

Enterprise Managerで実演！SQLパフォーマンスの高速化の限界を目指そう

Oracle Database Technology Night Lite 2019/09

日本オラクル株式会社
データベースソリューション部
丸山 毅
2019/09/24

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.

Oracle Database Technology Night Liteとは？ □

- 『Oracle Database Technology Night **Lite**』では、従来の「Oracle Database Technology Night」よりも基礎的でライトなテーマを取り上げ、基礎的でライトな解説/ディスカッションを行います。□

本日の内容について□

- 「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>
 - [第53回 SQLパフォーマンスの高速化の限界を目指せ！\(1\) 2017.6.27公開□](#)
 - [第54回 SQLパフォーマンスの高速化の限界を目指せ！\(2\) 2018.2.13公開□](#)
 - [第55回 SQLパフォーマンスの高速化の限界を目指せ！\(3\) 2018.5.25公開□](#)
- [Oracle Code Tokyo 2017 2017年05月18日 □Live Challenge!! SQLパフォーマンスの高速化の限界を目指せ！□](#)
 - <https://logmi.jp/business/articles/231003>
 - <https://logmi.jp/business/articles/231089>
 - <https://youtu.be/o33RSyOuHYM>
- [2017/8/18\(金\) □Oracle Database Technology Night 夏祭り□実演！SQLパフォーマンスの高速化の限界を目指せ□](#)



Scenario: 設定シナリオ□



分析ツール等で自動生成されるSQL文は□
複雑であるために、パフォーマンスに問題□
があってもその構文自体をチューニングす□
ることは非常に難しい傾向があります。□

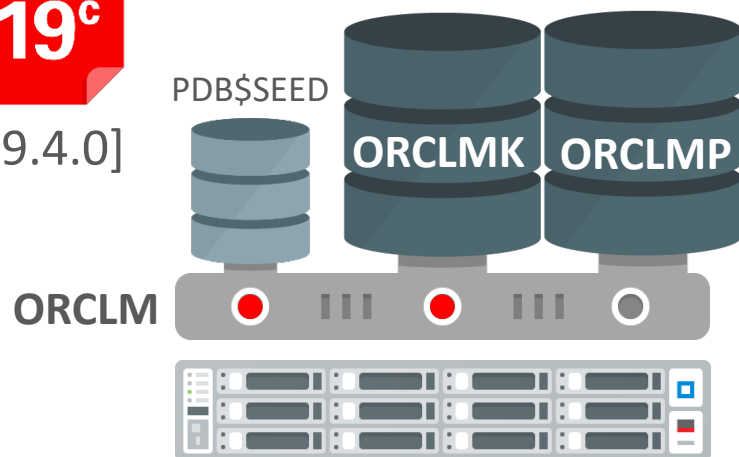
本セッションでは、最新Oracle Database 19c
の機能を有効活用することで、□
どこまでSQL処理が高速化していくのか?
をライブで限界にチャレンジします。□

注意: □
本スライドで掲載されている各種値は、□
実行環境やワークロード等の状況によって異な□
ります。各製品機能の効果を保証するものではありません。□

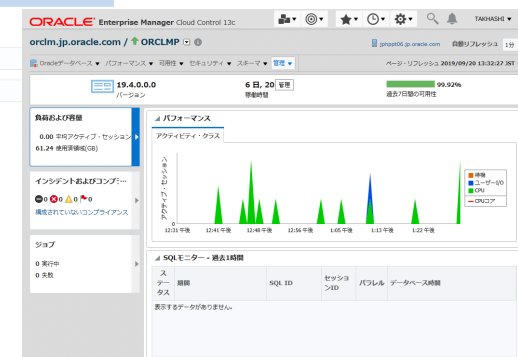
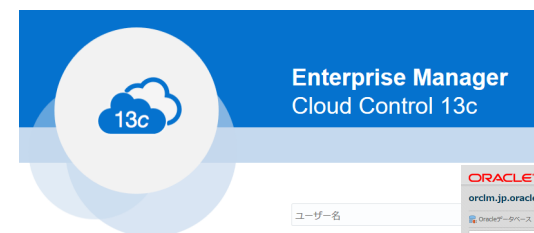
環境□

- 実行環境□

19^c
[19.4.0]



Sun Server X3-2L :
jphppt06.jp.oracle.com
Oracle Linux 7.6
6core x2socket=total 12 core 64GB RAM
600GB HDD x 12



Enterprise Manager
Cloud Control 13c
[13.3]
Sun Server X3-2L
ptvm52.jp.oracle.com



pdb の作成□

- Oracle Database 19c Enterprise Edition インストール済み、CDB作成済み□
の環境□
- 環境変数を設定□

```
[oracle@jphppt06 ~]$ env | grep ORA
ORACLE_SID=orclm
ORACLE_BASE=/u01/app/oracle
ORACLE_HOME=/u01/app/oracle/product/19c/dbhome_1
[oracle@jphppt06 ~]$ env | grep PATH
PATH=/u01/app/oracle/product/19c/dbhome_1/bin:/usr/lib64/qt-
3.3/bin:/usr/local/bin:/usr/bin:/usr/local/sbin:/usr/sbin:/home/oracl
e/.local/bin:/home/oracle/bin
```

pdb の作成□

- CDBへ接続□

```
[oracle@jphppt06 ~]$ which sqlplus  
/u01/app/oracle/product/19c/dbhome_1/bin/sqlplus  
[oracle@jphppt06 ~]$ sqlplus / as sysdba
```

```
SQL*Plus: Release 19.0.0.0.0 - Production on Tue Sep 10 16:16:41 2019  
Version 19.4.0.0.0
```

```
Copyright (c) 1982, 2019, Oracle. All rights reserved.
```

```
Connected to:
```

```
Oracle Database 19c Enterprise Edition Release 19.0.0.0.0 -  
Production  
Version 19.4.0.0.0
```

```
SQL>
```

pdb の作成□

- バージョンの確認□

```
17:21:05 SQL>
16:11:31 SQL> select * from v$version;
Oracle Database 19c Enterprise Edition Release 19.0.0.0.0 - Production
Oracle Database 19c Enterprise Edition Release 19.0.0.0.0 - Production
Version 19.4.0.0.0
Oracle Database 19c Enterprise Edition Release 19.0.0.0.0 - Production
0
```

```
Elapsed: 00:00:00.04
```

```
17:35:31 SQL>
```

pdb の作成□

- CDB名の確認□

```
SQL> show con_name;
```

```
CON_NAME
```

```
-----  
CDB$ROOT
```

```
SQL>
```

```
SQL> select instance_name, status from v$instance;
```

```
INSTANCE_NAME      STATUS
```

```
-----  
orclm              OPEN
```

```
SQL>
```

PDB の作成□

- 既存のPDBの確認□

```
SQL> show pdbs
```

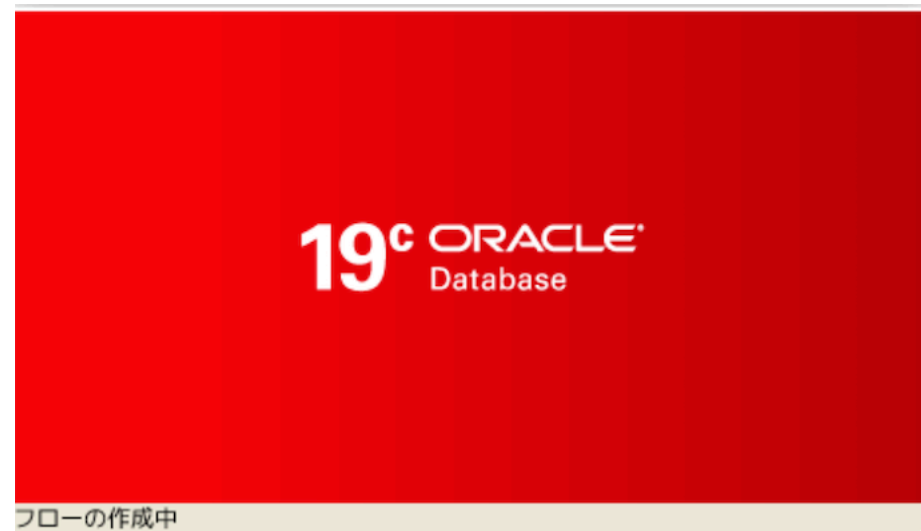
CON_ID	CON_NAME	OPEN	MODE	RESTRICTED
2	PDB\$SEED	READ	ONLY	NO
3	ORCLMP	READ	WRITE	NO

```
SQL>
```

pdb の作成□

- dbca で PDB “orclmq” を追加□

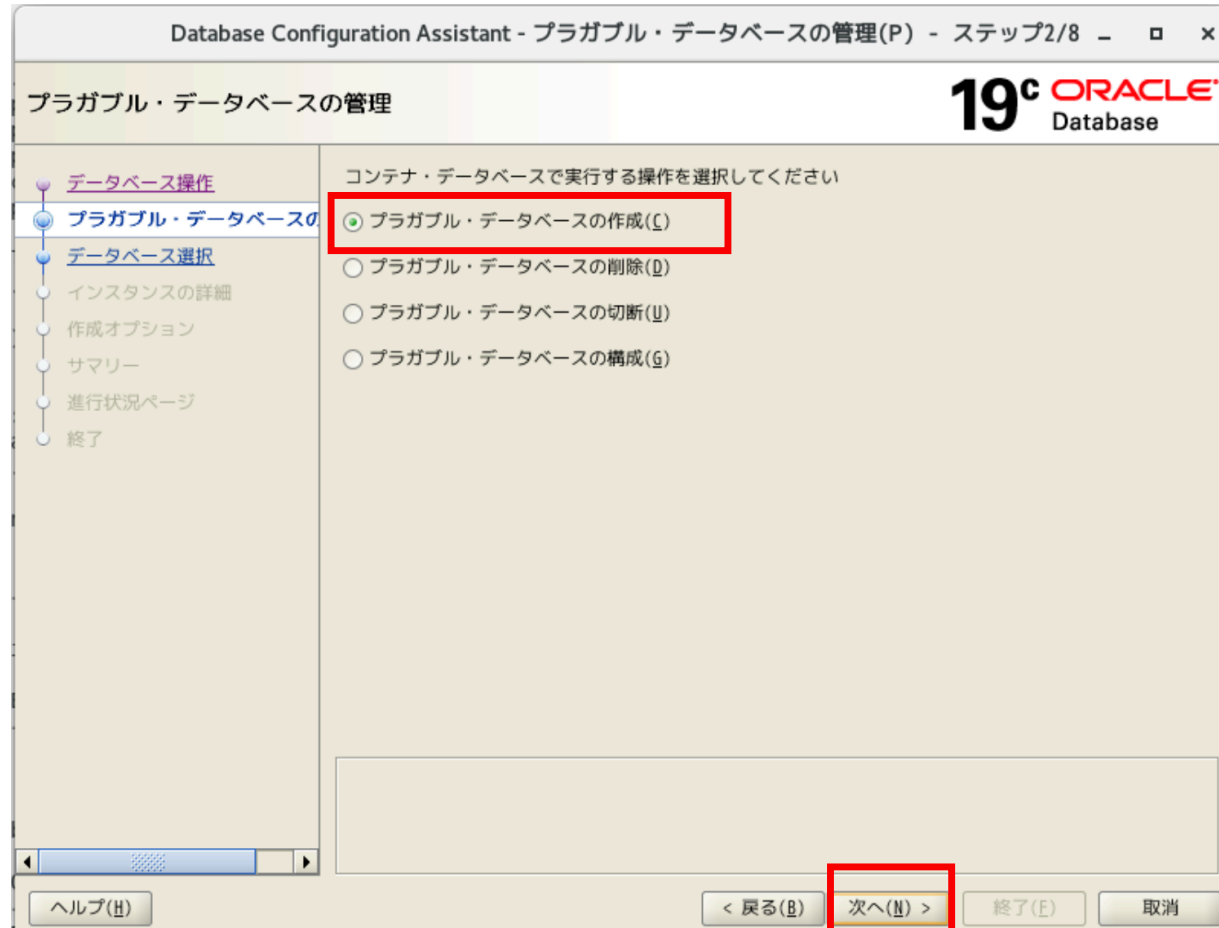
```
[oracle@jphppt06 ~]$ which dbca  
/u01/app/oracle/product/19c/dbhome_1/bin/dbca  
[oracle@jphppt06 ~]$ dbca
```



プラガブル・データベースの管理(P)



プラガブル・データベースの作成



プラグブル・データベースを作成するコンテナデータベースを選択□

Database Configuration Assistant - プラグブル・データベースの管理(P) - ステップ3/8

ソース・データベースの選択

19c ORACLE Database

データベース操作
プラグブル・データベースの管理
データベース選択
インスタンスの詳細
作成オプション
サマリー
進行状況ページ
終了

プラグブル・データベースを作成する必要があるコンテナ・データベースを選択します。

データベース	ローカル・インスタ...	タイプ
☐ orcl19c	orcl19c	Single Instance
☒ orclm	orclm	Single Instance

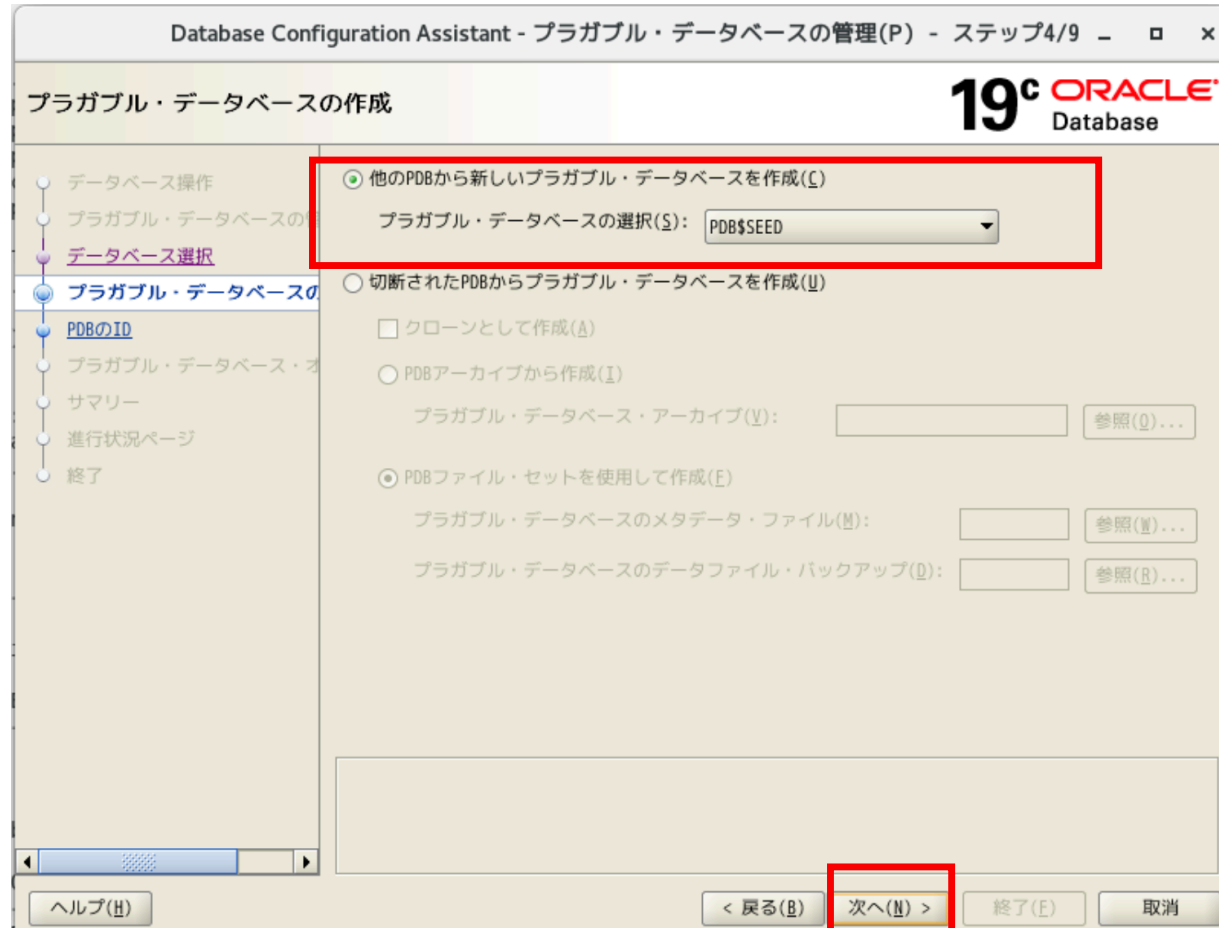
DBCAは、OSベース認証を使用してデータベースに接続します。OSベース認証が無効な場合、データベース資格証明が必要になることがあります。必要な場合、資格証明を指定してください。

ユーザー名(U):

パスワード(P):

ヘルプ(H) < 戻る(B) 次へ(N) > 終了(E) 取消

他のPDBから新しいプラガブル・データベースを作成□



PDBのID

Database Configuration Assistant - プラガブル・データベースの管理(P) - ステップ5/9

プラガブル・データベースIDのオプション

19c ORACLE Database

データベース操作
プラガブル・データベースの管理
データベース選択
プラガブル・データベースの作成
PDBのID
プラガブル・データベース・オプション
サマリー
進行状況ページ
終了

プラガブル・データベース名(P): orclmk

☒ 新規管理者の作成(E)

管理者のユーザー名(A): admin

管理パスワード(M):

管理者パスワードの確認(L):

☐ 既存のすべてのPDBユーザーのロック(L)

ヘルプ(H) < 戻る(B) **次へ(N) >** 終了(E) 取消

プラガブル・データベースオプション

Database Configuration Assistant - プラガブル・データベースの管理(P) - ステップ6/9

プラガブル・データベース・オプション **19c ORACLE Database**

データベース操作
プラガブル・データベースの管理
データベース選択
プラガブル・データベースの作成
PDBのID
プラガブル・データベース・オプション
サマリー
進行状況ページ
終了

PDB記憶域オプション(O)

選択したコンテナ・データベースはOMFにあります。PDBデータファイルは次の場所に配置されます。

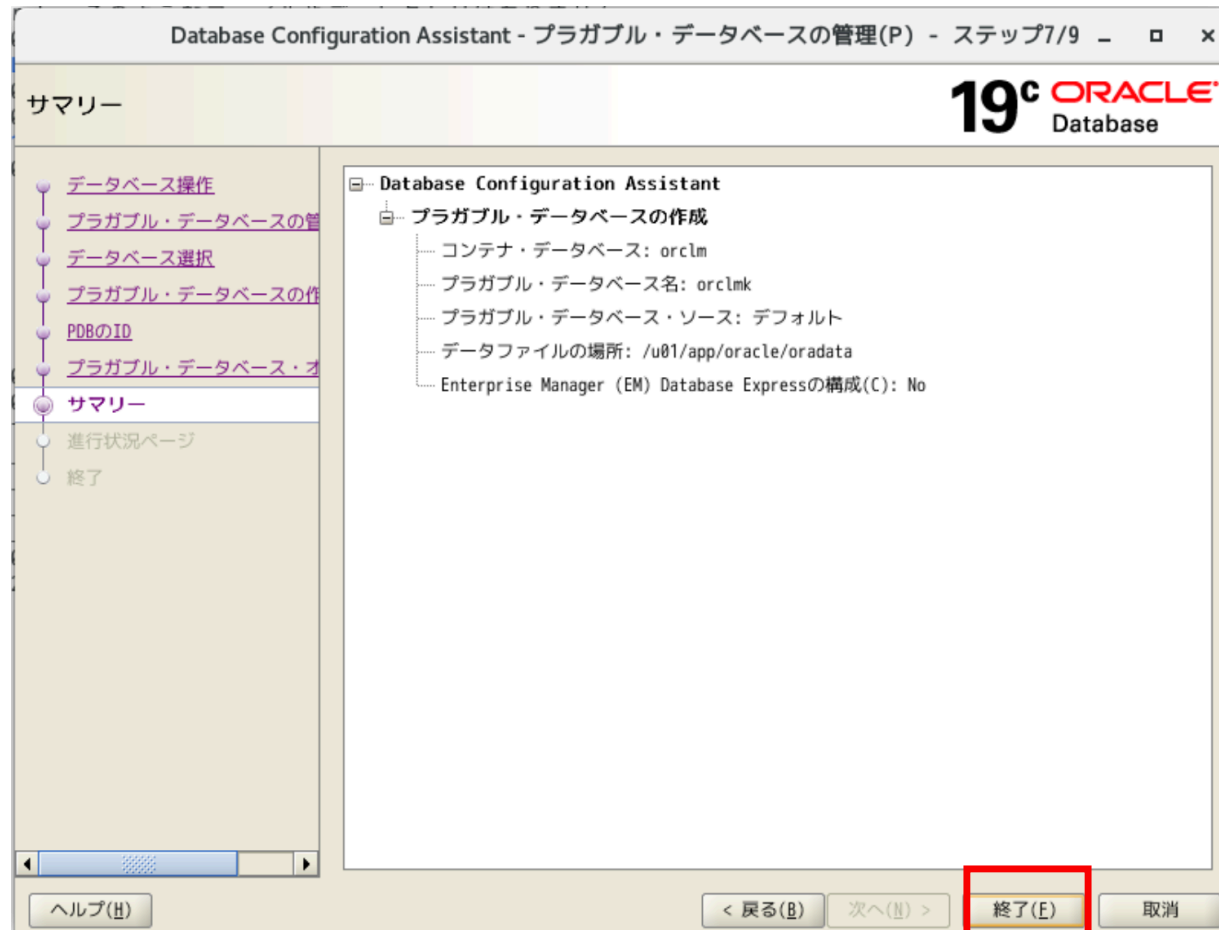
記憶域タイプ(S): ファイルシステム

データベースの場所(L): {ORACLE_BASE}/oradata/{DB_UNIQUE_NAME} 参照(R)...

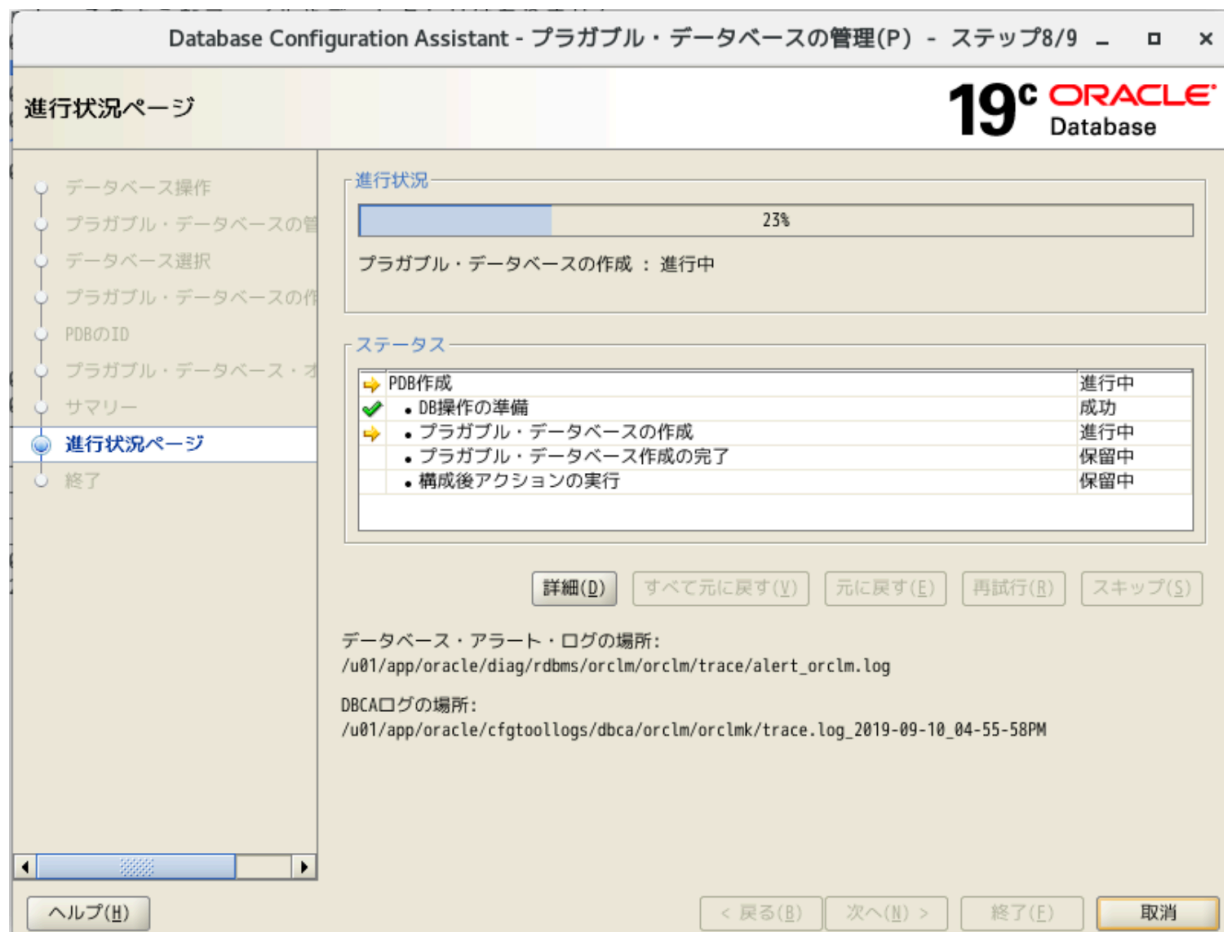
☒ デフォルト・ユーザー表領域の作成(I)

ヘルプ(H) < 戻る(B) **次へ(N) >** 終了(E) 取消

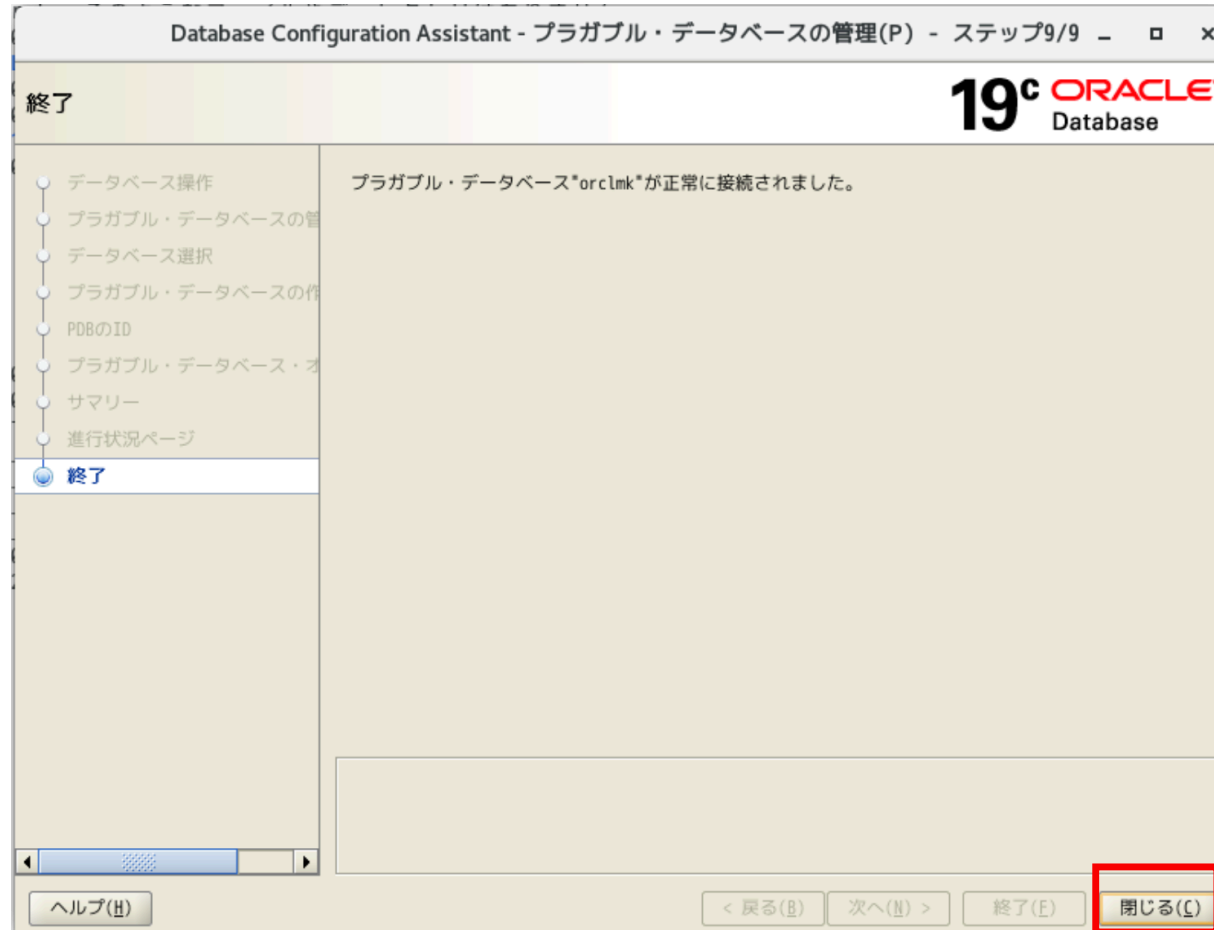
サマリー□



進行中□



作成完了(3分くらい?) □



作成されたPDBを確認□

```
SQL> show pdbs
```

CON_ID	CON_NAME	OPEN	MODE	RESTRICTED
2	PDB\$SEED	READ ONLY	NO	
3	ORCLMP	READ WRITE	NO	
4	ORCLMK	READ WRITE	NO	

```
SQL> alter session set container=orclmk;  
Session altered.
```

```
SQL> show con_name;
```

```
CON_NAME
```

```
ORCLMK
```

PDB接続のためのサービス名 (SERVICE NAME)の確認□

```
[oracle@jphppt06 ~]$ lsnrctl status
```

中略

```
Service "orclmk.jp.oracle.com" has 1 instance(s).
```

```
Instance "orclm", status READY, has 1 handler(s) for this service...
```

接続記述子(connect descriptor)の記載□

```
[oracle@jphppt06 admin]$ pwd
/u01/app/oracle/product/19c/dbhome_1/network/admin
```

以下を追記

```
[oracle@jphppt06 admin]$ cat tnsnames.ora
```

中略

```
ORCLMK =
  (DESCRIPTION =
    (ADDRESS = (PROTOCOL = TCP) (HOST = jphppt06.jp.oracle.com) (PORT = 1521))
    (CONNECT_DATA =
      (SERVER = DEDICATED)
      (SERVICE_NAME = orclmk.jp.oracle.com)
    )
  )
```

確認□

```
[oracle@jphppt06 admin]$ tnsping orclmk
```

```
TNS Ping Utility for Linux: Version 19.0.0.0.0 - Production on 11-SEP-2019  
16:24:11
```

```
Copyright (c) 1997, 2019, Oracle. All rights reserved.
```

```
Used parameter files:
```

```
/u01/app/oracle/product/19c/dbhome_1/network/admin/sqlnet.ora
```

```
Used TNSNAMES adapter to resolve the alias
```

```
Attempting to contact (DESCRIPTION = (ADDRESS = (PROTOCOL = TCP) (HOST =  
jphppt06.jp.oracle.com) (PORT = 1521)) (CONNECT_DATA = (SERVER = DEDICATED)  
(SERVICE_NAME = orclmk.jp.oracle.com)))
```

```
OK (10 msec)
```

```
[oracle@jphppt06 admin]$
```

PDB “orclmk” 上で□ データベースユーザー sh 作成、権限付与□

```
[oracle@jphppt06 ~]$ sqlplus / as sysdba
SQL*Plus: Release 19.0.0.0.0 - Production on Tue Sep 10 17:45:32 2019
Version 19.4.0.0.0
Copyright (c) 1982, 2019, Oracle. All rights reserved.
Connected to:
Oracle Database 19c Enterprise Edition Release 19.0.0.0.0 - Production
Version 19.4.0.0.0

SQL> alter session set container=orclmk;
Session altered.

SQL> create user sh identified by welcome1;
User created.

SQL> grant connect, resource to sh;
Grant succeeded.
```

作成したデータベースユーザーでPDB “orclmk”へ接続□

```
[oracle@jphppt06 ~]$ sqlplus /nolog
```

```
SQL*Plus: Release 19.0.0.0.0 - Production on Tue Sep 10 17:49:22 2019  
Version 19.4.0.0.0
```

```
Copyright (c) 1982, 2019, Oracle. All rights reserved.
```

```
SQL> connect sh/welcome1@orclmk;  
Connected.
```

```
SQL> show con_name;
```

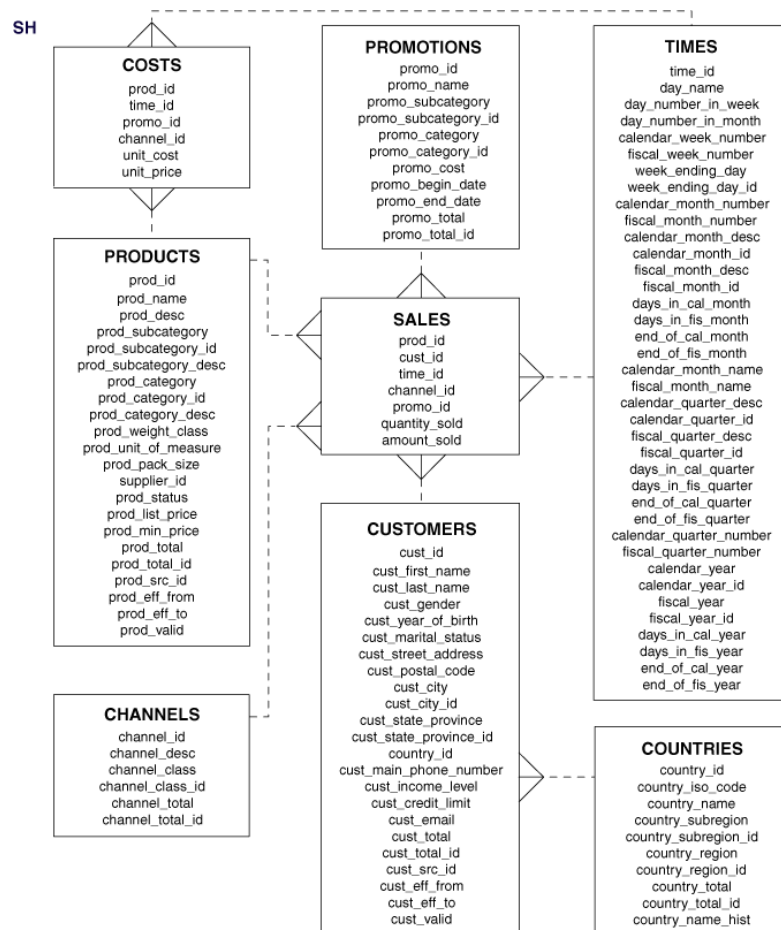
```
CON_NAME
```

```
-----
```

```
ORCLMK
```

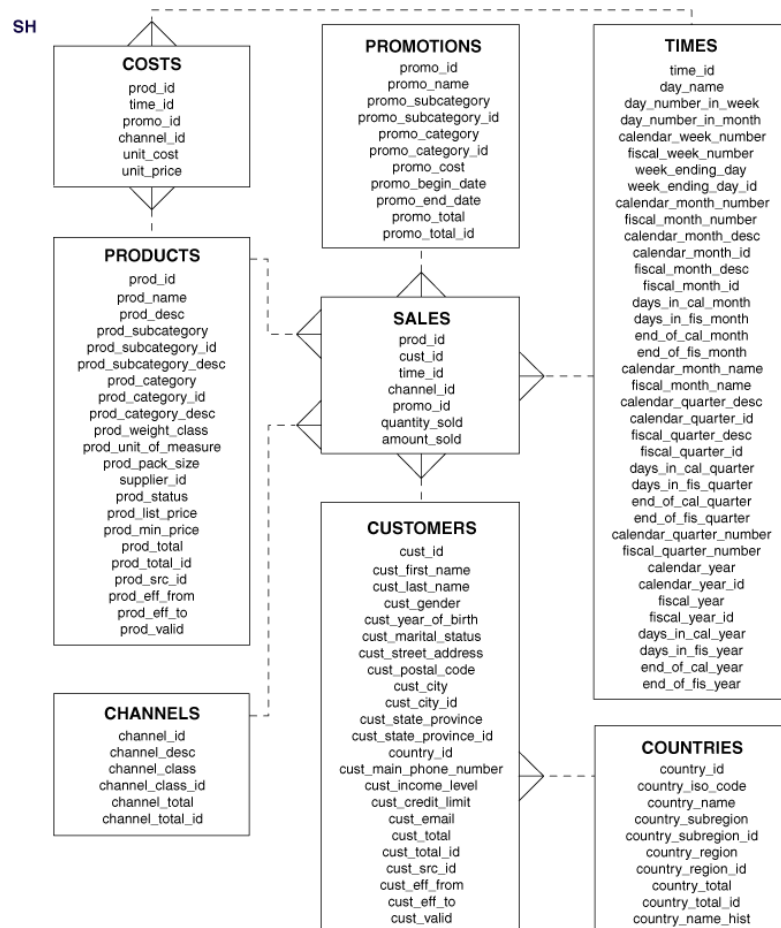
```
SQL>
```

Oracle Database Sample Schemas – SH Schema



- サンプル・スキーマのSHがベース□
 - [Oracle Database Release 19 Database サンプル・スキーマ](https://docs.oracle.com/cd/F19136_01/comsc/index.html)□
https://docs.oracle.com/cd/F19136_01/comsc/index.html
- 1. GitHubからダウンロード□
 - <https://github.com/oracle/db-sample-schemas/releases/latest>
 - 展開後、スクリプト内のインストールディレクトリ名を変□更□
- 2. サンプルスキーマ用表領域作成□40GB
- 3. サンプルスキーマのインストール□

Oracle Database Sample Schemas – SH Schema



- 4. SALES表のデータ量を増幅□
 - 元のSALES表を別名 (SALES_ORG) に変更□
 - ダミー列を2つ追加した新たなSALES表 (非パーティション表) □を作成□
 - 元のSALES表 (SALES_ORG) から新規作成したSALES表へレコードをコピー□
 - INSERT select 文を繰り返し実行し、データ容量 (レコード数) □を256MBから約32GBまで増幅□
- 5. SALES表の日付データに対応するように、TIMES表のカスタマイズを実施□
 - カスタマイズしたTIMES表に主キーを作成□
- 6. SHスキーマのオプティマイザ統計情報を収集□
- 7. FILESYSTEMIO_OPTIONS初期化パラメータを「SETALL」□に設定して、ファイルシステムに対する非同期I/OとダイレクトI/Oを有効化□

1. GitHub サンプルスキーマダウンロード

<https://github.com/oracle/db-sample-schemas/releases/latest>

The screenshot shows the GitHub repository page for 'oracle/db-sample-schemas'. The repository has 86 Watchers, 262 Stars, and 239 Forks. The 'Releases' tab is selected, showing the latest release 'v19c' (commit 004d5f3, Verified). The release description states: 'Use these scripts to create the Oracle Database 19c Sample Schemas referenced in the [documentation](#) and examples. The scripts install in Oracle Database 12c and upwards, including Oracle Database Cloud Services.' Under the 'Assets' section, there are two assets: 'Source code (zip)' and 'Source code (tar.gz)'. The 'Source code (zip)' asset is highlighted with a red rectangular box.

ダウンロードしたスクリプトをunzip

```
[oracle@jphppt06 orclmk]$ pwd
/home/oracle/technight/orclmk
[oracle@jphppt06 orclmk]$ unzip db-sample-schemas-19c.zip
Archive:  db-sample-schemas-19c.zip
004d5f3c99dd2e19dee8b72360b55afce458ea32
  creating: db-sample-schemas-19c/
  inflating: db-sample-schemas-19c/CONTRIBUTING.md
```

中略

```
[oracle@jphppt06 orclmk]$
[oracle@jphppt06 orclmk]$ cd db-sample-schemas-19c/
[oracle@jphppt06 db-sample-schemas-19c]$ ls
CONTRIBUTING.md  README.txt      human_resources  mkplug.sql      mkverify.sql    sales_history
LICENSE.md       bus_intelligence  info_exchange    mksample.sql    order_entry     shipping
README.md        drop_sch.sql     mk_dir.sql       mkunplug.sql    product_media
[oracle@jphppt06 db-sample-schemas-19c]$
```

スクリプト内の文字列を展開したディレクトリ名に変更□ (README.txt参照)□

```
[oracle@jphppt06 db-sample-schemas-19c]$ perl -p -i.bak -e  
's#__SUB__CWD__#'$ (pwd) '#g' *.sql */*.sql */*.dat  
[oracle@jphppt06 db-sample-schemas-19c]$ ls -lart  
total 184 中略  
drwxr-xr-x 3 oracle oinstall      68 Sep 10 17:52 ..  
-rw-r--r-- 1 oracle oinstall  2857 Sep 10 18:12 mk_dir.sql  
-rw-r--r-- 1 oracle oinstall  3633 Sep 10 18:12 drop_sch.sql  
-rw-r--r-- 1 oracle oinstall  6592 Sep 10 18:12 mkunplug.sql  
-rw-r--r-- 1 oracle oinstall  7595 Sep 10 18:12 mksample.sql  
-rw-r--r-- 1 oracle oinstall 28068 Sep 10 18:12 mkplug.sql  
drwxr-xr-x 9 oracle oinstall  4096 Sep 10 18:12 .  
-rw-r--r-- 1 oracle oinstall  6123 Sep 10 18:12 mkverify.sql  
drwxr-xr-x 2 oracle oinstall  4096 Sep 10 18:12 bus_intelligence  
drwxr-xr-x 2 oracle oinstall  4096 Sep 10 18:12 human_resources  
drwxr-xr-x 2 oracle oinstall  4096 Sep 10 18:12 info_exchange  
drwxr-xr-x 2 oracle oinstall  4096 Sep 10 18:12 shipping  
drwxr-xr-x 3 oracle oinstall  8192 Sep 10 18:12 order_entry  
drwxr-xr-x 2 oracle oinstall  4096 Sep 10 18:12 product_media  
drwxr-xr-x 2 oracle oinstall  4096 Sep 10 18:12 sales_history
```

2. sysdbaユーザーで表領域の作成(20分くらい) □

```
[oracle@jphppt06 db-sample-schemas-19c]$ sqlplus / as sysdba

SQL*Plus: Release 19.0.0.0.0 - Production on Tue Sep 10 18:18:19 2019
Version 19.4.0.0.0

Copyright (c) 1982, 2019, Oracle. All rights reserved.

Connected to:
Oracle Database 19c Enterprise Edition Release 19.0.0.0.0 - Production
Version 19.4.0.0.0

SQL> alter session set container=orclmk;

Session altered.

11:18:18 SQL> create bigfile tablespace TBS4 datafile
'/u01/app/oracle/oradata/ORCLM/datafile/datatbs04.dbf' size 40g ;

Tablespace created.
11:38:11 SQL>
```

3. サンプルスキーマの作成(引数はREADME.txt 参照) □

```
[oracle@jphppt06 db-sample-schemas-19c]$ sqlplus /nolog
SQL*Plus: Release 19.0.0.0.0 - Production on Wed Sep 11 18:29:23 2019
Version 19.4.0.0.0
Copyright (c) 1982, 2019, Oracle. All rights reserved.

SQL> set time on
18:29:26 SQL>
18:29:26 SQL>
18:29:27 SQL>
18:29:27 SQL> @mksample.sql Welcome$ Welcome$ welcome1 welcome1 welcome1
welcome1 welcome1 welcome1 TBS4 TEMP /home/oracle/technight/orclmk/db-sample-
schemas-19c/log/ orclmk
specify password for SYSTEM as parameter 1:
specify password for SYS as parameter 2:
specify password for HR as parameter 3:
specify password for OE as parameter 4:
specify password for PM as parameter 5:
```

サンプルスキーマの作成(引数はREADME.txt 参照) □ 約2分 □

```
specify password for IX as parameter 6:  
specify password for SH as parameter 7:  
specify password for BI as parameter 8:  
specify default tablespace as parameter 9:  
specify temporary tablespace as parameter 10:  
specify log file directory (including trailing delimiter) as parameter 11:  
specify connect string as parameter 12:  
Sample Schemas are being created ...  
中略
```

```
72 rows selected.
```

```
18:31:45 SQL>
```

```
/home/oracle/technight/orclmk/db-sample-schemas-19c/log/  
内のログファイルを確認して実行状況を確認
```

4. SALES表のデータ増幅□元のSALES表を別名に変更□ ダミー列を2つ追加した新たなSALES表(非パーティション表)を作成□

```
18:35:14 SQL> connect sh/welcome1@orclmk
```

```
Connected.
```

```
18:36:19 SQL> rename SALES to SALES_ORG;
```

```
Table renamed.
```

```
18:36:48 SQL> CREATE TABLE "SH"."SALES"
```

```
(  "PROD_ID" NUMBER NOT NULL ENABLE,  
  "CUST_ID" NUMBER NOT NULL ENABLE,  
  "TIME_ID" DATE NOT NULL ENABLE,  
  "CHANNEL_ID" NUMBER NOT NULL ENABLE,  
  "PROMO_ID" NUMBER NOT NULL ENABLE,  
  "QUANTITY_SOLD" NUMBER(10,2) NOT NULL ENABLE,  
  "AMOUNT_SOLD" NUMBER(10,2) NOT NULL ENABLE,  
  "DUMMY1" CHAR(100),  
  "DUMMY2" CHAR(110)  
);
```

```
18:37:03  2  18:37:03  3  18:37:03  4  18:37:03  5  18:37:03  6  18:37:03  7  18:37:03  8
```

```
18:37:03  9  18:37:03 10  18:37:03 11
```

```
Table created.
```

```
18:37:03 SQL> 18:37:03 SQL>
```

```
18:37:05 SQL>
```


元のSALES表(SALES_ORG)から新規作成したSALES表へレコードをコピー

```
18:37:52 SQL> insert /*+append */ into SH.SALES
```

```
nologging
```

```
  select PROD_ID,  
         CUST_ID,  
         TIME_ID+15*365+3,  
         CHANNEL_ID,  
         PROMO_ID,  
         QUANTITY_SOLD,  
         AMOUNT_SOLD,  
         rpad(to_char(mod(CUST_ID,30)),100,'dummy1'),  
         rpad(to_char(mod(CUST_ID,30)),110,'dummy2')  
  from SH.SALES_ORG;
```

```
commit;
```

```
18:37:54      2  18:37:54      3  18:37:54      4  18:37:54      5  18:37:54      6  18:37:54      7  
18:37:54      8  18:37:54      9  18:37:54     10  18:37:54     11  18:37:54     12
```

```
918843 rows created.
```

```
18:38:05 SQL>
```

```
Commit complete.
```

```
18:38:06 SQL> 18:38:06 SQL>
```

データ容量(レコード数)を256MBから約32GBまで増加(約□20分)□

```
18:38:41 SQL> alter session force parallel dml parallel 8 ;
alter session force parallel query parallel 8 ;
insert /*+append */ into SH.SALES nologging select * from SH.SALES;
commit;
insert /*+append */ into SH.SALES nologging select * from SH.SALES;
commit;
insert /*+append */ into SH.SALES nologging select * from SH.SALES;
commit;
insert /*+append */ into SH.SALES nologging select * from SH.SALES;
commit;
insert /*+append */ into SH.SALES nologging select * from SH.SALES;
commit;
insert /*+append */ into SH.SALES nologging select * from SH.SALES;
commit;
insert /*+append */ into SH.SALES nologging select * from SH.SALES;
commit;
Session altered.
18:39:30 SQL>
Session altered.
中略
18:47:51 SQL>
58805952 rows created.
18:57:20 SQL>
Commit complete.
```

増幅した結果確認□

```
10:38:26 SQL> col segment_name for a32
10:38:41 SQL> select SEGMENT_NAME, BYTES/1024/1024 from USER_SEGMENTS where
SEGMENT_NAME='SALES' ;
```

SEGMENT_NAME	BYTES/1024/1024
SALES	31739.125

```
1 row selected.
```

```
10:38:51 SQL>
```

5. SALES表の日付データに対応したTIMES表のカスタマイズ(1)

```
rename TIMES to TIMES_ORG;  
CREATE TABLE "SH"."TIMES"  
( "TIME_ID" DATE NOT NULL ,  
  "DAY_NAME" VARCHAR2(9) NOT NULL ,  
  "DAY_NUMBER_IN_WEEK" NUMBER(1) NOT NULL ,  
  "DAY_NUMBER_IN_MONTH" NUMBER(2) NOT NULL ,  
  "CALENDAR_WEEK_NUMBER" NUMBER(2) NOT NULL ,  
  "FISCAL_WEEK_NUMBER" NUMBER(2) NOT NULL ,  
  "WEEK_ENDING_DAY" DATE NOT NULL ,  
  "WEEK_ENDING_DAY_ID" NUMBER NOT NULL ,  
  "CALENDAR_MONTH_NUMBER" NUMBER(2) NOT NULL ,  
  "FISCAL_MONTH_NUMBER" NUMBER(2) NOT NULL ,  
  "CALENDAR_MONTH_DESC" VARCHAR2(8) NOT NULL ,  
  "CALENDAR_MONTH_ID" NUMBER NOT NULL ,  
  "FISCAL_MONTH_DESC" VARCHAR2(8) NOT NULL ,  
  "FISCAL_MONTH_ID" NUMBER NOT NULL ,  
  "DAYS_IN_CAL_MONTH" NUMBER NOT NULL ,  
  "DAYS_IN_FIS_MONTH" NUMBER NOT NULL ,  
  "END_OF_CAL_MONTH" DATE NOT NULL ,  
  "END_OF_FIS_MONTH" DATE NOT NULL ,  
  "CALENDAR_MONTH_NAME" VARCHAR2(9) NOT NULL ,  
  "FISCAL_MONTH_NAME" VARCHAR2(9) NOT NULL ,  
  "CALENDAR_QUARTER_DESC" CHAR(7) NOT NULL ,  
  "CALENDAR_QUARTER_ID" NUMBER NOT NULL ,  
  "FISCAL_QUARTER_DESC" CHAR(7) NOT NULL ,  
  "FISCAL_QUARTER_ID" NUMBER NOT NULL ,
```

```
"DAYS_IN_CAL_QUARTER" NUMBER NOT NULL ,  
  "DAYS_IN_FIS_QUARTER" NUMBER NOT NULL ,  
  "END_OF_CAL_QUARTER" DATE NOT NULL ,  
  "END_OF_FIS_QUARTER" DATE NOT NULL ,  
  "CALENDAR_QUARTER_NUMBER" NUMBER(1) NOT NULL ,  
  "FISCAL_QUARTER_NUMBER" NUMBER(1) NOT NULL ,  
  "CALENDAR_YEAR" NUMBER(4) NOT NULL ,  
  "CALENDAR_YEAR_ID" NUMBER NOT NULL ,  
  "FISCAL_YEAR" NUMBER(4) NOT NULL ,  
  "FISCAL_YEAR_ID" NUMBER NOT NULL ,  
  "DAYS_IN_CAL_YEAR" NUMBER NOT NULL ,  
  "DAYS_IN_FIS_YEAR" NUMBER NOT NULL ,  
  "END_OF_CAL_YEAR" DATE NOT NULL ,  
  "END_OF_FIS_YEAR" DATE NOT NULL )
```

続き

SALES表の日付データに対応したTIMES表のカスタマイズ(2)

```
PARTITION BY RANGE (TIME_ID)
(
partition p2013q1 values less than (to_date('2013/04/01','YYYY/MM/DD')),
partition p2013q2 values less than (to_date('2013/07/01','YYYY/MM/DD')),
partition p2013q3 values less than (to_date('2013/10/01','YYYY/MM/DD')),
partition p2013q4 values less than (to_date('2014/01/01','YYYY/MM/DD')),
partition p2014q1 values less than (to_date('2014/04/01','YYYY/MM/DD')),
partition p2014q2 values less than (to_date('2014/07/01','YYYY/MM/DD')),
partition p2014q3 values less than (to_date('2014/10/01','YYYY/MM/DD')),
partition p2014q4 values less than (to_date('2015/01/01','YYYY/MM/DD')),
partition p2015q1 values less than (to_date('2015/04/01','YYYY/MM/DD')),
partition p2015q2 values less than (to_date('2015/07/01','YYYY/MM/DD')),
partition p2015q3 values less than (to_date('2015/10/01','YYYY/MM/DD')),
partition p2015q4 values less than (to_date('2016/01/01','YYYY/MM/DD')),
partition p2016q1 values less than (to_date('2016/04/01','YYYY/MM/DD')),
partition p2016q2 values less than (to_date('2016/07/01','YYYY/MM/DD')),
partition p2016q3 values less than (to_date('2016/10/01','YYYY/MM/DD')),
partition p2016q4 values less than (to_date('2017/01/01','YYYY/MM/DD')),
partition p2017q1 values less than (to_date('2017/04/01','YYYY/MM/DD')),
partition p2017q2 values less than (to_date('2017/07/01','YYYY/MM/DD')),
partition p2017q3 values less than (to_date('2017/10/01','YYYY/MM/DD')),
partition p2017q4 values less than (to_date('2018/01/01','YYYY/MM/DD'))
)
TABLESPACE "TBS1" PCTFREE 10 INITRANS 1 MAXTRANS 255
LOGGING NOCOMPRESS
;
```

SALES表の日付データに対応したTIMES表のカスタマイズ(3)

```
insert /* +append */ into times
select
TIME_ID+15*365+4,
DAY_NAME,
DAY_NUMBER_IN_WEEK,
DAY_NUMBER_IN_MONTH,
CALENDAR_WEEK_NUMBER,
FISCAL_WEEK_NUMBER,
WEEK_ENDING_DAY,
WEEK_ENDING_DAY_ID,
CALENDAR_MONTH_NUMBER,
FISCAL_MONTH_NUMBER,
CALENDAR_MONTH_DESC,
CALENDAR_MONTH_ID,
FISCAL_MONTH_DESC,
FISCAL_MONTH_ID,
DAYS_IN_CAL_MONTH,
DAYS_IN_FIS_MONTH,
END_OF_CAL_MONTH,
END_OF_FIS_MONTH,
CALENDAR_MONTH_NAME,
FISCAL_MONTH_NAME,
to_number(to_char(TIME_ID,'YYYY')+15)||'-
'||lpad(CALENDAR_QUARTER_NUMBER,2,'0'),
```

```
CALENDAR_QUARTER_ID,
FISCAL_QUARTER_DESC,
FISCAL_QUARTER_ID,
DAYS_IN_CAL_QUARTER,
DAYS_IN_FIS_QUARTER,
END_OF_CAL_QUARTER,
END_OF_FIS_QUARTER,
CALENDAR_QUARTER_NUMBER,
FISCAL_QUARTER_NUMBER,
to_number(to_char(TIME_ID,'YYYY')+15),
CALENDAR_YEAR_ID,
FISCAL_YEAR,
FISCAL_YEAR_ID,
DAYS_IN_CAL_YEAR,
DAYS_IN_FIS_YEAR,
END_OF_CAL_YEAR,
END_OF_FIS_YEAR
from times_org;
commit;
```

SALES表の日付データに対応したTIMES表のカスタマイズ(4)

6. 統計情報の取得

--カスタマイズしたTIMES表に主キーを作成□

```
create unique index TIMES_NEW_PK on TIMES (TIME_ID) local;  
alter table TIMES add primary key (TIME_ID) using index;
```

-- SHスキーマのオプティマイザ統計情報を収集□

```
exec dbms_stats.gather_schema_stats (ownname=>'SH', degree=>8);
```

SALES表の日付データに対応したTIMES表のカスタマイズ(5)

```
10:46:04 SQL> @times_customize.sql
```

```
Table renamed.
```

```
Table created.
```

```
1826 rows created.
```

```
Commit complete.
```

```
Index created.
```

```
Table altered.
```

```
PL/SQL procedure successfully completed.
```

```
10:49:24 SQL>
```


7. ファイルシステムに対する非同期I/OとダイレクトI/Oを有 化

```
[oracle@jphppt06 ~]$ sqlplus / as sysdba
SQL*Plus: Release 19.0.0.0.0 - Production on Thu Sep 12 17:22:09 2019
Version 19.4.0.0.0
Copyright (c) 1982, 2019, Oracle. All rights reserved.
Connected to:
Oracle Database 19c Enterprise Edition Release 19.0.0.0.0 - Production
Version 19.4.0.0.0

SQL> alter system set FILESYSTEMIO_OPTIONS='SETALL' scope=spfile ;

System altered.

SQL> shutdown immediate
Database closed.
Database dismounted.
ORACLE instance shut down.
SQL> startup
ORACLE instance started.
Total System Global Area 2.0267E+10 bytes
Fixed Size 16240904 bytes
Variable Size 2818572288 bytes
Database Buffers 1.4194E+10 bytes
Redo Buffers 17309696 bytes
In-Memory Area 3221225472 bytes
Database mounted.
Database opened.
```

ファイルシステムに対する非同期I/OとダイレクトI/Oを有効化

```
SQL> alter pluggable database orclmk open;
```

```
Pluggable database altered.
```

```
SQL> show pdbs
```

CON_ID	CON_NAME	OPEN MODE	RESTRICTED
2	PDB\$SEED		READ ONLY NO
3	ORCLMP	MOUNTED	
5	ORCLMK	READ WRITE	NO

```
SQL>
```

PDBへのEMの接続□

ORACLE® Enterprise Manager Cloud Control 13c

Enterprise(E) ▼ ターゲット(T) ▼ ★ ▼ ⌚ ▼ ⚙ ▼ 🔍 🔔 SYSMAN ▼ ...

データベース

パフォーマンス ▼ 可用性 ▼ セキュリティ ▼ スキーマ ▼ 管理 ▼

表示 ○ データベース・ロードマップ ● 検索リスト

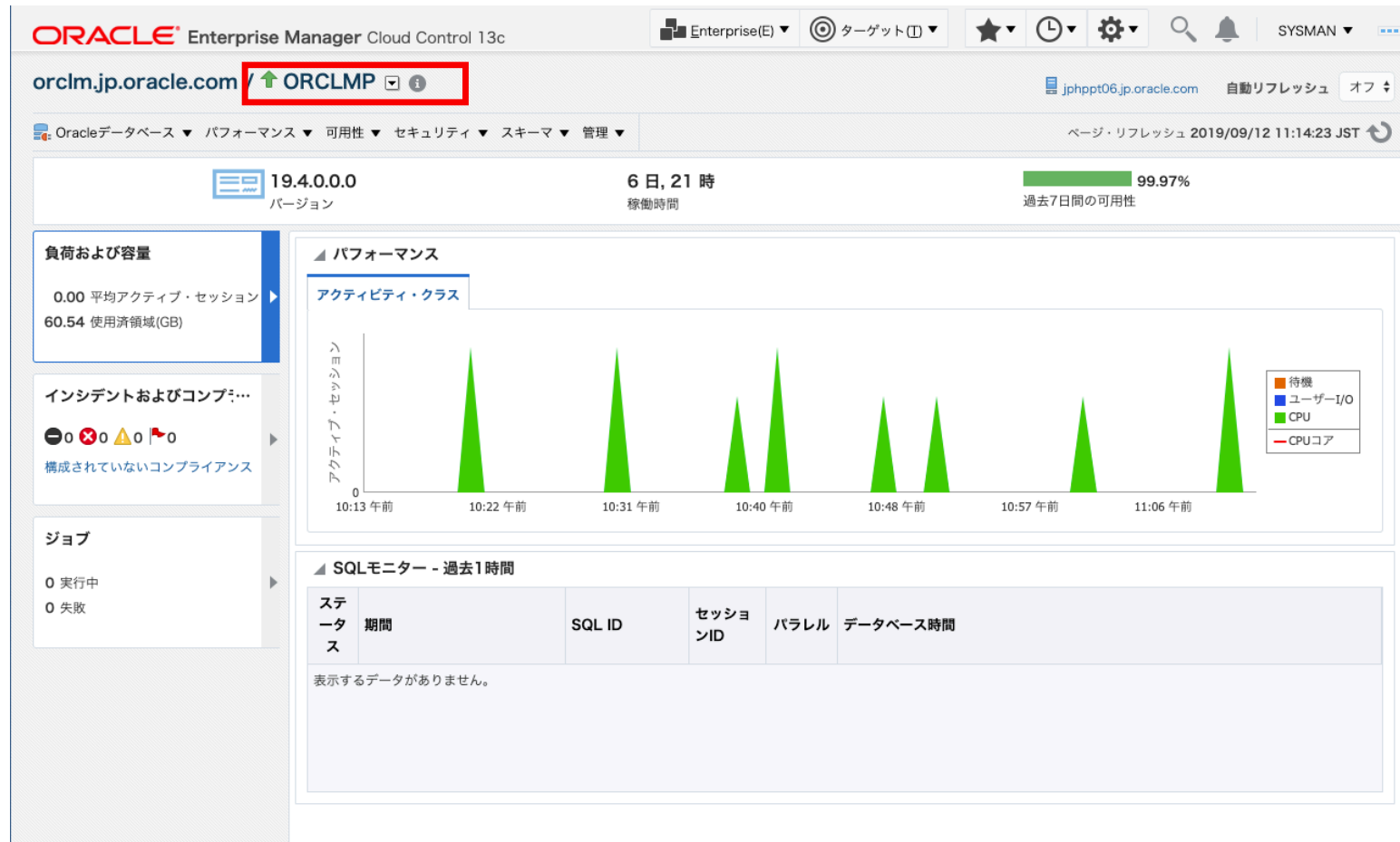
▲ 検索

検索 名前

表示 ▼ + 追加 ▼ ✕ 削除 構成

名前	タイプ	ステータス	ターゲット・バージョン	インシデント			平均コンプライアンス・スコア	メンバー・ステータス・サマリー			
				0	✕	!		↓	↑	✕	!
▶ orcl.jp.oracle.com	データベース・インスタンス: コンテナ	!	12.2.0.1.0	0	0	0	N/A	0	0	0	1
▶ orcl193s	データベース・インスタンス: コンテナ	!	19.3.0.0.0	0	2	2	N/A	0	0	0	1
▶ orclenc	データベース・インスタンス: コンテナ	!	19.3.0.0.0	0	3	3	N/A	0	0	0	2
▲ orclm.jp.oracle.com	データベース・インスタンス: コンテナ	↑	19.4.0.0.0	0	0	0	N/A	0	2	0	0
▲ プラガブル・データベース		N/A		0	0	0	N/A	0	2	0	0
orclm.jp.oracle.com_ORCLMK	プラガブル・データベース	↑	19.4.0.0.0	0	0	0	N/A	0	0	0	0
orclm.jp.oracle.com_ORCLMP	プラガブル・データベース	↑	19.4.0.0.0	0	0	0	N/A	0	0	0	0
test11gr2.jp.oracle.com	データベース・インスタンス	!	11.2.0.4.0	0	2	1	N/A	0	0	0	0
testlink	データベース・インスタンス	!	11.2.0.4.0	0	1	1	N/A	0	0	0	0

EMからPDBを確認□



接続するPDBの切り替え□

The screenshot displays the Oracle Enterprise Manager Cloud Control 13c interface. The top navigation bar includes the Oracle logo, 'Enterprise Manager Cloud Control 13c', and various utility icons. The main breadcrumb path is 'orclm.jp.oracle.com / ↑ ORCLMP'. A dropdown menu is open, showing the hierarchy: 'orclm.jp.oracle.com (コンテナ・データベース)' > 'ORCLMP' (highlighted with a red box) > 'ORCLMK' > 'CDB\$ROOT'. The left sidebar contains sections for 'Oracleデータベース', 'パフォーマンス', and '可用性', along with a version indicator '19.4.0.0.0'. The main content area shows a 'パフォーマンス' (Performance) section with a graph of 'アクティビティ・クラス' (Activity Class) over time, and an 'SQLモニター - 過去1時間' (SQL Monitor - Last 1 hour) section which is currently empty.

ORACLE® Enterprise Manager Cloud Control 13c

Enterprise(E) ▼ ターゲット(1) ▼

orclm.jp.oracle.com / ↑ ORCLMP

Oracleデータベース ▼ パフォーマンス ▼ 可用性 ▼

19.4.0.0.0
バージョン

ORCLMP
ORCLMK
CDB\$ROOT
すべてのコンテナ...

99.97%
過去7日間の可用性

アクティビティ・クラス

SQLモニター - 過去1時間

ステータス	期間	SQL ID	セッションID	パラレル	データベース時間
表示するデータがありません。					

ORA-12514エラーでPDBにうまくつながらないとき□

The screenshot displays the Oracle Enterprise Manager Cloud Control 13c interface. The top navigation bar includes the Oracle logo, the text "Enterprise Manager Cloud Control 13c", and various utility icons. Below this, the breadcrumb path "orclm.jp.oracle.com / ↑ ORCLMK" is visible. A red box highlights an error message in the "情報" (Information) section: "Listener refused the connection with the following error: ORA-12514, TNS:listener does not currently know of service requested in connect descriptor". Below the error message, a table shows database metrics: "バージョン" (Version) is N/A, "稼働時間" (Uptime) is N/A, and "過去7日間の可用性" (Availability over the last 7 days) is 99.97%. On the left sidebar, there are sections for "負荷および容量" (Load and Capacity), "インシデントおよびコンプライアンス" (Incidents and Compliance), and "ジョブ" (Jobs).

ORACLE® Enterprise Manager Cloud Control 13c

Enterprise(E) ▼ ターゲット(1) ▼ ★ ▼ ⌚ ▼ ⚙ ▼ 🔍 🔔 SYSMAN ▼ ...

orclm.jp.oracle.com / ↑ ORCLMK ▼ ⓘ

jphppt06.jp.oracle.com 自動リフレッシュ オフ

Oracleデータベース ▼ パフォーマンス ▼ 可用性 ▼ セキュリティ ▼ スキーマ ▼ 管理 ▼

ページ・リフレッシュ 2019/09/12 11:14:44 JST ↺

情報

Listener refused the connection with the following error: ORA-12514, TNS:listener does not currently know of service requested in connect descriptor

	N/A バージョン	N/A 稼働時間	99.97% 過去7日間の可用性
--	--------------	-------------	---------------------

負荷および容量

N/A 平均アクティブ・セッション
32.12 使用済領域(GB)

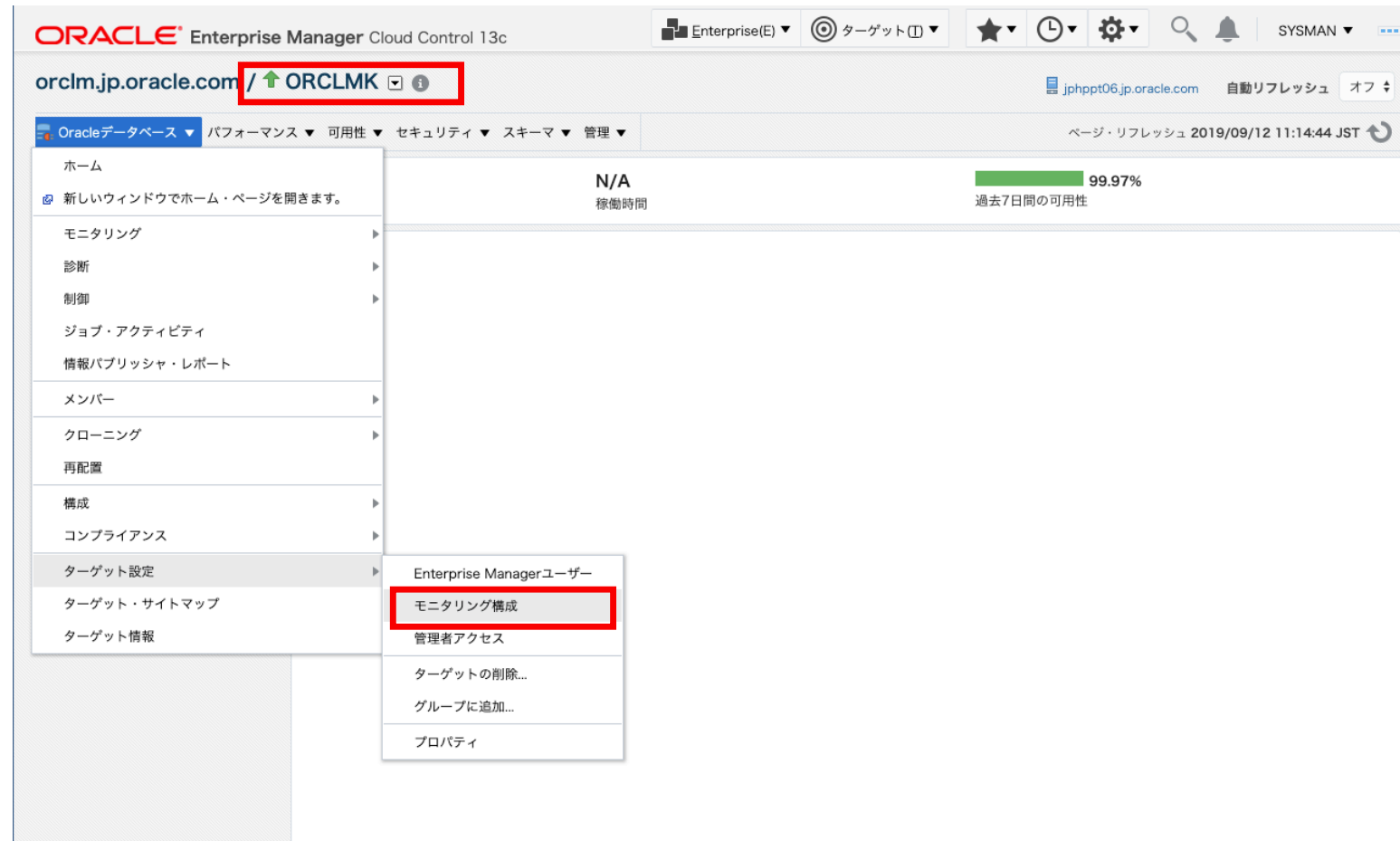
インシデントおよびコンプライアンス

0 0 0 0 0 0
構成されていないコンプライアンス

ジョブ

0 実行中
0 失敗

Oracleデータベース→ターゲット設定→モニタリング構成□



デフォルトサービス名 → 接続テスト

ORACLE® Enterprise Manager Cloud Control 13c SYSMAN ▾

プラグابل・データベースの編集: プロパティ 取消 続行

ターゲット名 orclm.jp.oracle.com_ORCLMK
ターゲット・タイプ プラグابل・データベース

接続テスト

名前	値
名前	ORCLMK
デフォルト・サービス名	orclmk
アプリケーションPDB	NO
アプリケーション・ルート	NO
コンテナ名	orclm.jp.oracle.com_

優先接続文字列
OMSでプラグابل・データベースへの接続時に使用する接続文字列を入力します。空白の場合、OMSではコンテナ・データベースに指定したホスト、ポート、SIDを使用して接続文字列が自動的に構成されます。

コンテナ・データベースのプロパティ 取消 続行

SERVICE_NAME=orclmk でエラー□

ORACLE® Enterprise Manager Cloud Control 13c SYSMAN ▾

取消 続行

✖ エラー

orclm.jp.oracle.com_ORCLMK - データベースへの接続に失敗しました: Listener refused the connection with the following error: ORA-12514, TNS:listener does not currently know of service requested in connect descriptor The Connect Descriptor was (DESCRIPTION = (ADDRESS = (PROTOCOL = tcp)(HOST = jphppt06.jp.oracle.com)(PORT = 1521))(CONNECT_DATA = (SERVICE_NAME = orclmk)))

ブラガブル・データベースの編集: プロパティ

ターゲット名: orclmprodclmcom_01dc6m

ターゲット・タイプ: ブラガブル・データベース

接続テスト

名前	値
名前	ORCLMK
デフォルト・サービス名	orclmk
アプリケーションPDB	NO
アプリケーション・ルート	NO
コンテナ名	orclm.jp.oracle.com_
優先接続文字列	OMSでブラガブル・データベースへの接続時に使用する接続文字列を入力します。空白の場合、OMSではコンテナ・データベースに指定したホスト、ポート、SIDを使用して接続文字列が自動的に構成されます。

コンテナ・データベースのプロパティ

取消 続行

デフォルト・サービス名をorclmk.jp.oracle.com へ変更□

ORACLE® Enterprise Manager Cloud Control 13c SYSMAN ▾

✖ エラー

orclmk.jp.oracle.com_ORCLMK - データベースへの接続に失敗しました: Listener refused the connection with the following error: ORA-12514, TNS:listener does not currently know of service requested in connect descriptor The Connect Descriptor was (DESCRIPTION = (ADDRESS = (PROTOCOL = tcp)(HOST = jphppt06.jp.oracle.com)(PORT = 1521))(CONNECT_DATA = (SERVICE_NAME = orclmk)))

ブラガブル・データベースの編集: プロパティ

ターゲット名 orclmk.jp.oracle.com_ORCLMK

ターゲット・タイプ ブラガブル・データベース

名前	値
名前	ORCLMK
デフォルト・サービス名	orclmk.jp.oracle.com
アプリケーションPDB	NO
アプリケーション・ルート	NO
コンテナ名	orclmk.jp.oracle.com_
優先接続文字列	

OMSでブラガブル・データベースへの接続時に使用する接続文字列を入力します。空白の場合、OMSではコンテナ・データベースに指定したホスト、ポート、SIDを使用して接続文字列が自動的に構成されます。

▷ コンテナ・データベースのプロパティ

接続テスト成功

ORACLE® Enterprise Manager Cloud Control 13c SYSMAN ▾

取消 続行

① 成功
orclm.jp.oracle.com_ORCLMK - 接続テストに成功しました。
プラグابل・データベースの編集: プロパティ
ターゲット名 orclm.jp.oracle.com_ORCLMK
ターゲット・タイプ プラグابل・データベース

接続テスト

名前	値
名前	ORCLMK
デフォルト・サービス名	orclm.jp.oracle.com
アプリケーションPDB	NO
アプリケーション・ルート	NO
コンテナ名	orclm.jp.oracle.com_
優先接続文字列 OMSでプラグابل・データベースへの接続時に使用する接続文字列を入力します。空白の場合、OMSではコンテナ・データベースに指定したホスト、ポート、SIDを使用して接続文字列が自動的に構成されます。	

▷ コンテナ・データベースのプロパティ

取消 続行

発行口

ORACLE[®] Enterprise Manager Cloud Control 13c

SYSMAN ▾ ⋮

取消戻る発行

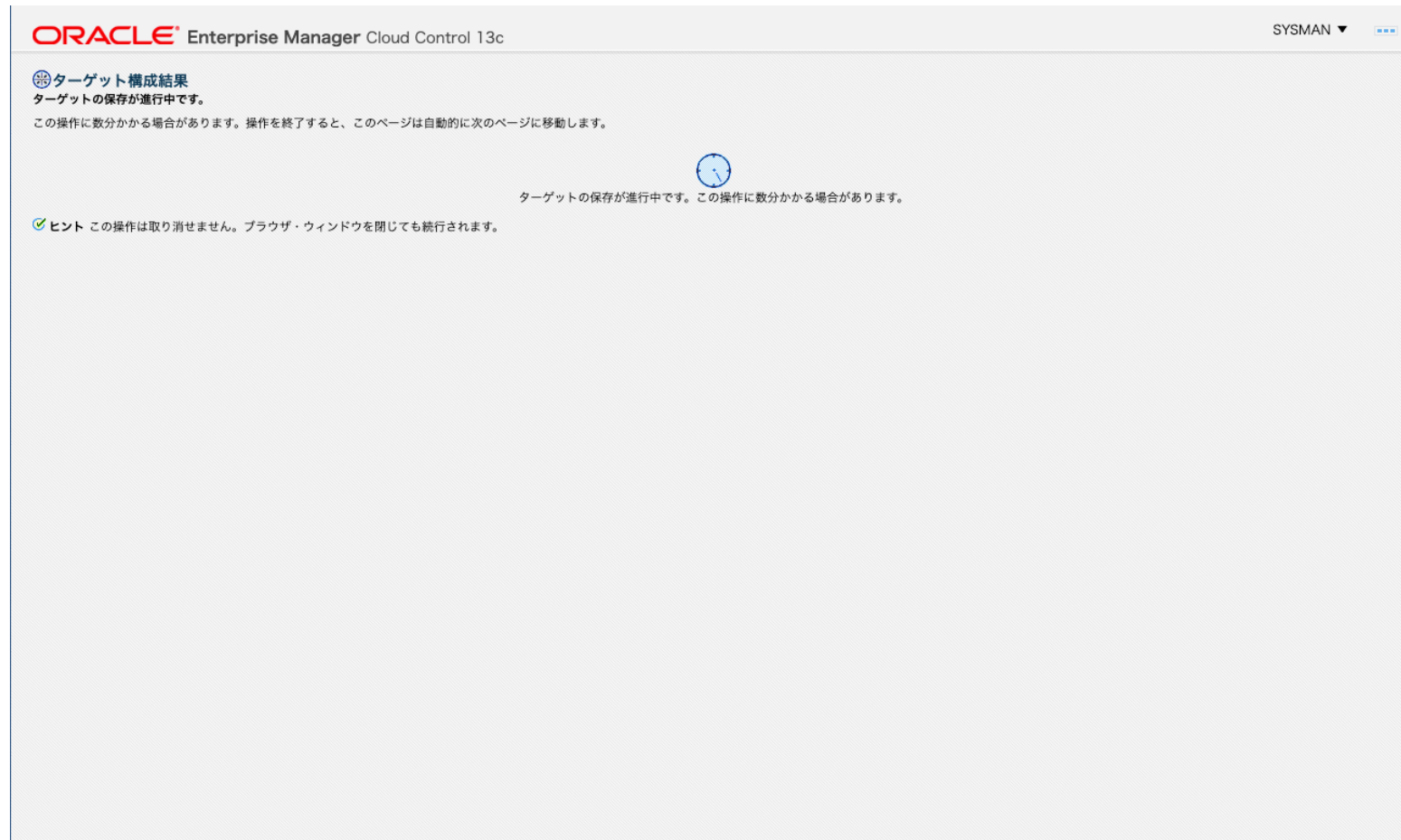
ブラガブル・データベースの編集: 確認
このブラガブル・データベースに対して行われた変更を確認してください。確認が終了したら「発行」を選択し、変更を編集する場合は「戻る」を選択します。

ターゲット名 orclm.jp.oracle.com_ORCLMK
ターゲット・タイプ ブラガブル・データベース

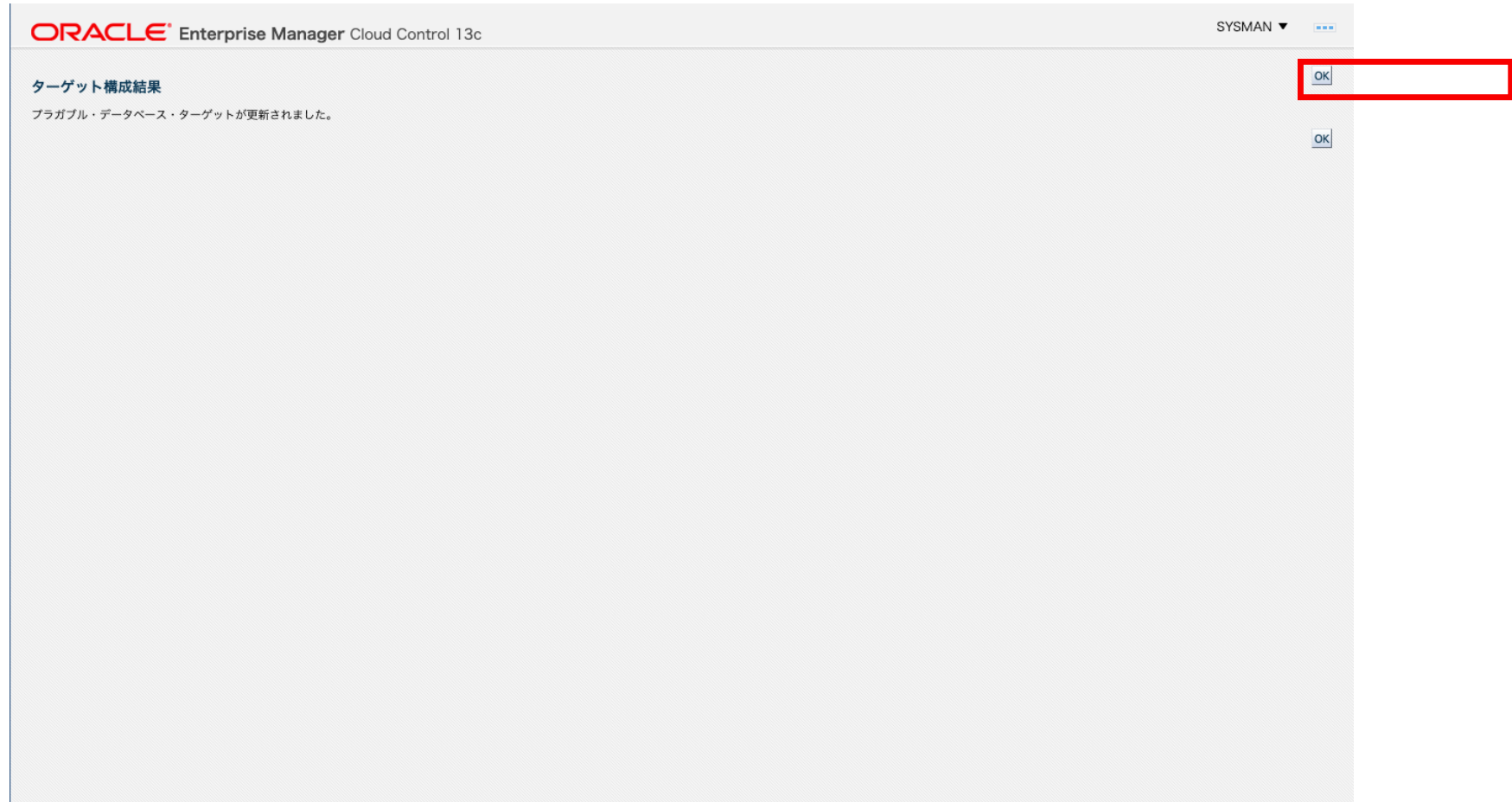
名前	値
名前	ORCLMK
デフォルト・サービス名	ordmk.jp.oracle.com
アプリケーションPDB	NO
アプリケーション・ルート	NO
コンテナ名	ordm.jp.oracle.com_
優先接続文字列	

取消戻る発行

進行中□



OK



チューニング対象のSQL文□

CPUバウンド

```
WITH /*+MONITOR */
DUMMY_SALES AS
( select * from (select 0 from CHANNELS ) D1, sales D2),
SACOMMON1340 AS
( select sum(T220.AMOUNT_SOLD) as c1, sum(T220.QUANTITY_SOLD) as c2,
      T147.CHANNEL_CLASS as c3, T228.CALENDAR_QUARTER_DESC as c4,
      T228.CALENDAR_YEAR as c5, T185.PROD_CATEGORY as c6
    from CHANNELS T147, PRODUCTS T185,
    DUMMY_SALES T220, TIMES T228
   where ( T220.TIME_ID < to_date('2014/01/01','YYYY/MM/DD')
         and T228.TIME_ID = T220.TIME_ID
         and T147.CHANNEL_ID = T220.CHANNEL_ID
         and T185.PROD_ID = T220.PROD_ID)
   group by T147.CHANNEL_CLASS,
            T185.PROD_CATEGORY,
            T228.CALENDAR_QUARTER_DESC,
            T228.CALENDAR_YEAR),
SAWITH0 AS
( select distinct 0 as c1, D1.c3 as c2, D1.c4 as c3, D1.c5 as c4,
      D1.c6 as c5, D1.c2 as c6, D1.c1 as c7, cast(NULL as DOUBLE PRECISION ) as c8
  from SACOMMON1340 D1),
SAWITH1 AS
( select D1.c1 as c1, D1.c2 as c2, D1.c3 as c3, D1.c4 as c4,
      D1.c5 as c5, D1.c6 as c6, D1.c7 as c7, D1.c8 as c8, sum(D1.c7) as c9
  from SAWITH0 D1
   group by D1.c1, D1.c2, D1.c3, D1.c4, D1.c5, D1.c6, D1.c7, D1.c8),
SAWITH2 AS
( select distinct 1 as c1, D1.c3 as c2, D1.c4 as c3, D1.c5 as c4,
      D1.c6 as c5, D1.c2 as c6, D1.c1 as c7
  from SACOMMON1340 D1),
SAWITH3 AS
( select D1.c1 as c1, D1.c2 as c2, D1.c3 as c3, D1.c4 as c4,
      D1.c5 as c5, D1.c6 as c6, D1.c7 as c7, sum(D1.c6) as c8, sum(D1.c7) as c9
  from SAWITH2 D1
   group by D1.c1, D1.c2, D1.c3, D1.c4, D1.c5, D1.c6, D1.c7),
SAWITH4 AS
(( select D1.c1 as c1, D1.c2 as c2, D1.c3 as c3, D1.c4 as c4, D1.c5 as c5,
      D1.c6 as c6, D1.c7 as c7, D1.c8 as c8,
      sum(D1.c9) over (partition by D1.c3, D1.c4, D1.c5) as c9
  from SAWITH1 D1
   union all
  select D1.c1 as c1, D1.c2 as c2, D1.c3 as c3, D1.c4 as c4, D1.c5 as c5,
      D1.c6 as c6, D1.c7 as c7,
      sum(D1.c8) over (partition by D1.c3, D1.c4, D1.c5) as c8,
      sum(D1.c9) over (partition by D1.c3, D1.c4, D1.c5) as c9
  from SAWITH3 D1 ))
select D1.c1 as c1, D1.c2 as c2, D1.c3 as c3, D1.c4 as c4, D1.c5 as c5,
      D1.c6 as c6, D1.c7 as c7, D1.c8 as c8, D1.c9 as c9
from SAWITH4 D1 order by c1, c3, c5, c4;
```

I/Oバウンド

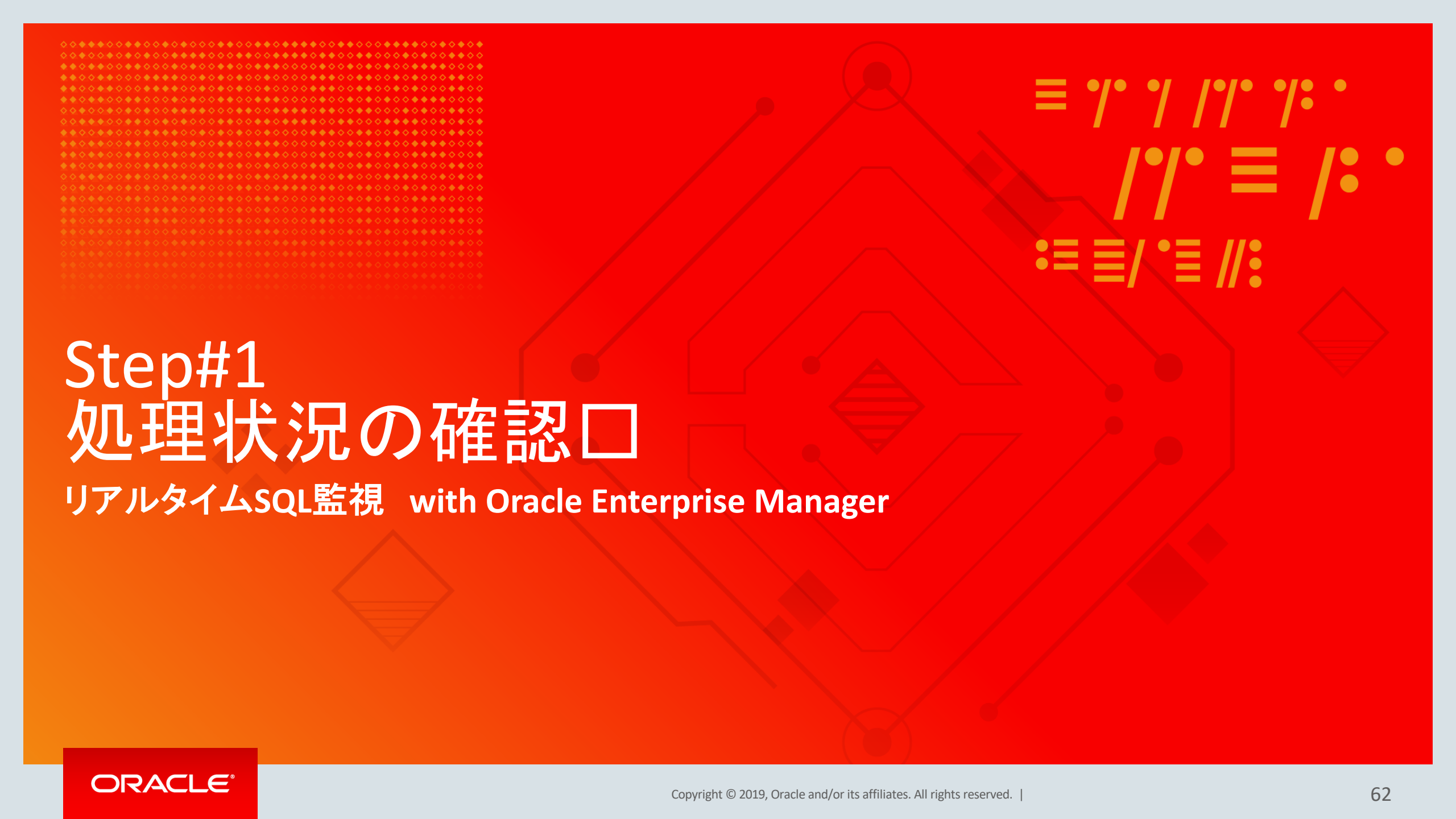
```
WITH /*+MONITOR */
SACOMMON1340 AS
( select sum(T220.AMOUNT_SOLD) as c1, sum(T220.QUANTITY_SOLD) as c2,
      T147.CHANNEL_CLASS as c3, T228.CALENDAR_QUARTER_DESC as c4,
      T228.CALENDAR_YEAR as c5, T185.PROD_CATEGORY as c6
    from CHANNELS T147, PRODUCTS T185,
    SALES T220, TIMES T228
   where ( T220.TIME_ID < to_date('2014/01/01','YYYY/MM/DD')
         and T228.TIME_ID = T220.TIME_ID
         and T147.CHANNEL_ID = T220.CHANNEL_ID
         and T185.PROD_ID = T220.PROD_ID)
   group by T147.CHANNEL_CLASS,
            T185.PROD_CATEGORY,
            T228.CALENDAR_QUARTER_DESC,
            T228.CALENDAR_YEAR),
SAWITH0 AS
( select distinct 0 as c1, D1.c3 as c2, D1.c4 as c3, D1.c5 as c4,
      D1.c6 as c5, D1.c2 as c6, D1.c1 as c7, cast(NULL as DOUBLE PRECISION ) as c8
  from SACOMMON1340 D1),
SAWITH1 AS
( select D1.c1 as c1, D1.c2 as c2, D1.c3 as c3, D1.c4 as c4,
      D1.c5 as c5, D1.c6 as c6, D1.c7 as c7, D1.c8 as c8, sum(D1.c7) as c9
  from SAWITH0 D1
   group by D1.c1, D1.c2, D1.c3, D1.c4, D1.c5, D1.c6, D1.c7, D1.c8),
SAWITH2 AS
( select distinct 1 as c1, D1.c3 as c2, D1.c4 as c3, D1.c5 as c4,
      D1.c6 as c5, D1.c2 as c6, D1.c1 as c7
  from SACOMMON1340 D1),
SAWITH3 AS
( select D1.c1 as c1, D1.c2 as c2, D1.c3 as c3, D1.c4 as c4,
      D1.c5 as c5, D1.c6 as c6, D1.c7 as c7, sum(D1.c6) as c8, sum(D1.c7) as c9
  from SAWITH2 D1
   group by D1.c1, D1.c2, D1.c3, D1.c4, D1.c5, D1.c6, D1.c7),
SAWITH4 AS
(( select D1.c1 as c1, D1.c2 as c2, D1.c3 as c3, D1.c4 as c4, D1.c5 as c5,
      D1.c6 as c6, D1.c7 as c7, D1.c8 as c8,
      sum(D1.c9) over (partition by D1.c3, D1.c4, D1.c5) as c9
  from SAWITH1 D1
   union all
  select D1.c1 as c1, D1.c2 as c2, D1.c3 as c3, D1.c4 as c4, D1.c5 as c5,
      D1.c6 as c6, D1.c7 as c7,
      sum(D1.c8) over (partition by D1.c3, D1.c4, D1.c5) as c8,
      sum(D1.c9) over (partition by D1.c3, D1.c4, D1.c5) as c9
  from SAWITH3 D1 ))
select D1.c1 as c1, D1.c2 as c2, D1.c3 as c3, D1.c4 as c4, D1.c5 as c5,
      D1.c6 as c6, D1.c7 as c7, D1.c8 as c8, D1.c9 as c9
from SAWITH4 D1 order by c1, c3, c5, c4;
```

チューニングStep

- 1 処理状況の確認□
- 2 パーティション化での処理量削減□
- 3 パラレル化で複数CPUコアの利用□
- 4 データ圧縮で更なるI/O量の削減□
- 5 インメモリ化での高速化□

チューニングStep

- 1 処理状況の確認□
- 2 パーティション化での処理量削減□
- 3 パラレル化で複数CPUコアの利用□
- 4 データ圧縮で更なるI/O量の削減□
- 5 インメモリ化での高速化□



Step#1 処理状況の確認□

リアルタイムSQL監視 with Oracle Enterprise Manager

リアルタイムSQL監視 (SQLモニタリング) へのアクセス口

Oracle Enterprise Manager Cloud Control 13c

データベース

パフォーマンス ▼ 可用性 ▼ セキュリティ ▼ スキーマ ▼ 管理 ▼

表示 ○ データベース ・ ロードマップ ● 検索リスト

検索 名前

表示 ▼ + 追加 ▼ ✕ 削除 構成

名前	タイプ	ステータス	ジョブ	平均コンプライアンススコア	メンバー・ステータス・サマリー
orcl.jp.oracle.com	データベース・インスタンス: コンテナ	🟡	12.2.0.1.0	0 0 0	N/A 0 0 0 1
orcl193s	データベース・インスタンス: コンテナ	🟡	19.3.0.0.0	0 2 2	N/A 0 0 0 1
orclenc	データベース・インスタンス: コンテナ	🟡	19.3.0.0.0	0 3 3	N/A 0 0 0 2
orclm.jp.oracle.com	データベース・インスタンス: コンテナ	🟢	19.4.0.0.0	0 0 0	N/A 0 1 0 0
プラガブル・データベース		N/A		0 0 0	N/A 0 1 0 0
orclm.jp.oracle.com_ORCLMP	プラガブル・データベース	🟢	19.4.0.0.0	0 0 0	N/A 0 0 0 0
test11gr2.jp.oracle.com	データベース・インスタンス	🟡	11.2.0.4.0	0 2 1	N/A 0 0 0 0
testlink	データベース・インスタンス	🟡	11.2.0.4.0	0 1 1	N/A 0 0 0 0

ORACLE Enterprise Manager Cloud Control 13c

orclm.jp.oracle.com / CDB\$ROOT

パフォーマンス ▼ 可用性 ▼ セキュリティ ▼ スキーマ ▼ 管理 ▼

6 日, 21 時 稼働時間 過去7日間の可用性 100%

負荷および容量

0.01 平均アクティブ・セッション 1.93 使用済領域 (GB)

インシデントおよびコンプライアンス

0 SQLチューニング

ジョブ

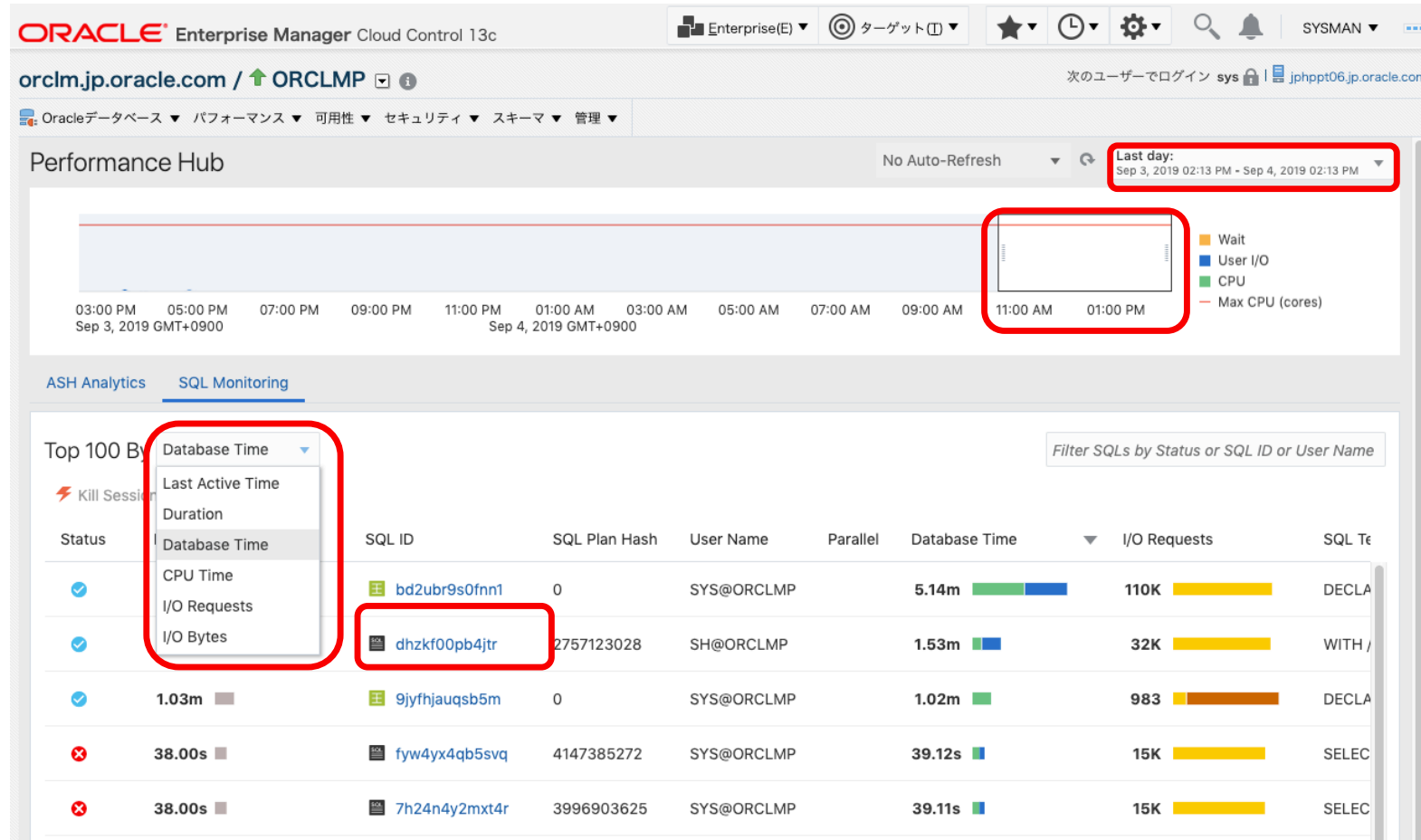
0 実行中 0 失敗

SQLモニタリング

SQLモニタリング - 過去1時間

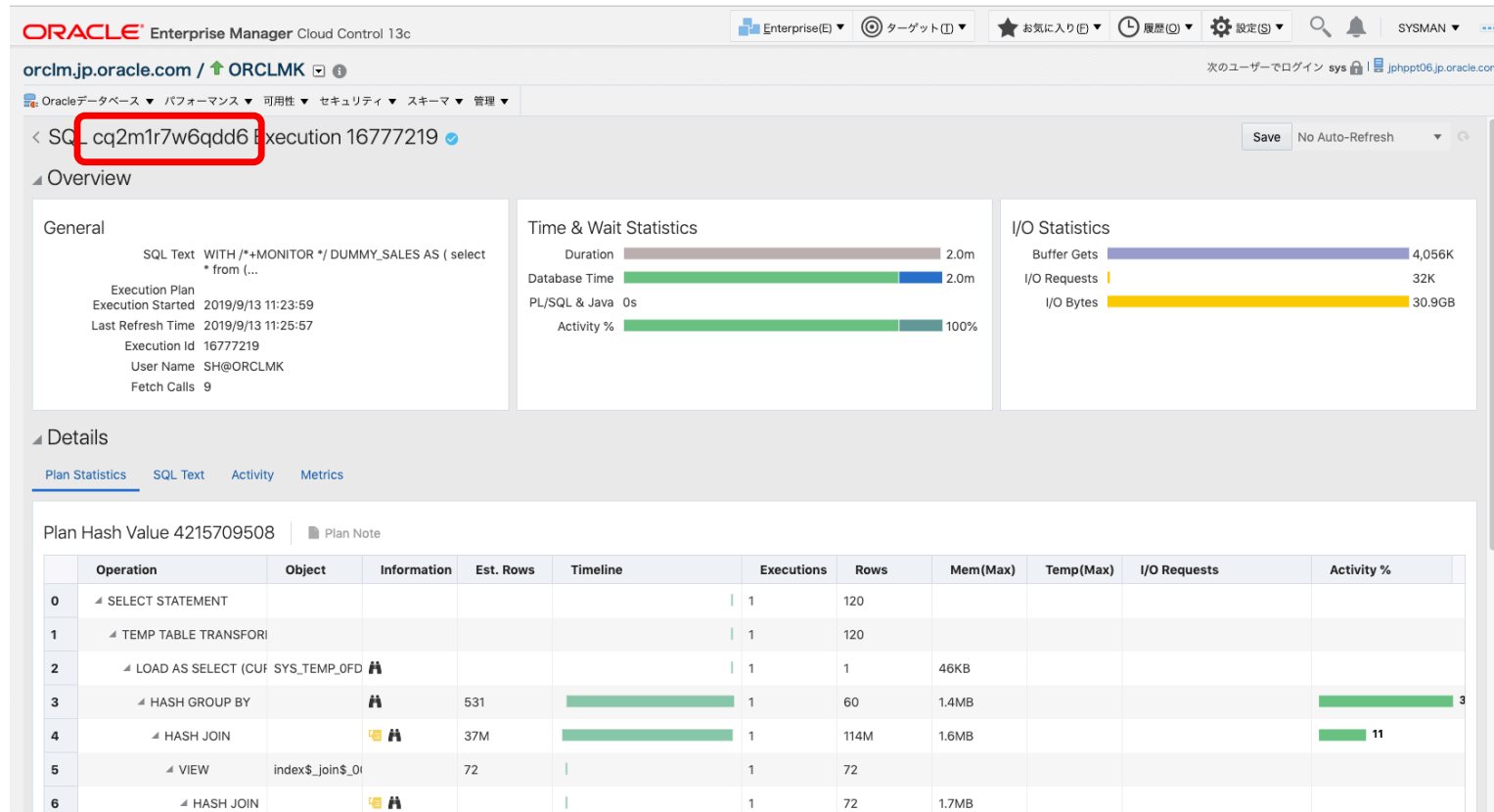
ステータス	期間	SQL ID	セッションID	パラレル	データベース時間
🟢	0.02 s	01ncdk23uqcks	129	2	0.02 s
🟢	0.01 s	01ncdk23uqcks	4	2	0.01 s
🟢	0.02 s	01ncdk23uqcks	2668	2	0.02 s
🟢	0.02 s	01ncdk23uqcks	129	2	0.02 s

リアルタイムSQL監視 (SQLモニタリング) へのアクセス口

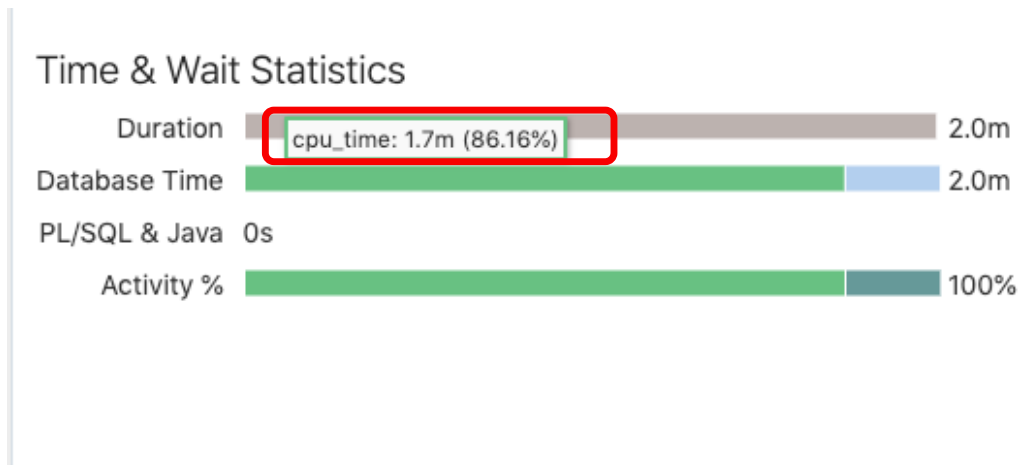


CPUバウンドなSQLの処理状況

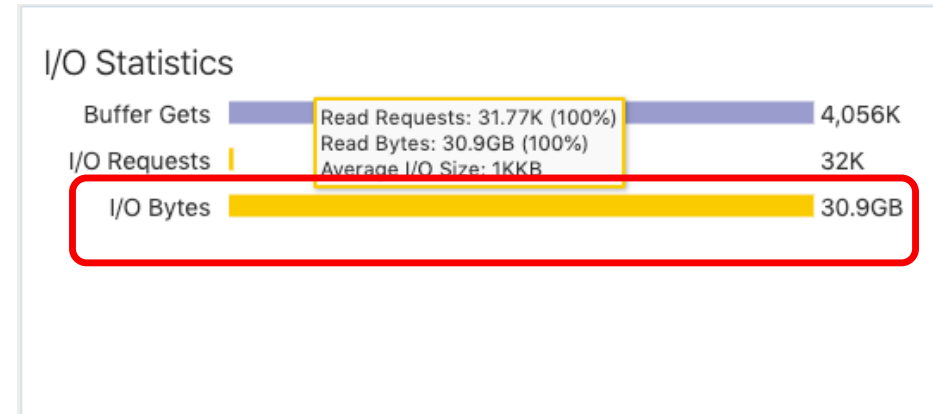
リアルタイムSQL監視アクティブ・レポート



CPUバウンドなSQLの処理状況(詳細) □



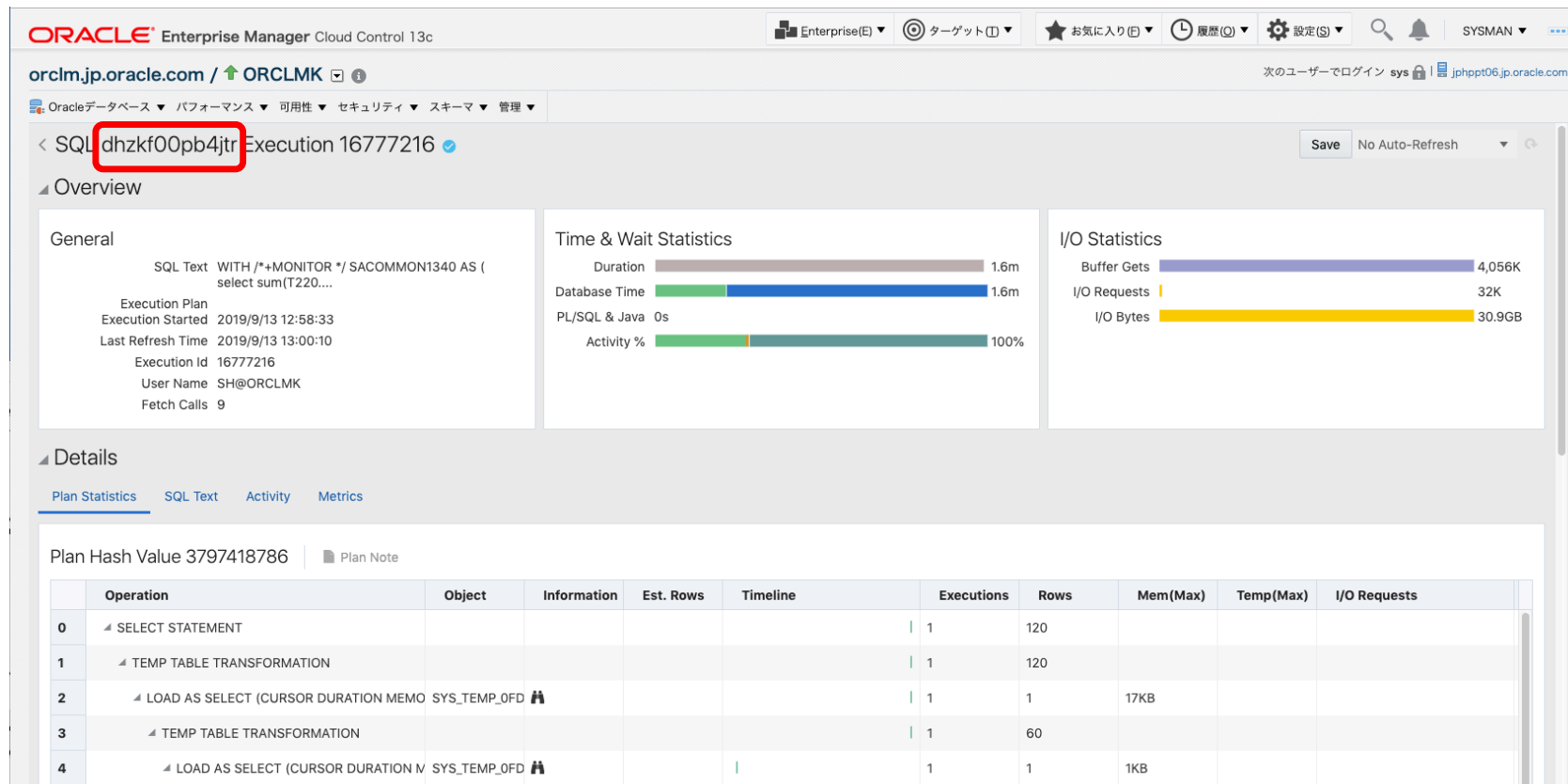
- Duration: SQL実行時間は2.0m(≒118秒)
- Database Time: **CPU=86%, User IO=14%**
- Activity%: 12%の割合でdirect path read 待機イベントが発生 □



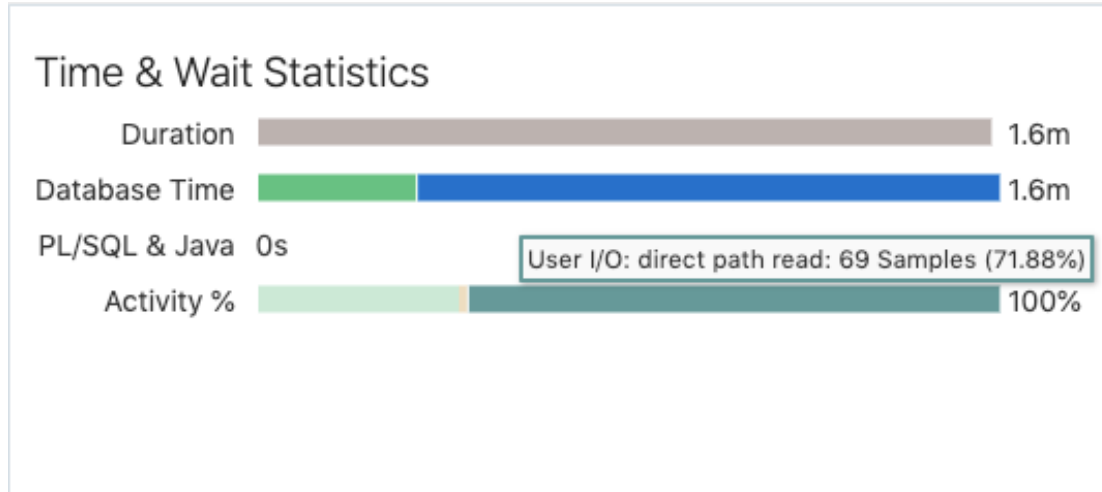
- Buffer Gets: 低いキャッシュ・ヒット率 □
- IO Requests: 平均I/Oサイズ1MBで32K回 □
- IO Bytes: **総読込み量は約31GB**

I/OバウンドなSQLの処理状況

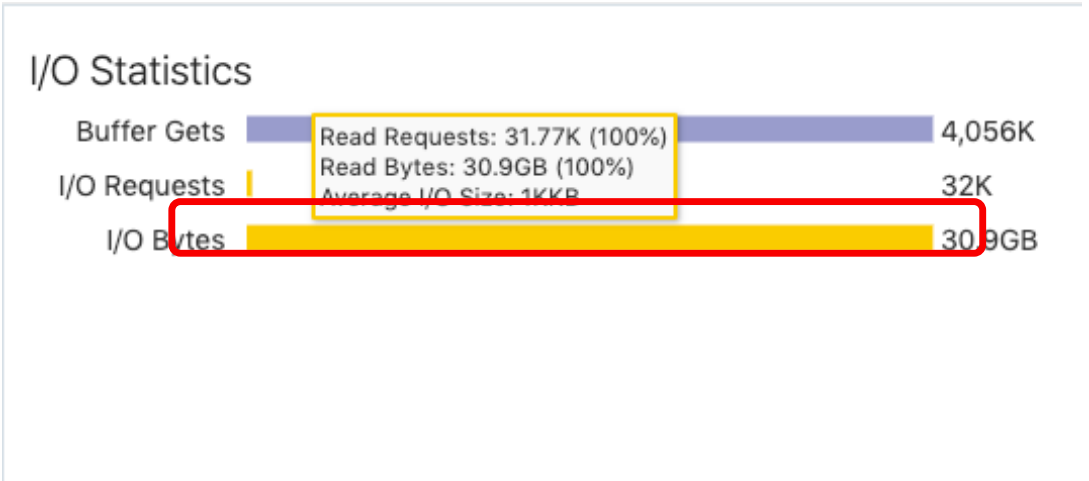
リアルタイムSQL監視アクティブ・レポート



I/OバウンドなSQLの処理状況(詳細) □



- Duration: SQL実行時間は1.6m(≒97秒)
- Database Time: CPU=21%, User IO=79%
- Activity%: 72%の割合でdirect path read 待機イベントが発生 □



- Buffer Gets: 低いキャッシュ・ヒット率 □
- IO Requests: 平均I/Oサイズ1MBで32K回 □
- IO Bytes: 総読込み量は約31GB

(参考)リアルタイムSQL監視□

大量のリソースを消費する長時間実行SQL文のパフォーマンス問題を効果的に特定

- 特徴□

- 経過時間、CPU時間、読取りと書込みの回数、I/O待機時間、その他の各種待機時間□
などの主要なパフォーマンス指標ごとに、実行計画の各ステップで追跡□

- 開始方法□

- 次のどちらかの条件(デフォルト)を満たす場合、自動的にSQL監視を開始□
 - SQL文が**パラレル**で実行される場合□
 - **1回の実行で5秒以上**のCPUまたはI/O時間を消費している場合□
- 明示的に、対象SQL文に“MONITOR”ヒント句を追記しても監視可能□

- SQL監視アクティブ・レポートの例やFAQはコチラ□

- <http://www.oracle.com/technetwork/jp/database/sqlmonitor-101860-ja.html>

チューニングStep

- 1 処理状況の確認□
- 2 パーティション化での処理量削減□
- 3 パラレル化で複数CPUコアの利用□
- 4 データ圧縮で更なるI/O量の削減□
- 5 インメモリ化での高速化□



Step#2 パーティション化での処理量削減

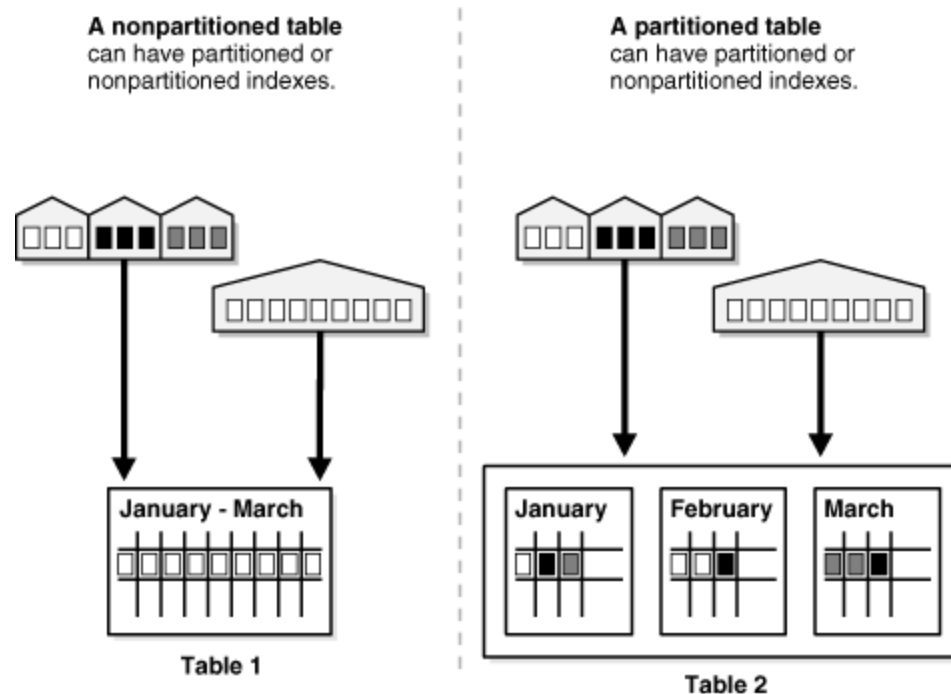
Oracle Partitioning Option

パーティション・アドバイザ(SQLアクセス・アドバイザ)

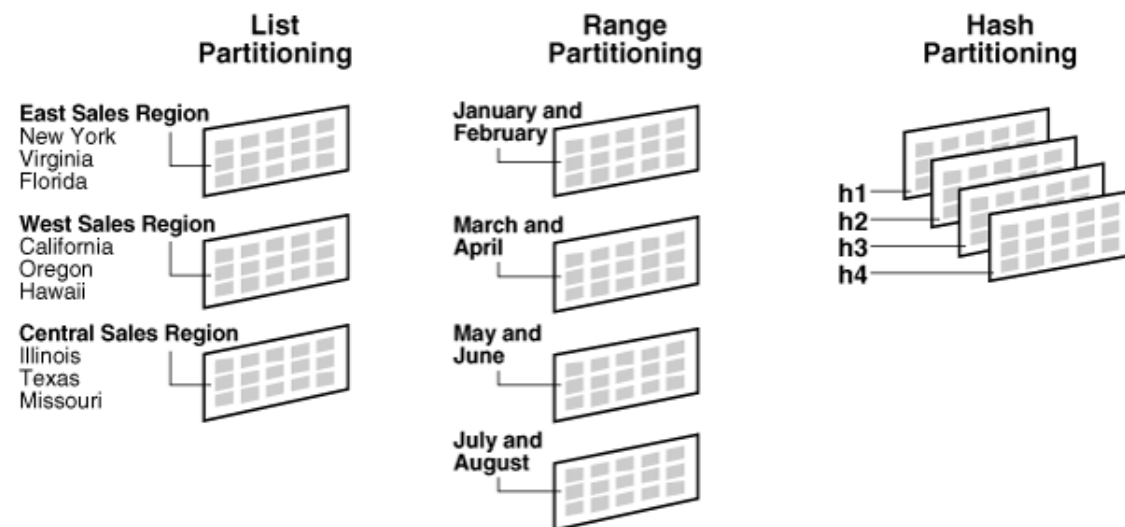
オンラインでパーティション表への変換(12.2~)

パーティションって何？ □

表を内部的に分割して管理することで、
パフォーマンス、管理性、可用性が向上する



3つの基本的なデータ配分方法を提供し、
アプリケーション(SQL)はそれらを意識しない



Oracle® Database 19c VLDBおよびパーティショニング・ガイド
パーティション化の概念

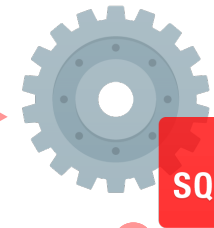
パーティション・プルーニング□

読み込み対象データを限定することで、処理量（CPU時間、I/O量）を削減可能

```
SQL> select * from TABLE1  
where COLOR = 'RED' ;
```

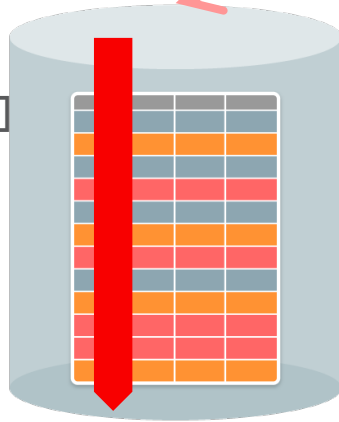


Oracle Client

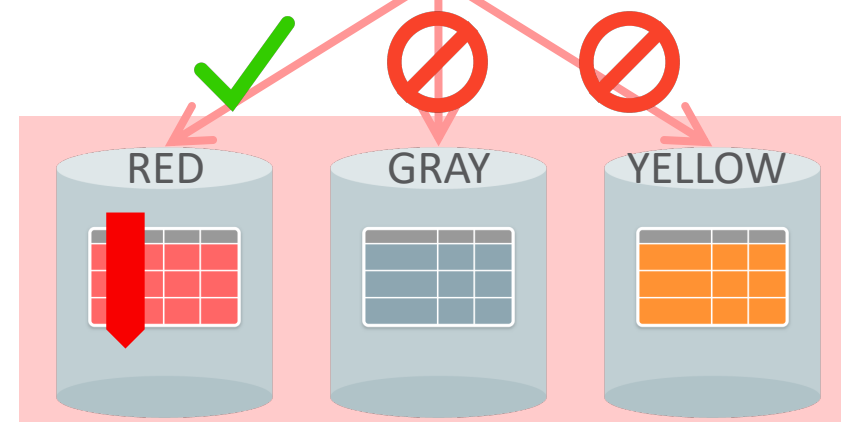


必要なデータ（RED）のみ□
を読み取り、最低限の□
フィルタリングを実現□

全データを読み取り、□
DBサーバーのCPUを□
使ってフィルタリング□



非パーティション表



パーティション表

SQLアクセス・アドバイザでパーティション化を実行

The screenshot shows the Oracle Enterprise Manager Cloud Control 13c interface. The breadcrumb navigation path is: Oracleデータベース > パフォーマンス > SQL > アドバイザ・ホーム. The 'パフォーマンス' (Performance) menu item is highlighted with a red box, and a red arrow points to the 'アドバイザ・ホーム' (Advisor Home) option. The main content area displays the 'Overview' tab for a specific SQL statement (ID: 16777239). It includes a 'General' section with execution details, 'Time & Wait Statistics' bar charts, and 'I/O Statistics' bar charts. The 'Details' section at the bottom shows the 'Plan Statistics' tab with a table of execution statistics.

	Operation	Object	Informat	Est. Rows	Timeline	Executio	Rows	Mem(Ma	Temp(Ma	I/O Requests	Activity %
0	SELECT STATEMENT					1	120				

SQLアドバイザ□を選択□

ORACLE® Enterprise Manager Cloud Control 13c

Enterprise(E) ▼ ターゲット(□) ▼ ★ ▼ ⌚ ▼ ⚙ ▼ 🔍 🔔 SYSMAN ▼ ...

orclm.jp.oracle.com / ↑ ORCLMP ▼ ⓘ 次のユーザーでログイン sys ⓘ jphppt06.jp.oracle.com

Oracleデータベース ▼ パフォーマンス ▼ 可用性 ▼ セキュリティ ▼ スキーマ ▼ 管理 ▼

アドバイザ チェッカ

データの表示 リアル・タイム: 15秒リフレッシュ ▾

アドバイザ

ADDM MTTRアドバイザ **SQLアドバイザ**
SQLパフォーマンス・アナライザ・ホーム Streamsパフォーマンス・アドバイザ
セグメント・アドバイザ データ・リカバリ・アドバイザ
自動UNDO管理 インメモリー・アドバイザ
メモリー・アドバイザ

アドバイザ・タスク デフォルトのパラメータの変更

検索

結果セットに表示されるデータをフィルタ処理するには、アドバイザ・タイプを選択し、オプションでタスク名を入力してください。

アドバイザ・タイプ タスク名 アドバイザ実行 ステータス

すべてのタイプ 過去31日間 すべて 実行

デフォルトでは、検索を行うと、入力した文字列で始まるすべて大文字の一致結果が戻されます。完全一致検索または大文字/小文字を区別する検索を実行するには、検索文字列を二重引用符で囲んでください。二重引用符で囲んだ文字列では、ワイルドカード記号(%)を使用できます。

結果

結果の表示 削除 アクション 再スケジュール ▾ 実行

選択	名前	アドバイザ・タイプ	説明	ユーザー	ステータス	開始時間 ▼	期間(秒)	有効期限の残り(日)
☒	SQLACCESS2220513	SQLアクセス・アドバイザ	SQLアクセス・アドバイザ	SYS	COMPLETED	2019/09/03 18:06:39	55	29
☐	SQLACCESS7185653	SQLアクセス・アドバイザ	SQLアクセス・アドバイザ	SYS	COMPLETED	2019/09/03 17:58:57	91	29
☐	SQLACCESS8247063	SQLアクセス・アドバイザ	SQLアクセス・アドバイザ	SYS	COMPLETED	2019/09/03 17:45:27	116	29
☐	AUTO_STATS_ADVISOR_TASK	Statistics Advisor		SYS	COMPLETED	2019/09/03 14:07:41	18	UNLIMITED
☐	SYS_AUTO_SPCADV605001403092019	セグメント・アドバイザ	Auto Space Advisor	SYS	COMPLETED	2019/09/03 14:00:05	2	29
☐	SYS_AUTO_SPM_EVOLVE_TASK	SPM Evolve Advisor	Automatic SPM Evolve Task	SYS	COMPLETED	2019/09/03 14:00:02	3	UNLIMITED
☐	SQLACCESS8562156	SQLアクセス・アドバイザ	SQLアクセス・アドバイザ	SYS	COMPLETED	2019/09/02 15:34:07	63	28

SQLアクセス・アドバイザーを選択

The screenshot displays the Oracle Enterprise Manager Cloud Control 13c web interface. The top navigation bar includes the Oracle logo, the product name 'Enterprise Manager Cloud Control 13c', and various utility icons. Below this, the breadcrumb trail shows 'orclm.jp.oracle.com / ORCLMP'. The main content area is titled 'SQLアクセス・アドバイザー' (SQL Access Advisor), which is highlighted with a red box. This section describes the tool's function: to evaluate the entire SQL workload and recommend indexes, partitioning, and materialized views to improve performance. Below this, there are links to 'SQLチューニング・アドバイザー' (SQL Tuning Advisor) and 'SQL修復アドバイザー' (SQL Patching Advisor), each with a brief description of their capabilities.

ORACLE® Enterprise Manager Cloud Control 13c

Enterprise(E) ▼ ターゲット 1 ▼

orclm.jp.oracle.com / ↑ ORCLMP ▼

次のユーザーでログイン sys | jhppt06.jp.oracle.com

Oracleデータベース ▼ パフォーマンス ▼ 可用性 ▼ セキュリティ ▼ スキーマ ▼ 管理 ▼

SQLアクセス・アドバイザー

SQLアクセス・アドバイザー SQLのワークロード全体を評価し、SQLワークロードの総合的なパフォーマンスを向上させる索引、パーティショニングおよびマテリアライズド・ビューを推奨します。

SQLチューニング・アドバイザー

SQLチューニング・アドバイザー 個々のSQL文を分析し、SQLのパフォーマンスに対してSQLプロファイル、統計、索引および再構築されたSQLを推奨します。

自動SQLチューニングの結果 監視されている高負荷のSQLにおける、SQLチューニング・アドバイザーの自動実行の結果を表示します。

SQL修復アドバイザー

SQL修復アドバイザーはエラーが発生したSQL文を分析し、場合によってはパッチを適用します。

SQLインシデント分析 SQLインシデント分析は、サポート・ワークベンチ・インシデントを生成しているSQL失敗に対するサポート・ワークベンチから開始されます。
ここをクリックしてサポート・ワークベンチに移動します。

SQL失敗分析 SQL失敗分析はインシデント以外のSQLエラーに対して使用され、SQL詳細またはSQLワークシートのどちらからアクセスできます。
ここをクリックしてSQLワークシートに移動します。

アクセス構造()のみの使用の検証☐を選択☐

ORACLE® Enterprise Manager Cloud Control 13c SYSMAN ▼

SQLアクセス・アドバイザ: 初期オプション
初期オプションのセットを選択します。

☒ アクセス構造(索引、マテリアライズド・ビュー、パーティショニングなど)のみの使用の検証

☐ 新規アクセス構造の推奨
☐ 保存済のタスクまたはテンプレートからの継承オプション

 **ヒント** ウィザードの開始地点を選択しています。すべてのオプションは、ウィザード内から変更可能です。

概要

SQLアクセス・アドバイザは、ワークロード・ソースのSQL文を評価し、ワークロード全体のパフォーマンスを向上させる索引、パーティション、マテリアライズド・ビューおよびマテリアライズド・ビュー・ログを提案します。

取消 続行

「現在と最近のSQLアクティビティ」□

ORACLE® Enterprise Manager Cloud Control 13c SYSMAN ▾

ワークロード・ソース 推奨オプション スケジュール 確認
SQLアクセス・アドバイザー: ワークロード・ソース

データベース orclm.jp.oracle.com [取消] ステップ 1 / 4 **次へ(X)**

ログイン時の権限 SYS

分析に使用するワークロードのソースを選択します。基礎となる表にアクセスするすべてのSQL文を代表するワークロードが、最も適しています。

☒ **現在と最近のSQLアクティビティ**
SQLがキャッシュから選択されます。

☐ 既存のSQLチューニング・セットを使用
SQLチューニング・セット 🔍

☐ 次のスキーマと表から仮想ワークロードを作成
表にディメンションまたは主キー/外部キー制約が含まれていれば、アドバイザにより仮想のワークロードが作成されます。

スキーマと表 [追加]

カンマ区切りリスト

✔ **ヒント** そのスキーマに属するすべての表を指定する場合は、スキーマ名を入力します。

▶ **フィルタ・オプション**

✔ **ヒント** SQL文が多いワークロードの場合は、フィルタ処理を使用して分析時間を短縮することをお勧めします。

[取消] ステップ 1 / 4 [次へ(X)]

推奨するアクセス構造□パーティショニング□

ORACLE® Enterprise Manager Cloud Control 13c SYSMAN ▾

ワークロード・ソース 推奨オプション スケジュール 確認

SQLアクセス・アドバイザー: 推奨オプション

データベース orclm.jp.oracle.com

ログイン時の権限 SYS

[取消] [戻る(K)] ステップ 2 / 4 [次へ(X)]

推奨するアクセス構造

☐ 索引

☐ マテリアライズド・ビュー

☒ パーティショニング

有効範囲

アドバイザは、制限モードまたは包括モードのどちらかで実行できます。制限モードは、特定のしきい値より小さいコストの文を無視し、最もコストの大きい文の処理後に迅速に推奨を返します。包括モードでは、完全な分析が行われます。

☒ 制限
分析は最もコストの大きい文に絞られます

☐ 包括
完全な分析が行われます

▷ 拡張オプション

[取消] [戻る(K)] ステップ 2 / 4 [次へ(X)]

SQLアクセス・アドバイザー: スケジュール即時

ORACLE® Enterprise Manager Cloud Control 13c SYSMAN ▼

ワークロード・ソース 推奨オプション スケジュール 確認

SQLアクセス・アドバイザー: スケジュール

データベース orclm.jp.oracle.com

ログイン時の権限 SYS

取消 戻る(K) ステップ 3 / 次へ(X)

アドバイザー・タスク情報

* タスク名 SQLACCESS5157555

タスクの説明 SQLアクセス・アドバイザー

ジャーナル・レベル 基本

ジャーナル・レベルにより、タスクの実行中にアドバイザーのジャーナルに記録される情報量が制御されます。この情報は、タスクの結果を表示する際に「詳細」タブに表示されます。

* タスクの有効期限(日) 30

バージされる前にこのタスクがデータベース内に保持される日数

* 合計時間の制限(分) 10000

スケジュール・オプション

スケジュール・タイプ 標準

タイムゾーン (UTC-08:00) アメリカ太平洋標準時

繰返し

繰返し 繰返しなし

開始

☒ 即時

☐ 後で

日付 2019/09/03 (例: 2019/09/04)

時間 7 30 00 AM PM

取消 戻る(K) ステップ 3 / 4 次へ(X)

タスク名を確認□

ORACLE® Enterprise Manager Cloud Control 13c SYSMAN ▼

ワークロード・ソース 推奨オプション スケジュール **確認**

SQLアクセス・アドバイザー: 確認

データベース orclm.jp.oracle.com

ログイン時の権限 SYS

取消 SQL表示 戻る(K) ステップ 4 **4 発行**

選択したSQLアクセス・アドバイザーのオプションの値を確認してください。

タスク名 SQLACCESS5157555

タスクの説明 SQLアクセス・アドバイザー

スケジュール開始時間 即時実行

オプション

すべてのオプションの表示

変更済	オプション	値	説明
✓	分析の有効範囲	実表でのパーティショニング	可能な推奨事項のタイプ

取消 SQL表示 戻る(K) ステップ 4 / 4 **発行**

タスクの作成を確認□

The screenshot displays the Oracle Enterprise Manager Cloud Control 13c web interface. At the top, the header includes the Oracle logo, the text "Enterprise Manager Cloud Control 13c", and navigation links for "Enterprise(E)", "ターゲット(T)", and "SYSMAN". Below the header, the breadcrumb trail shows "orclm.jp.oracle.com / ↑ ORCLMP". A navigation menu contains links for "Oracleデータベース", "パフォーマンス", "可用性", "セキュリティ", "スキーマ", and "管理". A yellow confirmation banner is visible, containing a document icon, the word "確認", and the text "SQLアクセス・アドバイザのタスクSQLACCESS5157555は正常に作成されました。" followed by a link "ジョブ詳細を表示". Below the banner, the "アドバイザ・セントラル" section is active, with tabs for "アドバイザ" and "チェッカ". The "アドバイザ" tab shows a list of advisors: ADDM, SQLパフォーマンス・アナライザ・ホーム, セグメント・アドバイザ, and 自動UNDO管理. The "MTTRアドバイザ" section lists "Streamsパフォーマンス・アドバイザ" and "データ・リカバリ・アドバイザ". The "SQLアドバイザ" section lists "インメモリー・アドバイザ" and "メモリー・アドバイザ". At the bottom right, there is a link "デフォルトのパラメータの変更".

タスク実行中 **RUNNING**

ORACLE Enterprise Manager Cloud Control 13c

Enterprise(E) ▼ ターゲット(T) ▼

orclm.jp.oracle.com / ORCLMP

Oracleデータベース ▼ パフォーマンス ▼ 可用性 ▼ セキュリティ ▼ スキーマ ▼ 管理 ▼

アドバイザ チェッカ

データの表示 リアル・タイム: 15秒リフレッシュ

アドバイザ

ADDM MTTRアドバイザ SQLアドバイザ

SQLパフォーマンス・アナライザ・ホーム Streamsパフォーマンス・アドバイザ インメモリー・アドバイザ

セグメント・アドバイザ データ・リカバリ・アドバイザ メモリー・アドバイザ

自動UNDO管理

アドバイザ・タスク

デフォルトのパラメータの変更

検索

結果セットに表示されるデータをフィルタ処理するには、アドバイザ・タイプを選択し、オプションでタスク名を入力してください。

アドバイザ・タイプ タスク名 アドバイザ実行 ステータス

すべてのタイプ 最後の実行 すべて 実行

デフォルトでは、検索を行うと、入力した文字列で始まるすべて大文字の一致結果が戻されます。完全一致検索または大文字/小文字を区別する検索を実行するには、検索文字列を二重引用符で囲んでください。二重引用符で囲んだ文字列では、ワイルドカード記号(%)を使用できません。

結果

結果の表示 削除 アクション 再スケジュール 実行

選択	名前	アドバイザ・タイプ	説明	ユーザー	ステータス	開始時間	期間(秒)	有効期限の残り(日)
☒	SQLACCESS5157555	SQLアクセス・アドバイザ	SQLアクセス・アドバイザ	SYS	RUNNING	2019/09/04 11:16:28		30
☐	SQLACCESS2220513	SQLアクセス・アドバイザ	SQLアクセス・アドバイザ	SYS	COMPLETED	2019/09/03 18:06:39	55	29
☐	SQLACCESS7185653	SQLアクセス・アドバイザ	SQLアクセス・アドバイザ	SYS	COMPLETED	2019/09/03 17:58:57	91	29
☐	SQLACCESS8247063	SQLアクセス・アドバイザ	SQLアクセス・アドバイザ	SYS	COMPLETED	2019/09/03 17:45:27	116	29
☐	SQLACCESS8562156	SQLアクセス・アドバイザ	SQLアクセス・アドバイザ	SYS	COMPLETED	2019/09/02 15:34:07	63	28
☐	SQLACCESS1539572	SQLアクセス・アドバイザ	SQLアクセス・アドバイザ	SYS	COMPLETED	2019/09/02 15:29:16	65	28
☐	SQLACCESS5198388	SQLアクセス・アドバイザ	SQLアクセス・アドバイザ	DBSNMP	COMPLETED	2019/09/02 13:44:26	2	28
☐	SYS_AUTO_INDEX_TASK	SQLアクセス・アドバイザ		SYS	CREATED			UNLIMITED

タスク完了 COMPLETED

ORACLE Enterprise Manager Cloud Control 13c

Enterprise(E) ▼ ターゲット(I) ▼ ★ ▼ ⌚ ▼ ⚙ ▼ 🔍 🔔 SYSMAN ▼ ...

orclm.jp.oracle.com / ↑ ORCLMP ▼ ⓘ 次のユーザーでログイン sys ⓘ jphppt06.jp.oracle.com

Oracleデータベース ▼ パフォーマンス ▼ 可用性 ▼ セキュリティ ▼ スキーマ ▼ 管理 ▼

アドバイザ チェッカ

データの表示 リアル・タイム: 15秒リフレッシュ ▼

アドバイザ

ADDM MTTRアドバイザ SQLアドバイザ
SQLパフォーマンス・アナライザ・ホーム Streamsパフォーマンス・アドバイザ インメモリー・アドバイザ
セグメント・アドバイザ データ・リカバリ・アドバイザ メモリー・アドバイザ
自動UNDO管理

アドバイザ・タスク デフォルトのパラメータの変更

検索

結果セットに表示されるデータをフィルタ処理するには、アドバイザ・タイプを選択し、オプションでタスク名を入力してください。

アドバイザ・タイプ タスク名 アドバイザ実行 ステータス

すべてのタイプ ▼ 最後の実行 ▼ すべて ▼ 実行

デフォルトでは、検索を行うと、入力した文字列で始まるすべて大文字の一致結果が戻されます。完全一致検索または大文字/小文字を区別する検索を実行するには、検索文字列を二重引用符で囲んでください。二重引用符で囲んだ文字列では、ワイルドカード記号(%)を使用できません。

結果

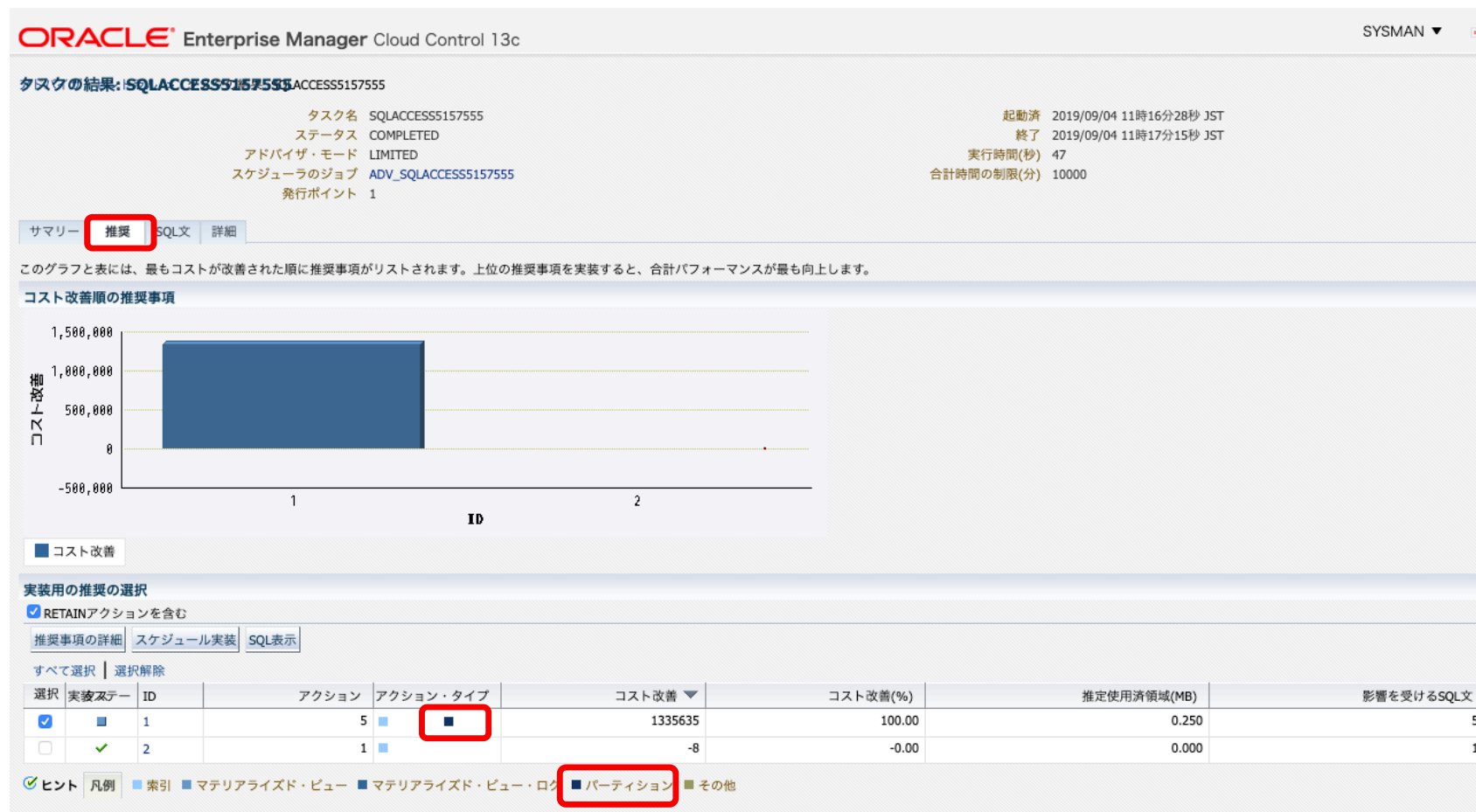
結果の表示 削除 アクション 再スケジュール ▼ 実行

選択	名前	アドバイザ・タイプ	説明	ユーザー	ステータス	開始時間 ▼	期間(秒)	有効期限の残り(日)
☒	SQLACCESS5157555	SQLアクセス・アドバイザ	SQLアクセス・アドバイザ	SYS	COMPLETED	2019/09/04 11:16:28	47	30
☐	SQLACCESS2220513	SQLアクセス・アドバイザ	SQLアクセス・アドバイザ	SYS	COMPLETED	2019/09/03 18:06:39	55	29
☐	SQLACCESS7185653	SQLアクセス・アドバイザ	SQLアクセス・アドバイザ	SYS	COMPLETED	2019/09/03 17:58:57	91	29
☐	SQLACCESS8247063	SQLアクセス・アドバイザ	SQLアクセス・アドバイザ	SYS	COMPLETED	2019/09/03 17:45:27	116	29
☐	SQLACCESS8562156	SQLアクセス・アドバイザ	SQLアクセス・アドバイザ	SYS	COMPLETED	2019/09/02 15:34:07	63	28
☐	SQLACCESS1539572	SQLアクセス・アドバイザ	SQLアクセス・アドバイザ	SYS	COMPLETED	2019/09/02 15:29:16	65	28
☐	SQLACCESS5198388	SQLアクセス・アドバイザ	SQLアクセス・アドバイザ	DBSNMP	COMPLETED	2019/09/02 13:44:26	2	28
☐	SYS_AUTO_INDEX_TASK	SQLアクセス・アドバイザ		SYS	CREATED			UNLIMITED

タスクの結果□サマリー□



タスクの結果概要



タスクの結果□詳細□

ORACLE Enterprise Manager Cloud Control 13c SYSMAN ▼

推奨事項の詳細 ラル > タスクの結果: SQLACCESS5157555 > 推奨事項の詳細

SQLアクセス・アドバイザは、デフォルトのオブジェクト名を生成し、タスクの作成中に指定されたデフォルトのスキーマと表領域を使用しますが、これらは変更できます。読み取り専用として表示されている名前や依存名を編集すると、適宜更新されます。「表領域」フィールドが空白の場合は、スキーマのデフォルト表領域が使用されます。「OK」をクリックすると、SQLスクリプトが変更されますが、推奨ページがSQL文ページで「スケジュール実装」を選択するまで実際には実行されません。

アクション

実装ステータス	推奨ID	アクション	オブジェクト名	オブジェクトの属	索引付けされた	元表	スキーマ	表領域	パーティション・キ	SQLパーティション	推定使用済領域(MB)
■	1	PARTITION_TABLE	SALES				SH		("TIME_ID")	PARTITION BY RANGE ("TIME_ID") INTERVAL(...)	0.0
✓	1	RETAIN_INDEX	TIMES_NEW_PK	BTREE	TIME_ID	SH.TIMES	SH	表領域			0.1
✓	1	RETAIN_INDEX	PRODUCTS_PK	BTREE	PROD_ID	SH.PRODUCTS	SH				0.1
✓	1	RETAIN_INDEX	PRODUCTS_PROD_CAT_IX	BTREE	PROD_CATEGORY	SH.PRODUCTS	SH				0.1
✓	1	RETAIN_INDEX	CHANNELS_PK	BTREE	CHANNEL_ID	SH.CHANNELS	SH				0.1

推奨の影響を受けるSQL

文ID	文	推奨ID	元のコスト	新規コスト	コスト改善	コスト改善(%)	実行数
11485	WITH /*+MONITOR */ SACOMMON1340 AS (select sum(T220.AMOUNT_SOLD) as c1, sum(T220.QUANTITY_SOLD) as c2, T147.CHANNEL_CLASS as c3, T228.CALENDAR_QUARTER_DESC as c4, T228.CALENDAR_YEAR as c5, T185.PROD_CATEGORY as c6 from CHANNELS T147, PRODU...	1	1102695	292394	810301	73.48	1
11485	WITH /*+MONITOR */ SACOMMON1340 AS (select sum(T220.AMOUNT_SOLD) as c1, sum(T220.QUANTITY_SOLD) as c2, T147.CHANNEL_CLASS as c3, T228.CALENDAR_QUARTER_DESC as c4, T228.CALENDAR_YEAR as c5, T185.PROD_CATEGORY as c6 from CHANNELS T147, PRODU...	1	1102695	292394	810301	73.48	1
11485	WITH /*+MONITOR */ SACOMMON1340 AS (select sum(T220.AMOUNT_SOLD) as c1, sum(T220.QUANTITY_SOLD) as c2, T147.CHANNEL_CLASS as c3, T228.CALENDAR_QUARTER_DESC as c4, T228.CALENDAR_YEAR as c5, T185.PROD_CATEGORY as c6 from CHANNELS T147, PRODU...	1	1102695	292394	810301	73.48	1
11485	WITH /*+MONITOR */ SACOMMON1340 AS (select sum(T220.AMOUNT_SOLD) as c1, sum(T220.QUANTITY_SOLD) as c2, T147.CHANNEL_CLASS as c3, T228.CALENDAR_QUARTER_DESC as c4, T228.CALENDAR_YEAR as c5, T185.PROD_CATEGORY as c6 from CHANNELS T147, PRODU...	1	1102695	292394	810301	73.48	2
11485	WITH /*+MONITOR */ SACOMMON1340 AS (select sum(T220.AMOUNT_SOLD) as c1, sum(T220.QUANTITY_SOLD) as c2, T147.CHANNEL_CLASS as c3, T228.CALENDAR_QUARTER_DESC as c4, T228.CALENDAR_YEAR as c5, T185.PROD_CATEGORY as c6 from CHANNELS T147, PRODU...	1	1102695	292394	810301	73.48	2
11485	WITH /*+MONITOR */ SACOMMON1340 AS (select sum(T220.AMOUNT_SOLD) as c1, sum(T220.QUANTITY_SOLD) as c2, T147.CHANNEL_CLASS as c3, T228.CALENDAR_QUARTER_DESC as c4, T228.CALENDAR_YEAR as c5, T185.PROD_CATEGORY as c6 from CHANNELS T147, PRODU...	1	1102695	292394	810301	73.48	1
11485	WITH /*+MONITOR */ SACOMMON1340 AS (select sum(T220.AMOUNT_SOLD) as c1, sum(T220.QUANTITY_SOLD) as c2, T147.CHANNEL_CLASS as c3, T228.CALENDAR_QUARTER_DESC as c4, T228.CALENDAR_YEAR as c5, T185.PROD_CATEGORY as c6 from CHANNELS T147, PRODU...	1	1102695	292394	810301	73.48	1
11485	WITH /*+MONITOR */ SACOMMON1340 AS (select sum(T220.AMOUNT_SOLD) as c1, sum(T220.QUANTITY_SOLD) as c2, T147.CHANNEL_CLASS as c3, T228.CALENDAR_QUARTER_DESC as c4, T228.CALENDAR_YEAR as c5, T185.PROD_CATEGORY as c6 from CHANNELS T147, PRODU...	1	1102695	292394	810301	73.48	1
12350	WITH /*+MONITOR */ DUMMY_SALES AS (select * from (select 0 from CHANNELS) D1, sales D2), SACOMMON1340 AS (select sum(T220.AMOUNT_SOLD) as c1, sum(T220.QUANTITY_SOLD) as c2, T147.CHANNEL_CLASS as c3, T228.CALENDAR_QUARTER_DESC as c4, T228.CALENDAR_YEAR as...	1	40162	587796	-547634	-1,363.56	2

前へ 1-10 / 13 次へ 3行

取消 OK

- SALES表のレンジ・パーティション化を推奨□

タスクの結果 □SQL文□(パーティション化) □

```
ORACLE Enterprise Manager Cloud Control 13c
SYSMAN
SQL表示 > セントラル > タスクの結果: SQLACCESS5157555 > 推奨事項の詳細 > SQL表示
終了

Rem
Rem 表 "SH"."SALES"の再パーティション
Rem

SET SERVEROUTPUT ON
SET ECHO ON

Rem
Rem パーティション化された新規の表を作成中です
Rem
CREATE TABLE "SH"."SALES1"
(
  "PROD_ID" NUMBER,
  "CUST_ID" NUMBER,
  "TIME_ID" DATE,
  "CHANNEL_ID" NUMBER,
  "PROMO_ID" NUMBER,
  "QUANTITY_SOLD" NUMBER(10,2),
  "AMOUNT_SOLD" NUMBER(10,2),
  "DUMMY1" CHAR(100),
  "DUMMY2" CHAR(110)
) SEGMENT CREATION IMMEDIATE
PCTFREE 10 PCTUSED 40 INITRANS 1 MAXTRANS 255
NOCOMPRESS LOGGING
STORAGE(INITIAL 65536 NEXT 1048576 MINEXTENTS 1 MAXEXTENTS 2147483645
PCTINCREASE 0 FREELISTS 1 FREELIST GROUPS 1
BUFFER_POOL DEFAULT FLASH_CACHE DEFAULT CELL_FLASH_CACHE DEFAULT)
TABLESPACE "USERS"
PARTITION BY RANGE ("TIME_ID") INTERVAL( NUMTOYMINTERVAL( 1, 'MONTH')) ( PARTITION
VALUES LESS THAN (TO_DATE(' 2013-01-01 00:00:00', 'YYYY-MM-DD HH24:MI:SS'
, 'NLS_CALENDAR=GREGORIAN')));

Rem
Rem パーティション化された新規の表に制約をコピー中です
Rem
ALTER TABLE "SH"."SALES1" MODIFY ("PROD_ID" NOT NULL ENABLE);
ALTER TABLE "SH"."SALES1" MODIFY ("CUST_ID" NOT NULL ENABLE);
ALTER TABLE "SH"."SALES1" MODIFY ("TIME_ID" NOT NULL ENABLE);
ALTER TABLE "SH"."SALES1" MODIFY ("CHANNEL_ID" NOT NULL ENABLE);
ALTER TABLE "SH"."SALES1" MODIFY ("PROMO_ID" NOT NULL ENABLE);
ALTER TABLE "SH"."SALES1" MODIFY ("QUANTITY_SOLD" NOT NULL ENABLE);
ALTER TABLE "SH"."SALES1" MODIFY ("AMOUNT_SOLD" NOT NULL ENABLE);
```

```
ORACLE Enterprise Manager Cloud Control 13c
SYSMAN
TABLESPACE "USERS"
PARTITION BY RANGE ("TIME_ID") INTERVAL( NUMTOYMINTERVAL( 1, 'MONTH')) ( PARTITION
VALUES LESS THAN (TO_DATE(' 2013-01-01 00:00:00', 'YYYY-MM-DD HH24:MI:SS'
, 'NLS_CALENDAR=GREGORIAN')));

Rem
Rem パーティション化された新規の表に制約をコピー中です
Rem
ALTER TABLE "SH"."SALES1" MODIFY ("PROD_ID" NOT NULL ENABLE);
ALTER TABLE "SH"."SALES1" MODIFY ("CUST_ID" NOT NULL ENABLE);
ALTER TABLE "SH"."SALES1" MODIFY ("TIME_ID" NOT NULL ENABLE);
ALTER TABLE "SH"."SALES1" MODIFY ("CHANNEL_ID" NOT NULL ENABLE);
ALTER TABLE "SH"."SALES1" MODIFY ("PROMO_ID" NOT NULL ENABLE);
ALTER TABLE "SH"."SALES1" MODIFY ("QUANTITY_SOLD" NOT NULL ENABLE);
ALTER TABLE "SH"."SALES1" MODIFY ("AMOUNT_SOLD" NOT NULL ENABLE);

Rem
Rem 元の表のデータを含む新しいパーティション表を移入しています
Rem
INSERT /*+ APPEND */ INTO "SH"."SALES1"
SELECT * FROM "SH"."SALES";
COMMIT;

begin
dbms_stats.gather_table_stats('SH', 'SALES1', NULL, dbms_stats.auto_sample_size);
end;
/

Rem
Rem パーティション化された新規の表に元の表の名前を付与するため、表名を変更中です
Rem
ALTER TABLE "SH"."SALES" RENAME TO "SALES11";
ALTER TABLE "SH"."SALES1" RENAME TO "SALES";

Rem
Rem パーティション化された新規の表で使用するマテリアライズドビューを再検証中です
Rem
BEGIN DBMS_MVIEW.REFRESH('SH"."FWEEK_PSCAT_SALES_MV', 'C', atomic_refresh => FALSE); END;
/

終了
```

スキーマ→データベースオブジェクト→表

SALES表のパーティション化を確認

ORACLE Enterprise Manager Cloud Control 13c

Enterprise(E) ターゲット 1 SYSMAN

orclm.jp.oracle.com / ORCLMP

Oracleデータベース パフォーマンス 可用性 セキュリティ スキーマ 管理

表

オブジェクト・タイプ 表

検索

結果セットに表示されるデータをフィルタ処理するには、オブジェクト・タイプを選択し、オプションでスキーマ名とオブジェクト名を入力してください。

スキーマ SH

オブジェクト名

実行

デフォルトでは、検索を行うと、入力した文字列で始まるすべて大文字の一致結果が戻されます。完全一致検索または大文字/小文字を区別する検索を実行するには、検索文字列を二重引用符で囲んでください。二重引用符で囲んだ文字列では、ワイルドカード記号(%)を使用できます。

選択モード 単一

作成

編集 表示 オプションを指定して削除 アクション 類似作成 実行

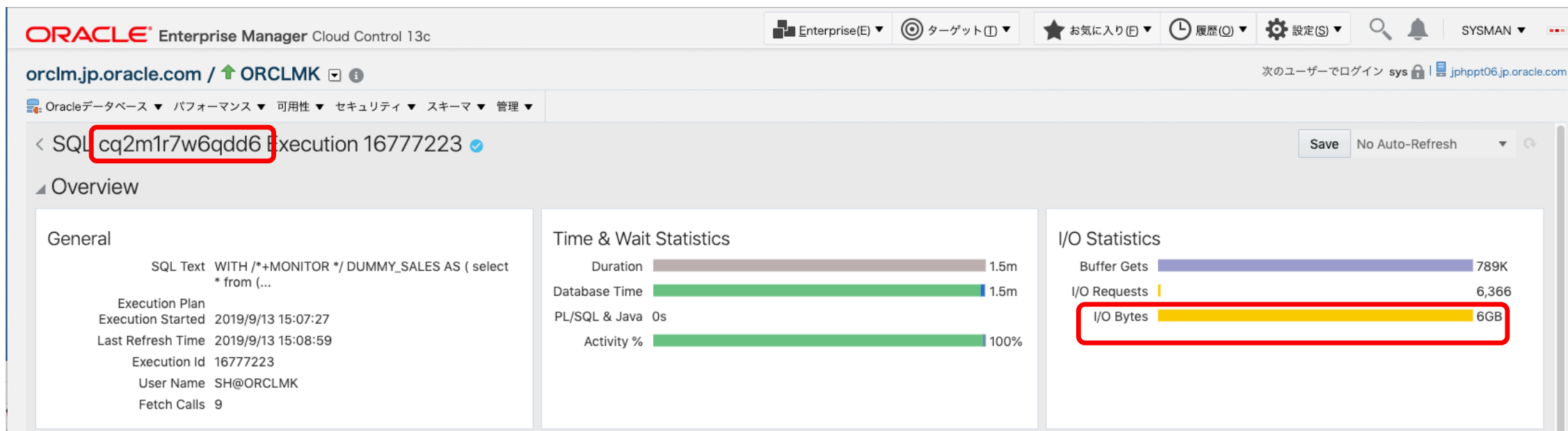
選択	スキーマ	表名	表領域	パーティション化	行	最後の分析
☒	SH	CAL_MONTH_SALES_MV	USERS	NO	48	2019/08/28 16:34:24 JST
☐	SH	CHANNELS	USERS	NO	5	2019/08/29 14:00:49 JST
☐	SH	COSTS		YES	82112	2019/08/28 16:34:25 JST
☐	SH	COUNTRIES	USERS	NO	23	2019/08/28 16:34:27 JST
☐	SH	CUSTOMERS	USERS	NO	55500	2019/08/28 16:34:27 JST
☐	SH	DR\$SUP_TEXT_IDX\$I	USERS	NO		
☐	SH	DR\$SUP_TEXT_IDX\$K	USERS	NO		
☐	SH	DR\$SUP_TEXT_IDX\$N	USERS	NO		
☐	SH	DR\$SUP_TEXT_IDX\$U	USERS	NO		
☐	SH	FWEEK_PSCAT_SALES_MV	USERS	NO	11215	2019/09/02 17:45:27 JST
☐	SH	PRODUCTS	USERS	NO	72	2019/08/28 16:34:27 JST
☐	SH	PROMOTIONS	USERS	NO	503	2019/08/28 16:34:27 JST
☐	SH	SALES		YES	117611904	2019/09/03 14:00:29 JST
☐	SH	SALES11	USERS	NO	117611904	2019/08/29 14:00:50 JST
☐	SH	SALES_ORG		YES	918843	2019/08/28 16:37:08 JST
☐	SH	SALES_TRANSACTIONS_EXT		NO	916039	2019/08/28 16:37:16 JST
☐	SH	SUPPLEMENTARY_DEMOGRAPHICS	USERS	NO	4500	2019/08/28 16:37:17 JST
☐	SH	TIMES		YES	1826	2019/08/29 14:00:49 JST
☐	SH	TIMES_ORG	USERS	NO	1826	2019/08/28 16:37:20 JST

編集 表示 オプションを指定して削除 アクション 類似作成 実行

CPUバウンドなSQL

パーティション化の効果

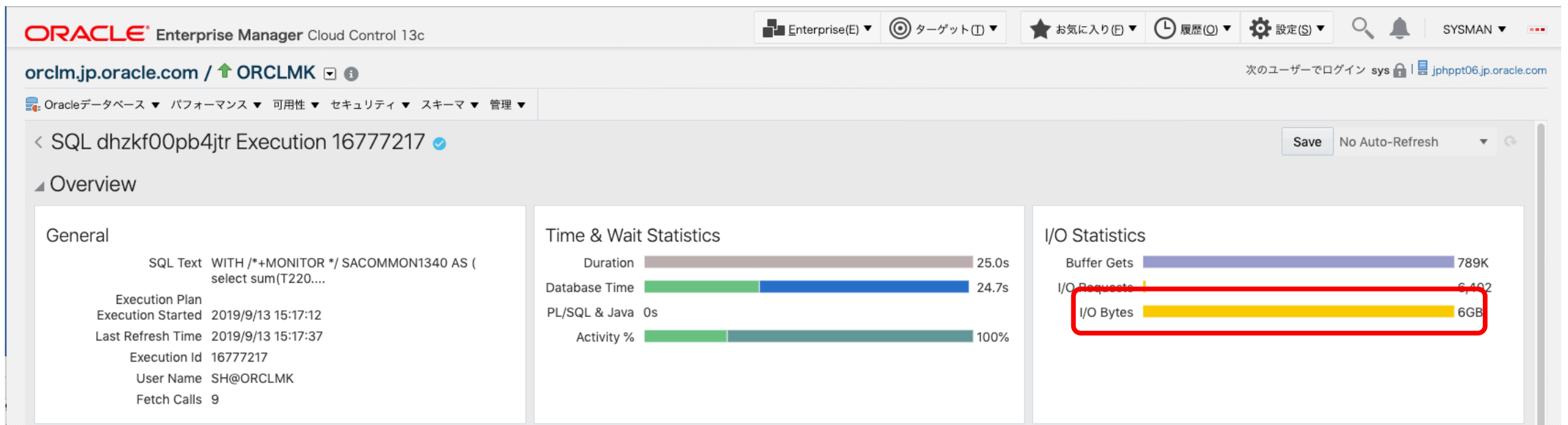
- SQL実行時間 (Duration): 2.0m(118秒) → 1.5m(91秒)へ改善□
 - IO Bytes: 31GB → 6GB へ大幅に削減し、データ読み込み時間は短縮□
 - 元々CPU時間が占める割合が高い(86%) 為、パーティション化の効果が高くない□



I/OバウンドなSQL

パーティション化の効果

- SQL実行時間 (Duration): 1.6m(97秒) → 24.7s(25秒)へ改善□
 - IO Bytes: 31GB → 6GB へ大幅に削減し、データ読み込み時間が短縮□
 - 元々User I/O時間が占める割合が高い(元々79%)為、パーティション化の効果高い□



対象SQLで期待されるパーティション・プルーニングの効果□

答え合わせ

- 今回のSALES表は、2013年～2016年の4年間分のデータを保持□
- しかし、対象の2つのSQL文では、2013年の1年間分のみが集計対象□

```
WITH /*+MONITOR */
SACOMMON1340 AS
(  select sum(T220.AMOUNT_SOLD) as c1,
      .....
      from CHANNELS T147,
           SALES T220,
      .....
      where ( T220.TIME_ID < to_date('2014/01/01','YYYY/MM/DD')
             and T147.CHANNEL_ID = T220.CHANNEL_ID
             and T185.PROD_ID = T220.PROD_ID)
      group by T147.CHANNEL_CLASS, .....
```


オンラインでパーティション表への変換□

Oracle Database 12c Release 2 ~

- 非パーティション表をパーティション表へ変換する”modify句”を提供□
さらに、”online”キーワードを指定すると、変換中でもDML操作が可能□
 - 以下は、今回のシナリオをSALES表に対して、TIME_ID列をパーティション・キーとした□
時間隔(一か月単位)レンジ・パーティションへとオンラインで変換する例□

```
SQL> alter table SALES
      modify partition by range (TIME_ID)
      interval (numtoyminterval(1, 'MONTH'))
      (partition values less than
        (to_date('2013-01-01 00:00:00',
                  'YYYY-MM-DD HH24:MI:SS'))) online
      update indexes;
```

Live Challenge!!

SQLパフォーマンスの高速化の限界を目指そう！

...

	Normal	+ Partitioning
--	--------	----------------

CPUバウンドなSQL
実行時間(秒)□

118

91

相対比[

x1

x1.2

I/OバウンドなSQL
実行時間(秒)□

97

25

相対比[

x1

x3.8

チューニングStep

- 1 処理状況の確認□
- 2 パーティション化での処理量削減□
- 3 パラレル化で複数CPUコアの利用□**
- 4 データ圧縮で更なるI/O量の削減□
- 5 インメモリ化での高速化□



Step#3 パラレル化で複数CPUコアの利用口

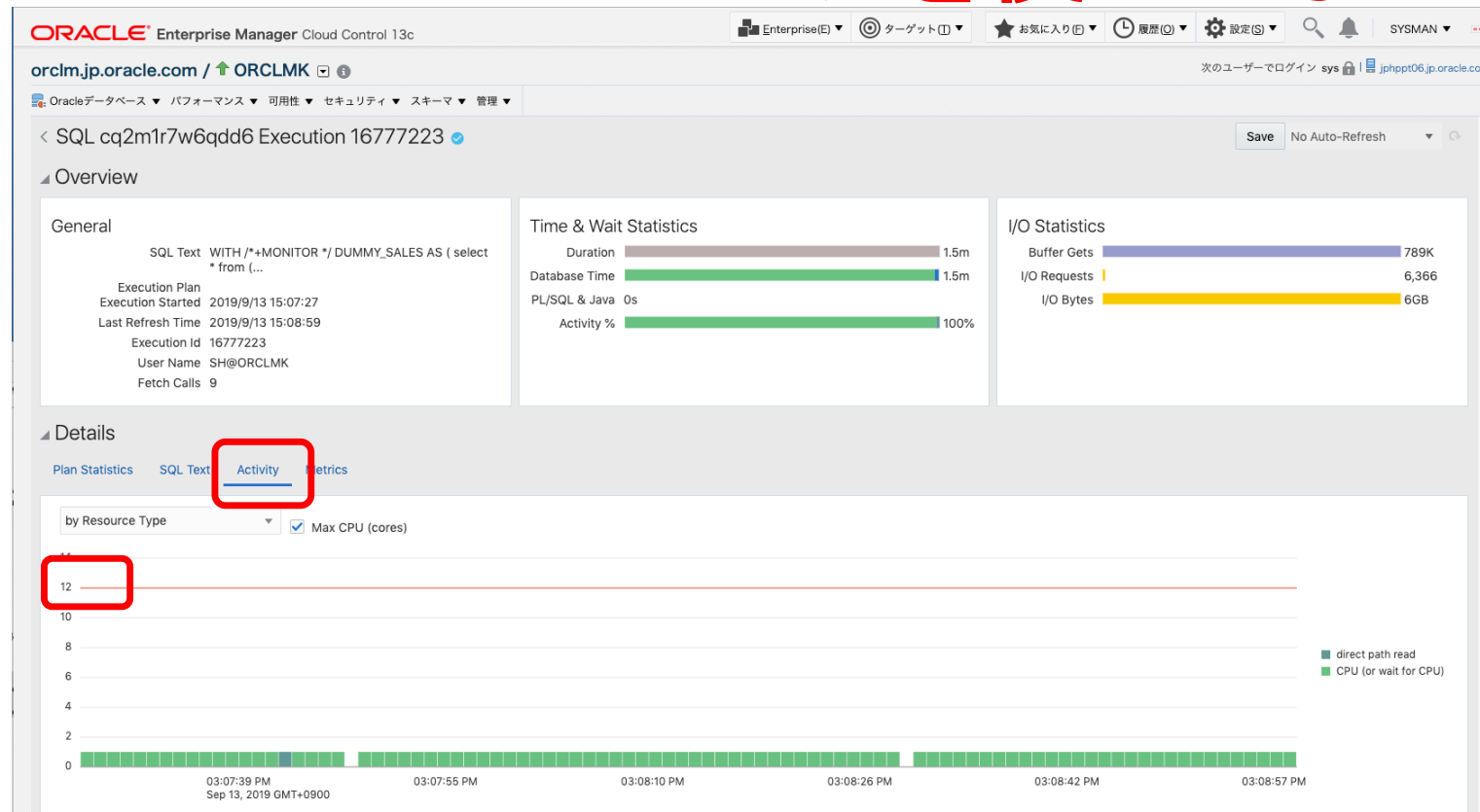
SQLのパラレル実行

自動パラレル度設定 (PARALLEL_DEGREE_POLICY初期化パラメータ)

CPUバウンドなSQLの「Activity」(パーティション化後) □

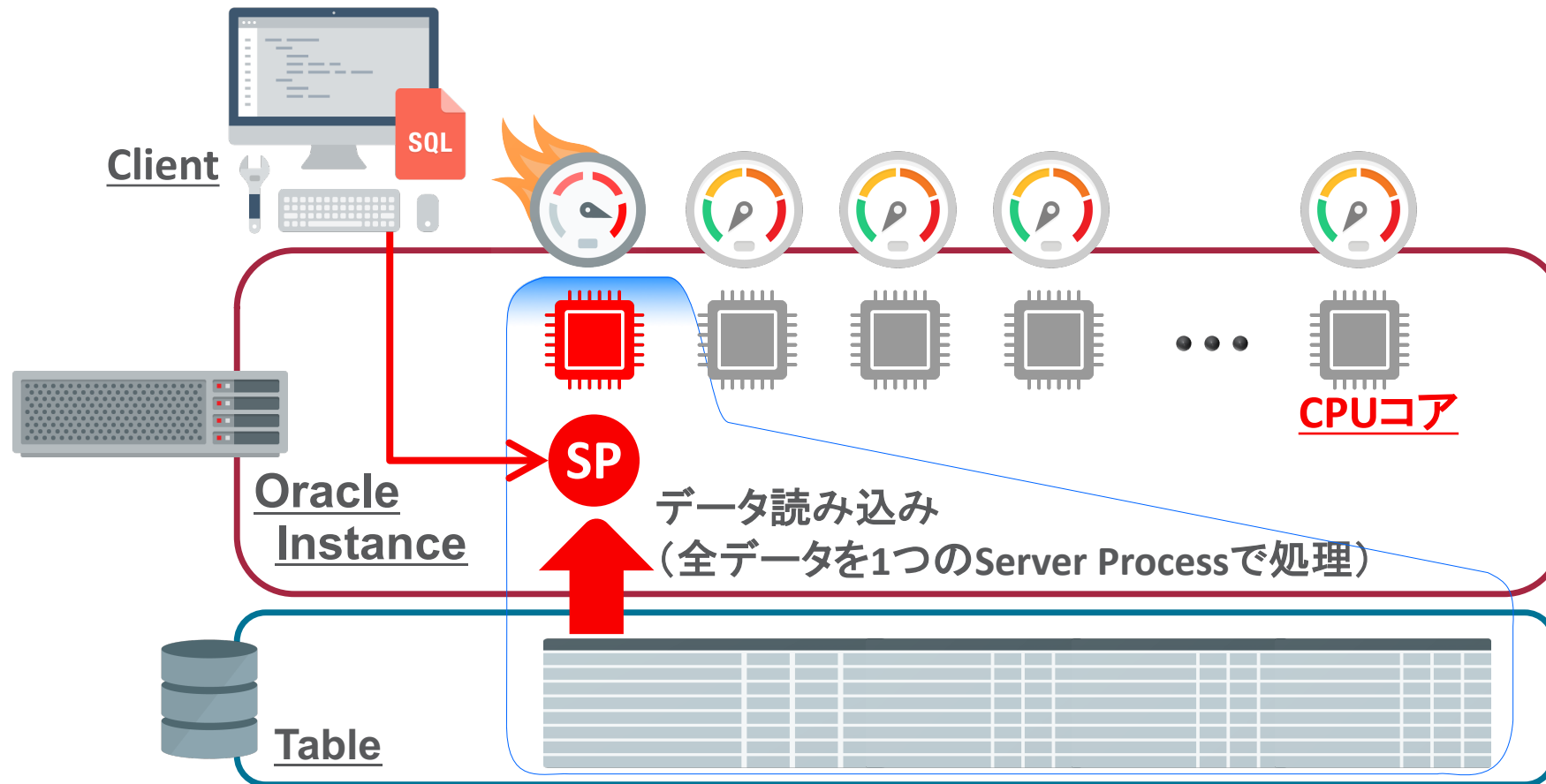
1CPUコアがボトルネックな状況

CPUコアを使いこなせない



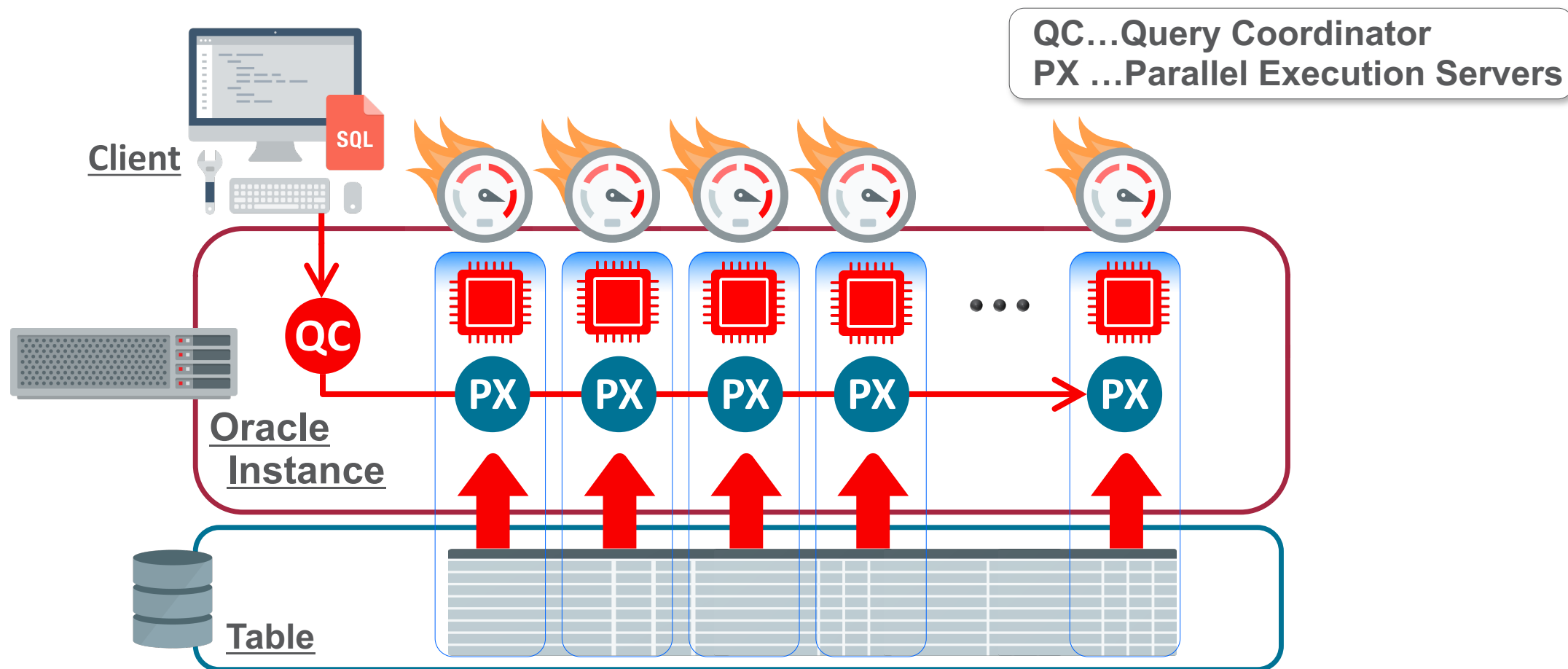
1つのSQLの実行では、1つのCPUコアしか活用できない□

通常の**シリアル実行**の場合



パラレル実行で、マルチコアの有効活用□

1つのSQLを複数のプロセスが自動的に分割実行



[White Paper] Oracle Database 12cでのパラレル実行の基本□

主なパラレル実行の方法□

手動と自動パラレル度設定

- 手動での強制パラレル度設定□

- 対象のSELECT文を実行する前に以下のコマンドを実行(*n*は数字を指定)□

```
SQL> alter session force parallel query parallel n ;
```

- 自動パラレル度設定(自動DOP: Automatic Degree)□

- 初期化パラメータ**PARALLEL_DEGREE_POLICY**で有効化を制御□

- デフォルト(MANUAL)以外の設定値(**LIMITED, AUTO, ADAPTIVE**)へ変更□

- システム・レベルまたは、セッション・レベルで適用可能□

- 詳細は、[White Paper] Oracle Database 18cでのパラレル実行の基本□

- <https://www.oracle.com/technetwork/jp/database/bi-datawarehousing/twp-parallel-execution-fundamentals-133639-ja.pdf>

その他のパラメータ□

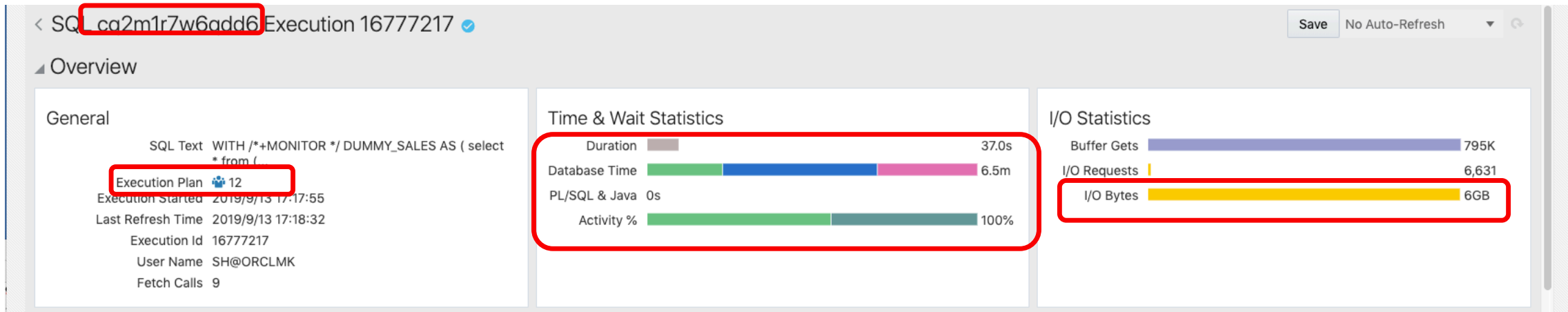
[White Paper] Oracle Database 18cでの□
パラレル実行の基本□からの抜粋□

パラメータ名	デフォルト値	推奨される値	説明
PARALLEL_MIN_SERVERS	CPU_COUNT * PARALLEL_THREAD S_PER_CPU * 2	デフォルト。連続使用 されるPXサーバーの 最小数まで増やすこと を検討してください。	データベース・インスタンスによって常に 割り当てられているパラレル実行サーバー の最小数を定義します。
PARALLEL_MAX_SERVERS	PARALLEL_THREAD S_PER_C PU * CPU_COUNT * concurrent _parallel_users * 5	デフォルト	データベース・インスタンスで割当て可能 なパラレル実行サーバーの最大数を定義し ます。これはハード・リミットであり、こ れを超過することはできません。
PARALLEL_ADAPTIVE_ MULTI_USER	TRUE	FALSE	同時ワークロードに基づいて文の DOPを引き下げます。応答時間を予 測できなくなる場合があります。
PARALLEL_FORCE_LOCAL	FALSE	デフォルト	Oracle RAC環境において、文が発行さ れたノードにパラレル・サーバー・プロ セスを限定するかどうかを制御します。
PARALLEL_MIN_PERCENT	0	デフォルト	パラレル実行に必要なパラレル実行 プロセスとして要求する数の最小割 合です。
PARALLEL_MIN_TIME_ THRESHOLD	AUTO	デフォルト	自動DOPが起動されるまでの文の最小実 行時間。デフォルトは10秒です。
PARALLEL_THREADS_ PER_CPU	2	ハイパースレッドを 有効にしたプラット フォームの場合は1、 それ以外のプラット フォームの場合は2	パラレル実行中にCPUで処理できるパラレ ル・プロセス数です。
PARALLEL_EXECUTION_ME SSAGE_SIZE	16KB	デフォルト。	パラレル・サーバー・プロセス同士および パラレル・サーバー・プロセスとQCとの通 信に使用されるバッファのサイズです。

CPUバウンドなSQL

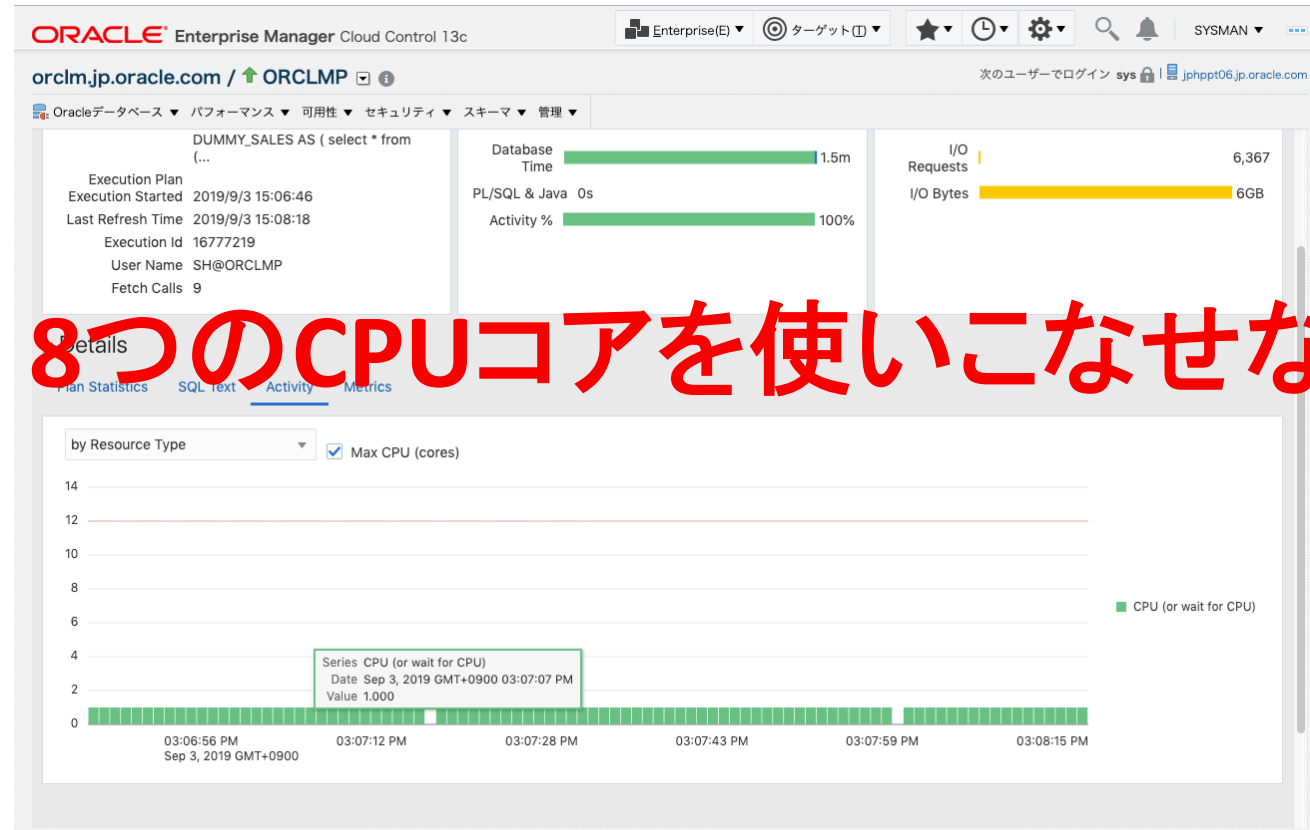
パラレル実行の効果

- SQL実行時間(Duration): 1.5m(92秒) → 37s(37秒)へ改善□
 - 複数プロセスが同時にCPUを使用した為、Database TimeがDurationよりも大きい□
 - 元々CPU時間が占める割合が高かった(99%) 為、パラレル実行の効果が高い(25%に削減)□
 - 逆に□direct path read が増えた□



CPUバウンドなSQLのActivity(パラレル実行の効果)□

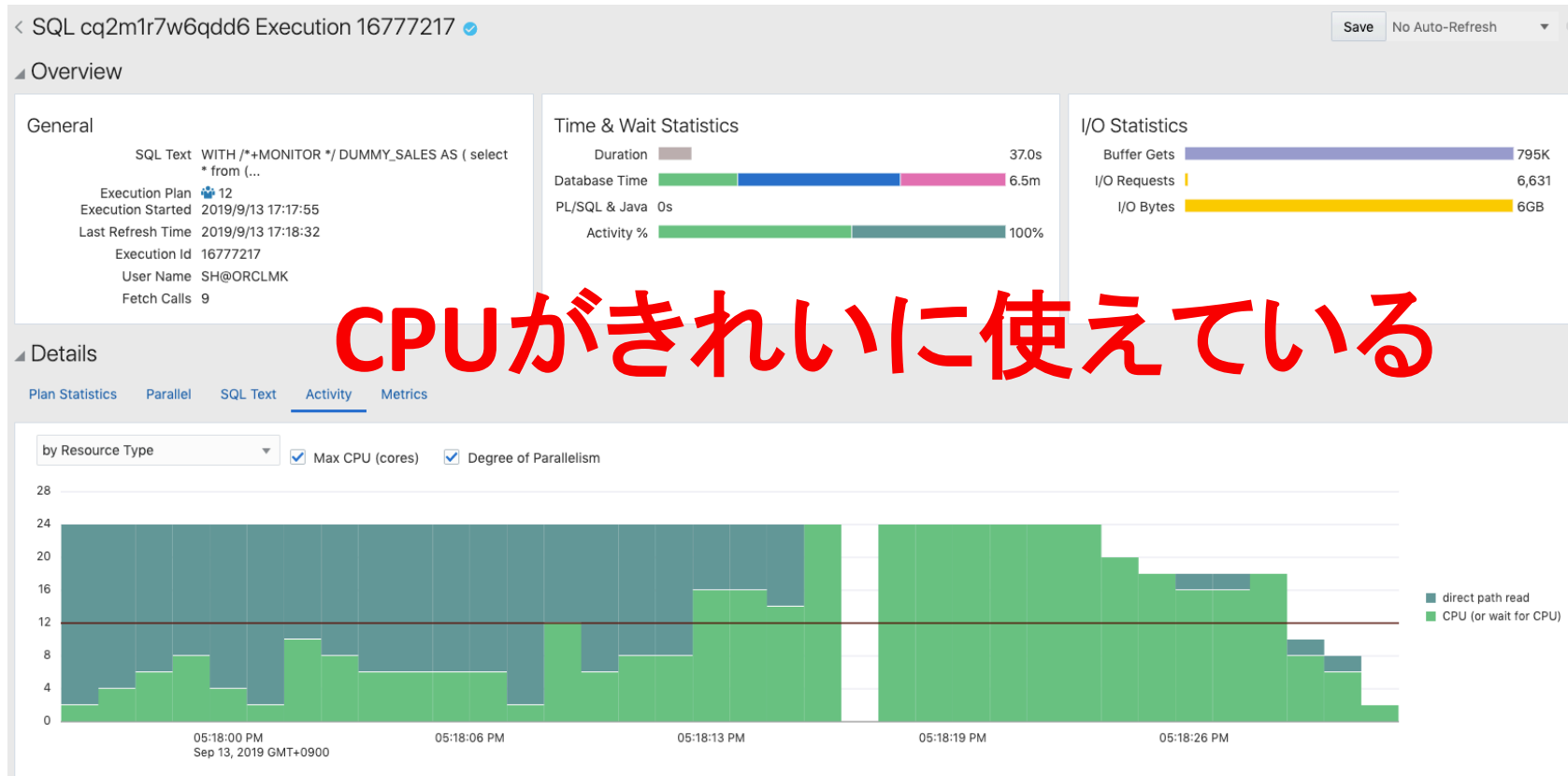
1CPUコアがボトルネック → 複数CPUコアを活用



8つのCPUコアを使いこなせない

CPUバウンドなSQLのActivity (パラレル実行の効果) □

1CPUコアがボトルネック → 複数CPUコアを活用

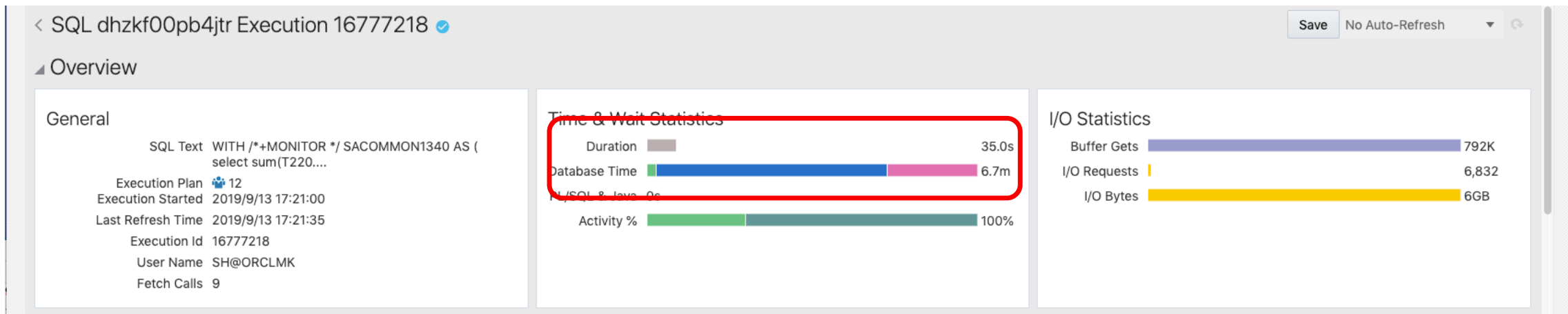


CPUがきれいに使えている

I/OバウンドなSQL

パラレル実行の効果

- SQL実行時間(Duration): 25.0s(25秒) → 34.2s(35秒)で**低速化**□



Live Challenge!!

SQLパフォーマンスの高速化の限界を目指そう！

...

	Normal	+ Partitioning	+ Parallel Query	
CPUバウンドなSQL 実行時間(秒)□	111	92	37	
相対比□	x1	x1.2	x3	
I/OバウンドなSQL 実行時間(秒)□	91	25	35	
相対比□	x1	x3.6	x3	

チューニングStep

- 1 処理状況の確認□
- 2 パーティション化での処理量削減□
- 3 パラレル化で複数CPUコアの利用□
- 4 データ圧縮で更なるI/O量の削減□**
- 5 インメモリ化での高速化□



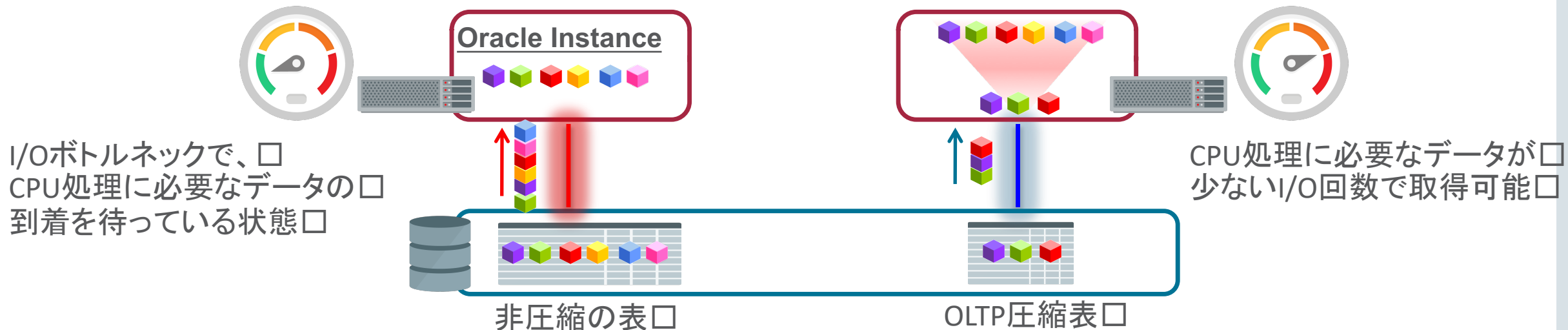
Step#4 データ圧縮で更なるI/O量の削減

Oracle Advanced Compression(表データの圧縮機能)
オンラインで特定パーティションを圧縮変換

表圧縮の主なメリット□

Oracle Databaseでは、いくつかの表圧縮の方法をサポート

- ディスク領域の節約するだけではなく、パフォーマンス観点でもメリット有り□
 - 一つのデータ・ブロック内に格納されるレコード数が増加する為□
 - **総ディスクI/O回数(I/O待機時間)の削減**(1回のI/Oで取得できるレコード数が増える)□
 - **キャッシュ・ヒット率の向上**(圧縮形式でバッファ・キャッシュ上にキャッシュされる)□



表圧縮の方法と特徴

Oracle® Databaseデータベース管理者ガイド19c 20.2.6.1 表圧縮について

表20-1 表圧縮方法

表の圧縮方法	圧縮レベル	CPUオーバーヘッド	アプリケーション	注意
基本表圧縮	高	最小	DSS	なし。
高度な行圧縮	高	最小	OLTP、DSS	なし。
ウェアハウス圧縮(ハイブリッド列圧縮)	より高い	より高い	DSS	圧縮レベルとCPUオーバーヘッドは、指定された圧縮レベル (LOWまたはHIGH)に応じて変化します。
アーカイブ圧縮(ハイブリッド列圧縮)	最高	最高	アーカイブ	圧縮レベルとCPUオーバーヘッドは、指定された圧縮レベル (LOWまたはHIGH)に応じて変化します。

表圧縮の方法と特徴□

Oracle® Databaseデータベース管理者ガイド19c 20.2.6.1 表圧縮について

表20-2 表の圧縮の特徴

表の圧縮方法	CREATE/ALTER TABLEの構文	ダイレクト・パス挿入または配列挿入	注意
基本表圧縮	ROW STORE COMPRESS [BASIC]	行は基本表圧縮方式で圧縮されます。	ROW STORE COMPRESSとROW STORE COMPRESS BASICは同等です。 ダイレクト・パス・インサートまたは配列挿入を使用せずに挿入された行および更新された行は圧縮されません。
高度な行圧縮	ROW STORE COMPRESS ADVANCED	行は高度な行圧縮方式で圧縮されます。	ダイレクト・パス・インサートまたは配列挿入の使用に関係なく、挿入された行と更新された行は高度な行圧縮を使用して圧縮されます。
ウェアハウス圧縮(ハイブリッド列圧縮)	COLUMN STORE COMPRESS FOR QUERY [LOW HIGH]	行はウェアハウス圧縮方式で圧縮されます。	この圧縮方式は高いCPUオーバーヘッドが発生する可能性があります。 更新された行およびダイレクト・パス・インサートまたは配列挿入を使用せずに挿入された行は、列形式ではなく行形式で格納されるため、圧縮レベルが低下します。
アーカイブ圧縮(ハイブリッド列圧縮)	COLUMN STORE COMPRESS FOR ARCHIVE [LOW HIGH]	行はアーカイブ圧縮方式で圧縮されます。	この圧縮方式は高いCPUオーバーヘッドが発生する可能性があります。 更新された行およびダイレクト・パス・インサートまたは配列挿入を使用せずに挿入された行は、列形式ではなく行形式で格納されるため、圧縮レベルが低下します。

圧縮表の作成、特定パーティションの設定変更方法□

- 高度な表圧縮が有効な表の作成例□

- 表を作成するCREATE TABLE文に、圧縮属性を指定するだけ□

- CREATE TABLE ... **ROW STORE COMPRESS ADVANCED**;

- 変更方法□

- 今後INSERTされる新規データのみを圧縮(既存データは非圧縮のまま)□

- ALTER TABLE ... **MODIFY PARTITION** ... COMPRESS ... ;

- 既存も新規データの両方とも圧縮□

- ALTER TABLE ... **MOVE PARTITION** ... COMPRESS ...;

- ※ **変更中にDML処理を受け付け可能なオンラインを選択する場合□**

- 上記のALTER TABLE ... MOVE ... COMPRESS ...文にONLINEオプションを追加□

- 表のオンライン再定義を利用□

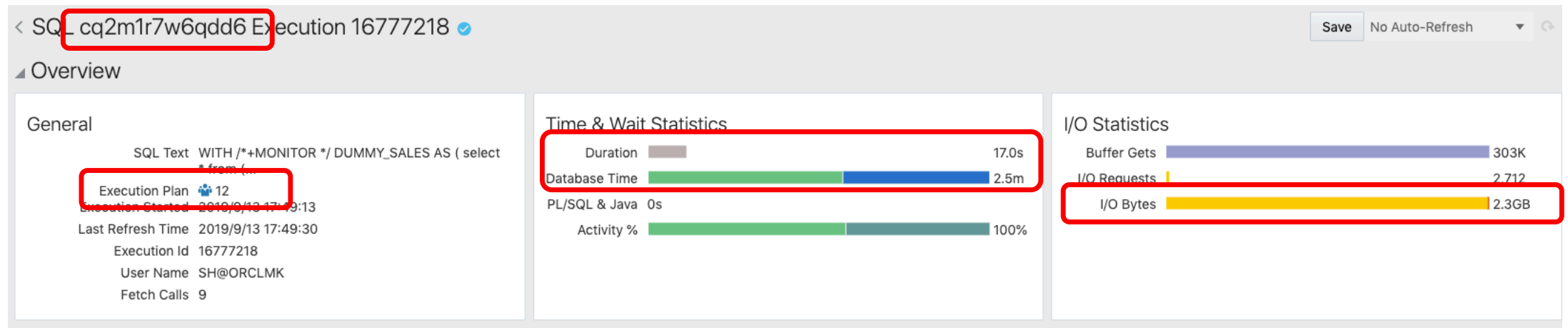
圧縮対象のパーティションの確認方法、圧縮方法 高度な行圧縮

```
SQL> col table_name format a10
SQL> col partition_name format a10
SQL>
select TABLE_NAME, PARTITION_NAME, HIGH_VALUE
  from USER_TAB_PARTITIONS
 where TABLE_NAME = 'SALES'
 order by 1 DESC, 2 ASC ;
TABLE_NAME PARTITION_ HIGH_VALUE
-----
SALES      SYS_P251    TO_DATE(' 2013-01-01 00:00:00', 'SYYYY-MM-DD HH24:MI:SS', 'NLS_CALENDAR=GREGORIA
SALES      SYS_P252    TO_DATE(' 2013-02-01 00:00:00', 'SYYYY-MM-DD HH24:MI:SS', 'NLS_CALENDAR=GREGORIA
SALES      SYS_P253    TO_DATE(' 2013-03-01 00:00:00', 'SYYYY-MM-DD HH24:MI:SS', 'NLS_CALENDAR=GREGORIA
SALES      SYS_P254    TO_DATE(' 2013-04-01 00:00:00', 'SYYYY-MM-DD HH24:MI:SS', 'NLS_CALENDAR=GREGORIA
SALES      SYS_P255    TO_DATE(' 2013-05-01 00:00:00', 'SYYYY-MM-DD HH24:MI:SS', 'NLS_CALENDAR=GREGORIA
SALES      SYS_P256    TO_DATE(' 2013-06-01 00:00:00', 'SYYYY-MM-DD HH24:MI:SS', 'NLS_CALENDAR=GREGORIA
SALES      SYS_P257    TO_DATE(' 2013-07-01 00:00:00', 'SYYYY-MM-DD HH24:MI:SS', 'NLS_CALENDAR=GREGORIA
SALES      SYS_P258    TO_DATE(' 2013-08-01 00:00:00', 'SYYYY-MM-DD HH24:MI:SS', 'NLS_CALENDAR=GREGORIA
SALES      SYS_P259    TO_DATE(' 2013-09-01 00:00:00', 'SYYYY-MM-DD HH24:MI:SS', 'NLS_CALENDAR=GREGORIA
SALES      SYS_P260    TO_DATE(' 2013-10-01 00:00:00', 'SYYYY-MM-DD HH24:MI:SS', 'NLS_CALENDAR=GREGORIA
SALES      SYS_P261    TO_DATE(' 2013-11-01 00:00:00', 'SYYYY-MM-DD HH24:MI:SS', 'NLS_CALENDAR=GREGORIA
SALES      SYS_P262    TO_DATE(' 2013-12-01 00:00:00', 'SYYYY-MM-DD HH24:MI:SS', 'NLS_CALENDAR=GREGORIA
以下略
17:38:13 SQL> alter session force parallel ddl parallel 8 ;
Session altered.
17:38:23 SQL> alter table SALES move partition SYS_P251 row store compress advanced online ;
Table altered.
Elapsed: 00:00:03.91
以下略
```

CPUバウンドなSQL

データ圧縮(高度な行圧縮)の 果

- SQL実行時間(Duration): 37.0s(37秒) → 16.68s(17秒)へ改善□
 - IO Bytes: 6GB → 2.3GB へ大幅に削減し、データ読み込み時間が短縮□
 - パラレル実行により、I/O待機時間の割合が高い状態だったため、圧縮の効果有り□

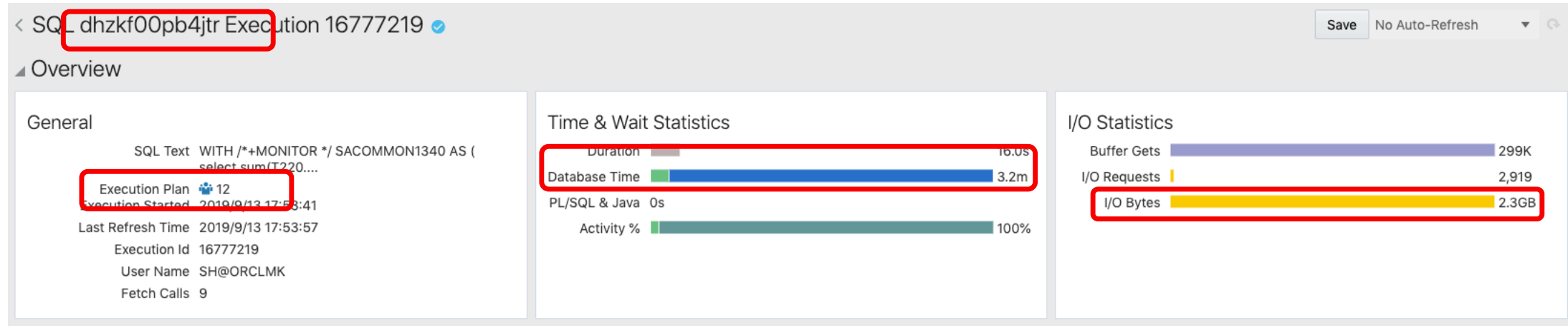


I/O時間の削減で、CPU時間(緑帯)の割合が高まる

I/OバウンドなSQL

データ圧縮(高度な行圧縮)の 果

- SQL実行時間(Duration): 35.0s(35秒) → 16.63s(17秒)へ改善□
 - IO Bytes: 6GB → 2.3GB へ大幅に削減し、データ読み込み時間が短縮□
 - パラレル実行により、I/O待機時間の割合が高い状態だったため、圧縮の効果有り□



まだまだ、I/O時間(青帯)の割合が多い状況

Live Challenge!!

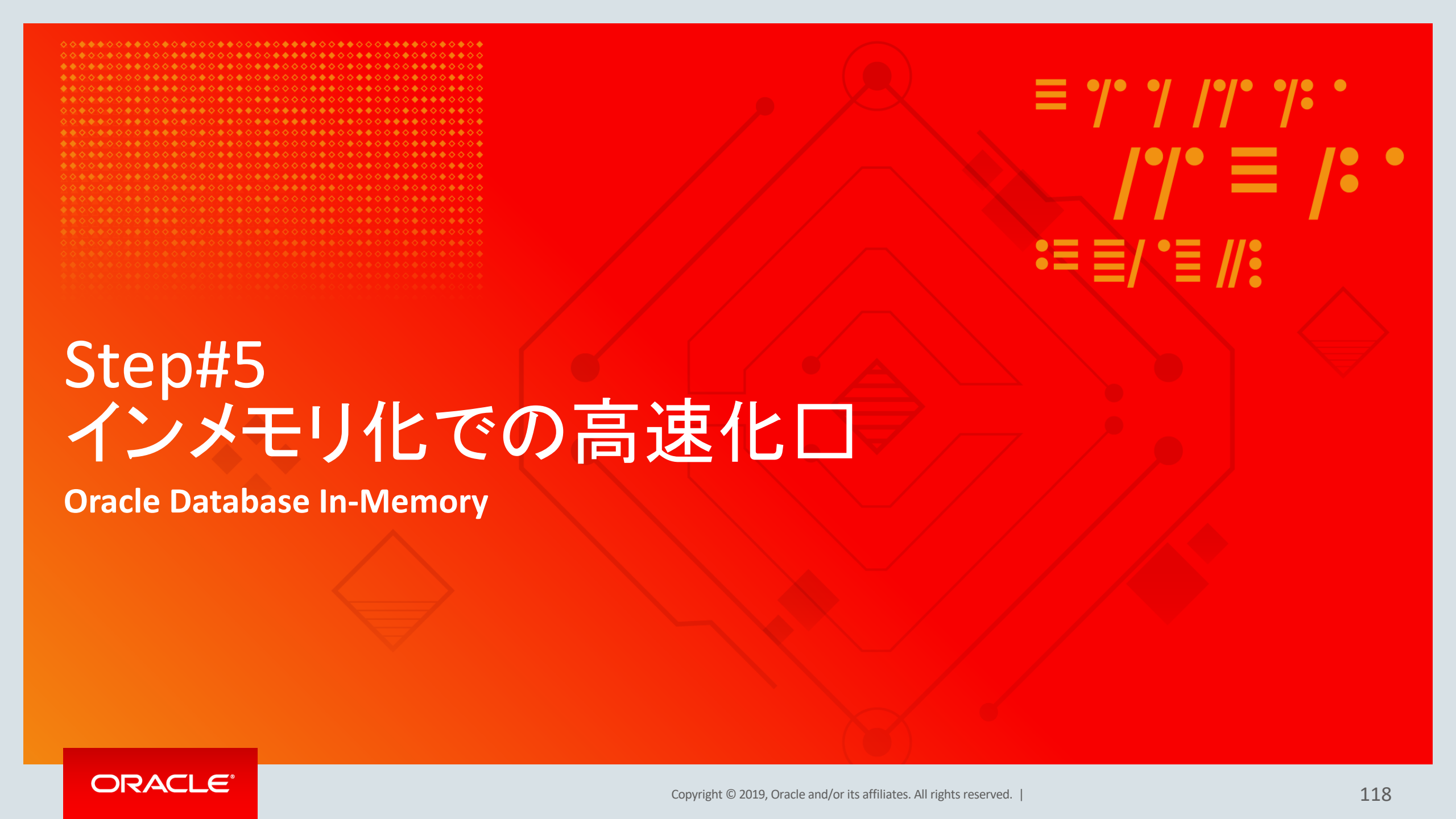
SQLパフォーマンスの高速化の限界を目指そう！

...

	Normal	+ Partitioning	+ Parallel Query	+ Compression	
CPUバウンドなSQL 実行時間(秒)□	111	92	37	17	
相対比□	x1	x1.2	x3	x6.5	
I/OバウンドなSQL 実行時間(秒)□	91	25	35	17	
相対比□	x1	x3.6	x3	x5.3	

チューニングStep

- 1 処理状況の確認□
- 2 パーティション化での処理量削減□
- 3 パラレル化で複数CPUコアの利用□
- 4 データ圧縮で更なるI/O量の削減□
- 5 インメモリ化での高速化□



Step#5 インメモリ化での高速化口

Oracle Database In-Memory

Oracle Database In-Memory

Oracle Database **12c** Release 1 ~

- 一つのデータベースにおいて、**2つのフォーマット**のデータを**メモリ上**で保持□

- ロウ（行）型フォーマット□

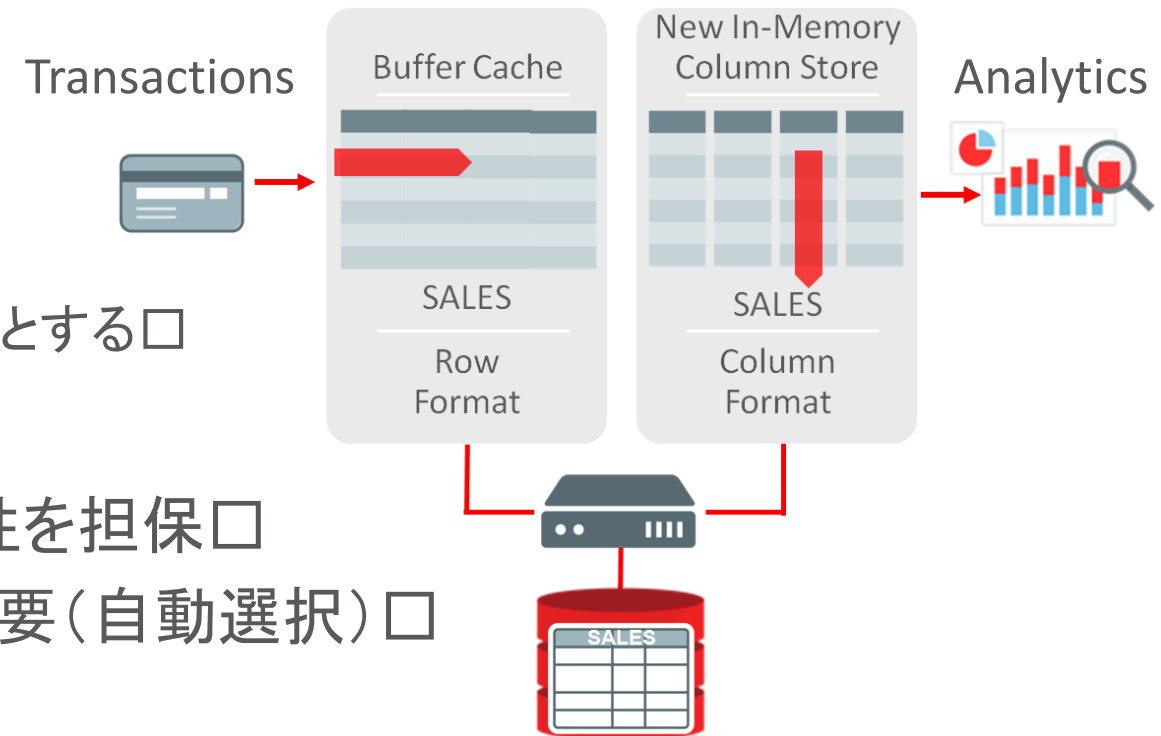
- 注文データの挿入や検索等に代表される、□
オンライン・トランザクション処理を得意とする□

- カラム（列）型フォーマット□

- 売上合計レポート等の少数の列（カラム）と□
多数の行（ロー）を高速演算する分析処理を得意とする□

- 同時利用可能で、トランザクションの一貫性を担保□

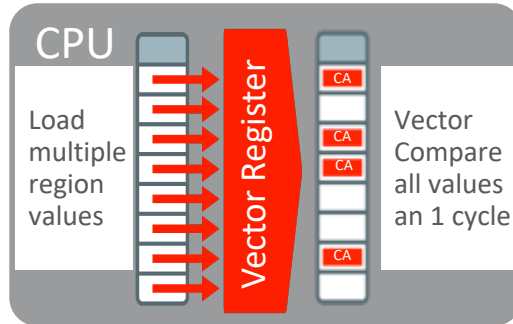
- SQLに制限なし、アプリケーションの改修不要（自動選択）□



Oracle Database In-Memory

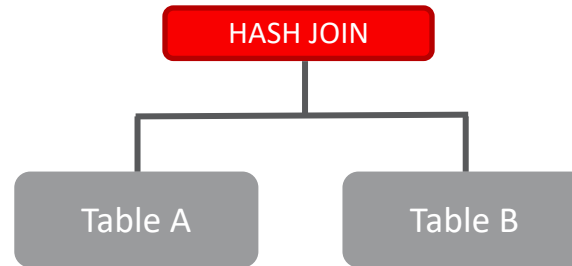
分析クエリーのあらゆる側面を改善

データ・スキャン



- メモリーの速度□
- スキャンとフィルタは必要□
なカラムに限定□
- ベクター・インストラクショ□
ン(CPU命令)□

ジョイン



- スター・ジョインを10倍高速な□
カラム・スキャンに変更□
- 大きな表は、小さな表に一致す□
る値にてスキャン□

インメモリー集計



- インメモリー・レポート・アウト□
ラインが作成され高速なス□
キャンと並行で同時に値が□
集計される□
- レポート作成が高速に□

Oracle Database In-Memory

設定の基本ステップ

1. In-Memory Column Storeのメモリ容量を設定□
 - 初期化パラメータ”`inmemory_size`”を設定□
2. In-Memory Column Store上に保持したい表やパーティションを選択□

```
SQL> -- 本セッションでのサンプル
alter table TIMES inmemory priority high;
alter table PRODUCTS inmemory priority high;
alter table CHANNELS inmemory priority high;
alter table SALES inmemory priority high;
```

3. (オプション) 分析クエリーで使用していた不要索引を削除□

インメモリ化

```
$ sqlplus / as sysdba  
SQL> alter system set inmemory_size=3g scope=spfile ;  
$ shutdown immediate  
$ startup
```

```
17:53:58 SQL> alter table TIMES inmemory priority high;  
alter table PRODUCTS inmemory priority high;  
alter table CHANNELS inmemory priority high;  
alter table SALES inmemory priority high;
```

Table altered.

Elapsed: 00:00:00.07

```
17:58:58 SQL>
```

Table altered.

Elapsed: 00:00:00.02

```
17:58:58 SQL>
```

Table altered.

ORACLE®

Elapsed: 00:00:00.01

インメモリー・セントラル□

orclm.jp.oracle.com / ↑ ORCLMK ⓘ

次のユーザーでログイン sys ⓘ jphppt06.jp.oracle.co

Oracleデータベース ▼ パフォーマンス ▼ 可用性 ▼ セキュリティ ▼ スキーマ ▼ 管理 ▼

ページ・リフレッシュ 2019/09/13 18:01:08 JST ↺

構成

構成

合計SGA (GB) 18.87

インメモリー領域(GB) 3.00

インメモリー問合せ 有効

インメモリー強制 デフォルト

デフォルトのインメモリー句 指定されていません

インメモリー移行サーバー最大数 12

パフォーマンス

CPUアクティビティ(過去5分間) 0.06 アクティブ・セッション

インメモリー・アドバイザー

インメモリー・アドバイザー

オブジェクト・サマリー

インメモリー・オブジェクト統計

すべてのオブジェクトの合計サイズ(GB) 59.04

インメモリー対応サイズ(%) 46.75

圧縮係数 25.01x

インメモリー・ポリシー数 0

優先度	使用中のメモリー (GB)	移行済(%)
高	0.2	18.27
合計	0.2	18.27

インメモリー・オブジェクト分散の表示

インメモリー・オブジェクトの検索

検索 インメモリー・オブジェクト ⓘ ☐ すべてのパーティションがインメモリー使用可能でない場合にセグメントを表示

フィルタ

インメモリー・セントラル

インメモリー・セントラル

Oracle Scheduler

レプリケーション

リソース・マネージャ

SH.SALES

インメモリー・セントラル□

orclm.jp.oracle.com / ↑ ORCLMK ⓘ

Oracleデータベース ▼ パフォーマンス ▼ 可用性 ▼ セキュリティ ▼ スキーマ ▼ 管理 ▼

次のユーザーでログイン sys ⓘ jphppt06.jp.oracle.co

ページ・リフレッシュ 2019/09/13 18:03:59 JST ↺

インメモリー移行サーバー最大数 12

パフォーマンス

CPUアクティビティ (過去5分間) 0.12 アクティブ・セッション

インメモリー・アドバイザー

インメモリー・アドバイザー

▲ オブジェクト・サマリー

インメモリー・オブジェクト統計

すべてのオブジェクトの合計サイズ(GB) 59.04

インメモリー対応サイズ(%) 46.75

圧縮係数 21.63x

インメモリー・ポリシー数 0

優先度	使用中のメモリー (GB)	移行済(%)
高	0.82	64.56
合計	0.82	64.56

インメモリー・オブジェクト分散の表示

▲ インメモリー・オブジェクトの検索

検索 インメモリー・オブジェクト ☐ すべてのパーティションがインメモリー使用可能でない場合にセグメントを表示

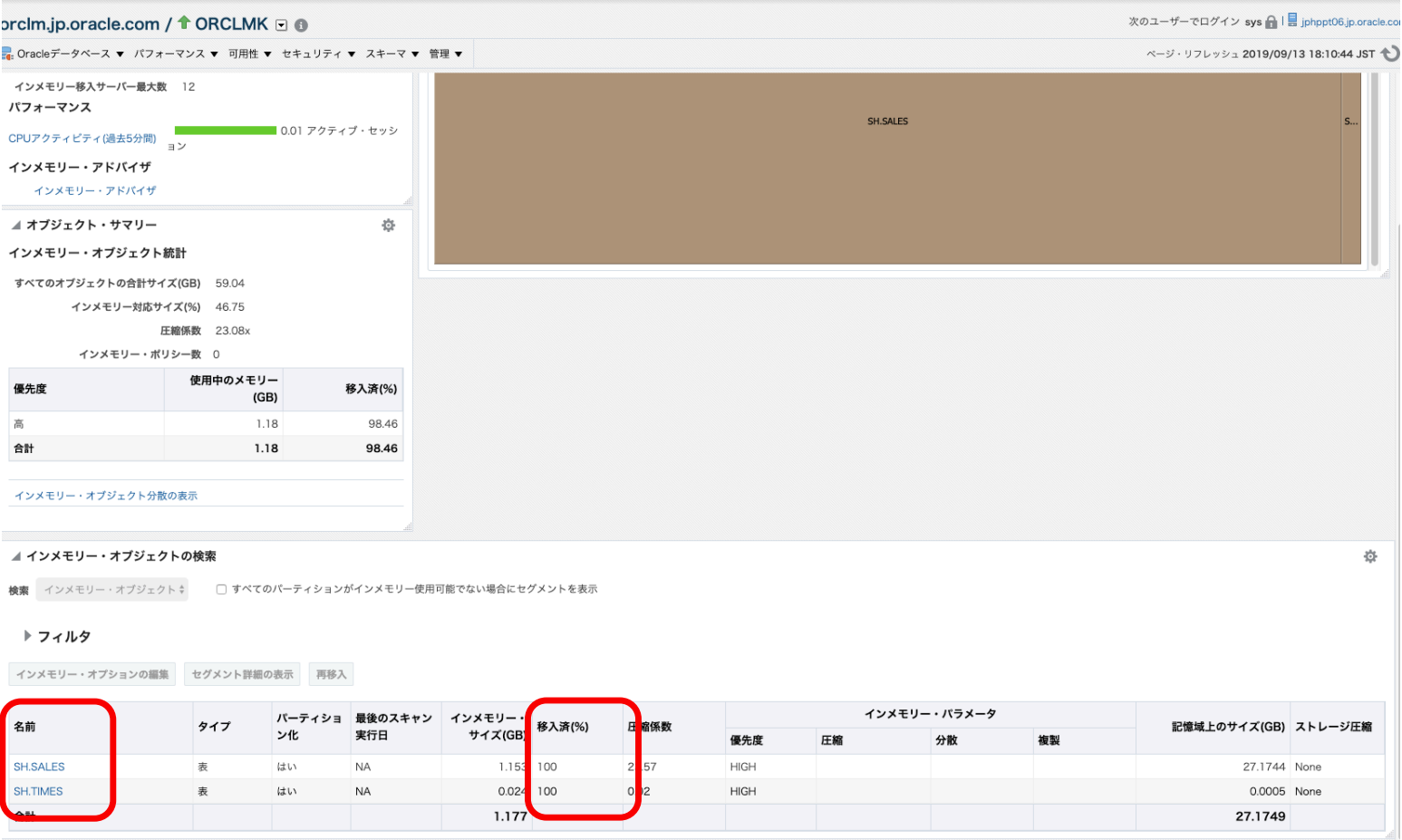
インメモリー・オブジェクトの検索

▼ フィルタ

インメモリー・オプションの編集 セグメント詳細の表示 再移入

名前	タイプ	パーティション化	最後のスキャン実行日	インメモリー・サイズ(GB)	移行済(%)	圧縮係数	インメモリー・パラメータ				記憶域上のサイズ(GB)	ストレージ圧縮
							優先度	圧縮	分散	複製		
SH.SALES	表	はい	NA	0.792	74.995	21.36	HIGH				23.7585	None
SH.TIMES	表	はい	NA	0.024	100	0.12	HIGH				0.0005	None
合計				0.821							23.7591	

インメモリー・セントラル□



補足説明□

(オプション) 分析クエリーで使用していた不要索引を削除

- オプションの記載の背景□

- 過去、分析クエリーの高速化を目的に索引を作成していた場合、Database In-Memoryを活用すること□で、その索引は使用しなくなります□
- オンライン・トランザクション処理で、レコードが挿入されるたびに、□索引のメンテナンス処理が行われています□
- よって、不要な索引を削除することで、オンライン・トランザクション側の処理を減らすことが可能です□

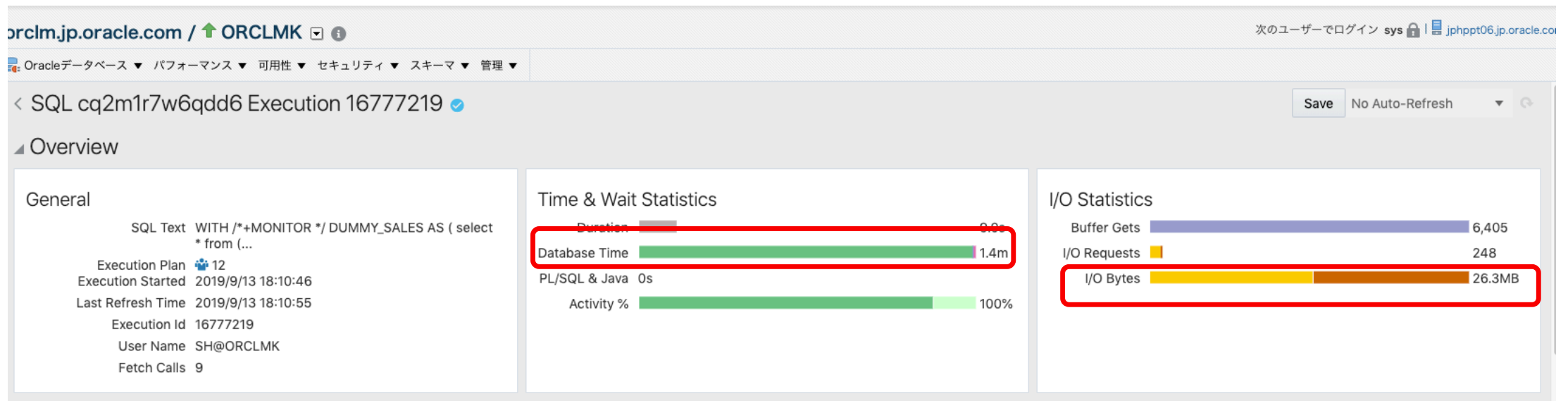
- 索引の使用状況の確認方法□

- (12.2~) ~~DBA~~ **DBA_INDEX_USAGE** ビューと ~~DB~~ **INDEX_USAGE_INFO** ビュー□
- alter index <INDEX_NAME> monitoring usage ; コマンド□
 - 詳細は、「[Oracle® Database 19c 管理者ガイド](#)□1.4.7 索引の使用状況の監視」□

CPUバウンドなSQL

Oracle Database In-Memoryの効果

- SQL実行時間(Duration): 20.0s(20秒) → 9.5s(9秒*)へ改善□
 - IO Bytes: 2.6GB → 26.2MB へ削減し、データ読込み時間がさらに短縮□

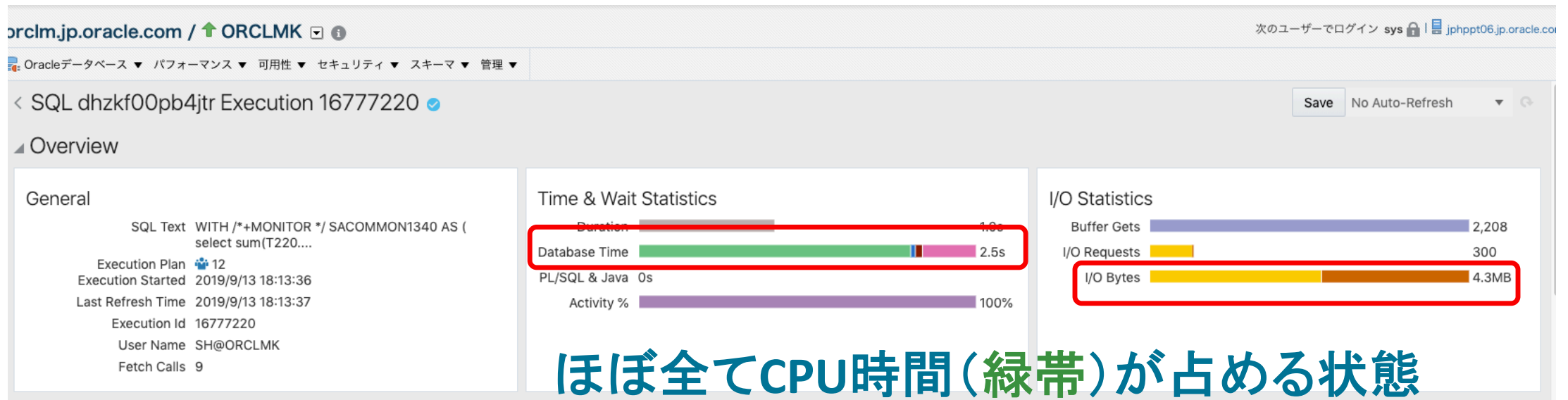


ほぼ全てCPU時間(緑帯)が占める状態

I/OバウンドなSQL

Oracle Database In-Memoryの効果

- SQL実行時間 (Duration): 20.0s(20秒) → 1.0以内(0.82秒*)へ改善□
* SQL*Plusで計測した値□
 - IO Bytes: 2GB → 4.3MB へ大幅に削減し、データ読み込み時間がほぼゼロへ□
 - I/O待機時間の割合が高い状態だったため、DBIMの効果が高い□



Live Challenge!!

SQLパフォーマンスの高速化の限界を目指そう！

...

	Normal	+ Partitioning	+ Parallel Query	+ Compression	+ Database In-Memory
CPUバウンドなSQL 実行時間(秒)□	111	92	37	20	9.50
相対比□	x1	x1.2	x3	x5.5	x11.6
I/OバウンドなSQL 実行時間(秒)□	91	25	35	20	0.82
相対比□	x1	x3.6	x3	x4.55	x110



x110

SQL文を書き換えることなく、
パフォーマンス向上を実現

パフォーマンス・チューニングの基本的な考え方□

超有名な公式と同じ

$$\text{時間} \downarrow = \text{処理量} \downarrow / (\text{速度} * \text{並列度}) \uparrow$$

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, Database In-Memory Column Format/Storage Index, 実行計画の改善, ...
- 高速化□
 - Buffer Cache, **Database In-Memory**, Flash Device, InfiniBand, Exafusion, ...
- 並列化□
 - **Parallel Query**, Multi-Core, RAC, ASM, ...

Oracle Database 19c

Oracle Cloud環境で今すぐ試してみてください！！

- 本セッションでご紹介した機能□
 - パーティション化 (Partitioning) □
 - データアクセス範囲を限定□
 - パラレル化 (Parallel Query) □
 - マルチコアの有効活用□
 - データ圧縮 (Compression) □
 - I/Oボトルネックの改善□
 - インメモリ化 (Database In-Memory) □
 - ディスクI/O時間の排除□
 - インメモリ独自の高速演算□

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®