**FUNDAÇÃO OSWALDO ARANHA**

**CENTRO UNIVERSITÁRIO DE VOLTA REDONDA**

**CURSO DE SISTEMAS DE INFORMAÇÃO**

**TRABALHO ACADÊMICO**

**VIEW E MATERIALIZED VIEW(SNAPSHOT)**

Autor

**Iuri de Souza Silva - 201500534**

**Volta Redonda**

**Maio/2017**

**Sumário**

[**1.** **Introdução** 3](#_Toc482908543)

[**2.** **View e Materialized View(Snapshot)** 3](#_Toc482908544)

[2.1 View 3](#_Toc482908545)

[2.1.1 Definição 3](#_Toc482908546)

[2.1.2 Sintaxe 4](#_Toc482908547)

[2.1.3 Aplicação 4](#_Toc482908548)

[2.2 Materialized View ou Snapshots 7](#_Toc482908549)

[2.2.1 Definição 7](#_Toc482908550)

[2.2.2 Sintaxe 7](#_Toc482908551)

[2.2.3 Aplicação 8](#_Toc482908552)

[2.3 Exemplo usando o schema HR do banco Oracle 8](#_Toc482908553)

[**3.** **Conclusão** 9](#_Toc482908554)

[**4.** **Bibliografia** 10](#_Toc482908555)

# **1. Introdução**

Na linguagem SQL existem várias instruções que as vezes são desconhecidas pelos usuários ou então os usuários não sabem diferenciá-las quanto a definição e aplicação. Neste momento iremos falar sobre as views e materiliazed view. Ao fim da leitura desse trabalho você saberá a definição de cada uma dessas instruções, saberá criá-las e saberá quando deve usar cada uma delas. Esse trabalho foi proposto pela disciplina de Banco de Dados II, cujo ministro é o professor Rosenclever e tem por objetivo gerar e disseminar conhecimento.

# **2. View e Materialized View(Snapshot)**

Os itens descritos no tema acima são instruções utilizadas no SQL com objetivos bem definidos. Abaixo será possível ter a definição de cada um dos temas, a aplicação, a sintaxe, exemplos ...

## **2.1 View**

### 2.1.1 Definição

View, ou visão no português, é uma consulta(select) que fica armazenada no banco de dados a fim de otimizar uma consulta, de manter a segurança do dados, de direcionar o que os usuários podem ver, entre outras coisas. Ela pode ser considerada uma tabela virtual.

Com uma view, não é necessário dar acesso direto a tabelas para os usuários. Conforme já dito, é possível dimensionar os campos que o usuário poderá ver, e com base nisso, dar acesso somente sobre a view. Levando em consideração que os campos serão selecionados para serem exibidos na view, neste momento se tem um ganho também na organização e exibição destes.

Vale ressaltar que uma view não grava os dados da consulta e o select utilizado na view não pode conter a clausula ORDER BY, diante disso, toda vez que uma view é executada a tabela no banco é consulta. Logo, é importante que se tenha um select bem definido e coerente para que sua view possa retornar os dados rapidamente e os dados sejam confiáveis.

### 2.1.2 Sintaxe

Neste momento, apresento uma sintaxe mais enxuta para a criação de uma view e logo após explico cada comando.

*CREATE OR REPLACE VIEW view\_name AS*

*SELECT column1, column2, ...*

*FROM tables*

*[WHERE conditions];*

Cada comando:

- CREATE: comando básico de criação;

- OR REPLACE: recria,atualiza a view caso ela já exista. Caso não exista ainda, será criada pela primeira vez;

- view\_name: nome da sua view;

- SELECT comando de consulta;

- Column1,2,etc: campos que serão retornados na view;

- FROM: origem dos campos consultados;

- Where: condição ou filtro para exibição dos campos. Esse comando é opcional, por conta disso que está entre colchetes.

### 2.1.3 Aplicação

Uma view também pode ser utilizada na inserção, atualização e remoção de dados. Uma view também é utilizada em consultas.

Vamos imaginar a seguinte situação: O departamento de qualidade deseja saber os pacientes e as suas respectivas doenças. Para isso temos as seguintes tabelas.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Tabela Paciente** | | | | | |
| **Código de Cliente** | **Nome** | **Idade** | **Titular** | **Dependente** | **CID** |
| 1795520 | Carlos Henrique | 55 | S | N | C001 |
| 1795521 | João Marcos | 32 | N | S | C001 |
| 1795522 | Mateus Benedito | 22 | S | N | C002 |
| 1795523 | Tiago Brandão | 45 | S | N | C003 |
| 1795524 | Kleber Machado | 18 | N | S | C002 |
| 1795525 | Galvão Bueno | 70 | S | N | C004 |

FONTE: Autoral

|  |  |
| --- | --- |
| **Tabela Doenca** | |
| **CID** | **Nome\_Doença** |
| C001 | Diabets |
| C002 | Tuberculose |
| C003 | Raiva |
| C004 | Febre |

FONTE: Autoral

Para atender a necessidade do setor, podemos criar a seguinte view:

*Create OR REPLACE View PAC\_DOENTES as*

*Select p.Nome, d.Nome\_Doenca*

*from Paciente P JOIN Doenca d*

*ON p.CID = d.CID;*

De posse da view, poderíamos fazer uma consulta nela para retornar os dados que se quer. Os comandos referentes a essa consulta seriam:

*SELECT \*FROM PAC\_DOENTES;*

O resultado desse select será:

|  |  |
| --- | --- |
| **PAC\_DOENTES** | |
| **NOME** | **NOME\_DOENCA** |
| Carlos Henrique | Diabets |
| João Marcos | Diabets |
| Mateus Benedito | Tuberculose |
| Tiago Brandão | Raiva |
| Kleber Machado | Tuberculose |
| Galvão Bueno | Febre |

FONTE: Autoral

Para finalizar esse assunto, ainda nos falta falar sobre a exclusão de uma view. A criação é feita através do CREATE, a atualização é feita através do OR REPLACE, e a exclusão é feita conforme demonstro abaixo:

*DROP VIEW name\_view;*

## **2.2 Materialized View ou Snapshots**

### 2.2.1 Definição

É também uma view só que diferentemente da “view padrão” armazena o resultado ao invés da consulta. Por conta disso o nome de view materializada.

Esta por sua vez tem a atualização dos dados feita sempre que a tabela consultada sofrer uma inclusão, alteração, exclusão. Ou seja, a atualização da view materializada é feita de forma periódica.

### 2.2.2 Sintaxe

Neste momento, apresento uma sintaxe mais enxuta para a criação de uma view e logo após explico cada comando.

*Create Materialized View nome\_MatView*

*Build Immediate*

*Refresh fast*

*as*

*Select column1, column2, …*

*FROM table*

*[WHERE conditions];*

Cada comando:

- Create: comando básico de criação;

- nome\_ MatView: nome da sua view;

- Build Immediate: utiliza imediatamente os dados do select, com isso se tem mais agilidade;

- Refresh fast: diz que as modificações serão utilizadas somente pela view que está sendo criada;

- Select: comando de consulta;

- Column1,2,etc: campos que serão retornados na view;

- FROM: origem dos campos consultados;

- Where: condição ou filtro para exibição dos campos. Esse comando é opcional, por conta disso que está entre colchetes.

### 2.2.3 Aplicação

Uma das principais utilizações de uma view materializada é realizar atualização de dados, pois você no primeiro momento não fará a atualização diretamente na tabela do banco, você irá alterar os dados da view. Depois da homologação, a atualização pode ser aplicada no banco com maior segurança. Esse tipo de rotina de atualizaçã utilizando uma view materializada é orientada pela própria ORACLE.

Esse tipo de instrução é também bastante utilizado em Data Warehouse, que é um depósito de dados organizados e relacionados que dão embasamento para tomada de decisões.

## 

## **2.3 Exemplo usando o schema HR do banco Oracle**

Vamos imaginar que o departamento de RH queira conhecer todos os funcionários e o seu endereço composto pelos campos: nome da rua, país e a região. Contudo, é necessário que o relatório traga somente funcionários que o nome comece com a letra J e o relatório deve ser ordenado de forma crescente pelo nome da rua.

Diante da necessidade acima exposta, a view ficaria da seguinte forma:

*create or replace view mvw\_Busca\_Nome\_Endereco as*

*select e.FIRST\_NAME NOME, l.STREET\_ADDRESS RUA,c.COUNTRY\_NAME PAÍS, r.REGION\_NAME REGIÃO*

*from LOCATIONS l*

*inner join COUNTRIES c on (c.COUNTRY\_ID = l.COUNTRY\_ID)*

*inner join REGIONS r on (r.REGION\_ID = c.REGION\_ID)*

*inner join DEPARTMENTS d on (d.LOCATION\_ID = l.LOCATION\_ID)*

*inner join EMPLOYEES e on (e.DEPARTMENT\_ID = d.DEPARTMENT\_ID)*

*where e.FIRST\_NAME like 'J%'*

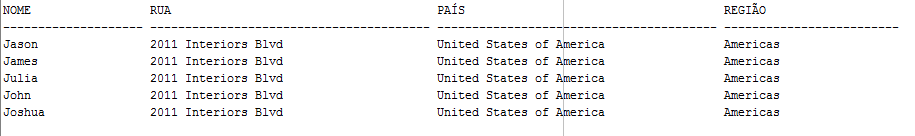
*and l.POSTAL\_CODE is not null*

*order by l.STREET\_ADDRESS;*

Para exibir essa view, basta executar os comandos abaixo:

*select \*from MVW\_BUSCA\_NOME\_ENDERECO;*

O resultado será:



FONTE: Autoral

# **3. Conclusão**

Diante das informações expostas acima, você pode estar se perguntado: Enfim, quando eu uso uma simples view ou uma view materializada?

A resposta é simples. Se você precisa que suas buscas tenham um desempenho melhor, use ma view materializada. Se você precisa fazer muitas alterações de dados nas tabelas, use uma view.

Após as pesquisas feitas e o material acima elaborado, não ficam dúvidas que o objetivo de adquirir conhecimento desse trabalho foi alcançado.

# **4. Bibliografia**

Disponível em: <<http://www.dicasdeprogramacao.com.br/qual-a-diferenca-entre-view-e-materialized-view/>>. Acessado em: 14 Maio 2017;

Disponível em: <<https://imasters.com.br/artigo/1773/oracle/views-materializadas/?trace=1519021197&source=single>>. Acessado em: 14 Maio 2017;

Disponível em: <<https://docs.oracle.com/cd/B19306_01/server.102/b14200/statements_6002.htm>>. Acessado em: 14 Maio 2017;

Disponível em: <https://www.techonthenet.com/oracle/views.php>. Acessado em: 14 Maio 2017;