



Universidade Federal  
de Campina Grande

Universidade Federal de Campina Grande - UFCG  
Programa de Pós-Graduação em Ciência da Computação (PPGCC)  
Centro de Engenharia Elétrica e Informática – CEEI

Disciplina: Banco de Dados  
Prof. Dr. Carlos Eduardo Santos Pires

# Sistema NoSQL

## Apache HBase



Nisston Moraes Tavares de Melo

# Roteiro

- O que é o HBase?
- O HBase vs RDBMS
- Características do HBase
- O armazenamento orientado por coluna
- Componentes da arquitetura
- Pontos negativos
- Pontos positivos
- Referências

# Soluções para os BIG DATA

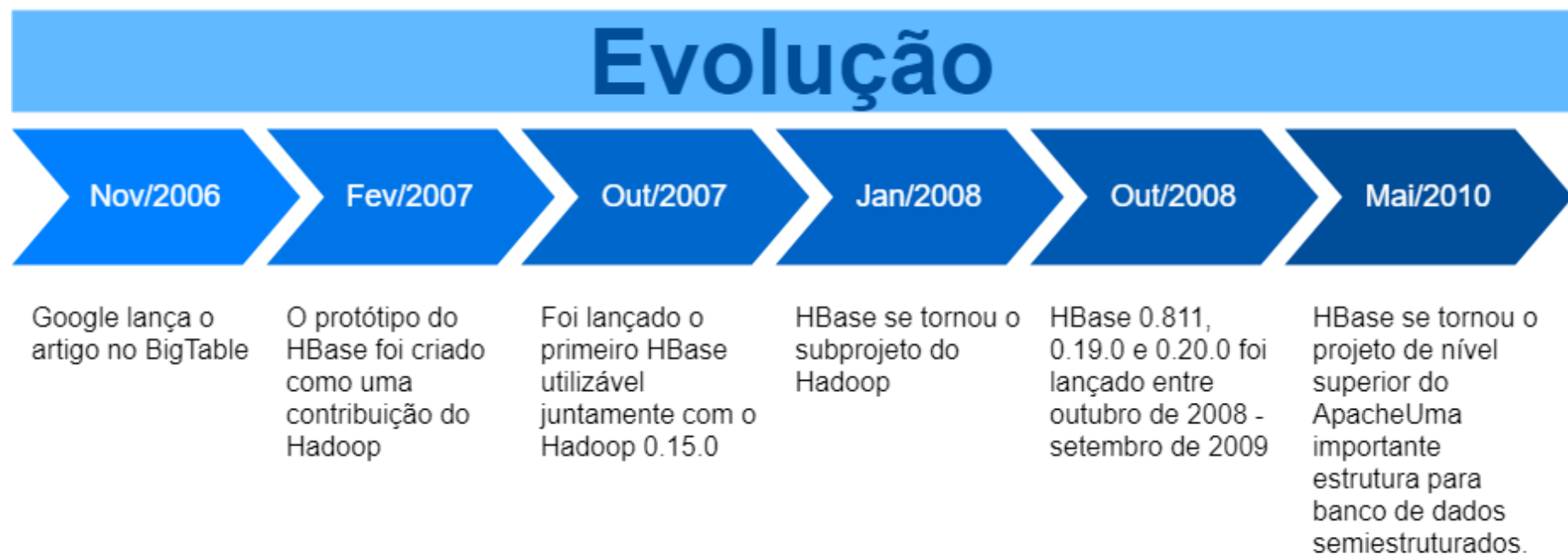


**Flink 1.0**

# O que é HBase?

- Foi desenvolvido com base no [BigTable do Google](#) que roda sobre o [Hadoop](#);
- É um sistema orientado a colunas;
- É uma aplicação open source;
- É um projeto horizontalmente escalável;
- Banco de dados NoSQL e escrito em Java.

# Um pouco da história do HBase



[Clique aqui para ver o artigo original](#): CHANG, Fay et al. Bigtable: A distributed storage system for structured data. **ACM Transactions on Computer Systems (TOCS)**, v. 26, n. 2, p. 1-26, 2008.

# Para que foi criado?

- Ele foi criado para dar acesso aleatório, totalmente estável e em tempo real a tabelas com bilhões de linhas e milhões de colunas.

# Empresas utilizando o HBase

- Hortonworks
- CapitalOne
- Facebook
- HubSpot
- Bank of America
- JPMorgan

Bank of America®



facebook®

HubSpot



J.P.Morgan



# Um exemplo de uso do HBase

- Sistema de telefonia da China
  - Geração de bilhões de registros e detalhes de chamadas.
  - Solução:
    - O HBase armazena bilhões de linhas de registros com detalhes de chamadas.

# Aplicações do HBase

- A indústria médica
- E-commerce
- Sports
- Bancos
- Redes sociais
- Setor financeiro

# HBase vs RDBMS

HBase	RDBMS
Não apresenta um esquema fixo para as estruturas de dados. Define apenas famílias de colunas.	É preciso apresentar um esquema fixo que descreva a estrutura das tabelas.
Funciona bem com dados que apresentam estruturas e semiestruturados.	Só funciona bem com dados estruturados.
Ele pode trabalhar com dados desnormalizados (podendo conter valores ausentes ou NA)	Pode armazenar apenas dados normalizados.
Construído para trabalhar com tabelas largas que podem ser escaladas horizontalmente.	Construído para tabelas finas que são difíceis de escalar.

RDBMS - Relational Database Management Systems

# Características do HBase

- Escalável
- Suporte a falhas automáticas
- Leitura e escrita consistente
- Oferecer uma API Java para acesso do cliente
- Modelo de dados flexível

# Como ele é sustentado?

- O Apache HBase é sustentado de modo nativo no [Amazon EMR](#) e você pode criar de maneira rápida e fácil clusters gerenciados do Apache HBase por meio do Console de [Gerenciamento da AWS](#), da [AWS CLI](#) ou da API do Amazon EMR.

# O armazenamento orientado por coluna

ID da linha	Família 1		Família 2			
	Coluna 1	Coluna 2	Coluna 3	Coluna 4	Coluna 5	Coluna 6
Linha 1						
Linha 1						
Linha 1						

# O armazenamento orientado por coluna

ID da linha	Família - A		Família - B	
	Nome	Fone	URL	
312010	Maria da Silva	98845-7879	1459088892829	http://exemplo.com/page1.html
312012	Pedro Marques		1459088892833	http://exemplo.com/page2.html
312017	João Paulo	95656-2325		

# Estrutura dos dados

```
1 {  
2   //...  
3   "aaaa" : {  
4     "bbb" : {  
5       "foo" : {  
6         15: "yy",  
7         4: "m"  
8       }  
9       "bar" : { ... }  
10    },  
11    "B" : {  
12      "ggg" : { ... }  
13    }  
14  },  
15  "aaaab" : {  
16    "bb" : {  
17      "foo" : { ... },  
18      "bar" : { ... },  
19      "joe" : { ... }  
20    },  
21    "B" : {  
22      "ccc" : { ... }  
23    }  
24  }  
25 },  
26 // ...
```

Tabela

Linha

Família de colunas

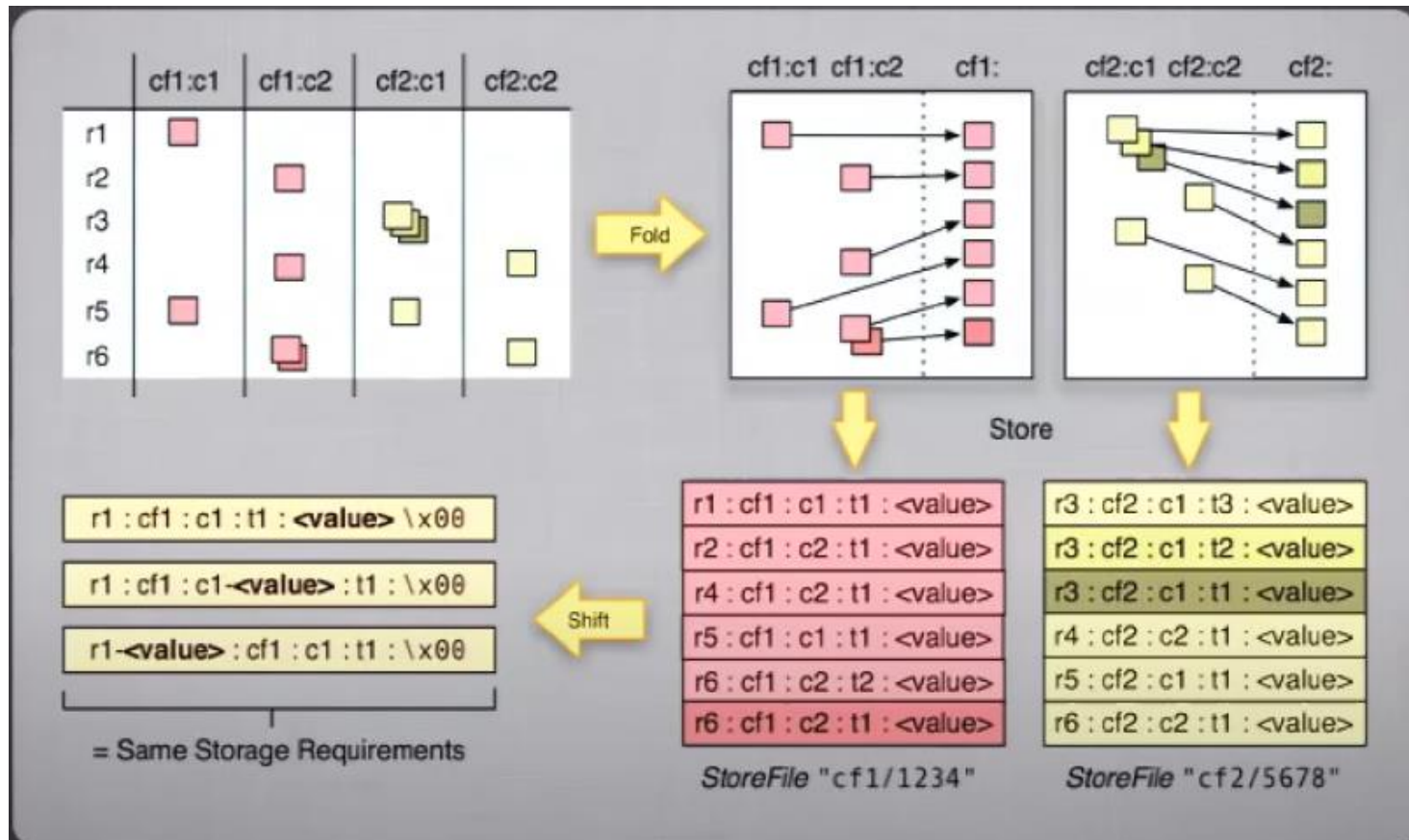
Coluna

Timestamp, value

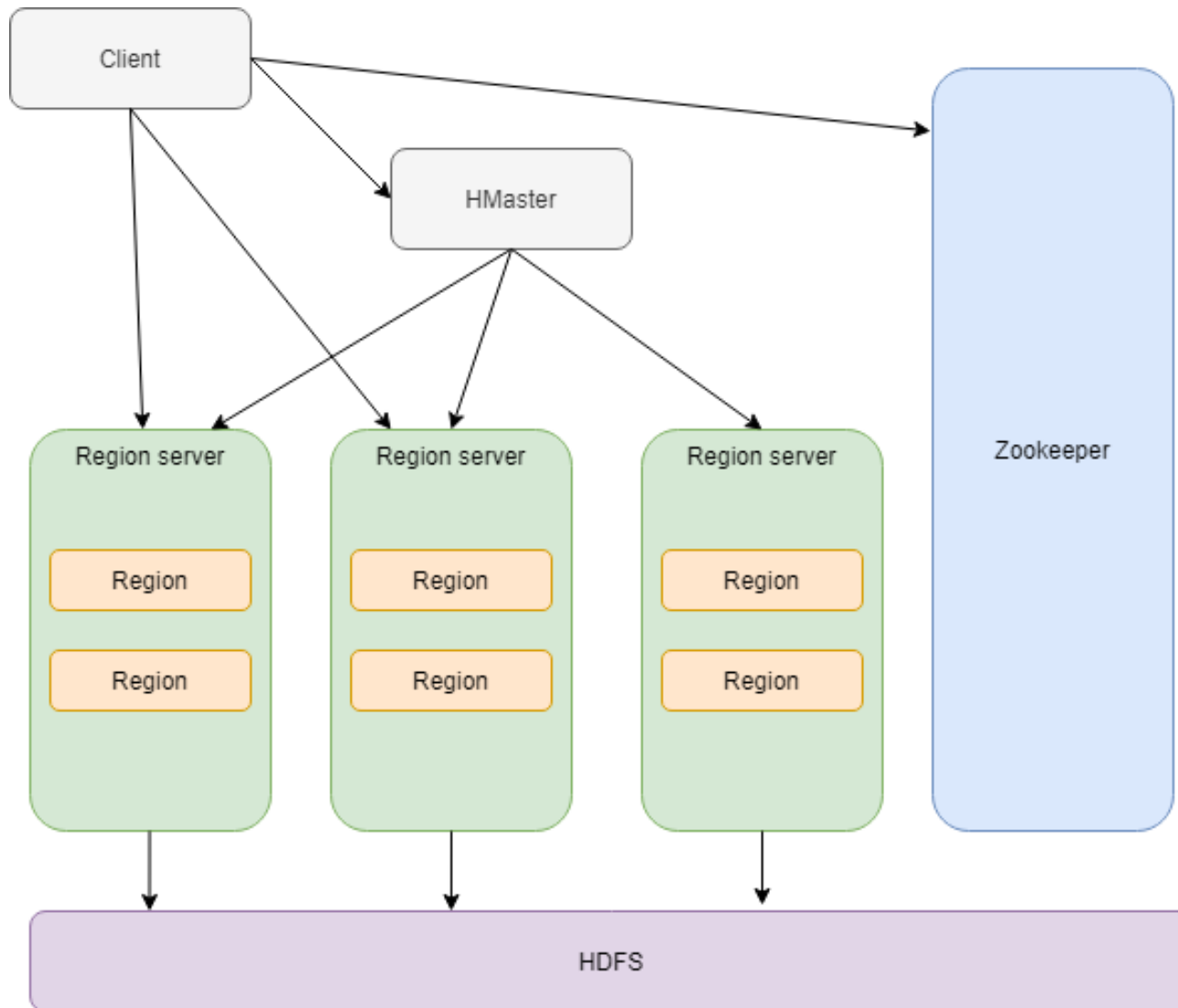
(Table, RowKey, Family, Column, Timestamp) -> Value



# A arquitetura de armazenamento do HBase



# Componentes da arquitetura



# Tipos de acesso HBase

- Writes (put = update)
- Rando reads (get)
- Sequential reads (scan)

# Pontos negativos

- Configuração e manutenção complexas;
- Não oferece índice secundário;
- Não disponibiliza operações de transações;
- É preciso construir um projeto de estrutura dos dados complexo.

# Pontos positivos

- Sua estrutura é distribuída;
- É escalável (fragmentação automática);
- Construído na pilha [Hadoop](#);
- Oferece interação com [Big Data](#);
- Alto desempenho para escrita e leitura;
- Sem [SPOF](#) (single point of failure);
- Tolerante a falhas, sem perda de dados;
- Tem uma comunidade ativa na net;

# CRUD

- Realizando as operações básicas no HBase
  - Criação
  - Leitura
  - Seleção
  - Exclusão

# Montando uma instância

- <https://cloud.google.com>

The screenshot shows the Google Cloud Platform console dashboard. The top navigation bar includes the Google Cloud Platform logo, the project name 'My Project 63260', and a search bar. Below the navigation bar, there are tabs for 'PAINEL', 'ATIVIDADE', and 'RECOMENDAÇÕES'. A red arrow points to the project name 'My Project 63260' in the navigation bar. The main content area displays project information and API usage.

**Informações do projeto**

- Nome do projeto: My Project 63260
- ID do projeto: upheld-terminus-295318
- Número do projeto: 913489897618

[ADICIONAR PESSOAS AO PROJETO](#)

[Acessar as configurações do projeto](#)

**API APIs**

Solicitações (solicitações/s)

No data is available for the selected time frame.

[Ir para a visão geral de APIs](#)

# Criando uma instância

< > ↺ | console.cloud.google.com/bigtable/create-instance

Google Cloud Platform My Project 63260 Pesquisar produtos e recursos

← Criar uma instância

A Cloud Bigtable instance is a container for your clusters. [Learn more](#)

**1 Nomear sua instância**

Nome da instância \*  
sistemahbase

Somente para fins de exibição

ID da instância \*  
sistemahbase

O ID é permanente

CONTINUAR

**2 Selecionar seu tipo de armazenamento**


**3 Configurar seu primeiro cluster**

US\$ 494,00 por mês (estimado)  
Trata-se de US\$ 0,69 uma hora com 1000 GB armazenado.

▼ MOSTRAR DETALHES

▼ MOSTRAR OPÇÕES AVANÇADAS

CRIAR CANCELAR





# Selecionando tipo de armazenamento

Google Cloud Platform

My Project 63260

Pesquisar produtos e recursos

← Criar uma instância

A Cloud Bigtable instance is a container for your clusters. [Learn more](#)

✓ Nomear sua instância

2 Selecionar seu tipo de armazenamento

3 Configurar seu primeiro cluster

US\$ 638,00 por mês (estimado)

Trata-se de US\$ 0,89 uma hora com 1000 GB armazenado.

✓ MOSTRAR DETALHES

☐ HDD

Latência maior em leituras aleatórias. Apresenta bom desempenho em verificações e geralmente é usada para análises em lote, como machine learning ou extração de dados.

☒ SSD

Menor latência e mais linhas lidas por segundo. Geralmente usada para casos de uso com exibição em tempo real, como veiculação de anúncios e recomendações de app para dispositivos móveis.

CONTINUAR

# Configuração de cluster e nós.

Google Cloud Platform

My Project 63260

Pesquisar produtos e recursos

← Criar uma instância

3 Configurar seu primeiro cluster

A cluster handles application requests for an instance. It contains nodes which determine your cluster's performance and storage limit.

Additional clusters can be added at any time.

### Selecionar um código de cluster

O ID é permanente

Cluster ID \*  
sistemahbase-c1

### Selecione um local

A escolha é permanente. Determina onde os dados de cluster são armazenados. Para reduzir a latência e aumentar a capacidade, armazene seus dados próximo aos serviços necessários. [Saiba mais](#)

Region \*

▼

Zone

▼

### Alocar nós

A contagem de nós pode ser atualizada a qualquer momento para atender às necessidades do seu cluster relativo à capacidade e ao armazenamento de dados, além das linhas lidas por segundo. Para melhorar o desempenho das instâncias, mantenha a utilização da CPU do cluster abaixo do limite recomendado de acordo com sua [política de roteamento do perfil de uso](#). [Entre em contato](#) se precisar aumentar a cota de nós. [Saiba mais](#)

Nodes \*  
1

US\$ 638,00 por mês (estimado)

Trata-se de US\$ 0,89 uma hora com 1000 GB armazenado.

▼ MOSTRAR DETALHES

# Instância criada

The screenshot shows the Google Cloud Platform console interface. At the top, there's a navigation bar with the Google Cloud Platform logo, the project name 'My Project 63260', and a search bar. Below this, the 'Bigtable' section is active, showing 'Instâncias' (Instances) with a '+ CRIAR INSTÂNCIA' button. A descriptive text about Cloud Bigtable is present, followed by a message to activate billing. The main content area displays a table of instances. A large red arrow points to the instance named 'sistemahbase'.

Google Cloud Platform | My Project 63260 | Pesquisar produtos e recursos

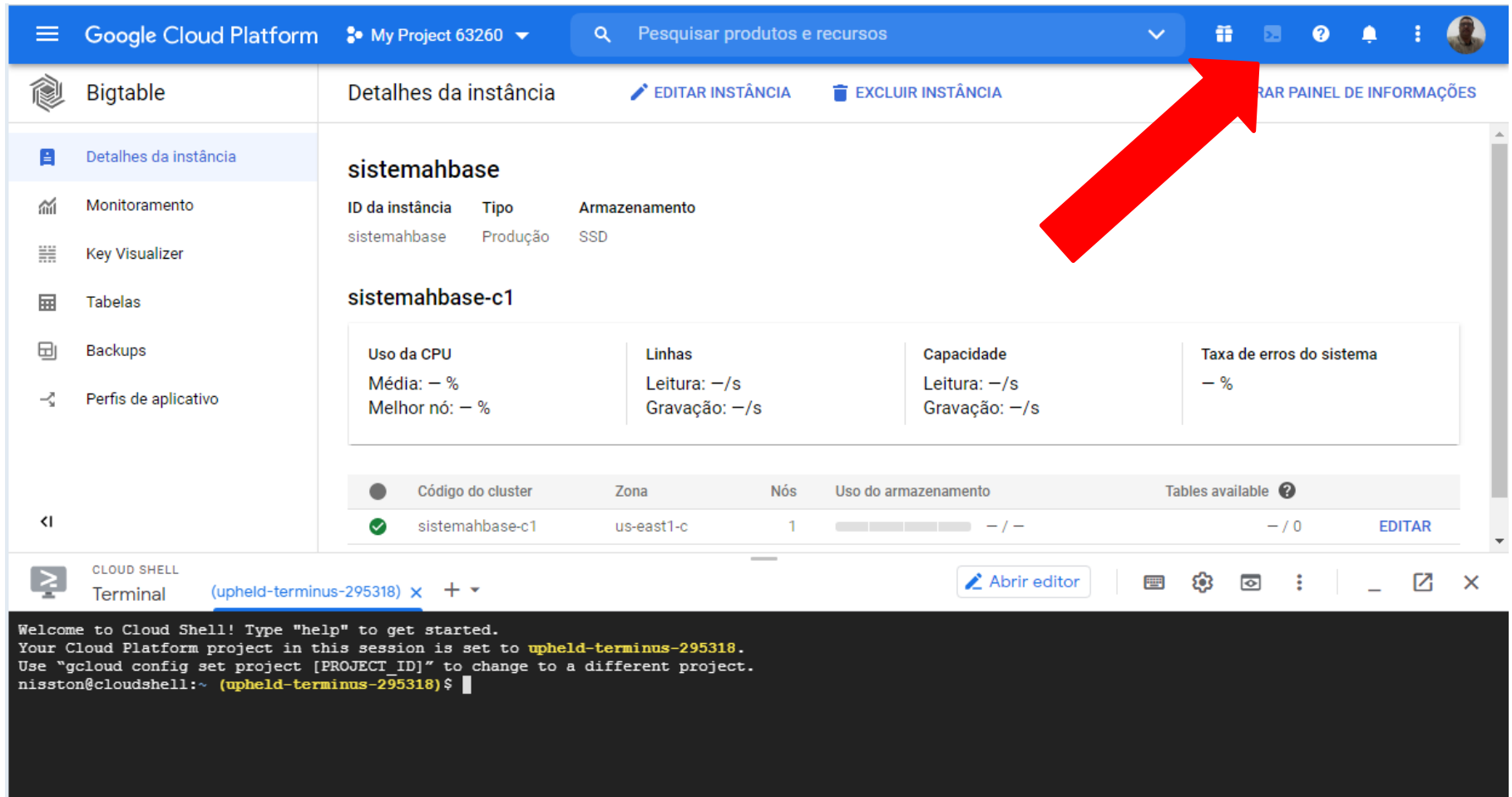
Bigtable | Instâncias | + CRIAR INSTÂNCIA

O Cloud Bigtable é um banco de dados NoSQL de coluna ampla e totalmente gerenciada que oferece baixa latência e replicação para alta disponibilidade. É possível fornecer instâncias do Cloud Bigtable para sua carga de trabalho e usar o cliente HBase do Bigtable para desenvolver aplicativos, bem como outras ferramentas de Big Data de código aberto padrão. [Learn more](#)

Ative o faturamento para usar o Bigtable neste projeto.

Selec	ID da instância	Nome da instância	Perfis de aplicativo	Zonas	Nós	Uso do armazenamento
<input type="checkbox"/>	✓	sistemahbase	default Adicionar	us-east1-c Adicionar	1	

# Ativando o (console) Cloud Shell



The screenshot displays the Google Cloud Platform console interface. At the top, the navigation bar includes the Google Cloud Platform logo, the current project 'My Project 63260', a search bar, and various utility icons. The main content area shows the details for an instance named 'sistemahbase'. A red arrow points to the 'Ativar' (Activate) button in the top right corner of the instance details page.

**Google Cloud Platform** My Project 63260 Pesquisar produtos e recursos

**Bigtable** Detalhes da instância [EDITAR INSTÂNCIA](#) [EXCLUIR INSTÂNCIA](#) [ATIVAR PAINEL DE INFORMAÇÕES](#)

**sistemahbase**

ID da instância: sistemahbase Tipo: Produção Armazenamento: SSD

**sistemahbase-c1**

Uso da CPU	Linhas	Capacidade	Taxa de erros do sistema
Média: — %	Leitura: —/s	Leitura: —/s	— %
Melhor nó: — %	Gravação: —/s	Gravação: —/s	

Código do cluster	Zona	Nós	Uso do armazenamento	Tables available
sistemahbase-c1	us-east1-c	1	— / —	— / 0

**CLOUD SHELL** Terminal (upheld-terminus-295318) [Abrir editor](#)

```
Welcome to Cloud Shell! Type "help" to get started.
Your Cloud Platform project in this session is set to upheld-terminus-295318.
Use "gcloud config set project [PROJECT_ID]" to change to a different project.
niston@cloudshell:~ (upheld-terminus-295318) $
```

# Parâmetros para o HBase

78 Comandos de configuração

79

80 `sudo apt-get update`

81 `sudo apt-get install openjdk-8-jdk-headless`

82 `export JAVA_HOME=$(update-alternatives --list java | tail -1 | sed -E 's/\/bin\/java//')`

83

84 Clonando o repositório

85

86 `git clone https://github.com/GoogleCloudPlatform/cloud-bigtable-examples.git`

87 `cd cloud-bigtable-examples/quickstart`

88


89 Iniciando o HBase


90

91 `./quickstart.sh`

92

	Código do cluster	Zona	Nós	Uso do armazenamento	Tables available ?
<	✓ sistemahbase-c1	us-east1-c	1	— / 2,5 TB	— / 0

 CLOUD SHELL  
Terminal (upheld-terminus-295318) x + ▾

 Abrir editor |     — [

```
W: Failed to fetch https://packages.cloud.google.com/apt/dists/cloud-sdk-buster/InRelease Temporary failure resolving 'packages.cloud.google.com'
W: Some index files failed to download. They have been ignored, or old ones used instead.
nisston@cloudshell:~ (upheld-terminus-295318)$ udo apt-get install openjdk-8-jdk-headless
-bash: udo: command not found
nisston@cloudshell:~ (upheld-terminus-295318)$ export JAVA_HOME=$(update-alternatives --list java | tail -1 | sed -E 's/\/bin\/java//')
nisston@cloudshell:~ (upheld-terminus-295318)$ git clone https://github.com/GoogleCloudPlatform/cloud-bigtable-examples.git
fatal: destination path 'cloud-bigtable-examples' already exists and is not an empty directory.
nisston@cloudshell:~ (upheld-terminus-295318)$ cd cloud-bigtable-examples/quickstart
nisston@cloudshell:~/cloud-bigtable-examples/quickstart (upheld-terminus-295318)$
```

# Comandos básicos

- Criando tabela
- Criando registros
- Selecionando famílias de colunas
- Selecionando colunas
- Apagando tabela

```
160 Criando tabela
161
162 create 'cliente', 'cf1'
163
164 Criando registros na tabela
165 put 'cliente', 'r1', 'cf1:c1', 'maria'
166 put 'cliente', 'r1', 'cf1:c2', '222'
167 put 'cliente', 'r1', 'cf1:c3', 'http://www.empresa.com.br'
168 put 'cliente', 'r1', 'cf1:c4', '1000'
169
170 Criando um novo registro da mesma coluna
171 put 'cliente', 'r2', 'cf1:c1', 'pedro'
172 put 'cliente', 'r2', 'cf1:c2', '44444'
173 put 'cliente', 'r2', 'cf1:c3', 'http://www.empresaA.com.br'
174 put 'cliente', 'r2', 'cf1:c4', '200'
175
176 recuperando dados das tabelas
177 scan 'cliente'
178
179 scan 'cliente', {COLUMNS => ['cf1:c1','cf1:c4']}
180
181 scan 'cliente', {COLUMNS => ['cf1:c1','cf1:c4'], LIMIT=> 1}
182
183 get 'cliente', 'r1'
184
185 get 'cliente', {COLUMNS => ['cf1:c1','cf1:c4']}
186
187 delete 'cliente', 'r1', 'cf1:c4'
```

# Comandos básicos

```
hbase(main):001:0> #Criando tabela
hbase(main):002:0* create 'cliente', 'cf1'
0 row(s) in 3.2300 seconds

=> Hbase::Table - cliente
hbase(main):003:0>
hbase(main):004:0* #Criando registros na tabela
hbase(main):005:0* put 'cliente', 'r1', 'cf1:c1', 'maria'
0 row(s) in 0.1590 seconds

hbase(main):006:0> put 'cliente', 'r1', 'cf1:c2', '222'
0 row(s) in 0.0880 seconds

hbase(main):007:0> put 'cliente', 'r1', 'cf1:c3', 'http://www.empresa.com.br'
0 row(s) in 0.0960 seconds

hbase(main):008:0> put 'cliente', 'r1', 'cf1:c4', '1000'
0 row(s) in 0.0710 seconds

hbase(main):009:0> list
TABLE
cliente
1 row(s) in 0.0580 seconds

=> ["cliente"]
hbase(main):010:0> █
```

```
hbase(main):023:0> disable 'cliente'
2020-11-15 12:53:19,634 WARN [org.jruby.Main.main()]
0 row(s) in 0.1260 seconds

hbase(main):024:0> drop 'cliente'
0 row(s) in 0.0700 seconds

hbase(main):025:0> list
TABLE
0 row(s) in 0.0160 seconds

=> []
hbase(main):026:0> █
```

```
hbase(main):015:0* scan 'cliente'
ROW                                COLUMN+CELL
r1                                 column=cf1:c1, timestamp=1605444433129, value=maria
r1                                 column=cf1:c2, timestamp=1605444433305, value=222
r1                                 column=cf1:c3, timestamp=1605444433498, value=http://www.empresa.com.br
r1                                 column=cf1:c4, timestamp=1605444433612, value=1000
1 row(s) in 0.2050 seconds

hbase(main):016:0>
hbase(main):017:0* scan 'cliente', {COLUMNS => ['cf1:c1','cf1:c4']}
ROW                                COLUMN+CELL
r1                                 column=cf1:c1, timestamp=1605444433129, value=maria
r1                                 column=cf1:c4, timestamp=1605444433612, value=1000
1 row(s) in 0.0830 seconds

hbase(main):018:0> █
```

# Referências

- FERREIRA, L. M.; SOUZA, SNA. **Benchmark of NoSQL Data Base Oriented to Column Family** **Análise de Desempenho de Banco de Dados NoSQL Orientado a Família de Colunas**.
- CHANG, Fay et al. Bigtable: A distributed storage system for structured data. **ACM Transactions on Computer Systems (TOCS)**, v. 26, n. 2, p. 1-26, 2008.
- <https://hbase.apache.org/>
- <https://aws.amazon.com/pt/emr/details/hbase>
- <https://abr.io/hbase-intro>
- <https://pinboard.in/u:lfcipriani/t:hbase>
- <https://learnhbase.net/2013/03/02/hbase-shell-commands/>
- Vídeos:
  - <https://www.youtube.com/watch?v=V1fXSCASVDc>
  - <https://youtu.be/-1g-MYuKpEo>
  - <https://www.infoq.com/br/presentations/conhecendo-apache-hbase/>
  - <https://www.youtube.com/watch?v=CjRkEywm1go&t=1010s>



Obrigado!