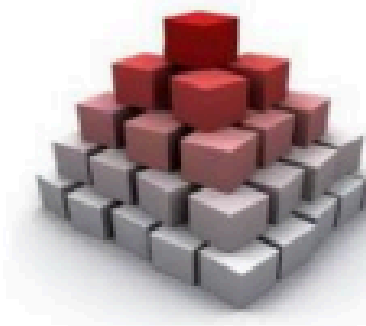


FINTECH

QUANTUMFINANCE

# DATA ARCHITECTURE, INTEGRATION AND INGESTION PROJETO INTEGRADO

IVAN FERREIRA GANCEV

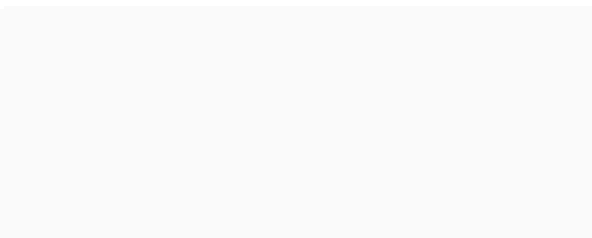


- • • • • +
- • • • •
- + • **Data Architecture, Integration and Ingestion**
- + •

A **QuantumFinance** está no mercado, e os investidores veem muito potencial na nova Fintech. Porém, com o crescimento e o investimento acelerado existe a necessidade de melhor escalabilidade. Com este cenário em mente, a **QuantumFinance** contrata a **SmartMind Analytics** para construir a arquitetura de dados, visando a escalabilidade de todo o ambiente de dados, habilitando maior agilidade, estabilidade e confiança na entrega, acompanhando o crescimento da empresa.

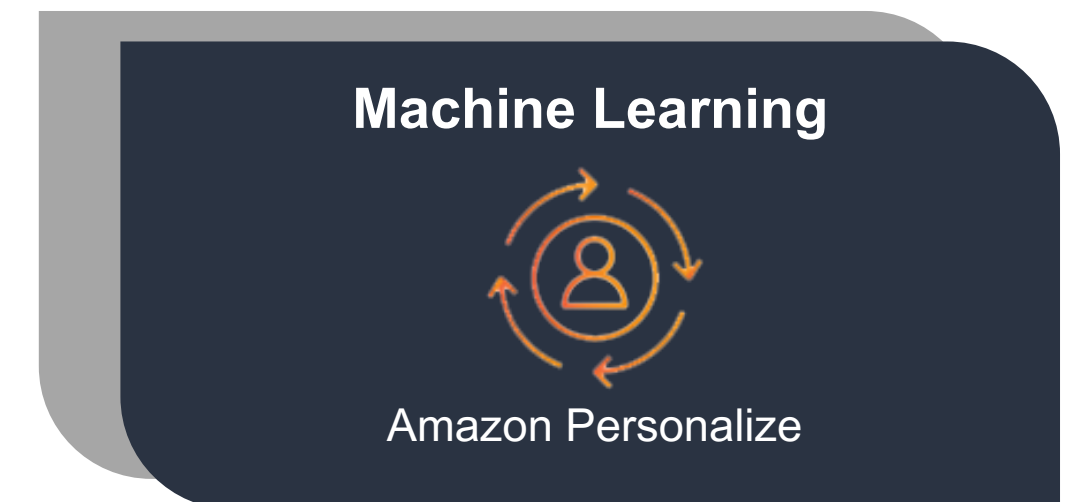
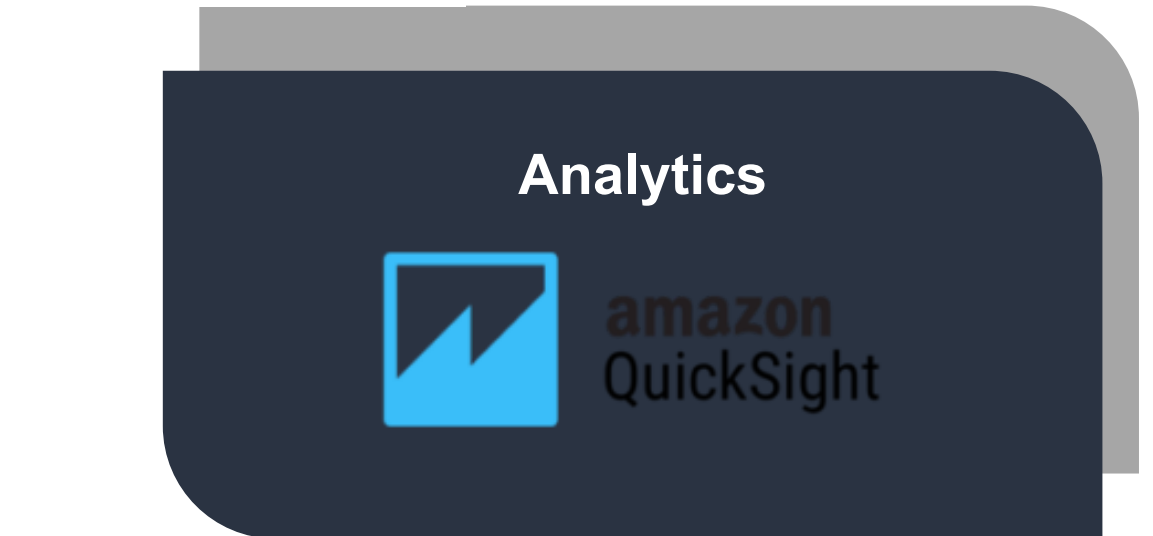
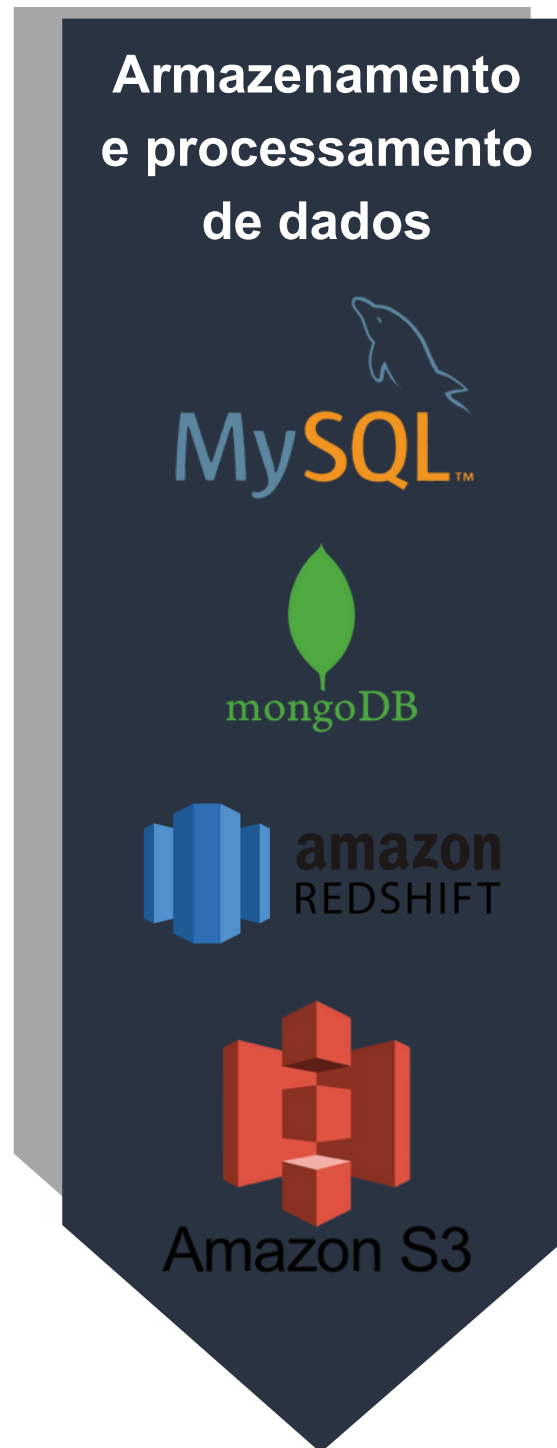
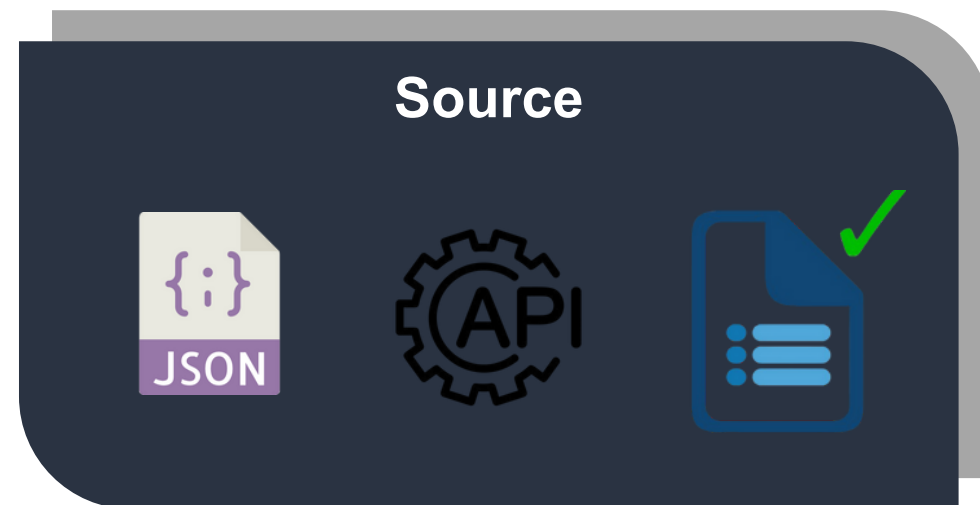
Essa arquitetura de dados, deve atender requisitos transacionais e analíticos, contemplando soluções para aplicações com necessidade não relacionais, transacionais e também o ambiente de análise e ciência de dados.

- 
- +
- □ •
- •
- 
- 



# FINTECH QUANTUMFINANCE

## Arquitetura de dados



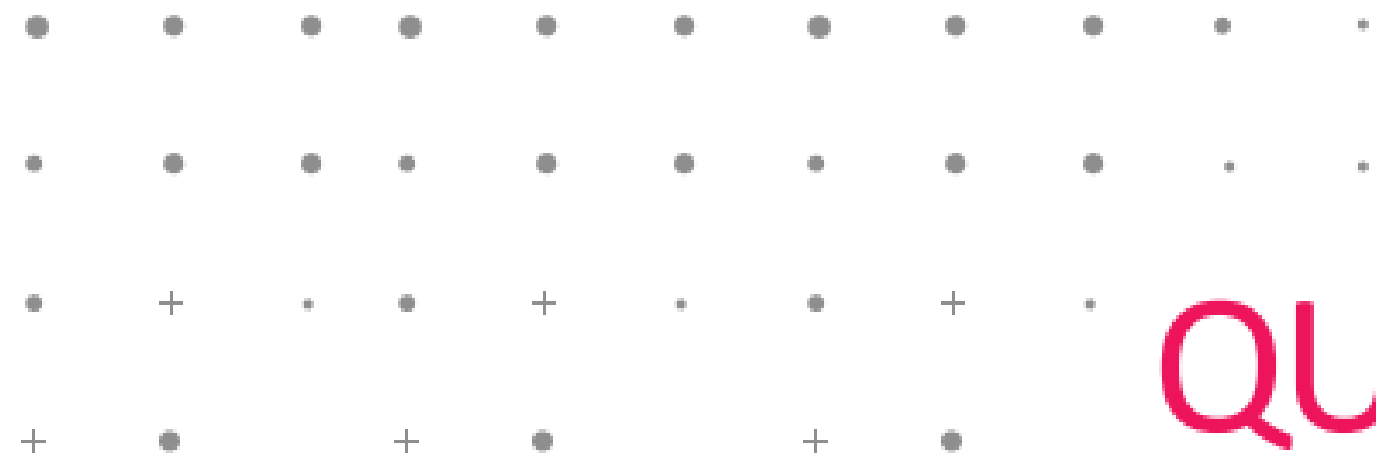


**Cadastro de Cliente com Fotos e Assinatura Eletrônica, e Informações dos Contatos do Cliente:** integração entre modelos relacional e não relacional para gerenciar diferentes tipos de dados no cadastro de clientes com fotos, assinatura eletrônica e informação de contato para maior integridade referencial dos bancos para dados estruturados e flexibilidade para dados não estruturados.

**Processo de Transações (Doc, TED, Pix):** utilização do APACHE NiFi para a coleta de arquivos (streaming) para gerenciar o fluxo de dados de forma eficiente, aplicando processadores como 'GetFile' para ingestão de arquivos e 'EvaluateJsonPath' para extrair dados .JSON e ingestão de arquivos MySQL para alta performance e alta escalabilidade para lidar com grande volume de transações.

**Recomendação de Produtos de Crédito para a Home:** com o modelo dimensional (estrela), existe maior facilidade para consultas analíticas e para cruzar informações de diversas origens (como Serasa, outros bancos, demográficos etc.) a partir do Amazon Personalize que treina e gera modelos de recomendações usando os dados para usuários em tempo real a partir de interações dos clientes (histórico de transações, cliques em produtos etc).

**Telas e Relatórios para Análise das Transações da Conta:** com o modelo dimensional (flocos de neve) a fim de diminuir a redundância, utiliza-se o data warehouse, RedShift, além de ser totalmente gerenciado e otimizado para análises de grandes volumes de dados de transações bancárias com alta performance para consultas analíticas e de integração fácil com o QuickSight para geração de telas e relatórios. Em adição, pode ser viável também em situações devido a flexibilidade com dados semiestruturados (logs de transações, dados de APIs em JSON), o Athena com o S3 para consultas ad-hoc.



FINTECH

QUANTUMFINANCE

Construção de um pipeline usando o APACHE NiFi que realize a ingestão e transformação de dados de um dos fluxos de dados modelados na disciplina de Data Modeling

### Fluxo escolhido

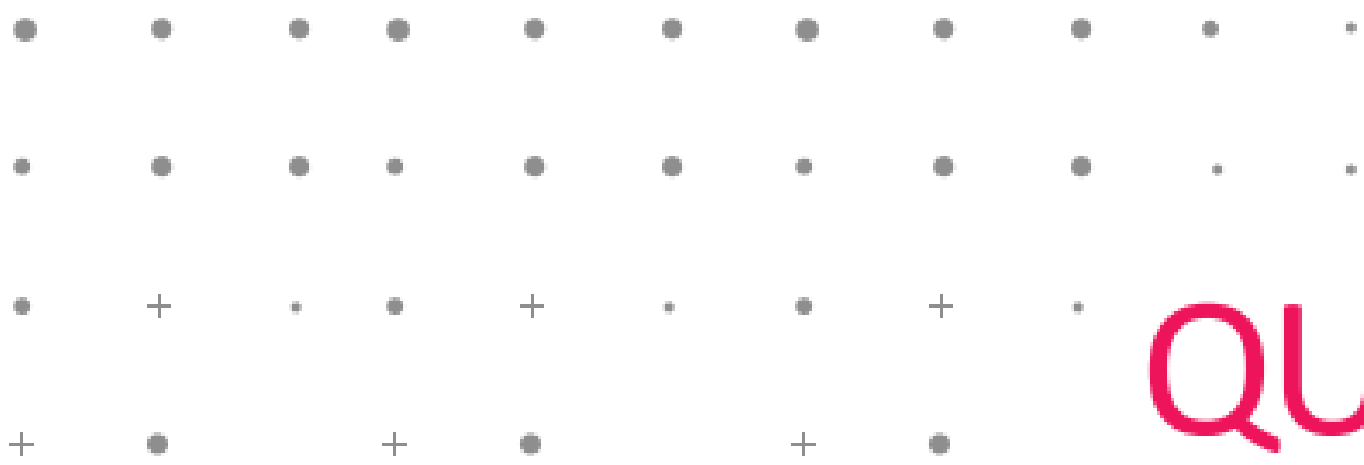
Processo de Transações

### Objetivo

Para maior rapidez no recebimento de transações, vamos criar um fluxo no NIFI que leia um arquivo .JSON e armazene os valores na tabela transação.

Essa ação de acordo com a coluna ID\_TIPO\_TRANSACAO poderá receber transações diversas tais como:

1. De acordo com o tipo de operação, registrar TED e Pix.
2. Agilidade na inserção de registros no banco de dados de operações de compra e venda de ações na bolsa de valores.

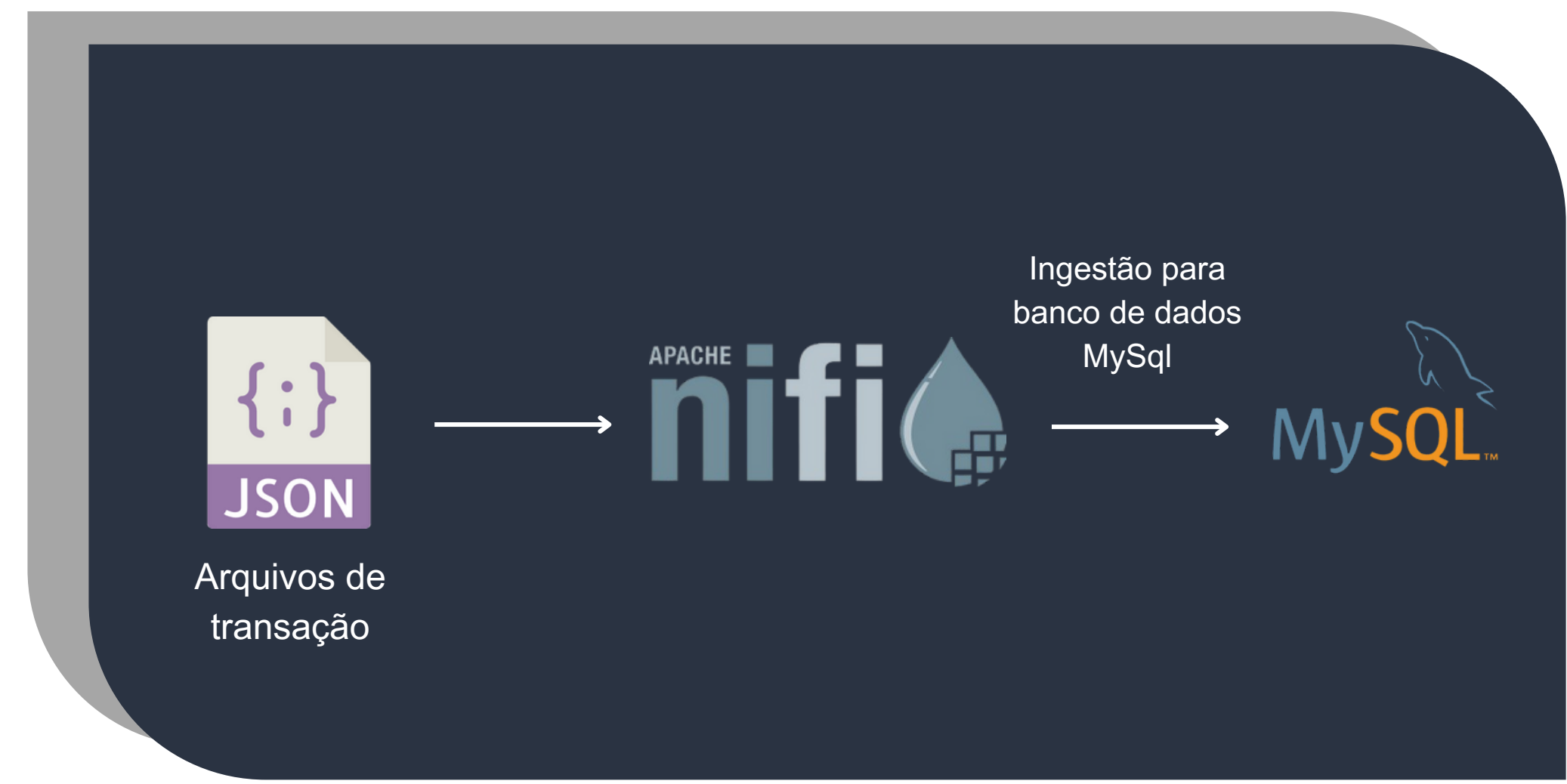


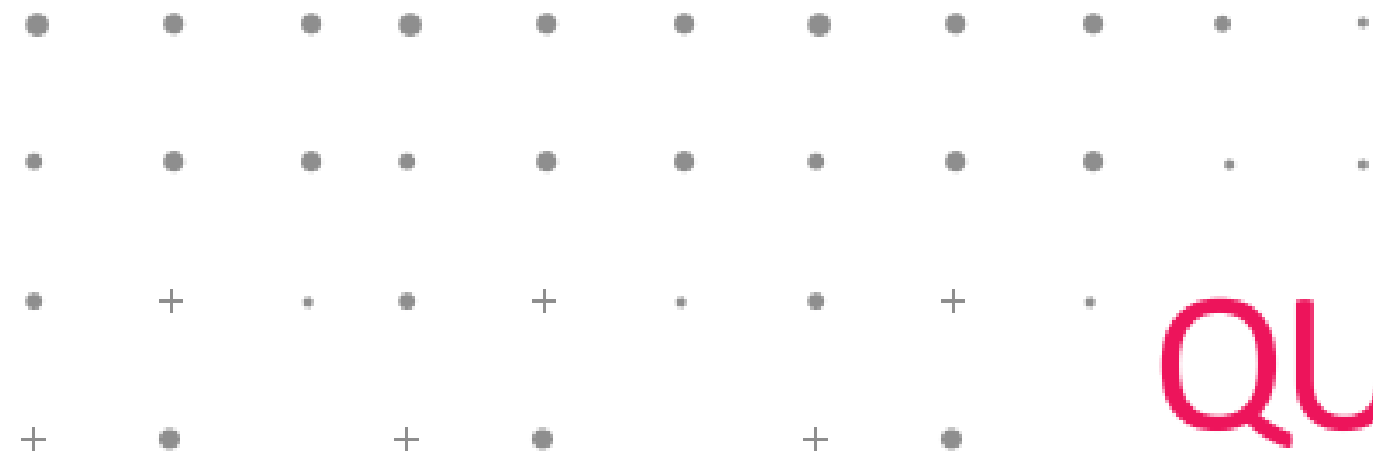
# FINTECH<sup>+</sup>

## QUANTUMFINANCE

Pipeline

Processo de transações (Doc, TED, Pix)





FINTECH

# QUANTUMFINANCE

## Etapas

1. Criação do banco

2. Criação da tabela:

```
CREATE TABLE TB_TRANSACAO (  
  ID_TRANSACAO INT PRIMARY KEY,  
  ID_CLIENTE INT,  
  ID_TIPO_TRANSACAO INT,  
  ID_PRODUTO INT,  
  ID_BANCO INT,  
  DATA_TRANSACAO VARCHAR(15),  
  VALOR_TRANSACAO DECIMAL(10, 2));
```

3. Inserção do JSON na pasta:

C:\Quantum

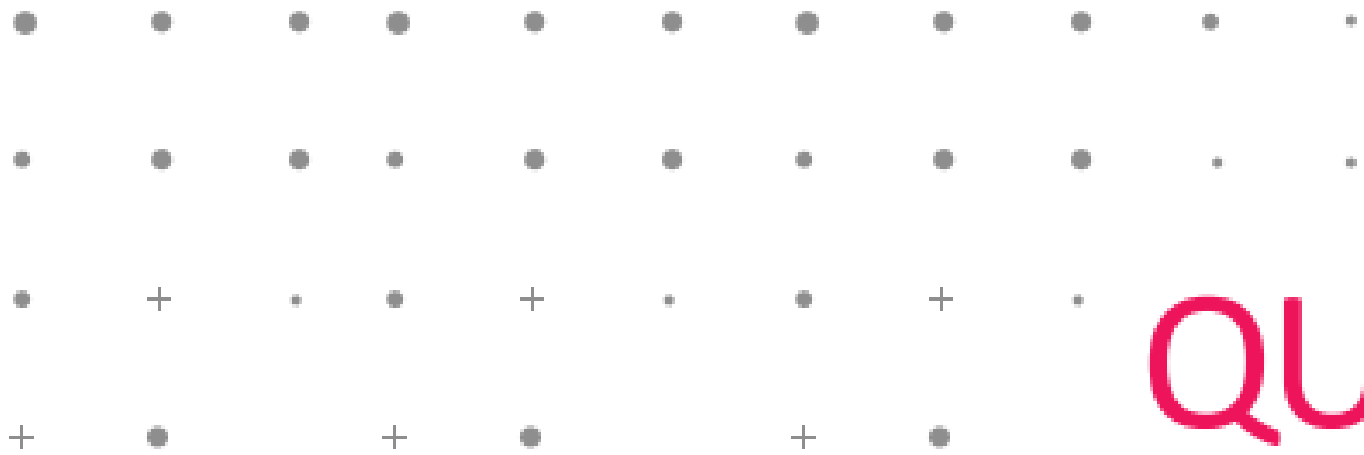
Exemplo JSON:

```
{  
  "ID_TRANSACAO": 90020,  
  "ID_CLIENTE": 3671,  
  "ID_TIPO_TRANSACAO": 5,  
  "ID_PRODUTO": 222,  
  "ID_BANCO": 43,  
  "ID_DATA": "2024-05-01",  
  "VALOR_TRANSACAO": "75.15"  
}
```

Obs.: a coluna ID\_TRANSACAO é chave primária. Para testes, alterar o número no arquivo

4. Rodar comando para mover para TEMP

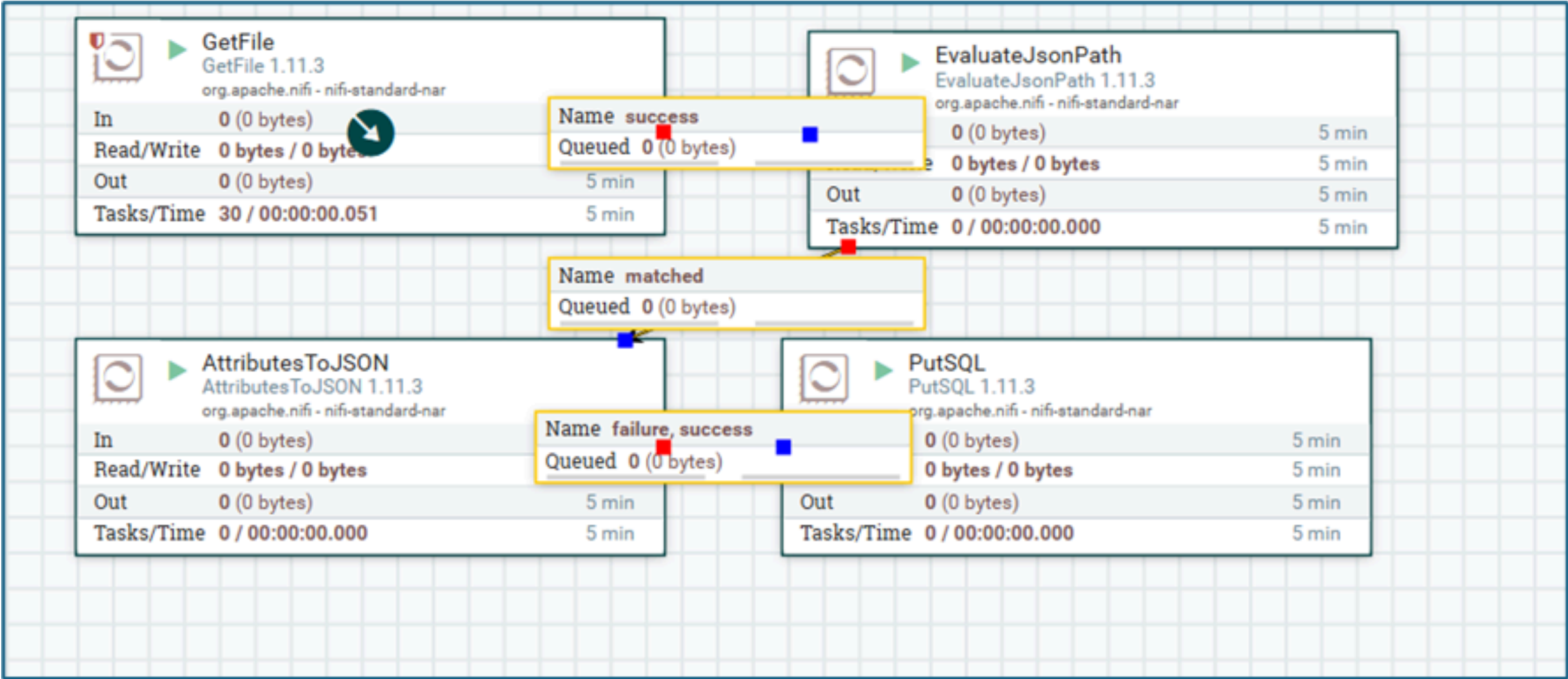
docker container cp "C:\Quantum\transacao.json" nifi:/tmp/nifi



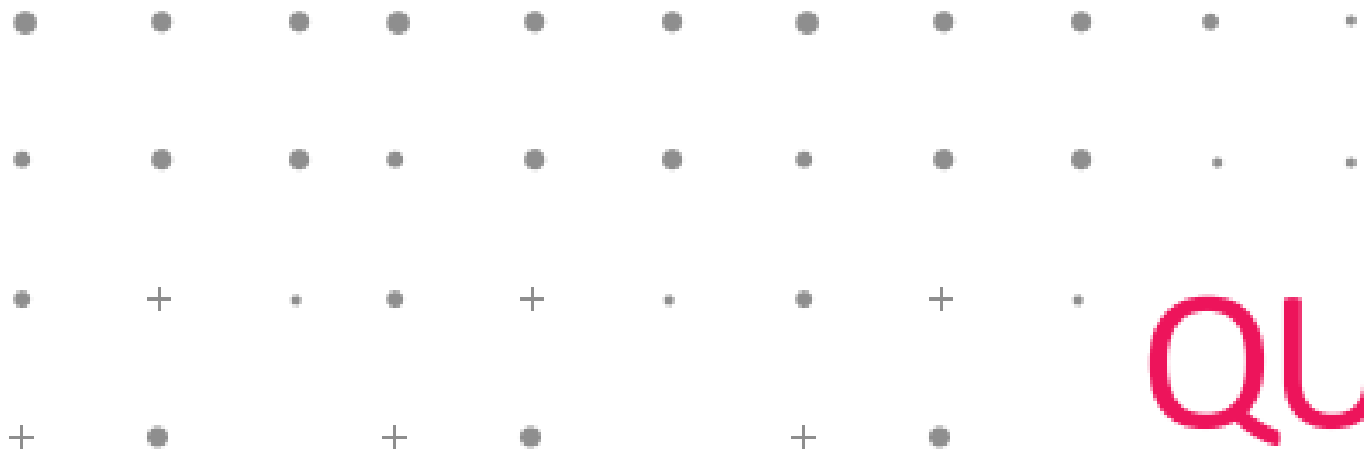
# FINTECH

# QUANTUMFINANCE

## 5. Executar o fluxo NIFI







# FINTECH

## QUANTUMFINANCE

### 6. Confirmando a execução

```
mysql> SELECT * FROM TB_TRANSACAO;
```

ID_TRANSACAO	ID_CLIENTE	ID_TIPO_TRANSACAO	ID_PRODUTO	ID_BANCO	DATA_TRANSACAO	VALOR_TRANSACAO
90020	3671	5	222	43		75.15

```
1 row in set (0.02 sec)
```

```
mysql> mysql> SELECT * FROM TB_TRANSACAO;
```

ID_TRANSACAO	ID_CLIENTE	ID_TIPO_TRANSACAO	ID_PRODUTO	ID_BANCO	DATA_TRANSACAO	VALOR_TRANSACAO
90020	3671	5	222	43		75.15
90021	3678	6	223	4		26.30

```
2 rows in set (0.00 sec)
```