

CNJ INOVA	Ciência de Dados e Inteligência Artificial
Desafio 2	Inconsistências de Dados nos Sistemas dos Tribunais
Time 7	Componentes: Nicholas Araujo Clarindo – nicholas.araujo@gmail.com Cláudio Cavalcante - claudio.cavalcantevas@gmail.com Juliana Junqueira - julianajunqueira78@gmail.com Marcus M Riether – marcus.riether@gmail.com Tharcisio Fernandes - tharcisiofernand@gmail.com
Data	21/10/2020

PROPOSTA TÉCNICA RELATÓRIO PRINCIPAL

Sumário

1. Introdução	2
2. O que foi pedido no Desafio 2.....	2
3. Nossa proposta	3
4. Identificação de necessidades.....	3
5. Workflow da arquitetura proposta	6
6. Conclusão técnica.....	25
7. Referências	27

PROPOSTA TÉCNICA RELATÓRIO PRINCIPAL

1. Introdução

O ciclo de inovação aberta CNJ INOVA, regido pelo Edital Nº 120/2020, PROCESSO Nº 04600.002925/2020-39, da ESCOLA NACIONAL DE ADMINISTRAÇÃO PÚBLICA – Enap, apresenta-se como uma oportunidade única para o desenvolvimento de soluções inovadoras pautadas no uso da ciência de dados e inteligência artificial. Direcionada à visão de aperfeiçoamento do trabalho do sistema judiciário brasileiro, principalmente no que diz respeito ao controle e à transparência administrativa e processual, essa iniciativa aporta ao CNJ um elemento de excelência em planejamento estratégico, governança e gestão judiciária, impulsionando a efetividade da Justiça brasileira no sentido de promover os valores de justiça e a paz social.

Formado por especialistas em infraestrutura e analítica em ecossistemas de big data, ciência de dados e direito, o time 7 conta com profissionais com experiência ampla e diversificada em diversos mercados, particularmente em empresas de grande porte. Apoiados em competências e habilidades técnicas e jurídicas e após analisar criteriosamente as especificações do edital, o time aponta seus esforços para propor uma solução moderna e de reconhecida efetividade em ambientes de tráfego volumoso e intenso de dados complexos, primando pela entrega de um produto de alta qualidade, viável e que atende às necessidades do CNJ em sua completude.

A proposta ora apresentada fundamenta-se nos seguintes elementos:

1. **Manutenção do ecossistema já existente nos Tribunais e no CNJ**, com ajustes e/ou redirecionamento de fluxos de dados e informações e modernização do parque tecnológico, tornando o sistema mais robusto, mais produtivo e mais eficiente;
2. **Priorização do uso de ferramentas *open source* em todo o espectro da solução;**
3. **Foco na limpeza e na gestão da qualidade dos dados dos repositórios nacionais**, em especial na identificação de ocorrências de campos ausentes, inconsistentes ou atípicos nos registros encaminhados pelos Tribunais;
4. **Interface direta com os Tribunais para o envio em tempo real de relatórios diagnósticos de inconsistências;**
5. **Preparação do ambiente para acoplamento de algoritmos de inteligência artificial que operem de forma integrada à solução proposta.**

Com suporte nesses elementos estruturantes, a proposta é dividida em 11 etapas que se integram, formando o ambiente em sua completude.

2. O que foi pedido no Desafio 2

O desafio 2 demanda o desenvolvimento de algoritmos capazes de serem utilizados, tanto como ferramentas de limpeza, em que seja possível corrigir informações errôneas, quanto como recursos de gestão de qualidade do Datajud. Atenção deve ser dada na identificação de campos ausentes, inconsistentes, atípicos ou fora do padrão estipulado pelo CNJ. Os algoritmos devem ser capazes, ainda, de propor soluções de saneamento dos registros, sempre que possível.

3. Nossa proposta

Partindo da especificação do desafio 2, conforme edital, o time 7 apresenta, neste relatório principal, uma **arquitetura orientada a eventos (EDA), construída sobre a plataforma Hadoop**. Ainda hoje, a existência de um grande número de silos de dados com baixa capacidade de integração e comunicação é uma realidade comum de muitas empresas. Em última instância, essa realidade afeta não somente os processos internos, mas também os externos e, claro, desvia o foco de trabalho, que deveria estar centrado no cliente da informação e em suas necessidades.

Nesta proposta, trazemos a **ideia de unificação das bases de processos judiciais, via implantação de uma estrutura central de gerenciamento de dados (master data management – MDM) operando em tempo real**. Como principais benefícios, destacamos:

- a. **Plataforma *omnichannel* de gestão de dados e com baixo acoplamento** (todas as comunicações acessam ou são devolvidas por canal único e mínimo grau de dependência entre aplicativos, permitindo que novas ferramentas sejam agregadas ao sistema de forma ágil e com baixo impacto);
- b. **Otimização da estrutura de governança de dados, incrementando a rastreabilidade das movimentações processuais:**
 - Viabilização de controle de regras e de processos perante as novas diretrizes da Lei Geral de Proteção de Dados Pessoais (LGPD).
 - Auditoria utilizando ferramentas com capacidade de rastrear e entender de maneira abrangente como os dados são usados no Big Data (Cloudera Navigator).
- c. **Base única centralizada com dados consistidos:**
 - Eliminação de redundâncias de dados;
 - Minimização dos aspectos de má qualidade de dados;
 - Uniformização e a consolidação de conceitos sobre os dados e informações técnicas e administrativas;
- d. **Baixo custo de implantação**, no que se refere a ferramentas, tendo sido priorizado o uso de ferramentas *open source*;
- e. **Suporte a dados oriundos de fontes e tipos diversos.**

Por questões de transparência, apontamos as principais restrições na implantação desta solução:

- a. Tempo de implantação médio/alto, não somente pela amplitude e abrangência do projeto em si, mas sobretudo pela complexidade do tema subjacente;
- b. Mão-de-obra de mercado ainda limitada.

4. Identificação de necessidades

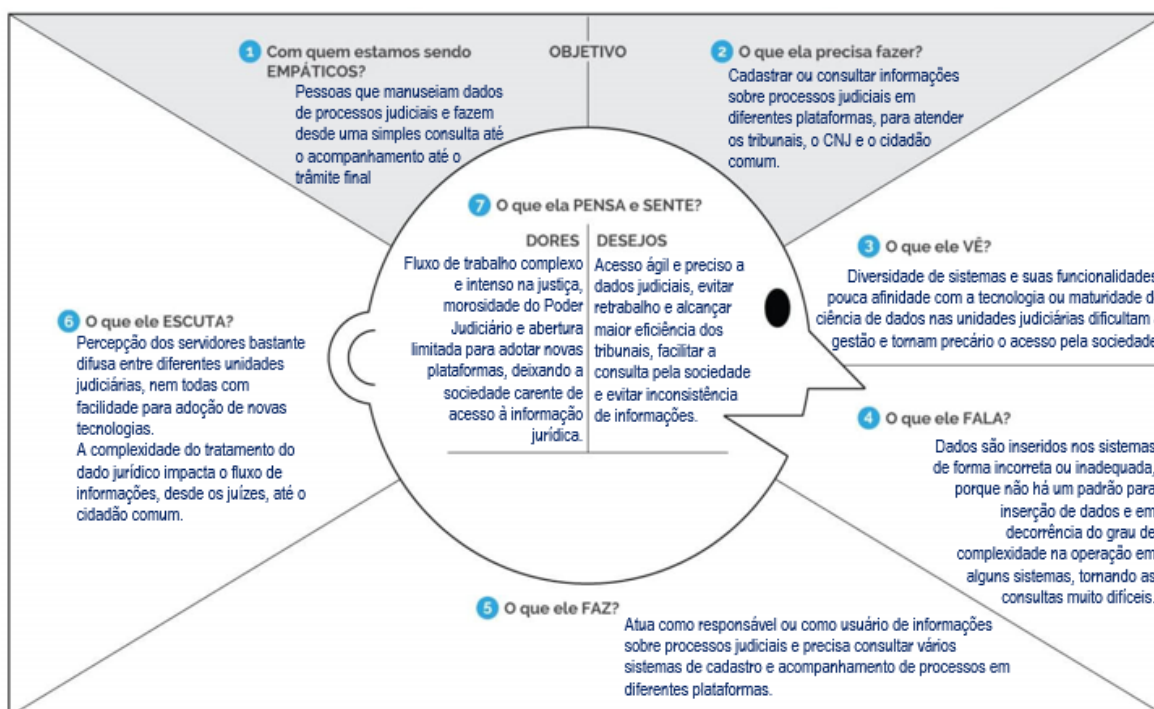
4.1. Mapa de empatia

A análise das personas propostas como material provocativo oferecido pelo CNJ INOVA leva à reflexão sobre quem seria de fato o usuário final de um ferramenta tecnológica mais avançado para atendimento às demandas de todo o sistema jurídico brasileiro.

O país convive, e o fará ainda por muito tempo, com deficiências de formação tecnológica. E essa não é uma realidade apenas do cidadão comum, mas atinge também uma grande faixa de pessoas de diferentes idades e com diferentes graus de formação e de atuação profissional.

Entendemos que as melhorias que devem ser promovidas na governança de dados jurídicos pelo CNJ e os reflexos que essas melhorias precisam causar no sistema jurídico brasileiro como um todo devem, em última instância, chegar até a sociedade e permitir que essas duas pontas, de um lado o CNJ, de outro o cidadão comum, sejam unidas por um processo jurídico mais eficiente, com benefícios para ambas as partes.

Com base nessas ideias, apresentamos o mapa de empatia a seguir, contendo dores, desejos, impressões e sentimentos que afetam todos aqueles que, em algum momento, interagem com o dado jurídico, seja porque o volume de trabalho é muito elevado e revela baixa produtividade em unidades judiciárias, seja porque as consultas aos processos judiciais são, por si só, complexas e ainda sofrem com sistemas pouco amigáveis, dificultando o acesso e o entendimento sobre o que realmente está se passando dentro dos Tribunais.



4.2. Matriz CSD

Em seguida, apresentamos a matriz CSD, por meio da qual expressamos o que é possível de se endereçar numa proposta dessa natureza. As suposições e as dúvidas, esperamos, deverão nos trazer bons e ricos momentos de discussão, com a qualificação do nosso trabalho para nos levar por esse caminho.

Matriz CSD – Certezas, Suposições e Dúvidas			
	Certezas	Suposições	Dúvidas
Atores	Os servidores nas diferentes unidades judiciárias possuem diferentes graus de afinidade com soluções tecnológicas e o cidadão comum tem dificuldade de acesso a informações judiciais.	Pode haver resistência por parte dos servidores em se adaptar a novas rotinas de trabalho. Do ponto de vista da sociedade, há pouco preparo para o uso de tecnologia. Além disso, a	É possível criar políticas públicas capazes de promover a consistência e uniformização de dados do judiciário, facilitando o trabalho do servidor e o acesso aos dados pela sociedade?

		disparidade social com relação à inclusão digital é latente.	
Cenários	As unidades judiciárias possuem sistemas de cadastro e acompanhamento de processos que seguem padrões tecnológicos distintos. Alguns desses sistemas com grau de complexidade alto, inclusive para atuantes do direito.	As unidades judiciárias dispõem de orçamentos limitados e equipes com diferente disposição para adotar novas plataformas tecnológicas. Há resistência por algumas pessoas na aceitação de atividades tecnológicas por medo de serem substituídas pela tecnologia e perderem o emprego.	É possível unificar regras de tecnologia em todo o sistema judiciário brasileiro? Em que parcela substituirá do trabalho do servidor do Judiciário a máquina substituirá o homem?
Regras	Promover a limpeza e correção de dados inconsistentes, promovendo maior celeridade e transparência no processo judiciário, simplificando a operação e aumentando a efetividade no saneamento das inconsistências cadastrais.	No futuro, a correção de dados inconsistentes deverá atingir todo o parque tecnológico do sistema judiciário.	A implementação da solução irá retroalimentar os sistemas onde as falhas de informação se originam, sendo capaz de corrigir as inconsistências atuais e impedir a ocorrência de novas?

4.3. 5W2H

Uma aplicação da ferramenta 5W2H foi adotada como checklist administrativo de atividades, prazos e responsabilidades que precisarão ser desenvolvidos com clareza e eficiência por todos os envolvidos na implantação da solução aqui proposta.

- 4.3.1. WHAT – Criar um processo de MDM para a base de processos judiciais, coletando dados oriundos dos sistemas dos Tribunais.
- 4.3.2. WHY – A solução será implantada para resolver o problema da inconsistência e das falhas de dados cadastrais e de movimentação de processos judiciais.
- 4.3.3. WHERE – A implantação do processo de MDM ocorrerá no ambiente do CNJ, onde serão mantidas todas as informações de todos os sistemas legados já existentes, inclusive sem impactos nos sistemas dos Tribunais.
- 4.3.4. WHO – A implantação será feita pelo time 7, com apoio da equipe interna da DPJ, para mapeamento das origens.
- 4.3.5. WHEN – A implantação da solução, o time obedecerá aos prazos de implantação conforme previsto no edital.
- 4.3.6. HOW – Toda base irá seguir um padrão, onde a transformação do dado ocorrerá durante o processo de carga em tempo real.
- 4.3.7. HOW MUCH – Para a versão inicial será necessário:
- 4.3.8. Indicadores de viabilidade do Projeto:

Infraestrutura

Os ambientes podem ser virtualizados, mas recomendamos como melhores práticas o descritivo abaixo:

- **Cloudera:**

3 servidores para alta disponibilidade para produção;
1 servidor para homologação;
1 servidor para desenvolvimento;
Valor aproximado: 140 mil cada servidor
Spark: Será executado os processos dentro do ambiente Hadoop;
Valor: n/a

- **Nifi:**

3 servidores para produção;
1 servidor para homologação (pode ser ambiente virtualizado);
1 servidor para desenvolvimento (pode ser ambiente virtualizado);
Valor aproximado: 140 mil cada servidor

4.4. Principais benefícios mensuráveis:

- **Otimização da estrutura de governança de dados e incremento da rastreabilidade das movimentações processuais, facilitando a aplicação de regras para cumprimento da LGPD;**
- **Criação de plataforma omnichannel de gestão de dados e com baixo acoplamento;**
- **Criação de base única centralizada de processos judiciais com dados consistidos:**
 - a) Eliminação de redundâncias de dados;
 - b) Minimização dos aspectos de má qualidade de dados;
 - c) Uniformização e consolidação de conceitos sobre os dados e informações técnicas e administrativas;
 - d) Criação de indicadores para tomadas de decisão;
- **Baixo custo de implantação, no que se refere a ferramentas;**
- **Suporte a dados oriundos de fontes e tipos diversos.**

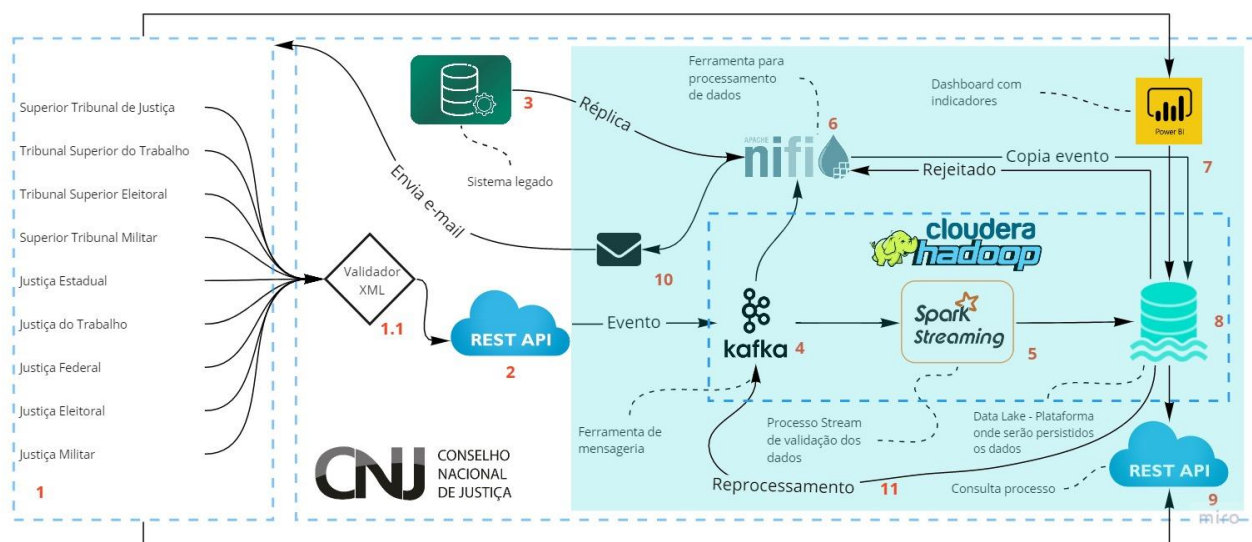
4.5. Principais benefícios não mensuráveis:

- **Melhoria no processo operacional com a padronização da arquitetura;**
- **Aumento na satisfação dos usuários internos e externos ao sistema judiciário;**
- **Visualização de todas as informações sobre processos judiciais de forma rápida e confiável;**

5. Workflow da arquitetura proposta

A figura a seguir apresenta um esquema da arquitetura completa e o fluxo de dados e informações. Para efeito didático, o processo foi dividido em 11 etapas distintas, que serão descritas mais adiante de forma pormenorizada. De modo resumido, o fluxo pode ser entendido da seguinte forma:

- Os arquivos XML são gerados pelos Tribunais e validados, utilizando o Programa Validador de Arquivos XML;
- Após validados, os arquivos são armazenados em um repositório e são gerados protocolos para cada tribunal de origem, informando a quantidade de processos recebidos, a detecção de possíveis problemas de envio e algumas informações adicionais;
- Com o número de protocolo, o Tribunal deverá acessar a aplicação responsável pela gestão dos protocolos, em que acompanhará, em tempo real, o processamento dos arquivos recebidos pelo CNJ;
- Será implantado um processo de validação automática dos dados, que verificará a existência de inconsistências na Chave Única Processual;
- Se for identificada inconsistência na chave, o registro será rejeitado por completo. Nesse caso, a solução de envio/recepção atualizará o protocolo de *status*, informando os motivos da rejeição. Diante disso, o tribunal poderá corrigir as inconsistências e proceder com o reenvio.
- Após o processo de validação, os registros validados passarão por uma fase de enriquecimento de dados e serão armazenados na base única de dados (data lake/HDFS) e, por fim, serão disponibilizados nos Painéis de Informação do DataJud [1].



Cabe destacar que a solução proposta é enfaticamente a implantação do ferramental contido na caixa azul do diagrama.

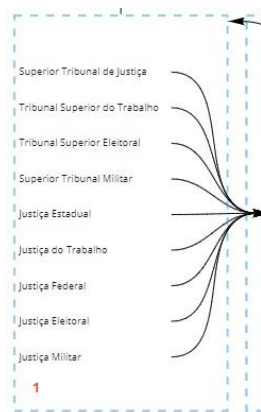
6.1 Etapas da arquitetura proposta

Para acesso ao ecossistema onde foi desenvolvido a solução, será necessário conectar na VPN. O arquivo de senhas e certificados estão no GITHUB.

https://github.com/nicholasclarindo/CNJInovaTime7Desafio2/blob/main/documentacao_tecnica/vpn-lab.zip

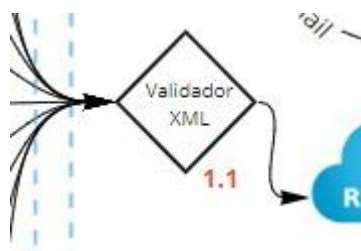
usuario: user_cnj / password: labcnj

- **Etapa 1 – Envio de dados pelos Tribunais**



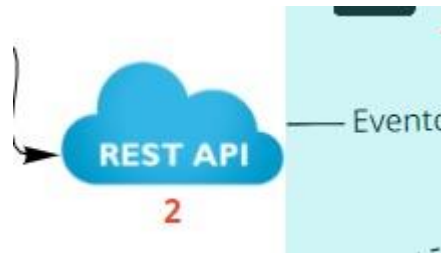
A etapa 1 da arquitetura proposta refere-se ao envio dos dados processuais pelos Tribunais ao CNJ, em conformidade com o disposto no artigo 3º da Resolução CNJ nº 331/2020. Em atendimento ao Edital 120/2020 - ENAP e com vistas a não impactar a atividade de envio e recebimento de dados, foi mantida a proposta tecnológica atualmente existente nos sistemas dos Tribunais, sem qualquer alteração. Assim sendo, os dados enviados serão recebidos e validados em formato XML, mantidas as exigências de envio de dados de todos os processos físicos ou eletrônicos, públicos ou sigilosos, de qualquer das classes previstas nas Tabelas Processuais Unificadas – TPUs, criadas pela Resolução CNJ nº 46/2007.

- **Item 1.1 – Programa validador de XML**



O item 1.1 do workflow indica que o programa de validação dos dados continuará sendo em formato XML, mantendo-se a forma já utilizada atualmente. [2]

- **Etapa 2 - Processo de envio de dados de acordo com o Modelo de Transmissão de Dados – MTD**



A etapa 2 vem complementar a etapa 1 com relação ao envio dos dados, conforme artigo 4º da Resolução CNJ nº 331/2020. Os arquivos XML são enviados ao CNJ, atendendo as especificações do Modelo de Transmissão de Dados – MTD. Os dados recebidos e validados em formato XML serão enviados para a API REST. **Nesse momento, o fluxo será alterado, de modo que a API remeta os dados recebidos em formato XML para o sistema de mensageria Kafka.** Este fluxo será o mesmo para todos os Tribunais, independentemente da origem e do grau de jurisdição.

- **Etapa 3 – Sistema legado contendo as informações de gestão das TPUs**



A etapa 3 confirma a manutenção da arquitetura dos sistemas legados do CNJ, **sendo que as tabelas processuais unificadas (TPUs) não serão alteradas** e continuarão sendo alimentadas de forma integrada aos sistemas já implantados nos Tribunais e atualmente em uso pela Justiça. [3]

- **Etapa 4 – Sistema de mensageria Apache Kafka**



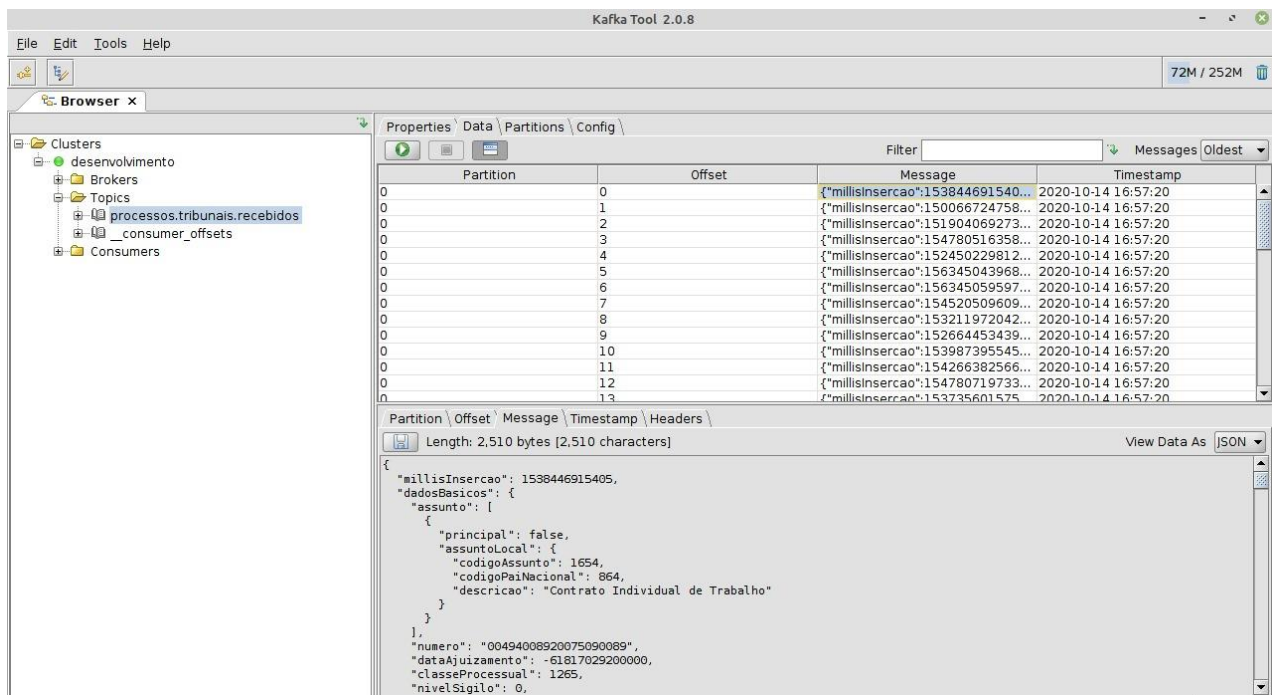
O Apache Kafka é uma plataforma open source de processamento de streams desenvolvida pela Apache Software Foundation e tem como objetivo fornecer uma plataforma unificada, de alta capacidade e baixa latência para tratamento de dados em tempo real [4]. Presta-se como sistema de mensageria, permitindo a comunicação simultânea entre várias fontes e seus consumidores em ambiente distribuído, realizando processamento de streams de dados. **Na solução proposta, o papel da ferramenta é informar a disponibilização dos dados recebidos pela API e o local onde é feito todo tratamento dos dados, validação e padronização das informações.**

Além de ser umas das melhores (e, por isso, mais populares) ferramentas disponíveis no mercado para processos de stream, o Kafka já faz parte da solução Hadoop Cloudera, dispensando-se, portanto, a necessidade de adquirir uma licença à parte.

Algumas razões da contínua popularidade e adoção do Kafka no mercado são [5]:

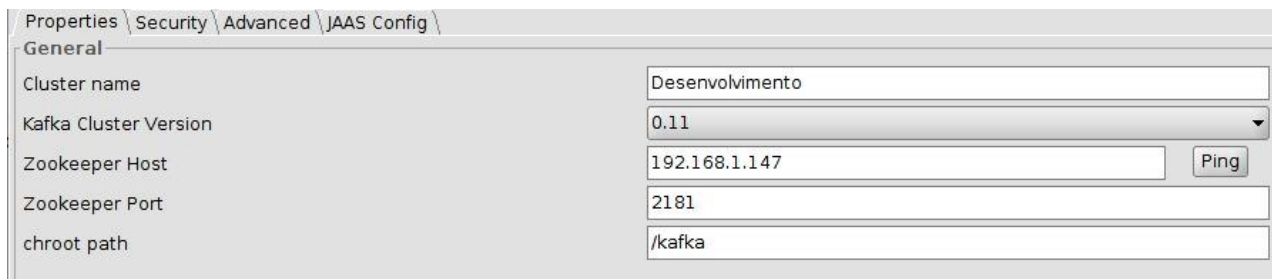
- a. **Escalabilidade** – dois importantes recursos do Kafka contribuem para sua escalabilidade. Um cluster Kafka pode se expandir ou encolher (os brokers podem ser adicionados ou removidos) enquanto opera, sem perigo de quedas. Ao mesmo tempo, um tópico Kafka pode ser expandido com o intuito de conter mais partições. Devido ao fato de que a partição não pode se expandir através de brokers múltiplos, sua capacidade está restrita ao espaço de disco do broker. Ser capaz de aumentar o número de partições e de brokers significa que não há limite na quantidade de informação que um tópico sozinho pode armazenar.
- b. **Tolerância a falhas e confiabilidade** – o Kafka foi projetado de maneira que uma falha com um broker seja detectável por outros em um cluster. Uma vez que cada tópico pode ser replicado em múltiplos brokers, o cluster pode se recuperar de tais falhas e continuar a operar sem nenhuma interrupção de serviço.
- c. **Desempenho** – os brokers conseguem armazenar e recuperar informações de forma eficiente em uma velocidade super-rápida.

As telas a seguir demonstram algumas etapas da ferramenta em operação.

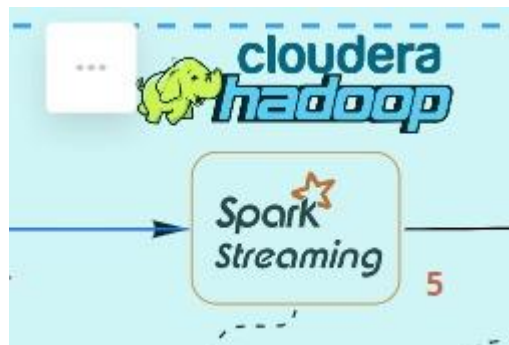


Para acesso ao Kafka, será necessário instalar um cliente ou via linha de comando.

No exemplo acima, foi utilizado o Kafka Tool. Segue abaixo para conexão:




- **Etapas 5 – Apache Spark Streaming**



O Spark Streaming com linguagem Scala é uma extensão da API core do Spark que goza de alta escalabilidade e alta disponibilidade de processamento em clusters com paralelismo, **sendo fortemente**



tolerante a falhas em streams de dados [6]. A ferramenta absorve facilmente dados de muitas fontes variadas e, na presente proposta, fará a extração de dados diretamente do Kafka (etapa 4), produzindo duas entregas:



RUNNING Applications

- Cluster
- About
- Nodes
- Node Labels
- Applications
- NEW
- SAVING
- SUBMITTED
- ACCEPTED
- RUNNING
- FINISHED
- FAILED
- KILLED
- Scheduler
- Tools

Cluster Metrics									
Apps Submitted	Apps Pending	Apps Running	Apps Completed	Containers Running	Memory Used	Memory Total	Memory Reserved	VCores	
2	0	2	0	3	9 GB	18 GB	0 B	3	

Cluster Nodes Metrics				
Active Nodes	Decommissioning Nodes	Decommissioned Nodes	Lost Nodes	Unhealthy Nodes
3	0	0	0	0

User Metrics for dr who									
Apps Submitted	Apps Pending	Apps Running	Apps Completed	Containers Running	Containers Pending	Containers Reserved	Memory Used	Memory Pending	Memory Reserved
0	0	0	0	0	0	0 B	0 B	0 B	0 B

Scheduler Metrics			
Scheduler Type	Scheduling Resource Type	Minimum Allocation	Maximum Allocation
Fair Scheduler	[memory-mb (unit=Mi), vcores]	<memory:3072, vCores:1>	<memory:18432, vCores:4>

ID	User	Name	Application Type	Queue	Application Priority	StartTime	LaunchTime	FinishTime	State	FinalStatus	Running Containers	Allocated CPU VCoers	Allocated Memory MB	Reserved CPU VCoers	Res Mem
application_1603072769825_0002	user_cnj	CNJDataProcessing-Streaming	SPARK	root.users.user_cnj	0	Sun Oct 18 23:03:34 -0300 2020	Sun Oct 18 23:03:35 -0300 2020	N/A	RUNNING	UNDEFINED	2	2	6144	0	0
application_1603072769825_0001	user_cnj	oozie:launcher:T=spark-W=CNJDataProcessing-Streaming-A=spark-7e7b-ID=0000000-201019020108957-oozie-oozi-W	Oozie Launcher	root.users.user_cnj	0	Sun Oct 18 23:03:16 -0300 2020	Sun Oct 18 23:03:14 -0300 2020	N/A	RUNNING	UNDEFINED	1	1	3072	0	0

Showing 1 to 2 of 2 entries

ETL:

Extract - Coletando dados em tempo real do tópico processos.tribunais.recebidos.

Transform – Cruzando e validando as informações com as tabelas de domínios (TPUs). O Job também utiliza e aplica regras de validação conforme documentação.

GIT:

https://github.com/nicholasclarindo/CNJInovaTime7Desafio2/blob/main/documentacao_tecnica/CNJ%20Planilha%20regras_Time7.xlsx

Load - O fluxo do Job se divide em dois, dados validos e inválidos, segue abaixo:

- Válidos – São considerados dados validos aqueles que passaram por todo o processo de validação dos metadados e possuem chave completa, verificação de campos ausentes, onde deveriam vir preenchidos e dados fora do padrão, segue abaixo as tabelas:

< cnj

Tables (12) + ↻

Filter...

codigolocalidade
competencia
movimento
mpm_serventias
processosrecebidos
rejeitados
relatorio_dtb_brasil_distrito
relatorio_dtb_brasil_municipio
relatorio_dtb_brasil_subdistrito
sgt_assuntos
sgt_classes
sgt_movimentos

- Rejeitados – São considerados dados inválidos, todos os registros que apesar da chave validada possuem campos inconsistentes, incoerentes ou ausentes.



Query: SELECT * FROM cnj.rejeitados LIMIT 100;

Query History

atributo	motivo	valor	numero	classeprocessual	codigoorgao	gr
dadosBasicos.valorCausa	Está fora do padrão.	NULL	00000696220109130001	11046	14141	G
dadosBasicos.valorCausa	Está fora do padrão.	NULL	00002202820109130001	11041	14141	G
dadosBasicos.valorCausa	Está fora do padrão.	NULL	00003821720109130003	11037	14143	G

Uma segunda vantagem no uso do Spark é sua **capacidade nata de processar algoritmos de inteligência artificial em streams de dados**. Embora essa seja uma possibilidade real e fortemente atrativa, o uso de modelos de Inteligência Artificial deve ser criterioso e exige tempo para sua aplicação de forma correta e responsável.

Abaixo segue o link contendo o endereço do git para o Spark:

<https://github.com/ClaudioCavalcanteVas/cnj-spark-streaming-data-processing-hackaton/tree/main>

Abaixo segue o link contendo o endereço do git para script das tabelas:

https://github.com/nicholasclarindo/CNJInovaTime7Desafio2/blob/main/documentacao_tecnica/criar_tabelas_inova.txt

- **Etapa 6 – Processamento dos dados coletados pela ferramenta Nifi**



A etapa 6 consiste na coleta e processamento de dados do Kafka pela ferramenta Nifi.

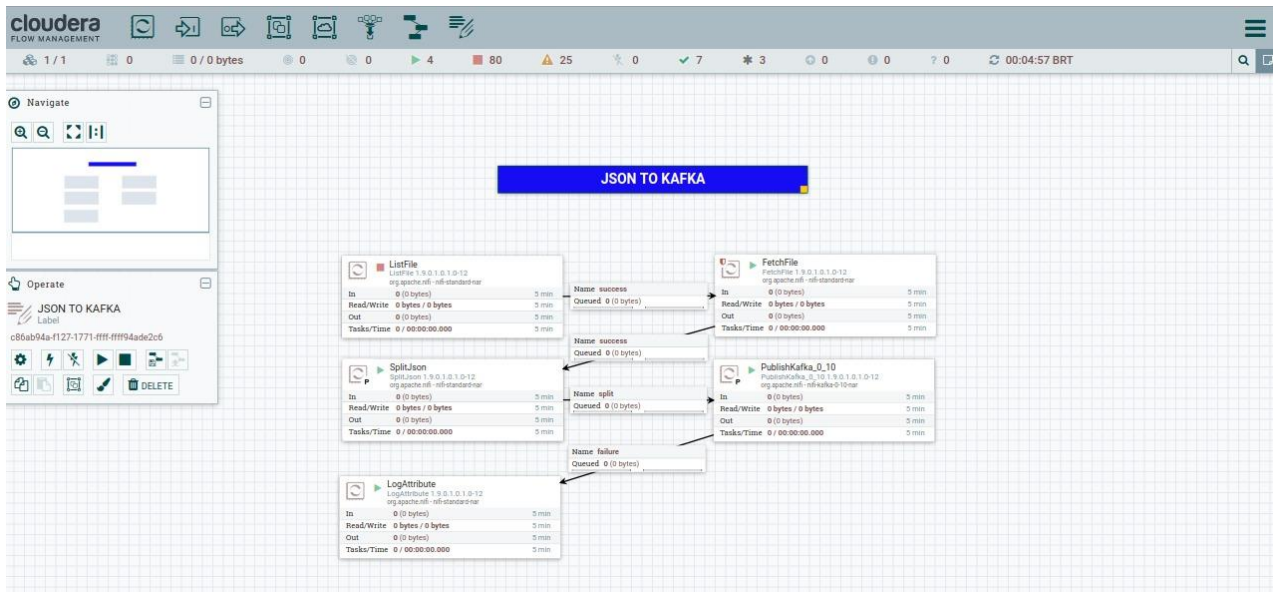
Nifi é um processador de dados que faz a extração e transformação dos mesmos, se necessário, realizando a movimentação de dados dentro da arquitetura e integrando as ferramentas.

O Nifi possui um repositório para controle de versão da pipelines, segue abaixo:

cloudera Nifi Registry / All				Sort by:	Name (a - z)
PIPE_IMPORT_TABLES - Cnj	Flow	VERSIONS 1			
PIPE_Postgres - Cnj	Flow	VERSIONS 3			
PIPE_CLASSES - Cnj	Flow	VERSIONS 2			
PIPE_KAFKA_TO_IMPALA - Caixa	Flow	VERSIONS 2			
PIPE_RETORNO_EMAIL - Cnj	Flow	VERSIONS 1			
PIPE_SEND_JSON_TO_KAFKA - Cnj	Flow	VERSIONS 8			
PIPE_TESTE_IMPALA - Cnj	Flow	VERSIONS 1			
PUT_FILES_CSV_HADOOP - Cnj	Flow	VERSIONS 5			
Teste Hbase - Caixa	Flow	VERSIONS 1			
Teste SQL - Caixa	Flow	VERSIONS 1			
Teste Script - Caixa	Flow	VERSIONS 1			

O Nifi será utilizado para 3 processos:

- **Backup** - coletar os eventos do Kafka e enviar uma cópia desses dados para dentro do Data Lake.



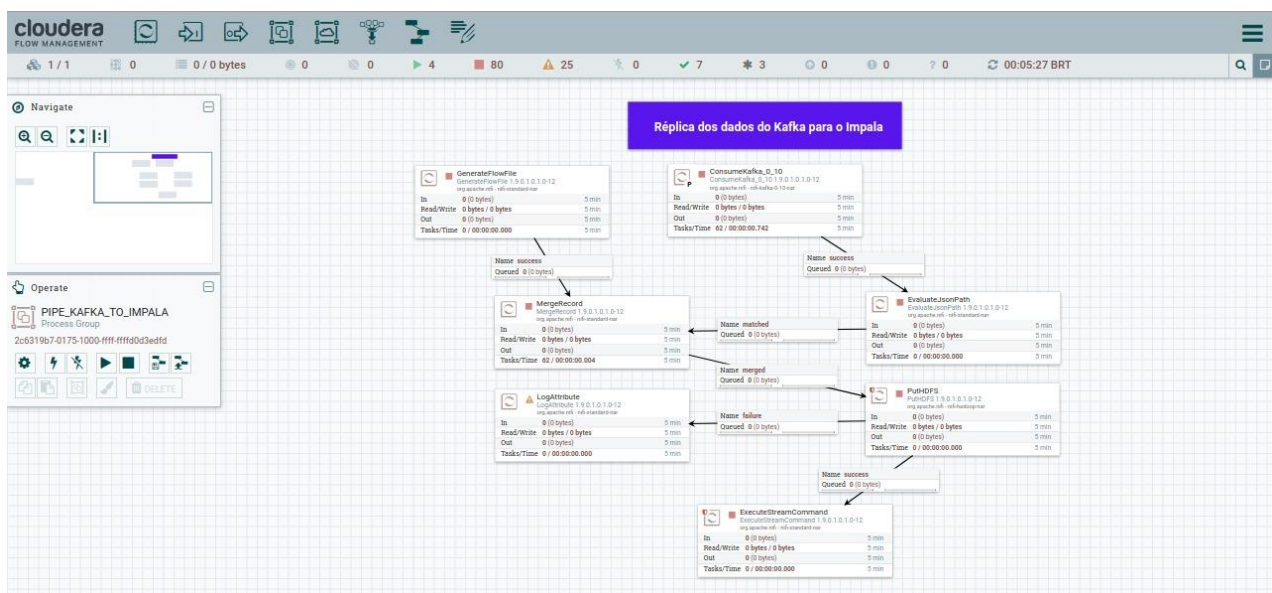


Query: `SELECT * FROM cnj.processosrecebidos LIMIT 100;`

Results (100):

arquivos
1
2
3
4
5
6
7
8
9
10

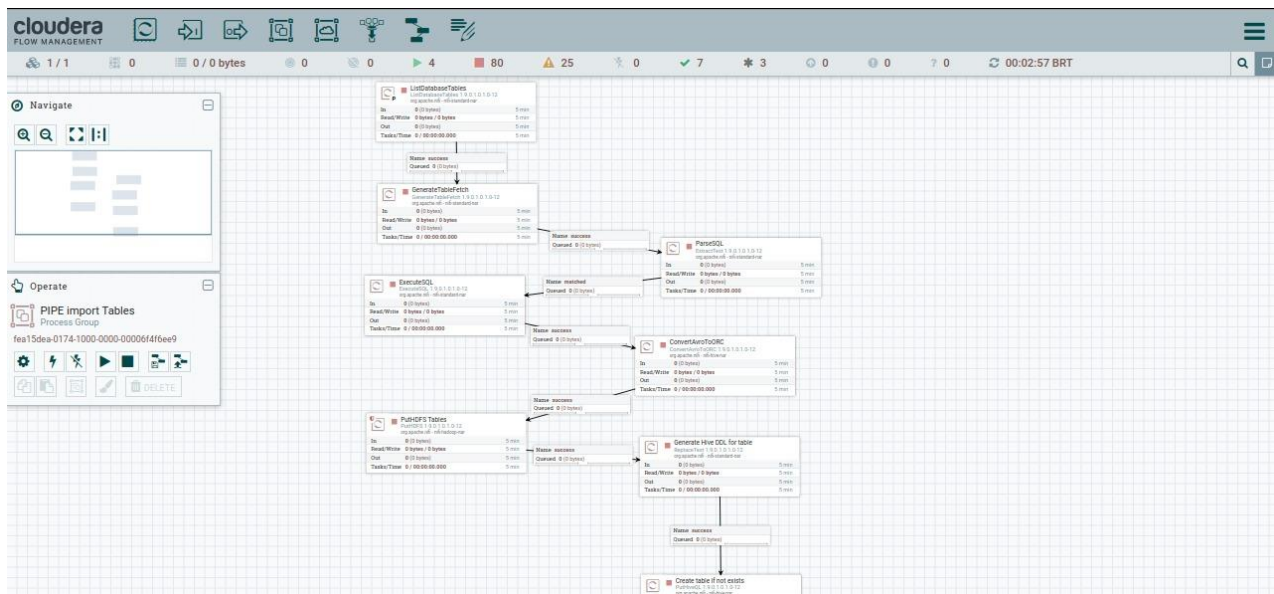
- **Réplica** - a base de dados no sistema legado será fotografada e persistida no Data Lake, onde serão utilizadas as tabelas de domínio no processamento do Job Spark, para validação e enriquecimento dos dados.



- **E-mail informativo** – o Nifi coleta dados das tabelas de rejeitados e envia as informações consolidadas por e-mail aos Tribunais uma vez ao dia.

Destaca-se que a presente estrutura está totalmente voltada para uma arquitetura em tempo real.

Abaixo, um print do ELT pipeline criada no Nifi:



Para conectar ao Nifi, será necessário acessar o link abaixo:

<http://192.168.1.147:8080/nifi/>

- **Etapa 7 – visualizando informações com o Power BI**



A etapa 7 refere-se à elaboração de dashboards que serão disponibilizados para os Tribunais e para o CNJ por meio do Power BI, apresentando informações na forma de indicadores de inconsistências, entre outros. Foram exportados dados para geração dos relatórios. A própria ferramenta poderá ser usada para apresentação das informações públicas que o CNJ desejar.

O Power BI é a mais recente ferramenta de BI da Microsoft, no estilo self-service. É uma coleção de serviços e aplicativos que trabalham de forma conjunta para tornar fontes de dados não relacionadas em dados interativos e de fácil visualização, sendo especialmente voltado para o acompanhamento de métricas e indicadores. Ele é capaz de conectar diferentes fontes de dados e operar de forma integrada as ferramentas de big data.

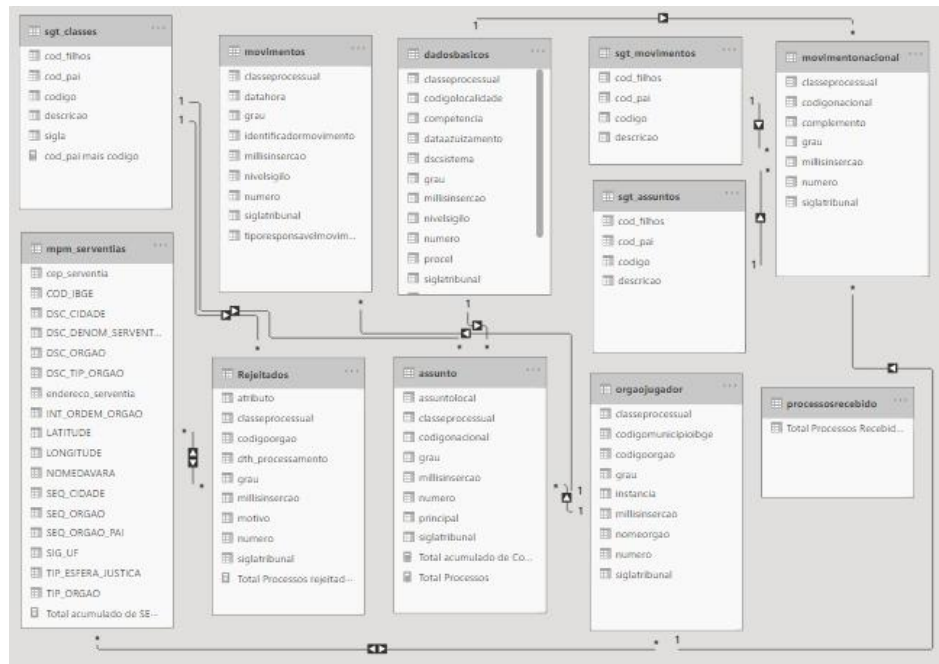
Observação: Por motivos de limitação na infraestrutura e banda de internet, importamos os dados do Hadoop para o desenvolvimento local. Onde foi desenvolvido apenas com uma pequena amostra dos dados.



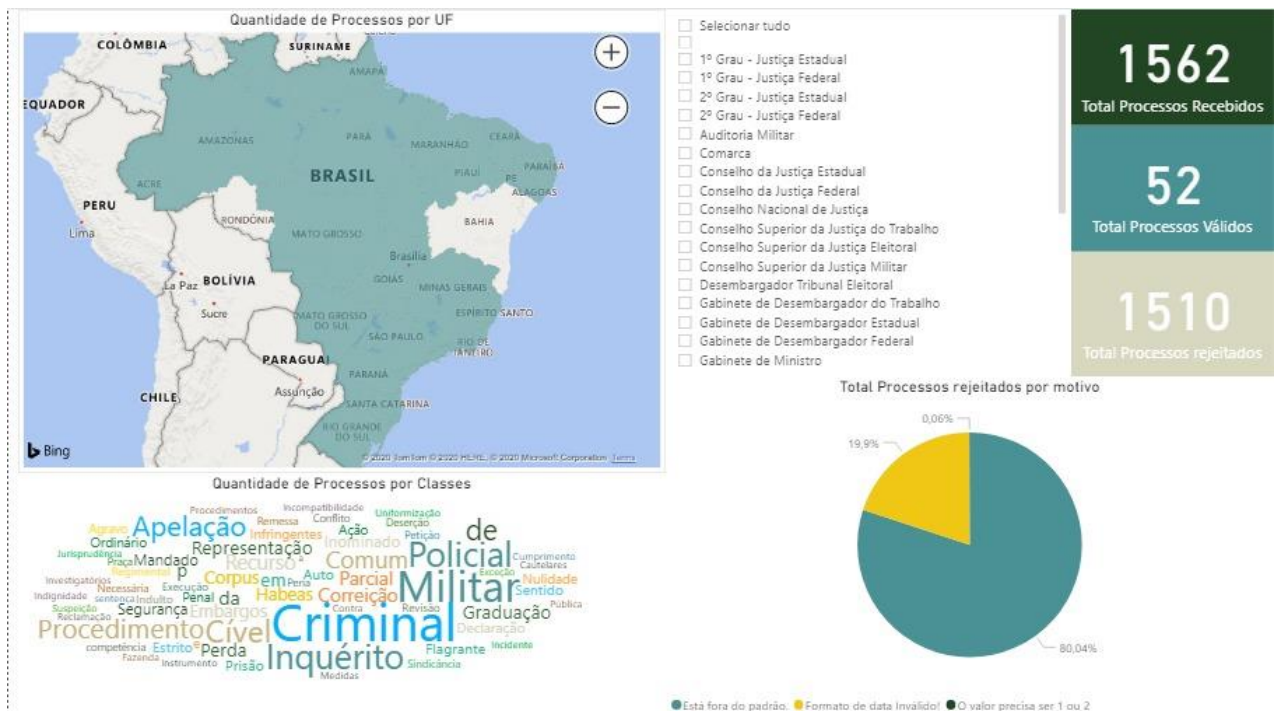
GIT dos arquivos para o Dashboard:

<https://github.com/nicholasclarindo/CNJInovaTime7Desafio2/blob/main/dashboard/tabelas%20para%20o%20Dashboard.rar>

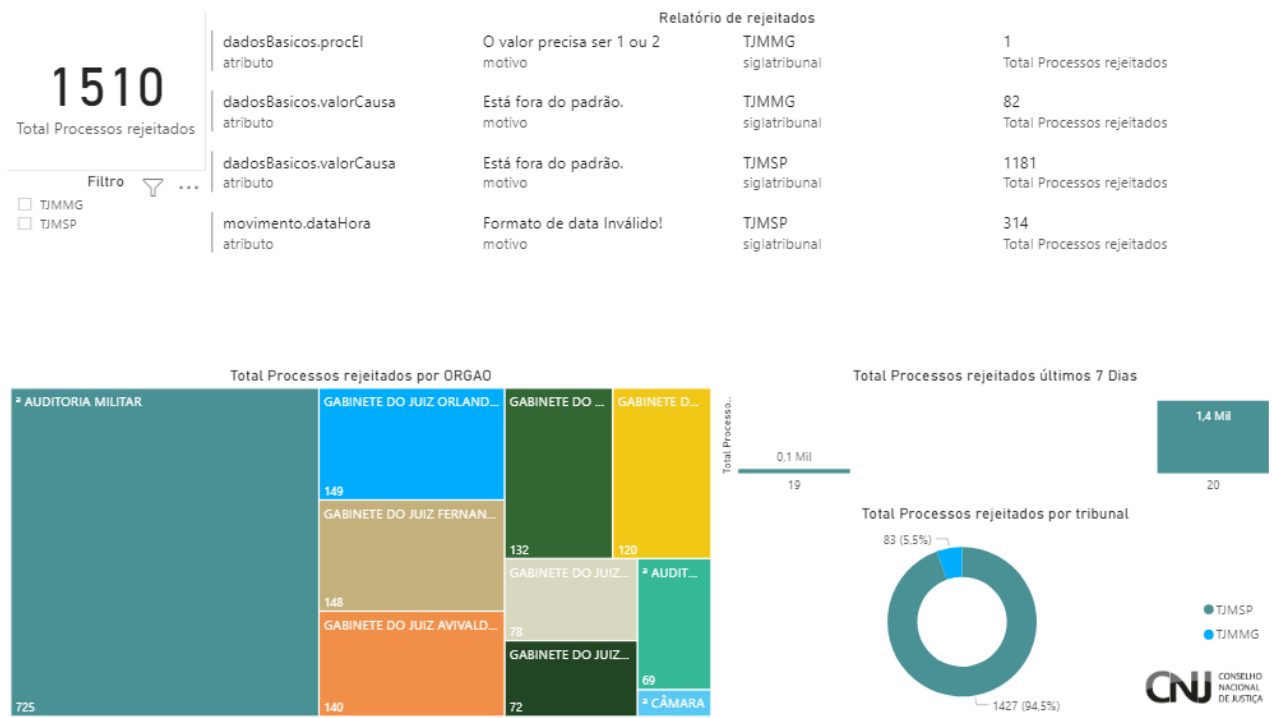
- Modelagem dos dados no Power BI



- Dashboard para o CNJ



- **Dashboard para os Tribunais**

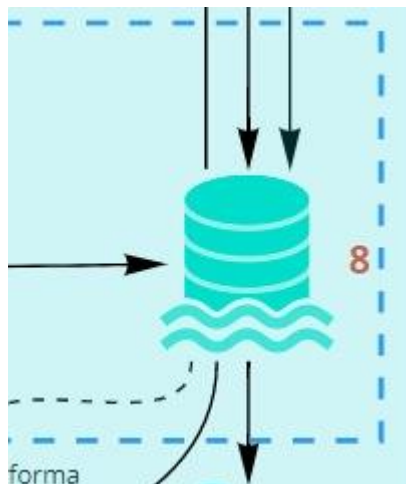


- **Dashboard Informações Públicas** – o painel (link) que já existe hoje pode ser reapontado para novas bases no Hadoop.
https://paineis.cnj.jus.br/QvAJAXZfc/opendoc.htm?document=qvw_l%2FPainelCNJ.qvw&host=QVS%40neodimio03&anonymous=true&sheet=shPDPrincipal

GIT do Dashboard:

https://github.com/nicholasclarindo/CNJInovaTime7Desafio2/blob/main/dashboard/DashboardCNJ_Time7.pbix

- **Etapa 8 – Armazenamento de dados e grandes arquivos pelo Data Lake.**



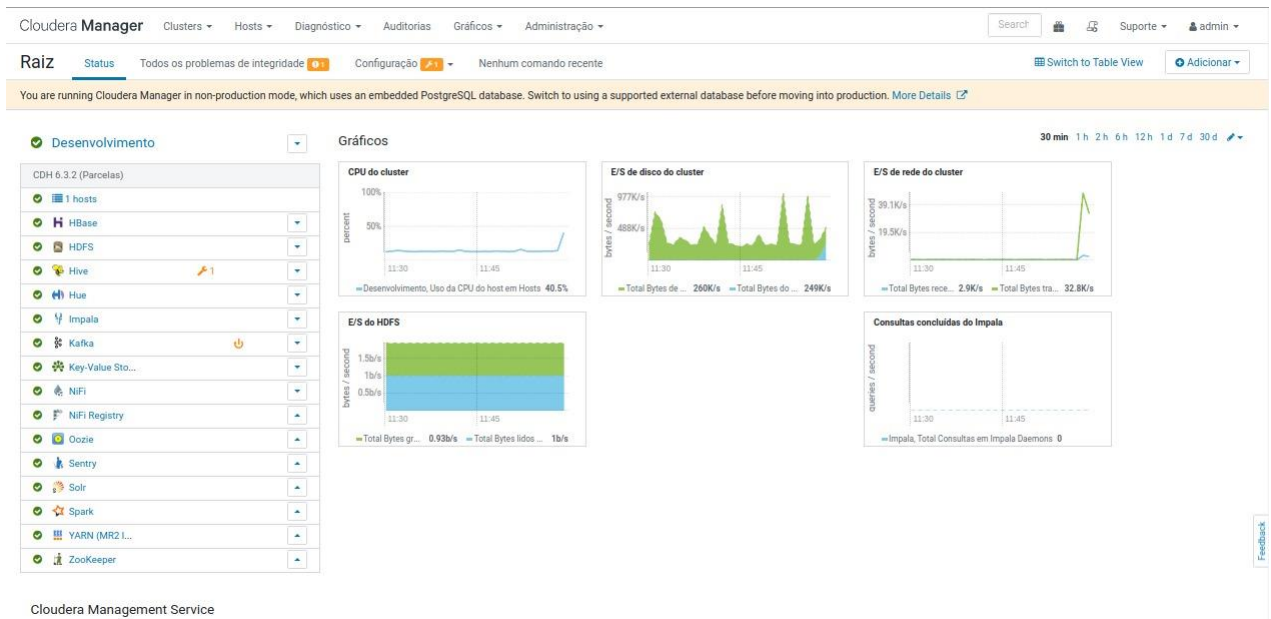
A etapa 8 ilustra a proposta de construção de um Data Lake, **estrutura que goza de alta disponibilidade, alta escalabilidade e capacidade de trabalhar com grandes volumes de dados. No Data Lake serão persistidas todas as informações do Datajud para o CNJ.**

“Um data lake ou lago de dados é um sistema ou repositório de dados armazenados em seu formato natural/bruto, geralmente objetos blobs ou arquivos. Um data lake geralmente é um armazenamento único de todos os dados corporativos, incluindo cópias brutas dos dados do sistema de origem e dados transformados usados para tarefas como relatórios, visualização, análise avançada e aprendizado de máquina. Um data lake pode incluir dados estruturados de bancos de dados relacionais (linhas e colunas), dados semiestruturados (CSV, logs, XML, JSON), dados não estruturados (emails, documentos, PDFs) e dados binários (imagens, áudio, vídeo).”. [7]

Quando falamos em Big Data, devemos sempre nos lembrar dos principais 5 V's do Big Data — que foram categorizados pela Gartner, uma empresa de pesquisas em Tecnologia da Informação que cunhou o termo.

- a. **Volume** — se vemos o big data como uma pirâmide, o volume é a base. Segundo a Gartner, o volume de dados que as empresas gerenciam disparou a partir da primeira década dos anos 2000. Desde então, esse volume dobra a cada 40 meses.
- b. **Velocidade** — além de gerenciar dados, as empresas precisam que essas informações fluam rapidamente — o mais próximo possível do tempo real. A velocidade pode ser mais importante que o volume, porque pode nos dar uma maior vantagem competitiva. Às vezes, é melhor ter dados limitados em tempo real do que muitos dados em baixa velocidade.
- c. **Variedade** — Uma empresa pode obter dados de várias fontes diferentes: de dispositivos internos à tecnologia GPS de smartphones ou o que as pessoas estão dizendo nas redes sociais. A importância dessas fontes de informação varia de acordo com a natureza do negócio. Por exemplo, um serviço ou produto de mercado de massa deve estar mais ciente das redes sociais do que uma empresa industrial. E esses dados podem ter muitas camadas, com valores diferentes.
- d. **Veracidade** — O quarto V é a veracidade, que neste contexto é equivalente à qualidade. Temos todos os dados, mas poderíamos estar perdendo alguma coisa? Os dados são “limpos” e precisos? Eles realmente têm algo a oferecer?
- e. **Valor** — Finalmente, o V de valor fica no topo da pirâmide de Big Data. Isso se refere à capacidade de transformar um tsunami de dados em negócios. Com uma boa estratégia de Big Data, as empresas podem, por exemplo, saber dentro de um período de dois meses quando é altamente provável que um cliente cancele sua compra recorrente ou quando estará mais propenso a fazer novas aquisições.”. [8]

A figura a seguir é uma cópia de tela demonstrando o ambiente já instalado e configurado.



HDFS – Hadoop File System

Hadoop	Overview	Datanodes	Datanode Volume Failures	Snapshot	Startup Progress	Utilities
--------	----------	-----------	--------------------------	----------	------------------	-----------

Browse Directory

/

Go!

Show 25 entries

Search:

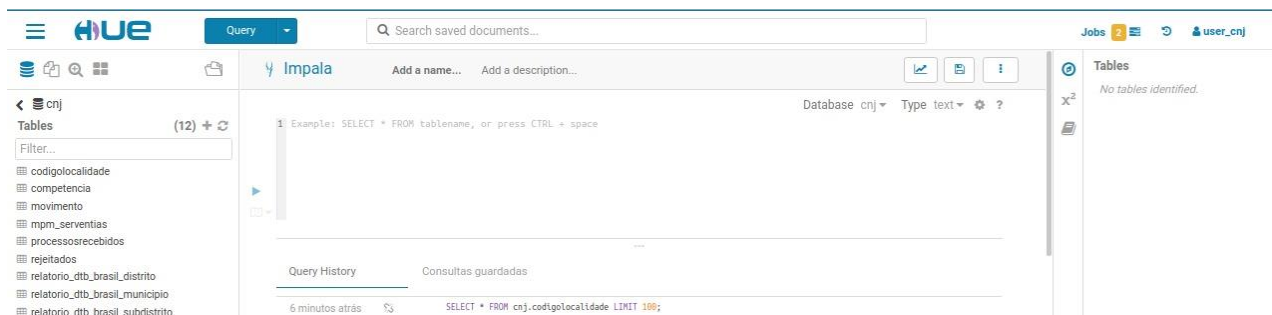
Permission	Owner	Group	Size	Last Modified	Replication	Block Size	Name
drwxrwx---	hive	hive	0 B	Sep 30 11:00	0	0 B	data
drwxr-xr-x	hbase	hbase	0 B	Oct 18 15:11	0	0 B	hbase
drwxrwxrwx	root	supergroup	0 B	Oct 06 10:24	0	0 B	lib
drwxrwxr---	solr	solr	0 B	Sep 15 00:06	0	0 B	solr
drwxrwxrwx	hdfs	supergroup	0 B	Sep 15 00:11	0	0 B	tmp
drwxrwxrwx	hdfs	supergroup	0 B	Sep 30 11:08	0	0 B	user

Showing 1 to 6 of 6 entries

Previous 1 Next

Hadoop, 2018.

HUE

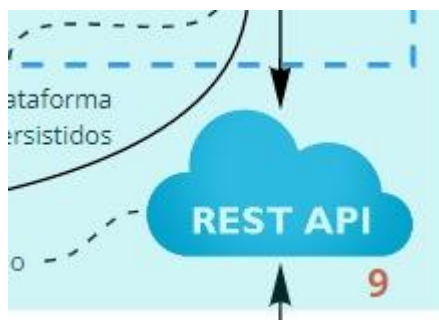


Para conectar ao HUE e toda a administração do ambiente, será necessário acessar o link abaixo:

<http://192.168.1.147:8889/>

usuario: user_cnj / password: labcnj

- **Etapa 9 – Criação de API para consultas de processos enviados**



A etapa 9 consiste na criação de uma API que permite a realização de consulta aos processos que foram enviados. Para tanto, as informações serão buscadas no Hadoop, para que os Tribunais possam consultar os processos e verificar os respectivos status.

Micro serviço: <http://192.168.1.147:3000>

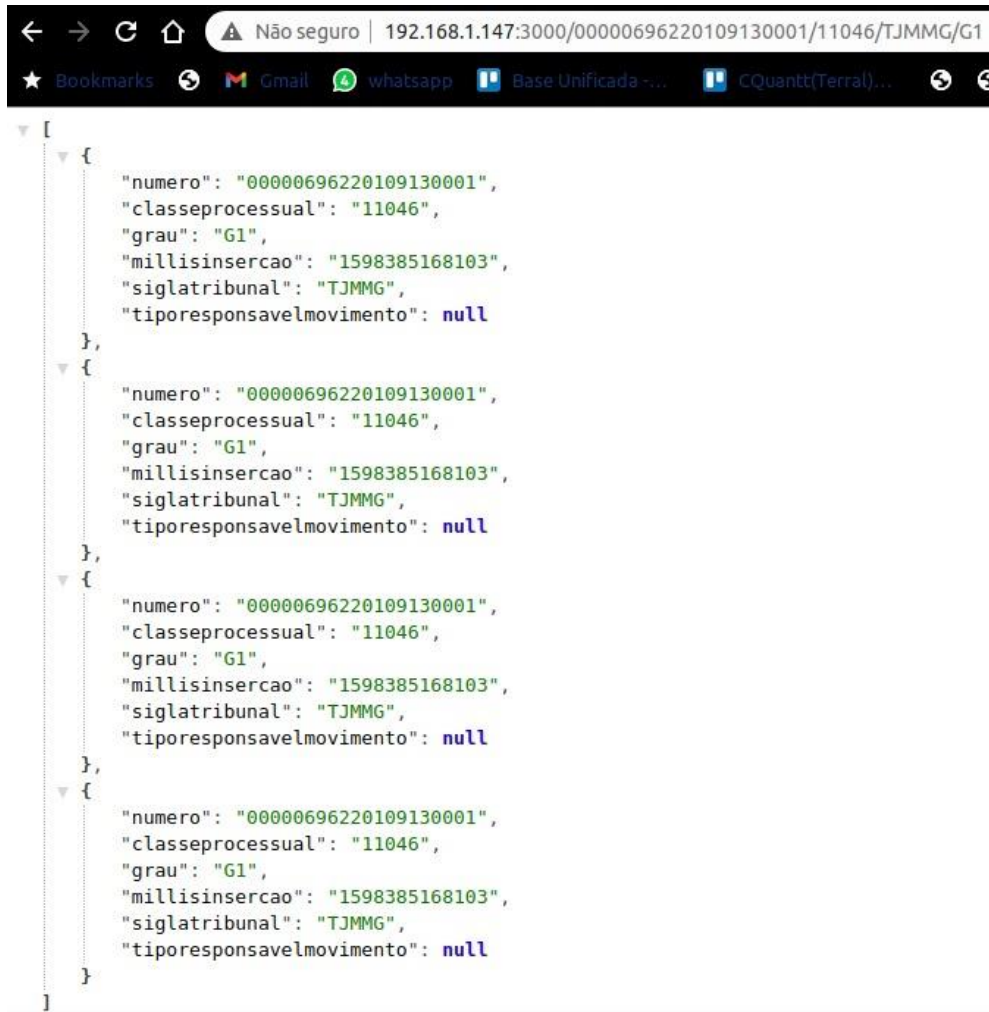
A API retorna os movimentos do processo consultado

Parameters

GET

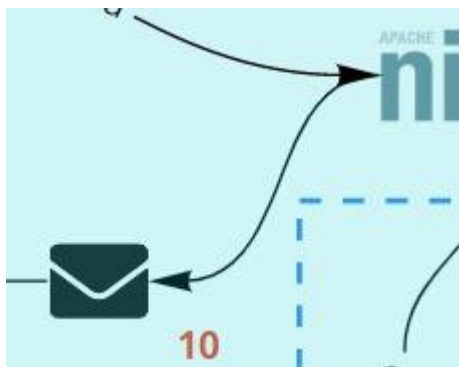
<http://192.168.1.147:3000/<numero>/<classeProcessual>/<siglaTribunal>/<grau>>

Exemplo: <http://192.168.1.147:3000/00000696220109130001/11046/TJMMG/G1>



GIT do micro serviço: <https://github.com/ClaudioCavalcanteVas/cnj-microservice-node-hackaton>

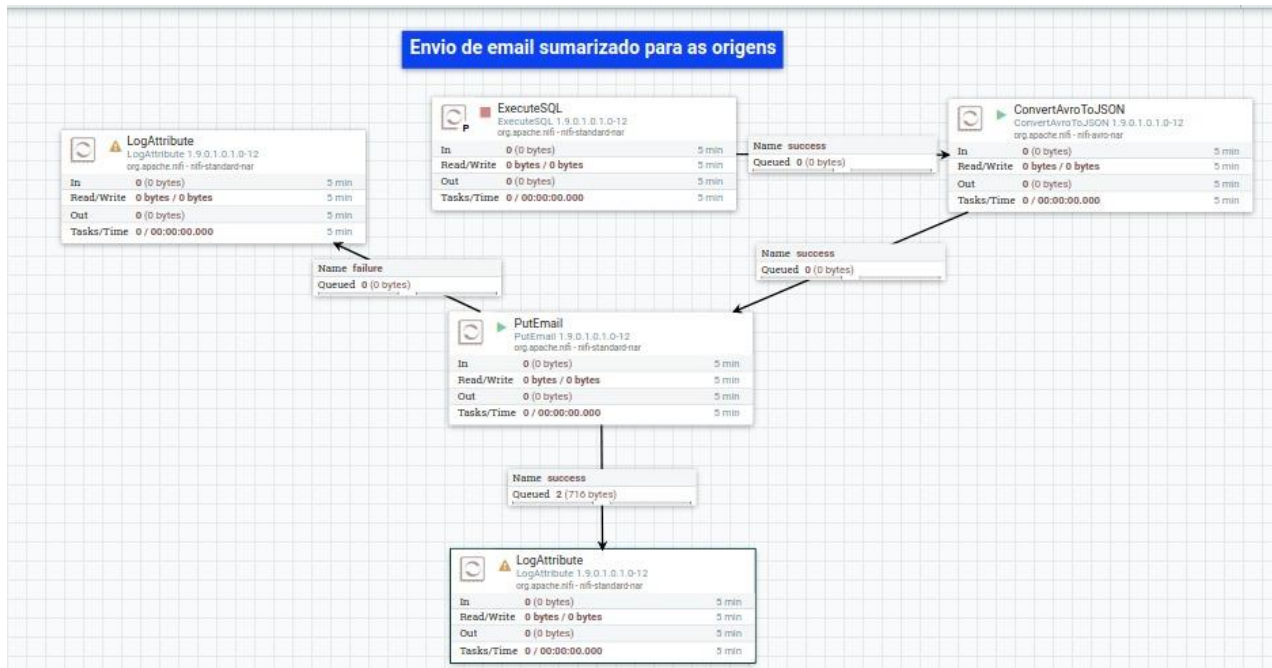
- ***Etapas 10 – Envio de e-mails aos Tribunais***



A etapa 10 consiste no processamento de envio de e-mails aos Tribunais, contendo informações resumidas sobre os dados coletados e processados.

Serão disparados e-mails uma vez ao dia para todos os Tribunais, com resumo das informações de todos os dados processados nesse mesmo dia. Esse disparo será realizado pela ferramenta Nifi, que buscará informações no Data Lake e gerará informações consolidadas para os Tribunais sobre a quantidade de informações processadas e de erros ocorridos.

Processo do Nifi



Template do e-mail enviado aos Tribunais.

Message from NiFi  Caixa de entrada x



tharcisiofernand@gmail.com

para mim ▾

Análise completa e bem-sucedida!

Segue um informativo resumizado do processamento do arquivo.

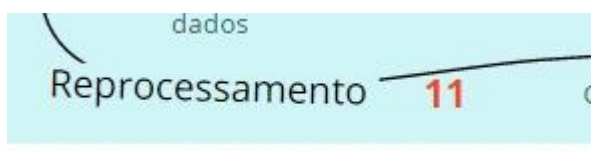
Tribunais	UF	Processos Rejeitados
TJMG	Minas Gerais	52
TJMMG	Minas Gerais	171

Para maiores informações acesse o link abaixo:

<http://cnj/informacoes/processos>

...

- Item 11 – Reprocessamento de dados rejeitados



Será criado um processo de reprocessamento dos eventos onde qualquer alteração ou evolução nas regras do processo de qualificação, podendo reprocessar os eventos históricos já recebidos pelo CNJ, sem a necessidade de envio novamente pelos Tribunais.

Print da pipeline:



6. Conclusão técnica

Observamos que os arquivos JSON recebidos para efeito do hackathon apresentavam diferentes padrões para os diferentes tribunais, o que, a rigor, não deveria ter acontecido, uma vez que o processo atual conta com o programa **validador de XML** operando localmente nos tribunais, donde se depreende que os metadados seguem o padrão estabelecido pelo CNJ, ainda que a massa de dados apresente as ausências e inconsistências que eram o objeto de trabalho proposto para o desafio 2.

Considerando que a falta de padrão dos arquivos JSON não era esperada de acordo com o Glossário, houve um gasto de tempo muito acima do razoável para se identificar uma forma de corrigir esse problema, que não fazia parte do escopo inicial do projeto. Durante as lives, houve questionamentos pelos concorrentes sobre a possibilidade de concentrar esforços em apenas um tribunal, porém, se a falta de padrão dos JSON era conhecida previamente, acreditamos que isso deveria ter sido anunciado, favorecendo aos times o investimento na correção dos erros e inconsistências, onde de fato ideias precisam ser implementadas.

Seguem abaixo dois exemplos:

processos-trt24_5

identificadorMovimento

tipoResponsavelMovimento

```
array ▶ 0 ▶ movimento ▶ 0 ▶
  ▾ array [287]
    ▾ 0 {5}
      millisInsercao : 1600849087451
                        2020-09-23T08:18:07.451Z
      dadosBasicos {11}
        siglaTribunal : TRT24
        movimento [34]
          ▾ 0 {6}
            identificadorMovimento : 9552193
            tipoResponsavelMovimento : 0
            movimentoNacional {1}
              nivelSigilo : 0
            orgaoJulgador {4}
              dataHora : 20200320133209
```

processos-tjal_10

```
array ▶ 0 ▶ movimento ▶ 2 ▶ movimentoNacional ▶
  ▾ array [926]
    ▾ 0 {5}
      millisInsercao : 1592185067070
                        2020-06-15T01:37:47.070Z
      dadosBasicos {8}
        siglaTribunal : TJAL
        movimento [14]
          ▶ 0 {2}
          ▶ 1 {2}
          ▾ 2 {2}
            movimentoNacional {1}
              codigoNacional : 1061
              dataHora : 20200319210427
```

processos-tjal_10

processos-tjmmg_2

dscSistema

procEl

grau

```
array ▶ 0 ▶ dadosBasicos ▶
  ▢ ▼ array [926]
  :: ▢ ▼ 0 {5}
  :: ▢ millisInsercao : 1592185067070
  :: ▢ 2020-06-15T01:37:47.070Z
  :: ▢ ▼ dadosBasicos {8}
  :: ▢ ▶ assunto [1]
  :: ▢ numero : 07002926020208020082
  :: ▢ dataAjuizamento : 20200318105109
  :: ▢ totalAssuntos : 1
  :: ▢ classeProcessual : 159
  :: ▢ nivelSigilo : 0
  :: ▢ ▶ orgaoJulgador {4}
  :: ▢ codigoLocalidade : 2704302
  :: ▢ siglaTribunal : TJAL
  :: ▢ ▼ movimento [14]
  :: ▢ ▶ 0 {2}
```

```
array ▶ 0 ▶ dadosBasicos ▶
  ▢ ▼ array [1989]
  :: ▢ ▼ 0 {5}
  :: ▢ millisInsercao : 1598385168103
  :: ▢ 2020-08-25T19:52:48.103Z
  :: ▢ ▼ dadosBasicos {10}
  :: ▢ ▶ assunto [2]
  :: ▢ dscSistema : 8
  :: ▢ numero : 00000696220109130001
  :: ▢ procEl : 1
  :: ▢ dataAjuizamento : 20100107000000
  :: ▢ totalAssuntos : 2
  :: ▢ classeProcessual : 11046
  :: ▢ nivelSigilo : 0
  :: ▢ ▶ orgaoJulgador {4}
  :: ▢ codigoLocalidade : 3106200
  :: ▢ siglaTribunal : TJMMG
  :: ▢ ▶ movimento [4]
  :: ▢ grau : G1
  :: ▢ ▶ 1 {5}
```




7. Referências

- (1) Fonte: <https://www.cnj.jus.br/sistemas/datajud/#>
- (2) Fonte: <https://www.cnj.jus.br/sistemas/datajud/orientacoes/#>
- (3) Fontes: https://www.cnj.jus.br/sgt/consulta_publica_classes.php
<https://www.cnj.jus.br/sgt/versoes.php>
- (4) Fonte: https://pt.wikipedia.org/wiki/Apache_Kafka
- (5) Fonte: [Tecmundo.com.br](https://www.tecmundo.com.br)
- (6) Fonte: https://pt.wikipedia.org/wiki/Apache_Spark
- (7) Fonte: https://pt.wikipedia.org/wiki/Data_Lake
- (8) Fonte: <https://in360.com.br/blog/cinco-vs-do-big-data-velocidade/>