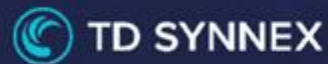


Diamante



Platina



DISCOVER

Ouro



VERTICA  
by opentext

Prata

TRACES



Apoio

FIAP



# Apresentação

**DBA BRASIL**  
DATA & CLOUD



Gilson Martins



17 anos de experiência na área de TI

15 anos focado em administração de banco de dados e tecnologias Oracle

Consultor Oracle e Instrutor de GoldenGate há 11 anos

Host no GoldenTalks e Golden Tips 

Alta disponibilidade com Grid Infrastructure, RAC, Data Guard, GoldenGate

Migração e replicação de dados

Certificações Database OCP 10g / 11g / 12c

Certificações OGG 11g / 12c

Atualmente → Engenharia de Dados e BigData



# Apresentação

**DBA BRASIL**  
DATA & CLOUD



Enterprise Architect

Formado em Big Data com especialização em Engenharia de Dados pela FIAP

Rafael Milanez



# Introdução



# Bancos Relacionais (Monolitos)



# Bancos Relacionais (Monolíticos)



# Bancos Relacionais (Monolíticos)





- ✓ Dados Históricos
- ✓ Dados Estruturados
- ✓ Banco de Dados Relacional





- ✓ Dados de várias fontes distintas e formatos diferentes
- ✓ Dados **Não-Estruturados**
- ✓ Banco de Dados **Não-Relacional**

## Os três Vs

- Variedade
- Volume
- Velocidade
  
- Valor
- Veracidade
- Visualização

- ✓ Json
- ✓ Avro
- ✓ Parquet
- ✓ XML
- ✓ TXT
- ✓ Etc...

# BIG DATA

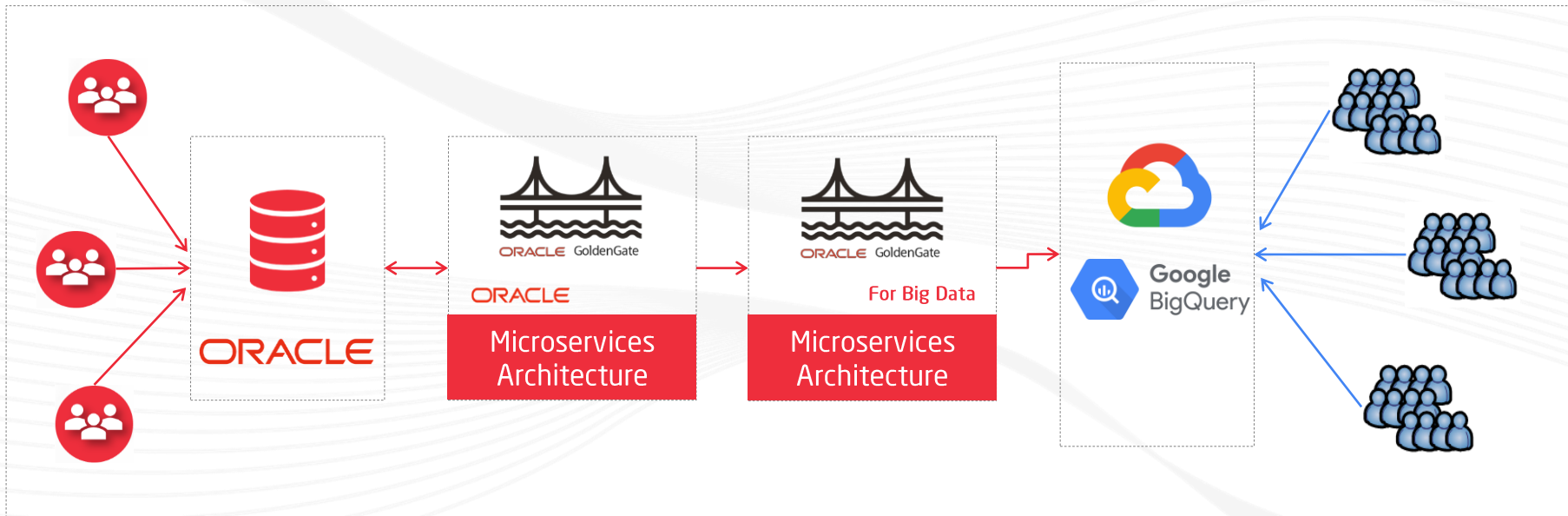


# ETL vs REAL-TIME (Streaming Data)



- ✓ Processamento Batch

# Engenheiro de Dados - Streaming Data



- ✓ Criação e sustentação de Pipelines
- ✓ Melhores ferramentas e Tecnologias

- ✓ Ingestão de Dados Real-Time

# O que é BigQuery?



# O que é BigQuery?



## 1 Datawarehouse Corporativo

- O BigQuery é um serviço de Datawarehouse Corporativo, disponível na Google Cloud.

## 2 Totalmente Gerenciado

- Não se preocupe em gerenciar ou dimensionamento infraestrutura, a Google faz isso para você.

## 3 Serverless

- Baseado em uma arquitetura que não demanda alocação de servidores.

## 4 Escala de acordo com sua demanda

- Permite dimensionar seus dados na escala de exabytes.

## Utiliza Criptografia por padrão 5

- Todos os dados do BigQuery são compactados e criptografados automaticamente por padrão.

## Suporta Linguagem SQL 6

- Suporta linguagem SQL compatível com ANSI:2011, ou seja, é o padrão SQL a partir de 2011 para frente.

## Alta Disponibilidade 7

- SLA de 99.99%

## Recursos Integrados Disponíveis 8

- Recursos integrados, como Aprendizado de Máquina, Análise Geoespacial, compatível com streaming etc...



# Por que o BigQuery é rápido?



- Ótimo para sistemas transacionais e casos de uso OLTP.
- Compressão de dados: Uma linha pode ter diferentes tipos de dados, algoritmo de compactação menos eficiente.

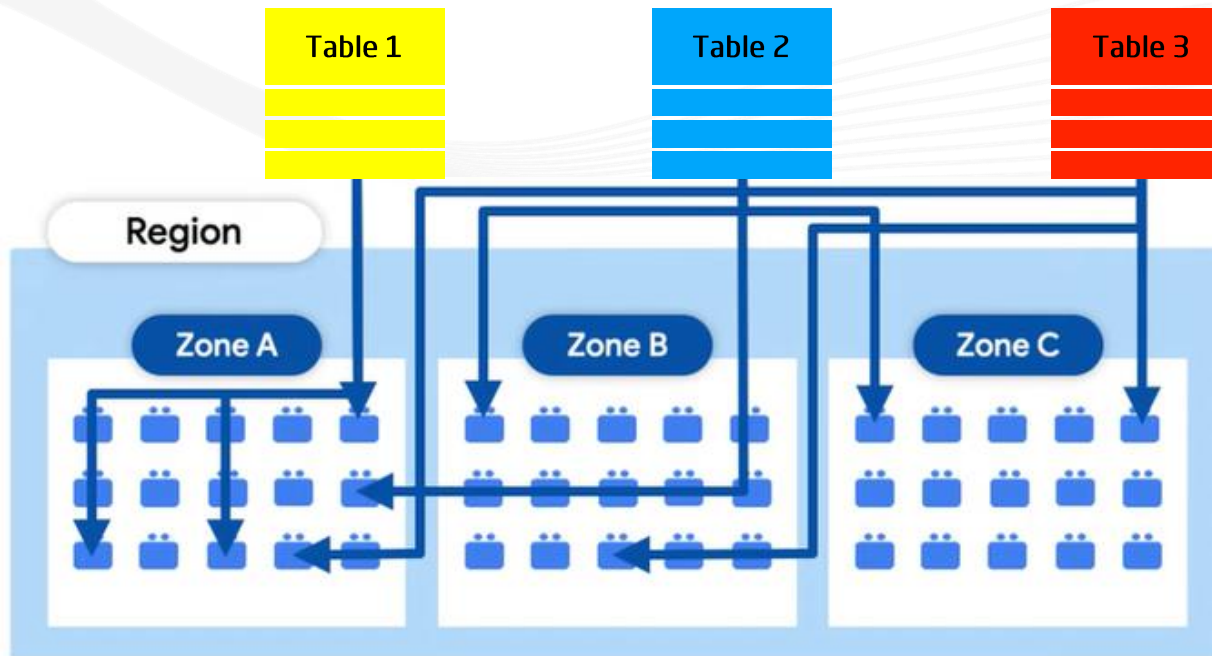
Tradicional RDBMS Storage

Carlos Tecnologia 4500	Ana RH 9300	João Financeiro 12500

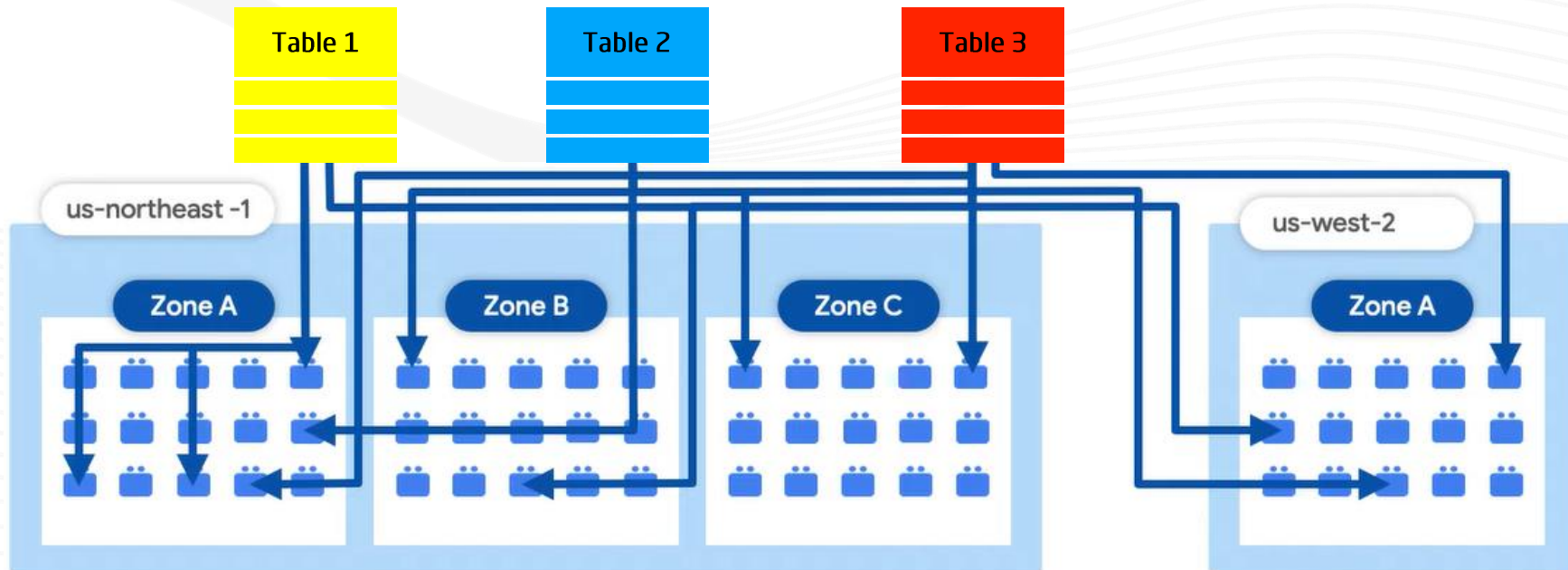
- As tabelas são armazenadas em formato colunar otimizado.
- Maior compressão, dados iguais são armazenados juntos.
- Excelente para operações de agregações e agrupamentos.

BigQuery Storage

Carlos	Tecnologia	4500
Ana	RH	9300
João	Financeiro	12500

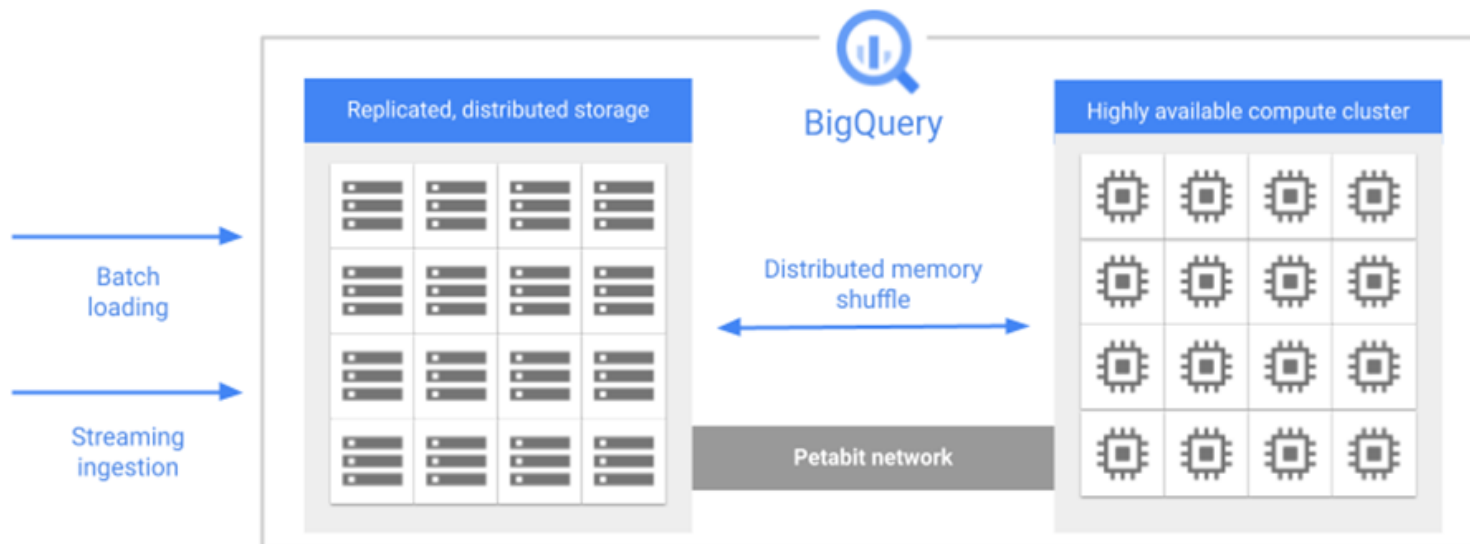






# Arquitetura





# Arquitetura

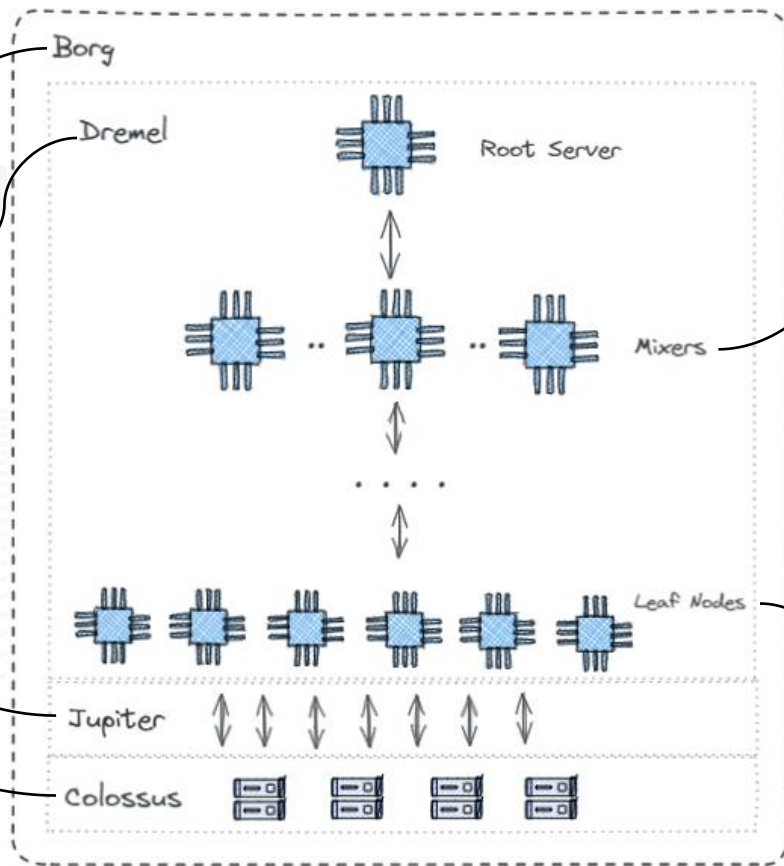


Plataforma que deu origem aos contenedores.  
Sistema de gerenciamento de cluster em larga escala do Google

Transforma a consulta SQL em uma árvore de execução

Rede Petabit

Sistema de arquivos distribuídos do Google (columnar storage)



Após a leitura dos dados pelos nós folha, temos os galhos, também conhecidos como *mixers*, onde é realizada a agregação dos dados que são retornados para o nível acima e para o usuário.

São responsáveis pela leitura massiva e processamento dos dados que estão armazenados no *FileSystem* do Google

# Interface com o BigQuery



# UI da Web



The screenshot displays the Google Cloud BigQuery console. At the top left, the Google Cloud logo and 'My First Project' are visible. A search bar at the top center contains the text 'bigquery'. Below the search bar, a dropdown menu lists 'PRODUCTS & PAGES' with 'BigQuery Data warehouse/analytics' highlighted. The main content area is divided into three sections: 'DOCUMENTATION & TUTORIALS' on the left, 'Explorer' in the middle, and a query editor on the right. The Explorer shows a tree view with 'SQL workspace' selected. The query editor is titled 'Untitled 2' and contains a single line of text: 'Type a query to get started'. The bottom of the interface features 'PERSONAL HISTORY' and 'PROJECT HISTORY' tabs, along with a 'REFRESH' button.



- Ferramenta de linha de comando baseada em Python do BigQuery.
- Cloud Shell ou Google SDK (gcloud) para interagir por meio de um terminal

## Diversas operações:

- **bq ls** - listar objetos.
- **bq load** - carregar dados para tabelas.
- **bq mk** - criar datasets, tabelas, views etc...
- **bq rm** - apagar datasets, tabelas, views etc...
- **bq show** - exibir detalhes de uma tabela.
- **bq help** - listar opções.

```
CLOUD SHELL
Terminal (bionic-spot-390417) X + v
dbabrasil2023@cloudshell:~ (bionic-spot-390417) $ bq version
This is BigQuery CLI 2.0.93
dbabrasil2023@cloudshell:~ (bionic-spot-390417) $ bq query --use_legacy_sql=false \
'select
  count(*)
from
  `bigquery-public-data`.samples.shakespeare'
+-----+
| f0_   |
+-----+
| 164656 |
+-----+
dbabrasil2023@cloudshell:~ (bionic-spot-390417) $ bq ls
datasetId
-----
dba_brasil
dbabrasil2023@cloudshell:~ (bionic-spot-390417) $
```

# Recursos do BigQuery







- Resultados das queries são armazenados no Cache.
- Execuções posteriores da mesma query acessam o Cache:
  - ✓ Melhor performance
  - ✓ Não é cobrado (sem custo)

## Restrições:

- Query precisa ser uma réplica exata da query original.
- Cache é armazenado em tabelas temporárias por aproximadamente 24h.
- Consultas Federadas não utilizam recurso de Cache.
- Tabelas utilizadas em ingestão de dados por streaming.
- Consulta Curinga (Prefixo/Sufixo)
- Caso a consulta utilize funções não determinísticas: **CURRENT\_TIMESTAMP()** , **CURRENT\_DATE** , **CURRENT\_USER()** , **SESSION\_USER()** , retornam valores diferentes dependendo de quando uma consulta é executada, não considera Cache.

The screenshot shows a web interface for configuring SQL Cache. The 'Query settings' section includes a 'Destination' dropdown, a 'Dataset' field, a 'Table Id' field, and 'Destination table write preference' options: 'Write if empty' (selected), 'Append to table', and 'Overwrite table'. A 'Cache preference' section at the bottom has a checked checkbox for 'Use cached results'. A warning dialog box is open, stating: 'This attempts to use results from a previous run of this query, as long as the referenced tables are unmodified. If cached results are returned, you will not be billed for any usage. Results are cached for approximately 24 hours. Caching cannot be enabled when a destination table is selected. [Learn more](#)'.



## Dados que estão armazenados fora do Bigquery:

- Cloud SQL
- Cloud Spanner
- BigTable
- Cloud Storage
- Google Drive
- BigQuery Omni

- Use a função `EXTERNAL_QUERY` para consultar o banco de dados externo.
- A performance é **degradada** em relação a uma consulta nativa.
- Não utiliza recurso de **Cache**.
- **Regiões devem ser compatíveis**, uma região única do BigQuery só pode consultar um recurso na mesma região.
- Uma consulta executada na multi-region EUA do BigQuery pode consultar qualquer região única na região geográfica dos EUA, como `us-central1`, `us-east4` ou `us-west2`.

## Formatos de arquivos compatíveis:

- JSON (apenas delimitado por nova linha)
- CSV
- Avro
- Parquet
- Iceberg
- ORC

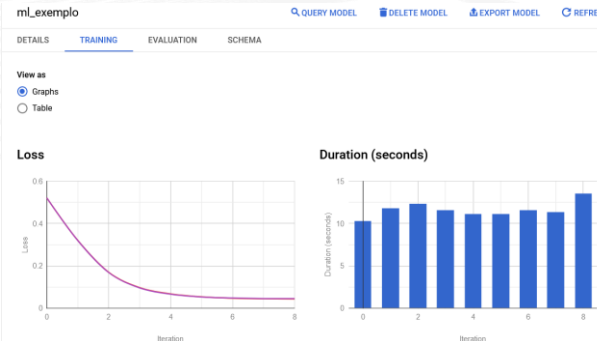
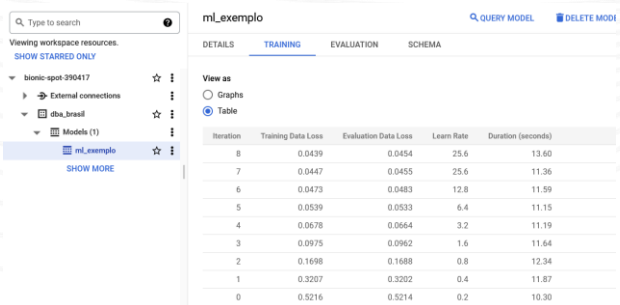
```
SELECT c.customer_id, c.name FROM
mydataset.customers AS c LEFT OUTER JOIN
EXTERNAL_QUERY('us.connection_id',
''SELECT customer_id, MIN(order_date) AS
first_order_date FROM orders
GROUP BY .....
```



## Existem muitos tipos de modelos integrados ao BQML:

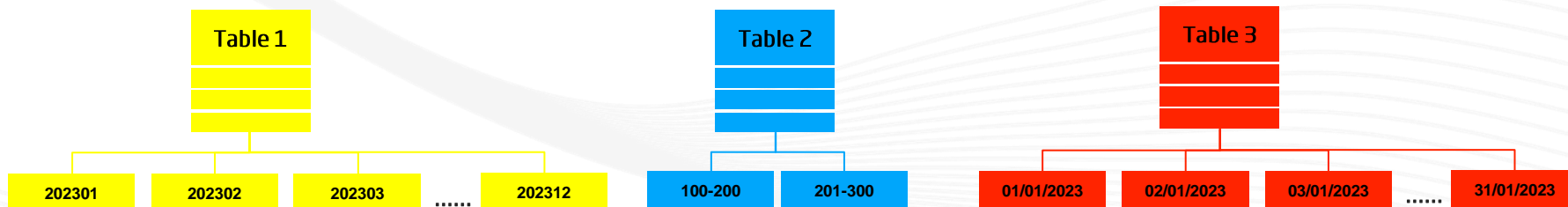
- Regressão Linear
- Regressão Logística
- Serie Temporal
- K-means
- Árvores aumentadas
- PCA
- Fatoração de array
- Suporta importação de modelos do TensorFlow

```
CREATE MODEL `dba_brasil.ml_exemplo`
OPTIONS(model_type='logistic_reg') AS
SELECT
  IF(totals.transactions IS NULL, 0, 1) AS label,
  IFNULL(device.operatingSystem, "") AS os,
  device.isMobile AS is_mobile,
  IFNULL(geoNetwork.country, "") AS country,
  IFNULL(totals.pageviews, 0) AS pageviews
FROM
  `bigquery-public-
  data.google_analytics_sample.ga_sessions_*`
WHERE
  _TABLE_SUFFIX BETWEEN '20160801' AND '20170630'
```





A tabela é dividida em segmentos/partições baseada em algum critério.



Tabelas podem ser particionadas por:

- Unidade de Tempo (Date - Timestamp - Datetime)
- Tempo de Ingestão (Dia, Hora)
- Integer (Range)

Observações:

- Melhora o desempenho das consultas.
- A chave do particionamento deve ser bem definida.
- Defina prazo de validade para as partições.
- Redução de custo. (FULL COLUMN SCAN).

```
CREATE TABLE
  mydataset.newtable (transaction_id INT64,
  transaction_date DATE)
PARTITION BY transaction_date
AS (
  SELECT
    transaction_id, transaction_date
  FROM
    mydataset.mytable
);
```

```
bq mk \
-t \
--schema
'ts:TIMESTAMP,qtr:STRING,sales:FLOAT' \
--time_partitioning_field ts \
--time_partitioning_type HOUR \
--time_partitioning_expiration 259200
\
mydataset.mytable
```



Você pode especificar um intervalo de sufixos:

```
SELECT coluna from dataset.tabela  
WHERE  
_TABLE_SUFFIX BETWEEN '2001' AND '2003'
```

```
SELECT coluna from dataset.tabela  
WHERE  
_TABLE_SUFFIX = '2001'
```

```
SELECT coluna from dataset.tabela  
WHERE  
_TABLE_SUFFIX < '2001'
```

➤ As tabelas precisam ter nomes parecidos, como PrefixoSufixo

➤ Exemplo: PART é o prefixo, o ano é sufixo





- Permite execução de queries baseadas em um snapshot de até 7 dias no passado.
- Ideal para utilizar em recuperação de dados alterados ou removidos acidentalmente
- É necessário adicionar a cláusula **FOR SYSTEM\_TIME AS OF**

```
SELECT *  
FROM `<projeto>.<dataset>.<tabela>`  
FOR SYSTEM_TIME AS OF  
TIMESTAMP_SUB(CURRENT_TIMESTAMP, INTERVAL 2  
HOURS)
```

```
create table  
`<projeto>.<dataset>.<tabela_restaurada>`  
as  
SELECT *  
FROM `<projeto>.<dataset>.<tabela>`  
FOR SYSTEM_TIME AS OF  
TIMESTAMP_SUB(CURRENT_TIMESTAMP, INTERVAL 2  
HOURS)
```



- Evite `SELECT*` , utilize somente as colunas necessárias nas queries.
- Utilize a opção `PREVIEW` para visualização dos dados (é grátis).
- Confira o quanto sua consulta será cobrada.
- Evite tabelas externas.
- Utilize Partições e/ou Clustering sempre que aplicável.
- Clausula `LIMIT` é sempre bem-vinda.



## Processamento de consultas:

- **Preço sob demanda:** cobrado pelo número de bytes processados por cada consulta efetuada.
- **Preço Fixo:** Você compra slots, que são CPU's virtuais - uma capacidade de processamento dedicada que você pode utilizar para executar e processar suas consultas

## Armazenamento:

- Preço sob dados armazenados no BigQuery

Operação	Preços	Detalhes
----------	--------	----------

Consultas (sob demanda)

O primeiro terabyte (1 TB) por mês é gratuito.

### Detalhes do preço

O preço de armazenamento tem como base a quantidade de dados armazenados nas suas tabelas quando eles são descompactados. O tamanho dos dados é calculado com base nos tipos de dados de cada coluna. Para uma explicação detalhada sobre como o tamanho dos dados é calculado, consulte [Cálculo do tamanho dos dados](#).

Os preços de armazenamento são rateados por MB, por segundo. Por exemplo, ao armazenar:

- 100 MB pela metade de um mês, você paga US\$ 0,001 (um décimo de centavo);
- 500 GB pela metade de um mês, você paga US\$ 5;
- 1 TB por um mês completo, você paga US\$ 20.





## Preços de ingestão de dados

O BigQuery oferece dois modos de ingestão de dados:

- **Carregamento em lote:** carrega os dados de origem em uma ou mais tabelas do BigQuery em uma única operação.
- **Streaming:** faz streaming de dados de um registro de cada vez ou em lotes pequenos.

Para mais informações sobre qual modo escolher, consulte [Introdução ao carregamento de dados](#).

Veja os seguintes dados: (),

Mensal

Operação	Preços	Detalhes
Carregamento em lote	Grátis usando o pool de slots compartilhado.	Os clientes podem escolher <a href="#">preços fixos</a> para a capacidade garantida. Quando os dados são carregados no BigQuery, a cobrança é feita.
Inserções por streaming ( <code>tabledata.insertAll</code> )		A cobrança é realizada conforme o número de linhas inseridas com sucesso. Cada linha é calculada usando um tamanho mínimo de 1 KB.
API BigQuery Storage Write		Os primeiros 2 TB por mês são gratuitos.

# Ingestão de dados com Oracle GoldenGate + Google BigQuery

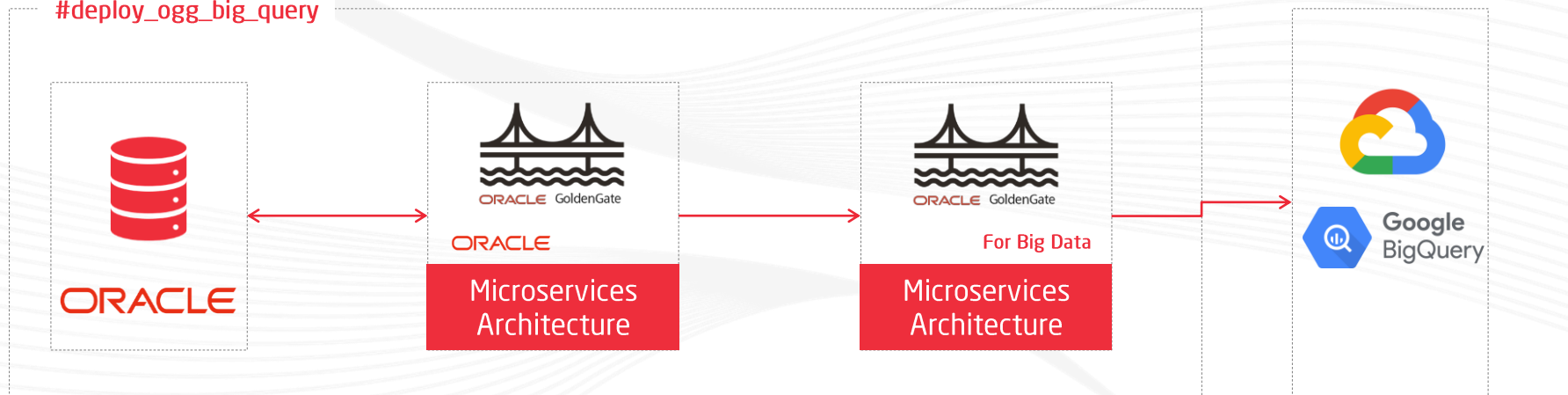


# Ambiente de Replicação

## GoldenLabs



#deploy\_ogg\_big\_query



- Data Warehouse
- **Big Data**
- Data Lake
- **Replicação Real-Time**
- Engenharia de Dados

# Perguntas?

**DBA BRASIL**  
DATA & CLOUD



Gilson Martins



Rafael Milanez



# Obrigado!