

Search Medium

[Write](https://medium.com/new-story?source=---two_column_layout_nav----------------------------------)

Programmer Muhammad

Get unlimited access to all of Medium for less than $1/week.

[Become a member](https://medium.com/plans?source=upgrade_membership---post_top_nav_upsell----------------------------------)

**PySpark Examples: Real-time, Batch, and Stream Processing for Data Professionals**

[[](https://pawankg.medium.com/?source=post_page-----21d4e3891f0f--------------------------------)](https://pawankg.medium.com/?source=post_page-----21d4e3891f0f--------------------------------)

[Pawan Kumar Ganjhu](https://pawankg.medium.com/?source=post_page-----21d4e3891f0f--------------------------------)

·

Follow

6 min read

·

May 25

5



[Source](https://astropics.com/photo/sequoia-hume-lake-pano/?gallery=panoramas)

Real-time processing, batch processing, and stream processing are three different approaches to handling data in various applications. Let’s explore each of them along with examples:

1. **Real-time Processing:** Real-time processing involves handling data as it arrives, providing immediate responses or actions. It is characterized by low latency and requires near-instantaneous processing. Real-time processing is commonly used in applications that require immediate feedback or timely actions. Some examples include:

* Online transaction processing (OLTP): In e-commerce, when a customer places an order, real-time processing ensures that the inventory is updated, payment is processed, and order confirmation is sent instantly.
* Sensor data processing: In industries like manufacturing and Internet of Things (IoT), real-time processing is crucial for monitoring sensor data in real-time. For instance, in a smart grid system, real-time processing is used to detect anomalies, manage power distribution, and respond to outages promptly.
* Fraud detection: Real-time processing is utilized in financial institutions to detect fraudulent transactions in real-time. Algorithms analyze patterns and behavior to identify suspicious activities and trigger immediate actions, such as blocking a transaction or notifying the customer.

2. **Batch Processing:** Batch processing involves collecting a set of data over a period of time and processing it as a batch or group. It is characterized by high throughput but usually incurs a delay between data collection and processing. Batch processing is commonly used when data can be processed in a non-time-critical manner. Some examples include:

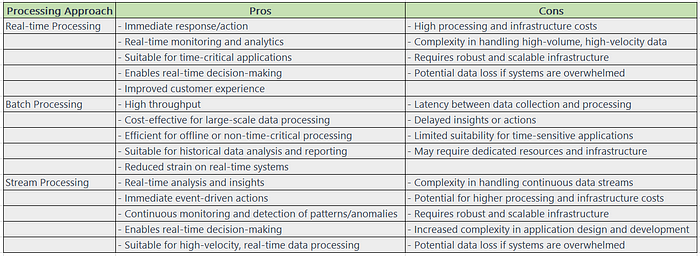
* Data analytics: In business intelligence and analytics, batch processing is often employed to analyze large volumes of historical data to derive insights. Data is collected over a period, and then processed offline or during non-peak hours to generate reports, perform data mining, or train machine learning models.
* Log processing: Systems generate log files that record various events or activities. Batch processing is used to analyze these logs periodically to identify trends, errors, or performance issues.
* Billing and invoicing: In finance or utility sectors, billing processes often involve aggregating usage data over a billing period and generating invoices or bills for customers. Batch processing is used to consolidate the data and perform calculations to generate accurate invoices.

3. **Stream Processing**: Stream processing involves continuously processing a sequence of data records in real-time. It focuses on processing data as it flows, enabling near-instantaneous analysis and action. Stream processing is commonly used in applications that require real-time analytics, monitoring, or event-driven actions. Some examples include:

* Social media analytics: Streaming platforms monitor and process social media feeds in real-time, analyzing user interactions, sentiment, and trending topics. This allows businesses to respond quickly to customer feedback or detect emerging trends.
* Stock market analysis: Stream processing is utilized in high-frequency trading, where real-time market data is continuously analyzed to make split-second trading decisions. Algorithms process streams of market data, apply complex models, and trigger trades based on predefined rules.
* Real-time monitoring: Stream processing is used in systems that monitor network traffic, server logs, or security events in real-time. The data is continuously analyzed, and alerts or actions are triggered when specific patterns or anomalies are detected.

It’s worth noting that these three processing approaches are not mutually exclusive and can be combined or used in conjunction depending on the requirements of the application.

**Pros and Cons**



It’s important to note that the advantages and disadvantages mentioned above are general in nature and can vary depending on the specific use case, infrastructure, and implementation details. Organizations should carefully evaluate their requirements and constraints before choosing the most appropriate processing approach.

**Challenges**

**Real-time Processing:**

1. High processing and infrastructure costs: Real-time processing requires powerful computing resources and efficient infrastructure, which can be costly to set up and maintain.
2. Handling high-volume, high-velocity data: Real-time systems need to handle large volumes of data and process it quickly, posing challenges in data ingestion, storage, and processing efficiency.
3. Data quality and consistency: Real-time processing relies on the availability of reliable and accurate data in real time. Ensuring data quality, consistency, and integrity can be challenging, especially when dealing with data from multiple sources or with varying formats.
4. Scalability and reliability: Real-time systems must be able to scale seamlessly to handle increasing data volumes and provide continuous availability without any downtime.
5. Complexity in application design and development: Real-time processing requires careful design and implementation, including efficient algorithms, event handling, and real-time analytics, which can be complex to develop and maintain.

**Batch Processing:**

1. Latency between data collection and processing: Batch processing involves a delay between data collection and processing, which may not be suitable for time-critical applications or real-time decision-making.
2. Delayed insights and actions: Since data is processed in batches, it may take time to generate insights or trigger actions based on the processed data.
3. Scalability and resource allocation: Handling large-scale batch processing efficiently requires proper resource allocation, including computing power, storage, and network bandwidth.
4. Data dependencies and dependencies on external systems: Batch processing often involves dependencies on other data sources or systems, which can complicate the processing workflow and introduce potential bottlenecks.
5. Balancing system utilization: Batch processing systems need to balance resource utilization to ensure optimal performance and avoid overloading or underutilizing resources during processing.

**Stream Processing:**

1. Handling continuous data streams: Stream processing must handle data streams that are continuous and potentially infinite in size. Managing the velocity, volume, and variety of incoming data can be challenging.
2. Handling out-of-order data: Data streams can arrive out of order, making it necessary to handle and process data in a way that maintains the correct temporal order for accurate analysis.
3. Scalability and fault tolerance: Stream processing systems must scale horizontally to handle increasing data volumes and be fault-tolerant to handle failures without losing data or compromising processing integrity.
4. Complex event processing: Stream processing often involves detecting complex patterns, correlations, or anomalies in real-time. Designing and implementing efficient algorithms for event processing can be challenging.
5. Data synchronization and state management: Stream processing may require synchronization and management of state across multiple processing units or instances, adding complexity to the system design and implementation.

It’s important to note that these challenges can vary depending on the specific use case, technology stack, and infrastructure in place. Organizations must carefully consider these challenges and devise appropriate strategies to overcome them when implementing real-time, batch, or stream processing solutions.

Here are examples of PySpark code snippets demonstrating how to handle real-time processing, batch processing, and stream processing using Apache Spark:

1. Real-time Processing with PySpark:

from pyspark.sql import SparkSession  
from pyspark.sql.functions import window  
  
# Create SparkSession  
spark = SparkSession.builder.appName("RealTimeProcessing").getOrCreate()  
  
# Read real-time data from a Kafka topic  
df = spark.readStream.format("kafka") \  
 .option("kafka.bootstrap.servers", "localhost:9092") \  
 .option("subscribe", "my\_topic") \  
 .load()  
  
# Apply real-time transformations or analytics  
result = df.select("value").withColumn("word\_count", F.size(F.split(F.col("value"), " ")))  
  
# Write the real-time processed data to an output sink  
query = result.writeStream.format("console").start()  
  
# Await termination of the streaming query  
query.awaitTermination()

In this example, PySpark reads real-time data from a Kafka topic, performs a transformation (calculating word count), and writes the processed data to the console. The awaitTermination() method ensures the streaming query continues until manually stopped.

2. Batch Processing with PySpark:

from pyspark.sql import SparkSession  
  
# Create SparkSession  
spark = SparkSession.builder.appName("BatchProcessing").getOrCreate()  
  
# Read batch data from a file  
df = spark.read.csv("input.csv", header=True, inferSchema=True)  
  
# Apply batch transformations or analytics  
result = df.groupBy("category").count()  
  
# Write the batch processed data to an output file  
result.write.csv("output.csv")

In this example, PySpark reads batch data from a CSV file, performs a transformation (grouping data by category and counting occurrences), and writes the processed data to a CSV file. It operates on the entire dataset at once, as opposed to streaming data.

3. Stream Processing with PySpark:

from pyspark.sql import SparkSession  
  
# Create SparkSession  
spark = SparkSession.builder.appName("StreamProcessing").getOrCreate()  
  
# Read data stream from a socket  
lines = spark.readStream.format("socket") \  
 .option("host", "localhost") \  
 .option("port", 9999) \  
 .load()  
  
# Apply stream transformations or analytics  
wordCounts = lines.groupBy("value").count()  
  
# Write the stream processed data to an output sink  
query = wordCounts.writeStream.outputMode("complete") \  
 .format("console").start()  
  
# Await termination of the streaming query  
query.awaitTermination()

In this example, PySpark reads a stream of data from a socket, performs a transformation (word count), and writes the processed data to the console. The awaitTermination() method ensures the streaming query continues until manually stopped.

These code snippets demonstrate the basic structure of handling real-time, batch, and stream processing with PySpark. However, please note that these examples provide a starting point, and additional configurations or customizations might be required based on specific data sources, transformations, or output sinks used in your environment.

[Pyspark](https://medium.com/tag/pyspark?source=post_page-----21d4e3891f0f---------------pyspark-----------------)

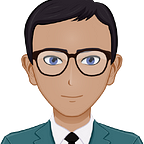
[Real Time Analytics](https://medium.com/tag/real-time-analytics?source=post_page-----21d4e3891f0f---------------real_time_analytics-----------------)

[Batch Processing](https://medium.com/tag/batch-processing?source=post_page-----21d4e3891f0f---------------batch_processing-----------------)

[Stream Processing](https://medium.com/tag/stream-processing?source=post_page-----21d4e3891f0f---------------stream_processing-----------------)

[Real Time Processing](https://medium.com/tag/real-time-processing?source=post_page-----21d4e3891f0f---------------real_time_processing-----------------)

5

[[](https://pawankg.medium.com/?source=post_page-----21d4e3891f0f--------------------------------)](https://pawankg.medium.com/?source=post_page-----21d4e3891f0f--------------------------------)

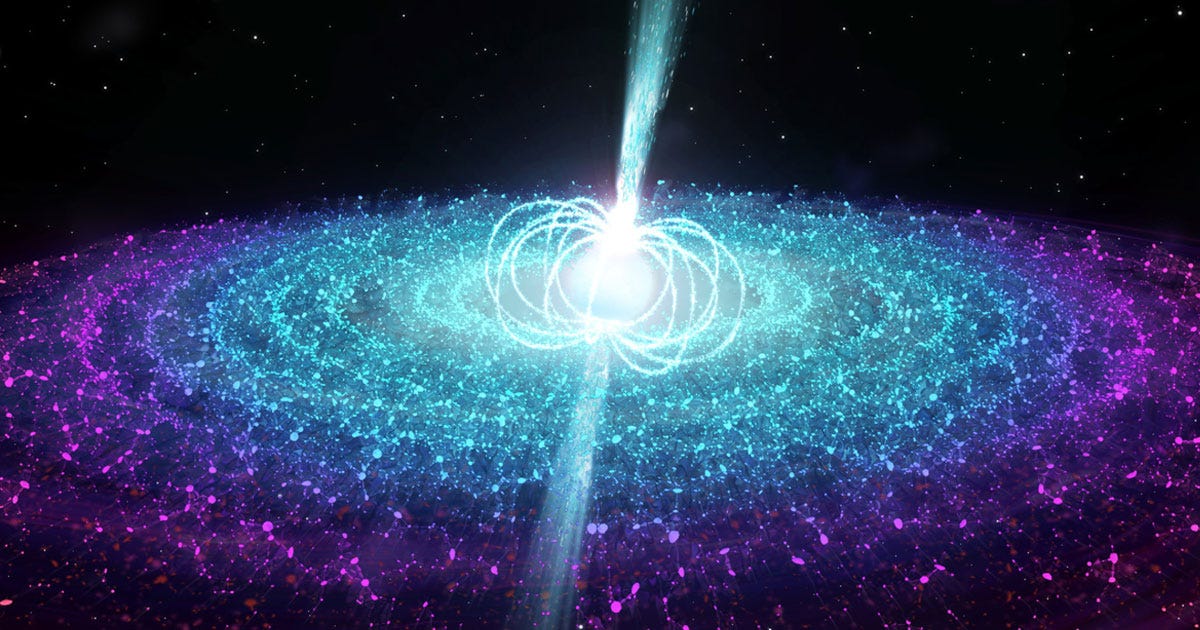
[Written by Pawan Kumar Ganjhu](https://pawankg.medium.com/?source=post_page-----21d4e3891f0f--------------------------------)

[72 Followers](https://pawankg.medium.com/followers?source=post_page-----21d4e3891f0f--------------------------------)

Data Engineer | Data & AI | R&D | Data Science | Data Analytics | Cloud

Follow

More from Pawan Kumar Ganjhu

[[](https://pawankg.medium.com/pyspark-for-data-engineering-beginners-an-extensive-guide-ce29520a78ca?source=author_recirc-----21d4e3891f0f----0---------------------c73415e4_e3f6_44ef_8f42_54798ba50818-------)](https://pawankg.medium.com/pyspark-for-data-engineering-beginners-an-extensive-guide-ce29520a78ca?source=author_recirc-----21d4e3891f0f----0---------------------c73415e4_e3f6_44ef_8f42_54798ba50818-------)

[[Pawan Kumar Ganjhu](https://pawankg.medium.com/?source=author_recirc-----21d4e3891f0f----0---------------------c73415e4_e3f6_44ef_8f42_54798ba50818-------)](https://pawankg.medium.com/?source=author_recirc-----21d4e3891f0f----0---------------------c73415e4_e3f6_44ef_8f42_54798ba50818-------)

[Pawan Kumar Ganjhu](https://pawankg.medium.com/?source=author_recirc-----21d4e3891f0f----0---------------------c73415e4_e3f6_44ef_8f42_54798ba50818-------)

**[PySpark for Data Engineering Beginners: An Extensive Guide](https://pawankg.medium.com/pyspark-for-data-engineering-beginners-an-extensive-guide-ce29520a78ca?source=author_recirc-----21d4e3891f0f----0---------------------c73415e4_e3f6_44ef_8f42_54798ba50818-------)**

[Introduction](https://pawankg.medium.com/pyspark-for-data-engineering-beginners-an-extensive-guide-ce29520a78ca?source=author_recirc-----21d4e3891f0f----0---------------------c73415e4_e3f6_44ef_8f42_54798ba50818-------)

[10 min read·May 10](https://pawankg.medium.com/pyspark-for-data-engineering-beginners-an-extensive-guide-ce29520a78ca?source=author_recirc-----21d4e3891f0f----0---------------------c73415e4_e3f6_44ef_8f42_54798ba50818-------)

15

[[](https://pawankg.medium.com/testing-data-pipelines-python-examples-for-etl-validation-and-verification-431f482a8d47?source=author_recirc-----21d4e3891f0f----1---------------------c73415e4_e3f6_44ef_8f42_54798ba50818-------)](https://pawankg.medium.com/testing-data-pipelines-python-examples-for-etl-validation-and-verification-431f482a8d47?source=author_recirc-----21d4e3891f0f----1---------------------c73415e4_e3f6_44ef_8f42_54798ba50818-------)

[[Pawan Kumar Ganjhu](https://pawankg.medium.com/?source=author_recirc-----21d4e3891f0f----1---------------------c73415e4_e3f6_44ef_8f42_54798ba50818-------)](https://pawankg.medium.com/?source=author_recirc-----21d4e3891f0f----1---------------------c73415e4_e3f6_44ef_8f42_54798ba50818-------)

[Pawan Kumar Ganjhu](https://pawankg.medium.com/?source=author_recirc-----21d4e3891f0f----1---------------------c73415e4_e3f6_44ef_8f42_54798ba50818-------)

in

[Towards Dev](https://towardsdev.com/?source=author_recirc-----21d4e3891f0f----1---------------------c73415e4_e3f6_44ef_8f42_54798ba50818-------)

**[Testing Data Pipelines: Python Examples for ETL Validation and Verification](https://pawankg.medium.com/testing-data-pipelines-python-examples-for-etl-validation-and-verification-431f482a8d47?source=author_recirc-----21d4e3891f0f----1---------------------c73415e4_e3f6_44ef_8f42_54798ba50818-------)**

[Testing data pipelines in an Extract, Transform, Load (ETL) process is crucial to ensure the accuracy and reliability of the data being…](https://pawankg.medium.com/testing-data-pipelines-python-examples-for-etl-validation-and-verification-431f482a8d47?source=author_recirc-----21d4e3891f0f----1---------------------c73415e4_e3f6_44ef_8f42_54798ba50818-------)

[4 min read·May 30](https://pawankg.medium.com/testing-data-pipelines-python-examples-for-etl-validation-and-verification-431f482a8d47?source=author_recirc-----21d4e3891f0f----1---------------------c73415e4_e3f6_44ef_8f42_54798ba50818-------)

115

[1](https://pawankg.medium.com/testing-data-pipelines-python-examples-for-etl-validation-and-verification-431f482a8d47?responsesOpen=true&sortBy=REVERSE_CHRON&source=author_recirc-----21d4e3891f0f----1---------------------c73415e4_e3f6_44ef_8f42_54798ba50818-------)

[[](https://pawankg.medium.com/exploring-pysparks-collection-types-a-comprehensive-guide-4a757ae0f6a8?source=author_recirc-----21d4e3891f0f----2---------------------c73415e4_e3f6_44ef_8f42_54798ba50818-------)](https://pawankg.medium.com/exploring-pysparks-collection-types-a-comprehensive-guide-4a757ae0f6a8?source=author_recirc-----21d4e3891f0f----2---------------------c73415e4_e3f6_44ef_8f42_54798ba50818-------)

[[Pawan Kumar Ganjhu](https://pawankg.medium.com/?source=author_recirc-----21d4e3891f0f----2---------------------c73415e4_e3f6_44ef_8f42_54798ba50818-------)](https://pawankg.medium.com/?source=author_recirc-----21d4e3891f0f----2---------------------c73415e4_e3f6_44ef_8f42_54798ba50818-------)

[Pawan Kumar Ganjhu](https://pawankg.medium.com/?source=author_recirc-----21d4e3891f0f----2---------------------c73415e4_e3f6_44ef_8f42_54798ba50818-------)

**[Exploring PySpark’s Collection Types: A Comprehensive Guide](https://pawankg.medium.com/exploring-pysparks-collection-types-a-comprehensive-guide-4a757ae0f6a8?source=author_recirc-----21d4e3891f0f----2---------------------c73415e4_e3f6_44ef_8f42_54798ba50818-------)**

[Stay tuned…. contents on the way](https://pawankg.medium.com/exploring-pysparks-collection-types-a-comprehensive-guide-4a757ae0f6a8?source=author_recirc-----21d4e3891f0f----2---------------------c73415e4_e3f6_44ef_8f42_54798ba50818-------)

[8 min read·Jun 14](https://pawankg.medium.com/exploring-pysparks-collection-types-a-comprehensive-guide-4a757ae0f6a8?source=author_recirc-----21d4e3891f0f----2---------------------c73415e4_e3f6_44ef_8f42_54798ba50818-------)

3

[[](https://pawankg.medium.com/exploring-sql-use-cases-case-statements-for-data-manipulation-and-analysis-26191b5534c9?source=author_recirc-----21d4e3891f0f----3---------------------c73415e4_e3f6_44ef_8f42_54798ba50818-------)](https://pawankg.medium.com/exploring-sql-use-cases-case-statements-for-data-manipulation-and-analysis-26191b5534c9?source=author_recirc-----21d4e3891f0f----3---------------------c73415e4_e3f6_44ef_8f42_54798ba50818-------)

[[Pawan Kumar Ganjhu](https://pawankg.medium.com/?source=author_recirc-----21d4e3891f0f----3---------------------c73415e4_e3f6_44ef_8f42_54798ba50818-------)](https://pawankg.medium.com/?source=author_recirc-----21d4e3891f0f----3---------------------c73415e4_e3f6_44ef_8f42_54798ba50818-------)

[Pawan Kumar Ganjhu](https://pawankg.medium.com/?source=author_recirc-----21d4e3891f0f----3---------------------c73415e4_e3f6_44ef_8f42_54798ba50818-------)

**[Exploring SQL Use Cases: CASE Statements for Data Manipulation and Analysis](https://pawankg.medium.com/exploring-sql-use-cases-case-statements-for-data-manipulation-and-analysis-26191b5534c9?source=author_recirc-----21d4e3891f0f----3---------------------c73415e4_e3f6_44ef_8f42_54798ba50818-------)**

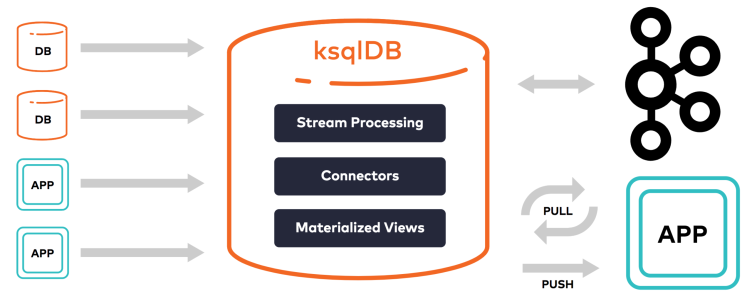
[CASE statements in SQL are versatile and can be used in various scenarios. Here are some common use cases for CASE statements:](https://pawankg.medium.com/exploring-sql-use-cases-case-statements-for-data-manipulation-and-analysis-26191b5534c9?source=author_recirc-----21d4e3891f0f----3---------------------c73415e4_e3f6_44ef_8f42_54798ba50818-------)

[5 min read·Jun 5](https://pawankg.medium.com/exploring-sql-use-cases-case-statements-for-data-manipulation-and-analysis-26191b5534c9?source=author_recirc-----21d4e3891f0f----3---------------------c73415e4_e3f6_44ef_8f42_54798ba50818-------)

2

[See all from Pawan Kumar Ganjhu](https://pawankg.medium.com/?source=post_page-----21d4e3891f0f--------------------------------)

Recommended from Medium

[[](https://medium.com/@thomas_reid/streaming-change-data-capture-cdc-data-between-databases-using-kafka-296b772e354?source=read_next_recirc-----21d4e3891f0f----0---------------------ea888665_65fb_414e_aa48_d4bf1b062307-------)](https://medium.com/@thomas_reid/streaming-change-data-capture-cdc-data-between-databases-using-kafka-296b772e354?source=read_next_recirc-----21d4e3891f0f----0---------------------ea888665_65fb_414e_aa48_d4bf1b062307-------)

[[Thomas Reid](https://medium.com/@thomas_reid?source=read_next_recirc-----21d4e3891f0f----0---------------------ea888665_65fb_414e_aa48_d4bf1b062307-------)](https://medium.com/@thomas_reid?source=read_next_recirc-----21d4e3891f0f----0---------------------ea888665_65fb_414e_aa48_d4bf1b062307-------)

[Thomas Reid](https://medium.com/@thomas_reid?source=read_next_recirc-----21d4e3891f0f----0---------------------ea888665_65fb_414e_aa48_d4bf1b062307-------)

in

[Level Up Coding](https://levelup.gitconnected.com/?source=read_next_recirc-----21d4e3891f0f----0---------------------ea888665_65fb_414e_aa48_d4bf1b062307-------)

**[Streaming change data capture (CDC) data between databases using Kafka](https://medium.com/@thomas_reid/streaming-change-data-capture-cdc-data-between-databases-using-kafka-296b772e354?source=read_next_recirc-----21d4e3891f0f----0---------------------ea888665_65fb_414e_aa48_d4bf1b062307-------)**

[This article will briefly introduce Kafka, how to connect database sources to it using the Kafka SQL client ksqlDB and create and…](https://medium.com/@thomas_reid/streaming-change-data-capture-cdc-data-between-databases-using-kafka-296b772e354?source=read_next_recirc-----21d4e3891f0f----0---------------------ea888665_65fb_414e_aa48_d4bf1b062307-------)

[·21 min read·Mar 27](https://medium.com/@thomas_reid/streaming-change-data-capture-cdc-data-between-databases-using-kafka-296b772e354?source=read_next_recirc-----21d4e3891f0f----0---------------------ea888665_65fb_414e_aa48_d4bf1b062307-------)

71

[[](https://medium.com/@ayyoubmaulana/how-to-build-your-own-apache-airflow-dag-factory-8ad895423881?source=read_next_recirc-----21d4e3891f0f----1---------------------ea888665_65fb_414e_aa48_d4bf1b062307-------)](https://medium.com/@ayyoubmaulana/how-to-build-your-own-apache-airflow-dag-factory-8ad895423881?source=read_next_recirc-----21d4e3891f0f----1---------------------ea888665_65fb_414e_aa48_d4bf1b062307-------)

[[Ayyoub Maulana](https://medium.com/@ayyoubmaulana?source=read_next_recirc-----21d4e3891f0f----1---------------------ea888665_65fb_414e_aa48_d4bf1b062307-------)](https://medium.com/@ayyoubmaulana?source=read_next_recirc-----21d4e3891f0f----1---------------------ea888665_65fb_414e_aa48_d4bf1b062307-------)

[Ayyoub Maulana](https://medium.com/@ayyoubmaulana?source=read_next_recirc-----21d4e3891f0f----1---------------------ea888665_65fb_414e_aa48_d4bf1b062307-------)

in

[Data Engineering Indonesia](https://medium.com/data-engineering-indonesia?source=read_next_recirc-----21d4e3891f0f----1---------------------ea888665_65fb_414e_aa48_d4bf1b062307-------)

**[How to Build Your Own Apache Airflow DAG Factory](https://medium.com/@ayyoubmaulana/how-to-build-your-own-apache-airflow-dag-factory-8ad895423881?source=read_next_recirc-----21d4e3891f0f----1---------------------ea888665_65fb_414e_aa48_d4bf1b062307-------)**

[This DAG Factory will utilize yaml file to define a code based on configuration, it will hide the complex logic behind it](https://medium.com/@ayyoubmaulana/how-to-build-your-own-apache-airflow-dag-factory-8ad895423881?source=read_next_recirc-----21d4e3891f0f----1---------------------ea888665_65fb_414e_aa48_d4bf1b062307-------)

[5 min read·4 days ago](https://medium.com/@ayyoubmaulana/how-to-build-your-own-apache-airflow-dag-factory-8ad895423881?source=read_next_recirc-----21d4e3891f0f----1---------------------ea888665_65fb_414e_aa48_d4bf1b062307-------)

8

Lists

[[](https://pawankg.medium.com/list/newreadinglist-d8e941a8265f?source=read_next_recirc-----21d4e3891f0f--------------------------------)](https://pawankg.medium.com/list/newreadinglist-d8e941a8265f?source=read_next_recirc-----21d4e3891f0f--------------------------------)

[[](https://pawankg.medium.com/list/newreadinglist-d8e941a8265f?source=read_next_recirc-----21d4e3891f0f--------------------------------)](https://pawankg.medium.com/list/newreadinglist-d8e941a8265f?source=read_next_recirc-----21d4e3891f0f--------------------------------)

[[](https://pawankg.medium.com/list/newreadinglist-d8e941a8265f?source=read_next_recirc-----21d4e3891f0f--------------------------------)](https://pawankg.medium.com/list/newreadinglist-d8e941a8265f?source=read_next_recirc-----21d4e3891f0f--------------------------------)

[New\_Reading\_List](https://pawankg.medium.com/list/newreadinglist-d8e941a8265f?source=read_next_recirc-----21d4e3891f0f--------------------------------)

[174 stories·18 saves](https://pawankg.medium.com/list/newreadinglist-d8e941a8265f?source=read_next_recirc-----21d4e3891f0f--------------------------------)

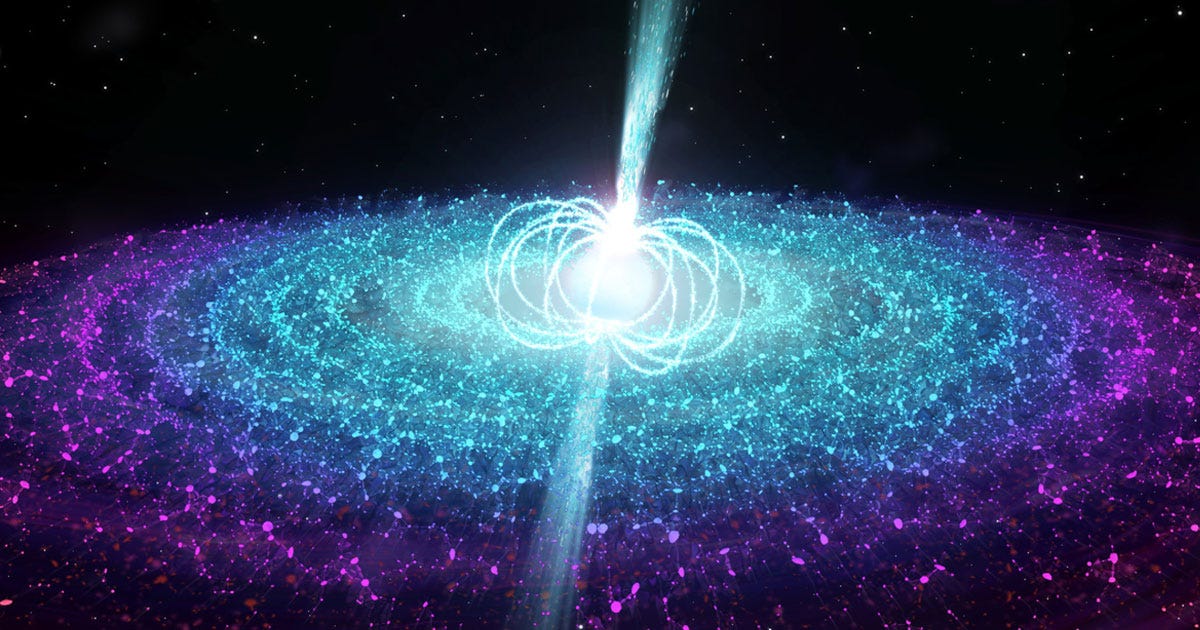
[[](https://pawankg.medium.com/list/natural-language-processing-0a856388a93a?source=read_next_recirc-----21d4e3891f0f--------------------------------)](https://pawankg.medium.com/list/natural-language-processing-0a856388a93a?source=read_next_recirc-----21d4e3891f0f--------------------------------)

[[](https://pawankg.medium.com/list/natural-language-processing-0a856388a93a?source=read_next_recirc-----21d4e3891f0f--------------------------------)](https://pawankg.medium.com/list/natural-language-processing-0a856388a93a?source=read_next_recirc-----21d4e3891f0f--------------------------------)

[[](https://pawankg.medium.com/list/natural-language-processing-0a856388a93a?source=read_next_recirc-----21d4e3891f0f--------------------------------)](https://pawankg.medium.com/list/natural-language-processing-0a856388a93a?source=read_next_recirc-----21d4e3891f0f--------------------------------)

[Natural Language Processing](https://pawankg.medium.com/list/natural-language-processing-0a856388a93a?source=read_next_recirc-----21d4e3891f0f--------------------------------)

[385 stories·43 saves](https://pawankg.medium.com/list/natural-language-processing-0a856388a93a?source=read_next_recirc-----21d4e3891f0f--------------------------------)

[[](https://pawankg.medium.com/pyspark-for-data-engineering-beginners-an-extensive-guide-ce29520a78ca?source=read_next_recirc-----21d4e3891f0f----0---------------------ea888665_65fb_414e_aa48_d4bf1b062307-------)](https://pawankg.medium.com/pyspark-for-data-engineering-beginners-an-extensive-guide-ce29520a78ca?source=read_next_recirc-----21d4e3891f0f----0---------------------ea888665_65fb_414e_aa48_d4bf1b062307-------)

[[Pawan Kumar Ganjhu](https://pawankg.medium.com/?source=read_next_recirc-----21d4e3891f0f----0---------------------ea888665_65fb_414e_aa48_d4bf1b062307-------)](https://pawankg.medium.com/?source=read_next_recirc-----21d4e3891f0f----0---------------------ea888665_65fb_414e_aa48_d4bf1b062307-------)

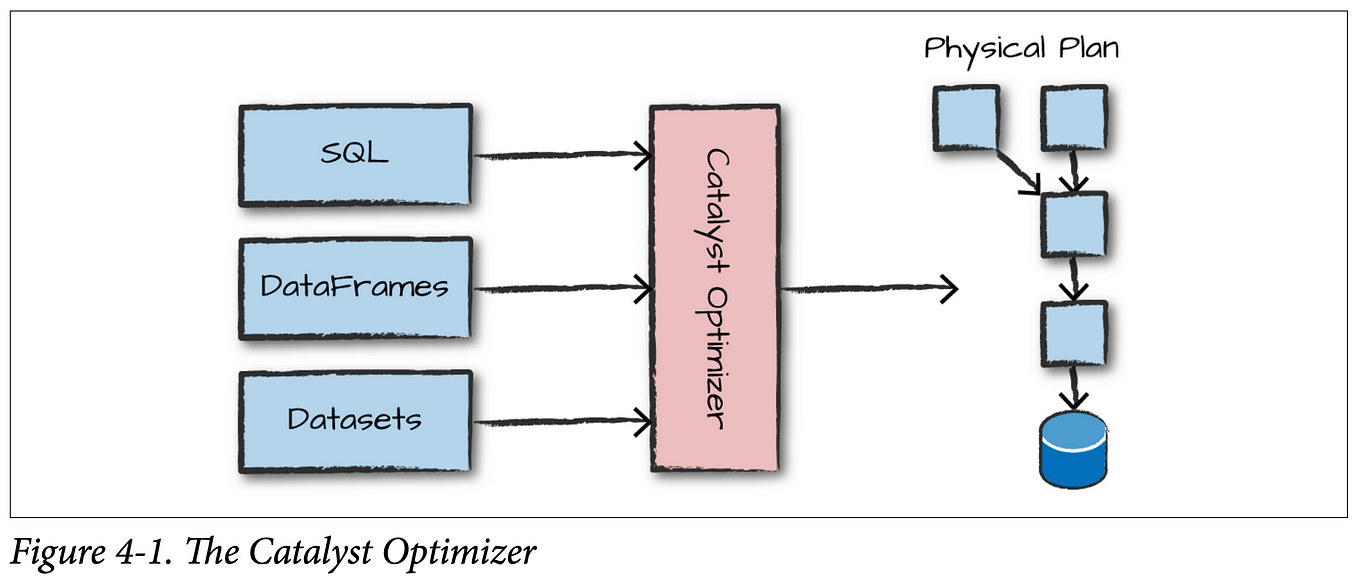
[Pawan Kumar Ganjhu](https://pawankg.medium.com/?source=read_next_recirc-----21d4e3891f0f----0---------------------ea888665_65fb_414e_aa48_d4bf1b062307-------)

**[PySpark for Data Engineering Beginners: An Extensive Guide](https://pawankg.medium.com/pyspark-for-data-engineering-beginners-an-extensive-guide-ce29520a78ca?source=read_next_recirc-----21d4e3891f0f----0---------------------ea888665_65fb_414e_aa48_d4bf1b062307-------)**

[Introduction](https://pawankg.medium.com/pyspark-for-data-engineering-beginners-an-extensive-guide-ce29520a78ca?source=read_next_recirc-----21d4e3891f0f----0---------------------ea888665_65fb_414e_aa48_d4bf1b062307-------)

[10 min read·May 10](https://pawankg.medium.com/pyspark-for-data-engineering-beginners-an-extensive-guide-ce29520a78ca?source=read_next_recirc-----21d4e3891f0f----0---------------------ea888665_65fb_414e_aa48_d4bf1b062307-------)

15

[[](https://medium.com/@balahari619/structured-api-execution-in-spark-4dabde8d0c6c?source=read_next_recirc-----21d4e3891f0f----1---------------------ea888665_65fb_414e_aa48_d4bf1b062307-------)](https://medium.com/@balahari619/structured-api-execution-in-spark-4dabde8d0c6c?source=read_next_recirc-----21d4e3891f0f----1---------------------ea888665_65fb_414e_aa48_d4bf1b062307-------)

[[Bala Hari](https://medium.com/@balahari619?source=read_next_recirc-----21d4e3891f0f----1---------------------ea888665_65fb_414e_aa48_d4bf1b062307-------)](https://medium.com/@balahari619?source=read_next_recirc-----21d4e3891f0f----1---------------------ea888665_65fb_414e_aa48_d4bf1b062307-------)

[Bala Hari](https://medium.com/@balahari619?source=read_next_recirc-----21d4e3891f0f----1---------------------ea888665_65fb_414e_aa48_d4bf1b062307-------)

**[Structured API Execution in Spark](https://medium.com/@balahari619/structured-api-execution-in-spark-4dabde8d0c6c?source=read_next_recirc-----21d4e3891f0f----1---------------------ea888665_65fb_414e_aa48_d4bf1b062307-------)**

[Understand the Structured API Execution in Spark](https://medium.com/@balahari619/structured-api-execution-in-spark-4dabde8d0c6c?source=read_next_recirc-----21d4e3891f0f----1---------------------ea888665_65fb_414e_aa48_d4bf1b062307-------)

[3 min read·4 days ago](https://medium.com/@balahari619/structured-api-execution-in-spark-4dabde8d0c6c?source=read_next_recirc-----21d4e3891f0f----1---------------------ea888665_65fb_414e_aa48_d4bf1b062307-------)

[[](https://medium.com/@shahrukhshl0/spark-data-engineering-project-8fb125fe68c7?source=read_next_recirc-----21d4e3891f0f----2---------------------ea888665_65fb_414e_aa48_d4bf1b062307-------)](https://medium.com/@shahrukhshl0/spark-data-engineering-project-8fb125fe68c7?source=read_next_recirc-----21d4e3891f0f----2---------------------ea888665_65fb_414e_aa48_d4bf1b062307-------)

[[Shahrukh Sohail](https://medium.com/@shahrukhshl0?source=read_next_recirc-----21d4e3891f0f----2---------------------ea888665_65fb_414e_aa48_d4bf1b062307-------)](https://medium.com/@shahrukhshl0?source=read_next_recirc-----21d4e3891f0f----2---------------------ea888665_65fb_414e_aa48_d4bf1b062307-------)

[Shahrukh Sohail](https://medium.com/@shahrukhshl0?source=read_next_recirc-----21d4e3891f0f----2---------------------ea888665_65fb_414e_aa48_d4bf1b062307-------)

**[Spark Data Engineering Project](https://medium.com/@shahrukhshl0/spark-data-engineering-project-8fb125fe68c7?source=read_next_recirc-----21d4e3891f0f----2---------------------ea888665_65fb_414e_aa48_d4bf1b062307-------)**

[Introduction: Embarking on an Emotional Journey](https://medium.com/@shahrukhshl0/spark-data-engineering-project-8fb125fe68c7?source=read_next_recirc-----21d4e3891f0f----2---------------------ea888665_65fb_414e_aa48_d4bf1b062307-------)

[4 min read·Jul 3](https://medium.com/@shahrukhshl0/spark-data-engineering-project-8fb125fe68c7?source=read_next_recirc-----21d4e3891f0f----2---------------------ea888665_65fb_414e_aa48_d4bf1b062307-------)

9

[[](https://romanglushach.medium.com/etl-vs-elt-choosing-the-right-approach-for-your-data-pipeline-eb9af0bc0993?source=read_next_recirc-----21d4e3891f0f----3---------------------ea888665_65fb_414e_aa48_d4bf1b062307-------)](https://romanglushach.medium.com/etl-vs-elt-choosing-the-right-approach-for-your-data-pipeline-eb9af0bc0993?source=read_next_recirc-----21d4e3891f0f----3---------------------ea888665_65fb_414e_aa48_d4bf1b062307-------)

[[Roman Glushach](https://romanglushach.medium.com/?source=read_next_recirc-----21d4e3891f0f----3---------------------ea888665_65fb_414e_aa48_d4bf1b062307-------)](https://romanglushach.medium.com/?source=read_next_recirc-----21d4e3891f0f----3---------------------ea888665_65fb_414e_aa48_d4bf1b062307-------)

[Roman Glushach](https://romanglushach.medium.com/?source=read_next_recirc-----21d4e3891f0f----3---------------------ea888665_65fb_414e_aa48_d4bf1b062307-------)

**[ETL vs ELT: Choosing the Right Approach for Your Data Pipeline](https://romanglushach.medium.com/etl-vs-elt-choosing-the-right-approach-for-your-data-pipeline-eb9af0bc0993?source=read_next_recirc-----21d4e3891f0f----3---------------------ea888665_65fb_414e_aa48_d4bf1b062307-------)**

[As data becomes increasingly important for businesses, it’s crucial to have an efficient data pipeline that can extract, transform, and…](https://romanglushach.medium.com/etl-vs-elt-choosing-the-right-approach-for-your-data-pipeline-eb9af0bc0993?source=read_next_recirc-----21d4e3891f0f----3---------------------ea888665_65fb_414e_aa48_d4bf1b062307-------)

[4 min read·Jun 19](https://romanglushach.medium.com/etl-vs-elt-choosing-the-right-approach-for-your-data-pipeline-eb9af0bc0993?source=read_next_recirc-----21d4e3891f0f----3---------------------ea888665_65fb_414e_aa48_d4bf1b062307-------)

5

[See more recommendations](https://medium.com/?source=post_page-----21d4e3891f0f--------------------------------)

[Help](https://help.medium.com/hc/en-us?source=post_page-----21d4e3891f0f--------------------------------)

[Status](https://medium.statuspage.io/?source=post_page-----21d4e3891f0f--------------------------------)

[Writers](https://about.medium.com/creators/?source=post_page-----21d4e3891f0f--------------------------------)

[Blog](https://blog.medium.com/?source=post_page-----21d4e3891f0f--------------------------------)

[Careers](https://medium.com/jobs-at-medium/work-at-medium-959d1a85284e?source=post_page-----21d4e3891f0f--------------------------------)

[Privacy](https://policy.medium.com/medium-privacy-policy-f03bf92035c9?source=post_page-----21d4e3891f0f--------------------------------)

[Terms](https://policy.medium.com/medium-terms-of-service-9db0094a1e0f?source=post_page-----21d4e3891f0f--------------------------------)

[About](https://medium.com/about?autoplay=1&source=post_page-----21d4e3891f0f--------------------------------)

[Text to speech](https://speechify.com/medium?source=post_page-----21d4e3891f0f--------------------------------)

[Teams](https://medium.com/business?source=post_page-----21d4e3891f0f--------------------------------)