The Write-Audit-Publish Pattern via Apache Iceberg

Samuel Redai - Developer Advocate, Tabular
Subsurface Live 2022

Samuel Redai - Developer Advocate, Tabular

Twitter: @samuelredai

GitHub: samredai

Email: sam@tabular.io

Tabular

Data Quality

Tabular

Data Quality

What exactly is “data quality”? Wikipedia tells us:

Data Quality

What exactly is “data quality”? Wikipedia tells us:

- “...People's views on data quality can often be in disagreement, even when discussing the same set of data used for the same purpose…”

Data Quality

What exactly is “data quality”? Wikipedia tells us:

- “...People's views on data quality can often be in disagreement, even when discussing the same set of data used for the same purpose…”
- “...as the number of data sources increases, the question of internal data consistency becomes significant, regardless of fitness for use for any particular external purpose…”

Data Quality

What exactly is “data quality”? Wikipedia tells us:

- “...People's views on data quality can often be in disagreement, even when discussing the same set of data used for the same purpose…”
- “...as the number of data sources increases, the question of internal data consistency becomes significant, regardless of fitness for use for any particular external purpose…”
- “...Defining data quality in a sentence is difficult…”

Tabular

Data Quality

What exactly is “data quality”? Wikipedia tells us:

- “...People's v in disagreement, even when di ed for the same purpose…”
- “...as the num e question of internal data consiste ess of fitness for use for any particular external purpose…”
- “...Defining data quality in a sentence is difficult…”

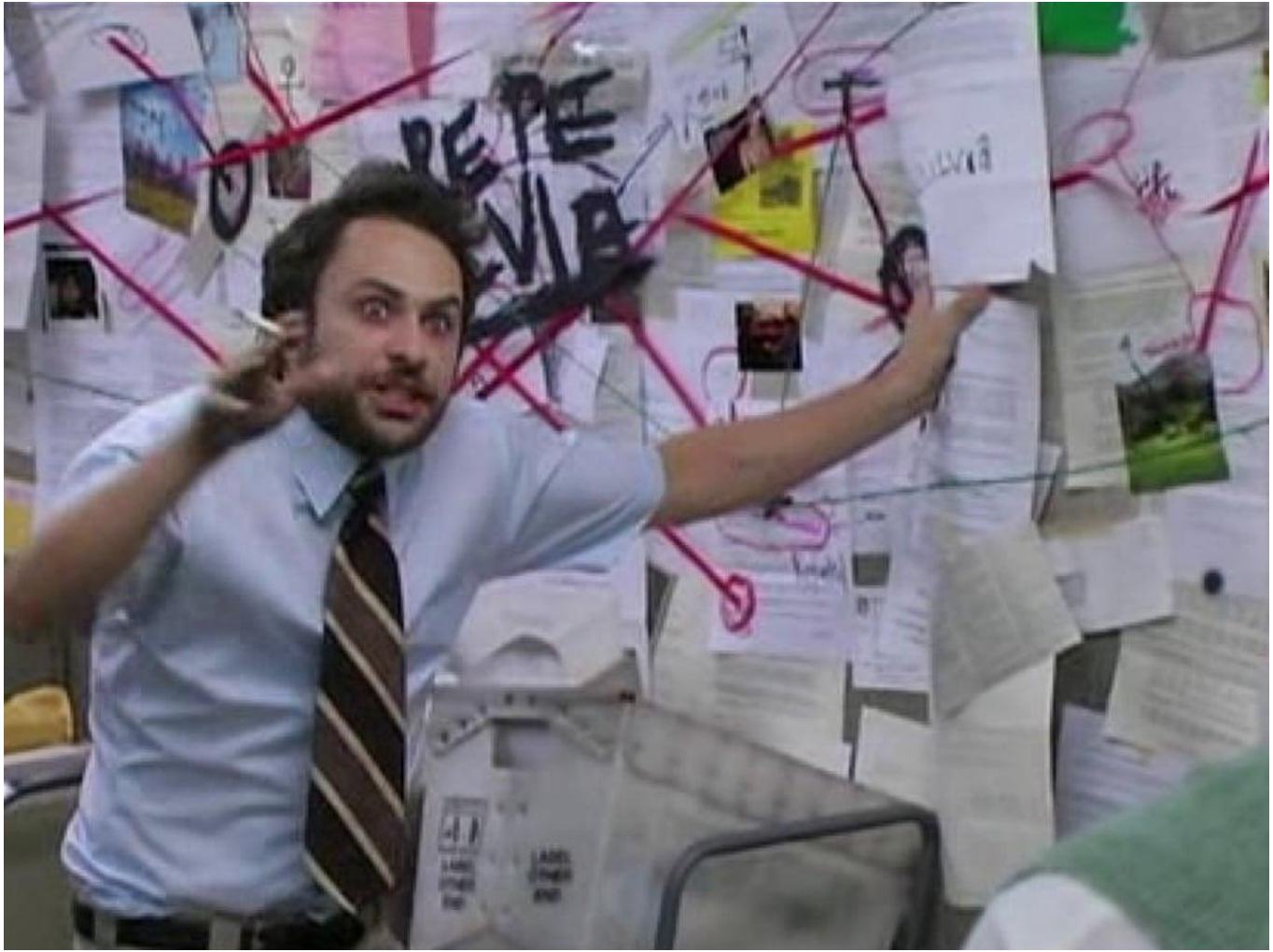
Tabular

How can you instill trust in your data?

Write your data to production and leave it to your consumers to run validations?

Write the data somewhere else first and make sure it looks good before rewriting it to production?

Generate data quality metrics as part of your pipeline and offer them as a reference?



Apache Iceberg Integrated Audits

Tabular

What is Apache Iceberg?

“Iceberg is a high-performance format for huge analytic tables. Iceberg brings the reliability and simplicity of SQL tables to big data, while making it possible for engines like Spark, Trino, Flink, Presto, and Hive to safely work with the same tables, at the same time.”

- iceberg.apache.org

Tabular

Apache Iceberg's Integrated Audits Feature – Overview

- Allows writing data to production in an **unpublished** state

Apache Iceberg's Integrated Audits Feature – Overview

- Allows writing data to production in an **unpublished** state
- Automatically tags the unpublished data with the `**spark.wap.id**` value from your spark session

Apache Iceberg's Integrated Audits Feature – Overview

- Allows writing data to production in an **unpublished** state
- Automatically tags the unpublished data with the `**spark.wap.id**` value from your spark session
- Using **Time Travel**, you can run *SELECT* queries against the snapshot of the unpublished data

Apache Iceberg's Integrated Audits Feature – Overview

- Allows writing data to production in an **unpublished** state
- Automatically tags the unpublished data with the `**spark.wap.id**` value from your spark session
- Using **Time Travel**, you can run *SELECT* queries against the snapshot of the unpublished data
- Once you have confidence in the data, publishing is a simple **metadata-only** operation via a **cherry-pick** operation

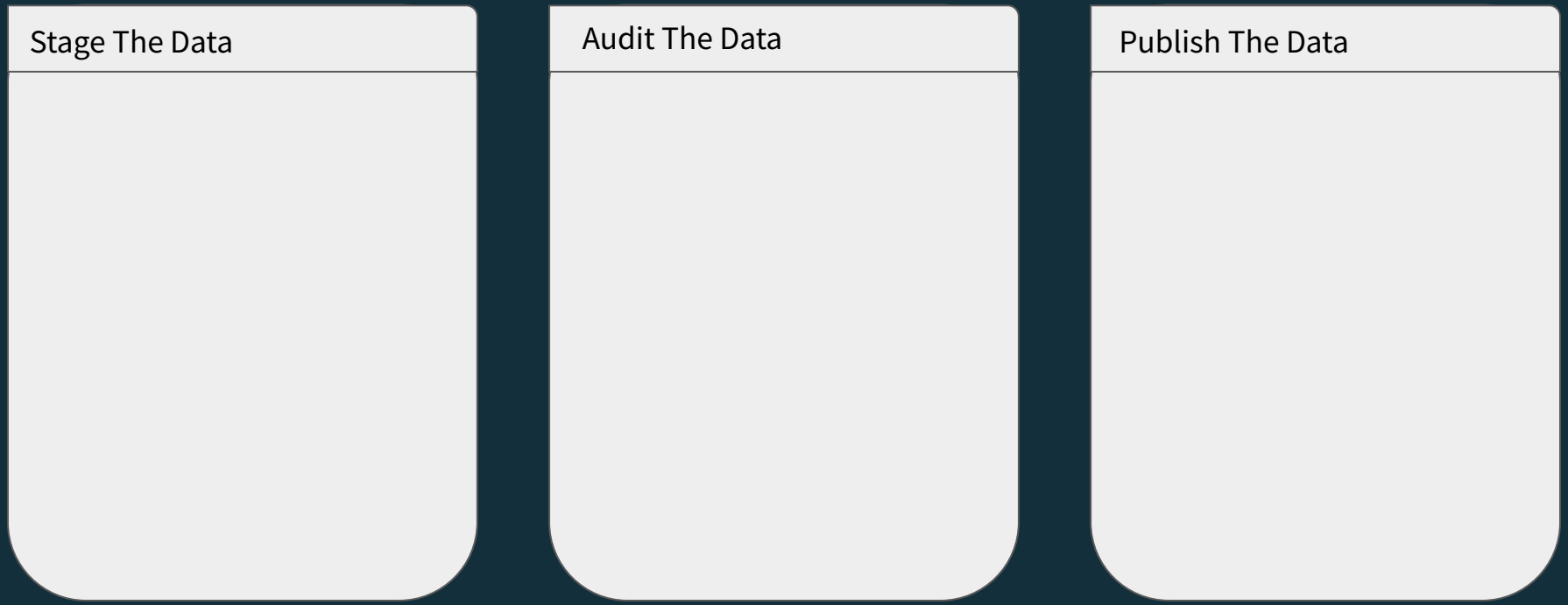
Tabular

Apache Iceberg's Integrated Audits Feature – Overview

- Allows writing data to production in an **unpublished** state
- Automatically tags the unpublished data with the `**spark.wap.id**` value from your spark session
- Using **Time Travel**, you can run *SELECT* queries against the snapshot of the unpublished data
- Once you have confidence in the data, publishing is a simple **metadata-only** operation via a **cherry-pick** operation
- If the data doesn't look good, just forget about it! Iceberg's **snapshot expiration** process will clean it up

Tabular

Apache Iceberg's Integrated Audits – More Details



Tabular

Apache Iceberg's Integrated Audits – More Details

Stage The Data	Audit The Data	Publish The Data
• Set **write.wap.enabled=true** on your table		

Tabular

Apache Iceberg's Integrated Audits – More Details

Stage The Data	Audit The Data	Publish The Data
• Set **write.wap.enabled=true** on your table • Set **spark.wap.id=<UUID>** in your Spark session conf		

Tabular

Apache Iceberg's Integrated Audits – More Details

Stage The Data	Audit The Data	Publish The Data
• Set **write.wap.enabled=true** on your table • Set **spark.wap.id=<UUID>** in your Spark session conf • Run your **production** ETL code		

Tabular

Apache Iceberg's Integrated Audits – More Details

Stage The Data

- Set **write.wap.enabled=true** on your table
- Set **spark.wap.id=<UUID>** in your Spark session conf
- Run your **production** ETL code

Audit The Data

- Find the snapshot ID from your **production** table's metadata that's tagged with the same **spark.wap.id**

Publish The Data

Tabular

Apache Iceberg's Integrated Audits – More Details

Stage The Data

- Set **write.wap.enabled=true** on your table
- Set **spark.wap.id=<UUID>** in your Spark session conf
- Run your **production** ETL code

Audit The Data

- Find the snapshot ID from your **production** table's metadata that's tagged with the same **spark.wap.id**
- Perform validations against this data (using ***any*** auditing tool or framework)

Publish The Data

Tabular

Apache Iceberg's Integrated Audits – More Details

Stage The Data

- Set **write.wap.enabled=true** on your table
- Set **spark.wap.id=<UUID>** in your Spark session conf
- Run your **production** ETL code

Audit The Data

- Find the snapshot ID from your **production** table's metadata that's tagged with the same **spark.wap.id**
- Perform validations against this data (using ***any*** auditing tool or framework)

Publish The Data

- If your audits **fail**, go back to the drawing board.

Tabular

Apache Iceberg's Integrated Audits – More Details

Stage The Data

- Set **write.wap.enabled=true** on your table
- Set **spark.wap.id=<UUID>** in your Spark session conf
- Run your **production** ETL code

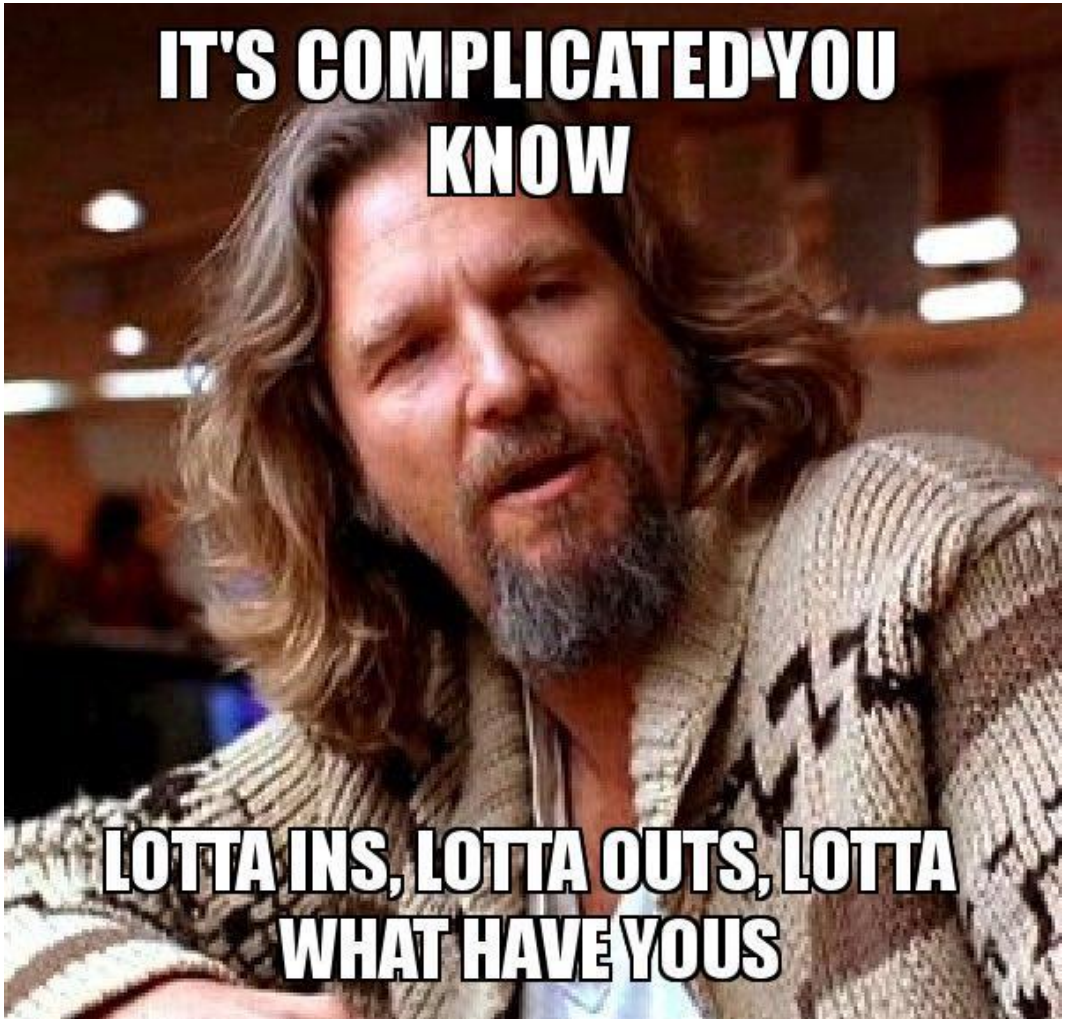
Audit The Data

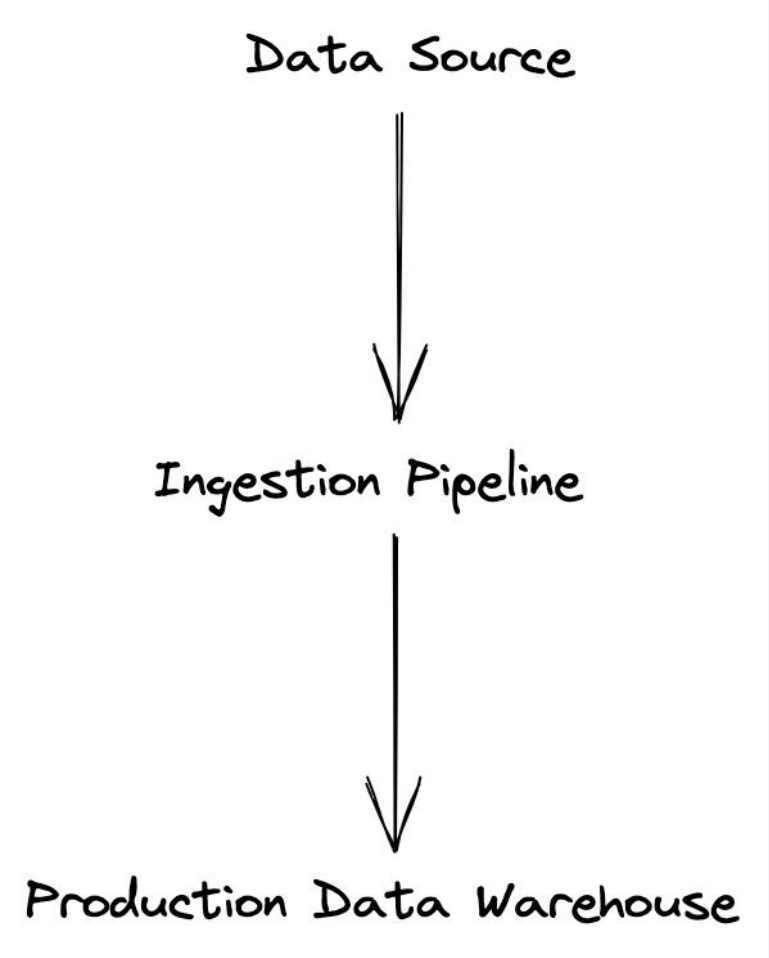
- Find the snapshot ID from your **production** table's metadata that's tagged with the same **spark.wap.id**
- Perform validations against this data (using ***any*** auditing tool or framework)

Publish The Data

- If your audits **fail**, go back to the drawing board.
- If your audits **pass**, execute an Iceberg **cherry-pick** of the snapshot ID. (A metadata only operation).

Tabular

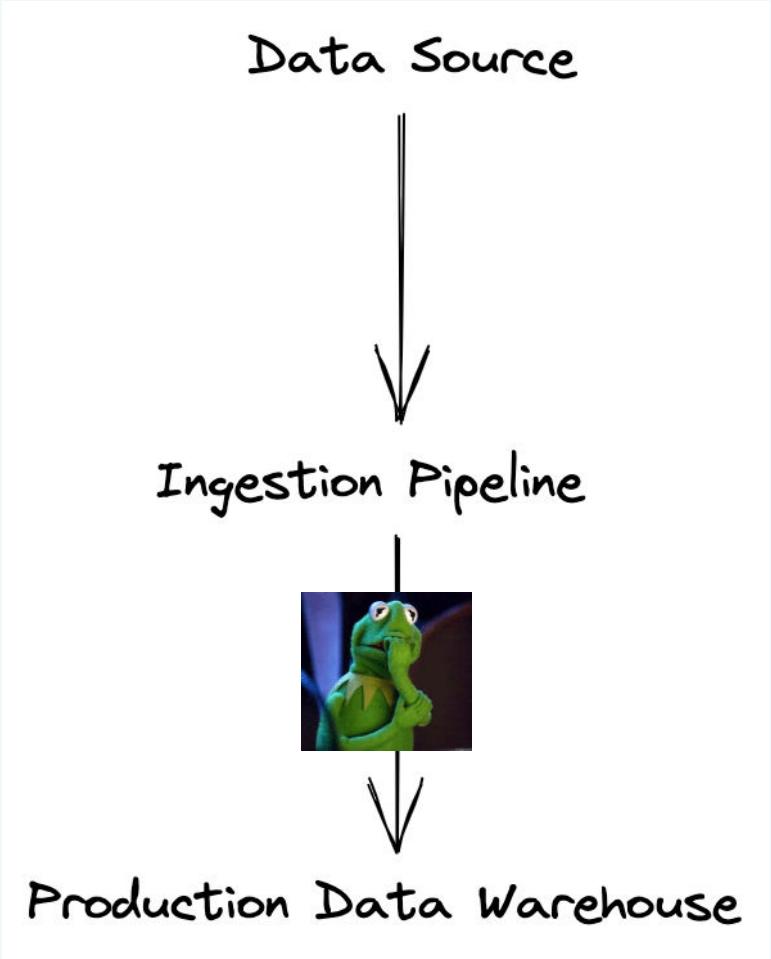

IT'S COMPLICATED YOU KNOW
LOTTA INS, LOTTA OUTS, LOTTA WHAT HAVE YOUS



Iceberg is a Data Quality Enabler

Through the integrated audits feature, Iceberg provides you the flexibility to allow auditing tools to scale with your data warehouse.

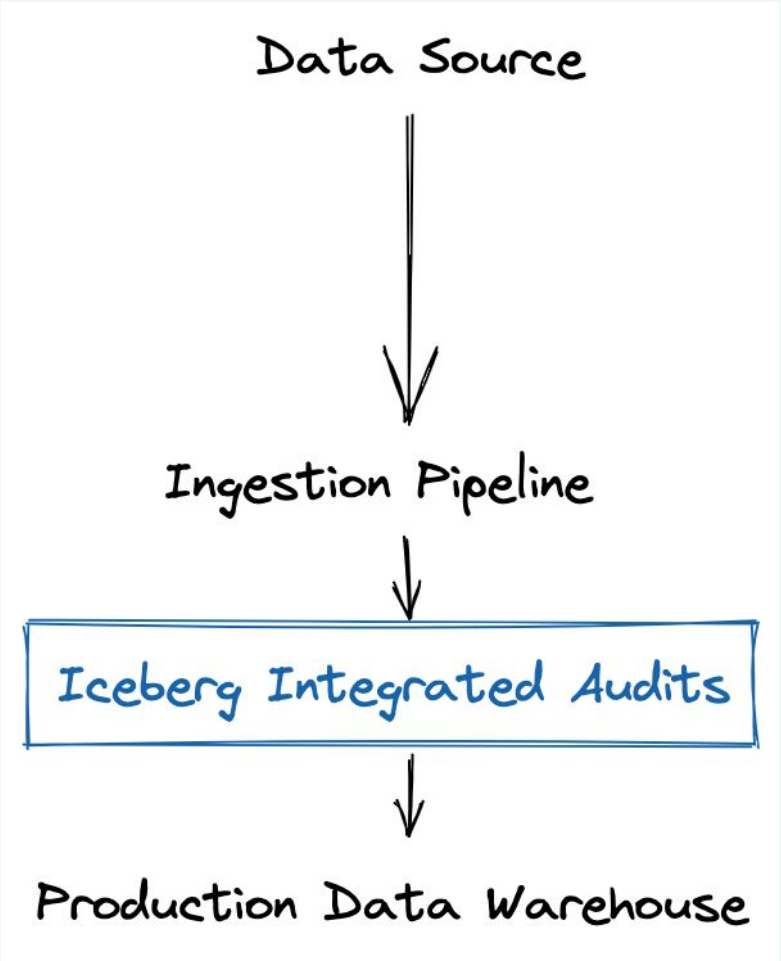
Tabular



Iceberg is a Data Quality Enabler

Through the integrated audits feature, Iceberg provides you the flexibility to allow auditing tools to scale with your data warehouse.

Tabular



Iceberg is a Data Quality Enabler

Through the integrated audits feature, Iceberg provides you the flexibility to allow auditing tools to scale with your data warehouse.

Tabular

No more…

- …writing your data twice
- …remembering to clean up artifacts like “test tables”
- …remembering to keep “test” and “prod” schemas synced
- …locking yourself into a single auditing tool
- …tight coupling of your ETL logic with your validation logic

Tabular

No more…

- …writing your data twice
- …remembering to clean up artifacts like “test tables”
- …remembering to keep “test” and “prod” schemas synced
- …locking yourself into a single auditing tool
- …tight coupling of your ETL logic with your validation logic

“No, It's all automated now. Come on.”

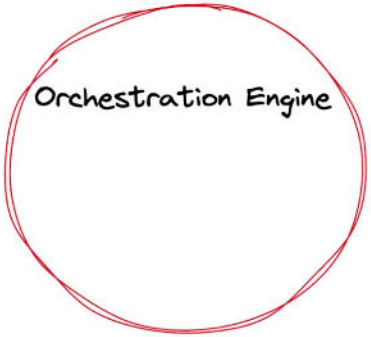
- Weekend at Bernie’s

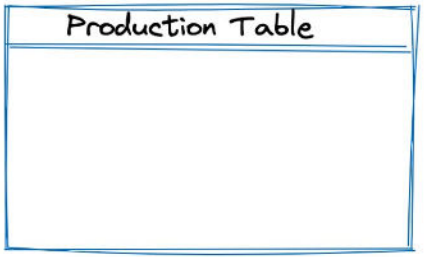
Tabular

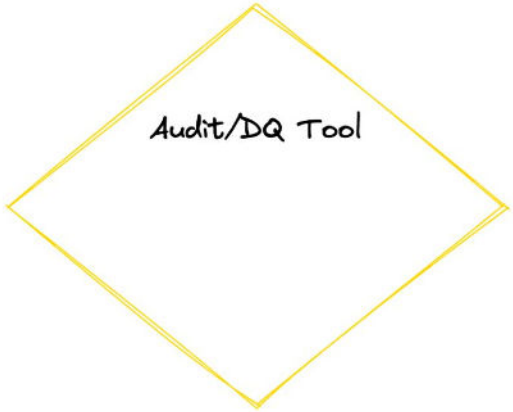
The Actual Audits

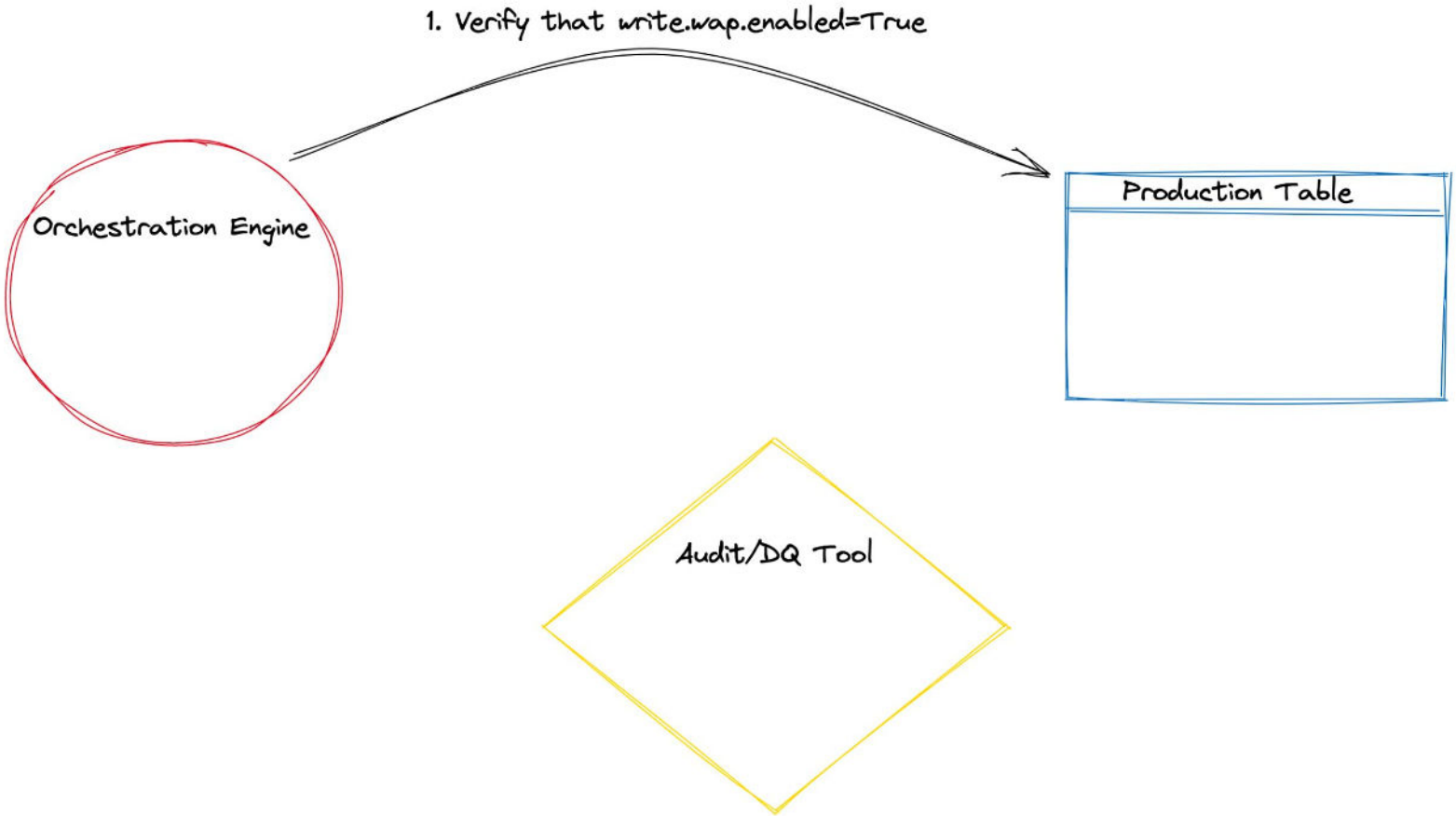
…and the role of your orchestration system

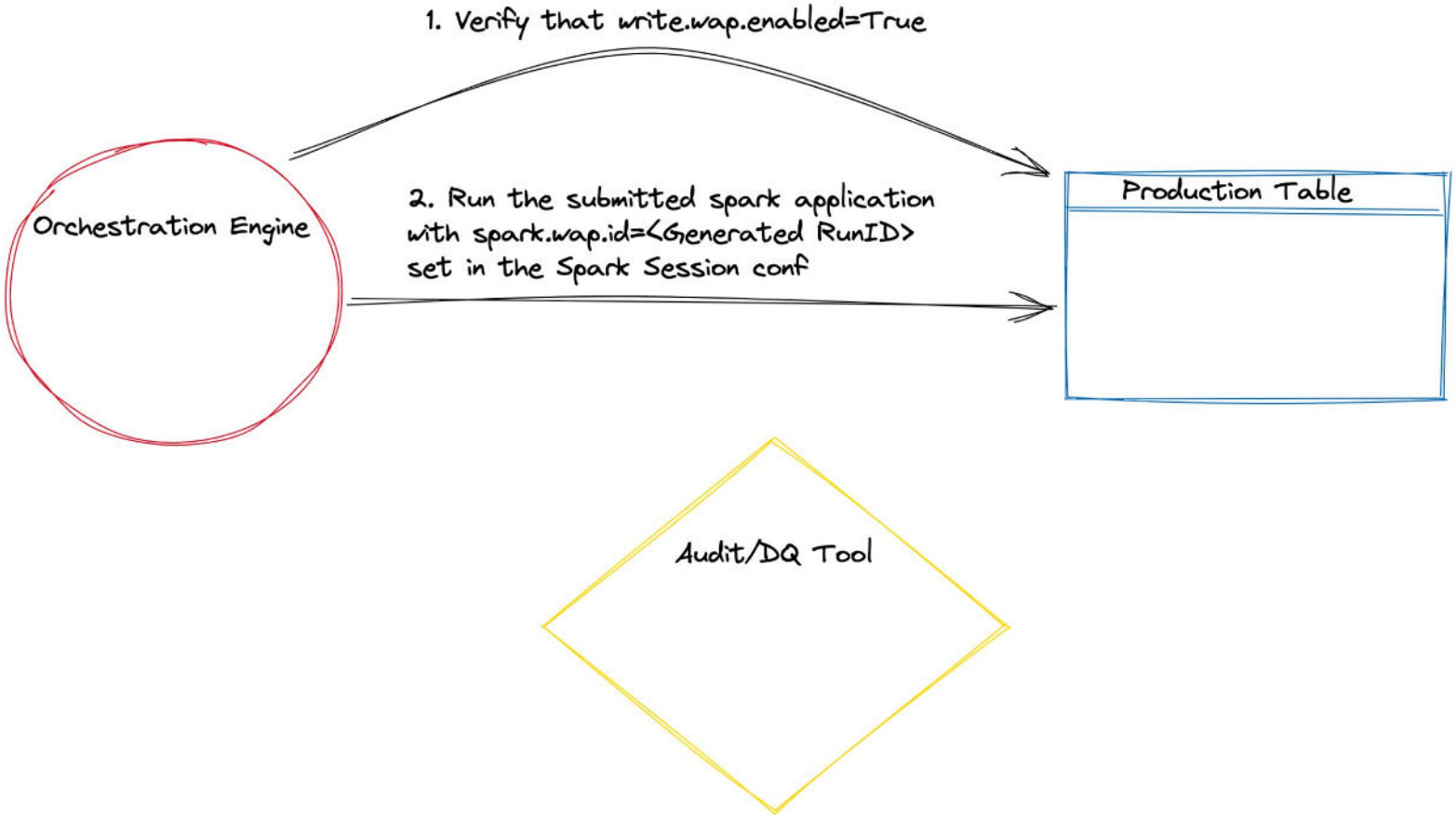
Tabular

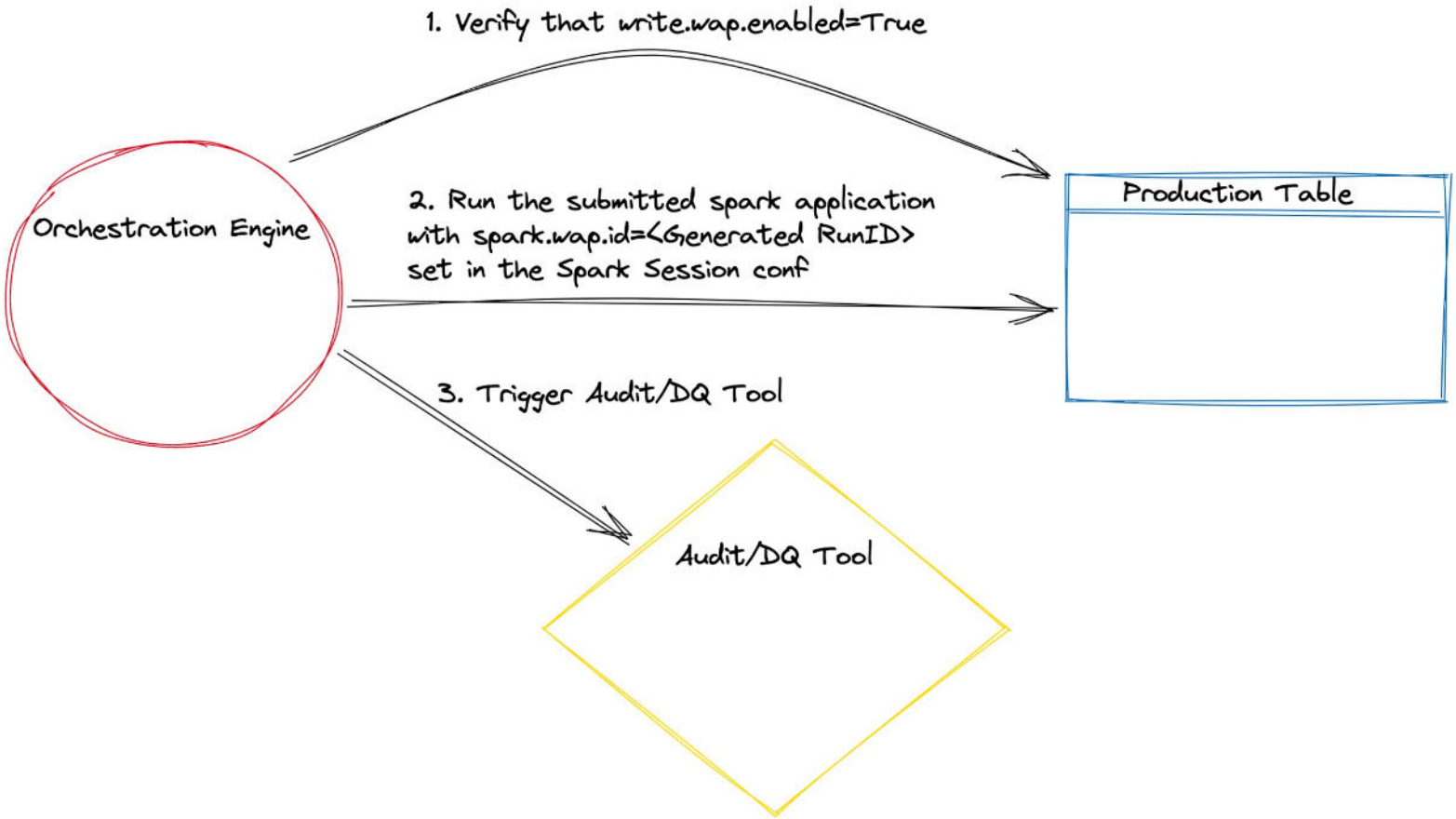

Orchestration Engine

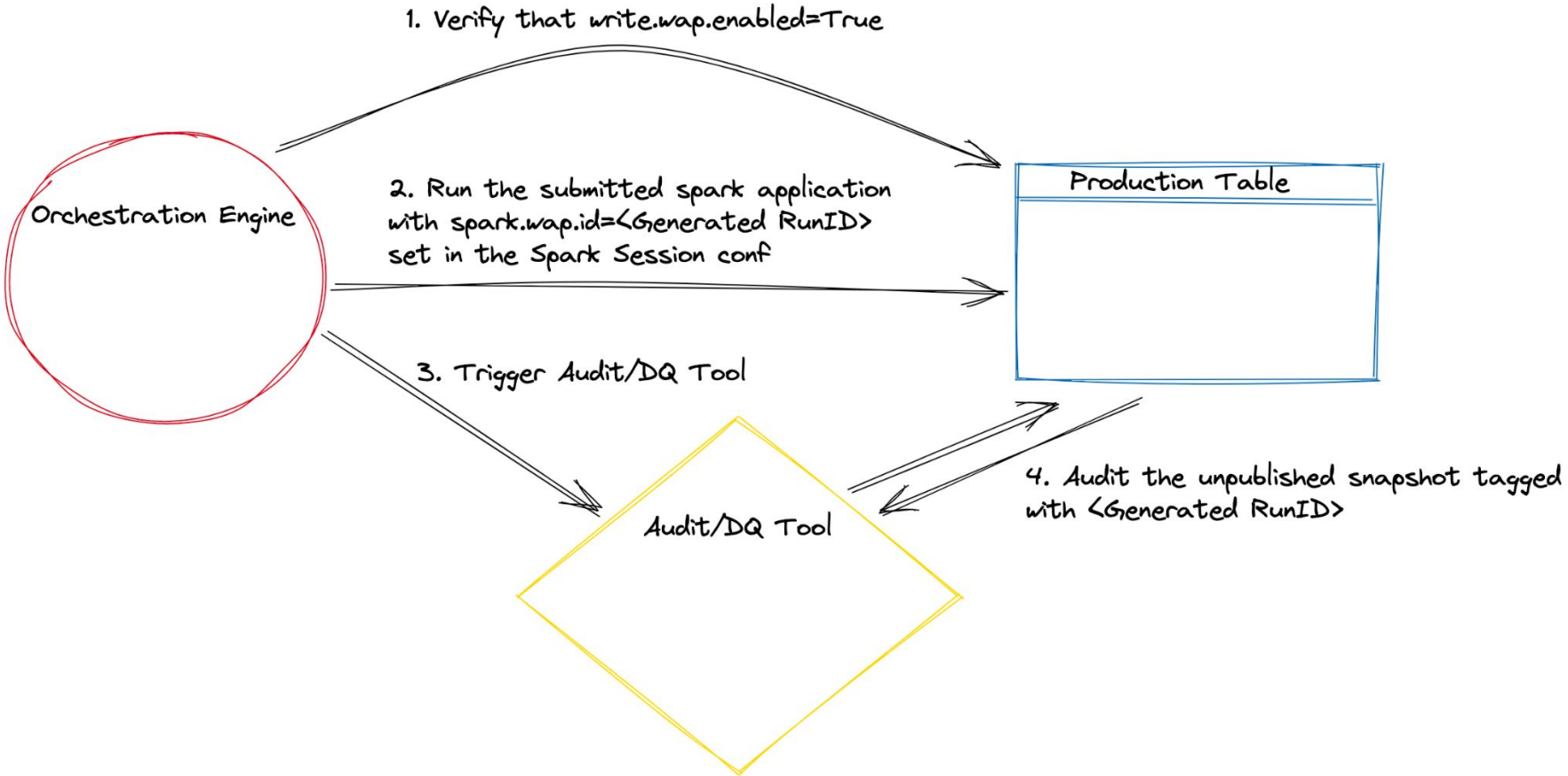

Production Table

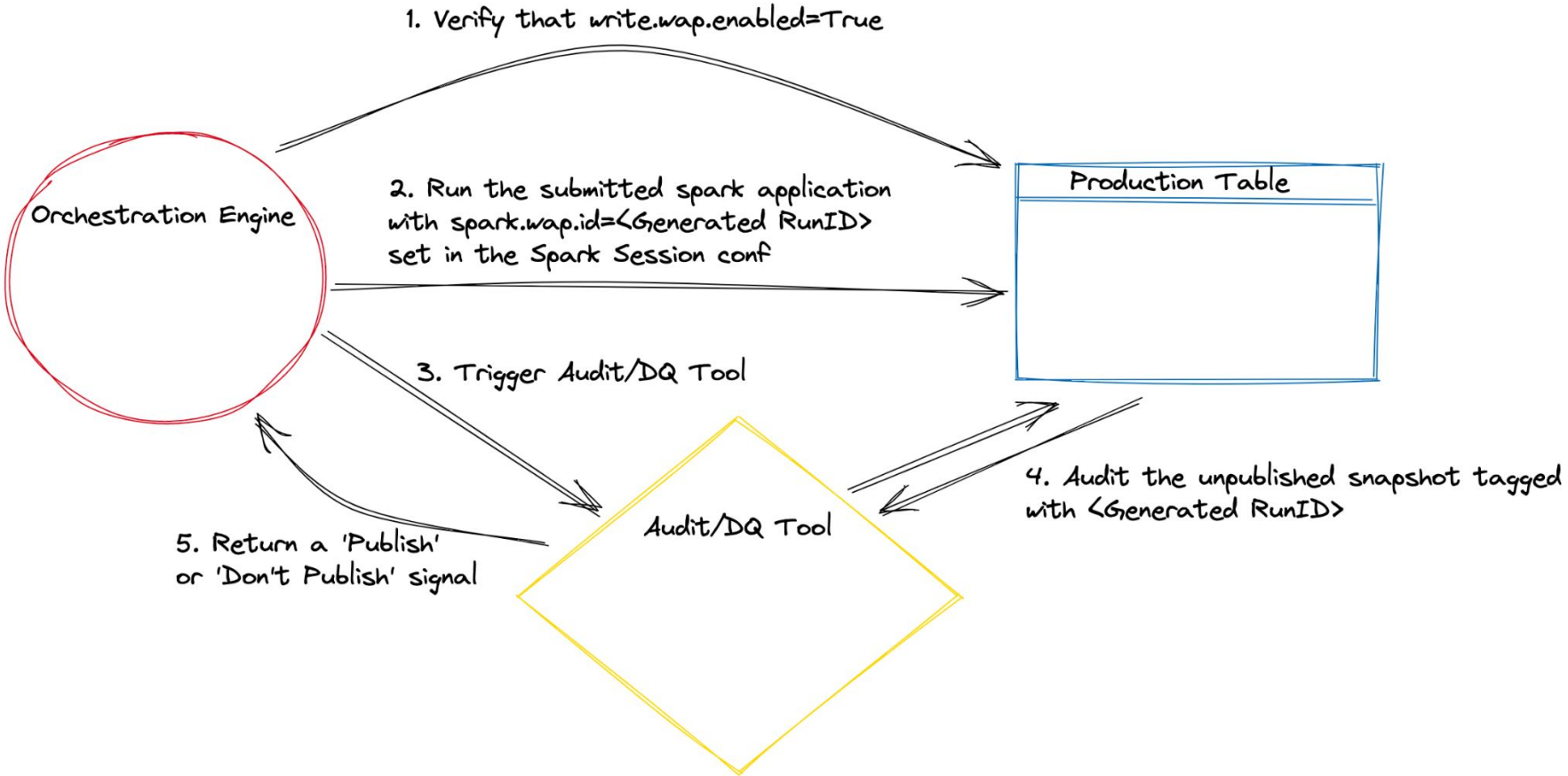

Audit/DQ Tool

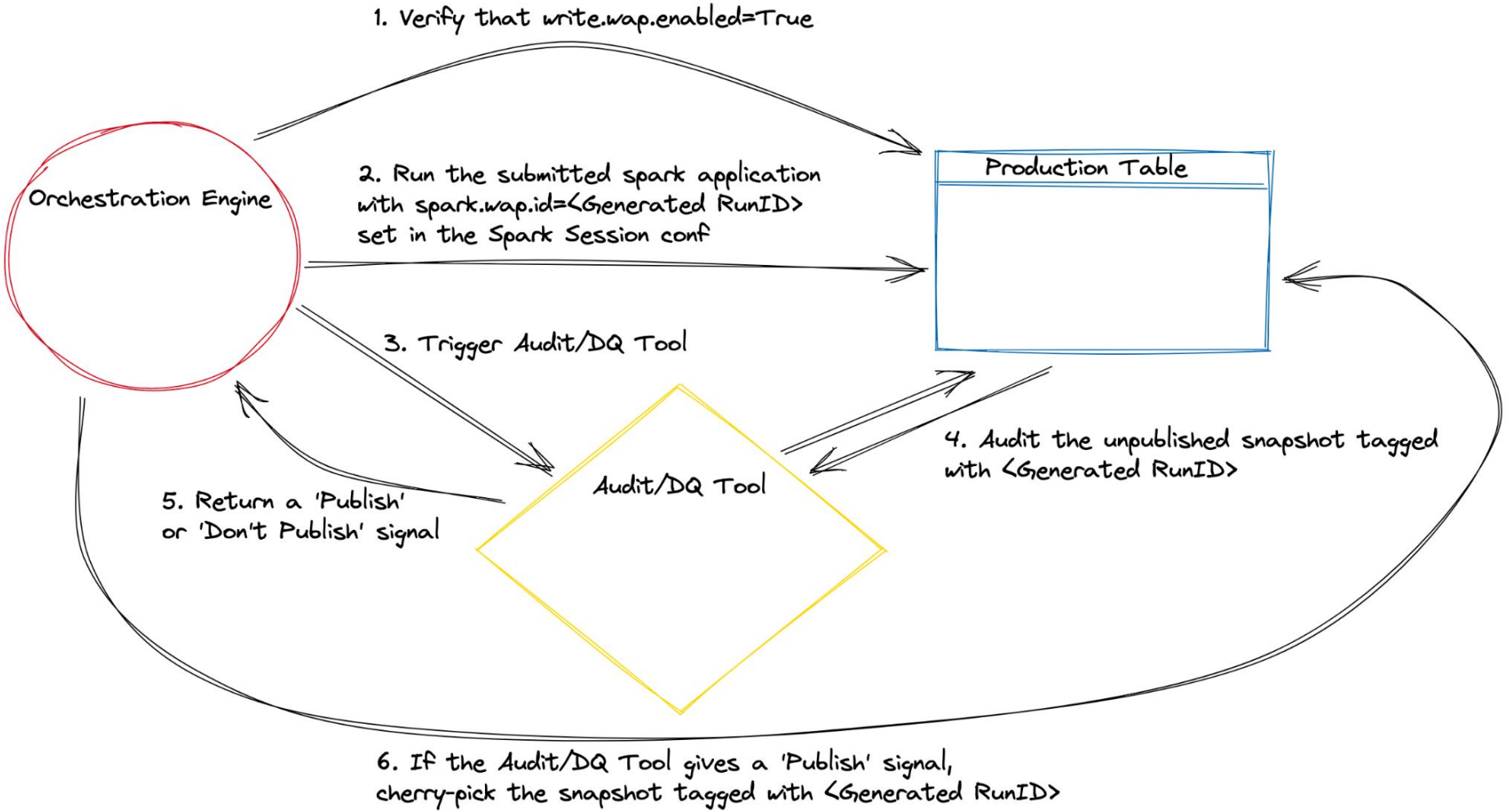

1. Verify that write.wap.enabled=True
Orchestration Engine
Production Table
Audit/DQ Tool


1. Verify that write.wap.enabled=True
Orchestration Engine
2. Run the submitted spark application
with spark.wap.id=<Generated RunID>
set in the Spark Session conf
Production Table
Audit/DQ Tool


1. Verify that write.wap.enabled=True
Orchestration Engine
2. Run the submitted spark application
with spark.wap.id=<Generated RunID>
set in the Spark Session conf
Production Table
3. Trigger Audit/DQ Tool
Audit/DQ Tool


1. Verify that write.wap.enabled=True
Orchestration Engine
2. Run the submitted spark application with spark.wap.id=<Generated RunID> set in the Spark Session conf
Production Table
3. Trigger Audit/DQ Tool
Audit/DQ Tool
4. Audit the unpublished snapshot tagged with <Generated RunID>


1. Verify that write.wap.enabled=True
Orchestration Engine
2. Run the submitted spark application
with spark.wap.id=<Generated RunID>
set in the Spark Session conf
Production Table
3. Trigger Audit/DQ Tool
Audit/DQ Tool
4. Audit the unpublished snapshot tagged
with <Generated RunID>
5. Return a 'Publish'
or 'Don't Publish' signal


1. Verify that write.wap.enabled=True
Orchestration Engine
2. Run the submitted spark application
with spark.wap.id=<Generated RunID>
set in the Spark Session conf
Production Table
3. Trigger Audit/DQ Tool
Audit/DQ Tool
4. Audit the unpublished snapshot tagged
with <Generated RunID>
5. Return a 'Publish'
or 'Don't Publish' signal
6. If the Audit/DQ Tool gives a 'Publish' signal,
cherry-pick the snapshot tagged with <Generated RunID>

Thank you

Learn more:
Integrated Audits: Streamlined Data Observability with Apache Iceberg (blog post)
https://tabular.io/blog/integrated-audits

Contact us at Tabular:
www.tabular.io

Iceberg Community Page:
iceberg.apache.org/community

Slack Workspace:
apache-iceberg

Follow me on twitter:
@samuelredai

Contact me through email:
sam@tabular.io

Tabular