Edition, High Performance, Extreme Performance)と Exadata Cloud Serviceで利用可能です。  
※以前RATは High Performance 以上で利用可能でした。

|                                 |                                                                                                |                                                                                                                                                                         |
|---------------------------------|------------------------------------------------------------------------------------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Oracle Real Application Testing | 追加費用オプション: <b>EE、EE-ES</b><br>付属オプション: <b>PE、DBCS</b><br><b>EE、DBCS EE-HP、DBCS EE-EP、ExaCS</b> | Oracle Real Application Testingに含まれる機能は、次のとおりです。 <ul style="list-style-type: none"><li>• データベース・リプレイ</li><li>• SQLパフォーマンス・アナライザ(SPA)</li><li>• データベース統合ワークベンチ</li></ul> |
|---------------------------------|------------------------------------------------------------------------------------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------|

Oracle Database データベース・ライセンス情報ユーザー・マニュアル 19c F16968-04(原本部品番号:E94254-09) 2019年9月

## Autonomous Database(ADW,ATP)では、RATは利用できません。

Oracle Cloud, Oracle Autonomous Data Warehouseの使用 E85417-68 2019年11月

付録 B 経験豊富なOracle DatabaseユーザーのためのAutonomous Data Warehouse 制限または削除されたOracle機能のリスト

制限または削除されたOracle機能のリスト

Autonomous Data Warehouseで使用できないOracle Database機能をリストします。また、管理用に設計されているデータベース機能も使用できません。

### 削除されたOracle機能のリスト

- Oracle Real Application Testing

• Oracle Database Vault

# アジェンダ

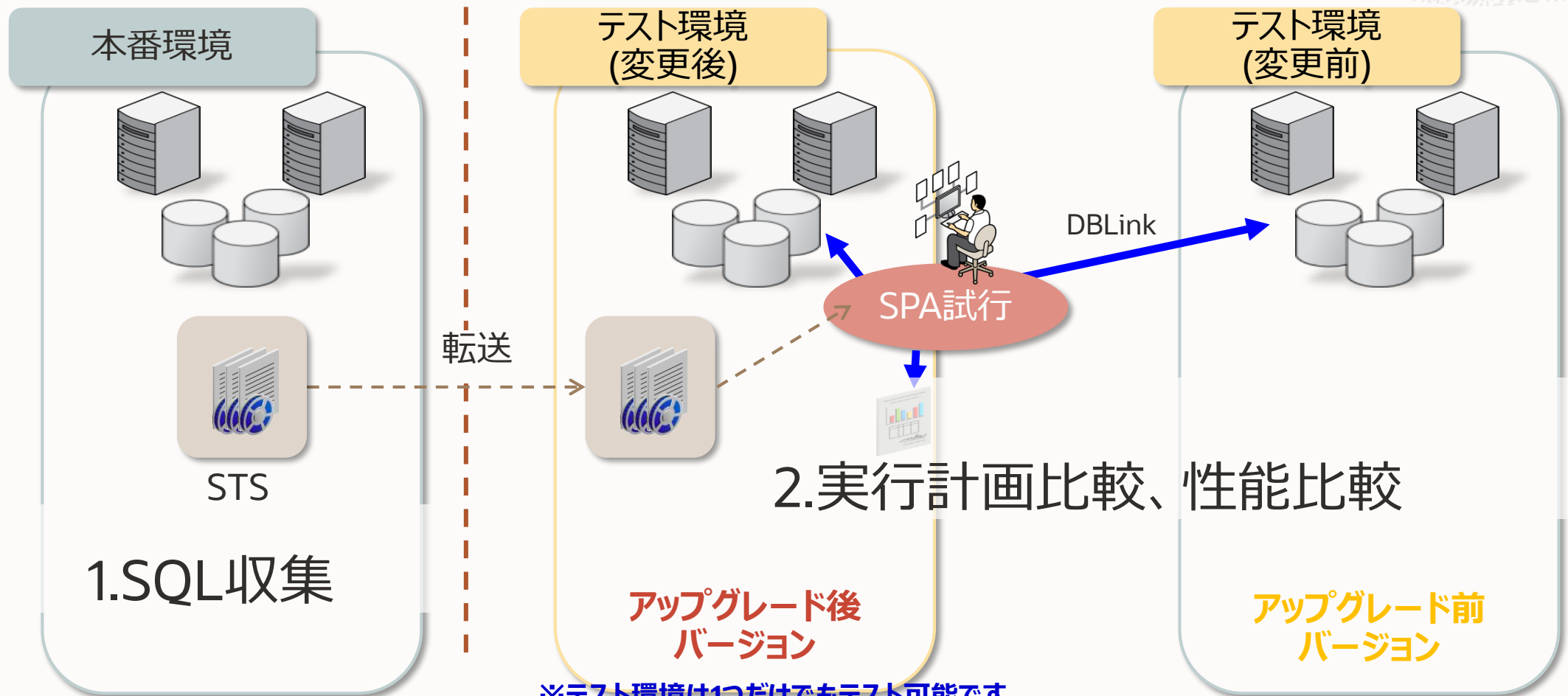
---

- 製品のコンセプトと2つの機能
  - 必要なライセンスに関して
  - STSとSPA
- 
- 1. STSによるSQLのキャプチャ
  - 2. SPAによるテスト
  - 3. チューニングにSTSを活用する
  - 4. SPAを自己学習していただくための記事の紹介
- DB Replay
  - 1. DB Replayの動作概要



# SPAを利用したSQL単体性能テスト手順概要

## 全体イメージ（概要）

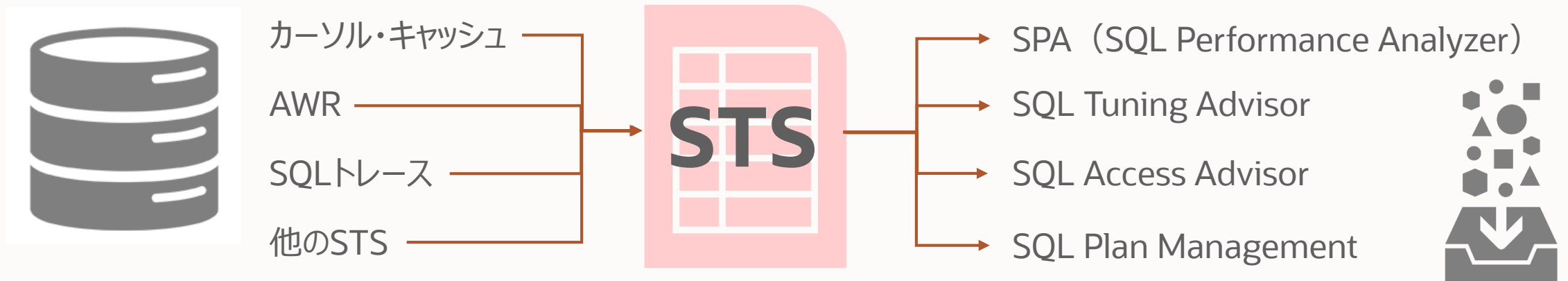


※テスト環境は1つだけでもテスト可能です。  
※Databaseのバージョンが異なる環境をデータベースリンクで接続してテストする場合、STSは上位のバージョンへインポートする必要があります。

# STSとは

## STS (SQL Tuning Set) とは...

- 下記の情報を含むデータベース・オブジェクトです(10g R2～)
  - SQL文
  - 実行コンテキスト (スキーマ、アプリケーション・モジュール名、バインド値など)
  - 実行統計 (実行時間、CPU時間、バッファ読み取り量、ディスク読み取り量など)
  - 実行計画
- RATのSPAやアドバイザ等に対するINPUTとして利用できます
- エクスポート、インポートでデータベース間を移動できます



# STS使用のTIPS

## STS使用のTIPS --STS格納時のフィルタ

- SQL情報をSTSへ格納する際、不要なSQLを除外するため、**フィルタリングの設計を行うことが重要**です
- SQLSET\_ROW型に含まれる項目は、DBMS\_SQLTUNEパッケージ使用時に、”**Basic Filter**”機能を使用してフィルタリングが可能です
- STSへ格納する際に件数を指定することも可能です

### ■フィルタリング設計例

| 項目              | Filtering目的            |
|-----------------|------------------------|
| モジュール           | DataPump処理を対象外とする      |
| スキーマ            | APの実行スキーマのみ対象とする       |
| PLAN_HASH_VALUE | “0”のものは実行計画がないので取得しない  |
| SQL_テキスト        | 特定のキーワードを持つSQLを取得対象しない |

【注意】  
レコードの数だけ判定が入るのでパフォーマンスへ影響する可能性があります

※11.2.0.4から再帰SQLを除外するパラメータ”recursive\_sql”が追加されています

# STS使用のTIPS

## STS使用のTIPS --キャプチャ時の負荷、STSのサイジング

- カーソルキャッシュからSQLをSTSに取得する場合、**負荷がかかる(\*)**ため**本番環境で実施する場合は、事前に負荷を確認しておくことを推奨**します。

一度に多数のSQLをSTSにキャプチャするケースでは、キャプチャしている期間でプロセッサコアを1つ占有することになります。

- STSはSYSAUX表領域に格納されます。多くのSQLを格納すると、SYSAUXが肥大化するため、**事前にサイジングを行う必要**があります。目安として、1SQLあたり5～20KB程度を使用します。

(※あくまで事例ベースの目安です。システム特性によって前後する可能性があります)

以下のビューからSTSを含む領域の使用量を確認することが可能です

**V\$SYSAUX\_OCCUPANTS** : コンポーネント単位

# アジェンダ

---

- 製品のコンセプトと2つの機能
- 必要なライセンスに関して
- STSとSPA
  1. STSによるSQLのキャプチャ
  2. SPAによるテスト
  3. チューニングにSTSを活用する
  4. SPAを自己学習していただくための記事の紹介
- DB Replay
  1. DB Replayの動作概要



# SPAを利用したSQL単体テスト

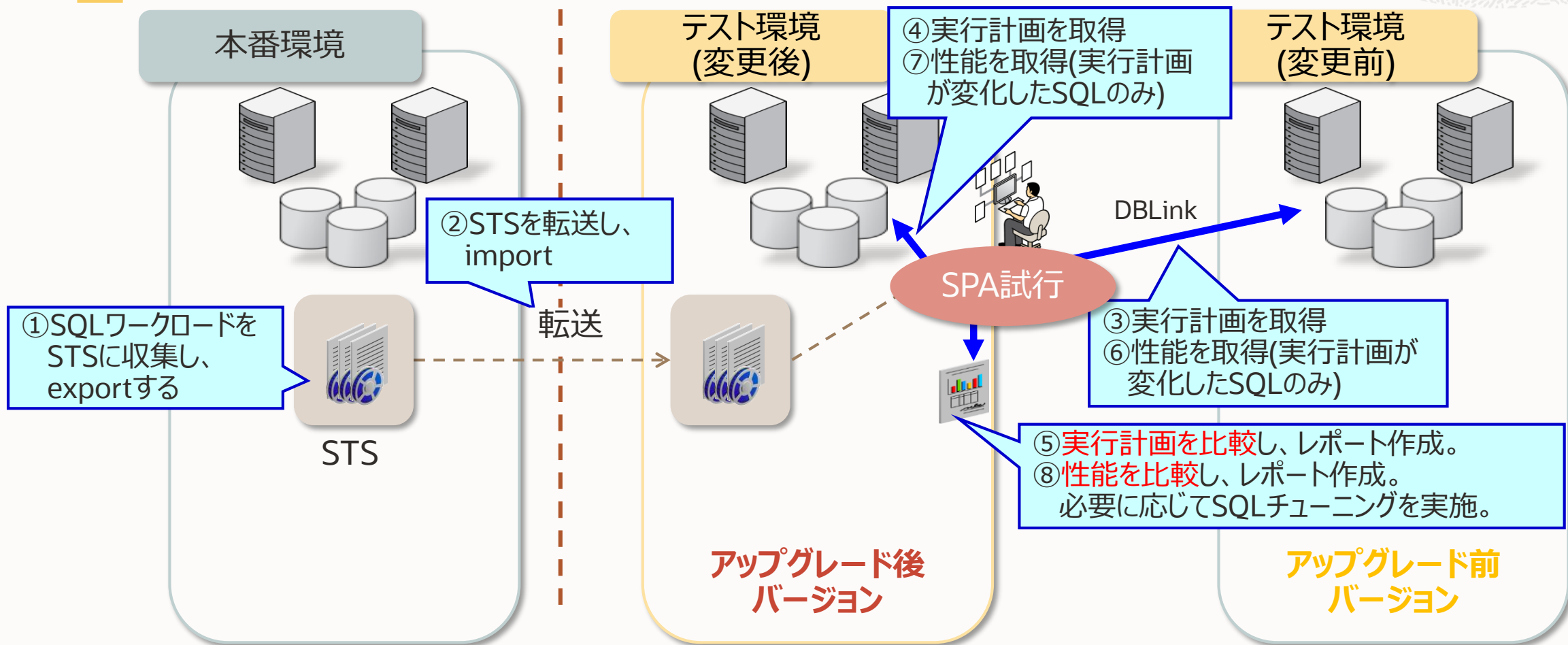
問題となるSQLの絞り込みイメージ





# SPAを利用したSQL単体テスト手順概要

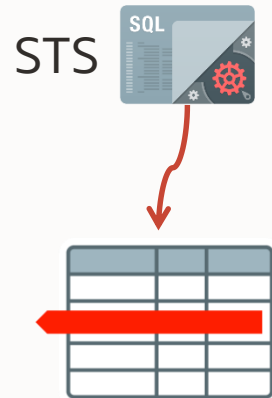
## 全体イメージ（概要フロー）





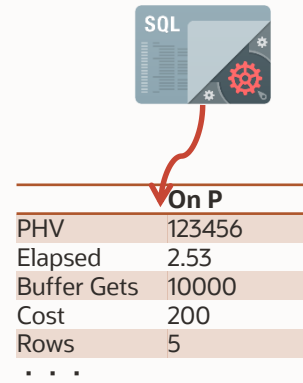
# SPA試行の3つのモード

## “Test Execute”



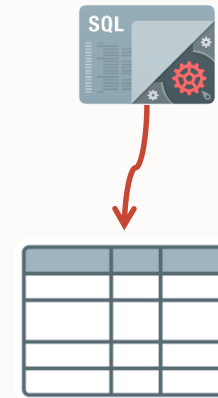
STS内のSQL文を順に実行し、実行計画や性能を記録。STS内のバインド値を使用  
**テスト環境でテストデータが使用可能な場合に使用**

## “Convert SQLSET”



STS内の実行計画・実行計画等をテスト結果として使用  
**変更前と同等の環境で性能をテストすることができない場合などに使用**

## “Explain Plan”



STS内のSQL文の実行計画を順に生成し、記録。  
EXPLAIN PLAN文と異なり、STS内に記録されたバインド値を使用  
**テスト環境でテストデータが使用できない場合に使用**

# SPAを利用したSQL単体テスト

## 実行計画比較結果イメージ

### 実行計画比較レポートのサンプル

#### Report Summary

##### Projected Workload Change Impact:

Overall Impact : 0%  
Improvement Impact : 0%  
Regression Impact : 0%

##### SQL Statement Count

| SQL Category | SQL Count | Plan Change Count |
|--------------|-----------|-------------------|
| Overall      | 97        | 7                 |
| Unchanged    | 47        | 7                 |
| with Errors  | 50        | 0                 |

} トータルのSQL数や、実行計画が変化したSQL数、実行時エラーが発生したSQL数などが分かる

##### Top 47 SQL Sorted by Absolute Value of Change Impact on the Workload

| object_id | sql_id                         | Impact on Workload | Execution Frequency | Metric Before | Metric After | Impact on SQL | Plan Change |
|-----------|--------------------------------|--------------------|---------------------|---------------|--------------|---------------|-------------|
| 251       | <a href="#">90qvs91313pid</a>  | .04%               | 1                   | 68            | 3            | 95.59%        | y           |
| 268       | <a href="#">c4572dkq4i9qt</a>  | .04%               | 1                   | 68            | 3            | 95.59%        | y           |
| 284       | <a href="#">f6krnqfs5q0xr</a>  | .04%               | 1                   | 68            | 3            | 95.59%        | y           |
| 256       | <a href="#">9k43zr1uannr6</a>  | -.04%              | 30                  | 15            | 17           | -13.33%       | n           |
| 238       | <a href="#">76ccckj4yysvua</a> | .03%               | 46                  | 2             | 1            | 50%           | n           |

} 具体的にどのSQLの実行計画が変化したかを特定できる

# SPAを利用したSQL単体テスト

## 性能比較結果イメージ

### 性能比較レポートのサンプル

Report Details

SQL Details:

Object ID

:

6

Schema Name

:

SQLDEMO

SQL ID

:

90gys91313pjd

Execution Frequency

:

1

SQL Text

:

SELECT \* FROM tab1 WHERE c1 <= 9

Execution Statistics:

| Stat Name             | Impact on Workload | Value Before | Value After | Impact on SQL |
|-----------------------|--------------------|--------------|-------------|---------------|
| elapsed_time          | 66.45%             | .000307      | .000103     | 66.45%        |
| parse_time            | -45.45%            | .000154      | .000224     | -45.45%       |
| cpu_time              | 0%                 | 0            | 0           | 0%            |
| user_io_time          | 0%                 | 0            | 0           | 0%            |
| buffer_gets           | 98.42%             | 190          | 3           | 98.42%        |
| cost                  | 95.59%             | 68           | 3           | 95.59%        |
| reads                 | 0%                 | 0            | 0           | 0%            |
| writes                | 0%                 | 0            | 0           | 0%            |
| io_interconnect_bytes | 0%                 | 0            | 0           | 0%            |
| rows                  |                    | 9            | 9           |               |

指定した性能項目(SQL実行時間等)で比較を行い、SQL毎に比較レポートが作成される。

アップグレード前バージョンとアップグレード後バージョンで実行した際の性能比較結果が表示される。  
許容できる変動量を超えたものをチューニング対象とする。

# SPAを利用したSQL単体テスト

SPAで実行性能の比較を実施にあたり注意する点

テスト環境で実行性能を測定するためには、以下を注意する必要があります。

- 本番相当のデータが存在している
- 適切なタイミングで統計情報が取得されている
- オブティマイザの動作に影響のある初期化パラメータを変更する際は、性能測定を再度やり直す必要がある。  
(影響のあるパラメータはV\$SYS\_OPTIMIZER\_ENVで確認可能)

# テスト用クラウド環境に本番環境と同等のデータを準備する際の障壁と対策

本番環境と同等のデータを準備する際の障壁

- **大量データを転送できない**

(対策案) テスト環境にデータがない場合でも、実行計画の生成ならば可能  
SPA試行のモード「Explain Plan」の活用、テスト環境には統計情報を移動

※この場合、変更前後の比較の範囲が「エラー発生有無」「実行計画の比較」までになります。

- **セキュリティを考慮して本番環境のデータを持ち出せない**

(対策案) データをマスクして持ち出す

Oracle Data Masking and Subsetting Packの活用

# 構成パターン案A(基本形)

## 本番環境

(またはSQL取得が可能な同等の環境)

AP



1. SQLチューニングセット(STS)を使用し、SQL情報を取得

STS



DB: 旧バージョン

2. 実データ(一定量のサブセット)をテスト環境にセットして、統計情報も取得



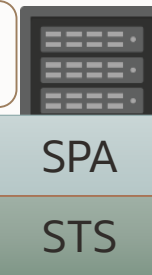
## Oracle Cloud上の テスト環境

ORACLE®

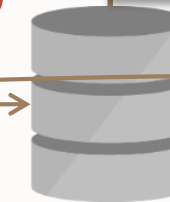
変更後環境  
(新バージョン)

変更前環境  
(旧バージョン)

3. STSをテスト環境に  
インポート



STS



DB: 新バージョン

4. 変更前/変更後の環境に  
対してSQL実行、性能測定  
(SPA試行)

DB Link



「エラー発生有無」  
「実行計画の比較」  
「実行性能の比較」を実施

5. レポート生成



DB: 旧バージョン



# 構成パターン案B (テスト用のデータをマスキングで作成する)

## 本番環境

(またはSQL取得が可能な同等の環境)

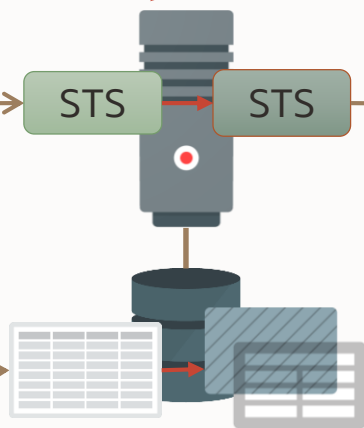


1. SQLチューニングセット(STS)を使用し、SQL情報を取得

## マスキング環境

(または本番環境内)

2. データ/STSをマスキング



4. STSをテスト環境にインポート

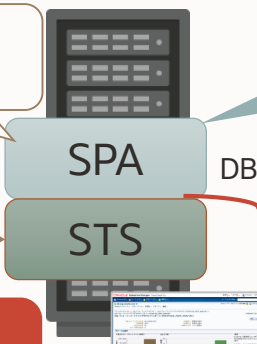
3. マスクしたデータを  
ステージング環境にセット  
統計情報も取得

## Oracle Cloud上の テスト環境

ORACLE®

変更後環境  
(新バージョン)

変更前環境  
(旧バージョン)



5. 変更前/変更後の環境に  
対してSQL実行、性能測定  
(SPA試行)

「エラー発生有無」  
「実行計画の比較」  
「実行性能の比較」を実施

6. レポート生成

DB: 旧バージョン



# 構成パターン案C (SPA試行のモード「Explain Plan」「Convert SQLSETの活用)

**本番環境**

(またはSQL取得が可能な同等の環境)

AP



**エクスポート環境**

(または本番環境内)

1. SQLチューニングセット(STS)を使用し、SQL情報を取得

STS



DB: 旧バージョン

2. 本番データを用いて  
新バージョンの**オプティマイザ**  
統計情報を取得※



DB: 新バージョン

**データ移行なし**



**Oracle Cloud上の**

**テスト環境**

**ORACLE®**

変更後環境  
(新バージョン)

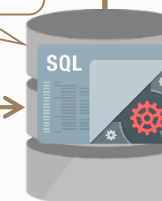
4. STSをCloud環境に  
インポート



SPA

STS

3. オプティマイザ統計情報を  
テスト環境にセット



DB: 新バージョン

5. 変更後の環境に対して  
SPA試行(Explain Plan)  
旧バージョンの実行計画は  
STSの中から取り出す  
(Convert SQLSET)

**「エラー発生有無」  
「実行計画の比較」を実施**

6. レポート生成



## 各構成パターン特徴・考慮事項

| 考慮事項            | 構成パターンA | 構成パターンB | 構成パターンC |
|-----------------|---------|---------|---------|
| テスト精度           | 高       | 高       | 中       |
| 本番相当のデータ利用      | 利用      | 利用(マスク) | 利用なし    |
| SQL実行性能の比較      | 可       | 可       | 不可      |
| 統計情報のインポート      | 不要      | 不要      | 必要(*1)  |
| 比較対象(旧バージョン)の考慮 | 必要(*2)  | 必要(*2)  | 不要      |

(\*1), (\*2): 考慮事項として次スライドに記載  
コストは保有しているライセンスに依存

## テスト環境に、本番環境と同等のデータ/バージョンの準備が難しい場合の対策 (前スライドの考慮事項に関して)

### 統計情報のインポート (\*1)

- 統計情報は、リリースにより変更が入る(取得情報に差異)
- バージョンアップのための検証でテスト精度をあげるためには、テスト環境へ本番データ(相当)を用いた統計情報が必要
- 本番環境からテスト環境へ新バージョンでの統計情報のエクスポート/インポートを実施

➡ **構成パターンCは、新バージョンでの統計情報をテスト環境に持ち込むことを検討**

### 比較対象(旧バージョン)の考慮 (\*2)

- テスト環境では、本番環境と同じOracle DatabaseのバージョンでSPA試行が実施できないケースがある
- SPA試行のモード「Convert SQLSET」を使用することで、本番環境で取得したSTS内の実行計画、実行性能をテスト結果として使用できる。ただしSTS取得時のSQL実行環境とテスト環境での比較になる点に注意が必要（H/W性能差など）。

➡ **構成パターンCは、STS内に取得されている情報とテスト環境での実施結果を比較**

# アジェンダ

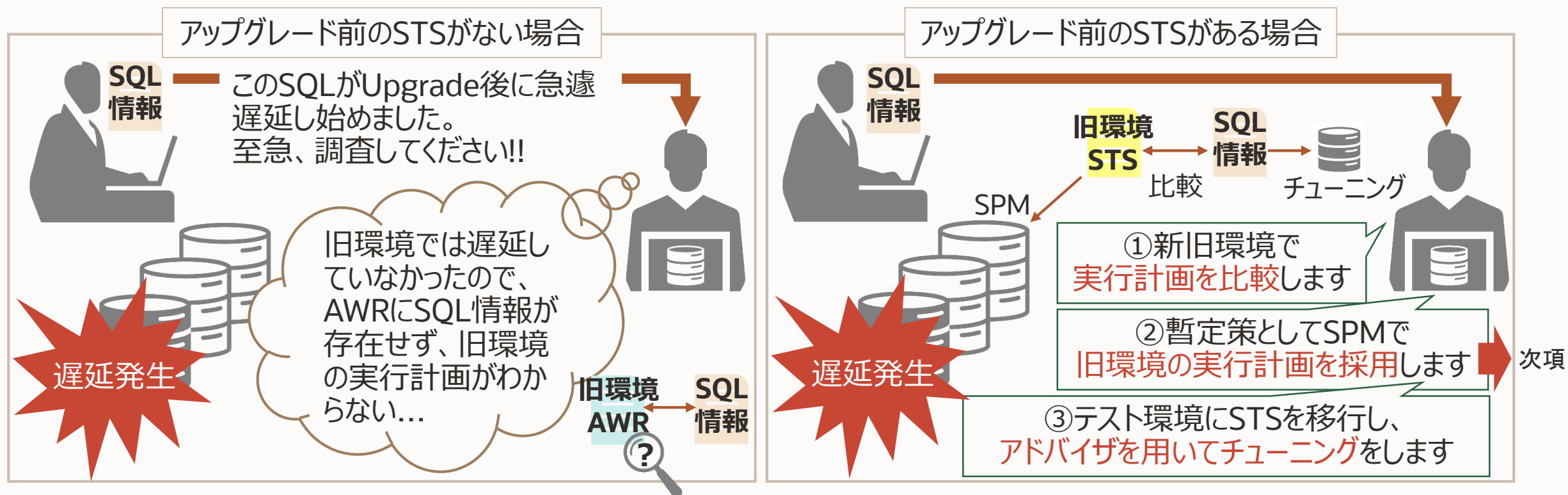
---

- 製品のコンセプトと2つの機能
- 必要なライセンスに関して
- STSとSPA
  1. STSによるSQLのキャプチャ
  2. SPAによるテスト
  - ➡ 3. チューニングにSTSを活用する
  4. SPAを自己学習していただくための記事の紹介
- DB Replay
  1. DB Replayの動作概要

# STS使用シナリオ

## チューニングにSTSを活用する

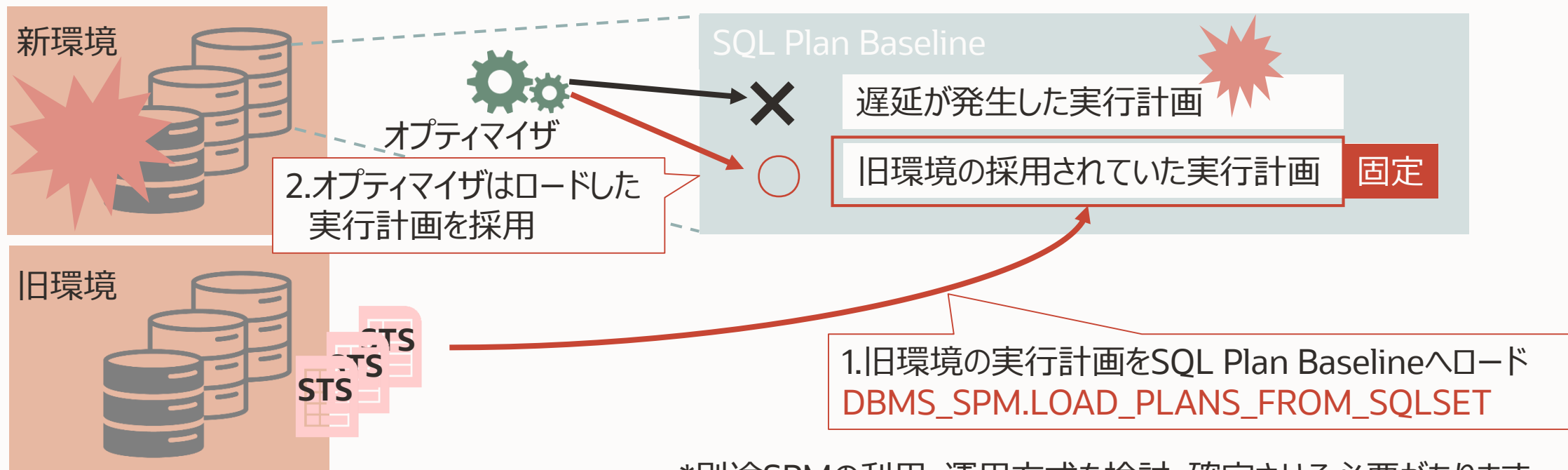
STSに格納されたアップグレード前のSQL情報は次のように利用可能です



# STS使用シナリオ

チューニングにSTSを活用する(STSとSQL Plan Managementで実行計画を制御)

SQL Plan Management(SPM)で旧環境の実行計画を採用(\*)  
旧環境で取得したSTSに含まれる実行計画をSQL Plan Baselineにロードし、SPM管理下に置くことで、  
旧環境の実行計画を採用させることが可能になります。障害に対する暫定処置として活用できることがあります。



\*別途SPMの利用・運用方式を検討・確定させる必要があります。





# アジェンダ

---

- 製品のコンセプトと2つの機能
- 必要なライセンスに関して
- STSとSPA
  1. STSによるSQLのキャプチャ
  2. SPAによるテスト
  3. チューニングにSTSを活用する
  - ➡ 4. SPAを自己学習していただくための記事の紹介
- DB Replay
  1. DB Replayの動作概要

## (参考) SPAを自己学習していただくための記事の紹介

DBインスタンスとSQL実行環境でSPA機能を自己学習いただくため記事を公開しています。

A. Oracle DatabaseのSQL Tuning Set(STS)の活用 : STSへSQL実行時の情報を記録する

[https://qiita.com/ts\\_carp/items/1b1f3ec0b8ec55c56c67](https://qiita.com/ts_carp/items/1b1f3ec0b8ec55c56c67)

B. Oracle DatabaseのSQL Tuning Set(STS)の活用 : STSをデータベース間で移動 (コピー) する

[https://qiita.com/ts\\_carp/items/eee8f1fa3d69c3b91f8a](https://qiita.com/ts_carp/items/eee8f1fa3d69c3b91f8a)

C. Oracle Real Application TestingのSQL Performance Analyzerの活用 : データベースのアップグレードでのSQLテスト

[https://qiita.com/ts\\_carp/items/a52c7a4063f9e6226dd6](https://qiita.com/ts_carp/items/a52c7a4063f9e6226dd6)

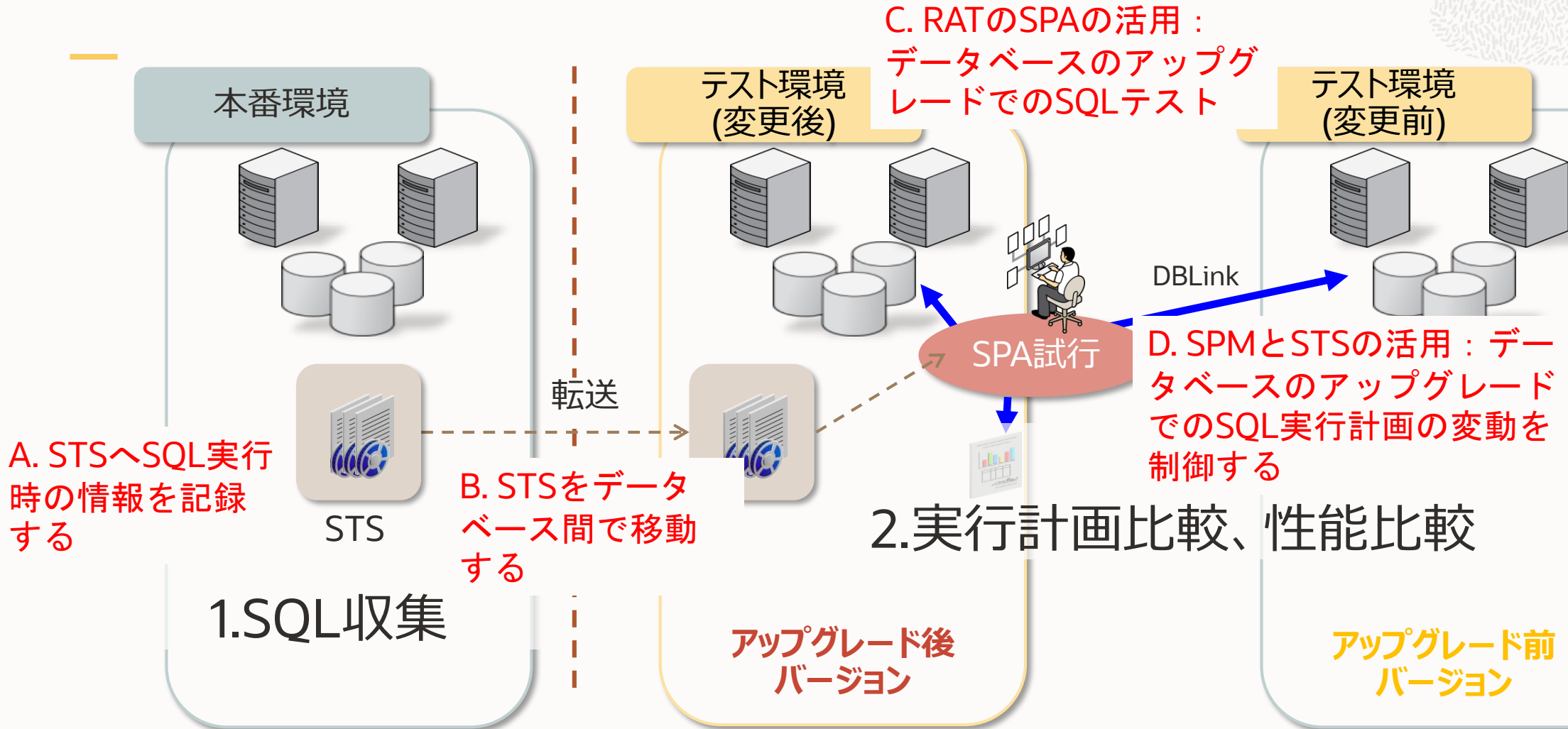
データベースのアップグレードを想定したシナリオでSPAの利用手順を具体的に説明しています。

D. Oracle DatabaseのSQL Plan Management(SPM)とSQL Tuning Set(STS)の活用 : データベースのアップグレードでのSQL実行計画の変動を制御する

[https://qiita.com/ts\\_carp/items/1edb0e6780b755dfcd8c](https://qiita.com/ts_carp/items/1edb0e6780b755dfcd8c)

STSとSPMを組み合わせて、実行計画の変動を制御する手順を説明しています。

## (参考) SPAを自己学習していただくための記事の紹介



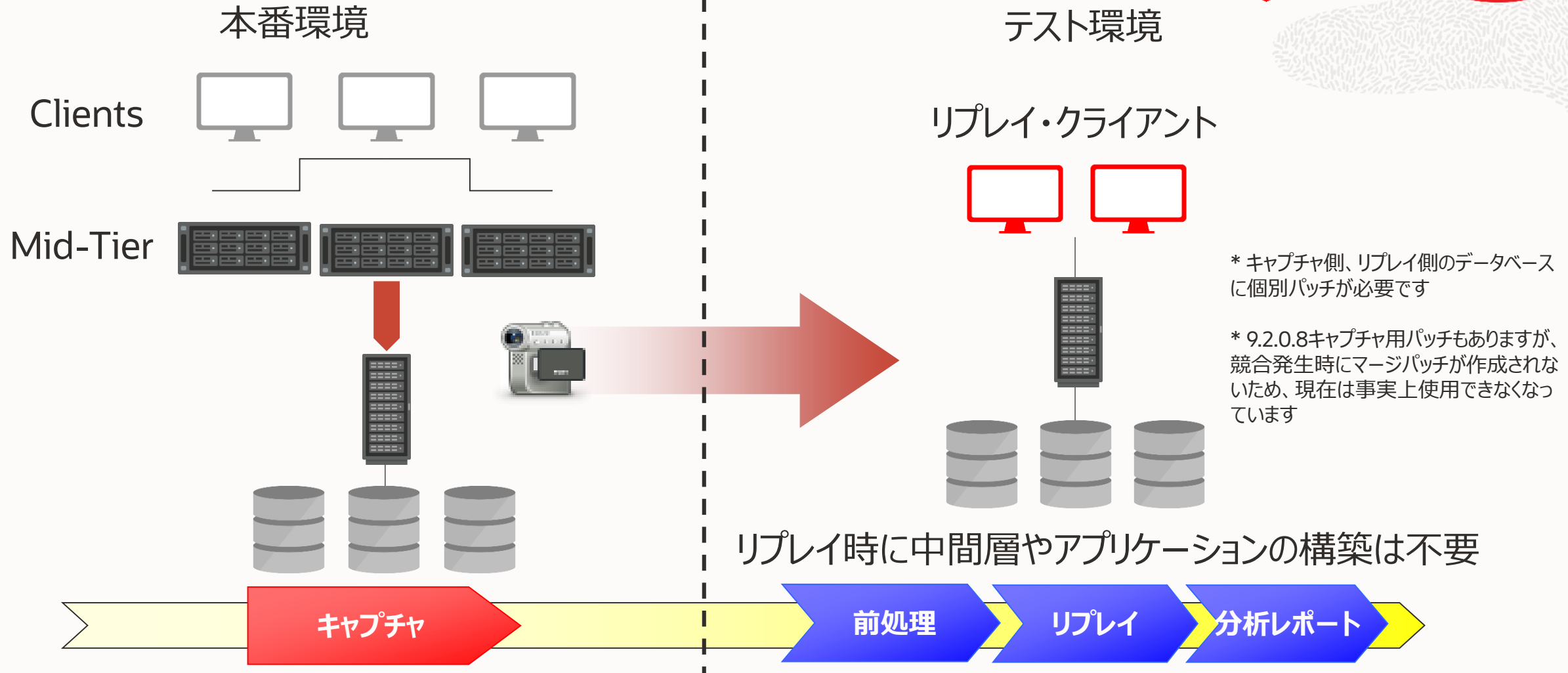
※テスト環境は1つだけでもテスト可能です。  
※Databaseのバージョンが異なる環境をデータベースリンクで接続してテストする場合、STSは上位のバージョンへインポートする必要があります。

# アジェンダ

---

- 製品のコンセプトと2つの機能
- 必要なライセンスに関して
- STSとSPA
  1. STSによるSQLのキャプチャ
  2. SPAによるテスト
  3. チューニングにSTSを活用する
  4. SPAを自己学習していただくための記事の紹介
- ➡ • DB Replay
  1. DB Replayの動作概要

# Database Replayを利用したデータベースワークロードのテスト

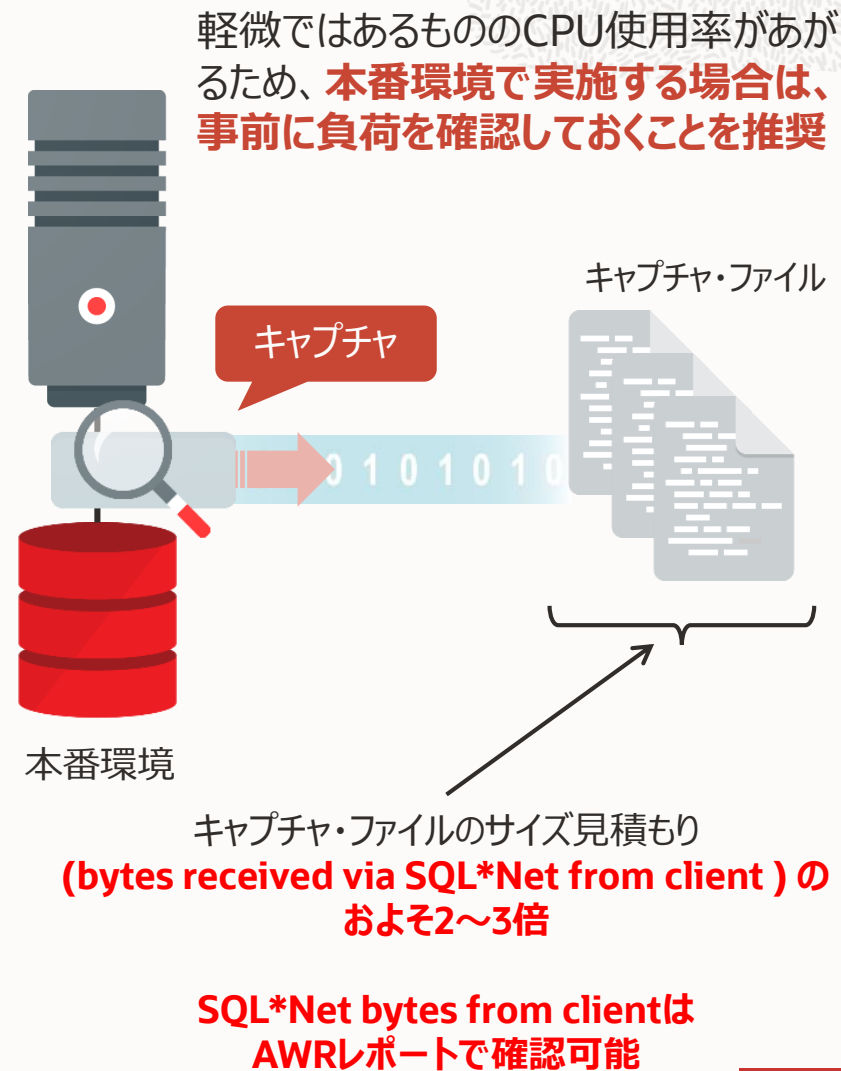


**Database Replay** : スループット、リソース使用量のテストができる

# Database Replayを利用したデータベースワークロードのテスト

## Step 1. ワークロードのキャプチャ

- クライアントからの全リクエストをバイナリファイルにキャプチャ可能(キャプチャ・ファイル)
  - バックグラウンド処理は除外
- フィルターを指定し、特定のワークロードの取得/除外が可能
  - セッションごとに、\*\*\*.rec というファイルを作成
- キャプチャの実行時間を指定
- RAC 環境ではキャプチャ・ファイルの保存先に共有ファイルとローカルファイルの両方をサポート





# Database Replayを利用したデータベースワークロードのテスト

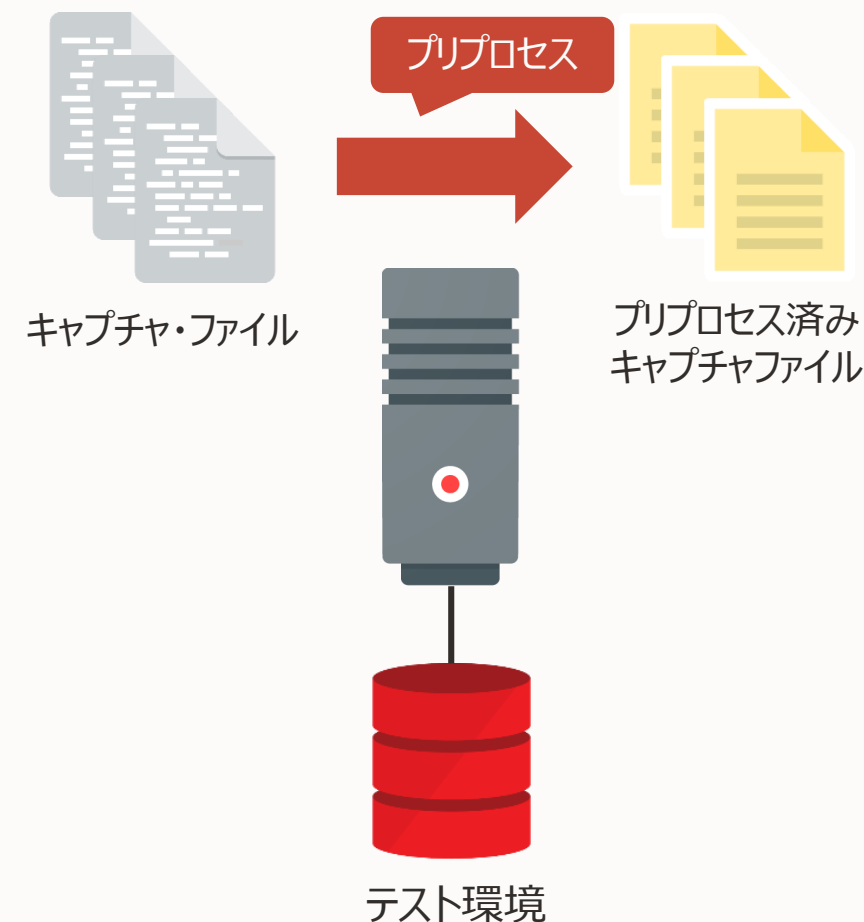
## Step 2. ワークロードの前処理

### 事前準備

- 本番環境で取得したキャプチャ・ファイルをテスト環境へコピー
- データを本番環境でのキャプチャ開始時点の内容に合わせておく

### プリプロセス

- キャプチャ・ファイルをリプレイ可能なフォーマットに変換  
※リプレイする環境で実施することを推奨
- プリプロセス済みファイルは何度でもリプレイ可能
- RAC 環境でキャプチャした場合は、全インスタンスで生成されたキャプチャ・ファイルを一か所に集め、プリプロセスを実施



# Database Replayを利用したデータベースワークロードのテスト

## Step 3. ワークロードのリプレイ

- 本番環境で取得されたワークロードの実行時間、並列度、コミット順を再現してリプレイ
- リプレイ・クライアントがリクエストをDBサーバに送信
- クライアントはスレッドで実行されるため、1プロセスで複数並列度を再現可能。  
負荷が大きい場合には必要に応じてクライアントを複数プロセス起動する（1プロセス最大50セッション）。

リプレイ前に、リプレイクライアントをキャリブレートモードで実行することで、必要なリプレイクライアント数の情報が表示される。キャリブレーション・モード（wrc mode = calibrate）

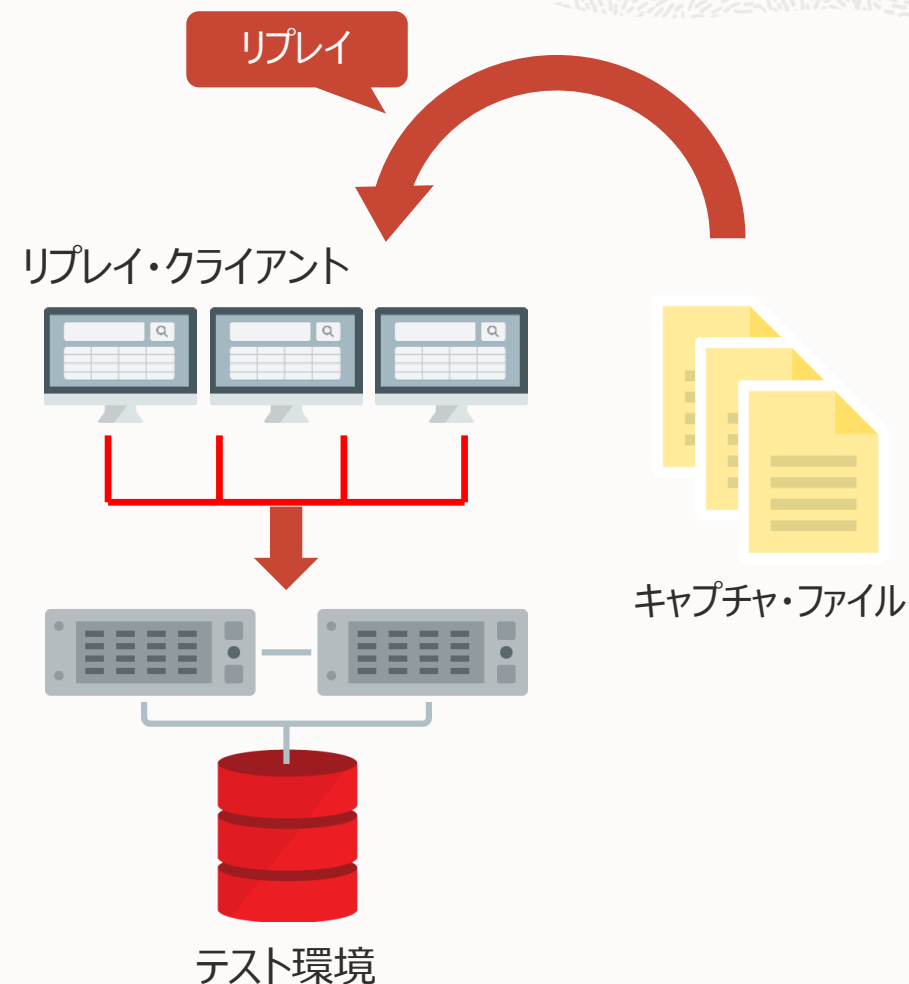
参考：

マニュアル Oracle Database Testingガイド 19c

[データベース・リプレイ] - [データベース・ワークロードのリプレイ]

- [リプレイ・クライアントの設定] - [リプレイ・クライアントの較正]

How To Start Multiple Workload Replay Clients in a Single Terminal Session (WRC) (Doc ID 2261505.1)



# Database Replayを利用したデータベースワークロードのテスト

## Step 4. 分析およびレポート

### キャプチャ時とリプレイ時の違いをレポート

- パフォーマンスの違い
- エラーの違い
  - リプレイ時に発生した新規エラー
  - キャプチャ時に発生していたがリプレイ時に発生しなかったエラー
  - キャプチャ時に発生していたがリプレイ時に変異したエラー
- データの違い
  - キャプチャ時と異なる行数が変更された DML
  - キャプチャ時と異なる件数が返された SELECT

### リプレイ実行後、比較レポートを生成

ワークロード・リプレイの表示: REPLAY-single-20071116164115

ステータス 完了

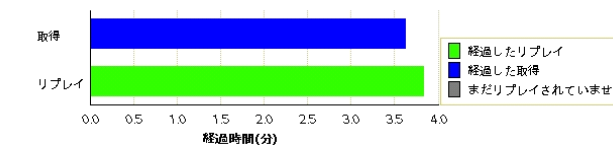
#### ▼サマリー

|                |                              |                |                         |
|----------------|------------------------------|----------------|-------------------------|
| リプレイ名          | REPLAY-single-20071116164115 | 取得名            | capture_test            |
| ディレクトリ・オブジェクト  | RAT_DIR ID                   | 継続時間(hh:mm:ss) | 00:03:50                |
| データベース名        | SINGLE                       | 準備時間           | 2007/11/16 16:41:22 JST |
| DBID           | 3609572006                   | 開始時間           | 2007/11/16 16:41:31 JST |
| リプレイ・エラー・コード   | N/A                          | 終了時間           | 2007/11/16 16:45:21 JST |
| リプレイ・エラー・メッセージ | なし                           |                |                         |

ワークロード・プロファイル 接続マッピング パラメータのリプレイ レポート

ネットワーク時間(hh:mm:ss) 00:00:15 クライアント 1  
思考時間(hh:mm:ss) 00:06:44 完了済クライアント 1

#### 経過時間の比較



#### リプレイの評価

「経過時間の比較」グラフは、リプレイされたワークロードで、取得されたのと同じ作業量に達するまでにかかった時間を示します。

リプレイ・バーが取得バーよりも短い場合は、リプレイ環境が取得環境よりも速くワークロードを処理しています。

相違表は、リプレイ環境と取得環境の間のデータおよびエラー両方の矛盾点に関する情報を提供します。これは、リプレイの質の測定値として使用できます。

#### Replay Information

| Information          | Replay                        | Capture                      |
|----------------------|-------------------------------|------------------------------|
| Name                 | 10GREPLAY                     | 10g_capture                  |
| Status               | COMPLETED                     | COMPLETED                    |
| Database Name        | IZIZ1JIZ                      | POPD1TOM                     |
| Database Version     | 11.2.0.3.0                    | 10.2.0.3.0                   |
| Start Time           | 06-09-12 11:01:53             | 05-09-12 13:43:23            |
| End Time             | 06-09-12 13:21:22             | 05-09-12 14:56:16            |
| Duration             | 2 hours 19 minutes 29 seconds | 1 hour 12 minutes 53 seconds |
| Directory Object     | RECDIR                        | RECDIR                       |
| Directory Path       | /home/oracle/ocs/RECDIR       | /home/oracle/ocs/RECDIR      |
| AWR DB Id            | 2083442946                    |                              |
| AWR Begin Snap Id    | 2863                          |                              |
| AWR End Snap Id      | 2878                          |                              |
| Replay Schedule Name |                               |                              |

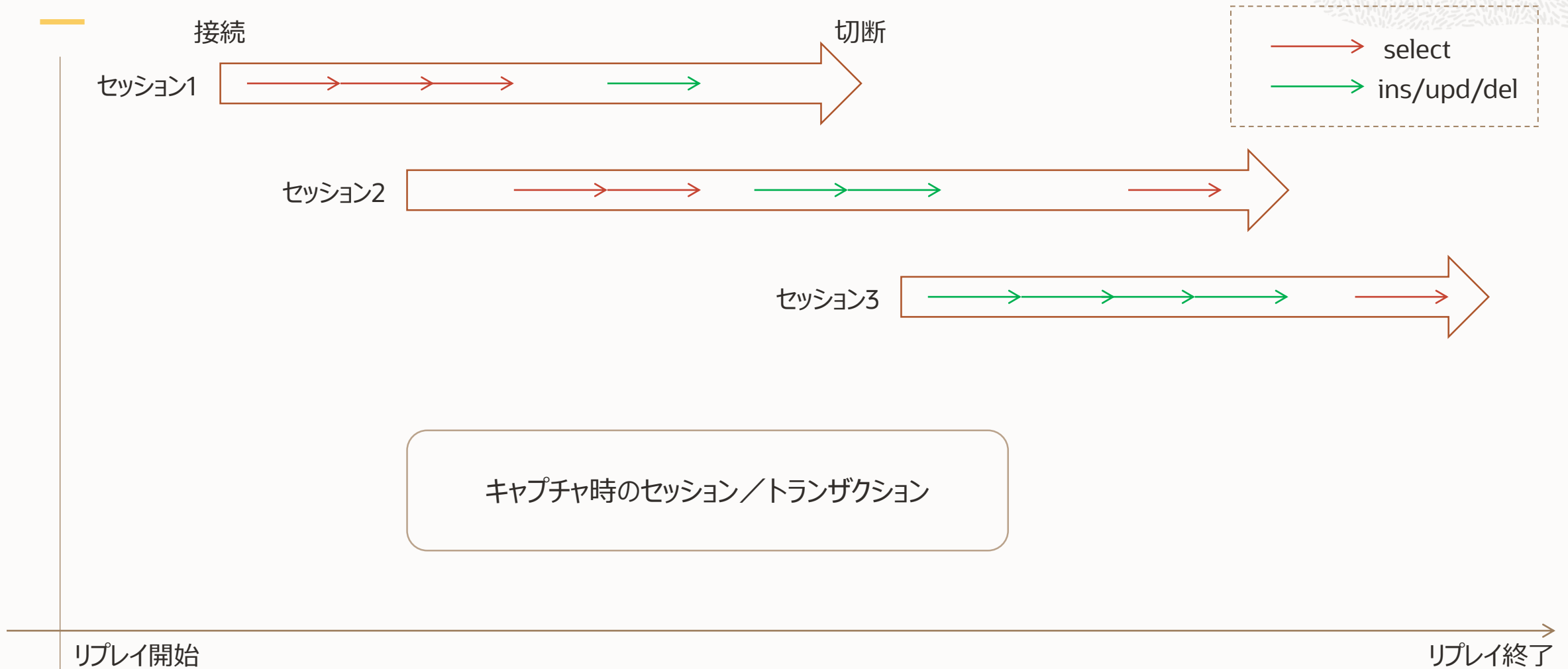
#### Replay Options

| Option Name             | Value                      |
|-------------------------|----------------------------|
| Synchronization         | SCN                        |
| Connect Time            | 0%                         |
| Think Time              | 0%                         |
| Think Time Auto Correct | TRUE                       |
| Number of WRC Clients   | 7 (7 Completed, 0 Running) |

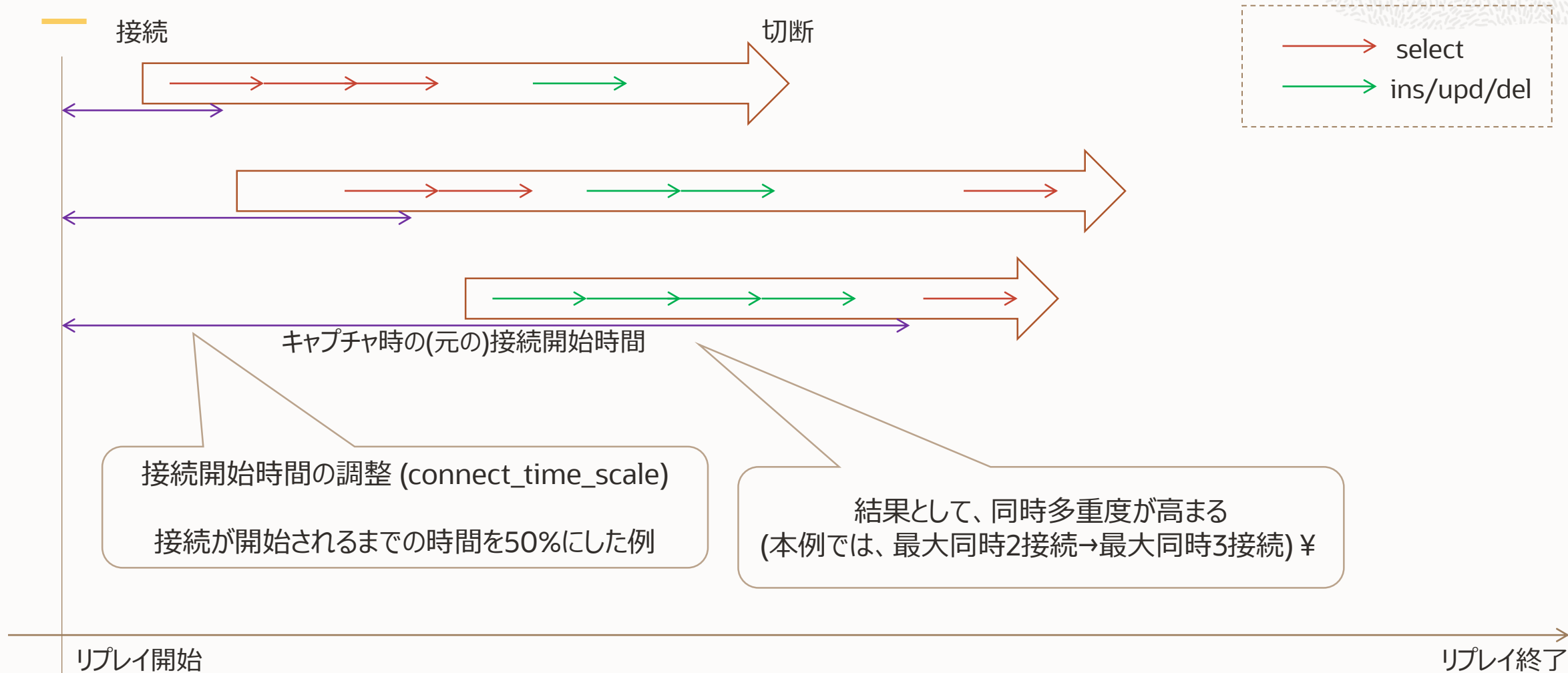
## 参考 : リプレイ・オプション

| パラメータ                   | 説明                                                                                                                                                                                                                               |
|-------------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Synchronization         | <p>このパラメータはワークロード・リプレイ時に使用される同期のタイプを決定します。このパラメータがSCNに設定された場合、取得されたワークロードの COMMIT 順序はリプレイ中もグローバルに保持され、取得時間SCNが小さいすべてのCOMMITアクションが完了した後でのみ、リプレイされたすべてのリクエストが実施されます。デフォルト値はSCNです。</p> <p>TIMEに設定された場合、同期は取得時のアクションの実行時間に基づきます。</p> |
| connect_time_scale      | <p>ワークロード取得が開始されてから、指定した値でセッションが接続されるまでの経過時間を変更します。入力は、%値として解釈されます。</p> <p>ワークロードのリプレイ中に同時ユーザー数を増加または削減する場合に使用できます。DEFAULT VALUEは100です。</p>                                                                                      |
| think_time_scale        | <p>同じセッションからの2つの連続したユーザー・コール間の経過時間を変更します。入力は、%値として解釈されます。</p> <p>ワークロードのリプレイ中に同時ユーザー数を増加または削減する場合に使用できます。DEFAULT VALUEは100です。</p>                                                                                                |
| think_time_auto_correct | <p>リプレイでのユーザー・コールの完了にかかる時間が、元の取得で同じユーザー・コールの完了にかかった時間よりも長くなる場合に、コール間の思考時間を適切に自動修正します。</p> <p>DEFAULTのTRUEでは、リプレイが取得よりも遅くなった場合に思考時間が短縮されます。</p>                                                                                   |

# Database Replayのリプレイオプションでできること(1/5)



## Database Replayのリプレイオプションでできること(2/5)



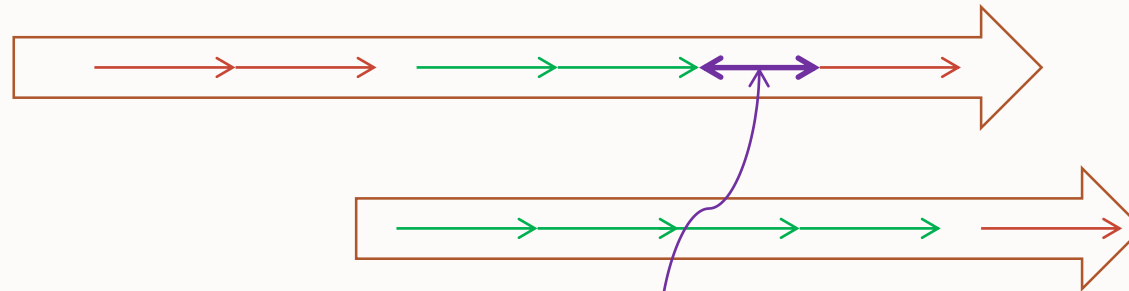


## Database Replayのリプレイオプションでできること(3/5)

接続

切断

→ select  
→ ins/upd/del



思考時間の調整 (think\_time\_scale)

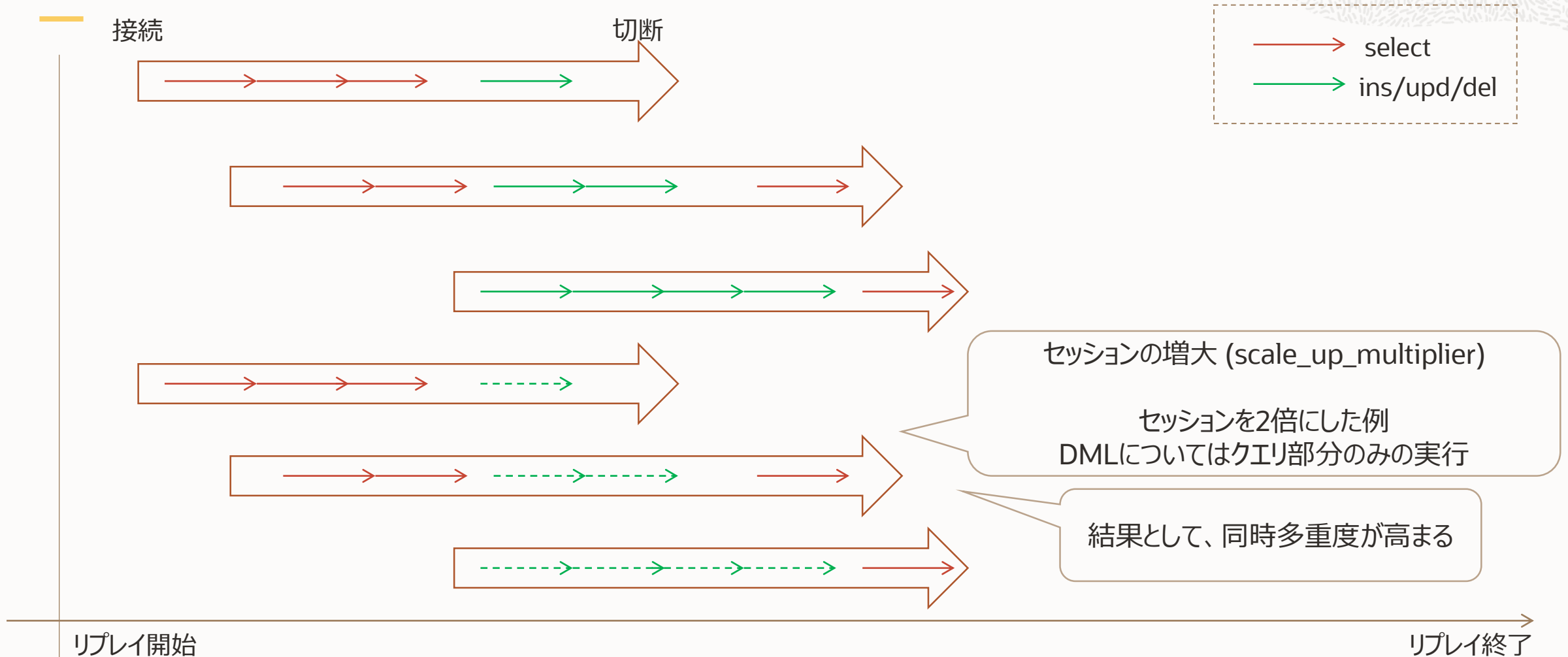
セッション内のコール間の思考時間を50%にした例

結果として、同時多重度が高まる

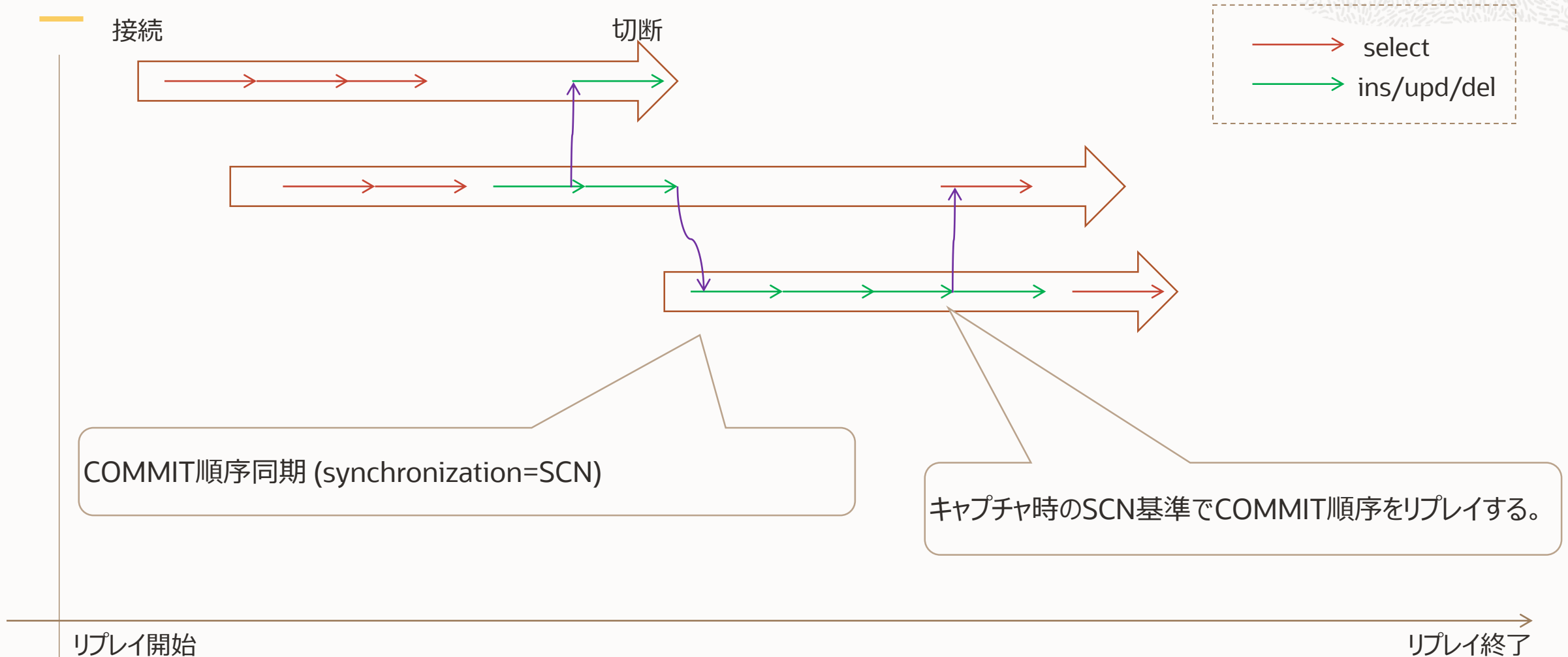
リプレイ開始

リプレイ終了

## Database Replayのリプレイオプションでできること(4/5)



# Database Replayのリプレイオプションでできること(5/5)



# (参考) Database Replayを自己学習していただくための記事の紹介

DBインスタンスとSQL実行環境でDatabase Replay機能を自己学習いただくため記事を公開しています。

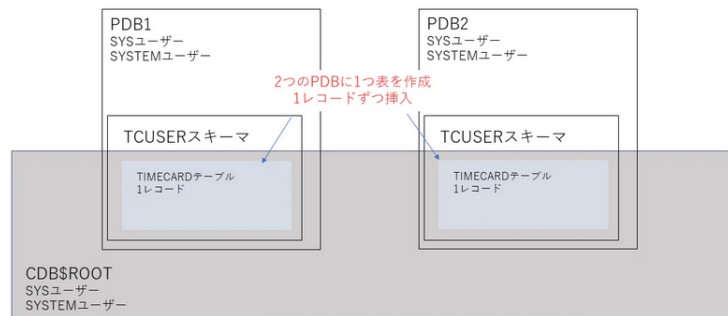
Oracle Real Application TestingのDatabase Replayの簡単な動作確認の手順

[https://qiita.com/ts\\_carp/items/da4d4a4488568231a63f](https://qiita.com/ts_carp/items/da4d4a4488568231a63f)

## 2. 本記事の作業の流れ

本記事で紹介しているDB Replayの動作確認の手順の流れを図で説明します。流れをおおよそ頭に入れておくと、実際の作業の理解がしやすいと思います。まずは、DB Replayの最初のステップであるキャプチャの前の準備です。

キャプチャ前の準備



### 概要

1. 本記事の作業の前提
2. 本記事の作業の流れ
3. キャプチャ前の準備
4. キャプチャの実施
5. (オプション)キャプチャ後の作業
6. リプレイ前の準備
7. ワークロードの前処理
8. (オプション)ワークロードの前処理後の作業
9. リプレイの実行
10. (オプション)リプレイ後の作業
11. 繰り返し実行する場合の準備
12. 参考

## (参考) Database Replayマニュアル

### Database Replay

#### **Oracle Database Testingガイド 19c**

- 9 データベース・リプレイの概要 ～ 13 取得およびリプレイ済ワークロードの分析

# Appendix



# 事例情報

## Oracle Cloud(PaaS/IaaS)：お客様活用事例

<https://blogs.oracle.com/oracle4engineer/oracle-cloud-platformpaasiaas>



Oracle Cloud PaaS お客様活用事例

検索

### 開発・検証の効率化（データベース・アップグレード）

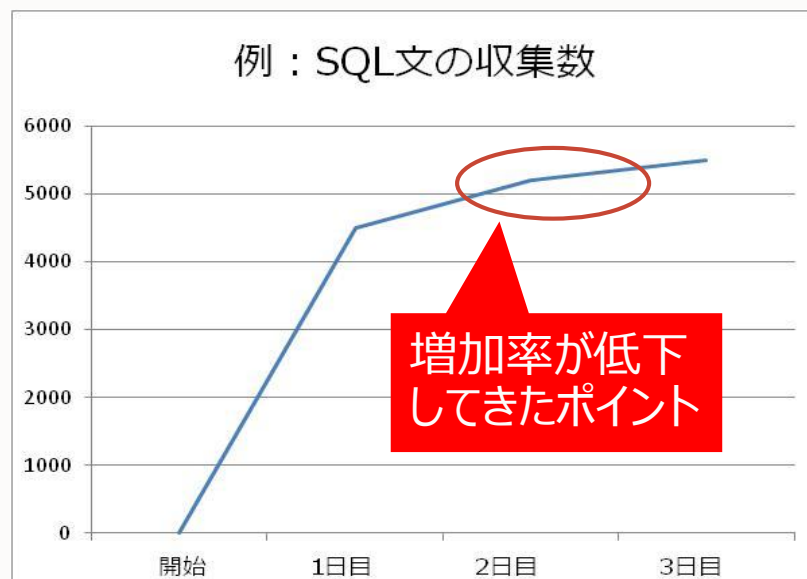
| 社名                  | 内容                                                                                                                                                                                                                                                                             | サービス                                |
|---------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-------------------------------------|
| 外為どっとコム             | <ul style="list-style-type: none"><li>・オラクルのクラウドを活用し、FXシステムの安定稼働による高品質サービスを実現</li><li>・2100本超のSQL全量テストの工数を約7割削減：これが「RAT in Cloud」の威力！ 外為どっとコムがデータベースアップグレードの期間とコストを大幅圧縮 - オラクルデータベースインサイダー</li></ul>                                                                           | Database + Real Application Testing |
| カブドットコム証券           | Oracle Cloudを導入し、アップデートのテストを自動化、工数を75%削減                                                                                                                                                                                                                                       | Database + Real Application Testing |
| 関西電力                | <ul style="list-style-type: none"><li>・テストのコストと時間を大幅に縮減</li><li>・Oracle Cloud+Oracle RATで約23,000のSQLを漏れなく事前検証、本番稼働後の性能障害を阻止</li></ul>                                                                                                                                          | Database + Real Application Testing |
| ライトウェル（住友重機械工業グループ） | <ul style="list-style-type: none"><li>・「Oracle Database Cloud Service」と「Oracle Real Application Testing」を活用し、数十分でデータベース・アップグレードの環境準備が可能に</li><li>・Oracle RATによる新旧DBの性能検証をクラウドでも：DBアップグレードのベストプラクティスはOracle Database Cloud Serviceでも使える！ ライトウェルが検証 - オラクルデータベースインサイダー</li></ul> | Database + Real Application Testing |

# STS使用のTIPS

## STS使用のTIPS --SQL収集期間

カーソルキャッシュからSQLをキャプチャする場合、どのタイミングでキャプチャーを終えれば良いかわからない場合があります

その場合はSTSに収集したSQL数を定期的を確認し、**期間あたりの増加率が低下してきたポイントで、収集が終わったと判断**することができます



### ※リテラル値を使用したSQLについて

同じ意味のSQLであっても、異なるSQLと見なされるため、格納されるSQLが増加し続けます。この場合は新規で収集されるSQLを確認し、収集済みのSQLと同類のSQLであれば収集完了と判断します。

### ※収集期間に流れないSQLについて

必要に応じて、本番機での収集が完了後、テスト環境にて、収集できなかった月次/年次バッチ等のSQLを流してSTSへ格納します。



## 参考：本番環境（取得環境）、テスト環境のデータベースに必要なパッチの情報

### Mandatory Patches for Database Testing Functionality for Current and Earlier Releases (ドキュメントID 560977.1\*)

データベースのリリースの組み合わせに応じた個別パッチの情報

- Table 1: **Database Replay** Availability Information for All Platforms except Windows
- Table 2: **Database Replay** Availability Information Windows Platform
- Table 3: **SQL Performance Analyzer** Availability Information

\* Oracle® Database Testingガイド 19c「9 データベース・リプレイの概要」参照

# RATがインストール済みか確認する方法

- RATがインストール済みが否かは下記SQLで確認できます

```
SQL> select * from v$option where parameter='Real Application Testing';
```

```
PARAMETER
```

```
-----  
VALUE  
-----  
CON_ID  
-----  
Real Application Testing  
TRUE  
0
```

← インストールされている場合 TRUE になる。

- RATを追加でインストールするはchoptツールを追加して追加でインストールできます。  
choptツールを実行する前にデータベースを停止し、データベース・オプションを追加または削除した後でデータベースを起動する必要があります。

(参考) マニュアルOracle Database Databaseインストレーション・ガイド 19c for Linux の  
「インストール後のOracle Databaseオプションの有効化および無効化 choptツール」

[https://docs.oracle.com/cd/F19136\\_01/ladbi/chopt-tool.html#GUID-057E4EFC-74ED-43B3-B03B-C83C5A5D3C7F](https://docs.oracle.com/cd/F19136_01/ladbi/chopt-tool.html#GUID-057E4EFC-74ED-43B3-B03B-C83C5A5D3C7F)

# Oracle Database 19c Database Replay機能の拡張

## PDB単位のワークロードキャプチャとリプレイ

### メリット

- PDB管理者による、ワークロードのキャプチャとリプレイが可能

### 機能概要

- 18c までの Real Application Testing, Database Replay
  - マルチテナント環境においてCDB(CDB root)に接続してのワークロードのキャプチャ、リプレイが可能
  - 特定のPDBに接続して、ワークロードのキャプチャ、リプレイはできない
  - (CDB管理者でない) 特定のPDB管理者は、管理しているPDBのワークロードのキャプチャ、リプレイを実施できない
- 19c からの Real Application Testing, Database Replay
  - 特定のPDBに接続して、ワークロードのキャプチャ、リプレイが可能
  - 接続しているPDBのワークロードのキャプチャを実施。同様に、接続しているPDBでワークロードのリプレイを実施。
  - 特定のPDB管理者が、管理しているPDBのワークロードのキャプチャ、リプレイを実施可能
  - ただしPDBレベルのキャプチャ、リプレイを同時に複数のPDBで実行することはできない



ありがとうございました

---

