Tiger Analytics

TIGER Data Observability

Overview Deck

Data Observability Glossary

Data Downtime: Periods of time when data is partial, erroneous, missing, or otherwise inaccurate — only multiplies as data systems become increasingly complex, supporting an endless ecosystem of sources and consumers

Data downtime = Number of data incidents X (Time-to-Detection + Time-to-Resolution)

SLAs (Service Level Agreement) : Data SLA refers to the quality and accessibility of your organization's data. SLAs must be defined after closely working with the business stakeholder. Some examples of Data SLAs are
When do you expect data to arrive the dashboard?
Time taken to complete the execution of the pipeline
What is the ideal time to detect data quality issues in the pipeline
What is the ideal time to resolve broken pipeline

In short, **SLA is the agreement you make with the clients or end users. SLA breach could incur loss or penalty**

SLIs(Service Level Indicator) : SLI, is measured by identifying key metrics that we can track and measure to achieve the agreed-upon SLA. Following are some of the examples of SLIs

- Pipeline - Total runtime in minutes
- Pipeline - Failure/Success rates
- Data Quality Metrics

In short, **SLIs are the actual KPIs or Metrics on the performance of Pipelines**

SLOs(Service Level Objective): SLO, is measured by aggregating Failure and Success metrics and validate if the defined Data downtime SLA is within the acceptable range.
E.g. 210 data sets are delivered every calendar year across all regions, and only 200 data sets are complete and meet the SLA. These successful deliveries translate to a 95.99% success rate for that year. The 10 failed (incomplete) data sets occurred at an acceptable error rate of 4%.

In short **SLOs are the objectives that are measured internally to see if we are meeting the SLAs**

Overview

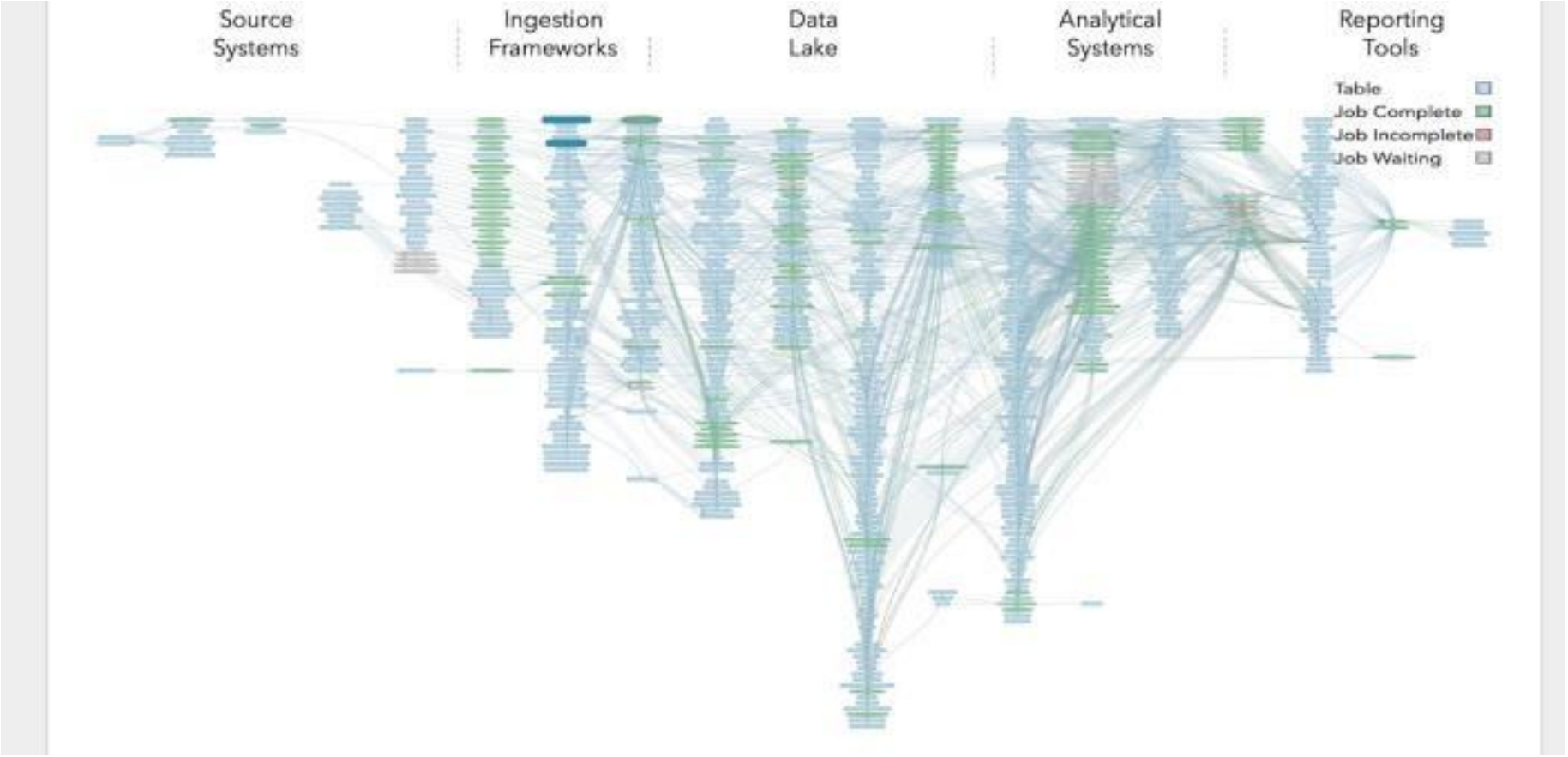
Companies spend upwards of **$15 million annually** tackling Data Downtime. In the year 2021 alone Gartner suggests, the cost of poor data quality reached upwards of $12.9 million per year.

Gartner predicts that, 70% of organizations will rigorously track data quality levels via metrics, improving it by 60% to significantly reduce operational risks and costs.

Data Engineers, Data scientists spend at least **30% of their time** tackling data quality issues and broken pipelines.

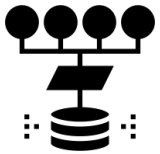
End users find data issues before the data team does. Cost of fixing data quality errors increases exponentially if the issue is detected late in the pipeline

A Bird's Eye view of modern data platform



Challenges in Modern Data Platform

Complex Landscape



Big data platforms are complex as it involves many distributed services to ingest, store and process data in the value chain. There is single pane of glass to monitor the Data Reliability, Quality, Performance in a single pane of glass

Data Quality

Large volumes of data can never be 100% error-free. Duplicate data, inconsistent data, schema changes, data drift are some of the common challenges when data pipelines keeps increasing in an Enterprise Data Lake

Poor Performance

Downtime, Latency, Data Quality , Schema Drift affects the critical business operations resulting in revenue loss

Productivity

Data Engineers and Data Scientists spend several hours and days to troubleshoot broken data pipelines in Production environment rather focusing on business objectives

Solution: Data Observability

Forbes **defines data observability as a set of tools that allows data and analytics teams to track the health of enterprise data systems, identify, troubleshoot, and fix problems when things go wrong**. In other words, data observability refers to an organization's ability to maintain a constant pulse of their data systems by tracking, monitoring, and troubleshooting incidents to minimize and eventually prevent data issues, downtime and improve data quality.

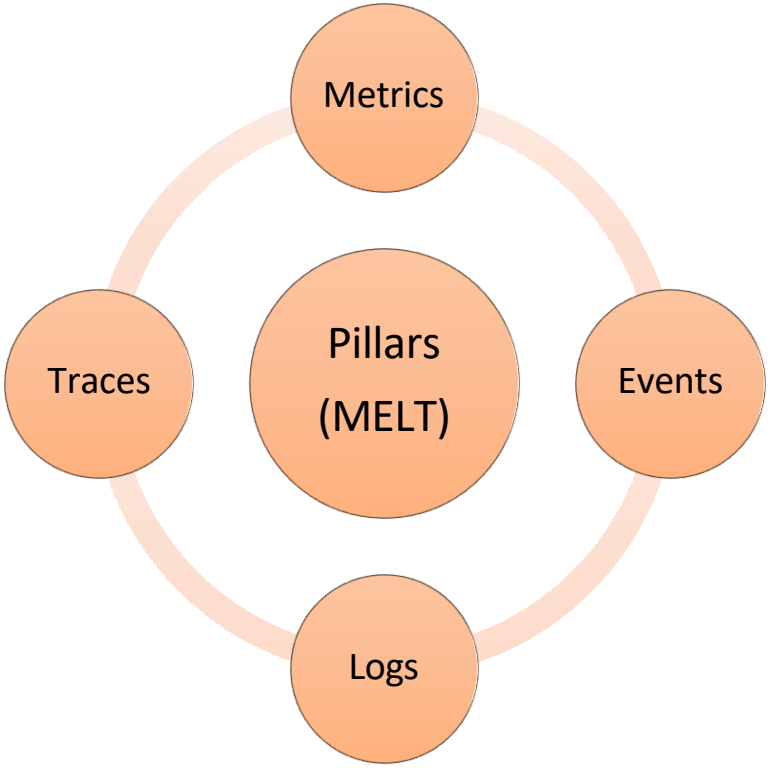
As enterprise data systems become multi-layered and more complex**, data observability answers the WHY questions behind broken pipelines and helps organizations speed up innovation**, boost efficiency and eventually reduce IT costs by avoiding unnecessary over-provisioning and optimizing data infrastructure.

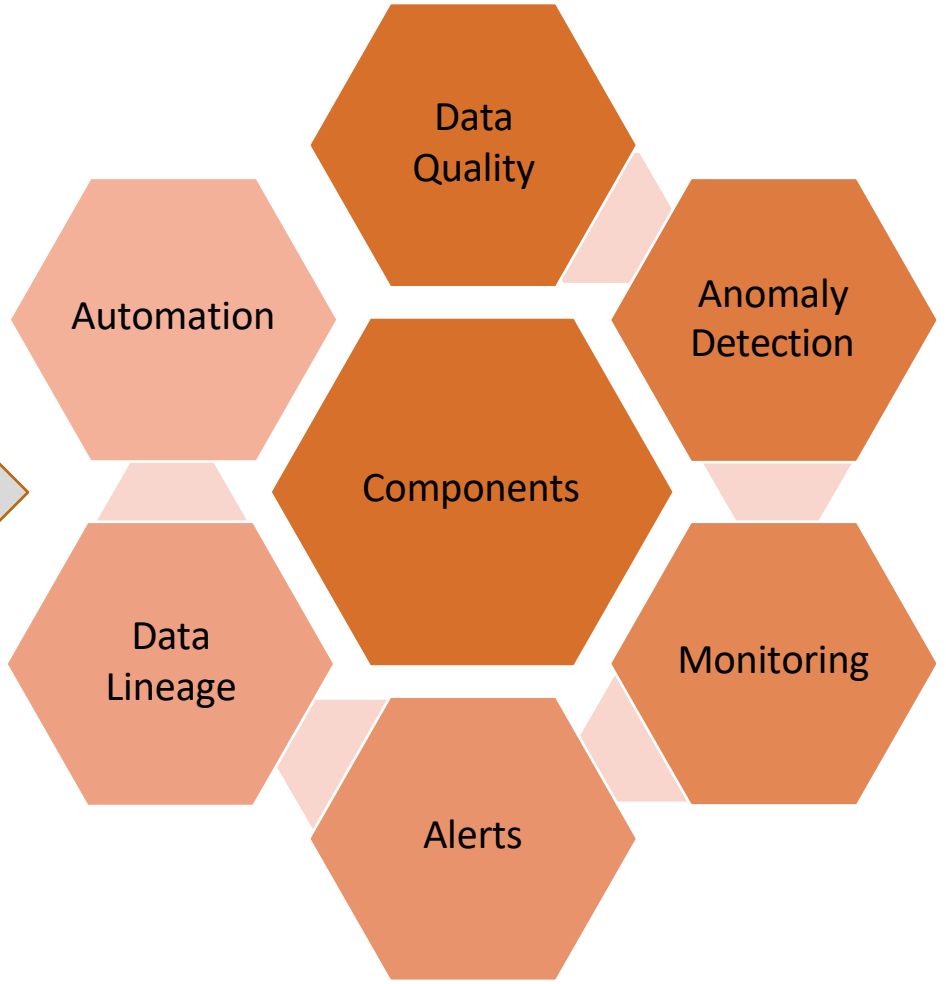
HEALTHIER PIPELINES = PRODUCTIVE ENGINEERING TEAMS = HAPPY CUSTOMER

Data Observability: User Persona , Use case and Outcome

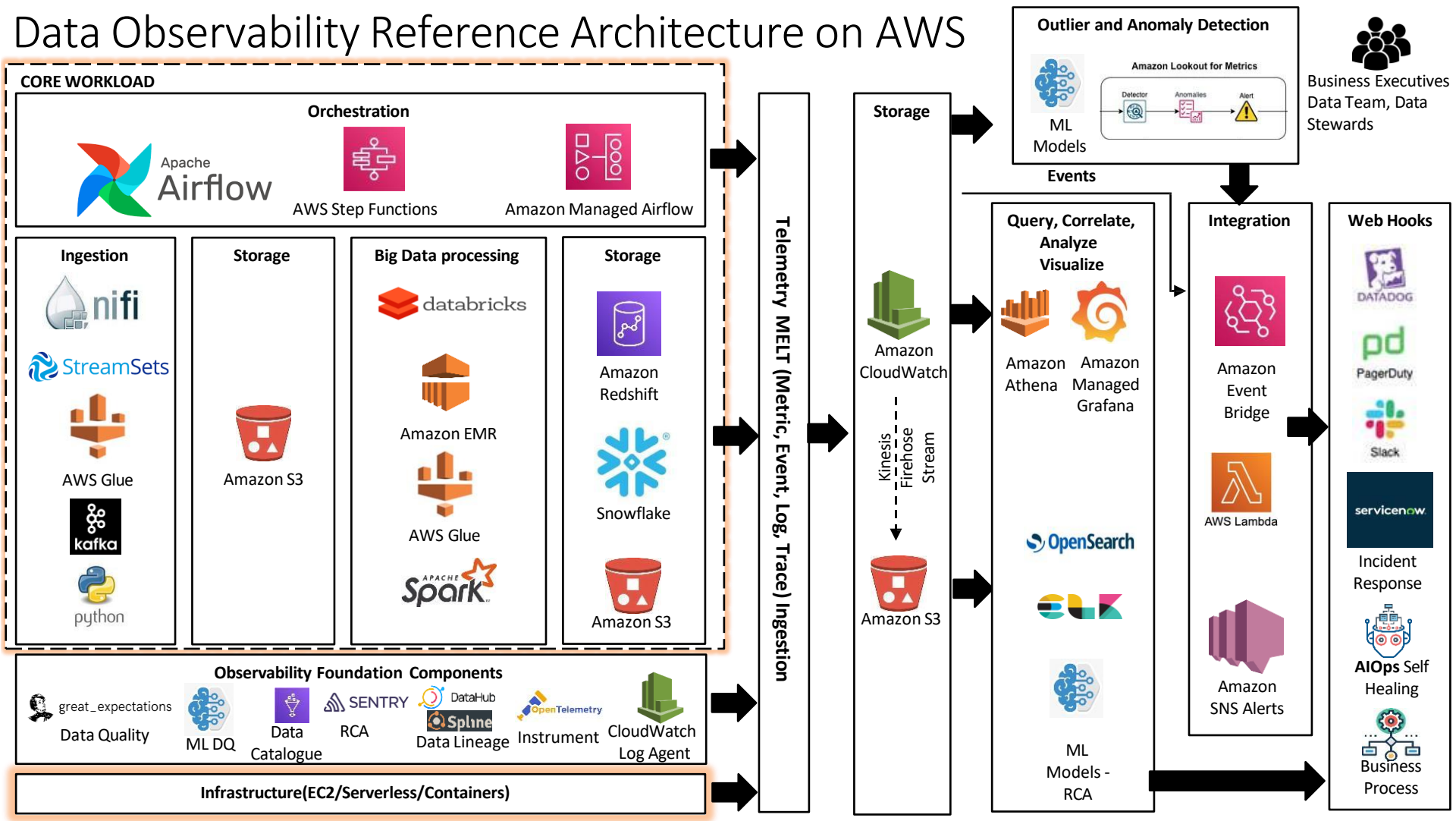
User Persona	Use case/WHY Questions	Outcome
End Users/Business Executives	➢ Why is my dataset/report not accurate ➢ Why is the data missing for a timeframe ➢ Why my mandatory fields are missing data ➢ Why is there a delay in seeing data in my report	➢ Improves the quality of insights, ➢ Improves trust and confidence in decision making
Data Engineers	➢ Why is my pipeline failing? ➢ Why my pipeline takes longer time to run today when compared to last week? ➢ Why is my pipeline not performing or meeting the SLA?	➢ Win Customer delight ➢ Improves Productivity ➢ Spend less time in troubleshooting and focus on data initiatives
Data Reliability Engineers	➢ Why there is data downtime ➢ Track the health of my pipeline ➢ RCA on pipeline failures ➢ Detect late arriving data	➢ Avoid unplanned outages ➢ Speed up MTTR ➢ Save troubleshooting time
Platforms Team	➢ Infrastructure monitoring	➢ Avoid resource(compute, storage) contention and increasing the reliability of data pipelines
Data Scientists	➢ Identity Schema, Data drift earlier in the pipeline ➢ Why the data is affecting my model predictions	➢ Stop bad data flowing to the Model consumption
Data Stewards	➢ Define & Manage Data Quality rules ➢ Data Profiling ➢ Monitor Data Quality KPIs/Metrics	➢ Ensure the quality of data promised to business is met ➢ Increase Data adoption across enterprise

Data Observability Pillars and Components





Data Observability Reference Architecture on AWS



Use Case 1- Data Quality

What, Why & How

Overview Data Quality Framework

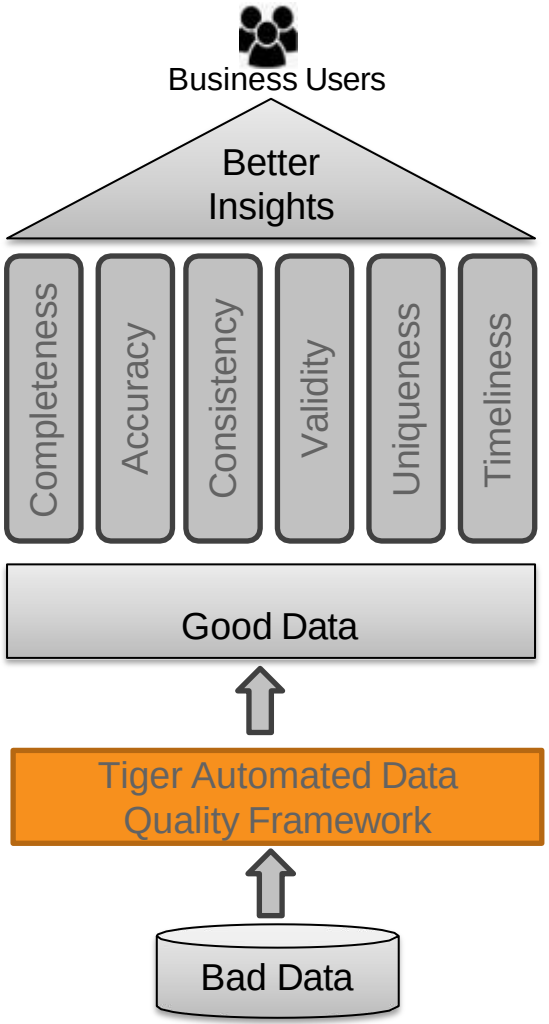
Tiger's Data Quality Framework is built to detect the data quality problems and notify the business via Data Quality Dashboards and notifications so that the business and data team gets better visibility about the health of the data

The framework is built on top of Great Expectations Library, and we have abstracted some of its functionalities and made it configurable and parametrized for the calling applications.

The framework also comes with custom dashboards and notifications module to visualize the data quality metrics and generate alerts based on configurable thresholds

Key Features

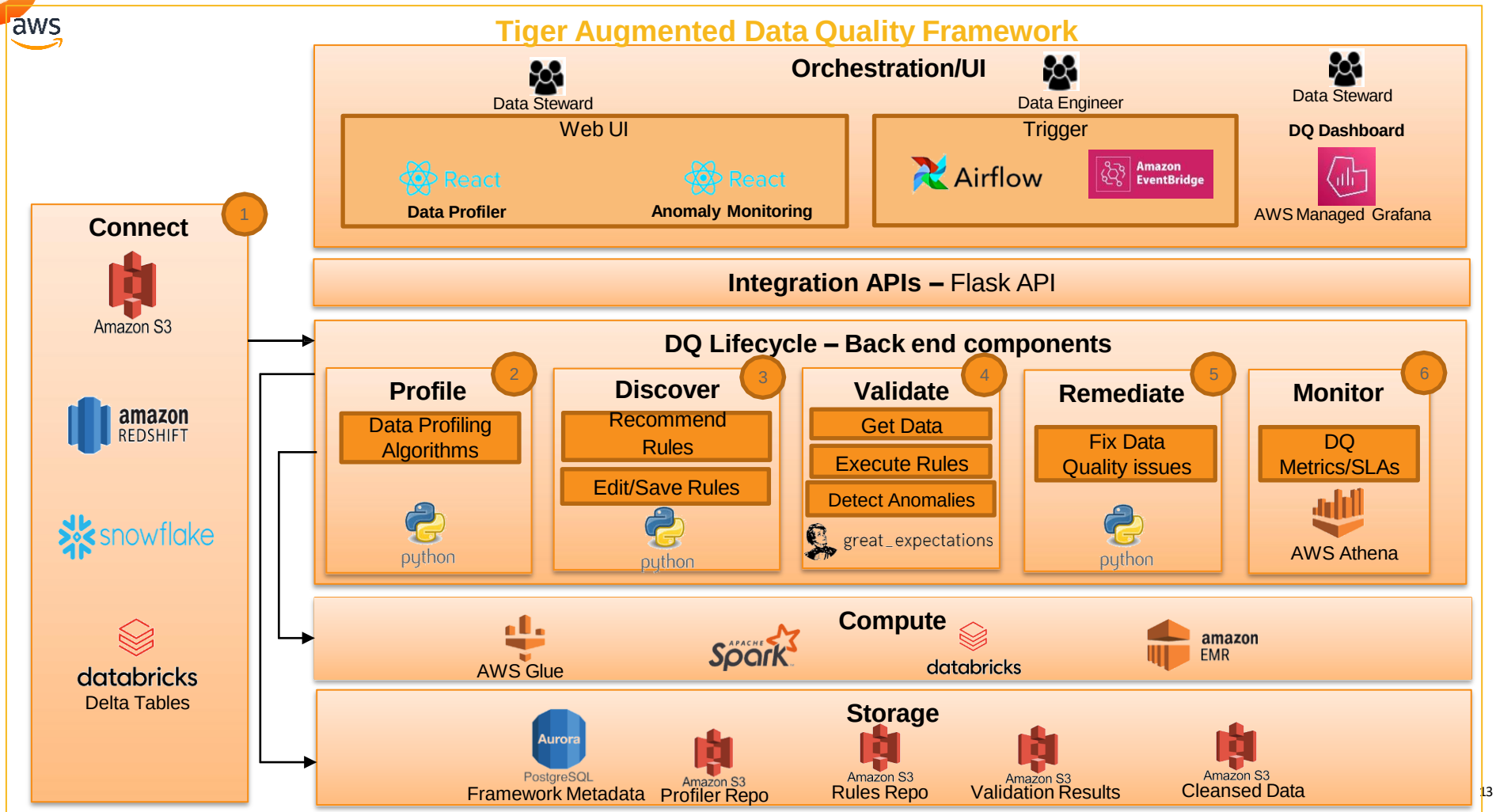
- Supports variety of data sources like S3, GCS, Databricks Tables many more
- Metadata driven makes it easy to configue the data quality rule
- Can be integrated to AWS Services like Glue, EMR
- Available as a plug and play PySpark package



Key Principles of Data Quality Framework

Self Service	The data profiling, rule recommendation services can be accessed by end users using a web based interface
Scalability	Scalable framework that can cater to high number and variety of checks
Configurability	Focus on configuration driven framework to eliminate need for ongoing development as use cases increase
Portability	Cloud agnostic and open-source tool set to ensure plug and play design
Extensibility	Ability to add custom profilers, custom rulesets
Maintainability	Modular and metadata driven framework for ease of maintenance
Integration	Ability to integrate with existing pipelines, or function independently in a decoupled fashion with ad hoc invocation or event driven patterns

DQ Framework – The Big Picture



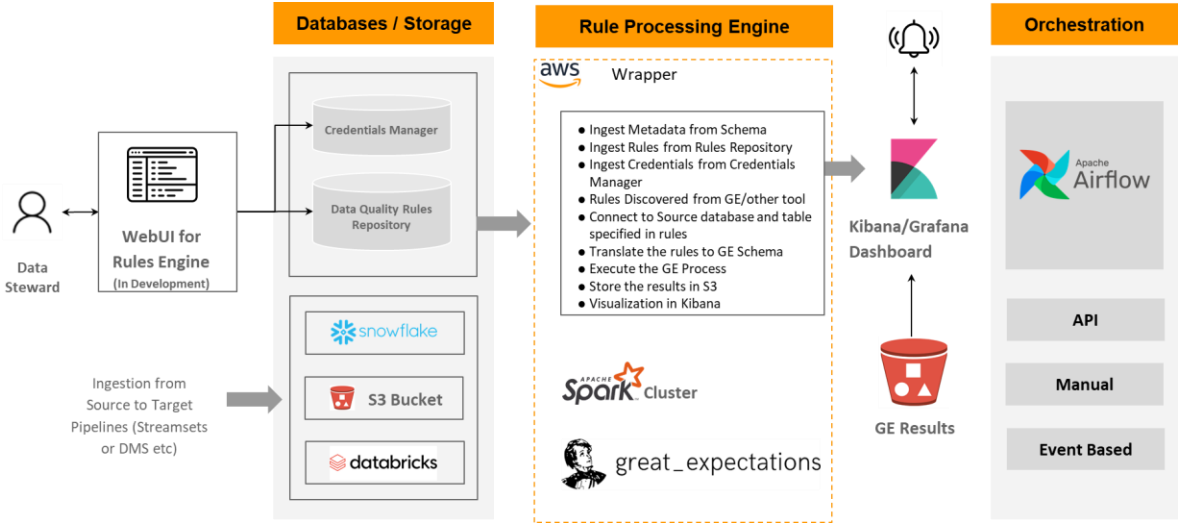
Case Study

How Tiger built an Observability Platform for Leading Financial Management Company

Case Study | Tiger Data Observability & Data Quality for a large financial Services Company

Business context

- Our client, a Fortune 500 financial services organization, had migrated a lot of their data and analytics workloads to a data and analytics platform on the cloud.
- The data platform was a central hub for customer, sales, and marketing data, and supported key analytics and ML Ops use cases. There were numerous additional data sources planned to be onboarded, and several additional analytical models planned for deployment soon.
- There were constant issues around data quality, timeliness of data, frequent and unnoticed job failures adding a lot of overhead and pressure on the data operations team.



Challenges

- Frequent data quality issues that were identified after errors were observed in the downstream applications
- No monitoring for data drift leading to significant unplanned downstream impact
- Poor timeliness due to failed and 'stuck' jobs led to missed SLAs on critical deliverables
- Lot of operational effort spent on troubleshooting 'silent' job failures
- Multiple silos of logs that required analysis to troubleshoot errors
- Failure alerts had limited information and did not aid analysis

Solution Approach

- A two-week assessment was performed to understand the challenges and produce a roadmap to address them
- Tiger Analytics implemented components from its Tiger Observability Framework to enable transparent data operations, and proactive monitoring and alerting
- A data quality framework was configured using Great Expectations and custom validations for time series and advanced validations.
- A central log store was set up to enable the log parsing and interpretation modules to constantly monitor data quality and processing and alert/act on unexpected signals
- Alerts were generated using emails, MS Teams and automated ServiceNow tickets to ensure timely action by the operations teams

Use Case 2 – Pipeline Observability

Monitor Health of Data Pipelines

Overview Reliable Data Pipeline

Tiger's Pipeline Observability Framework is built to identify pipeline failures due to bad data, run-time errors, application errors, infrastructure issues. The framework aims to accelerate the root cause analysis for the platform engineers who can easily troubleshoot and pinpoint the root cause of failure without parsing too many logs.

The framework is built on top of S3 as the log aggregation layer and ELK as the log analytics platform. Use of open-source tools like Sentry will quickly mine the logs and pinpoint the root cause of the pipeline failure

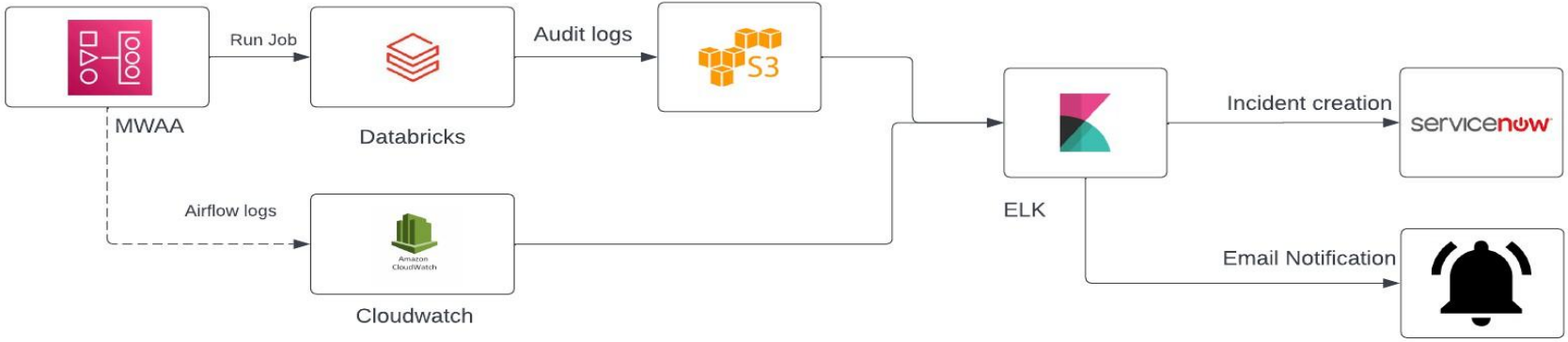
The framework also comes with custom dashboards and notifications module to visualize the health of the pipeline and generate alerts based on configurable thresholds

Key Features

- Supports variety of logs from Managed Airflow, Glue, Databricks
- Can be easily integrated to existing pipelines
- Alerts can be integrated with Servicenow
- Easily correlate logs using Run IDs

Pipeline Observability Solution

To get metric alerts , job failure notifications



1. MWAA triggers Databricks jobs
2. On completion of MWAA job, task logs are moved into CloudWatch
3. Databricks Audit logs capture status of every event and can be used to monitor job status. Audit logs are written to S3
4. Insights are generated from the Audit logs and CloudWatch logs on ELK
5. For failed jobs, incidents are created on S-Now and users are notified via email

Key Takeaways

- Build end to end data reliability with healthier pipelines
- Discover data problems before they negatively impact the business KPIs
- Deliver high-quality data for the consumption use cases
- Increased trust in data for key business decisions
- Efficient health monitoring and alerting in a complex data eco system
- Accelerate Troubleshooting and identify the root case issue quickly

Key Takeaways

- Build end to end data reliability with healthier pipelines
- Discover data problems before they negatively impact the business KPIs
- Deliver high-quality data for the consumption use cases
- Increased trust in data for key business decisions
- Efficient health monitoring and alerting in a complex data eco system
- Accelerate Troubleshooting and identify the root case issue quickly

Next Steps

- Deep dive discovery workshop with your SMEs (Engineering, Operations, Business)to understand the AS-IS maturity level of Data Observability
- Understand AS-IS Pipeline debts
- Understand the gaps and recommend areas of improvement
- Recommend end state solution for the specific workload

TIGER ANALYTICS

Thank You

Do you have any questions?

Ajeeth Amarasekaran

Dataobserv.support@tigeranalytics.com

www.tigeranalytics.com