

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

Discussion Night #12 パフォーマンス・チューニングの 改善方法について語ろう

しばちょう先生 & ゆっきー



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.

皆さんのご質問に答えまくります（希望）

Live Q & A

しばちょう & ゆっきー

統計情報の取得

Q.統計情報等の取得タイミング等知りたい
東京都 A.Iさん

統計情報の取得

Q.統計情報等の取得タイミング等知りたい

A1. AUTO_SAMPLE_SIZE + GATHER_AUTOがお勧め

SAMPLE=10%より精度の高い統計情報が高速に取得

さらに、取得が必要なオブジェクトに限定

A2. STALE_PERCENT

統計情報が失効している表の確認

[しばちょう先生の連載]

[第8回 オプティマイザ統計情報の管理 ～統計収集の高速化を体験してみる～ 2012.07.27](#)

[第9回 オプティマイザ統計情報の管理 ～統計収集の失効を制御してみる～ 2012.08.17](#)

統計情報の効率向上

ESTIMATE_PERCENTパラメータ(AUTO_SAMPLE_SIZE in 11g ~)

- Oracle Database 11gのAUTO_SAMPE_SIZEは、新しいハッシュ・ベースのサンプリング・アルゴリズム
 - ESTIMATE_PERCENTパラメータのデフォルト設定(DBMS_STATS.AUTO_SAMPLE_SIZE)の場合のみ適用されるアルゴリズム

		1% SAMPLE	AUTO_SAMPLE_SIZE	100% SAMPLE
Elapsed Time (sec)		797	1,908	18,772
NDV	L_ORDERKEY	225,000,000	450,000,000	450,000,000
	L_COMMENT	7,244,885	177,499,684	181,122,127

LINEITEM table(230GB) from the TPC-H benchmark with a scale factor of 300
[Reference] Best Practices for Gathering Optimizer Statistics

統計情報の効率向上

STALE_PERCENTパラメータ

- 表の統計情報が古い(失効)していると判断する閾値を指定するパラメータ
 - デフォルト値は10(%)であり、SET_*_PREFSで設定変更可能
 - 次の場合にこのパラメータの値が使用される
 - 自動統計収集ジョブの実行時
 - GATHER_DATABASE/SCHEMA_STATSプロシージャ実行時
 - ただし、OPTIONS='GATHER AUTO' or 'GATHER STALE'が必要
- 統計情報が失効している表の確認方法
 - DBMS_STATS.GATHER_DATABASE/SCHEMA_STATS
 - Doc ID 252597.1
 - USER_TAB_MODIFICATIONSディクショナリ・ビュー

統計情報の効率向上

統計情報が失効している表の確認1

```
SQL> SET SERVEROUTPUT ON
DECLARE
  ObjList dbms_stats.ObjectTab;
BEGIN
  dbms_stats.gather_database_stats(objlist=>ObjList, options=>'LIST STALE');
  -- dbms_stats.gather_schema_stats(ownname=>'SH', objlist=>ObjList, options=>'LIST STALE');
  FOR i in ObjList.FIRST..ObjList.LAST LOOP
    dbms_output.put_line(ObjList(i).ownname || '.' || ObjList(i).ObjName || ' ' ||
                        ObjList(i).ObjType || ' ' || ObjList(i).partname);
  END LOOP;
END;
/

DBSNMP.BSLN_BASELINES TABLE
DBSNMP.BSLN_STATISTICS TABLE
SYS.APPLY$_SOURCE_SCHEMA TABLE
SYS.AQ$_ALERT_QT_S TABLE
SYS.AQ$_QUEUE_TABLE_AFFINITIES TABLE
SYS.AQ$_SUBSCRIBER_TABLE TABLE
.....
```

空の表を確認
→ 「LIST
EMPTY」

失効or空の表を
確認
→ 「LIST
AUTO」

統計情報の効率向上

統計情報が失効している表の確認2 (Doc ID 252597.1)

```
SQL> -- 10万行格納されているSTALETTEST表へ新たに10万行INSERT
insert into STALETTEST select level+100000,'AAA' from dual connect by level <=100000;
commit;
-- Flush
exec dbms_stats.FLUSH_DATABASE_MONITORING_INFO();

select u.TIMESTAMP, t.last_analyzed, u.table_name, u.inserts, u.updates, u.deletes, d.num_rows,
       decode(d.num_rows,0,'Table Stats indicate No Rows',
             nvl(TO_CHAR(((U.inserts+u.deletes+u.updates)/d.num_rows) * 100,'999.99'),
             'Null Value in USER_TAB_MODIFICATIONS')) percent
  from user_tables t,USER_TAB_MODIFICATIONS u,dba_tables d
 where u.table_name = t.table_name
       and d.table_name = t.table_name
       and d.owner = sys_context('USERENV', 'CURRENT_USER')
       and (u.inserts > 10000 or u.updates > 10000 or u.deletes > 10000)
 order by t.last_analyzed;
```

TIMESTAM	LAST_ANA	TABLE_NAME	INSERTS	UPDATES	DELETES	NUM_ROWS	PERCENT
12-05-25	12-05-25	STALETTEST	100000	0	0	100000	100.00

統計情報の取得

Q.統計情報等の取得タイミング等知りたい

A3. INCREMENTAL + GRANULARITY

”APPROX_GLOBAL AND PARTITION”は、一部のパーティションのみの統計を再収集した際、表全体を読み込まずに、表レベルの統計情報を更新

A4. DEGREE と CONCURRENT

多くのPartitionの統計情報を並列で取得できる

注意点) CONCURRENTの設定は、統計情報の取得が高速化すること検証の上、活用してください。

統計情報の効率向上

DEGREEパラメータ

- 統計収集で内部的に実行されるSQL文の平行度を指定するパラメータ
 - デフォルト設定(NULL)では、対象表のDEGREE属性に従う
 - 表のDEGREE属性のデフォルト設定は「1」の為、シリアル実行がデフォルト
 - DBMS_STATS.AUTO_DEGREEを設定した場合
 - 表サイズを基に、自動的に平行度を決定
 - 選択される平行度は、1 ~ DBMS_STATS.DEFAULT_DEGREE
 - $DEFAULT_DEGREE = (PARALLEL_THREADS_PER_CPU \times CPU_COUNT)$
- 各表のサイズを分析して個別設定 (SET_TABLE_PREFS) を行うより、SET_GLOBAL_PREFSでデフォルト設定を変更する方が効率的

統計情報の効率向上

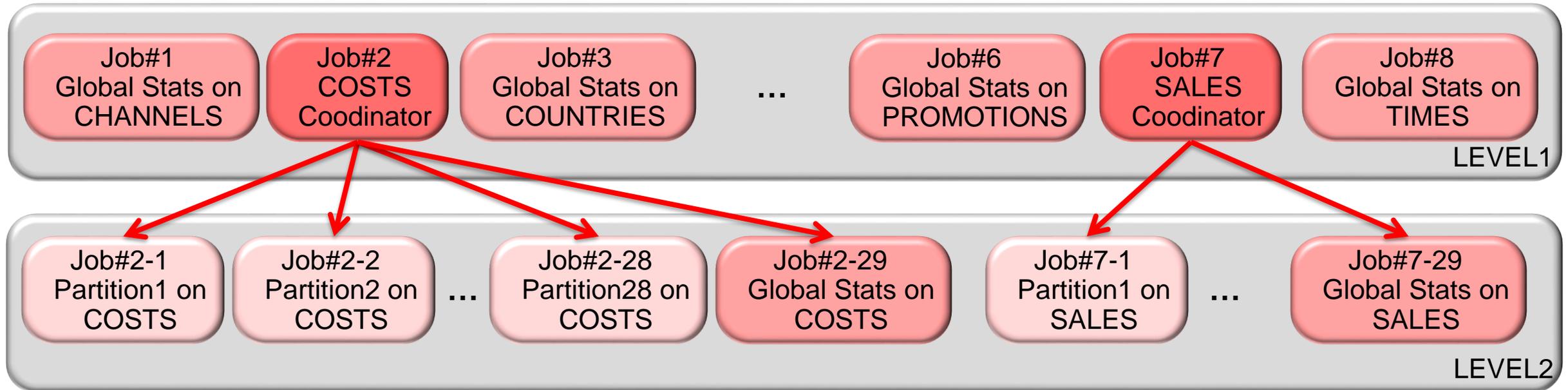
CONCURRENT in Oracle Database 11g Release 2(11.2.0.2)~

- 複数オブジェクトの統計を並列で収集するか否かを指定するパラメータ
 - SET_GLOBAL_PREFSで「TRUE」に設定(デフォルト:FALSE)した場合、複数のオブジェクトの統計を並列で収集し、総所要時間を大幅に短縮することが可能
 - 1つの表の統計をパラレル実行で収集するDEGREEより上位での並列化
 - 特に、多数の小さなサイズの表があるデータベースでは、DEGREEよりもCONCURRENTによる恩恵を受け易い(組合せ使用は可能)
 - 制限
 - 並列可能な数は、JOB_QUEUE_PROCESS初期化パラメータに依存
 - デッドロックの可能性排除の為、並列化が許可されるパーティション表は1つ
 - 自動統計収集ジョブは、この設定に関わらず表毎にシリアルで収集

統計情報の効率向上

CONCURRENTの動作例

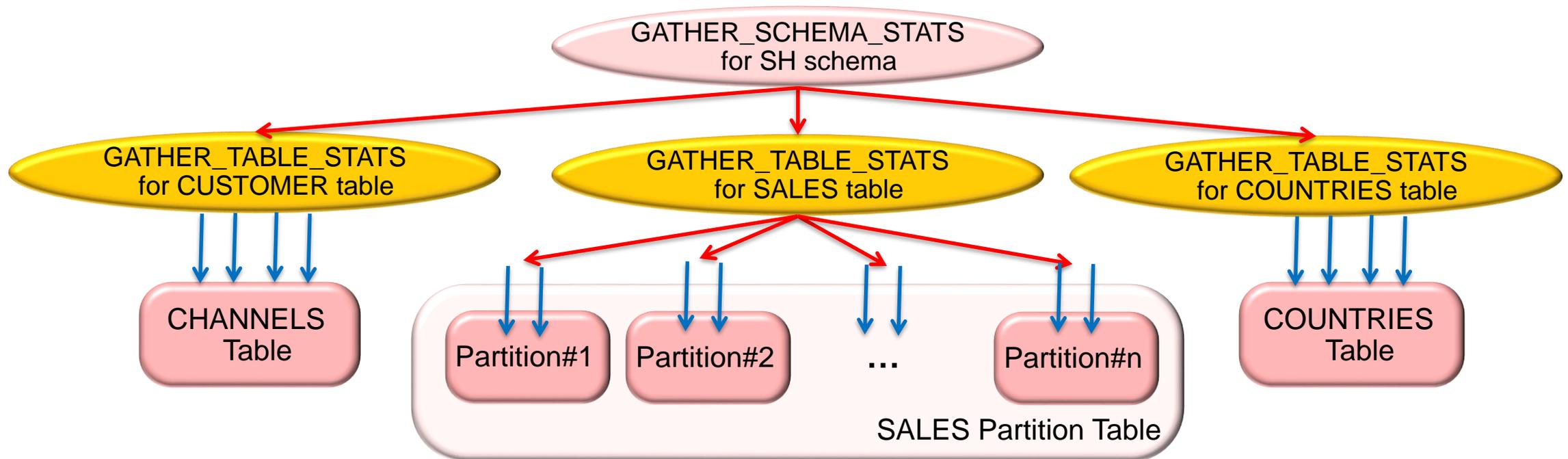
- SHスキーマに対し、DBMS_STATS.GATHER_SCHEMA_STATSを実行
 - 条件: COSTSとSALES表の二つのパーティション表、JOB_QUEUE_PROCESS=32
 - 初期は3つの非パーティション表とCOSTS表のパーティション(28+1)の32オブジェクト
 - COSTS表が統計収集が完了するまで、SALES表の収集は開始しない



統計情報の効率向上

CONCURRENT & DEGREEパラメータ

- 各パラメータが影響する階層が異なる為、共存させることが可能
 - CONCURRENT: 複数の表を同時に統計収集(→)
 - DEGREE: ある一つの表を平行実行で統計収集(→)



実行計画の変動

- Q.実行計画変動によるパフォーマンス劣化を
どう対策したら良いのか知りたい。東京都 Y.Sさん
- Q.実行計画がぶれる cardinality feedback
のような実行計画チューニング機能の動きが
不安。東京都 Y.Sさん(別の方)

実行計画の変動 (Answer)

A1. 適切なタイミングで統計情報を取得

統計情報と実態に乖離が存在しているからなので、前述で紹介した方法で効率的な統計情報の収集をお勧めします

A. PENDING_STATS

新たに取得した統計情報を直ぐに公開せずに保留

その保留統計で重要なSQLの性能が劣化しないことを検証した上で、公開する運用

[第10回 オプティマイザ統計情報の管理 ～保留中の統計情報を有効活用してみる～ 2012.09.27公開](#)

A. SQL Plan Management

SPMについて

Q.SPMの使い方を詳しく知りたい。

A. まずは、DBA & Developer Day 2013のコンサル資料を読んでみてください

<http://www.oracle.com/webfolder/technetwork/jp/ondemand/ddd2013/A-2.pdf>

実行計画の細かい質問：東京都 O.Yさん
(EXPLAIN PLAN と実行された実行計画の差違)

Q.最近のOracleではAdaptive Cursor Sharingの影響か、EXPLAIN PLANで表示された実行計画と実際に使用された実行計画が異なる事が多いです。

特にSQL*PlusのAUTOTRACE機能で表示される実行計画が異なる事があるので不便です。

対策として、SQL*Plusを2つ立ち上げて、1つはAUTOTRACE有効でのデバッグ実行用、もう1つはdbms_xplan.display_cursor表示用としています。

実行計画の細かい質問：東京都 O.Yさん (EXPLAIN PLAN と実行された実行計画の差違)

- A. EXPLAIN PLANはあくまで簡易的に実行計画を確認するツールで正しくない可能性がある
実際にSQLを実行した後に、`dbms_xplan.display_cursor`で確認する実行計画が正しい

参考) しばちょう先生の連載

[第5回 SQLの実行計画からパフォーマンスの違いを読み解く 2012.03.29公開](#)

[第6回 続・SQLの実行計画からパフォーマンスの違いを読み解く 2012.04.26公開](#)

ベストプラクティスについて：神奈川県 M.Mさん

Q. Tablespaceは小さいサイズをたくさんに分割するのと大きいのを少ない数で持つのとどちらがいいのか？

A. SMALLFILE/BIGFILE表領域のメリット・デメリット

- ✓ 性能観点だと、データファイル数は少ない方が良いかもしれないが、経験上、そこまでする差はないはず
- ✓ データファイルのリストア時間を考慮するとSMALLFILEだが、管理性は確かにBigFile
- ✓ BigFileを使うにしても、数百GB程度に抑えておいた方が良いかもしれない
 - ✓ 500GBを超える表は、基本的にはパーティション表化して、パーティション毎に表領域を分ければ良いでしょう。

ベストプラクティスについて

Q. Indexが**180GB**もある。Dataが**100GB**に対して。
これは正常なのか？

A. 起こり得るケースとして、
索引が多い or 索引の断片化が原因と推測できる。
一つの表に数十個~百数十個の索引を作成しているの
かもしれませんね。
または、索引の再作成を実行し、断片化を解消してみ
てはいかがでしょうか。

ロックについて

Q.ロック待機の原因になったSQLを確認する方法は？

A. Active Session Historyで確認可能

他には、マニュアルに記載のある、catblock.sql

https://docs.oracle.com/cd/E60665_01/db112/REFRN/scripts002.htm

-- 事前準備

@?/rdbms/admin/catblock.sql

-- 実行

@?/rdbms/admin/utllockt.sql

=====

UserA

SQL> update emp set ename='Itai-san' where empno=1;
1 row updated.

SQL> update emp set ename='Mizuguchi-san'
where empno=2;
1 row updated.

UserB

SQL> select * from emp where empno=1 for update wait 10;

10秒経過後は、以下のエラーが発生

RROR at line 1:

ORA-30006: resource busy; acquire with WAIT timeout expired

ちなみに、nowaitの場合は、以下のエラーが発生

ERROR at line 1:

ORA-00054: resource busy and acquire with NOWAIT specified
or timeout expired

その待機中に、以下のオペレーションを実行すると...

Sys User

SQL>

set linesize 150

set pagesize 1000

@?/rdbms/admin/utllockt.sql

WAITING_SESSION	LOCK_TYPE	MODE_REQUESTED
MODE_HELD	LOCK_ID1	LOCK_ID2

395	None		
597	Transaction	Exclusive	Exclusive
589841	7436		

ロックについて

Q.ASHの効果的活用法について教えて欲しい

A.【DBA&Developer Days 2014】オラクル・コン
サルが語る！プロフェッショナルのデータベース
性能分析手法 AWR／ASHを活用した分析事例

パフォーマンスチューニングのやり方

基本から

パフォーマンスチューニングのやり方について

基礎編

Q. 性能トラブル発生の原因への
アプローチの仕方を知りたい

Q. パフォーマンスチューニングのポイント解説

Q. パフォーマンスチューニングの判断方法を
知りたい。

→ A. 津島博士: 第17回 チューニングについて

パフォーマンスチューニングのやり方について

問題分析・特定

Q.問題のあるSQLの特定が難しい。

→ A. 津島博士の連載

1. 第32回の「1. SQL統計による問題分析について」
2. 第50回の「2. ライブラリ・キャッシュの待機」
3. 第44回の「2. ダイレクト・リード」

パフォーマンスチューニングのやり方について

SQLチューニング

Q. SQLの改善方法がわからない。

Q. 一般的なSQLのチューニング方法を
教えてほしいです

→ A. 津島博士の連載

1. 第38回 SQLチューニングについて
2. 第49回の「2.Bツリー索引の圧縮」
3. 第55回 オプティマイザ・ヒントについて
4. 第59回の「アウトラインの出力について」

パフォーマンスチューニングのやり方について

全般

TechNight#2(8月) パフォーマンス・チューニングの極意 ～津島博士の明日から使えるSQLチューニング～

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

パフォーマンス・チューニングの極意
津島博士の
明日から使えるSQLチューニング

ORACLE DATABASE 12c
Plug into the Cloud



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

ORACLE

Copyright © 2010 Oracle and/or its affiliates. All rights reserved.

12c オプティマイザ新機能

Q. 12c新機能の適応計画とダイナミックサンプリング機能
効果があるものの一部過度なパフォーマンス劣化を引き起こすケースがある

A0. 12.2では改善されている

A1. 津島博士：第57回 適応問合せ最適化について

A2. Tech Night#7 オラクル・コンサルが語る！

SQL性能を最大限に引き出すDB 12cクエリー・オプティマイザ新機能活用と統計情報運用の戦略

12c 新機能による**高速化**

Q. 12cに更新することにより、過去バージョンより
高速になったSQLのタイプを知りたい。

Q. 12c になり、特に強化されたパフォーマンス向上
に関する機能等

12c 新機能による高速化について

津島博士のパフォーマンス講座 より

- ✓ 第34回 : 「(2) Oracle Database 12cの索引スキャン」
- ✓ 第39回 : 「2.Oracle Database 12c の拡張について」
- ✓ 第42回 : 「1.Oracle Database 12cからの新機能」
- ✓ 第54回 : 「(3)ベクターGroup By (Vector Group By)」
- ✓ 第56回 : 「1.分析ファンクションの平行実行」と
「2. その他の平行実行機能」
- ✓ 第60回 : Oracle Database 12cR2で強化されたSQL処理について
- ✓ 第61回 : Oracle Database 12cR2のOracle Database In-Memoryについて



以上、

Discussion Night #12

パフォーマンス・チューニングの
改善方法について語ろう

でした！

ゆっきーからの宣伝



https://www.youtube.com/watch?v=aCW_f5znwOE

**9月1日より、毎週金曜日に
皆様に価値のあるコンテンツを30秒で宣伝させていただきます!!
よろしくお祈いします！**

～ みなさまの投稿をお待ちしております ～



Twitter

#OracleTechNight

こんな時、かけこむ会社が増えています。



ビジネスプロセスを
改善したい!



今のシステムは
使いにくい!



システムコストを
下げたい!



パフォーマンスを
良くしたい!



経営分析を
したいのだが...



どんなソリューションが
あるの?



見積りはどれくらい
なんだろう?



楽に管理を
したい!

Oracle Digitalは、オラクル製品の導入をご検討いただく際の総合窓口。
電話とインターネットによる直接的なコミュニケーションで、どんなお問い合わせにもすばやく対応します。
もちろん、無償。どんなことでも、ご相談ください。

お問い合わせは電話またはWebフォーム



 **0120-155-096**

受付時間:月~金9:00~12:00 / 13:00~18:00(祝日・年末年始休業日を除く)

<http://www.oracle.com/jp/contact-us>

Integrated Cloud

Applications & Platform Services

ORACLE®