



Semantix

Hbase - Básico

Aula 8

Quem sou eu?

Eu sou Rodrigo Augusto Rebouças.

Engenheiro de dados da Semantix
Instrutor do Semantix Academy

Você pode me encontrar em:
rodrigo.augusto@semantix.com.br



An abstract graphic on the left side of the slide. It features two overlapping profiles of human heads facing each other. The profiles are filled with a blue, wavy, line-art pattern. Inside the profiles, there are faint, semi-transparent images of data visualizations, including bar charts and line graphs. The background is white.

Armazenamento de Dados

Introdução

Armazenamento de Dados

- <https://db-engines.com/>



- HDFS

- MongoDB



- Redis

- Elasticsearch



- Cassandra



- Neo4j

- HBase



- Kudu

Armazenamento Dados

- HDFS
 - Filesystem
 - Armazenamento Estático
- HBase
 - Armazenamento em Real Time
 - Construído sobre o HDFS ("StoreFiles")
 - Base de dados não relacional



Apache HBase

- Banco de Dados Hadoop
 - Armazenamento distribuído
 - Escalável
- Uso para acesso de leitura / gravação aleatório em tempo real
- Banco de dados não relacional
 - "NoSQL"
 - Orientado a Colunas
 - Schema da tabela são famílias de colunas
 - Pares de chave-valor
 - Internamente usa tabela Hash

- 2006 - Google publicou sobre um sistema de armazenamento distribuído para dados estruturados
- 2007 - Criação do Apache HBase
 - Implementação open-source da arquitetura Bigtable
- 2008 – Tornou um subprojeto do Hadoop

Bigtable: A Distributed Storage System for Structured Data

Fay Chang, Jeffrey Dean, Sanjay Ghemawat, Wilson C. Hsieh, Deborah A. Wallach
Mike Burrows, Tushar Chandra, Andrew Fikes, Robert E. Gruber
{fay,jeff,sanjay,wilsonh,kerr,m3b,tushar,fikes,gruber}@google.com

Google, Inc.

Abstract

Bigtable is a distributed storage system for managing structured data that is designed to scale to a very large size: petabytes of data across thousands of commodity servers. Many projects at Google store data in Bigtable, including web indexing, Google Earth, and Google Finance. These applications place very different demands on Bigtable, both in terms of data size (from URLs to web pages to satellite imagery) and latency requirements (from backend bulk processing to real-time data serving). Despite these varied demands, Bigtable has successfully provided a flexible, high-performance solution for all of

achieved scalability and high performance, but Bigtable provides a different interface than such systems. Bigtable does not support a full relational data model; instead, it provides clients with a simple data model that supports dynamic control over data layout and format, and allows clients to reason about the locality properties of the data represented in the underlying storage. Data is indexed using row and column names that can be arbitrary strings. Bigtable also treats data as uninterpreted strings, although clients often serialize various forms of structured and semi-structured data into these strings. Clients can control the locality of their data through careful

<https://research.google.com/archive/bigtable.html>



Arquitetura



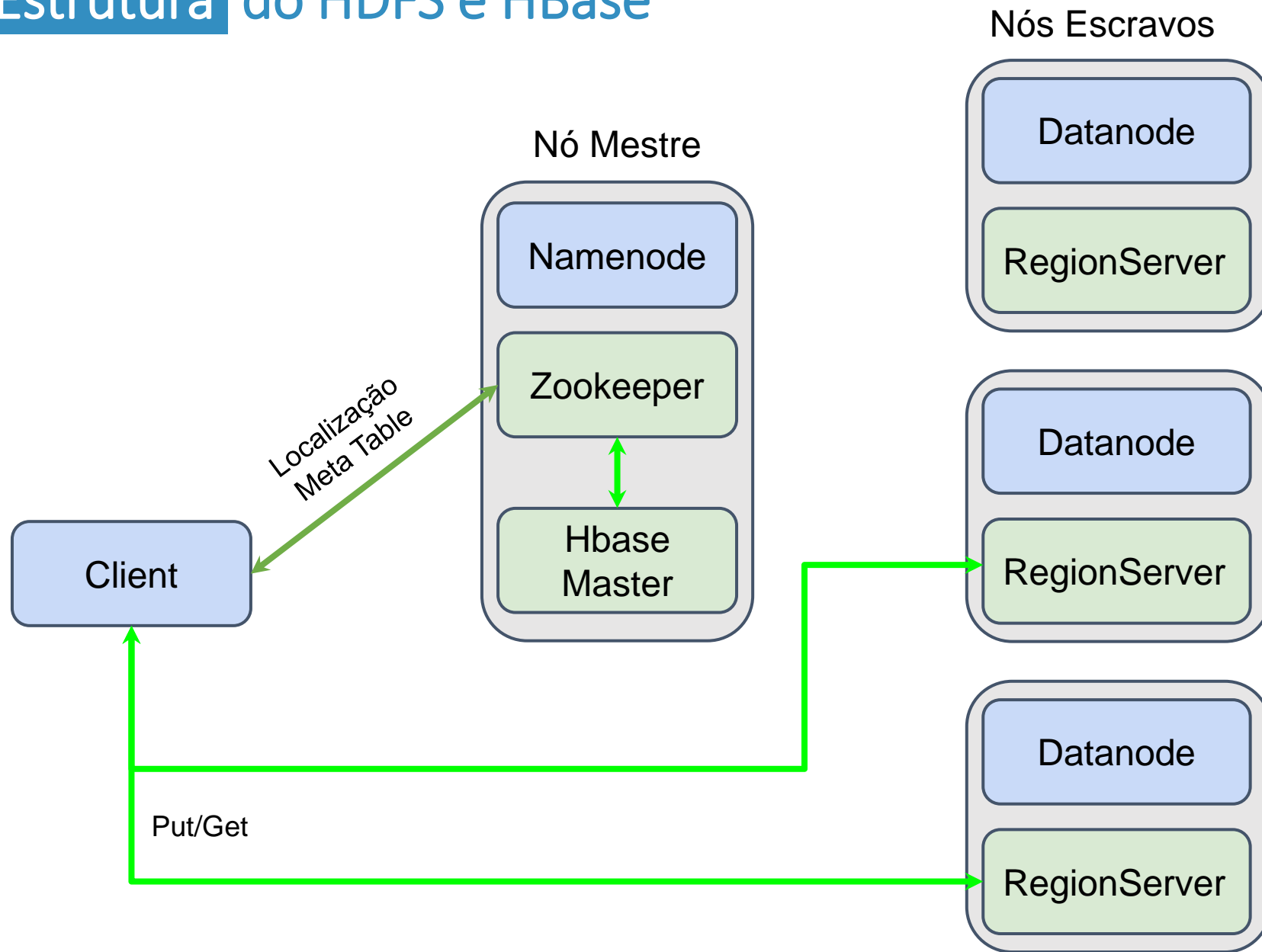
HBase Funcionamento

- Dados são distribuídos automaticamente no cluster
 - Chave serve para distribuir os dados nos nós
 - Cada nó armazena um subconjunto de dados (RegionServer)

HBase Componentes

- RegionServer
 - Tabela são divididas por região
- HBase Master
 - Gerenciar as RegionServer
- Zookeeper
 - Monitoramento e Coordenação dos HBase Master
 - Arquivos de Configurações
- HDFS
 - Namenode/Standby Namenode
 - Datanode

Estrutura do HDFS e HBase





Estrutura dos Dados



Modelo Dados

- Tabela
 - Organizar os dados
- Chave da linha
 - Identificador da linha que armazena os dados
- Família da coluna
 - Colunas são agrupadas em famílias
- Nome da coluna
 - Identificador da coluna
- Versão
 - valores em célula são controlados pelo número da versão (timestamp)
- Célula (Chave -> Valor)
 - Id linha, família col., coluna e timestamp = valor



Comandos Básicos HBase

HBase Acesso

- Acessar através do HBase shell ou API Java
- HBase shell
 - Script que auxilia a manipulação do HBase por linha de comando
 - Inicializar
 - `$ hbase shell`

Comandos Gerais

- status
 - Status do HBase
 - Número de servidores
- version
 - Versão do HBase
- table_help
 - Ajuda sobre comandos de tabelas
- whoami
 - Informações sobre o usuário
- help “comando”
 - Ajuda sobre o comando

Comandos Manipulação de Tabela

- create
- list
- disable
- is_disabled
- enable
- is_enabled
- describe
- alter
- exists
- drop
- drop_all

Comandos Manipulação dos dados da Tabela

- put
- get
- delete
- deleteall
- scan
- count
- truncate



Manipulação de Tabelas



Comandos Criação de Tabela

- Criar tabela

- Definir nome das famílias de coluna

- Sintaxe

- `create 'nomeTabela',{NAME=>'família'},{NAME=>'família'}`
- `create 'nomeTabela', 'família','família'`

- Ex.

```
hbase> create 'clientes', {NAME=>'endereco', VERSIONS=>2}, {NAME=>'pedido'}
```

```
hbase> create 'clientes', 'endereco', 'pedido'
```


Comandos Opções de Tabela

- Listar todas as tabelas
 - `hbase> list`
- Estrutura da tabela
 - `describe 'nomeTabela'`
 - Ex. `hbase> describe 'clientes',`
- Deletar tabela (precisa desabilitar)
 - `drop 'nomeTabela' ou drop_all 'c.*'`
 - Ex. `hbase> drop 'clientes'`
- Verificar se a tabela existe
 - `hbase> exists 'nomeTabela'`
 - Verificar tabelas desabilitadas

Comandos Habilitar e Excluir Tabela

- Desabilitar tabela
 - `hbase> disable 'nomeTabela'`
 - `hbase> scan 'nomeTabela'` (erro)
 - `hbase> disable_all 'r.*'`
- Verificar se uma tabela está desabilitada
 - `hbase> is_disabled 'nomeTabela'`
- Habilitar tabela
 - `hbase> enable 'nomeTabela'`
- Verificar se uma tabela está habilitada
 - `hbase> is_enable 'nomeTabela'`
- Desabilitar e recriar tabela
 - `hbase> truncate 'nomeTabela'`



Inserção e Alteração de Dados

Inserção e Alteração de Dados

○ Inserir dados

- `put 'nomeTabela','chave','família:coluna','valor'`
- Ex.

```
hbase> put 'clientes', 'emiranda', 'endereco:cidade', 'BH'
```

```
hbase> put 'clientes', 'emiranda', 'endereco:estado', 'MG'
```

```
hbase> put 'clientes', 'emiranda', 'pedido:data', '2/2/2016'
```

```
hbase> put 'clientes', 'emiranda', 'pedido:nitem', '1234D'
```

○ Alterar dados

```
hbase> put 'clientes', 'emiranda', 'pedido:nitem', '1234A'
```




Leitura de Dados

- Consultas de Valores, Colunas ou Família de colunas
 - Pela Chave
 - get
 - Pela Tabela
 - scan

Consultar Registros da Tabela

- Consultar valores pela **chave**

- `get 'nomeTabela','chave'`
- Ex.

```
hbase> get 'clientes', 'emiranda'
```

- Consultar valores da chave pela família de coluna

- `get 'nomeTabela','chave',{COLUMNS=>['família']}`

Ex.

```
hbase> get 'clientes', 'emiranda',{COLUMNS=>['endereco']}
```

- Consultar valores da chave pela coluna

- `get 'nomeTabela','chave', {columns=>['família:coluna']}`
- Ex.

```
hbase> get 'clientes', 'emiranda',{COLUMNS=>['endereco:cidade']}
```

Consultar Tabela

- Consultar todas as linhas e colunas da tabela

- scan 'nomeTabela'
- Ex.

hbase> scan 'clientes'

- Consultar todas as linhas da tabela pela família de coluna

- scan 'nomeTabela',{COLUMNS=>['família']}
- Ex.

hbase scan 'clientes', {COLUMNS=>['endereco']}

- Consultar todas as linhas da tabela pela coluna

- scan 'nomeTabela',{COLUMNS=>['família:coluna']}
- Ex.

hbase scan 'clientes', {COLUMNS=>['endereco:cidade']}

Consultar Tabela

- Consultar todas as linhas da tabela pela versão de uma coluna

- scan 'nomeTabela',{COLUMNS=>['família:valor']}
- Ex.

```
hbase> scan 'clientes', {COLUMNS=>['pedido:nitem'], VERSIONS=>5}
```

- Consultar todas as linhas da tabela pela chave

- scan 'nomeTabela',{STARTROW=>'chave', COLUMNS=>['família:valor']}
- Ex.

```
hbase> scan 'clientes', {STARTROW=>'emi', COLUMNS=>['endereco']}
```




Deletar Dados

Deletar Dados

○ Deletar Coluna

- delete 'nomeTabela','chave','família:coluna'
- Ex.

```
hbase> delete 'clientes', 'emiranda', 'endereco:cidade'
```

○ Deletar Família de coluna

- delete 'nomeTabela','chave','família'
- Ex.

```
hbase> delete 'clientes', 'emiranda', 'endereco'
```

○ Deletar uma chave

- deleteall 'nomeTabela','chave'
- Ex.

```
hbase> deleteall 'clientes', 'emiranda'
```



Outras opções com Tabelas

Alterar Tabela

- Alterar a família de coluna da tabela
 - alter 'nomeTabela', {NAME=>'família', VERSIONS=>Nº}
 - Ex.
 - Armazenar mais versões

```
hbase> alter 'clientes', {NAME=>'pedido', VERSIONS=>5}
```

- Alterar a tabela para deletar a família de coluna
 - alter 'nomeTabela', 'delete'=>'família'
 - Ex.

```
hbase> alter 'employee','delete'=>'professional'
```

Contar Dados

- Contar o número de registros na tabela

- count 'nomeTabela'
- Ex.

```
hbase> count 'clientes'
```



Laboratório

Resolução de exercícios



Exercício HBase

1. Criar a tabela 'controle' com os dados

id	produto		fornecedor	
	nome	qtd	nome	estado
1	ram	100	TI Comp	SP
2	hd	50	Peças PC	MG
3	mouse	150	Inf Tec	SP

2. Listar as tabelas e verificar a estrutura da tabela 'controle'
3. Contar o número de registros da tabela 'controle'
4. Alterar a família de coluna produto para 3 versões
5. Alterar a quantidade para 200 do id 2
6. Pesquisar as versões do id 2 da coluna quantidade
7. Excluir os id do estado de SP
8. Deletar a coluna estado da chave 2

9. Pesquisar toda a tabela controle



Semantix

Obrigado!

Alguma pergunta?



Você pode me encontrar em:
rodrigo.augusto@semantix.com.br

GET SMARTER