

大阪SAS勉強会

SASを用いてデータベース接続処理を実施する際の チューニング方法

2024/10/18

SAS Institute Japan

Toshiaki Habu



注意事項

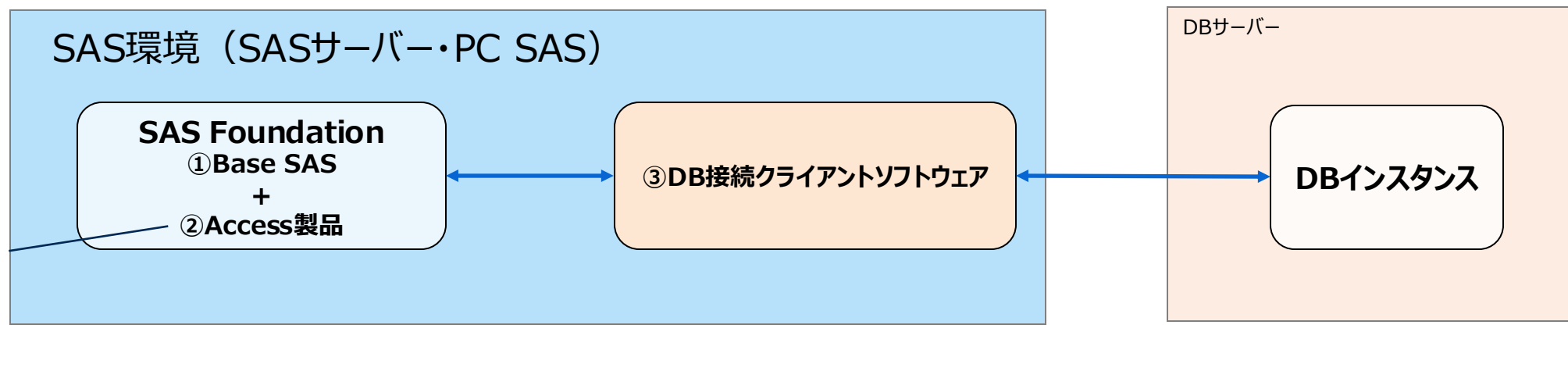
- 本プレゼンテーションの内容・発表は、私個人の意見や見解に基づくものであり、必ずしも所属会社の公式な立場や意見を代表するものではありません。
- 前回の本勉強会に参加させて頂き、面白そうだからエントリーしてみました。
1 SASプログラマーとしての発表になります。

はじめに

- 皆さんが利用されているSASからデータベース（OracleやらPostgres）を利用している方はいらっしゃいますか？
- 近年、治験以外のデータを取り扱うケースの問い合わせが増えております． どのように取り扱うべきか？等を含め、色々とお問い合わせを頂く事がございます．
（例：Real World Dataのデータ処理等）
- 直ぐには使う事はないかもしれませんが、SASがどのようにデータベースを扱っているか？のアーキテクチャ含め説明させて頂ければと思います．
- **目標：SASのDB処理のアーキテクチャを理解し、どのような対応が出来るか把握する**

DBへのアクセスについて

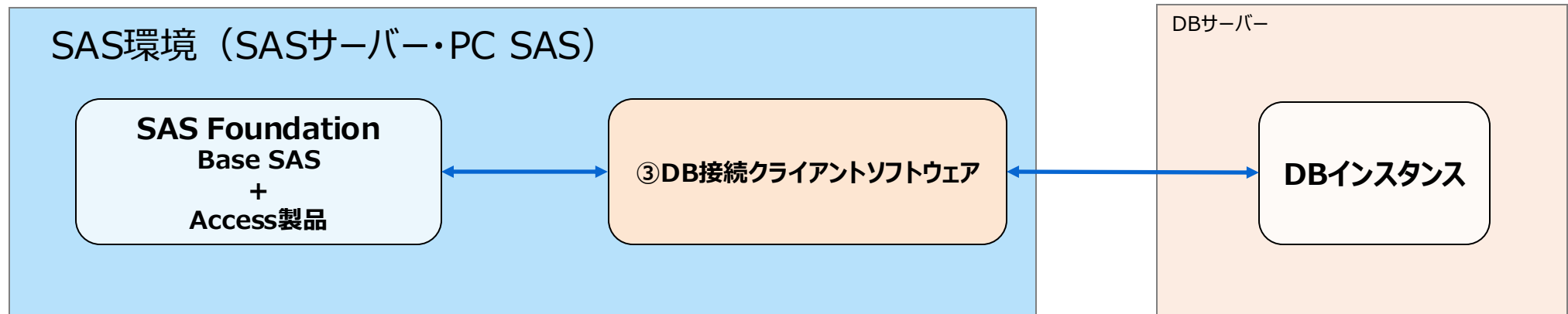
- SAS製品（SAS Foundation=Base SAS）からDB上のデータへアクセスする際は、**SAS環境に導入したDB接続クライアントソフトウェアを介して接続/アクセスします。** よって、事前にSASサーバーへ接続対象DBに合わせた接続クライアントソフトウェアをインストール、接続設定を行う必要があります。



- Oracleへアクセスする場合、①Base SAS、②Access Interface to Oracle or ODBCといったSAS製品以外に、③DB接続クライアントソフトウェアとしてOracle Clientソフトウェア、もしくはOracle ODBCドライバをSAS環境へインストールし、接続設定を行う必要があります

Access製品の仕組みについて

SASのデータステップにおける処理ステップ



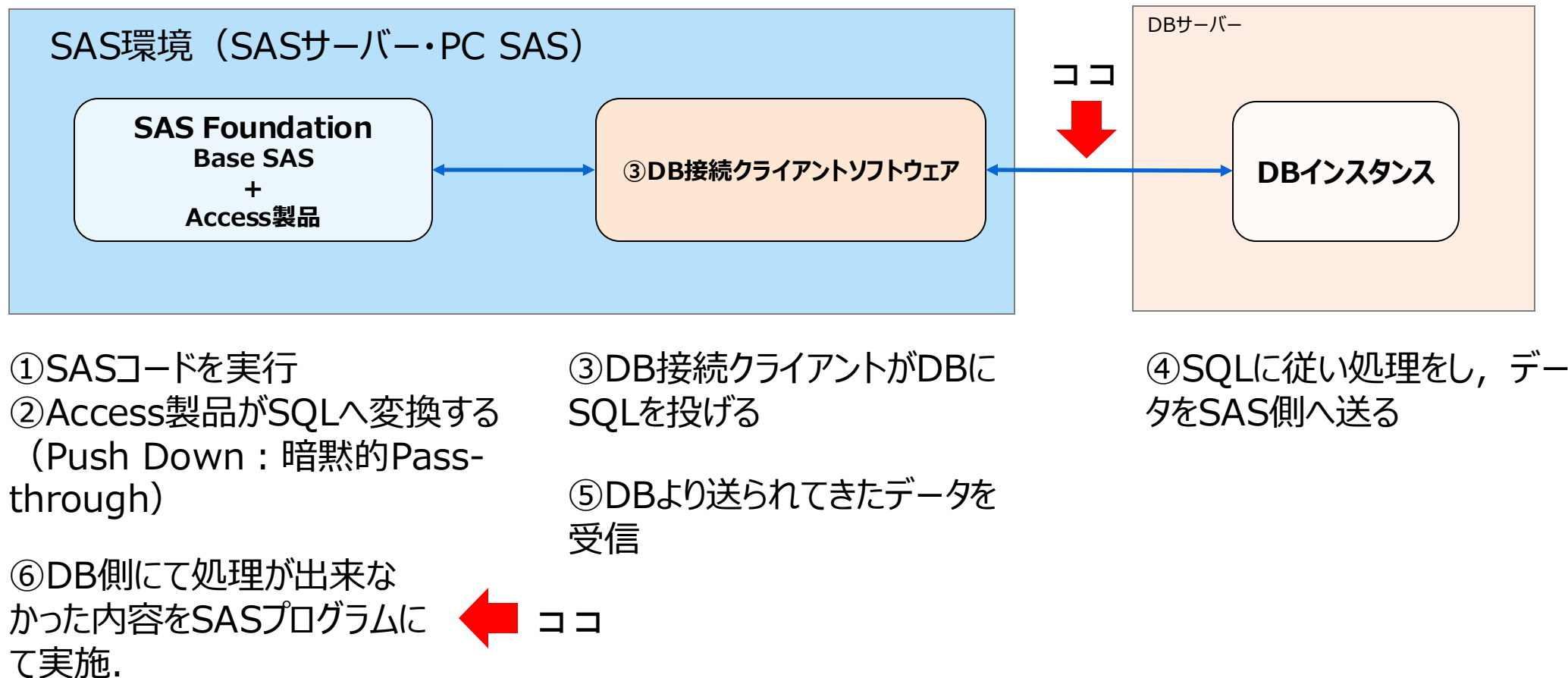
- ①SASコードを実行
- ②Access製品がSQLへ変換する
(Push Down : 暗黙的Pass-through)
- ⑥DB側にて処理が出来なかった内容をSASプログラムにて実施.

- ③DB接続クライアントがDBにSQLを投げる
- ⑤DBより送られてきたデータを受信

- ④SQLに従い処理をし、データをSAS側へ送る

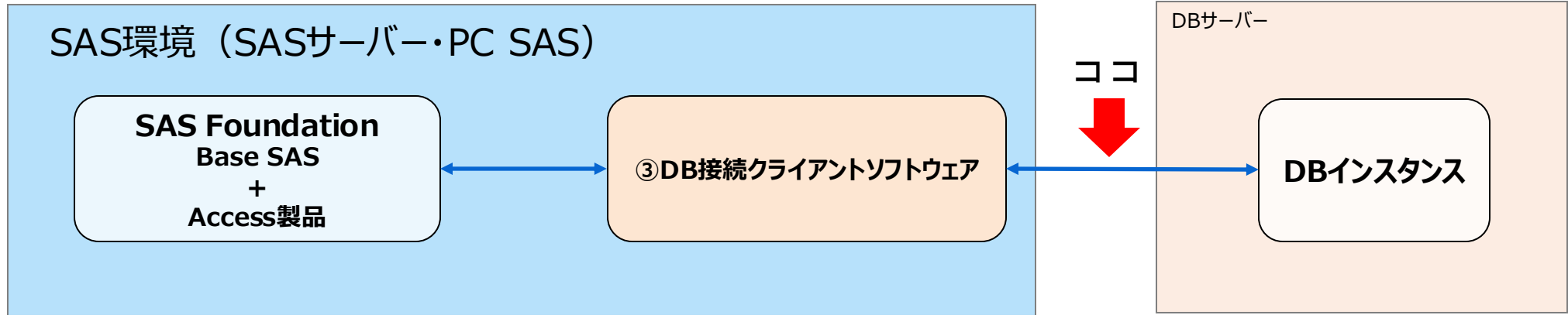
Access製品の仕組みについて

実行処理速度を上げる為のポイント（ボトルネックになりやすい箇所）



Access製品の仕組みについて

実行処理速度を上げる為のポイント（ボトルネックになりやすい箇所）



SAS環境とのデータのやり取りを減らすこと大事.

① Passing Function（SASの機能をDB側にて実行できる変数）を使う。

公開されているマニュアルに例示あり [SAS Help Center: Passing SAS Functions to Oracle](#)

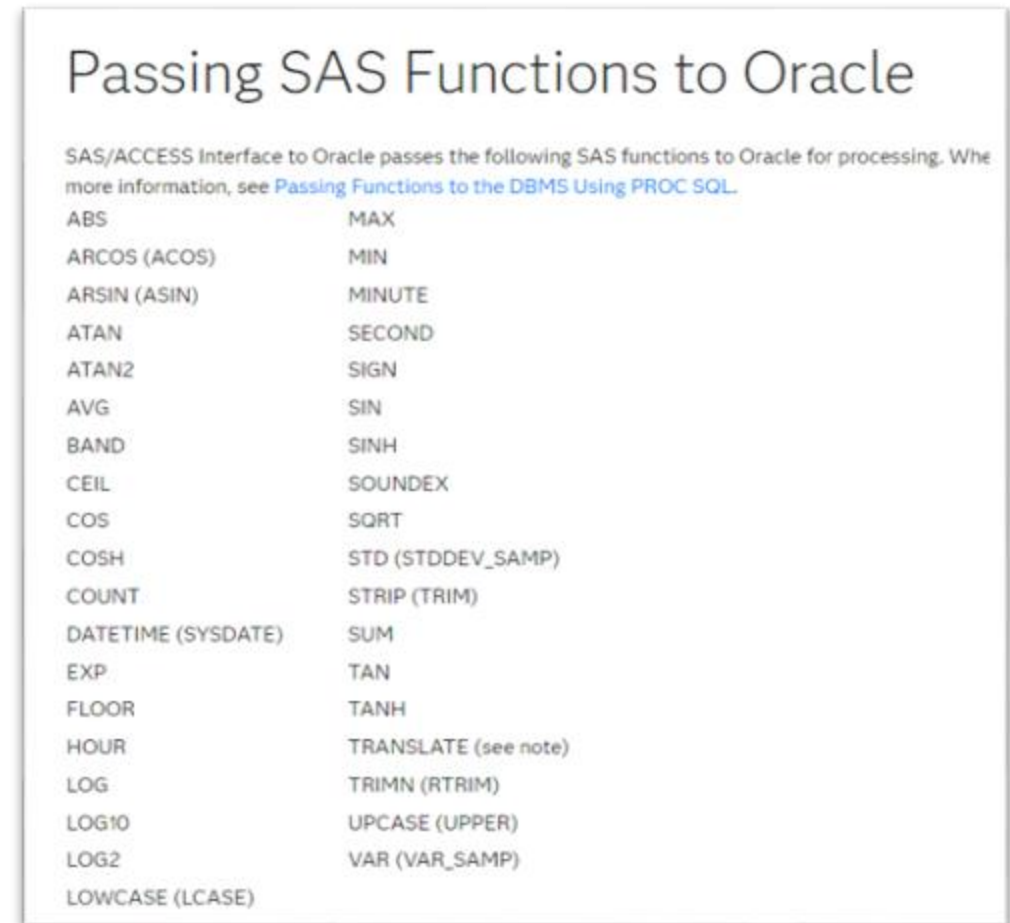
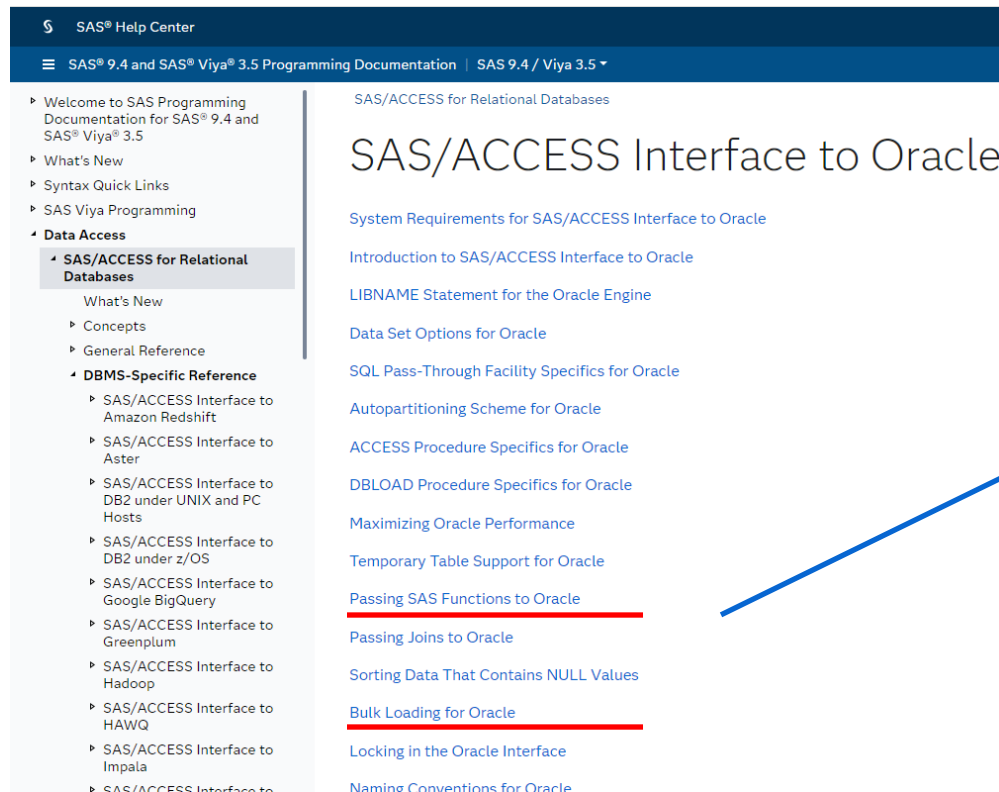
① DB側にて処理をしたいSQLを書けるのであれば、execute(明示的Pass-through)の機能を利用する。

② 発行しているSQLを確認し、より最適な処理が実行できないか検討する

`options sastrace=',,,d' sastraceloc=saslog nostsuffix msglevel=i;`

Passing Function

- SASのプログラミングマニュアルにてData Accessの部分に記載あり



SAS® 9.4 and SAS® Viya® 3.5 Programming Documentation

https://go.documentation.sas.com/doc/jp/pgmsascdc/9.4_3.5/pgmsaswlcmm/home.htm

SQL Procedure: execute

- 例文 :

```
proc sql;  
  execute(drop table ' My Invoice ') by db;  
  execute(create table ' My Invoice '(  
    ' Invoice Number ' LONG not null,  
    ' Billed To ' VARCHAR(20),  
    ' Amount ' CURRENCY,  
    ' BILLED ON ' DATETIME)) by db;  
  execute(insert into ' My Invoice '  
    values( 12345, 'John Doe', 123.45, #11/22/2003#)) by db;  
quit;
```

SAS® 9.4 and SAS® Viya® 3.5 Programming Documentation
https://go.documentation.sas.com/doc/en/pgmsascdc/9.4_3.5/sqlproc/n0tpd3yaqvep53n1g8wahav3hgco.htm

SASのAccess製品は、SASのデータステップをSQLへ変換していますが、このプロシジャを利用するとそのままDB側へSubmitすることが可能です。

Proc sql;

Execute(DB側で処理をしたいSQL)

Quit;

DBにSubmitをしたい最適なSQLが解る方は、こちらの方法にて実行する事により、より高速に処理が出来る可能性があります。

SASTRACE= SAS System Option

- 例文 :

```
options sastrace=',,,d'  
sastraceloc=saslog nostsuffix;  
  
libname mydblib oracle user=myusr1  
password=mypwd1 schema=bday_data;  
  
data mydblib.snow_birthdays; set  
work.winter_birthdays; run;  
  
libname mydblib clear;
```

SAS® 9.4 and SAS® Viya® 3.5 Programming Documentation
https://go.documentation.sas.com/doc/en/pgmsascdc/9.4_3.5/acreldb/n0732u1mr57ycrn1urf24gzo38sc.htm

本Optionをつける事で、Data StepにてDBにSubmitしたSQLがログファイルに出力されます。

```
CREATE TABLE SNOW_BIRTHDAYS(empid NUMBER ,birthdat DATE,lastname VARCHAR2 (18))  
ORACLE_5: 準備済み:  
INSERT INTO SNOW_BIRTHDAYS (empid,birthdat,lastname) VALUES  
(:empid,TO_DATE(:birthdat,'DDMONYYYY','NLS_DATE_LANGUAGE=American'),:lastname)  
注記: データ セット WORK.WINTER_BIRTHDAYS から読み取られた観測値は 3 つでした。  
ORACLE_6: 実行済み:
```

発行しているSQLにて無駄な処理が無いか？を確認する為に利用できます。

Thank you!!

貴社環境に適した高速化を検討したい場合、SASのコンサルティングサービス（有償支援）を活用する事も可能です。
私のような小童ではなく、ガチ勢が多くいますので、必要に応じご検討ください。

