**Ame Digital - Teste Data Engineer**

O desafio a seguir tem como objetivo avaliar seus conhecimentos e experiências com dados e habilidade de resolver problemas. Ao solucioná-lo, você nos mostrará:

Sua capacidade de extrair dados de uma fonte, processá-los e transformá-los em informações.

* Seu entendimento sobre tecnologias de Big Data.
* Seu conhecimento em SQL.

**Desafio**

A seguir você encontrará links para dois conjuntos de dados que contém requisições HTTP para os servidores da NASA - Kennedy Space Center para os períodos de Julho e Agosto de 1995. Kennedy Space Center - Julho: Nasa Kennedy Server - July Kennedy Space Center - Agosto : Nasa Kennedy Sever - August Os logs contém as seguintes informações:

* Host que está realizando a requisição
* Timestamp do momento em que a requisição aconteceu
* Requisição
* Código de retorno da chamada HTTP
* Total de bytes retornados

**Pre-requisitos**

Foram utilizadas as seguintes ferramentas para desenvolver a solução técnica:

* [Anaconda3](https://www.anaconda.com/distribution/) - Main Environment
* [MicrosoftVsCode](https://code.visualstudio.com/) - Development IDE
* [JupyterNotebook](https://jupyter.org/) - Development IDE

Linguagens:

* [Python3.7](https://www.python.org/)
* [ApacheSpark](https://spark.apache.org/)
* [SparkSQL](https://spark.apache.org/sql/)

**Instalação e execução**

Para obter o resultado do teste técnico, basta execute as linhas do caderno

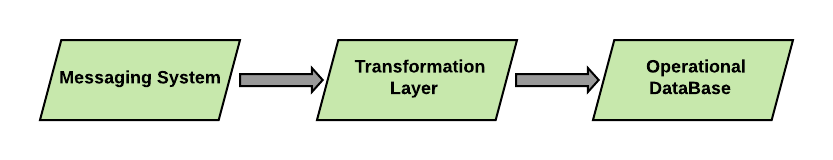
ame\_resolucao.ipynb

### BIG DATA - TEÓRICO - Respostas

1. **Considerando que a Ame possui diferentes aplicações, resultando em diferentes fontes de dados (como bancos relacionais e noSQL), de que maneira você construiria uma arquitetura para realizar a ingestão desses dados em uma plataforma de Big Data?**

**Descreva as tecnologias que você escolheria para realizar a ingestão, bem como o fluxo de dados entre elas (lembrando que o objetivo é disponibilizar as informações o mais próximo de "real-time" possível).**

Uma pipeline de dados normalmente recebe dados de diversas fontes externas, sendo essas, banco de dados de aplicações, Data Warehouses, arquivos manuais (xlsx, csv, txt, etc), bem como também dados em tempo real gerado por eventos (Dados de localização GPS, páginas web, sensores, celulares, por exemplo). Pensando em um pipeline focado em real-time para as soluções da Ame Digital, precisamos acomodar essa infraestrutura em três principais camadas: 1. camada in-memory storage para ingestão rápida; 2. Uma arquitetura com escalabilidade horizontal; 3. Que os dados sejam consultáveis e que permita exploração interativa, em tempo real. Pensando nesse contexto, teríamos uma pipeline na seguinte estrutura:



Baseado na estrutura da imagem acima, utilizaria **Apache** **Kafka** como o sistema de mensagens, que orquestraria o recebimento de dados entre os sistemas externos (que chamamos de produtores) e disponibilizaria os dados para os sistemas internos (os consumidores). Esse processo é escalável, uma vez que o Kafka possui características de sistema distribuído.

Para a fase de transformação de dados, Spark pode ser utilizado como um captador de dados do Kafka, para tratamento de um dataset, enriquecimento dos dados e/ou persistência dos dados captados em tempo real para uma base de dados

E para análise dos dados em real-time e histórico, os dados podem ser trabalhados além do ambiente de streaming e de transformação, podem então persisti-los em base de dados.

Desta maneira, desenhei um exemplo de pipeline que capta dados de diversas fontes (sejam essas estruturadas e/ou não estruturadas), utilizando de tecnologias que possam gerar tópicos de captação do dado, armazenamento distribuído, tratamento e enriquecimento do dado, persistência de dados in-memory e por fim, a disponibilização desse dado para cientistas e para analistas. Além disso, adicionei um layer que permite o controle dos processos do workflow, armazenando logs de execução e tratamento e disponibilizando dashboard analítico para auxiliar engenheiros na manutenção do ambiente.

A close up of a map

Description automatically generated

1. **Ao utilizar ferramentas de processamento distribuído como Spark ou Hive, é muito comum enfrentar problemas relacionados à má distribuição de dados entre as máquinas do cluster, diminuindo drasticamente a performance das aplicações, principalmente em operações relacionadas a agregação ou join. Utilizando seus conhecimentos e experiências, descreva uma possível solução para o problema em questão.**

Data skew é distribuição do processamento é assimétrico de dados em operações de joins e aggregation que acarreta um processamento mais lento, mais custo e com necessidade de maior recurso.

Como uma possível solução, podemos realizar o reprocessamento dos dados, quebrando uma estrutura em partições menores de processamento. Por exemplo, se temos uma tabela assimétrica A com muitos dados com campos nulos que precisam ser processados, podemos particionar essa tabela A em dois menores blocos (“a” com dados non-null e “b” com dados null) e realizar os joins ou aggregation separadamente.

1. **O dia a dia de um engenheiro de dados, dentre outras tarefas, é disponibilizar as informações em alta performance (próximos a real-time) para Analistas e Cientista de Dados de modo a possibilitar à análise e criação de modelos estatísticos. De que modo e quais tecnologias você usaria para disponibilizar os dados para estas pessoas.**

Dentro do exemplo dado no exercício 1, todo o dado recebido na pipeline ficará disponível em ambiente distribuído (como HDFS) e poderá ser consumindo por jupyter notebooks em um sandbox para exploração de dados por cientistas e analistas. Além disso, todo o dado histórico e analítico pode ser armazenado em um data warehouse (como o Hive DB ou quando falamos de soluções em nuvem como AWS Redshift ou um banco de dados Oracle Data Warehouse), para consumo de dashboards, como também para elaboração de Business reports. Além disso, é possível que realizar análises da própria pipeline dentro da estrutura de documentos do Elasticsearch, que permite também ser consumido por Dashboard no Kibana, em tempo real.

1. **Por fim, tendo em mente o crescimento exponencial dos dados e utilização massiva da plataforma de Big Data, quais métodos de organização e/ou governança você implementaria para manter o ambiente sustentável?**

Quando pensando em governança de dados em um ambiente Big data, considero que temos pilares que devem ser colocadas em pauta, como segurança e custo efetivo.

Quando tratamento de segurança, podemos considerar o controle de acessos de usuários dentro clusters como hadoop utilizando do Apache Ambari, pois viabiliza um controle total de todo o processo do ambiente distribuído, bem como controle das roles de usuário. Se falamos de ambiente Cloud (como, por exemplo, AWS), podemos considerar o controle de toda a pipeline de dados por meio do IAM, onde podemos adicionar policies para as roles, garantindo a segurança de acesso.

Sobre custo efetivo e identificar qual o real uso de uma infraestrutura de dados está sendo consumido baseado no seu custo financeiro. Desta maneira, penso que uma equipe Devops precisa estar a par de cada solução cloud (seja clusters ou serviços) ou física (servidores) para garantir que não exista infraestrutura sendo consumida indevidamente.

### BIG DATA - PRÁTICA - Respostas

Queremos que você nos ajude a responder as questões abaixo e, para isso, queremos que você trate e estruture os dados utilizando Spark e realize as consultas para responder às questões abaixo utilizando SparkSQL .

### 1. Número de HOSTs únicos.

#### Resposta:

137978

### 2. O total de erros 404 dentro do período.

#### Resposta:

20901

### 3. Quais dias do período especificado tiveram o maior número de erros 404.

#### Resposta:

Abaixo é possível ver a lista ordenada, sendo o dia 06/07/1995 o dia com maior índice de erro 404, seguido do dia 19/07/1995

Dia 06-07-1995: 640 ocorrências de Erro 404

Dia 19-07-1995: 639 ocorrências de Erro 404

Dia 30-08-1995: 571 ocorrências de Erro 404

Dia 07-07-1995: 570 ocorrências de Erro 404

Dia 07-08-1995: 537 ocorrências de Erro 404

Dia 13-07-1995: 532 ocorrências de Erro 404

Dia 31-08-1995: 526 ocorrências de Erro 404

Dia 05-07-1995: 497 ocorrências de Erro 404

Dia 03-07-1995: 474 ocorrências de Erro 404

Dia 11-07-1995: 471 ocorrências de Erro 404

Dia 12-07-1995: 471 ocorrências de Erro 404

Dia 18-07-1995: 465 ocorrências de Erro 404

Dia 25-07-1995: 461 ocorrências de Erro 404

Dia 20-07-1995: 428 ocorrências de Erro 404

Dia 24-08-1995: 420 ocorrências de Erro 404

Dia 29-08-1995: 420 ocorrências de Erro 404

Dia 25-08-1995: 415 ocorrências de Erro 404

Dia 14-07-1995: 413 ocorrências de Erro 404

Dia 28-08-1995: 410 ocorrências de Erro 404

Dia 17-07-1995: 406 ocorrências de Erro 404

Dia 10-07-1995: 398 ocorrências de Erro 404

Dia 08-08-1995: 391 ocorrências de Erro 404

Dia 06-08-1995: 373 ocorrências de Erro 404

Dia 27-08-1995: 370 ocorrências de Erro 404

Dia 26-08-1995: 366 ocorrências de Erro 404

Dia 04-07-1995: 359 ocorrências de Erro 404

Dia 09-07-1995: 348 ocorrências de Erro 404

Dia 04-08-1995: 346 ocorrências de Erro 404

Dia 23-08-1995: 345 ocorrências de Erro 404

Dia 27-07-1995: 336 ocorrências de Erro 404

Dia 26-07-1995: 336 ocorrências de Erro 404

Dia 21-07-1995: 334 ocorrências de Erro 404

Dia 24-07-1995: 328 ocorrências de Erro 404

Dia 15-08-1995: 327 ocorrências de Erro 404

Dia 01-07-1995: 316 ocorrências de Erro 404

Dia 10-08-1995: 315 ocorrências de Erro 404

Dia 20-08-1995: 312 ocorrências de Erro 404

Dia 21-08-1995: 305 ocorrências de Erro 404

Dia 03-08-1995: 304 ocorrências de Erro 404

Dia 08-07-1995: 302 ocorrências de Erro 404

Dia 02-07-1995: 291 ocorrências de Erro 404

Dia 22-08-1995: 288 ocorrências de Erro 404

Dia 14-08-1995: 287 ocorrências de Erro 404

Dia 09-08-1995: 279 ocorrências de Erro 404

Dia 17-08-1995: 271 ocorrências de Erro 404

Dia 11-08-1995: 263 ocorrências de Erro 404

Dia 16-08-1995: 259 ocorrências de Erro 404

Dia 16-07-1995: 257 ocorrências de Erro 404

Dia 18-08-1995: 256 ocorrências de Erro 404

Dia 15-07-1995: 254 ocorrências de Erro 404

Dia 01-08-1995: 243 ocorrências de Erro 404

Dia 05-08-1995: 236 ocorrências de Erro 404

Dia 23-07-1995: 233 ocorrências de Erro 404

Dia 13-08-1995: 216 ocorrências de Erro 404

Dia 19-08-1995: 209 ocorrências de Erro 404

Dia 12-08-1995: 196 ocorrências de Erro 404

Dia 22-07-1995: 192 ocorrências de Erro 404

Dia 28-07-1995: 94 ocorrências de Erro 404

### 4. O total de bytes retornados no período, com uma visão acumulada. Por exemplo, se no dia 1 tivemos 50 bytes, dia 2 tivemos 100 bytes e dia 3 mais 150 bytes, sua resposta deverá ser: Dia 1 = 50 bytes, Dia 2 = 150 bytes, Dia 3 = 300 bytes.

#### Resposta:

Abaixo a lista ordenada por data e proporção acumulada de bytes retornados

Dia 01-07-1995: 64714 Bytes

Dia 02-07-1995: 124979 Bytes

Dia 03-07-1995: 214563 Bytes

Dia 04-07-1995: 285015 Bytes

Dia 05-07-1995: 379590 Bytes

Dia 06-07-1995: 480550 Bytes

Dia 07-07-1995: 567783 Bytes

Dia 08-07-1995: 606650 Bytes

Dia 09-07-1995: 641922 Bytes

Dia 10-07-1995: 714782 Bytes

Dia 11-07-1995: 795189 Bytes

Dia 12-07-1995: 887725 Bytes

Dia 13-07-1995: 1021928 Bytes

Dia 14-07-1995: 1106031 Bytes

Dia 15-07-1995: 1151563 Bytes

Dia 16-07-1995: 1199417 Bytes

Dia 17-07-1995: 1274398 Bytes

Dia 18-07-1995: 1338680 Bytes

Dia 19-07-1995: 1411418 Bytes

Dia 20-07-1995: 1478011 Bytes

Dia 21-07-1995: 1542640 Bytes

Dia 22-07-1995: 1577907 Bytes

Dia 23-07-1995: 1617106 Bytes

Dia 24-07-1995: 1681365 Bytes

Dia 25-07-1995: 1744064 Bytes

Dia 26-07-1995: 1802913 Bytes

Dia 27-07-1995: 1864593 Bytes

Dia 28-07-1995: 1891714 Bytes

Dia 01-08-1995: 1925710 Bytes

Dia 03-08-1995: 1967098 Bytes

Dia 04-08-1995: 2026655 Bytes

Dia 05-08-1995: 2058548 Bytes

Dia 06-08-1995: 2090968 Bytes

Dia 07-08-1995: 2148330 Bytes

Dia 08-08-1995: 2208487 Bytes

Dia 09-08-1995: 2268945 Bytes

Dia 10-08-1995: 2330193 Bytes

Dia 11-08-1995: 2391439 Bytes

Dia 12-08-1995: 2429510 Bytes

Dia 13-08-1995: 2465990 Bytes

Dia 14-08-1995: 2525868 Bytes

Dia 15-08-1995: 2584715 Bytes

Dia 16-08-1995: 2641368 Bytes

Dia 17-08-1995: 2700356 Bytes

Dia 18-08-1995: 2756602 Bytes

Dia 19-08-1995: 2788696 Bytes

Dia 20-08-1995: 2821659 Bytes

Dia 21-08-1995: 2877199 Bytes

Dia 22-08-1995: 2934961 Bytes

Dia 23-08-1995: 2993058 Bytes

Dia 24-08-1995: 3045610 Bytes

Dia 25-08-1995: 3102931 Bytes

Dia 26-08-1995: 3134539 Bytes

Dia 27-08-1995: 3167362 Bytes

Dia 28-08-1995: 3222858 Bytes

Dia 29-08-1995: 3290846 Bytes

Dia 30-08-1995: 3371487 Bytes

Dia 31-08-1995: 3461612 Bytes

### Versionamento:

* Github repositório: <https://github.com/DennCardoso/Ame-Digital-Nasa>
* Denncardoso’s Github: <https://github.com/denncardoso>