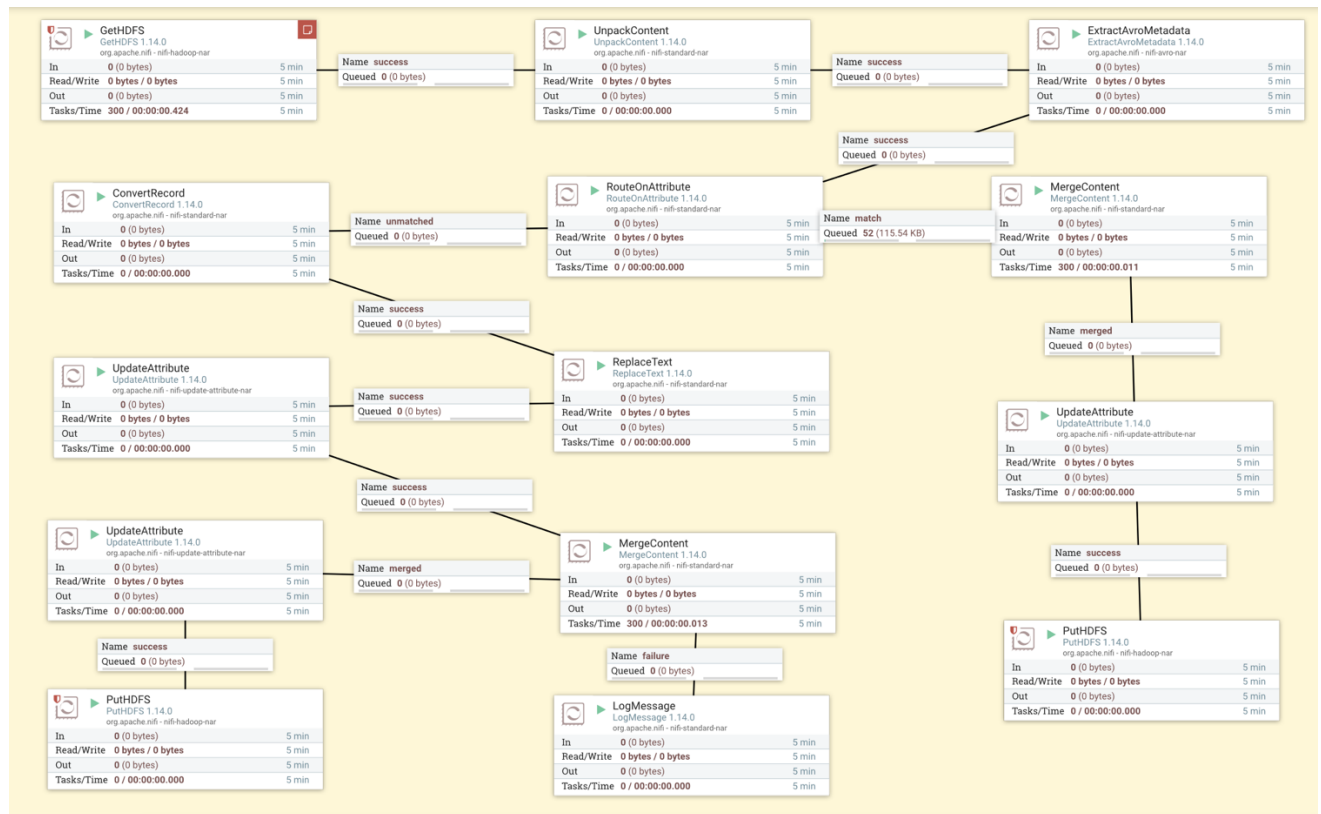
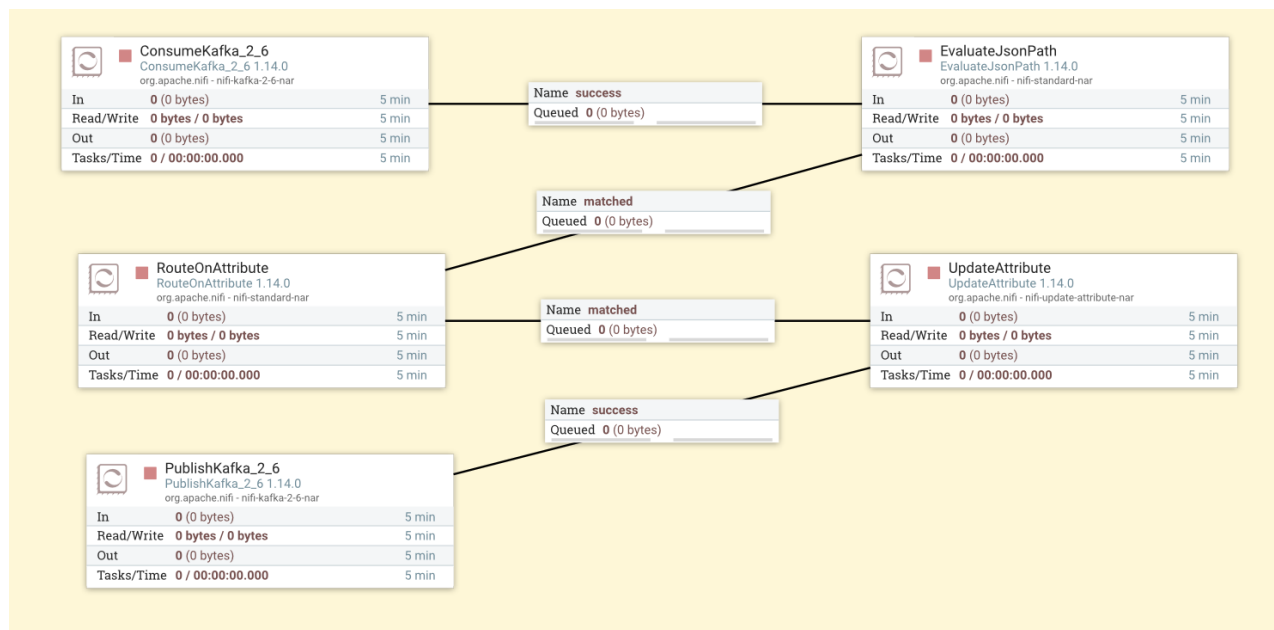


1. HDFS Data Processing and Transformation



I start by retrieving data from HDFS using the GetHDFS processor, which allows me to access and read the input files from a specified directory. I then use ConvertRecord to transform the format of the data, such as converting JSON or Avro to CSV, enabling downstream compatibility. If the data is compressed or packaged, I use UnpackContent to extract the raw content. Next, I apply RouteOnAttribute to filter data based on logical conditions or attribute values, ensuring only relevant records proceed. Using ReplaceText, I modify text-based fields in the dataset as needed, preparing them for further processing. To optimize storage, I combine smaller files into larger chunks using MergeContent, making the output more efficient to write and read. I update metadata and assign custom attributes to the data using UpdateAttribute, tailoring it to the specific requirements of the flow. At the end, I save the processed and aggregated data back into HDFS using the PutHDFS processor. Finally, I leverage LogMessage to monitor the process and ensure all operations are running smoothly, logging key events for troubleshooting or validation.

2. Kafka Stream Processing for IoT Alerts



I begin by subscribing to a Kafka topic named `iotinfo` using the `ConsumeKafka_2_6` processor to stream incoming messages in real time. Once I retrieve the data, I extract specific fields from the JSON payload, such as `timestamp`, `co`, and `smoke`, using the `EvaluateJsonPath` processor. I then filter these records using `RouteOnAttribute`, ensuring that only messages exceeding thresholds for `co` and `smoke` (e.g., 0.005 and 0.02, respectively) are passed forward. To improve readability, I use `UpdateAttribute` to convert timestamps into a human-readable date and time format. After processing, I publish the filtered alerts to a new Kafka topic named `alerts` using the `PublishKafka_2_6` processor. This allows me to redirect important notifications to another system for further action. Throughout the process, I ensure the flow is optimized for low-latency handling of IoT messages, enabling me to monitor environmental conditions dynamically and react quickly to any issues or anomalies.