



PL/SQL CAMP - MEETUP #4

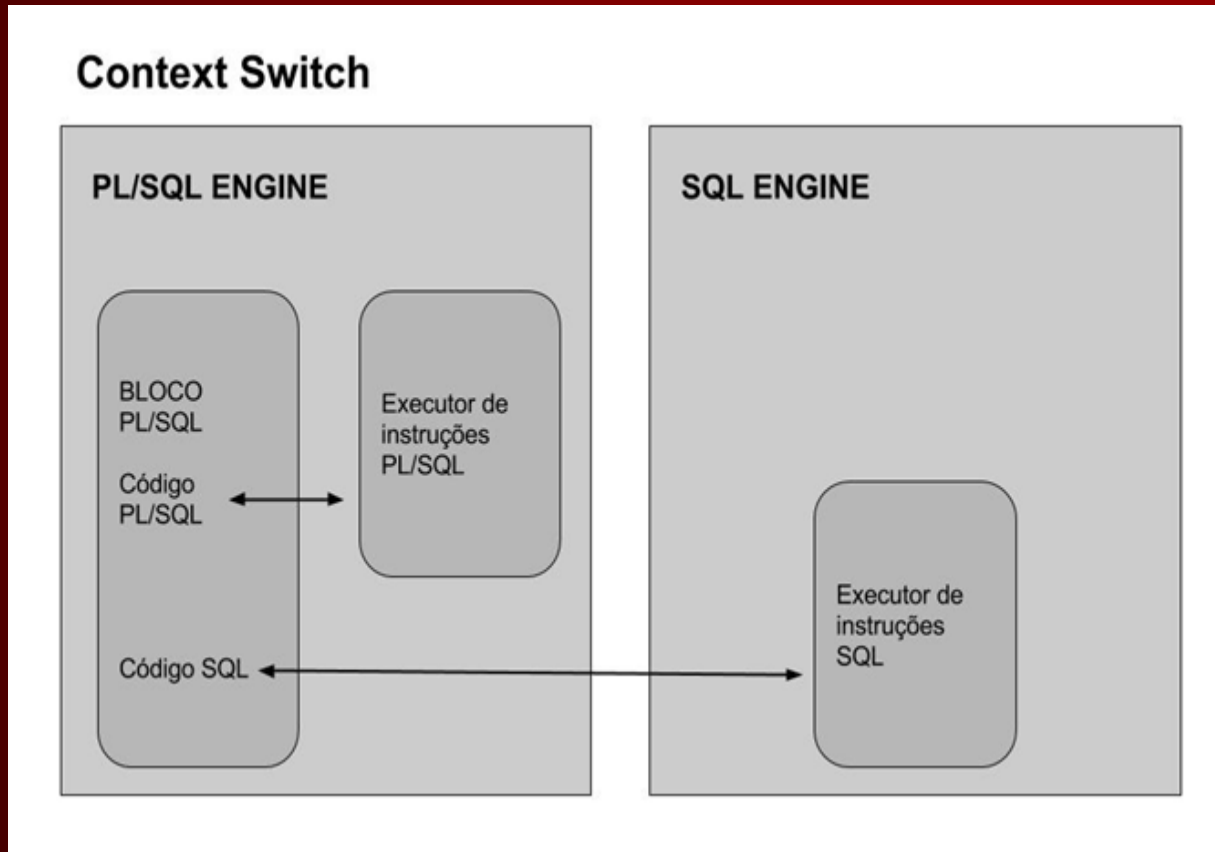
18/10/2018

# BULK COLLECT & FORALL

# INTRODUÇÃO

Esta apresentação é útil para quem trabalha no desenvolvimento dentro do banco de dados Oracle utilizando PL/SQL, e que deseja melhorar a performance, fazendo tuning de suas consultas ou outras operações DML, como INSERT, UPDATE e DELETE, utilizando as instruções BULK COLLECT e o FORALL.

# CONTEXT SWITCH



O Context Switch nada mais é do que a troca entre as engines SQL e PL/SQL. Quando a engine do PL/SQL encontra uma instrução SQL, como um select, ele para a execução e passa essa instrução para a engine do SQL. Quando a instrução é finalizada, a engine do SQL retorna para a engine do PL/SQL, que continua a execução.

Em loop de 10 mil linhas irá acontecer 10 mil contexto switch.

Fica claro que devemos evitar que nossos códigos façam muitos Context Switches...

...Mas como não conseguimos fugir dos loops devido ao grande volume de dados, é aí que entram as instruções BULK COLLECT e o FORALL

# BULK COLLECT

O comando BULK COLLECT permite codificar declarações SQL sem a necessidade de busca individual de informação utilizando o código PL/SQL, ou seja, seu uso torna desnecessária a utilização de vários loops para popular as coleções evitando que o processamento da Stored Procedure seja demorado.

# SINTAXE COM DECLARAÇÃO DA CLÁUSULA BULK COLLECT

```
SELECT coluna1[, coluna2, ...]  
BULK COLLECT INTO colecao1[, colecao2, ...]  
FROM tabela ...
```

```
FETCH cursor [parametros...]  
BULK COLLECT INTO colecao1[, colecao2, ...]
```

# LIMIT

Pode acontecer de o Oracle gastar tempo demais administrando o cursor na memória e quase nenhum tempo executando-o no bloco PL/SQL. Uma solução melhor seria se o Bulk Collect fosse feito mais de uma vez para dividir o trabalho e o esforço.

Para que seja possível que o nosso cursor retorne menos linhas, temos que adicionar uma nova cláusula, a LIMIT.

# SINTAXE COM DECLARAÇÃO DA CLÁUSULA BULK COLLECT COM LIMIT

FETCH cursor [parametros...]

BULK COLLECT INTO colecao1[, colecao2, ...]

LIMIT XXX



# FORALL

A palavra-chave FORALL permite que uma declaração Data Manipulation Language (DML) seja baseada no conteúdo de uma coleção, sendo executada com bastante eficiência. Quando FORALL é usada, a declaração DML é executada apenas uma vez, para toda a coleção.

O desempenho resultante é muito melhor do que se tivesse codificando comandos DELETE, UPDATE ou INSERT dentro de loops (interações), gerando interrupções da linguagem PL/SQL

# SINTAXE COM DECLARAÇÃO DA CLÁUSULA FORALL

FORALL <index\_name> IN <lower\_boundary> .. <upper\_boundary>  
<sql\_statement>  
SAVE EXCEPTIONS;

FORALL <index\_name> IN INDICES OF <collection>  
[BETWEEN <lower\_boundary> AND <upper\_boundary>]  
<sql\_statement>  
SAVE EXCEPTIONS;

FORALL <index\_name> IN INDICES OF <collection>  
VALUES OF <index\_collection>  
<sql\_statement>  
SAVE EXCEPTIONS;

# LINKS

- <https://www.devmedia.com.br/processamento-em-bulk-no-oracle/37001>
- <https://www.devmedia.com.br/artigo-sql-magazine-8-otimizacao-de-store-procedures-em-pl-sql/5162>

OBRIGADO!!!

CLEBER FLAITT

CFLAITT@GMAIL.COM

[www.linkedin.com/in/cleber-flaitt](http://www.linkedin.com/in/cleber-flaitt)