Enable Dynamic Topic Destinations Using Java pub/subIO writeMessageDynamic() in Python pipelines

Olu Akinlaja

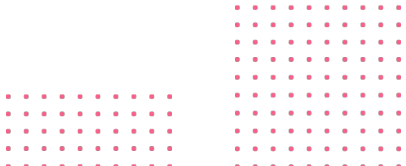
BEAM SUMMIT

September 4-5, 2024

Sunnyvale, CA. USA

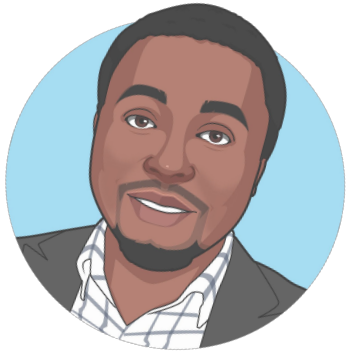
Agenda

1. Introduction
    1. About Myself
    2. What do we do at doit International?
2. Use-case background information
    1. write to a single Cloud pub/sub topic
3. What is Dynamic Topic Destinations in Cloud pub/sub?
    1. Alternatives to Dynamic Topic Destinations
    2. Benefits of the Dynamic Topic Destinations
4. Implementation Dynamic topic destination
    1. PubSubIO.writeMessagesDynamic() as an external transform
5. Demo
6. BENEFITS of MULTI Language Pipelines:
7. Resources



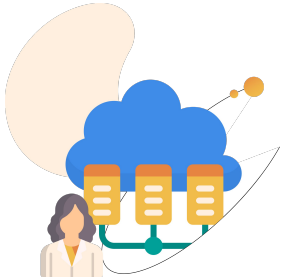
About Myself

- Data & Cloud Enthusiast
- Based in Montreal, Canada
- Data Engineer with DoIT International
  - Primary Focus in Data processing in GCP
    - Data pipelining with Apache Beam
  - Collaborate on Projects related to Google's GenAI applications
    - Deliver workshops relating to these.



Olu Akinlaja
Data Engineer
ww.linkedin.com/in/oakinlaja

What do we do at doit International?



CONSULTING

Let's get those projects moving

Dedicated senior customer reliability engineers, based on specialization and customer needs.

TRAINING

Build that knowledge

With more than +150 certifications across GCP, AWS and Azure, our engineers deliver world-class **instructor-led training.**

UNLIMITED SUPPORT

There for you from the start

Our design guidance, DevOps enablement, and platform support is second to none.

Follow the sun support with **zero access to customer data.**

Simple and straightforward and all at next to zero costs

Background information

USE-CASE

- Python Real-time Pipeline running on Cloud Dataflow
  - Data Ingestion through the Google Cloud pub/sub
  - Pub/Sub ingests data from multiple data sources
    - messages being published to multiple topics
    - based on particular message attributes in real-time
  - The necessity to handle this in real-time

write to a single Cloud pub/sub topic

Java SDK

```
public static void main(String[] args) {
  var options = PipelineOptionsFactory.fromArgs(args).withValidation().as(Options.class);
  var pipeline = Pipeline.create(options);
  pipeline
      // Create some data to write to Pub/Sub; the variable messages below being the message payload
      .apply(Create.of(messages))
      // Convert the data to Pub/Sub messages.
      .apply(MapElements
          .into(TypeDescriptor.of(PubsubMessage.class))
          .via((message -> {
            byte[] payload = message.product.getBytes(StandardCharsets.UTF_8);
            // Create attributes for each message.
            HashMap<String, String> attributes = new HashMap<String, String>();
            attributes.put("buyer", message.name);
            attributes.put("timestamp", Long.toString(message.timestamp));
            return new PubsubMessage(payload, attributes);
          })))
      // Write the messages to Pub/Sub.
      .apply(PubsubIO.writeMessages().to(options.getTopic()));
  pipeline.run().waitUntilFinish();
 }
```

write to a single Cloud pub/sub topic

Python SDK

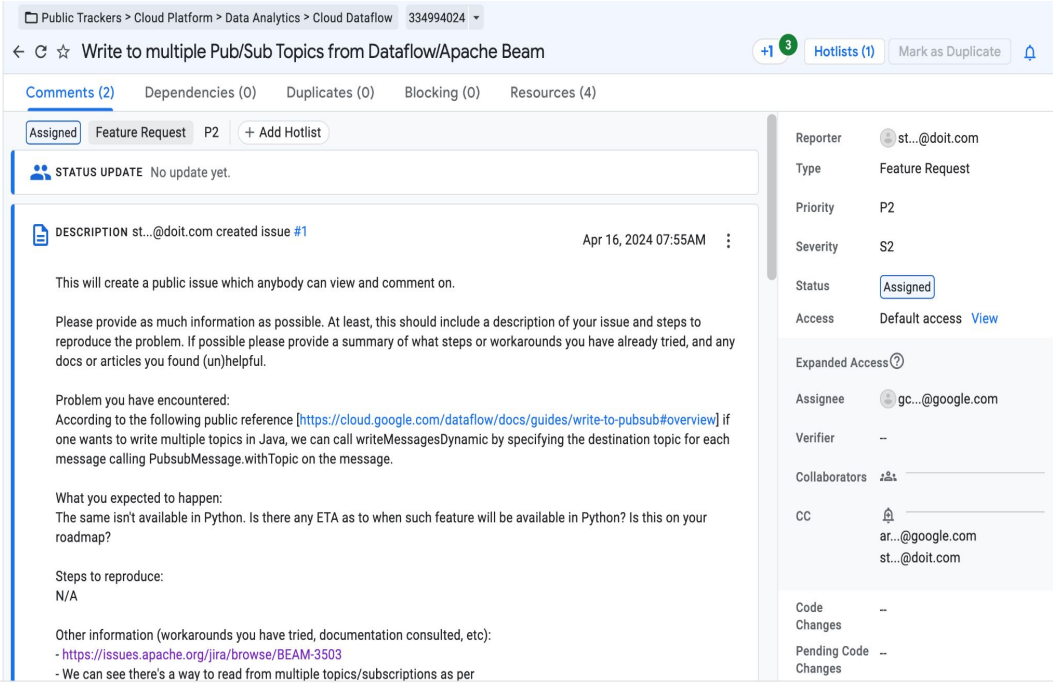
```
with beam.Pipeline(options=options) as pipeline:
    (
        pipeline
        | "Create elements" >> beam.Create(example_data)
        | "Convert to Pub/Sub messages" >> beam.Map(item_to_message)
        | WriteToPubSub(
            topic=options.topic,
            with_attributes=True)
    )

print('Pipeline ran successfully.')

```

What is Dynamic Topic Destinations in Cloud pub/sub?

- This feature allows publishing pub/sub messages to multiple cloud pub/sub topics, based on particular message attributes in real-time.
- With this feature, it is possible to use a single publisher client to dictate which messages go to which topics
- Java SDK
  - pubsubIO.writeMessagesDynamic() method implements Dynamic Topic Destinations on the Dataflow Runner
- Python SDK
  - No support presently for this feature yet.
  - Beam-only fix is insufficient



Alternatives to Dynamic Topic Destinations?

- Topic Partitions:
  - predefined set of categories for the pub/sub messages
  - Feature available in cloud pub/sub lite
    - Cloud pub/sub lite is deprecated
    - Existing users: Pub/Sub Lite remains functional until March 18, 2026.
    - If Pub/Sub Lite is not used before September 24, 2024
      - No new access to Pub/Sub Lite would be permitted.
    - Consider the Google Cloud Managed Service for Apache Kafka as an alternative
- Streaming Engine Routing:
  - Streaming Engines like Apache Flink or Apache Beam offer message routing capabilities within the processing pipeline itself.

Benefits of the Dynamic Topic Destination in Cloud pub/sub

1. SIMPLIFIED MANAGEMENT:
   - use a single publisher client to send messages to various topics based on message attributes
2. IMPROVED ORGANIZATION:
   - Data segregation based on message content leads to cleaner topic structures and easier downstream processing
3. ENHANCED SECURITY:
   - By directing messages to specific topics, it is easier to enforce granular access controls on the data.

Implementation Dynamic topic destination

FUNCTION EXTRACT METHOD

```
avros.apply(PubsubIO.writeAvros(MyType.class).
    to((ValueInSingleWindow<Event> quote) -> {
        String country = quote.getCountry();
        return "projects/myproject/topics/events_" + country;
       });
```

For more details about this approach and the code details, please refer to the Introducing dynamic topic destinations in Pub/Sub using Dataflow.

Implementation Dynamic topic destination(Cont.)

PubSubIO.writeMessagesDynamic Method

```
events.apply(MapElements.into(new TypeDescriptor<PubsubMessage>() {})
                .via(e -> new PubsubMessage(
                    e.toByteString(), Collections.emptyMap()).withTopic(e.getCountry())))
            .apply(PubsubIO.writeMessagesDynamic());
```

For more details about this approach and the code details, please refer to the Introducing dynamic topic destinations in Pub/Sub using Dataflow.

PubSubIO.writeMessagesDynamic() as an external transform

- Main Java Class
  - Defines the core logic
    - In this use-case:
      - handles reading the PCollection<String> as input
      - processing each string to create a PubsubMessage
      - Writing pub/sub messages dynamically to multiple pub/sub topic
        - using PubsubIO.writeMessagesDynamic() method
- Configuration.Java Class
  - Serves the configuration settings to be used by DynamicBuilder.Java
  - This can include various configuration options, allowing for greater flexibility and customization of the transform's behavior.
    - In this use-case:
      - No specific configuration required.

PubSubIO.writeMessagesDynamic() as an external transform (Cont.)

- Registrar.Java
  - Implements an ExternalTransformRegistrar Java Interface
  - registers the Main Java transform with a unique URN, used to define the Environment to be used
    - Environments for executing Beam UDFs(such as DoFn, CombineFn)
    - Environments are chosen by the Beam Runners
    - An Environment would typically consist of :
      - A URN – which defines the type of environment and
      - A Payload – which are parameters that uniquely identify the environment.

- Builder.Java Class
  - Implements an ExternalTransformBuilder Java Interface
  - Uses the configuration from the Configuration Java object to configure and build the transform.

Quick Demo

BEAM SUMMIT

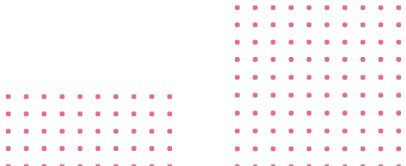
BENEFITS of MULTI Language Pipelines:

Reduced Cost of Software Development

- Develop once and offer to all SDK Languages
- I/O Connectors can be easily shared
- Easier to share codes between development teams

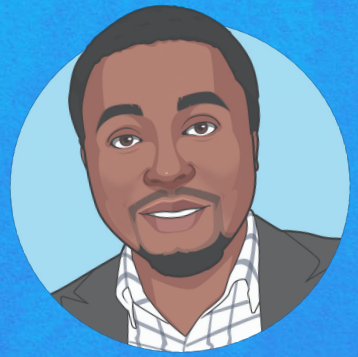
Reduced maintenance overheads

- No more multiple implementations of Complex transforms
- Evolve development teams without re-implementing
- Easily use transforms developed by third parties as you'd have more flexibility
- Uniform User experience when using multiple SDKs



Thank you!

Questions?



WebsiteURL: https://www.doit.com/contact/

LinkedIn: www.linkedin.com/in/oakinlaja

Twitter: https://twitter.com/Olusayo_

BEAM SUMMIT

Resources

Python multi language pipelines

Multi-Language Pipelines

Beam Summit 2023 | Multi-language pipelines - Chamikara Jayalath

Beam Learning: Using Java transforms in a multi language Python pipeline

Multi-language pipelines with Apache Beam

The pub/sub IO write Messages Dynamic

GCP Feature Request: Dynamic Topic Destination Function in Python SDK