



Apache Spark Streaming

Daniel Monteiro Valério
Gabriel Mace dos Santos Ferreira
Marcus Vinícius Souza Fernandes

The image features a dark navy blue background. On the left side, there are two overlapping parallelogram shapes. The front one is a vibrant blue, and the one behind it is a light mint green. Both shapes are oriented diagonally, with their longer sides running from the top-left towards the bottom-right. The word "Spark" is written in a clean, white, sans-serif font, positioned to the right of these shapes.

Spark

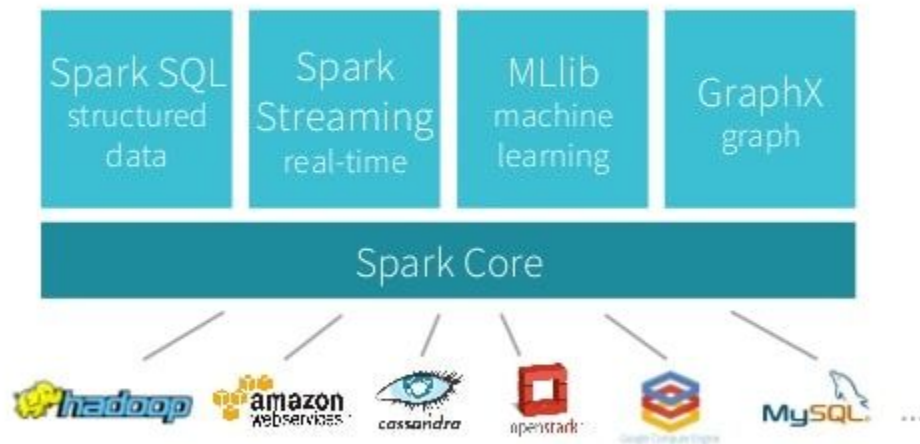


Spark

Ferramenta unificada para o processamento de dados em larga escala. A partir de suas ferramentas principais é possível realizar o processamento de grafos, machine learning e computação incremental e processamento em stream.



A General Engine



The image features a dark navy blue background. On the left side, there are two overlapping geometric shapes: a blue parallelogram and a light green parallelogram, both tilted at an angle. The text 'Spark Streaming' is positioned to the right of these shapes.

Spark Streaming



Spark Streaming

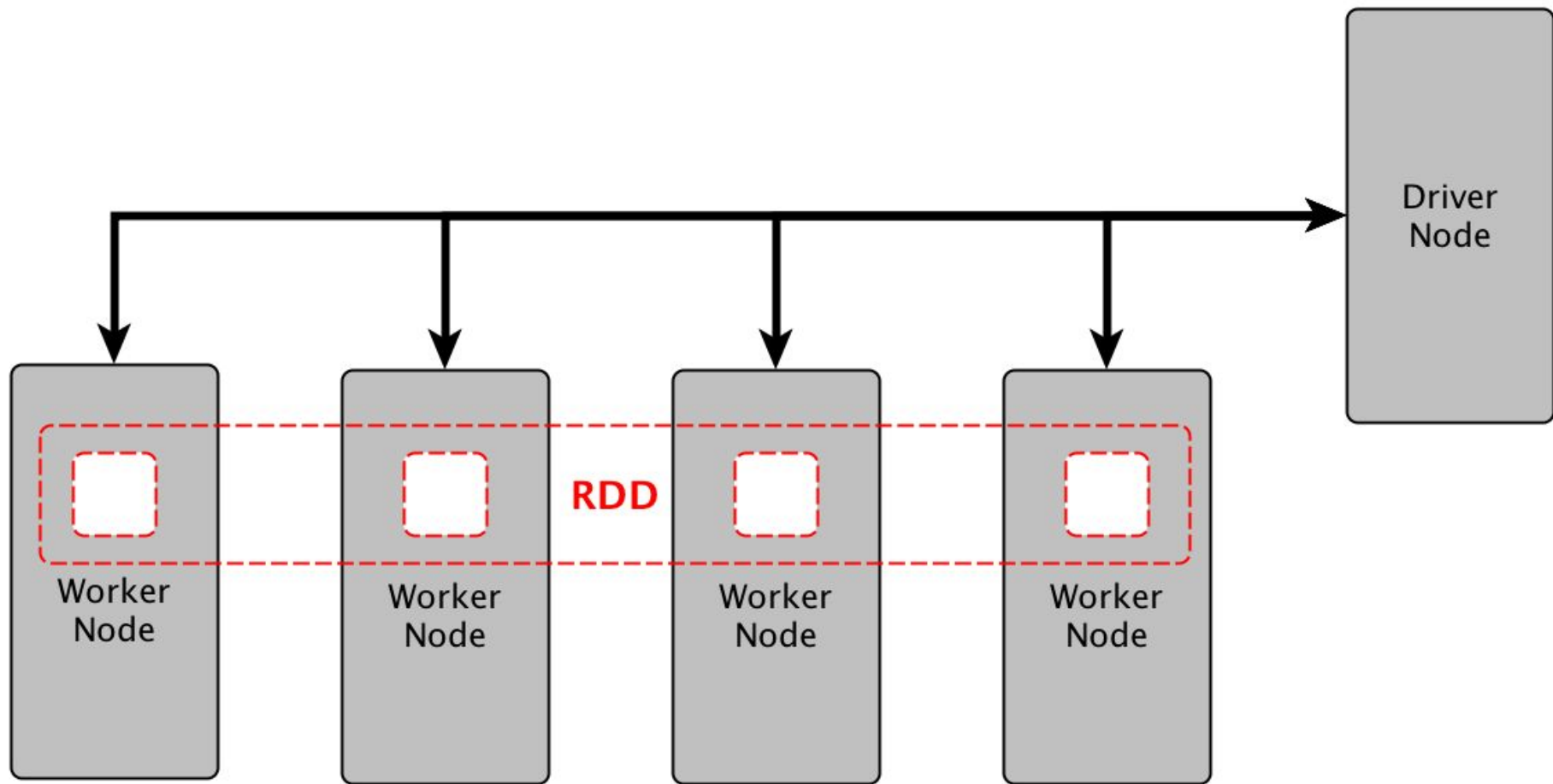
Spark Streaming permite o processamento de dados em stream à partir de diversas fontes. É possível processar os dados com funções de alto nível como: map, reduce, join e window.

Por sua vez, os dados podem ser armazenados em sistemas de arquivo, bancos de dado e tabelas ao vivo.

A decorative graphic on the left side of the slide. It consists of a blue parallelogram and a light green parallelogram, both tilted at an angle. The blue shape is in the foreground, and the green shape is partially behind it. They are set against a dark blue background with diagonal stripes.

Arquitetura Spark







Casos de Uso

NETFLIX

YAHOO!

ebay



Tentativas



Diferentes API's

1º



2º





Aplicação

Tecnologias



Google Cloud



Google
Cloud Pub/Sub



Google Cloud
Functions





Tecnologias

1 - Dataproc: O Dataproc é um serviço totalmente gerenciado para executar o Apache Spark.

2 - Pub/Sub: Fornece uma vasta análise de streaming e pipelines de integração de dados para gerir e distribuir dados.

3 - Datastore: O Datastore é um banco de dados NoSQL.

4 - Cloud Functions: É uma solução de computação sem servidor do Google para criar aplicativos com base em eventos.

Overview





Visão geral do código

O código a seguir configura o stream do Pub/Sub para processar os dados de entrada.

```
33 // [START stream_setup]
34 val sparkConf = new SparkConf().setAppName("TrendingHashtags")
35 val ssc = new StreamingContext(sparkConf, Seconds(slidingInterval.toInt))
36
37 // Set the checkpoint directory
38 val yarnTags = sparkConf.get("spark.yarn.tags")
39 val jobId = yarnTags.split(",").filter(_.startsWith("dataproc_job")).head
40 ssc.checkpoint(checkpointDirectory + '/' + jobId)
41
42 // Create stream
43 val messagesStream: DStream[String] = PubsubUtils
44   .createStream(
45     ssc,
46     projectID,
47     None,
48     "tweets-subscription", // Cloud Pub/Sub subscription for incoming tweets
49     SparkGCPCredentials.builder.build(), StorageLevel.MEMORY_AND_DISK_SER_2)
50   .map(message => new String(message.getData(), StandardCharsets.UTF_8))
```



Visão geral do código

O app do Spark extrai e conta todas as hashtags usando este canal simples.

```
24 // [START extract]
25 private[demo] def extractTrendingTags(input: RDD[String]): RDD[Popularity] =
26   input.flatMap(_.split("\\s+")) // Split on any white character
27   .filter(_.startsWith("#")) // Keep only the hashtags
28   // Remove punctuation, force to lowercase
29   .map(_.replaceAll("[.,!?:;]", "").toLowerCase)
30   // Remove the first #
31   .map(_.replaceFirst("^#", ""))
32   .filter(!_.isEmpty) // Remove any non-words
33   .map( (_, 1)) // Create word count pairs
34   .reduceByKey(_ + _) // Count occurrences
35   .map(r => Popularity(r._1, r._2))
36   // Sort hashtags by descending number of occurrences
37   .sortBy(r => (-r.amount, r.tag), ascending = true)
```



Visão geral do código

O app do Spark salva as 10 principais hashtags em uma nova linha do banco de dados no Datastore.

```
29 // [START convert_identity]
30 private[demo] def convertToEntity(hashtags: Array[Popularity],
31                                   keyFactory: String => KeyFactory): FullEntity[IncompleteKey] = {
32   val hashtagKeyFactory: KeyFactory = keyFactory("Hashtag")
33
34   val listValue = hashtags.foldLeft[ListValue.Builder](ListValue.newBuilder()){
35     (listValue, hashTag) => listValue.addValue(convertToDatastore(hashtagKeyFactory, hashTag))
36   }
37
38   val rowKeyFactory: KeyFactory = keyFactory("TrendingHashtags")
39
40   FullEntity.newBuilder(rowKeyFactory.newKey())
41     .set("datetime", Timestamp.now())
42     .set("hashtags", listValue.build())
43     .build()
44 }
```



Implantação



Passos

- 1 - Ativação dos serviços (Pub/Sub, Dataproc e Cloud functions) e do banco de dados (Datastore).
- 2 - Clonar projeto.
- 3 - Criar tópico e assinatura do Pub/Sub.
- 4 - Criar conta no dataproc e fornecer acesso aos serviços.
- 5 - Criar o cluster.
- 6 - Fazer o download e ou atualizar as dependências.
- 7 - Criar e iniciar o Job.
- 8 - Criar ambiente em python, instalar dependências e ativar o tweet-generator.
- 9 - Visualizar página web com os top trendings.



Configurações do Cluster

```
gcloud dataproc clusters create demo-cluster \  
  --region=us-central1 \  
  --zone=us-central1-a \  
  --scopes=pubsub,datastore \  
  --image-version=1.2 \  
  --service-account="$SERVICE_ACCOUNT_NAME@$PROJECT.iam.gserviceaccount.com"
```

Configurações do Cluster

← Detalhes do cluster		➕ ENVIAR JOB	↻ ATUALIZAR	▶ INICIAR	■ INTERROMPER	🗑 EXCLUIR	☰ VER REGISTROS
Região	us-central1						
Zona	us-central1-a						
Escalaonamento automático	Desativado						
Metastore do Dataproc	Nenhum						
Exclusão programada	Desativado						
Nó mestre	Padrão (1 mestre, N workers)						
Tipo de máquina	n1-standard-4						
Número de GPUs	0						
Tipo de disco principal	pd-standard						
Tamanho do disco principal	500 GB						
SSDs locais	0						
Nós de trabalho	2						
Tipo de máquina	n1-standard-4						
Número de GPUs	0						
Tipo de disco principal	pd-standard						
Tamanho do disco principal	500 GB						
SSDs locais	0						
Nós de trabalho secundários	0						
Inicialização segura	Desativada						
VTPM	Desativada						
Monitoramento de integridade	Desativada						
Bucket de preparação do Cloud Storage	dataproc-staging-us-central1-241164993869-3kdljyd0						
Rede	default						
Tags de rede	Nenhum						
Apenas IP interno	Não						
Versão da imagem [?]	1.2.102-debian9						
Criado em	18 de out. de 2021 20:15:14						
Propriedades	Mostrar propriedades						
Segurança avançada	Desativado						
Marcadores	goog-dataproc...demo-clust...						▼
Tipo de criptografia	Chave gerenciada pelo Google						

Configurações do Job

[←](#) Detalhes do job

[CLONAR](#) [EXCLUIR](#) [INTERROMPER](#) [ATUALIZAR](#)

ID do job	d0191edbd49042f48bb42a040d94e246
UUID do job	35899217-a438-3fe4-86eb-4ab904124894
Tipo	Job do Dataproc
Status	Em execução

MONITORAMENTO

CONFIGURAÇÃO

[EDITAR](#)

Horário de início:	18 de out. de 2021 20:17:54
Tempo decorrido:	9 min 16 s
Status:	Em execução
Região	us-central1
Cluster	demo-cluster
Tipo de job	Spark
Classe principal ou jar	gs://dataproc-staging-us-central1-241164993869-3kdlyjd0/google-cloud-dataproc-metainfo/fedd608e-e78f-4f86-85b4-25cb214afc00/jobs/d0191edbd49042f48bb42a040d94e246/staging/spark-streaming-pubsub-demo-1.0-SNAPSHOT.jar
Máximo de reinicializações por hora	10
Propriedades	
spark.dynamicAllocation.enabled	false
spark.streaming.receiver.writeAheadLog.enabled	true
Argumentos	sparkstream-329422 60 20 60 hdfs:///user/spark/checkpoint

Output do Job

Saída

LINHA DE ENCAPSULAMENTO: DESATIVADA 

region, 6

Window ending 2021-10-18T23:27:40.807000000Z for the past 60 seconds

Trending hashtags in this window:

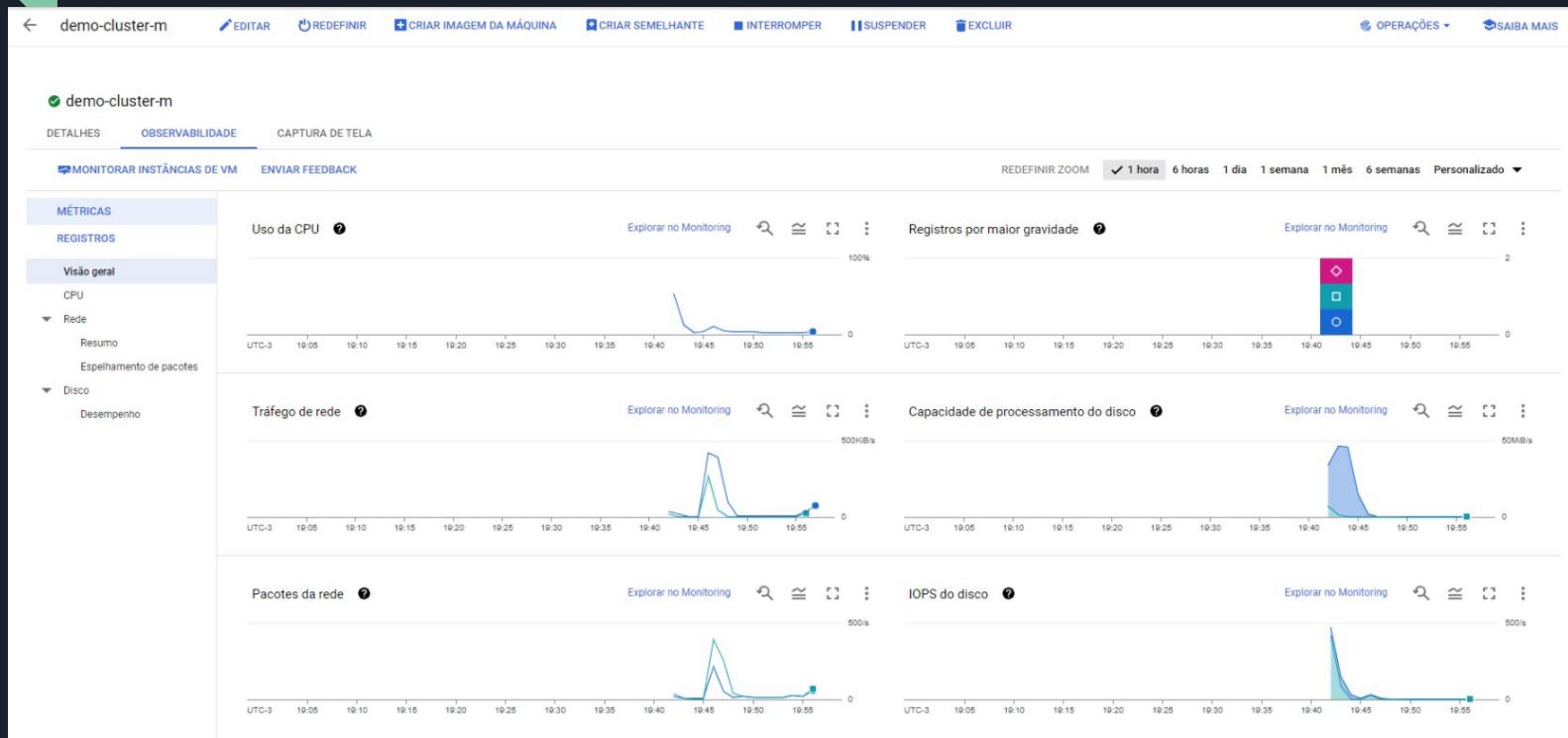
citizen, 9
difference, 7
discuss, 7
industry, 7
medical, 7
religious, 7
structure, 7
system, 7
anything, 6
country, 6

A saída pendente é de streaming

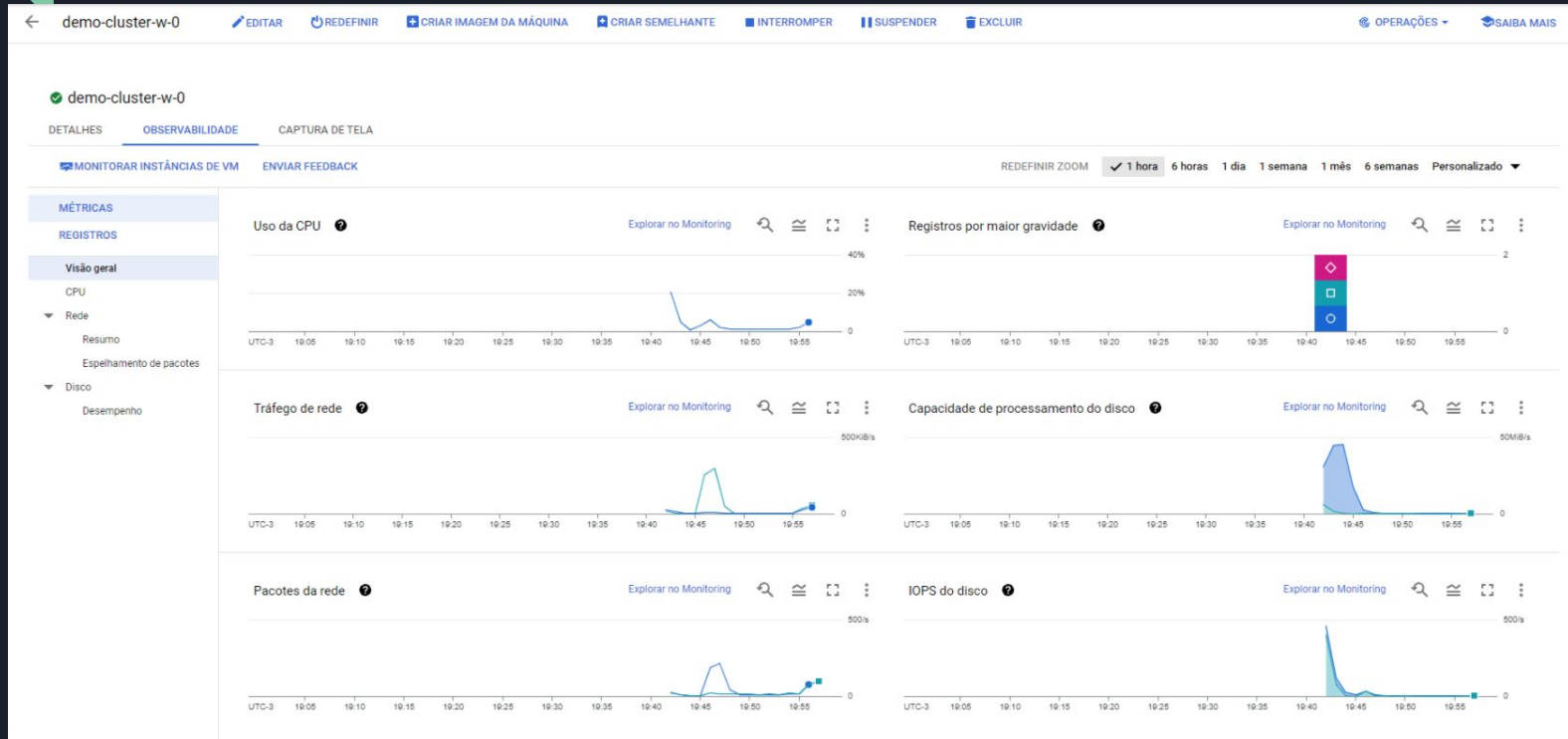


Funcionamento

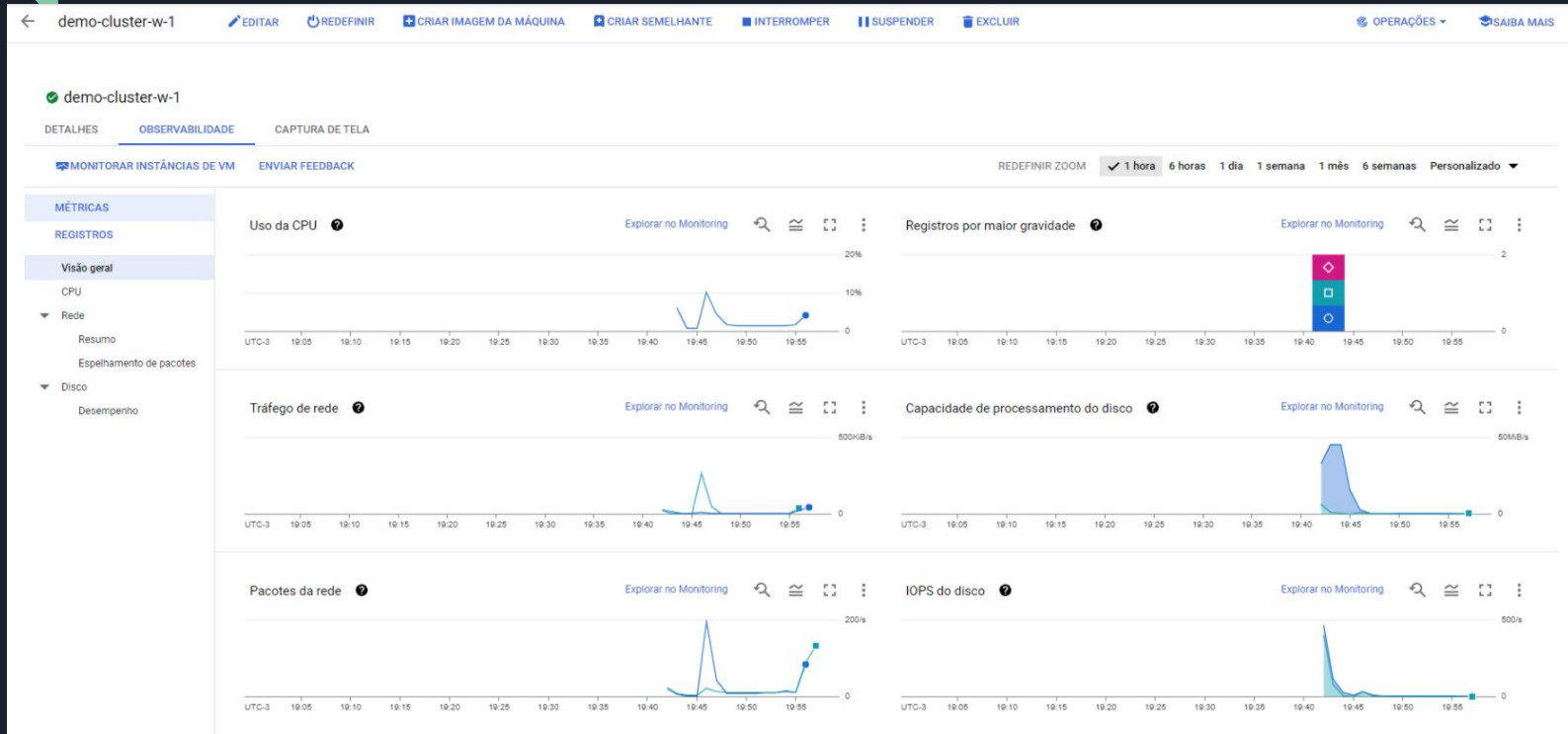
Processamento mestre



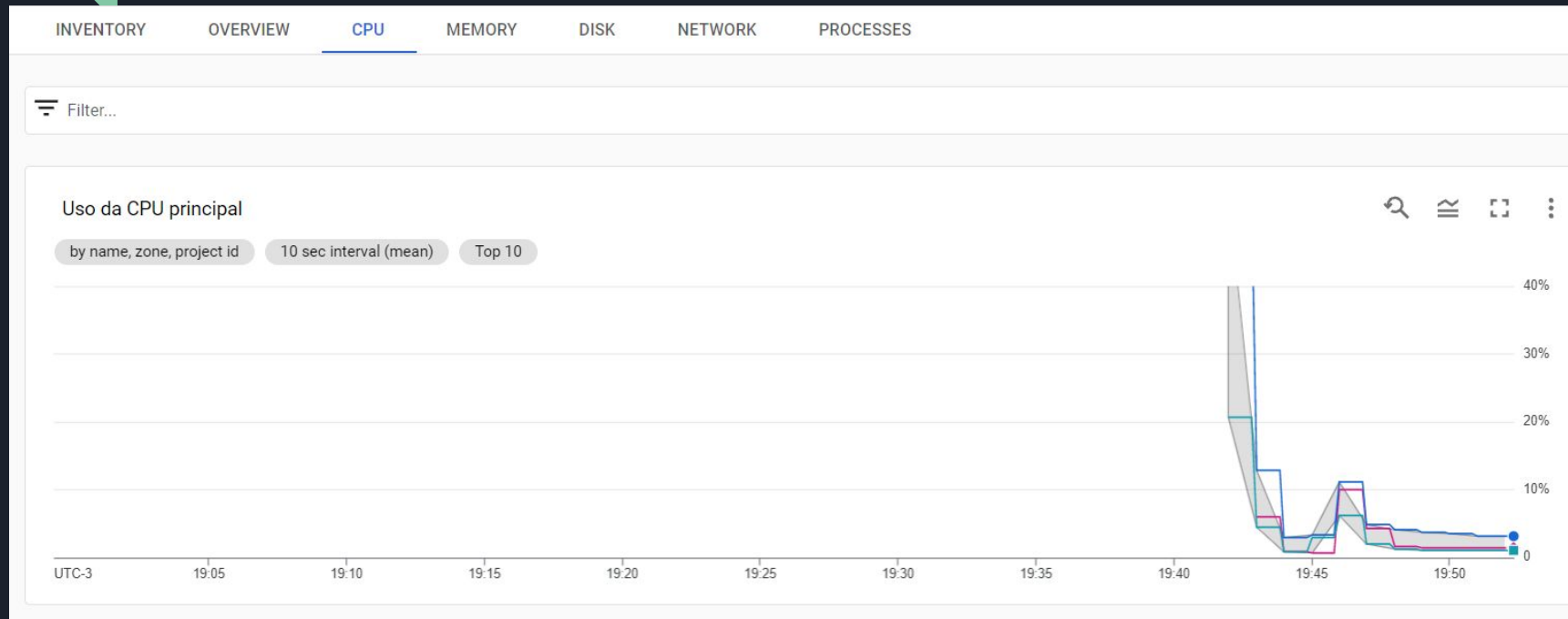
Processamento worker 0



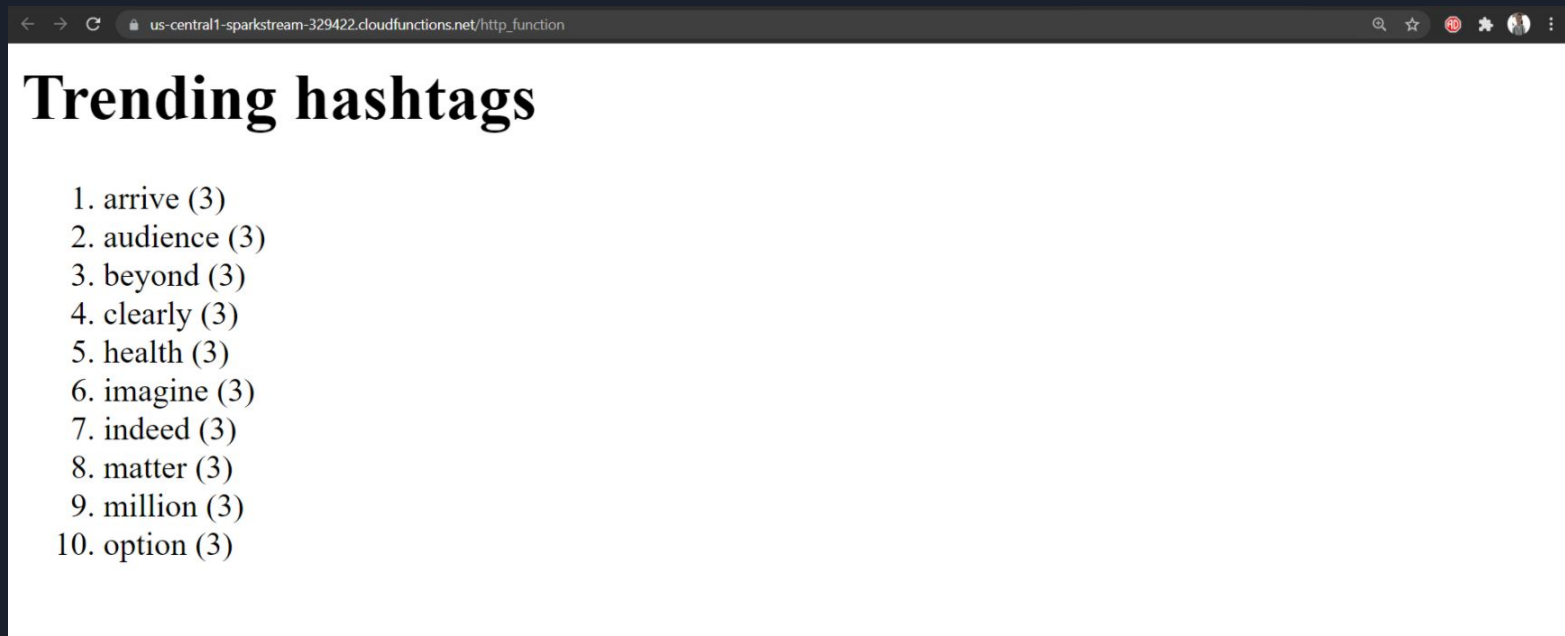
Processamento worker 1



Processamento geral



Output web





Muito obrigado!