

Agregador de Vendas em Tempo Real - Black Friday

Disciplina: Processamento Massivo de Dados

Autores: Ariel Guiliane (2025200218) e João Pedro Guedes (2025200237)

Data: Dezembro de 2025

Visão Geral

Sistema de processamento de vendas em tempo real utilizando tecnologias Big Data, simulando um cenário de Black Friday com vendas geradas continuamente, processadas em streaming e agregadas por categoria.

Tecnologias Utilizadas: - Apache Kafka (Message Broker) - Apache Spark Streaming (Processamento em tempo real) - MongoDB (Armazenamento de dados agregados) - Docker (Orquestração de containers) - Python (Geração de dados)

Arquitetura

Producer (Python) → Apache Kafka → Spark Streaming → MongoDB

2 vendas/s Faker + JSON	Topic: black- friday-sales	Janela: 1min Watermark: 2min	Agregações por categoria
----------------------------	-------------------------------	---------------------------------	-----------------------------

Fluxo de Dados:

1. Producer gera vendas fake (Faker) e envia JSON para Kafka
 2. Kafka distribui mensagens em 3 partições
 3. Spark processa em janelas de 1 minuto, agregando por categoria
 4. MongoDB armazena: {categoria, faturamento_total, volume_vendas}
-

Pré-requisitos

- Docker e Docker Compose
 - Python 3.8+
 - 8GB RAM recomendado
-

Passo a Passo de Execução

Etapa 1: Subir Ambiente Docker

```
cd bigdata_docker
docker-compose up -d
```

Verificar containers ativos:

```
docker ps | grep -E "kafka|mongo|jupyter-spark|namenode|zookeeper"
```

Etapa 2: Configurar Ambiente Python

```
python3 -m venv venv
source venv/bin/activate
pip install kafka-python-ng Faker python-dateutil
```

Etapa 3: Configurar Conectividade Kafka

Adicionar entrada no /etc/hosts:

```
echo "127.0.0.1 kafka" | sudo tee -a /etc/hosts
```

Etapa 4: Criar Tópico Kafka

```
docker exec -it kafka kafka-topics.sh \
  --create \
  --topic black-friday-sales \
  --bootstrap-server localhost:9092 \
  --partitions 3 \
  --replication-factor 1
```

Etapa 5: Executar Producer (Terminal 1)

```
source venv/bin/activate
python producer.py
```

Etapa 6: Executar Spark Streaming (Terminal 2)

```
./run_spark_streaming.sh
```

Etapa 7: Consultar Resultados no MongoDB

Via terminal:

```
docker exec -it mongo mongo -u root -p root --authenticationDatabase admin black_friday \
  --eval "db.faturamento_tempo_real.find().pretty()"
```

Ou via interface web: <http://localhost:8081>

Componentes do Sistema

Componente	Função	Porta
Kafka	Message broker para ingestão	9092
Zookeeper	Coordenação do Kafka	2181
Spark	Processamento streaming	4040
MongoDB	Armazenamento agregado	27017
Mongo Express	Interface web MongoDB	8081

Conceitos de Big Data Aplicados

1. **Streaming Processing:** Processamento contínuo via micro-batches
 2. **Windowing:** Agregação por janelas temporais de 1 minuto
 3. **Watermarking:** Tolerância de 2 minutos para dados atrasados
 4. **Particionamento:** 3 partições Kafka para paralelismo
 5. **Checkpoint:** Garantia de exactly-once semantics
-

Estrutura de Dados

Entrada (Producer → Kafka):

```
{  
  "id_pedido": "uuid",  
  "categoria": "Eletronicos|Livros|Casa|Gamer",  
  "valor": 1299.99,  
  "timestamp": "2024-11-25T10:30:00"  
}
```

Saída (Spark → MongoDB):

```
{  
  "janela_inicio": "2024-11-25T10:30:00",  
  "janela_fim": "2024-11-25T10:31:00",  
  "categoria": "Eletronicos",  
  "faturamento_total": 15234.50,  
  "volume_vendas": 12  
}
```

Resultados Esperados

Após executar o sistema por alguns minutos:

1. Producer envia ~2 vendas por segundo
 2. Spark agrega vendas em janelas de 1 minuto por categoria
 3. MongoDB armazena faturamento total e volume de vendas por categoria/janela
 4. Consultas permitem análise em tempo real do desempenho de vendas
-

Prints de Execução

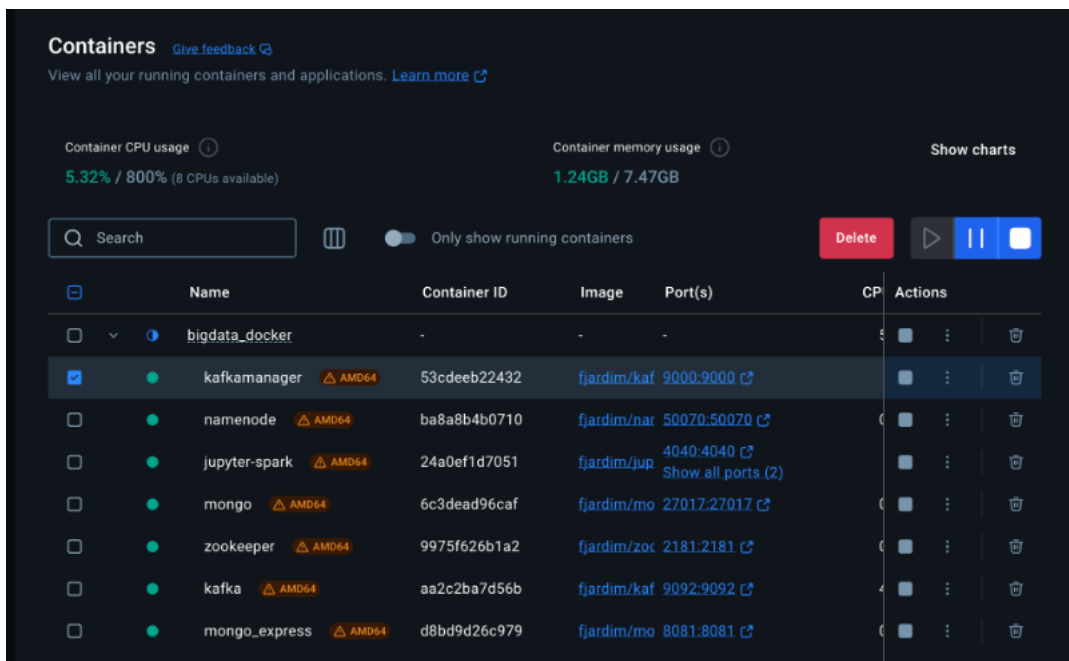


Figura 1: 1. Containers Docker em execução

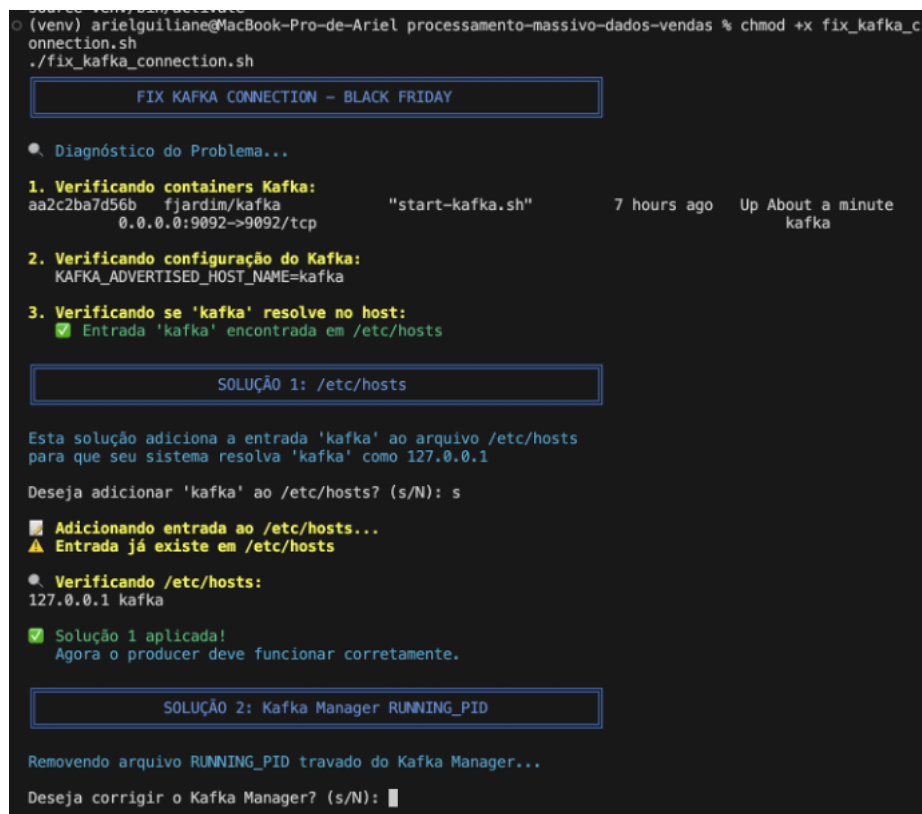


Figura 2: 2. Teste de Conexão com Kafka - Parte 1

```
SOLUÇÃO 2: Kafka Manager RUNNING_PID

Removendo arquivo RUNNING_PID travado do Kafka Manager...

Deseja corrigir o Kafka Manager? (s/N): s

❌ Removendo RUNNING_PID...
Error response from daemon: container 53cdeeb22432d972276d439a676d103309c499944b279c5e31c92182e986a
53a is not running
🔄 Reiniciando Kafka Manager...
kafkamanager
✅ Kafka Manager reiniciado!
Aguarde ~30 segundos e acesse: http://localhost:9000

TESTE DE CONECTIVIDADE

Deseja testar a conectividade com Kafka? (s/N): s

🔧 Testando conexão com Kafka...

Teste 1: Listar tópicos
__consumer_offsets
black-friday-sales

Teste 2: Descrever tópico black-friday-sales
Topic:black-friday-sales PartitionCount:3 ReplicationFactor:1 Configs:segment.by
es=1073741824
Topic: black-friday-sales Partition: 0 Leader: 1002 Replicas: 1002 Isr: 1002
Topic: black-friday-sales Partition: 1 Leader: 1002 Replicas: 1002 Isr: 1002
Topic: black-friday-sales Partition: 2 Leader: 1002 Replicas: 1002 Isr: 1002

✅ Testes concluídos!

CORREÇÃO CONCLUÍDA! 🏁
```

Figura 3: 2. Teste de Conexão com Kafka - Parte 2


```
PRODUCER BLACK FRIDAY - INICIANDO...

Kafka Servers: ['localhost:9092']
Tópico: black-friday-sales
Categorias: ['Eletronicos', 'Livros', 'Casa', 'Gamer']
Faixa de Valores: R$ 10.00 - R$ 5000.00
Intervalo: 0.5s

✓ Produtor Kafka conectado com sucesso!
✚ Iniciando geração de vendas... (Ctrl+C para parar)

✓ Venda enviada | Tópico: black-friday-sales | Partição: 1 | Offset: 482 | Categoria: Eletronicos
| Valor: R$ 3595.13
✓ Venda enviada | Tópico: black-friday-sales | Partição: 2 | Offset: 464 | Categoria: Gamer | Valor: R$ 4117.88
✓ Venda enviada | Tópico: black-friday-sales | Partição: 2 | Offset: 465 | Categoria: Eletronicos
| Valor: R$ 857.99
✓ Venda enviada | Tópico: black-friday-sales | Partição: 0 | Offset: 473 | Categoria: Casa | Valor: R$ 2778.28
✓ Venda enviada | Tópico: black-friday-sales | Partição: 2 | Offset: 466 | Categoria: Livros | Valor: R$ 425.63
✓ Venda enviada | Tópico: black-friday-sales | Partição: 2 | Offset: 467 | Categoria: Eletronicos
| Valor: R$ 531.71
✓ Venda enviada | Tópico: black-friday-sales | Partição: 0 | Offset: 474 | Categoria: Gamer | Valor: R$ 4018.60
✓ Venda enviada | Tópico: black-friday-sales | Partição: 0 | Offset: 475 | Categoria: Eletronicos
| Valor: R$ 3727.97
✓ Venda enviada | Tópico: black-friday-sales | Partição: 1 | Offset: 483 | Categoria: Casa | Valor: R$ 623.91
✓ Venda enviada | Tópico: black-friday-sales | Partição: 2 | Offset: 468 | Categoria: Gamer | Valor: R$ 1660.90
✓ Venda enviada | Tópico: black-friday-sales | Partição: 2 | Offset: 469 | Categoria: Eletronicos
| Valor: R$ 2464.77
✓ Venda enviada | Tópico: black-friday-sales | Partição: 0 | Offset: 476 | Categoria: Livros | Valor: R$ 4717.00
✓ Venda enviada | Tópico: black-friday-sales | Partição: 1 | Offset: 484 | Categoria: Casa | Valor: R$ 917.91
✓ Venda enviada | Tópico: black-friday-sales | Partição: 2 | Offset: 470 | Categoria: Gamer | Valor: R$ 4492.28
✓ Venda enviada | Tópico: black-friday-sales | Partição: 1 | Offset: 485 | Categoria: Casa | Valor: R$ 1895.79
✓ Venda enviada | Tópico: black-friday-sales | Partição: 0 | Offset: 477 | Categoria: Gamer | Valor: R$ 1176.59
✓ Venda enviada | Tópico: black-friday-sales | Partição: 1 | Offset: 486 | Categoria: Eletronicos
| Valor: R$ 490.99
✓ Venda enviada | Tópico: black-friday-sales | Partição: 1 | Offset: 487 | Categoria: Casa | Valor: R$ 4106.65
```

Figura 5: 3. Producer enviando vendas para Kafka

SUBMISSÃO SPARK STREAMING - BLACK FRIDAY

Configurações:

Container: jupyter-spark
Script: spark_processor.py
Spark Version: 2.4.1
Scala Version: 2.11

Copiando script para o container...

Successfully copied 8.7kB to jupyter-spark:/mnt/notebooks/spark_processor.py

✓ Arquivo copiado com sucesso!

Iniciando Spark Streaming...

Ivy Default Cache set to: /root/.ivy2/cache
The jars for the packages stored in: /root/.ivy2/jars
:: loading settings :: url = jar:file:/opt/spark-2.4.1-bin-without-hadoop/jars/ivy-2.4.0.jar!/org/apache/ivy/core/settings/ivysettings.xml
org.apache.spark#spark-sql-kafka-0-10_2.11 added as a dependency
org.mongodb.spark#mongo-spark-connector_2.11 added as a dependency
:: resolving dependencies :: org.apache.spark#spark-submit-parent-35dfd26b-5383-4276-bae7-743dd0eb2d89;1.0
 confs: [default]
 found org.apache.spark#spark-sql-kafka-0-10_2.11;2.4.1 in central
 found org.apache.kafka#kafka-clients;2.0.0 in central
 found org.lz4#lz4-java;1.4.0 in central
 found org.xerial.snappy#snappy-java;1.1.7.1 in central
 found org.slf4j#slf4j-api;1.7.16 in central
 found org.spark-project.spark#unused;1.0.0 in central
 found org.mongodb.spark#mongo-spark-connector_2.11;2.4.0 in central
 found org.mongodb#mongo-java-driver;3.9.0 in central
:: resolution report :: resolve 715ms :: artifacts dl 29ms
 :: modules in use:
 org.apache.kafka#kafka-clients;2.0.0 from central in [default]
 org.apache.spark#spark-sql-kafka-0-10_2.11;2.4.1 from central in [default]
 org.lz4#lz4-java;1.4.0 from central in [default]
 org.mongodb#mongo-java-driver;3.9.0 from central in [default]
 org.mongodb.spark#mongo-spark-connector_2.11;2.4.0 from central in [default]
 org.slf4j#slf4j-api;1.7.16 from central in [default]
 org.spark-project.spark#unused;1.0.0 from central in [default]
 org.xerial.snappy#snappy-java;1.1.7.1 from central in [default]

conf	number	modules			artifacts	
		search	downloaded	evicted	number	downloaded
default	8	0	0	0	8	0

Figura 6: 4. Spark Streaming processando dados - Parte 1


```

:: retrieving :: org.apache.spark#spark-submit-parent-35dfd26b-5383-4276-bae7-743d
d0eb2d89
  confs: [default]
  0 artifacts copied, 8 already retrieved (0kB/18ms)
=====
SPARK STREAMING BLACK FRIDAY - INICIANDO...
=====

25/12/01 20:08:03 INFO spark.SparkContext: Running Spark version 2.4.1
25/12/01 20:08:03 WARN spark.SparkConf: Note that spark.local.dir will be overrid
en by the value set by the cluster manager (via SPARK_LOCAL_DIRS in mesos/standalo
ne/kubernetes and LOCAL_DIRS in YARN).
25/12/01 20:08:03 INFO spark.SparkContext: Submitted application: BlackFriday-Real
Time-Aggregator
25/12/01 20:08:03 INFO spark.SecurityManager: Changing view acls to: root
25/12/01 20:08:03 INFO spark.SecurityManager: Changing modify acls to: root
25/12/01 20:08:03 INFO spark.SecurityManager: Changing view acls groups to:
25/12/01 20:08:03 INFO spark.SecurityManager: Changing modify acls groups to:
25/12/01 20:08:03 INFO spark.SecurityManager: SecurityManager: authentication disa
bled; ui acls disabled; users with view permissions: Set(root); groups with view
permissions: Set(); users with modify permissions: Set(root); groups with modify
permissions: Set()
25/12/01 20:08:04 INFO util.Utils: Successfully started service 'sparkDriver' on p
ort 42135.
25/12/01 20:08:04 INFO spark.SparkEnv: Registering MapOutputTracker
25/12/01 20:08:04 INFO spark.SparkEnv: Registering BlockManagerMaster
25/12/01 20:08:04 INFO storage.BlockManagerMasterEndpoint: Using org.apache.spark.
storage.DefaultTopologyMapper for getting topology information
25/12/01 20:08:04 INFO storage.BlockManagerMasterEndpoint: BlockManagerMasterEndpo
int up
25/12/01 20:08:04 INFO storage.DiskBlockManager: Created local directory at /tmp/b
lockmgr-873c3d61-81bc-431a-acc7-dcf6eb856b2a
25/12/01 20:08:04 INFO memory.MemoryStore: MemoryStore started with capacity 366.3
MB
25/12/01 20:08:04 INFO spark.SparkEnv: Registering OutputCommitCoordinator
25/12/01 20:08:04 INFO util.log: Logging initialized @4995ms
25/12/01 20:08:04 INFO server.Server: jetty-9.3.z-SNAPSHOT, build timestamp: unkno
wn, git hash: unknown
25/12/01 20:08:04 INFO server.Server: Started @5110ms
25/12/01 20:08:04 INFO server.AbstractConnector: Started ServerConnector@7ad968ca{
HTTP/1.1,[http/1.1]}{0.0.0.0:4040}
25/12/01 20:08:04 INFO util.Utils: Successfully started service 'SparkUI' on port
4040.
25/12/01 20:08:04 INFO handler.ContextHandler: Started o.s.j.s.ServletContextHandl
er@5c1096d6{/jobs,null,AVAILABLE,@Spark}

```

Figura 7: 4. Spark Streaming processando dados - Parte 2

```

er@bc01795a1/static/sql,null,AVAILABLE,@spark/
25/12/01 20:08:06 INFO state.StateStoreCoordinatorRef: Registered StateStoreCoordinator endpoint
>>> SparkSession criada com sucesso!
>>> Conectando ao Kafka: kafka:9092
>>> Topico: black-friday-sales

>>> Configurando escrita no MongoDB: mongoddb://root:root@mongo:27017
>>> Database: black_friday
>>> Collection: faturamento_tempo_real

>>> Pipeline de streaming iniciado com sucesso!
>>> Checkpoint: file:///tmp/spark-checkpoint-black-friday
>>> Janela de agregacao: 1 minute

>>> Aguardando dados... (Ctrl+C para parar)

25/12/01 20:08:16 WARN state.HDFSBackedStateStoreProvider: The state for version 7
doesn't exist in loadedMaps. Reading snapshot file and delta files if needed...Note that this is normal for the first batch of starting query.
25/12/01 20:08:16 WARN state.HDFSBackedStateStoreProvider: The state for version 7
doesn't exist in loadedMaps. Reading snapshot file and delta files if needed...Note that this is normal for the first batch of starting query.
25/12/01 20:08:16 WARN state.HDFSBackedStateStoreProvider: The state for version 7
doesn't exist in loadedMaps. Reading snapshot file and delta files if needed...Note that this is normal for the first batch of starting query.
25/12/01 20:08:16 WARN state.HDFSBackedStateStoreProvider: The state for version 7
doesn't exist in loadedMaps. Reading snapshot file and delta files if needed...Note that this is normal for the first batch of starting query.

```

Figura 8: 4. Spark Streaming processando dados - Parte 3

```

c -it mongo mongo -u root -p root --authenticationDatabase admin black_friday \
--eval "db.faturamento_tempo_real.find().pretty()"
MongoDB shell version v4.2.3
connecting to: mongod://127.0.0.1:27017/black_friday?authSource=admin&compressors
=disabled&gssapiServiceName=mongodb
Implicit session: session { "id" : UUID("8412a21c-01dc-4f66-bcb1-6a32acc4569b") }
MongoDB server version: 4.2.3
{
  "_id" : ObjectId("692d946330c49e018dd00c95"),
  "janela_inicio" : ISODate("2025-12-01T10:10:00Z"),
  "janela_fim" : ISODate("2025-12-01T10:11:00Z"),
  "categoria" : "Eletronicos",
  "faturamento_total" : 12160.34,
  "volume_vendas" : NumberLong(4)
}
{
  "_id" : ObjectId("692d946330c49e018dd00c96"),
  "janela_inicio" : ISODate("2025-12-01T10:10:00Z"),
  "janela_fim" : ISODate("2025-12-01T10:11:00Z"),
  "categoria" : "Livros",
  "faturamento_total" : 9423.85,
  "volume_vendas" : NumberLong(4)
}
{
  "_id" : ObjectId("692d946330c49e018dd00c97"),
  "janela_inicio" : ISODate("2025-12-01T10:10:00Z"),
  "janela_fim" : ISODate("2025-12-01T10:11:00Z"),
  "categoria" : "Gamer",
  "faturamento_total" : 13223.36,
  "volume_vendas" : NumberLong(5)
}
{
  "_id" : ObjectId("692d946330c49e018dd00c98"),
  "janela_inicio" : ISODate("2025-12-01T10:09:00Z"),
  "janela_fim" : ISODate("2025-12-01T10:10:00Z"),
  "categoria" : "Casa",
  "faturamento_total" : 1018.3,
  "volume_vendas" : NumberLong(1)
}
{
  "_id" : ObjectId("692d946330c49e018dd00c94"),
  "janela_inicio" : ISODate("2025-12-01T10:10:00Z"),
  "janela_fim" : ISODate("2025-12-01T10:11:00Z"),
  "categoria" : "Casa",
  "faturamento_total" : 6418.5599999999995,
  "volume_vendas" : NumberLong(2)
}
{
  "_id" : ObjectId("692d946a30c49e018dd00ca4"),
  "janela_inicio" : ISODate("2025-12-01T10:10:00Z"),
  "janela_fim" : ISODate("2025-12-01T10:11:00Z"),
  "categoria" : "Casa",
  "faturamento_total" : 80089.67000000001,
  "volume_vendas" : NumberLong(30)
}
}

```

Figura 9: 5. Resultados no MongoDB (Terminal)

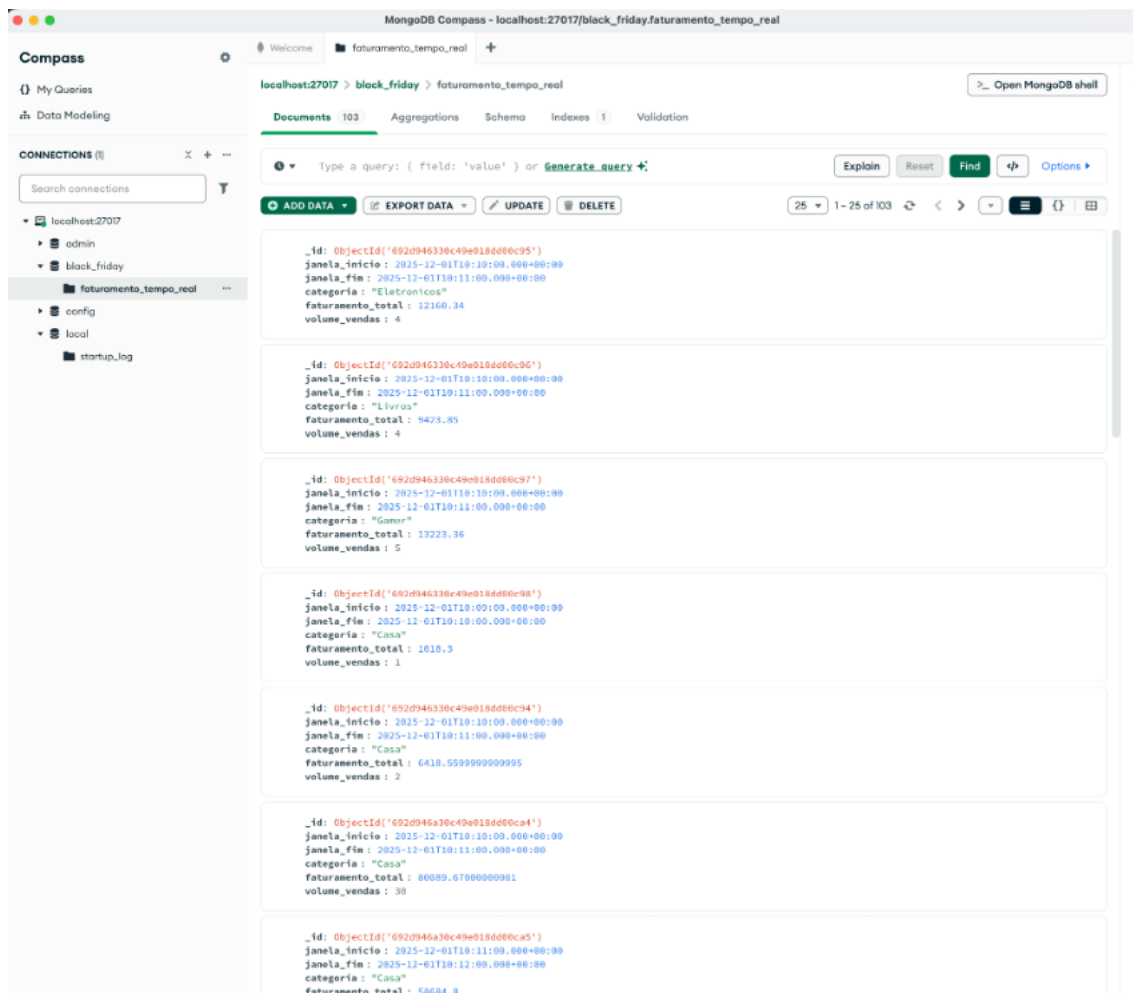


Figura 10: 6. Mongo Express - Interface Web - Parte 1

localhost:27017 > black_friday > faturamento_tempo_real

Documents 103 Aggregations Schema Indexes 1 Validation

Type a query: { field: 'value' } or [Generate query](#)

+ ADD DATA EXPORT DATA UPDATE DELETE

```
_id: ObjectId('692d946330c49e018dd00c95')
janela_inicio: 2025-12-01T10:10:00.000+00:00
janela_fim: 2025-12-01T10:11:00.000+00:00
categoria: "Eletronicos"
faturamento_total: 12160.34
volume_vendas: 4
```

```
_id: ObjectId('692d946330c49e018dd00c96')
janela_inicio: 2025-12-01T10:10:00.000+00:00
janela_fim: 2025-12-01T10:11:00.000+00:00
categoria: "Livros"
faturamento_total: 9423.85
volume_vendas: 4
```

```
_id: ObjectId('692d946330c49e018dd00c97')
janela_inicio: 2025-12-01T10:10:00.000+00:00
janela_fim: 2025-12-01T10:11:00.000+00:00
categoria: "Gamer"
faturamento_total: 13223.36
volume_vendas: 5
```

Figura 11: 6. Mongo Express - Interface Web - Parte 2


	<pre>_id: ObjectId('692d946a30c49e018dd00cac') janela_inicio : 2025-12-01T10:13:00.000+00:00 janela_fim : 2025-12-01T10:14:00.000+00:00 categoria : "Livros" faturamento_total : 8106.7 volume_vendas : 3</pre>
	<pre>_id: ObjectId('692d946f30c49e018dd00cb0') janela_inicio : 2025-12-01T10:13:00.000+00:00 janela_fim : 2025-12-01T10:14:00.000+00:00 categoria : "Eletronicos" faturamento_total : 20149.77 volume_vendas : 9</pre>
	<pre>_id: ObjectId('692d946f30c49e018dd00cad') janela_inicio : 2025-12-01T10:13:00.000+00:00 janela_fim : 2025-12-01T10:14:00.000+00:00 categoria : "Casa" faturamento_total : 30743.6 volume_vendas : 14</pre>
	<pre>_id: ObjectId('692d946f30c49e018dd00cae') janela_inicio : 2025-12-01T10:13:00.000+00:00 janela_fim : 2025-12-01T10:14:00.000+00:00 categoria : "Livros" faturamento_total : 11646.58 volume_vendas : 5</pre>
	<pre>_id: ObjectId('692d946f30c49e018dd00caf') janela_inicio : 2025-12-01T10:13:00.000+00:00 janela_fim : 2025-12-01T10:14:00.000+00:00 categoria : "Gamer" faturamento_total : 3766.64 volume_vendas : 2</pre>

Figura 12: 6. Mongo Express - Interface Web - Parte 3

localhost:27017 > local > startup_log

Documents

4

Aggregations

Schema

Indexes

1

Validation



Type a query: { field: 'value' } or [Generate query](#)



ADD DATA



EXPORT DATA



UPDATE



DELETE

```
_id: "mongo-1764594132547"
hostname: "mongo"
startTime: 2025-12-01T13:02:12.000+00:00
startTimeLocal: "Mon Dec 1 13:02:12.549"
▶ cmdLine: Object
pid: 28
▶ buildinfo: Object
```

```
_id: "mongo-1764594138132"
hostname: "mongo"
startTime: 2025-12-01T13:02:18.000+00:00
startTimeLocal: "Mon Dec 1 13:02:18.132"
▶ cmdLine: Object
pid: 1
▶ buildinfo: Object
```

```
_id: "mongo-1764594248231"
hostname: "mongo"
startTime: 2025-12-01T13:04:08.000+00:00
startTimeLocal: "Mon Dec 1 13:04:08.231"
▶ cmdLine: Object
pid: 1
▶ buildinfo: Object
```

```
_id: "mongo-1764618117989"
hostname: "mongo"
startTime: 2025-12-01T19:41:57.000+00:00
startTimeLocal: "Mon Dec 1 19:41:57.989"
▶ cmdLine: Object
pid: 1
▶ buildinfo: Object
```

Figura 13: 6. Mongo Express - Interface Web - Parte 4

Repositório Base Docker: https://github.com/fabiogjardim/bigdata_docker

Repositório do Projeto: <https://github.com/UFG-Pos/processamento-massivo-dados-vendas>