

INOVAR PE UPE

HighByte

# Antes de começarmos...

<https://guide.highbyte.com/>

## HighByte User Guide

Welcome to the HighByte Intelligence Hub User Guide.

### Setup

- [System Requirements](#)
- [Installation](#)
  - [Installing Java](#)
  - [Licensing](#)
  - [Application State](#)
- [Upgrade](#)
  - [Upgrading from 3.4 to 4.0](#)
- [Docker](#)
- [Redundancy](#)
- [Application Settings](#)

### Troubleshooting

- [Event Log](#)
- [Status & Metrics](#)
- [Debug Logging](#)
- [Debugging Pipelines](#)

### Configuration

- [Connect](#)
  - [Inputs](#)
    - [Dynamic References](#)
  - [Outputs](#)
    - [Dynamic Outputs](#)
    - [Output Templates](#)
  - [Connections](#)
    - [Amazon Redshift](#)
    - [Amazon S3](#)
    - [AWS IoT SiteWise](#)
    - [AWS Kinesis Data Firehose](#)
    - [AWS Kinesis Data Streams](#)
    - [Azure Blob Storage](#)
    - [Azure Event Hubs](#)
    - [Azure IoT Edge](#)
    - [Azure IoT Hub](#)
    - [CSV](#)
    - [File](#)
    - [Google BigQuery](#)
    - [Google PubSub](#)
    - [InfluxDB](#)
    - [JDBC Driver](#)
    - [Kafka](#)

# Índice

- Introdução ao Highbyte
- Arquiteturas do Highbyte
- Major Release – Versão 4.0
- Conexões e Coletas de Tags
- Modelagem de Dados
- Flows
- Pipelines
- Conditions
- Exemplo prático

# Índice

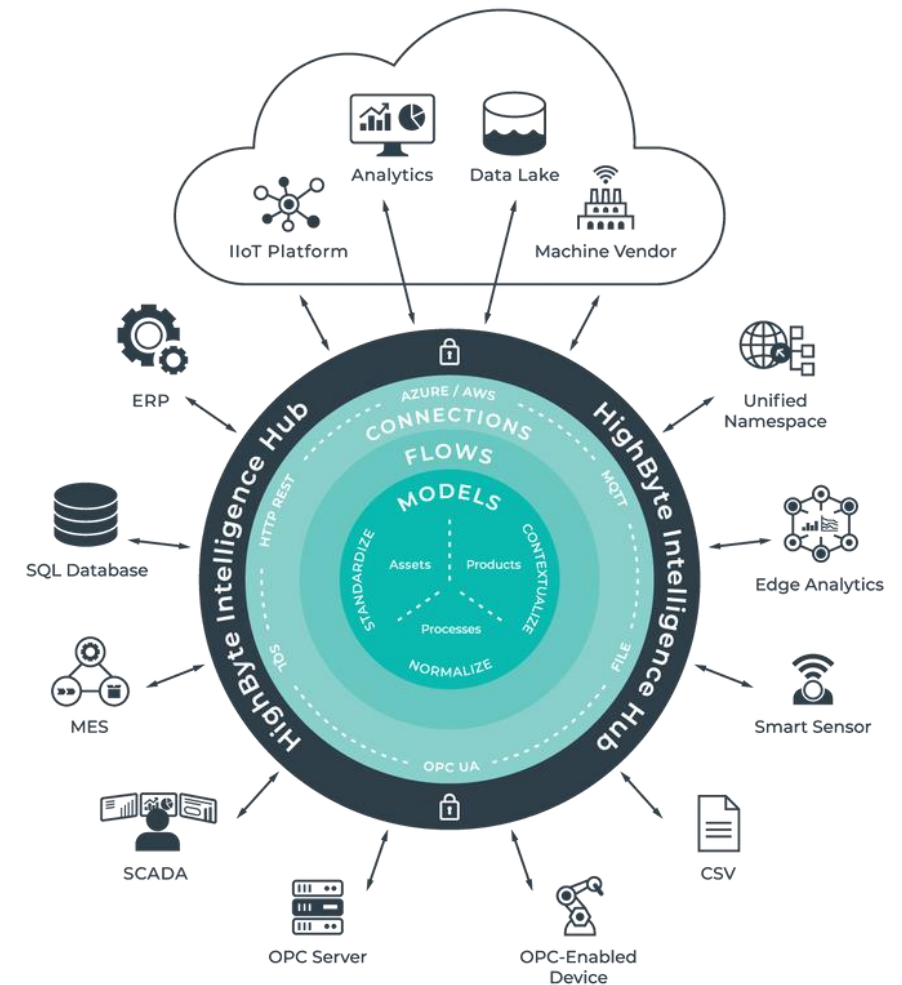
- Introdução ao Highbyte
- Arquiteturas do Highbyte
- Major Release – Versão 4.0
- Conexões e Coletas de Tags
- Modelagem de Dados
- Flows
- Pipelines
- Conditions
- Exemplo prático

# Introdução ao Highbyte

- Ferramenta de Integração de Dados Industriais
- Transformar, modelar e enviar dados
- IIoT/Indústria 4.0

## Diferenciais:

- Conectividade com diferentes sistemas
- Modelagem de dados
- Pipelines de processamentos
- Versatilidade de Arquitetura

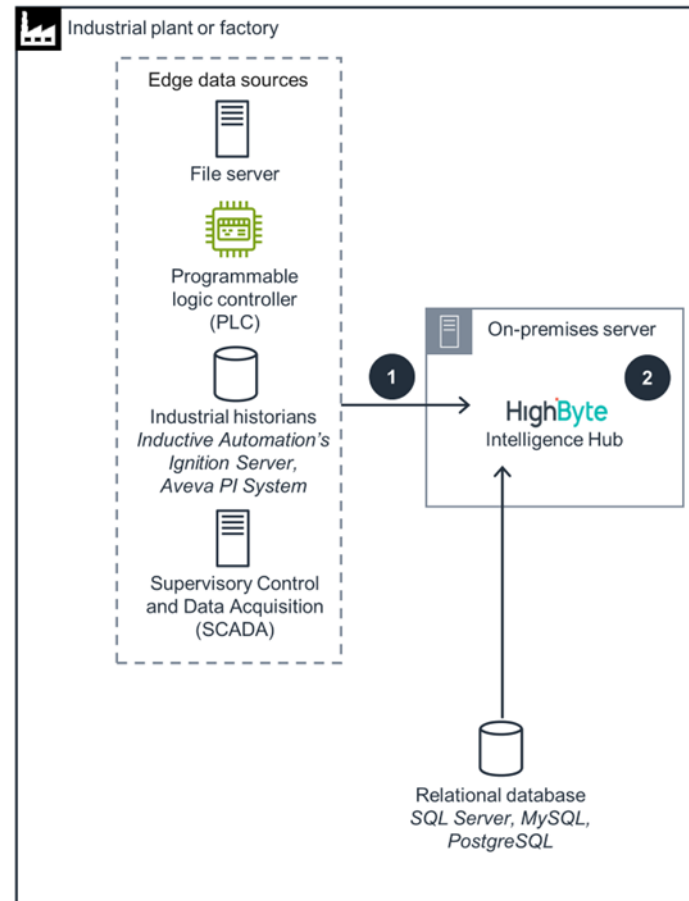


# Índice

- Introdução ao Highbyte
- **Arquiteturas do Highbyte**
- Major Release – Versão 4.0
- Conexões e Coletas de Tags
- Modelagem de Dados
- Flows
- Pipelines
- Conditions
- Exemplo prático

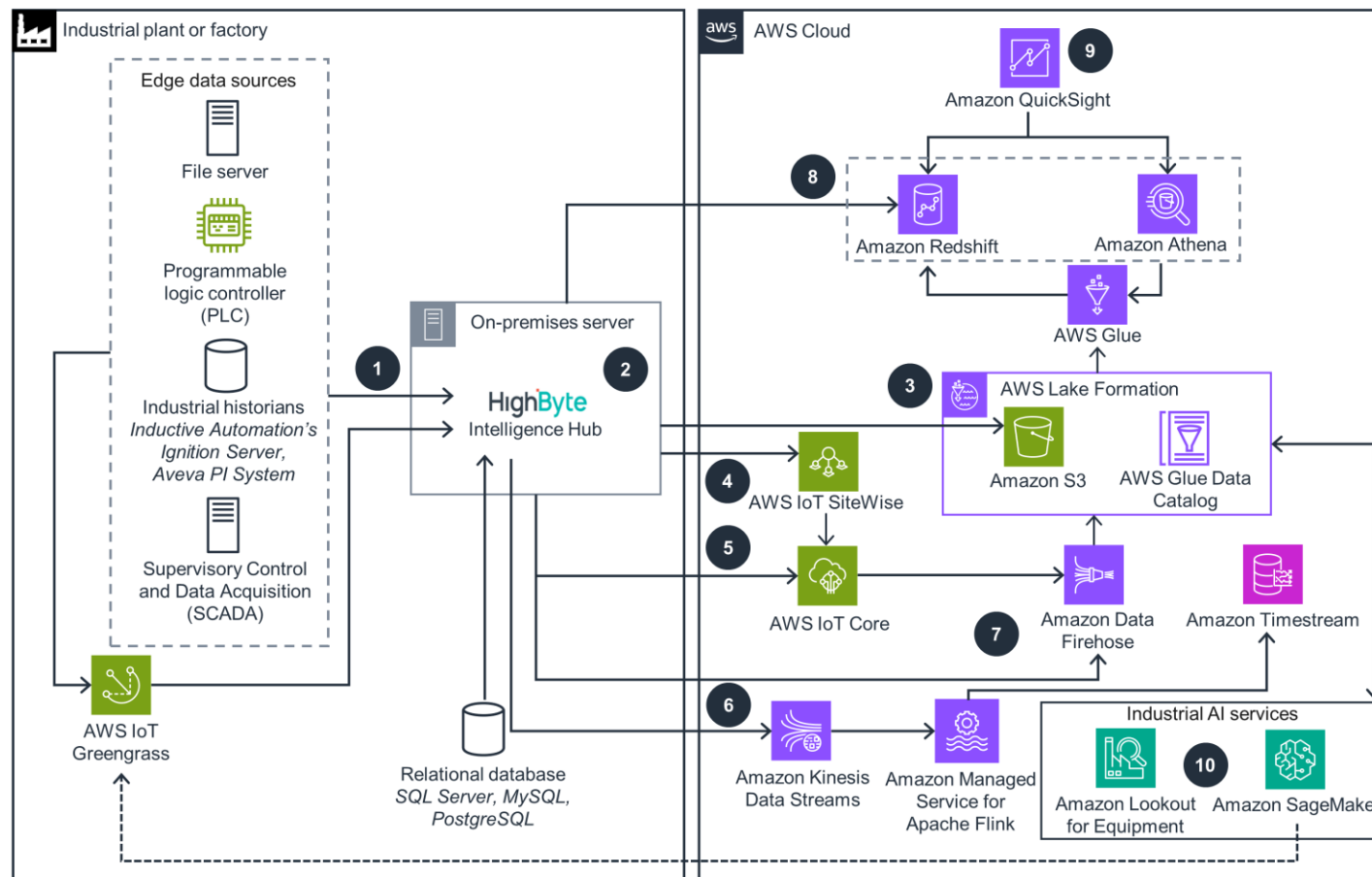
# Arquiteturas do Highbyte

- Arquitetura 100% on-premise



# Arquiteturas do Highbyte

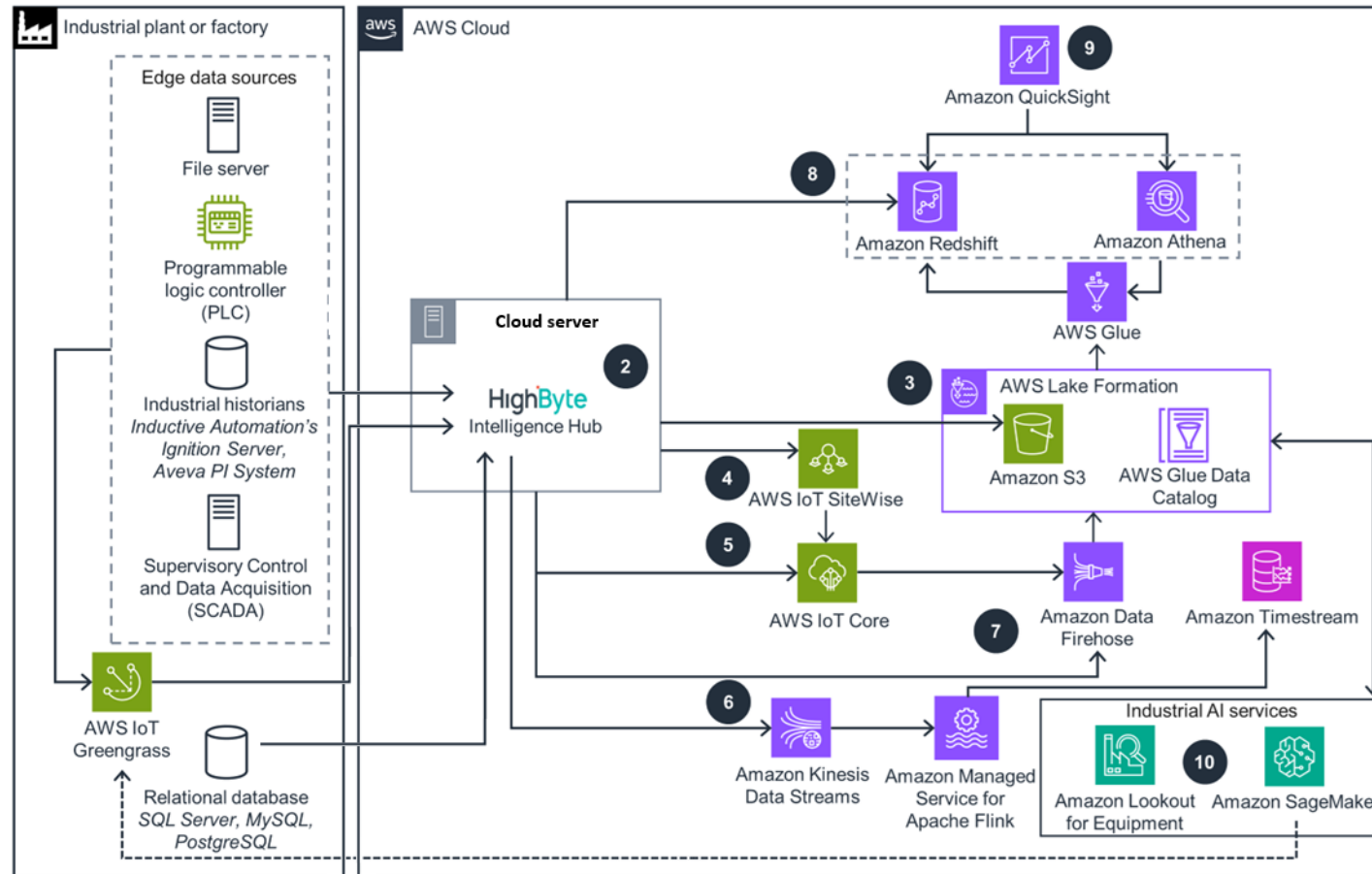
- Servidor on-premise com integração em nuvem





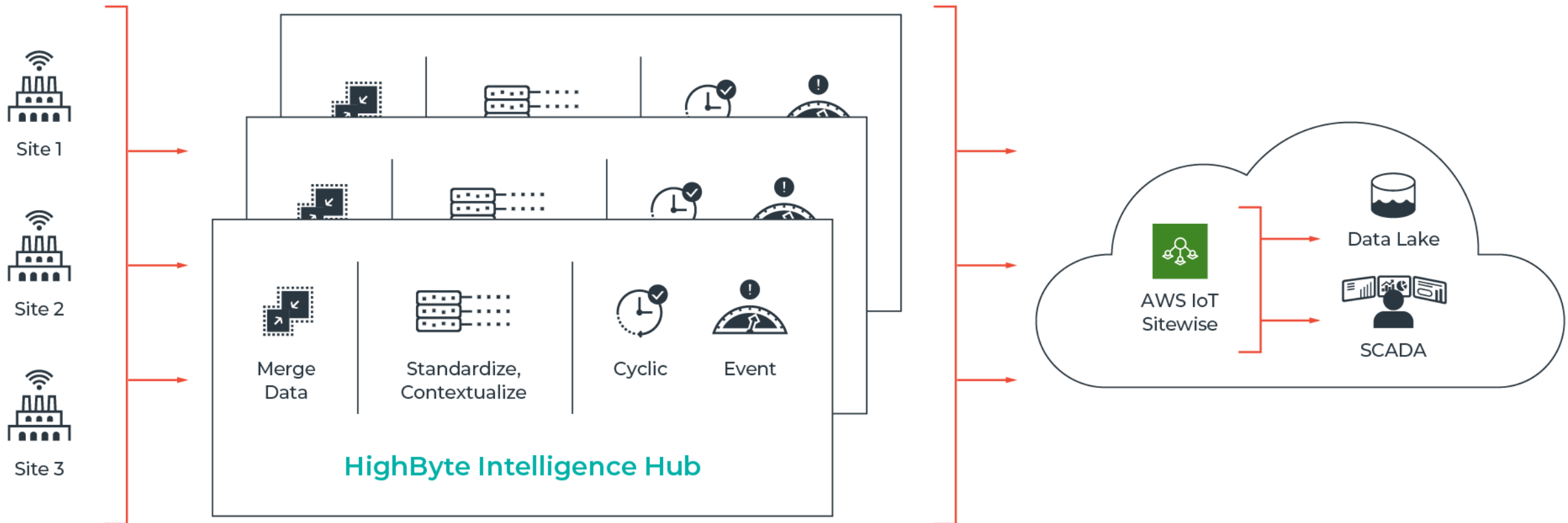
# Arquiteturas do Highbyte

- Servidor em nuvem (com devido firewall)



# Arquiteturas do Highbyte

- Múltiplos sites (um Highbyte por site)

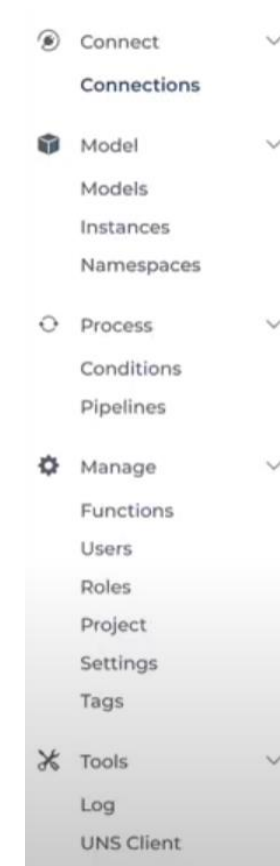
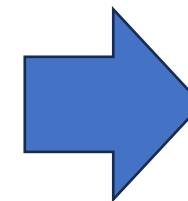
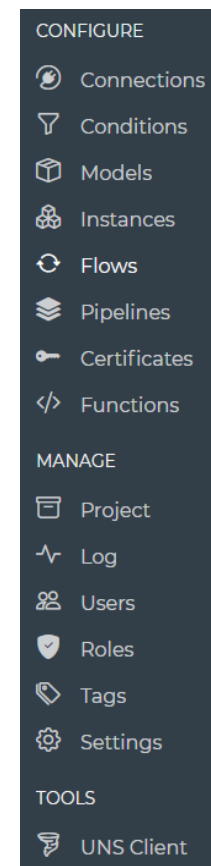
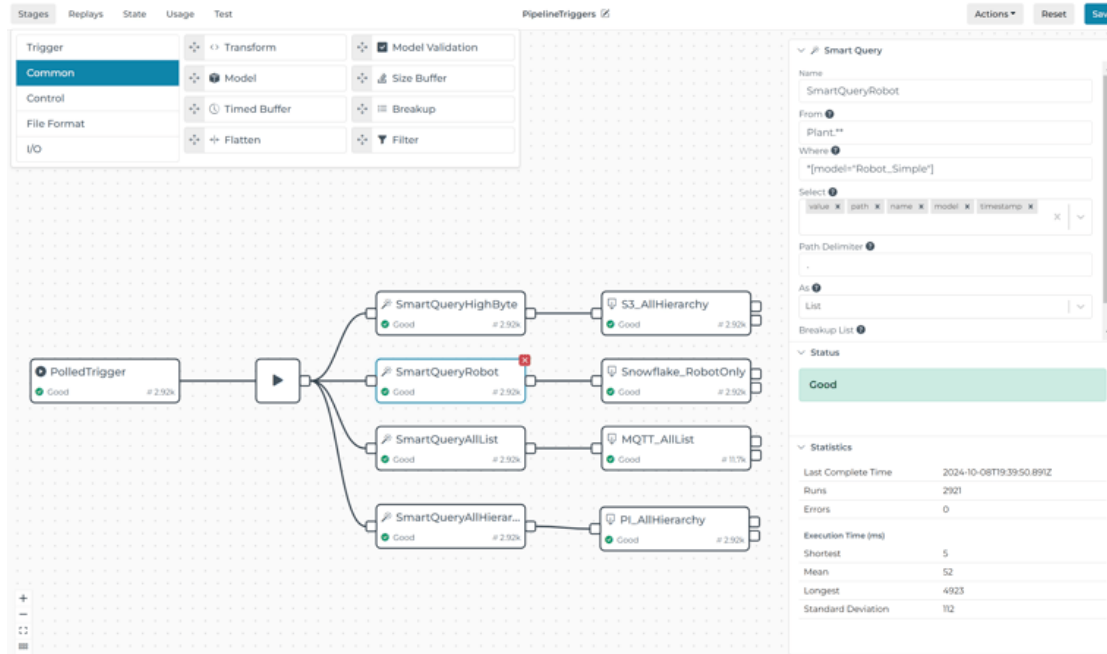


# Índice

- Introdução ao Highbyte
- Arquiteturas do Highbyte
- **Major Release – Versão 4.0**
- Conexões e Coletas de Tags
- Modelagem de Dados
- Flows
- Pipelines
- Conditions
- Exemplo prático

# Major Release – v4.0 (09/Out)

- Remoção dos *flows* - “*pipeline-centric approach*”
- Mudanças em UI/UX
- Bug fixes (novos bugs?)



# Índice

- Introdução ao Highbyte
- Arquiteturas do Highbyte
- Major Release – Versão 4.0
- **Conexões e Coletas de Tags**
- Modelagem de Dados
- Flows
- Pipelines
- Conditions
- Exemplo prático

# Conexões

- Tanto para **entrada** (input) de dados quanto para **saída** (output)
- Data sources:
  - Cloud (AWS, Azure)
  - Arquivos (CSV, Parquet, etc)
  - Big Data (Google, Snowflake, Kafka)
  - HTTP REST/Webhook
  - Industrial Network (MQTT, Sparkplug, Modbus TCP, OPC UA)
  - SQL (MSSQL, MySQL, PGSQL, InfluxDB, Oracle, SQLite, ...)
  - PI-System



# Conexões

- Tanto para **entrada** (input) de dados quanto para **saída** (output)

- Data sources:

- **Cloud (AWS, Azure)**

- Arquivos (CSV, Parquet, etc)

- Big Data (Google, Snowflake, ...)

- HTTP REST/Webhook

- Industrial Network (MQTT, S...

- SQL (MSSQL, MySQL, PGSQL, InfluxDB, Oracle, SQLite, ...)

- PI-System

The screenshot shows a web interface with three tabs: '1 Details', '2 Protocol', and '3 Settings'. The 'Details' tab is active. It contains the following fields and controls:

- Access Key**: A text input field.
- Secret Key**: A text input field with a toggle icon (key symbol) on the right.
- Region**: A text input field.
- Enable Store & Forward**: A toggle switch currently set to 'Off'.

At the bottom of the form, there are three buttons: 'Previous' (disabled), 'Submit' (highlighted in red), and 'Cancel'.

# Conexões

- Tanto para **entrada** (input) de dados quanto para **saída** (output)

- Data sources:

- Cloud (AWS, Azure)
- **Arquivos (CSV, Parquet, etc)**
- Big Data (Google, Snowflake, K)
- HTTP REST/Webhook
- Industrial Network (MQTT, Spa)
- SQL (MSSQL, MySQL, PGSQL, I)
- PI-System

1 Details

2 Protocol

3 Settings

File Directory

Processed File Directory

Error Directory

Flatten Modeled Values

Off

Enable Store & Forward

Off

Previous

Submit

Cancel



# Conexões

- Tanto para **entrada** (input) de dados quanto para **saída** (output)

- Data sources:

- Cloud (AWS, Azure)
- Arquivos (CSV, Parquet, etc)
- **Big Data (Google, Snowflake, Kafka)**
- HTTP REST/Webhook
- Industrial Network (MQTT, Sparkplug,
- SQL (MSSQL, MySQL, PGSQL, InfluxDB,
- PI-System

The screenshot shows a web interface with three tabs: '1 Details', '2 Protocol', and '3 Settings'. The '1 Details' tab is active. It contains the following fields:

- Account Identifier:
- User:
- Role:
- Database:
- Schema:
- Warehouse:

At the bottom of the form, there are three buttons: 'Previous' (disabled), 'Submit' (highlighted in red), and 'Cancel' (disabled).

# Conexões

- Tanto para **entrada** (input) de dados quanto para **saída** (output)

- Data sources:

- Cloud (AWS, Azure)
- Arquivos (CSV, Parquet, e)
- Big Data (Google, Snowfla
- **HTTP REST/Webhook**
- Industrial Network (MQT
- SQL (MSSQL, MySQL, PGS
- PI-System

1 Details

2 Protocol

3 Settings

Base URL

Header

Name	Value
No parameters defined	

+ Add Value

Authentication Type

None

Ignore Self-Signed Certificate

Off

Use Proxy

Off

Flatten Modeled Values

Off

Enable Store & Forward

Off

Previous

Submit

Cancel

# Conexões

- Tanto para **entrada** (input) de dados quanto para **saída** (output)
- Data sources:
  - Cloud (AWS, Azure)
  - Arquivos (CSV, Parquet, etc)
  - Big Data (Google, Snowflake, Kafka)
  - HTTP REST/Webhook
  - **Industrial Network (MQTT, Sparkplug, Modbus TCP, OPC UA)**
  - SQL (MSSQL, MySQL, PGSQL, InfluxDB, Oracle, S
  - PI-System

The screenshot shows a configuration interface with three tabs: '1 Details', '2 Protocol', and '3 Settings'. The 'Details' tab is active, displaying various configuration fields:

- Client Id**: Text input field.
- Username**: Text input field.
- Password**: Text input field with a toggle for visibility.
- Connection Timeout (seconds)**: Text input field with value '10'.
- Keep Alive (seconds)**: Text input field with value '60'.
- Request Timeout (ms)**: Text input field with value '5000'.
- Redundant Brokers**: Text input field with value 'Add new broker' and an 'Add' button.
- Use SSL?**: Toggle switch set to 'Off'.
- Input Discovery Topic**: Text input field with a help icon.
- Clean Session**: Toggle switch set to 'On'.
- Flatten Modeled Values**: Toggle switch set to 'Off'.
- Enable Store & Forward**: Toggle switch set to 'Off'.

At the bottom of the 'Details' tab, there are 'Previous' and 'Submit' buttons. A 'Cancel' button is located at the bottom right of the 'Settings' tab.

# Conexões

- Tanto para **entrada** (input) de dados quanto para **saída** (output)

- Data sources:

- Cloud (AWS, Azure)
- Arquivos (CSV, Parquet, e
- Big Data (Google, Snowfl
- HTTP REST/Webhook
- Industrial Network (MQT
- **SQL (MSSQL, MySQL, PGSQL, InfluxDB, Oracle, SQLite, ...)**
- PI-System

The screenshot shows a web-based configuration interface for setting up a data source connection. It features three tabs at the top: '1 Details', '2 Protocol', and '3 Settings'. The 'Details' tab is currently selected and contains the following elements:

- Database:** A text input field.
- Username:** A text input field.
- Password:** A text input field with a toggle icon (key symbol) on the right.
- JDBC Connection String:** A text input field.
- Enable Store & Forward:** A toggle switch currently set to 'Off'.
- Navigation Buttons:** 'Previous' (disabled), 'Submit' (active/red), and 'Cancel' (disabled).

# Conexões

- Tanto para **entrada** (input) de dados quanto para **saída** (output)

- Data sources:

- Cloud (AWS, Azure, Google Cloud)
- Arquivos (CSV, Parquet, JSON)
- Big Data (Google BigQuery, Amazon Redshift)
- HTTP REST/Webhooks
- Industrial Networks (OPC UA, Modbus)
- SQL (MSSQL, MySQL, PostgreSQL)

- **PI-System**

1 Details

2 Protocol

3 Settings

Token

Connect Timeout (seconds)

5

Request Timeout (ms)

5000

Compression

None

PI Agent Authentication ?

Windows

Flatten Modeled Values

Off

Enable Store & Forward

Off

Previous

Submit

Cancel

# Conexões

- Após configurada a conexão, **inputs** e **outputs** podem ser configurados

- Ex: Input (Query)

Connection TimescaleDB Vivix Vidros Planos - Professional Help

Overview **Inputs** Outputs Statistics Usage

New Input Cancel Create

Name

Type

Query

Index

Enable Index ☐ Off

Cache

Enable Cache ☐ Off

Templating

Use Template? ☐ Off

Results Test Input

```
[
  {
    "machine_id": 1,
    "dt_start": "2024-10-16T17:00:00.000Z",
    "dt_end": "2024-10-16T17:30:00.000Z",
    "description": "Limpeza de bico"
  },
  {
    "machine_id": 1,
    "dt_start": "2024-10-15T11:00:00.000Z",
    "dt_end": "2024-10-15T11:45:00.000Z",
    "description": "Troca de filtro"
  },
  {
    "machine_id": 2,
    "dt_start": "2024-10-15T12:15:00.000Z",
    "dt_end": "2024-10-15T12:35:00.000Z",
    "description": "Ajuste de pressão"
  },
  {
    "machine_id": 3
  }
]
```

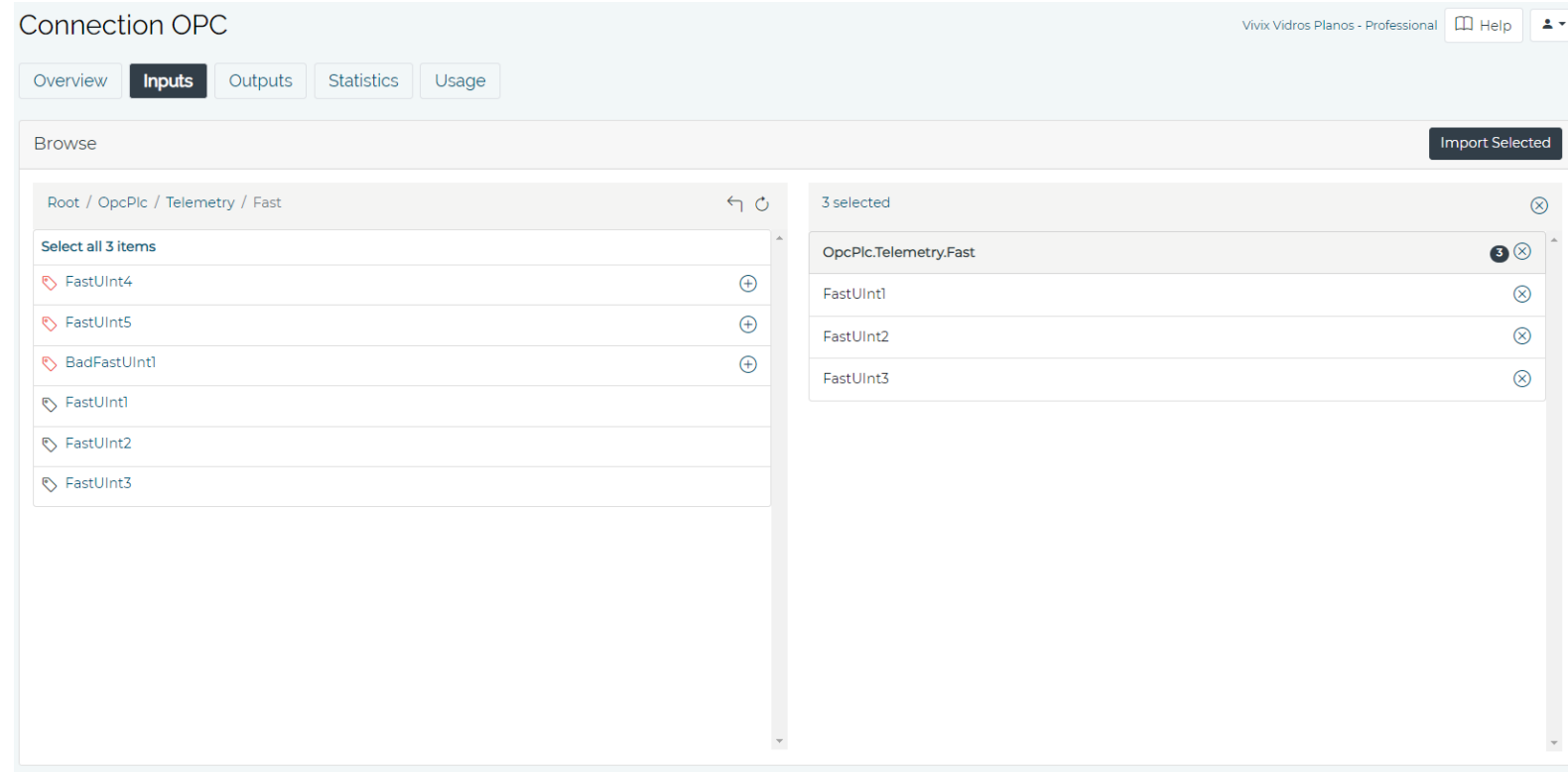
> Show Stats Open

Object Explorer

# Conexões

- Após configurada a conexão, **inputs** e **outputs** podem ser configurados

- Ex: Input (OPC)
- Possível utilizar Browse



# Conexões

- Após configurada a conexão, **inputs** e **outputs** podem ser configurados

- Ex: Output (Query)

Connection TimescaleDB Vivix Vidros Planos - Professional Help

Overview Inputs **Outputs** Statistics Usage

New Output Cancel Create

Name

Table

Write Type  ▼

Log as JSON ☐ Off

Create Table  ▼

Transforms

Breakup Arrays ? ☐ Off

Attribute Filter ?  Add



# Índice

- Introdução ao Highbyte
- Arquiteturas do Highbyte
- Major Release – Versão 4.0
- Conexões e Coletas de Tags
- **Modelagem de Dados**
- Flows
- Pipelines
- Conditions
- Exemplo prático

# Modelagem de Dados

- No Highbyte, a modelagem está dividida em **Modelos** e **Instâncias**
  - Modelo**: define a estrutura do dado
  - Instância**: ocorrência real do modelo
  - Exemplo**: um modelo SENSOR\_TEMPERATURA pode ter múltiplas instâncias (TT-101, TT-102, TT-103, ...), mas cada a instância é atribuída um único modelo (SENSOR\_TEMPERATURA)

New Model

Vivix Vidros Planos - Professional Help

1 Details 2 Attributes

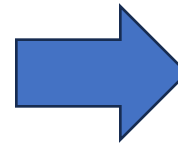
Name: VariavelProcesso

Description: Modelo generico de uma variavel de processo

Tags: Select...

Group As:

Next Cancel



New Model

Vivix Vidros Planos - Professional Help

1 Details 2 Attributes

Name	Type	Array	Required
Timestamp	DateTime	Off	Off
Valor	Real64	Off	Off
Setpoint	Real64	Off	Off
USL	Real64	Off	Off
LSL	Real64	Off	Off

New Attribute

Previous Submit Cancel

# Modelagem de Dados

- No Highbyte, a modelagem está dividida em **Modelos** e **Instâncias**
  - Modelo:** define a estrutura do dado
  - Instância:** ocorrência real do modelo
  - Exemplo:** um modelo SENSOR\_TEMPERATURA pode ter múltiplas instâncias (TT-101, TT-102, TT-103, ...), mas cada a instância é atribuída um único modelo (SENSOR\_TEMPERATURA)

New Instance

1 Details 2 Model 3 Attributes

Name: SensorTemperatura

Description:

Tags: Select...

Group As:

Expand Arrays: ☐ ON

Next Cancel

1 Details 2 Model 3 Attributes

Model: VariavelProcesso

Previous Next Cancel

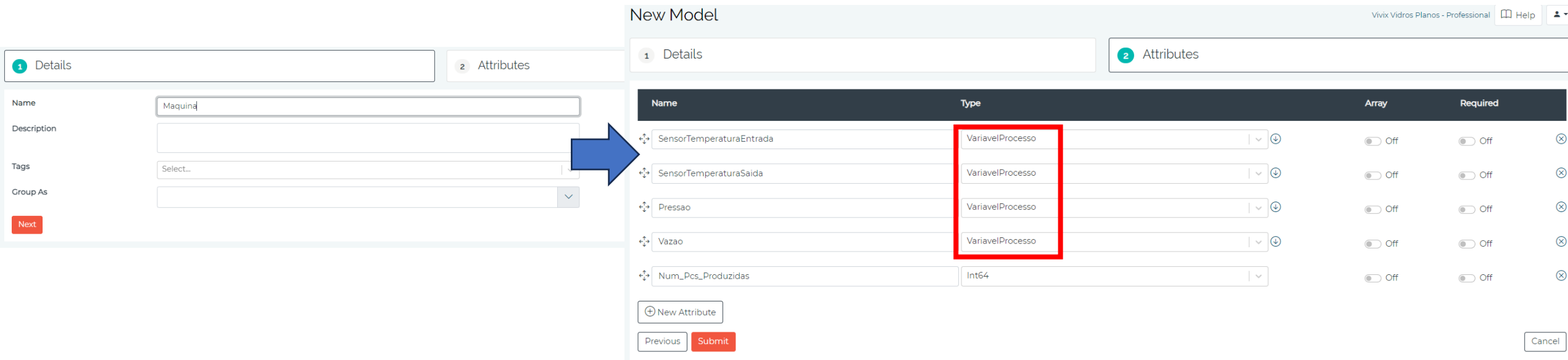
1 Details 2 Model 3 Attributes

Name	Expression	Default
Timestamp DateTime		
Valor Real64		
Setpoint Real64		
USL Real64		
LSL Real64		

Previous Submit Cancel

# Modelagem de Dados

- Cada modelo possui atributos, **que podem ser outros modelos**
  - OBS: evite aninhar muitos modelos, não é performático



**New Model** Vivix Vidros Planos - Professional

1 Details 2 Attributes

Name: Maquina

Description:

Tags: Select...

Group As:

Next

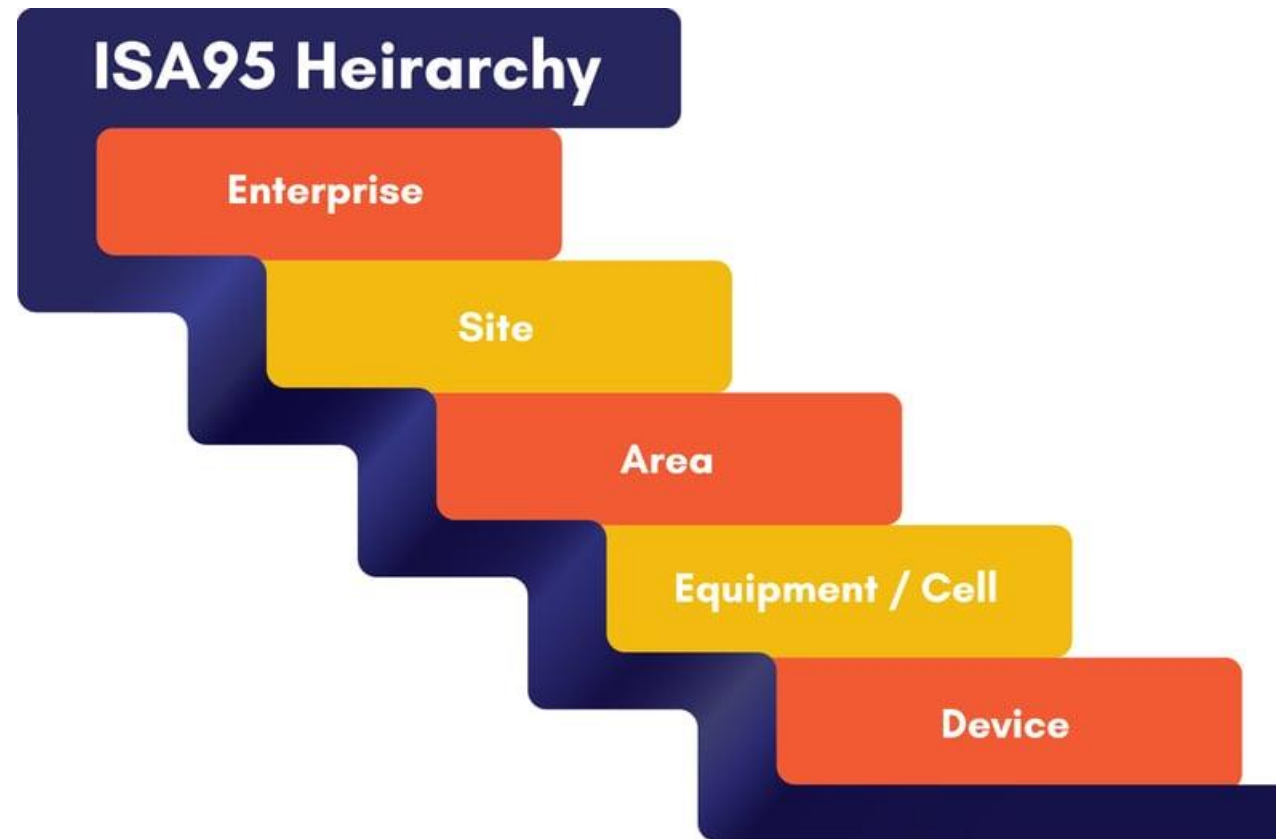
Name	Type	Array	Required
SensorTemperaturaEntrada	VariavelProcesso	Off	Off
SensorTemperaturaSaida	VariavelProcesso	Off	Off
Pressao	VariavelProcesso	Off	Off
Vazao	VariavelProcesso	Off	Off
Num_Pcs_Produzidas	Int64	Off	Off

+ New Attribute

Previous Submit Cancel

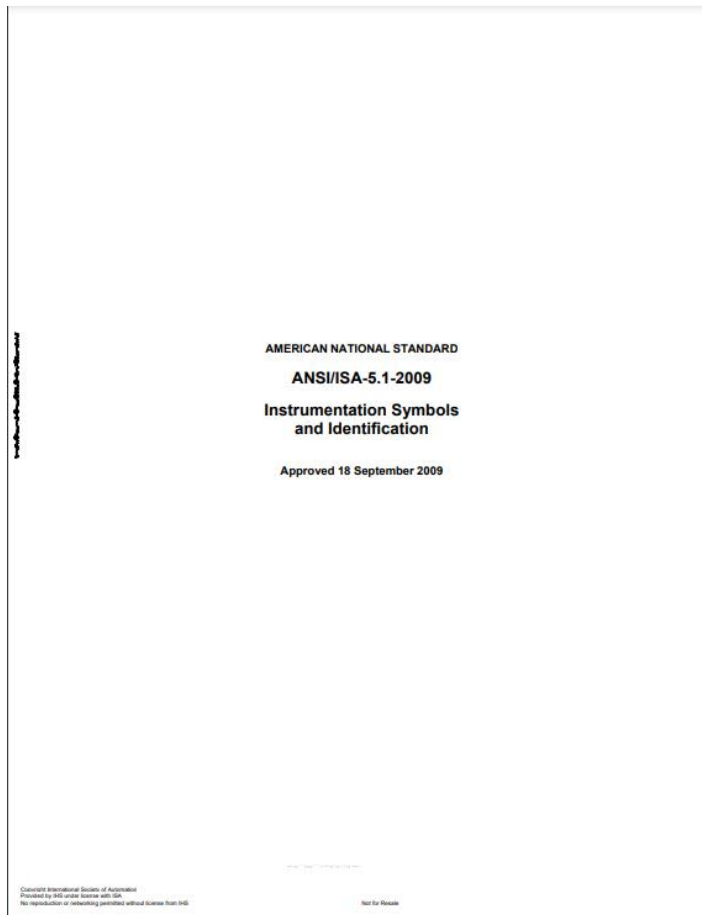
# Modelagem de Dados – Tópicos importantes

- UNS – Unified Namespace



# Modelagem de Dados – Tópicos importantes

- ISA-5.1-2009 – Instrumentation Symbols and Identifications



**Table 4.1 — Identification letters**

Note: Numbers in parentheses refer to the preceding explanatory notes in Clause 4.2.

	First letters (1)		Succeeding letters (15)		
	Column 1	Column 2	Column 3	Column 4	Column 5
	Measured/Initiating Variable	Variable Modifier (10)	Readout/Passive Function	Output/Active Function	Function Modifier
<b>A</b>	Analysis (2)(3)(4)		Alarm		
<b>B</b>	Burner, Combustion (2)		User's Choice (5)	User's Choice (5)	User's Choice (5)
<b>C</b>	User's Choice (3a)(5)			Control (23a)(23e)	Close (27b)
<b>D</b>	User's Choice (3a)(5)	Difference, Differential, (11a)(12a)			Deviation (28)
<b>E</b>	Voltage (2)		Sensor, Primary Element		
<b>F</b>	Flow, Flow Rate (2)	Ratio (12b)			
<b>G</b>	User's Choice		Glass, Gauge, Viewing Device (16)		
<b>H</b>	Hand (2)				High (27a)( 28a)(29)
<b>I</b>	Current (2)		Indicate (17)		
<b>J</b>	Power (2)		Scan (18)		
<b>K</b>	Time, Schedule (2)	Time Rate of Change (12c)(13)		Control Station (24)	
<b>L</b>	Level (2)		Light (19)		Low (27b)( 28b)(29)
<b>M</b>	User's Choice (3a)(5)				Middle, Intermediate (27c)(28) (29)
<b>N</b>	User's Choice (5)		User's Choice (5)	User's Choice (5)	User's Choice (5)
<b>O</b>	User's Choice (5)		Orifice, Restriction		Open (27a)
<b>P</b>	Pressure (2)		Point (Test Connection)		
<b>Q</b>	Quantity (2)	Integrate, Totalize (11b)	Integrate, Totalize		
<b>R</b>	Radiation (2)		Record (20)		Run
<b>S</b>	Speed, Frequency (2)	Safety(14)		Switch (23c)	Stop
<b>T</b>	Temperature (2)			Transmit	
<b>U</b>	Multivariable (2)(6)		Multifunction (21)	Multifunction (21)	
<b>V</b>	Vibration, Mechanical Analysis (2)(4)(7)			Valve, Damper, Louver (23c)(23e)	
<b>W</b>	Weight, Force (2)		Weld, Probe		
<b>X</b>	Unclassified (8)	X-axis (11c)	Accessory Devices (22), Unclassified (8)	Unclassified (8)	Unclassified (8)
<b>Y</b>	Event, State, Presence (2)(9)	Y-axis (11c)		Auxiliary Devices (23d)( 25)( 26)	
<b>Z</b>	Position, Dimension (2)	Z-axis (11c), Safety Instrumented System (30)		Driver, Actuator, Unclassified final control element	

# Índice

- Introdução ao Highbyte
- Arquiteturas do Highbyte
- Major Release – Versão 4.0
- Conexões e Coletas de Tags
- Modelagem de Dados
- **Flows**
- Pipelines
- Exemplo prático

# Flows

- Forma simples de integração entre input e output
- Será “removida” a partir da versão 4.0

New Flow

Vivix Vidros Planos - Professional

Help

1 Details

2 Sources & Targets

3 Settings

Sources

{{Instance.Sensor1}}

Targets

{{Connection.TimescaleDB.Timeseries\_Flows}}

Previous

Next

Cancel

References

Type

Output

Connections

TimescaleDB

Filter

Filter

Timeseries\_Flows

Timeseries\_Pipeline

⏮

<

1 of 1

>

⏭

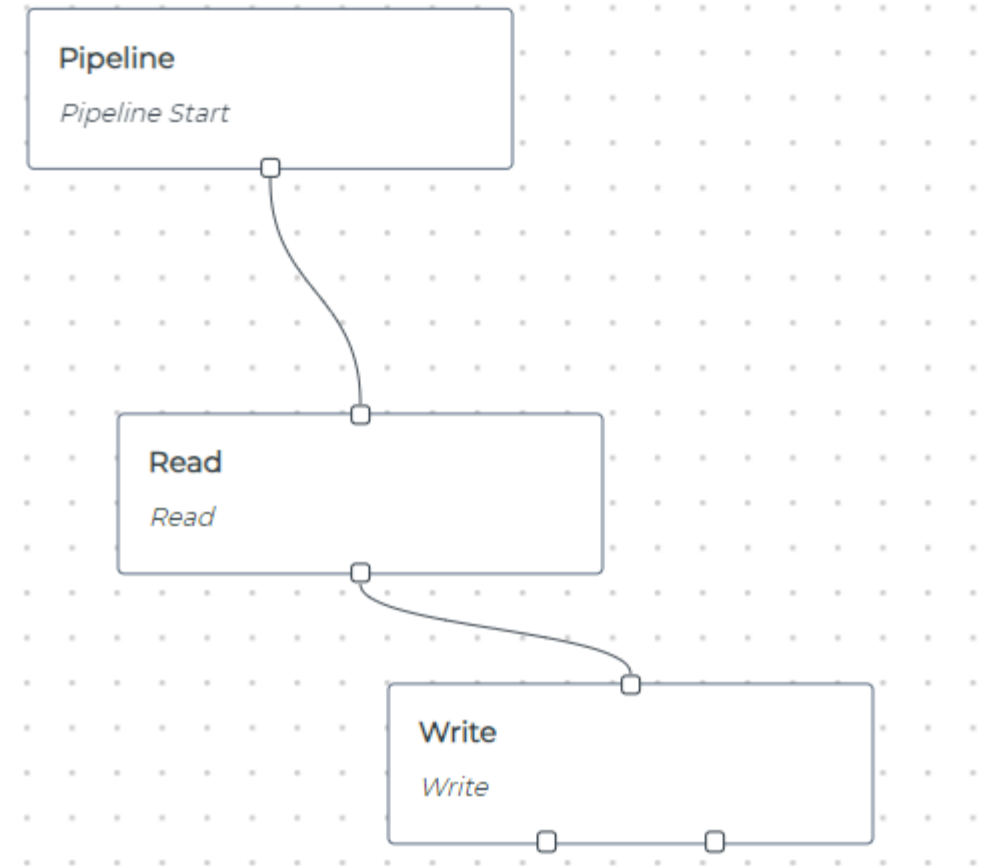


# Índice

- Introdução ao Highbyte
- Arquiteturas do Highbyte
- Major Release – Versão 4.0
- Conexões e Coletas de Tags
- Modelagem de Dados
- Flows
- **Pipelines**
- Conditions
- Exemplo prático

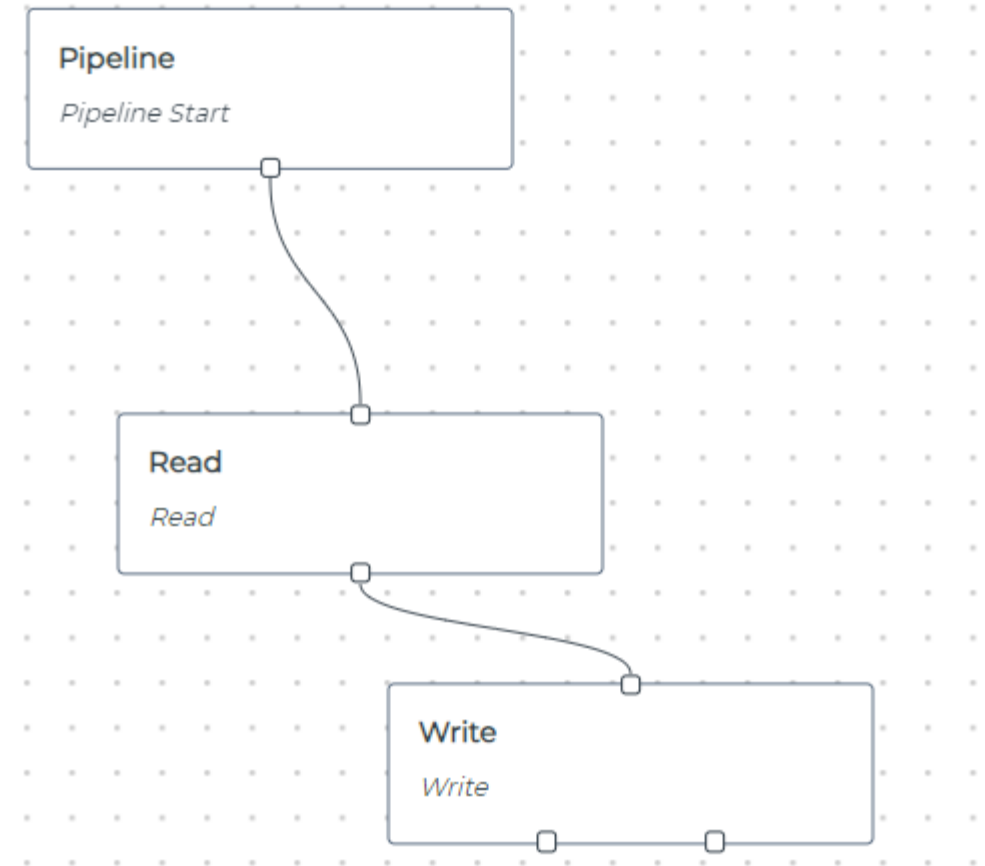
# Pipelines

- Forma mais avançada de manipulação de dados
- Possibilidade de trabalhar com fluxos complexos de dados (if, switch, case fail/succeed, etc)
- Permite criação de fluxos de transformação de dados (script)



# Pipelines

- Outras funcionalidades:
  - Quebrar payloads complexos
  - Bufferizar dados
  - Publicar em um único payload
  - Persistir dados
- OBS (v3.4): pipelines precisam ser executadas por um flow



# Índice

- Introdução ao Highbyte
- Arquiteturas do Highbyte
- Major Release – Versão 4.0
- Conexões e Coletas de Tags
- Modelagem de Dados
- Flows
- Pipelines
- **Conditions**
- Exemplo prático

# Conditions

Forma mais simples de transformação de dados:

- Aggregate
  - RAW, MIN, MAX, AVG, DELTA, COUNT
- Alarm
- Custom
- Deadband

The screenshot shows a configuration form for a data condition. It includes the following fields and controls:

- Condition Type:** A dropdown menu set to "Aggregate".
- Sources:** A text area containing the expression `{{Connection.OPC.FastUInt1}}`. To the right of the text area are icons for editing (a pencil) and deleting (an 'X').
- Scan Interval:** A numeric input field with the value "1" and a unit dropdown menu set to "Seconds".
- Storage Interval:** A numeric input field with the value "10" and a unit dropdown menu set to "Minutes".
- Window Type:** A dropdown menu set to "Sliding".
- Window Interval:** A numeric input field with the value "1" and a unit dropdown menu set to "Minutes".
- Enabled:** A toggle switch that is currently turned on, labeled "On".
- Navigation:** At the bottom, there are two buttons: "Previous" (outlined) and "Submit" (solid red).

# Índice

- Introdução ao Highbyte
- Arquiteturas do Highbyte
- Major Release – Versão 4.0
- Conexões e Coletas de Tags
- Modelagem de Dados
- Flows
- Pipelines
- Conditions
- Exemplo prático

# Exemplo prático

## Pré-requisitos:

- Docker (WSL ou Ubuntu)
  - Powershell (admin): wsl --install + <https://docs.docker.com/engine/install/ubuntu/>
  - Ou <https://www.docker.com/products/docker-desktop/>
- Imagem docker do highbyte (<https://www.highbyte.com/trial-program>)
  - Trial de 30 dias sob solicitação
- Opcional: DBeaver (<https://dbeaver.io/download/>)



**docker**®

**HighByte**



# Exemplo prático

Ambiente simulado em Docker:

- Highbyte
- OPC Server (IOT-Edge OPC Server)
- Banco de Dados (PostgreSQL)

```
git clone git@github.com:hafonseca/highbyte-training
```

Ou: <https://github.com/hafonseca/highbyte-training>

```
docker compose up -d
```



# Exemplo prático

- Verificando se os serviços estão rodando:

```
docker ps
```

CONTAINER ID	IMAGE	COMMAND	CREATED	STATUS	PORTS
39440c3ce859	mcr.microsoft.com/iotedge/opc-plc:latest	"/opcplc --pn=50000..."	6 hours ago	Up 11 minutes	0.0.0.0:8080->8080/tcp, :::8080->8080/tcp,
0.0.0.0:50000->50000/tcp, :::50000->50000/tcp				opcplc-server	
268162ace958	timescale/timescaledb:latest-pg14	"docker-entrypoint.s..."	6 hours ago	Up 13 minutes	0.0.0.0:5432->5432/tcp, :::5432->5432/tcp
				timescaledb	
9f780f6ec088	highbyte:3.4.1	igen..."	30 hours ago	Up 10 minutes	0.0.0.0:1885->1885/tcp, :::1885->1885/tcp,
0.0.0.0:8885->8885/tcp, :::8885->8885/tcp				highbyte	

NAMES
opcplc-server
timescaledb
highbyte

# Exemplo prático

- Criando conexões – OPC

**1** Details

**2** Protocol

**3** Settings

Name	<input type="text" value="OPC"/>
Description	<input type="text"/>
Tags	<input type="text" value="Select..."/>

Next

Cancel

# Exemplo prático

- Criando conexões – OPC

1 Details

2 Protocol

3 Settings

Protocol

OPC UA TCP

Host

opcplc-server

Port

50000

Path

Uri

opc.tcp://opcplc-server:50000

Previous

Next

Cancel

# Exemplo prático

- Criando conexões – OPC

1 Details

2 Protocol

3 Settings

Security

Basic256Sha256-SignEncrypt

▼

Authentication Type

Anonymous

▼

Client Certificate ?

app-certificate-private-key

×

▼

Connect Timeout (seconds)

5

Request Timeout (ms)

5000

# Exemplo prático

- Criando conexões – OPC – Criando Input

The screenshot shows the IIT interface with the following elements highlighted by red boxes and arrows:

- Inputs Tab:** Located at the top of the interface, next to Overview, Outputs, Statistics, and Usage.
- Browse Button:** Located in the Actions section, next to CSV Import and New Input.
- Path Selection:** The breadcrumb path "Root / OpcPlc / Telemetry / Fast" is highlighted.
- Item Selection:** Three items are selected in the left pane: FastUInt4, FastUInt5, and BadFastUInt1. A vertical box highlights the selection icons (+) for these items.
- Import Selected Button:** Located in the top right of the right pane, next to the "3 selected" header.

The right pane shows the selected items: OpcPlc.Telemetry.Fast (with a count of 3), FastUInt1, FastUInt2, and FastUInt3.

# Exemplo prático

- Criando conexões – OPC – Criando Input (ajuste os nomes se preferir)

Connection OPC Vivix Vidros Planos - Professional Help

Overview **Inputs** Outputs Statistics Usage

Filter Actions Browse CSV Import New Input

☐ All Inputs

Name	Type
<input type="checkbox"/> BadFastUInt1	Tag
<input type="checkbox"/> FastUInt1	Tag
<input type="checkbox"/> FastUInt2	Tag
<input type="checkbox"/> FastUInt3	Tag
<input type="checkbox"/> FastUInt4	Tag
<input type="checkbox"/> FastUInt5	Tag

25 ⏪ < 1 > ⏩

# Exemplo prático

- Criando conexões – Banco de Dados

1 Details

2 Protocol

3 Settings

Name

Timescale

Description

Tags

Select... | v

Next

Cancel

# Exemplo prático

- Criando conexões – Banco de Dados

1 Details

2 Protocol

3 Settings

Protocol

PostgreSQL

Host

timescaledb

Port

5432

Previous

Next

Cancel



# Exemplo prático

- Criando conexões – Banco de Dados

1 Details

2 Protocol

3 Settings

Database

postgres

Username

admin

Password

1234

JDBC  
Connection  
String

Enable Store &  
Forward

☐ Off

Previous

Submit

Cancel

# Exemplo prático

- Criando conexões – Banco de Dados – Criando output (flows)

Connection TimescaleDB Vivix Vidros Planos - Professional Help

[Overview](#) [Inputs](#) **[Outputs](#)** [Statistics](#) [Usage](#)

New Output Cancel Create

**Name**

**Table**

**Write Type**  ▼

**Log as JSON** ☐ Off

**Create Table**  ▼

**Transforms**

**Breakup Arrays** ? ☐ Off

**Attribute Filter** ?  Add

# Exemplo prático

- Criando conexões – Banco de Dados – Criando output (pipelines)

Connection TimescaleDB Vivix Vidros Planos - Professional Help

[Overview](#) [Inputs](#) **[Outputs](#)** [Statistics](#) [Usage](#)

New Output Cancel Create

**Name**

**Table**

**Write Type**  ▼

**Log as JSON** ☐ Off

**Create Table**  ▼

**Transforms**

**Breakup Arrays** ? ☐ Off

**Attribute Filter** ?  Add

# Exemplo prático

- Criando transformação de dados – Conditions

1 Details

2 Settings

Name

Aggr\_1

Description

Tags

Select... | v

Next

Cancel

# Exemplo prático

- Criando transformação de dados – Conditions

1 Details

2 Settings

Condition Type

Aggregate

Sources

{{Connection.OPC.FastUInt1}}

Scan Interval

1

Seconds

Storage Interval

10

Minutes

Window Type

Sliding

Window Interval

1

Minutes

Enabled

☒ On

Previous

Submit

Cancel

# Exemplo prático

- Criando modelo – Variavel\_Processo

1 Details

2 Attributes

Name	<input type="text" value="Variavel_Processo"/>
Description	<input type="text"/>
Tags	<input type="text" value="Select..."/>
Group As	<input type="text"/>

Next

Cancel

# Exemplo prático

- Criando modelo – Variavel\_Processo

1 Details

2 Attributes

Name	Type	Array	Required
↕ ValorAtual	Any	<input type="checkbox"/> Off	<input type="checkbox"/> Off
↕ AVGMinuto	Any	<input type="checkbox"/> Off	<input type="checkbox"/> Off
↕ STDMinuto	Any	<input type="checkbox"/> Off	<input type="checkbox"/> Off

⊕ New Attribute

Previous

Submit



Cancel

# Exemplo prático

- Criando modelo – Maquina

**1** Details

**2** Attributes

Name	<input type="text" value="Maquina"/>
Description	<input type="text"/>
Tags	<input type="text" value="Select..."/> 
Group As	<input type="text"/> 

Next

Cancel



# Exemplo prático

- Criando modelo – Maquina

1 Details

2 Attributes

Name	Type	Array	Required
↕ Sensor1	Variavel_Processoo   v	⬇	Off Off
↕ Sensor2	Variavel_Processoo   v	⬇	Off Off
↕ Sensor3	Variavel_Processoo   v	⬇	Off Off

⊕ New Attribute

Previous

Submit

Cancel

# Exemplo prático

- Criando instância – Sensor 1 (Repita para 2 e 3)

1 Details

2 Model

3 Attributes

Name

Sensor\_1

Description

Tags

Select... | v

Group As

 v

Expand Arrays

?

☐ Off

Next

Cancel

# Exemplo prático

- Criando instância – Sensor 1 (Repita para 2 e 3)

1 Details

2 Model

3 Attributes

Model

Variavel\_Processo

Previous

Next

Cancel

# Exemplo prático

- Criando instância – Sensor 1 (Repita para 2 e 3)

1 Details
2 Model
3 Attributes

Name	Expression	Default
ValorAtual Any	<code>{{Condition.Aggregate. Connection_OPC_FastUInt1_RAW}}</code>	
AVGMinuto Any	<code>{{Condition.Aggregate. Connection_OPC_FastUInt1_AVG}}</code>	
STDMinuto Any	<code>{{Condition.Aggregate. Connection_OPC_FastUInt1_DELTA}}</code>	

Previous
Submit
Cancel

nt1\_MIN  
↕ Connection\_OPC\_FastUI ▾  
nt1\_MAX  
↕ Connection\_OPC\_FastUI ▾  
nt1\_AVG  
↕ Connection\_OPC\_FastUI ▾  
nt1\_DELTA  
↕ Connection\_OPC\_FastUI ▾  
nt1\_COUNT  
↕ Connection\_OPC\_FastUI ▾  
nt2\_RAW  
↕ Connection\_OPC\_FastUI ▾  
nt2\_MIN  
↕ Connection\_OPC\_FastUI ▾  
nt2\_MAX  
↕ Connection\_OPC\_FastUI ▾  
nt2\_AVG

# Exemplo prático

- Criando instância – Maquina 1

1 Details

2 Model

3 Attributes

Name

Description

Tags

Group As

Expand Arrays ☐ Off

Next

Cancel

# Exemplo prático

- Criando instância – Maquina 1

1 Details

2 Model

3 Attributes

Name

Description

Tags

Group As

Expand Arrays ☐ Off

Next

Cancel

# Exemplo prático

- Criando instância – Maquina 1

1 Details

2 Model

3 Attributes

Model

Maquina

Previous

Next

Cancel

# Exemplo prático

- Criando instância – Maquina 1

1 Details

2 Model

3 Attributes

Name	Expression	Default
Sensor1 Variavel_Processo	Sensor_1	↓
Sensor2 Variavel_Processo	Sensor_2	↓
Sensor3 Variavel_Processo	Sensor_3	↓

Previous

Submit

Cancel



# Exemplo prático

- Criando Flow

1 Details

2 Sources & Targets

3 Settings

Name

To\_DB

Description

Tags

Select...

Group As

Next

Cancel

# Exemplo prático

- Criando Flow – Inserindo sources/targets

1 Details

2 Sources & Targets

3 Settings

Sources

{{Instance.Sensor\_1}}

{{Instance.Sensor\_2}}

{{Instance.Sensor\_3}}

Targets

{{Connection.TimescaleDB.Timeseries\_Flows}}

Previous

Next

Cancel

# Exemplo prático

- Criando Flow – Ajustes de intervalo de transporte

New Flow Vivix Vidros Planos - Professional Help

1 Details 2 Sources & Targets 3 Settings

Type Polled | v

Interval 1 Seconds | v

Mode Always | v

Publish Mode All values | v

Enabled ☐ Off

Previous Submit Cancel

# Exemplo prático

- Criando Flow – Verificando o Status do flow

☐ All Flows

Name	Status
<input type="checkbox"/> To_DB	<span>✓ OK</span>

25
 

⏮
 <
 1
 >
 ⏭

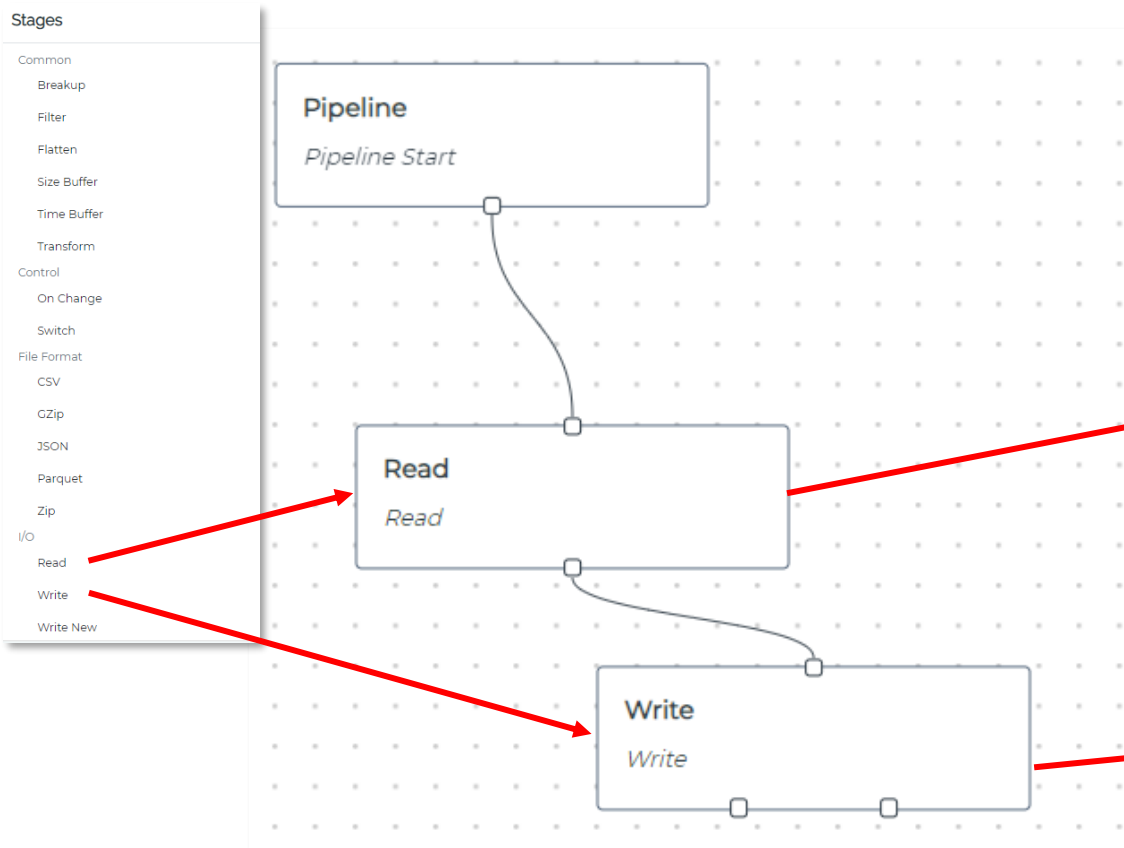
# Exemplo prático

- Criando Flow – Verificando os dados no banco (DBeaver)

testdsada							
Properties Data ER Diagram							
testdsada Enter a SQL expression to filter results (use Ctrl+Space)							
Grid	123_id	A-Z_name	A-Z_model	123_timestamp	123_AVGMinuto	123_STDMinuto	
1	1	Sensor_2	Variavel_Processo	1,729,045,038,338	6,618	60	
2	2	Sensor_3	Variavel_Processo	1,729,045,038,338	6,618	59	
3	3	Sensor_1	Variavel_Processo	1,729,045,039,336	6,619	60	
4	4	Sensor_2	Variavel_Processo	1,729,045,039,336	6,619	60	
5	5	Sensor_3	Variavel_Processo	1,729,045,039,336	6,619	59	
6	6	Sensor_1	Variavel_Processo	1,729,045,040,337	6,620	60	
7	7	Sensor_2	Variavel_Processo	1,729,045,040,337	6,620	60	
8	8	Sensor_3	Variavel_Processo	1,729,045,040,337	6,620	59	
9	9	Sensor_1	Variavel_Processo	1,729,045,041,342	6,621	60	
10	10	Sensor_2	Variavel_Processo	1,729,045,041,342	6,621	60	
11	11	Sensor_3	Variavel_Processo	1,729,045,041,342	6,621	59	
12	12	Sensor_1	Variavel_Processo	1,729,045,042,337	6,622	60	
13	13	Sensor_2	Variavel_Processo	1,729,045,042,337	6,622	60	
14	14	Sensor_3	Variavel_Processo	1,729,045,042,337	6,622	59	
15	15	Sensor_1	Variavel_Processo	1,729,045,043,317	6,623	60	
16	16	Sensor_2	Variavel_Processo	1,729,045,043,317	6,623	60	

# Exemplo prático

- Criando pipeline básica



### Read Stage

**Name**  
Read

**Input Event Value** ⓘ  
Drop

**Sources** ⓘ

**Merge Read Value** ⓘ  
Inline

**Reference**  
{{Instance.Sensor1}}

+ Add Source

### Write Stage

**Name**  
Write

**Targets**

{{Connection.TimescaleDB  
.Timeseries\_Pipeline}} ⓘ

# Exemplo prático

- Criando pipeline básica

☰

Test Write

✎

Expression to write

Write Pipeline

🔔

1

🖨️

📊

Results

Status

Success

⌚

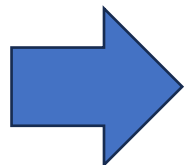
Execution Time

< 1 ms

📄

Truncated

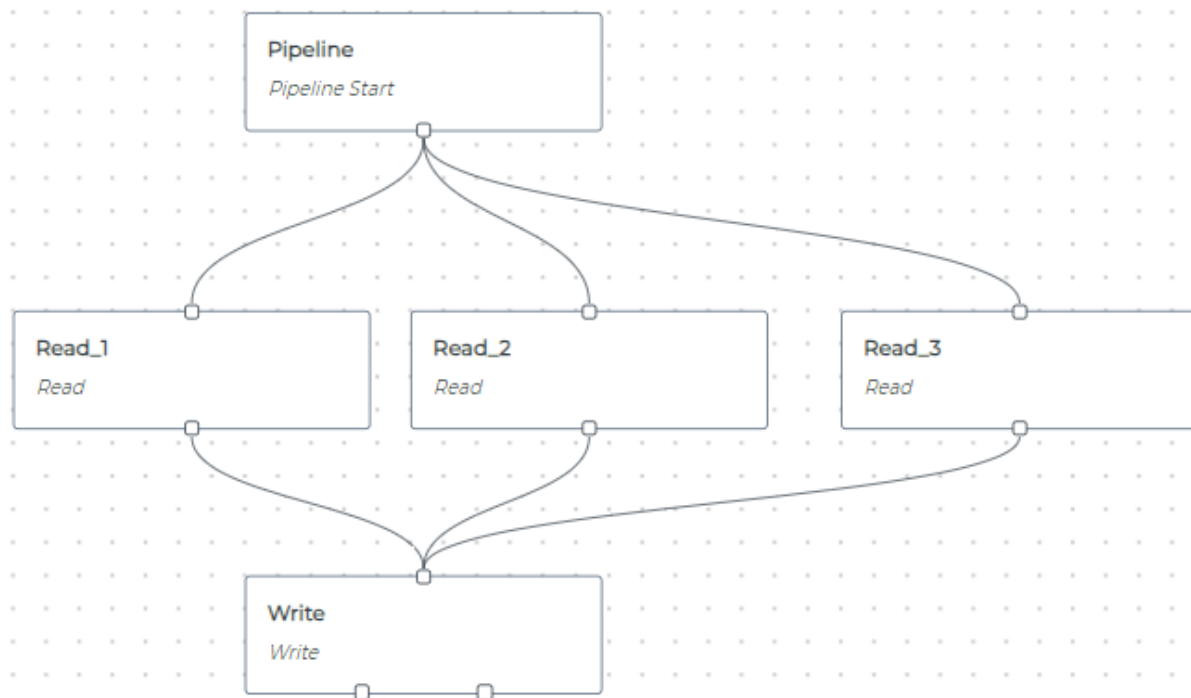
No



123_id	ABC_name	ABC_model	123_timestamp	123 AVGMinuto	123 STDMinuto
1		ComplexData	1,729,076,211,215	8,756	60
2		ComplexData	1,729,076,215,859	8,761	60
3		ComplexData	1,729,076,219,773	8,765	60
4		ComplexData	1,729,076,239,479	8,784	60
5		ComplexData	1,729,076,239,586	8,784	60
6		ComplexData	1,729,076,239,676	8,784	60
7		ComplexData	1,729,076,239,779	8,785	60
8		ComplexData	1,729,076,239,863	8,785	60
9		ComplexData	1,729,076,239,967	8,785	60
10		ComplexData	1,729,076,240,456	8,785	60
11		ComplexData	1,729,076,240,543	8,785	60
12		ComplexData	1,729,076,240,654	8,785	60
13		ComplexData	1,729,076,240,758	8,786	60
14		ComplexData	1,729,076,240,821	8,786	60
15		ComplexData	1,729,076,240,925	8,786	60
16		ComplexData	1,729,076,241,028	8,786	60

# Exemplo prático

- Criando pipeline básica



**Read Stage**

Name  
Read\_1

Input Event Value  
Drop

Sources  
Merge Read Value  
With Key

Key  
Sensor\_1

Reference  
{{Instance.Sensor\_1}}

+ Add Source

**Read Stage**

Name  
Read\_2

Input Event Value  
Drop

Sources  
Merge Read Value  
With Key

Key  
Sensor\_2

Reference  
{{Instance.Sensor\_2}}

+ Add Source

**Read Stage**

Name  
Read\_3

Input Event Value  
Drop

Sources  
Merge Read Value  
With Key

Key  
Sensor\_3

Reference  
{{Instance.Sensor\_3}}

+ Add Source



# Exemplo prático

- Criando pipeline básica

Test Write

Expression to write

Write Pipeline

1

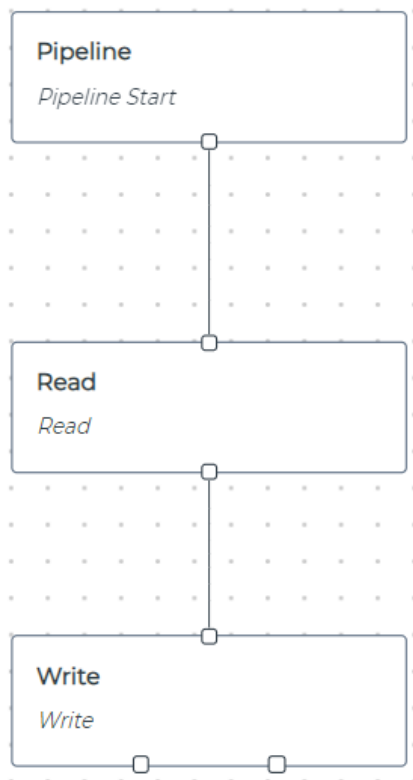
Results



123_id	ABC_name	ABC_model	123_timestamp	123 Sensor_1_AVGMinuto	123 Sensor_1_STDMinuto	123 Sensor_2_AVGMinuto	123 Sensor_2_STDMinuto	123 Sensor_3_AVGMinuto
1		ComplexData	1,729,076,438,609	8,983	60	[NULL]	[NULL]	[NULL]
2		ComplexData	1,729,076,438,609	[NULL]	[NULL]	8,983	60	[NULL]
3		ComplexData	1,729,076,459,950	[NULL]	[NULL]	[NULL]	[NULL]	[NULL]
4		ComplexData	1,729,076,459,950	[NULL]	[NULL]	9,005	60	[NULL]
5		ComplexData	1,729,076,459,950	9,005	60	[NULL]	[NULL]	[NULL]

# Exemplo prático

- Criando pipeline básica



**Read Stage**

Name  
Read

Input Event Value  
Drop

Sources

Merge Read Value  
With Key

Key  
Sensor\_1

Reference  
{{Instance.Sensor\_1}}

Merge Read Value  
With Key

Key  
Sensor\_2

Reference  
{{Instance.Sensor\_2}}

Merge Read Value  
With Key

Key  
Sensor\_3

Reference  
{{Instance.Sensor\_3}}

+ Add Source

# Exemplo prático

- Criando pipeline básica

Test Write

Expression to write

Write Pipeline

1

Results



model	123_timestamp	123 Sensor_1_AVGMinuto	123 Sensor_1_STDMinuto	123 Sensor_2_AVGMinuto	123 Sensor_2_STDMinuto	123 Sensor_3_AVGMinuto	123 Sensor_3_STDMinuto
plexData	1,729,076,763,207	9,308	60	9,308	60	9,308	60
plexData	1,729,076,771,375	9,316	60	9,316	60	9,316	60
plexData	1,729,076,771,546	9,316	60	9,316	60	9,316	60
plexData	1,729,076,771,730	9,316	60	9,316	60	9,316	60
plexData	1,729,076,771,895	9,316	60	9,316	60	9,316	60
plexData	1,729,076,772,166	9,317	60	9,317	60	9,317	60

# Exemplo prático

- Integrando pipeline a flow (v3.4)

Output Timeseries\_Flow\_Pipeline

**Details** Usage

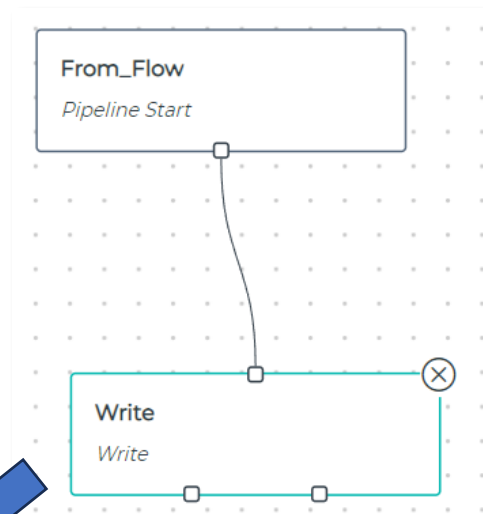
Name: Timeseries\_Flow\_Pipeline

Table: sensorflowspipeline

Write Type: Insert

Log as JSON: ☐ Off

Create Table: Create & Update



**Write Stage**

Name: Write

Targets:

{{Connection.TimescaleDB.Timeseries\_Flow\_Pipeline}}



**Test Write**

Expression to write: Write Pipeline

{{Instance.Sensor\_1}}

**Results**

Status	Success
Execution Time	16 ms
Truncated	No



123_id	ABC_name	ABC_model	123_timestamp	123_AVGMinuto	123_STDMinuto
1	Sensor_1	Variavel_Processo	1,729,079,499,814	12,045	60

# Exemplo prático

- Integrando pipeline a flow (v3.4)

1 Details

2 Sources & Targets

3 Settings

Name

To\_Pipeline

Description

Tags

Select...

Group As

Next

Cancel

1 Details

2 Sources & Targets

3 Settings

Sources

{{Instance.Sensor\_1}}

{{Instance.Sensor\_2}}

{{Instance.Sensor\_3}}

Targets

{{Pipeline.From\_Flow}}

Previous

Next

Cancel

1 Details

2 Sources & Targets

3 Settings

Type

Polled

Interval

1

Seconds

Mode

Always

Publish Mode

All values

Enabled

Off

Previous

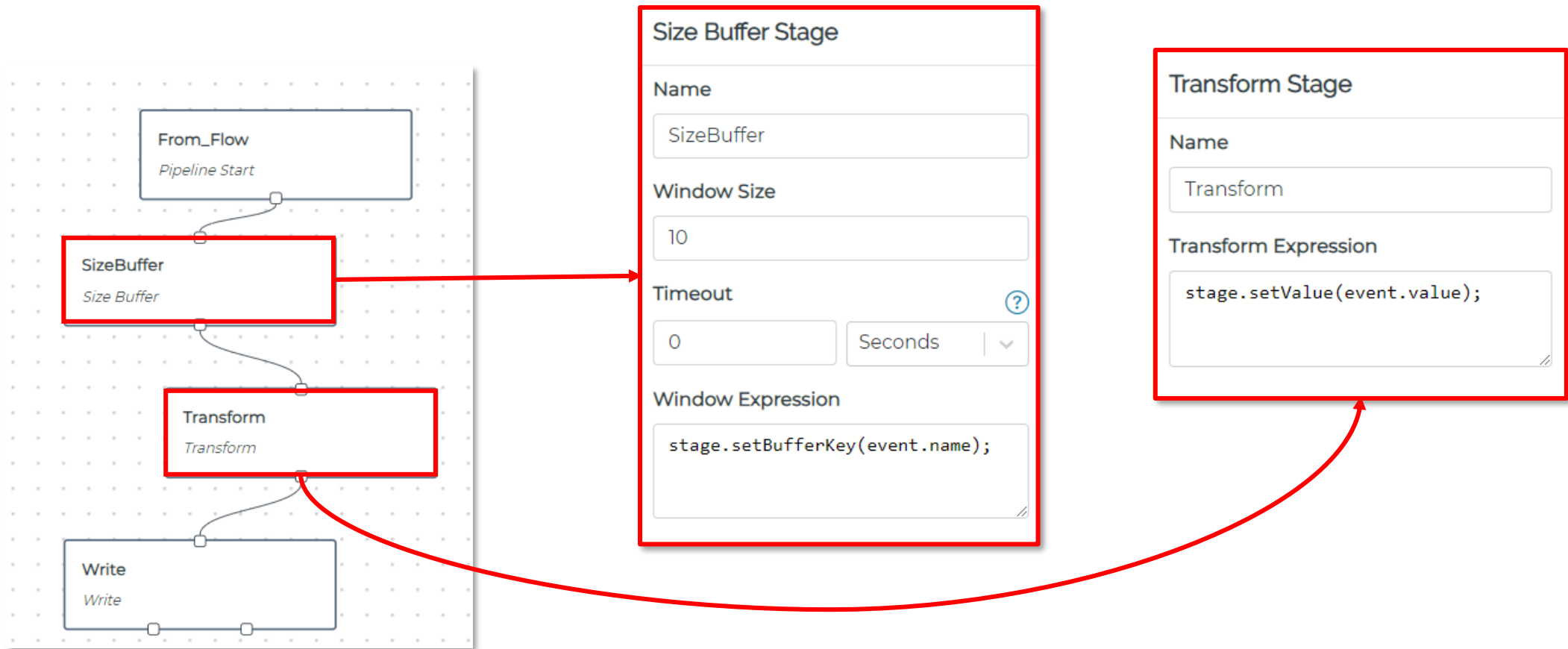
Submit

Cancel

123_id	ABC_name	ABC_model	123_timestamp	123_AVGMinuto	123_STDMinuto
43	Sensor_3	Variavel_Processo	1,729,079,652,745	12,198	60
42	Sensor_2	Variavel_Processo	1,729,079,652,745	12,198	60
41	Sensor_1	Variavel_Processo	1,729,079,652,745	12,198	60
40	Sensor_3	Variavel_Processo	1,729,079,651,738	12,197	60
39	Sensor_2	Variavel_Processo	1,729,079,651,738	12,197	60
38	Sensor_1	Variavel_Processo	1,729,079,651,738	12,197	60
37	Sensor_3	Variavel_Processo	1,729,079,650,729	12,196	60
36	Sensor_2	Variavel_Processo	1,729,079,650,729	12,196	60
35	Sensor_1	Variavel_Processo	1,729,079,650,729	12,196	60
34	Sensor_3	Variavel_Processo	1,729,079,649,743	12,195	60
33	Sensor_2	Variavel_Processo	1,729,079,649,743	12,195	60
32	Sensor_1	Variavel_Processo	1,729,079,649,743	12,195	60
31	Sensor_3	Variavel_Processo	1,729,079,648,737	12,194	60
30	Sensor_2	Variavel_Processo	1,729,079,648,737	12,194	60
29	Sensor_1	Variavel_Processo	1,729,079,648,737	12,194	60
28	Sensor_3	Variavel_Processo	1,729,079,647,754	12,193	60
27	Sensor_2	Variavel_Processo	1,729,079,647,754	12,192	60

# Exemplo prático

- Criando pipeline com transformação



# Exemplo prático

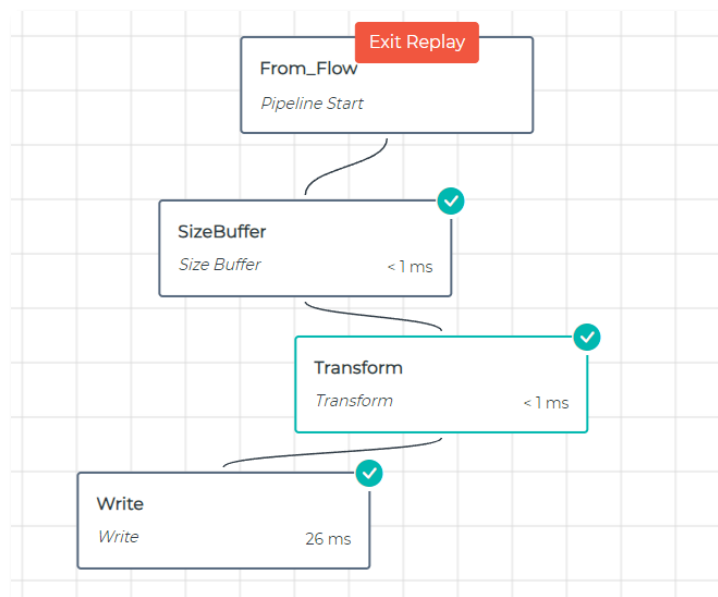
- Criando pipeline com transformação

Details

Replay Clear Refresh

Only show Failures ☐ Off

To_Pipeline	Success
16/10/2024, 09:30:32	
31 seconds ago	28 ms
Exit Replay	
To_Pipeline	Success
16/10/2024, 09:30:32	
31 seconds ago	28 ms
To_Pipeline	Success
16/10/2024, 09:30:32	
31 seconds ago	22 ms
To_Pipeline	Success
16/10/2024, 09:30:32	
31 seconds ago	13 ms
To_Pipeline	Success
16/10/2024, 09:30:31	
32 seconds ago	< 1 ms



Transform Stage

Name

Transform

Activity Stats Config

Event Info

Type

Data Event

Value ?

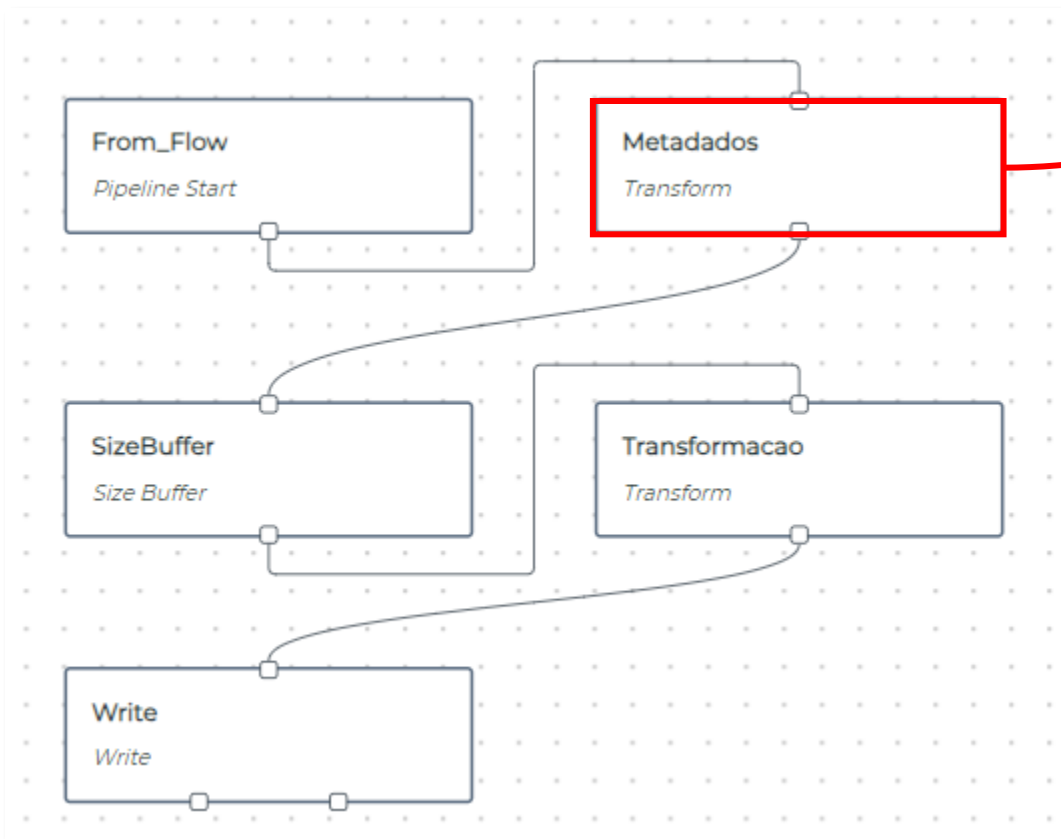
```
[
  {
    "AVGMinuto": 14375,
    "STDMinuto": 60
  },
  {
    "AVGMinuto": 14375,
    "STDMinuto": 60
  }
]
```

Metadata ?

```
{
  "bufferKey": "Sensor_3"
}
```

# Exemplo prático

- Criando pipeline com transformação



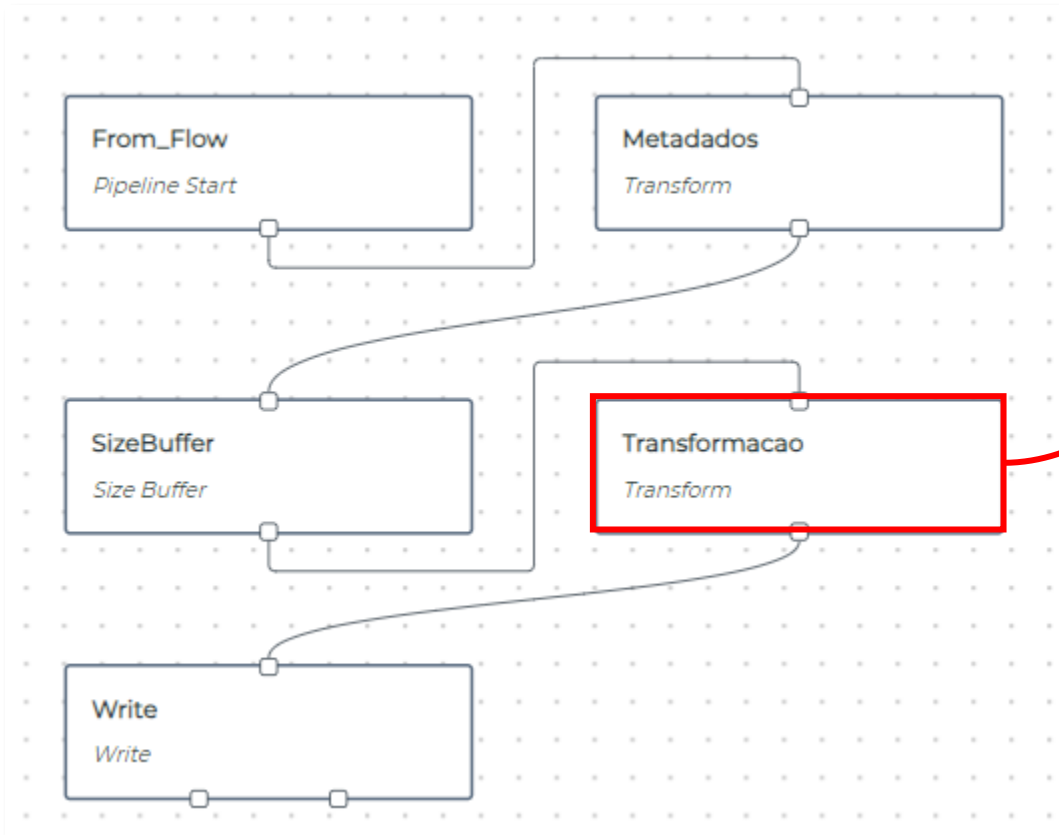
```

1  const avgMinutos = event.value.AVGMinuto;
2  const stdMinutos = event.value.STDMinuto;
3
4  event.value = [
5    {
6      "name": event.name,
7      "model": event.model,
8      "AVGMinuto": avgMinutos,
9      "STDMinuto": stdMinutos,
10   }
11 ];
12
13 stage.setValue(event.value);
  
```



# Exemplo prático

- Criando pipeline com transformação



```

1  const avgMinutos = event.value.map(item => item.AVGMinuto);
2
3  const firstAVG = avgMinutos[0];
4  const lastAVG = avgMinutos[avgMinutos.length - 1];
5
6  // Calculate standard deviation
7  const meanAVG = avgMinutos.reduce((sum, value) => sum + value, 0) / avgMinutos.length;
8  const variance = avgMinutos.reduce((sum, value) => sum + Math.pow(value - meanAVG, 2), 0) / avgMinutos.length;
9  const stdAVG = Math.sqrt(variance);
10
11 // Calculate energy AVG (sqrt of sum of squares)
12 const energyAVG = Math.sqrt(avgMinutos.reduce((sum, value) => sum + Math.pow(value, 2), 0));
13
14 // Extract name and model from the first object
15 const eventName = event.value[0].name;
16 const eventModel = event.value[0].model;
17
18 // Create the new event.value array
19 event.value = [
20   {
21     "name": eventName,
22     "model": eventModel,
23     "FirstAVG": firstAVG,
24     "LastAVG": lastAVG,
25     "StdAVG": stdAVG,
26     "CountAVG": avgMinutos.length,
27     "EnergyAVG": energyAVG
28   }
29 ];
30
31 // Commit the new event.value
32 stage.setValue(event.value);
  
```

# Exemplo prático

- Criando pipeline com transformação

123_id ↓	ABC name ↓	ABC model ↓	123 FirstAVG ↓	123 LastAVG ↓	123 StdAVG ↓	123 CountAVG ↓	123 EnergyAVG ↓	123_timestamp ↓	ABC_name ↓
70	Sensor_2	Variavel_Processo	15,600	15,606	2	20	69,779.4086604351	1,729,083,061,522	
69	Sensor_3	Variavel_Processo	15,593	15,600	2	20	69,749.6689741249	1,729,083,055,512	
68	Sensor_1	Variavel_Processo	15,587	15,593	2	20	69,719.9292526893	1,729,083,048,525	
67	Sensor_1	Variavel_Processo	15,587	15,587	0	3	26,997.4759375761	1,729,083,042,305	
66	Sensor_1	Variavel_Processo	15,586	15,586	0	3	26,995.7438867685	1,729,083,041,313	
65	Sensor_1	Variavel_Processo	15,585	15,585	0	3	26,994.011835961	1,729,083,040,312	
64	Sensor_1	Variavel_Processo	15,584	15,584	0	3	26,992.2797851534	1,729,083,039,309	
63	Sensor_1	Variavel_Processo	15,583	15,583	0	3	26,990.5477343458	1,729,083,038,297	
62	Sensor_1	Variavel_Processo	15,582	15,582	0	3	26,988.8156835382	1,729,083,037,292	
61	Sensor_1	Variavel_Processo	15,581	15,581	0	3	26,987.0836327307	1,729,083,036,298	
60	Sensor_1	Variavel_Processo	15,580	15,580	0	3	26,985.3515819231	1,729,083,035,311	
59	Sensor_1	Variavel_Processo	15,579	15,579	0	3	26,983.6195311155	1,729,083,034,302	
58	Sensor_1	Variavel_Processo	15,578	15,578	0	3	26,981.887480308	1,729,083,033,298	
57	Sensor_1	Variavel_Processo	15,577	15,577	0	3	26,980.1554295004	1,729,083,032,309	
56	Sensor_1	Variavel_Processo	15,576	15,576	0	3	26,978.4233786928	1,729,083,031,297	
55	Sensor_1	Variavel_Processo	15,575	15,575	0	3	26,976.6913278853	1,729,083,030,288	
54	Sensor_1	Variavel_Processo	15,574	15,574	0	3	26,974.9592770777	1,729,083,029,304	
53	Sensor_1	Variavel_Processo	15,573	15,573	0	3	26,973.2272262701	1,729,083,028,293	
52	Sensor_1	Variavel_Processo	15,572	15,572	0	3	26,971.4951754626	1,729,083,027,304	
51	Sensor_1	Variavel_Processo	15,571	15,571	0	3	26,969.763124655	1,729,083,026,296	
50	Sensor_1	Variavel_Processo	15,570	15,570	0	3	26,968.0310738474	1,729,083,025,289	
49	Sensor_1	Variavel_Processo	15,569	15,569	0	3	26,966.2990230398	1,729,083,024,285	

# OBRIGADO!

# DÚVIDAS?



E-mail: [haf@poli.br](mailto:haf@poli.br)



WhatsApp:  [\(81\) 9 9292-0944](https://api.whatsapp.com/send?phone=5581992920944)



Discord: [Hugo Fonsêca#7312](https://discord.com/users/HugoFonsêca7312)