Document Intelligence pipeline

End-to-End Architecture & Data Processing Strategy

AGENDA

01 **Introduction**

- What the System Does
- Key Components & Architecture

02 **Ingestion Layer**

- Cloud Run API
- Raw Storage Design

03 **Processing layers**

- OCR Pipeline (DAG 1)
- Base ETL (DAG 2)
- Analytics & Features (DAG 3)
- Data Quality & Lineage

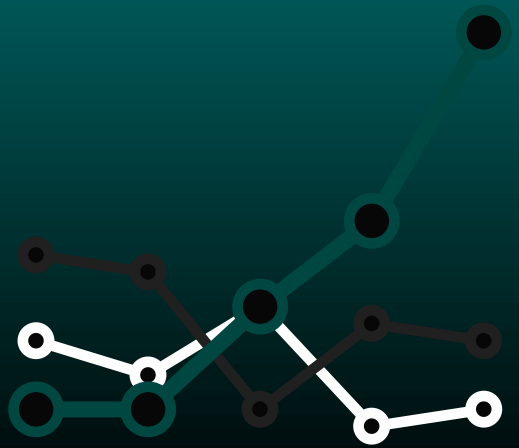
04 **Governance & Operations**

- Error Handling & Retries
- Security & Compliance
- Monitoring & Observability

05 **Strengths & Reliability**

- Autoscaling Components

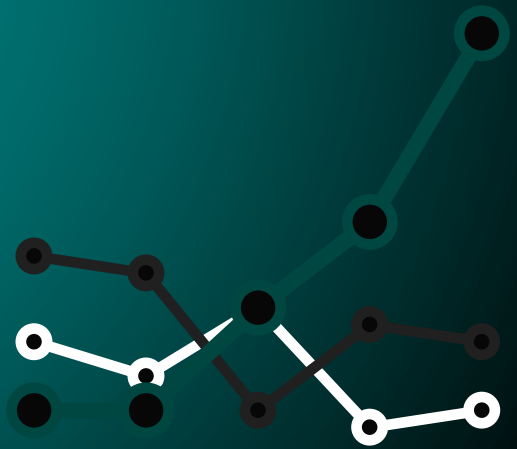
06 **Q&A**



INTRODUCTION

- The Document Intelligence Pipeline in this presentation is a scalable, cost-efficient, and auditable document processing system built on Google Cloud Platform (GCP). It ingests documents from upstream applications, applies OCR with dynamic cost-aware routing, transforms extracted content into structured datasets, and serves analytics- and feature-ready tables to downstream systems.

- The design balances simplicity and extensibility while ensuring strong governance, lineage, and operational visibility.



01

System Purpose & Goals

What the system does?

1. Ingests and processes diverse documents (bank statements, invoices, IDs, CRs).
2. Extracts text reliably at scale.
3. Converts unstructured OCR output into structured tables.
4. Enables analytics, lineage, and operational monitoring.

Primary goals are :

- ✓ Scalability.
- ✓ Cost-efficiency.
- ✓ Auditability and traceability.
- ✓ Metadata-driven transformation.

02

High-Level Architecture

Overview

1. Cloud Run ingestion

2. GCS Raw storage

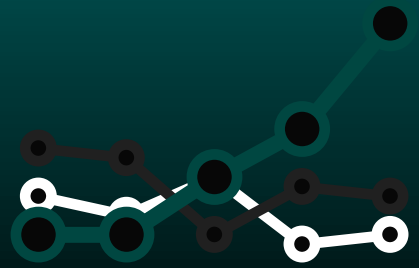
03. Pub/Sub buffering

4. Airflow OCR DAG

5. OCR JSON in GCS

6. Base ETL DAG BigQuery

7. Analytics DAG facts/features



03

Ingestion Layer

Cloud Run API:

- Accepts file uploads from client systems.
- Writes raw documents to GCS (basira-raw)
- Publishes metadata messages into dedicated Pub/Sub topics

Why this design?

✓ Decouples ingestion from processing.
✓ Pub/Sub acts as a durable event buffer.

04

Processing Layer

PROCESS HIGHLIGHT

Three Airflow DAGs orchestrate all processing:

DAG 1

OCR Pipeline

Extracts text via Tesseract or GCP OCR.

DAG 2

Base Layer ETL

Converts OCR JSON outputs into structured BigQuery tables.

DAG3

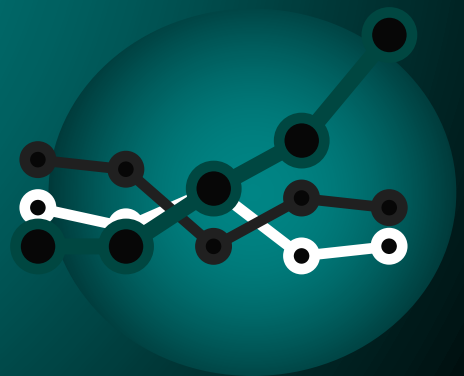
Analytics & Features

Builds fact and feature-ready datasets

✓ Outcome:
Full end-to-end traceability and clean separation of responsibilities.

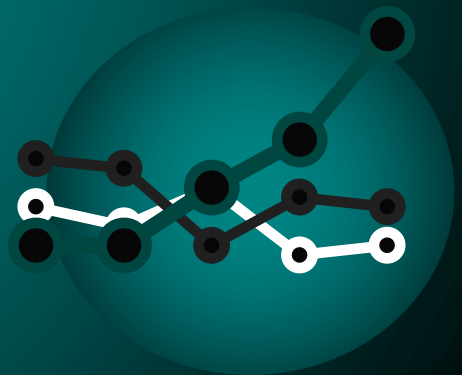
DAG 1 : OCR PIPLINE

- Batch-driven OCR workflow.
- Airflow pulls messages from Pub/Sub every (eg: 10 minutes).
- Performs cost-aware routing:
  - Light documents → **Tesseract OCR**.
  - Complex documents → **GCP Vision / Document AI.**
- Writes OCR text to GCS **basira-ocr.**
- Logs details to **logging.OCR_logs** in BigQuery.
- Ensures predefined load, low cost, and clear auditability.



DAG 2: Base Layer ETL

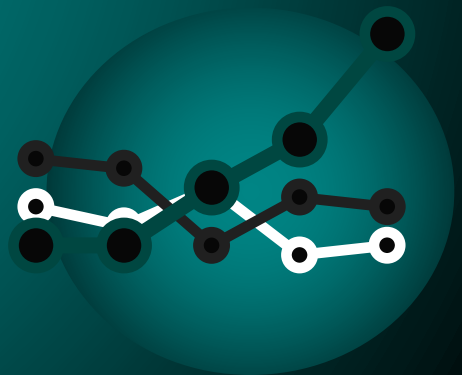
- **Metadata-Driven Transformation:**
  - Reads OCR JSON output
  - Reads YAML metadata from basira-metadata
  - ❖ YAML defines:
    - Extraction patterns.
    - Field mappings.
    - Data types.
    - Document-specific DQ rules.
- **Outputs:**
  - Structured BigQuery tables:
    - Bank statements.
    - Invoices.
    - ID cards.
    - Commercial_regs.



DAG 3 : Analytics & Feautures

- Analytics Layer Outputs:
- Fact tables (e.g., fact_customers).
- Summary KPIs (engine usage, DQ pass rate, latency).
- Feature tables for modeling or enrichment.

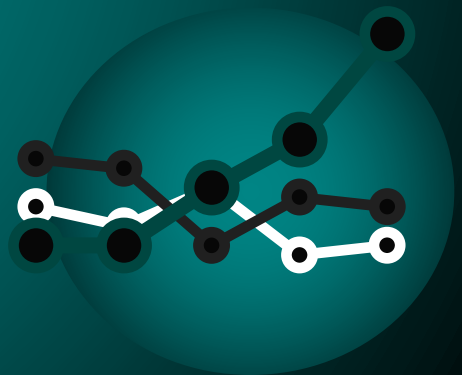
- Fully supports BI dashboards, reporting, and ML pipelines.



Data Quality & Lineage

- DQ Checks:
- YAML-defined rules (null checks, regex validation, completeness).
- Results stored in **monitoring.dq_results.**
- Base ETL logs in **logging.baselayer_logs.**

- Lineage:
- Every row contains **doc_id**, **doc_location**, **yaml_version**, **ocr_engine**.
- Enables traceability from **raw document** → **OCR** → **base** → **analytics.**



05

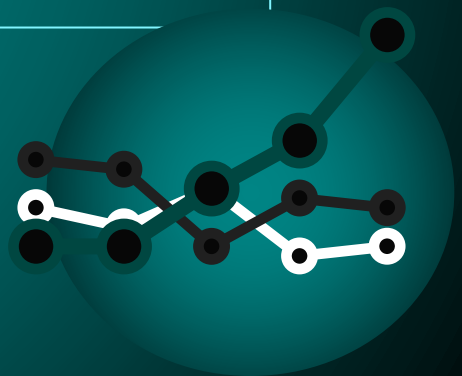
Governance, Security, and Masking

Enterprise controls:

- IAM + VPC-Service Controls on GCS and BigQuery.
- Sensitive fields masked using BigQuery policy tags.
- Access to raw documents restricted.
- Full audit log history for all stages.

Outcome:

Strong compliance and audit readiness across the full lifecycle



06

Why This Architecture Works?

Key strengths

1 Uses modular and decoupled services (Ingestion + 3 DAG layout).

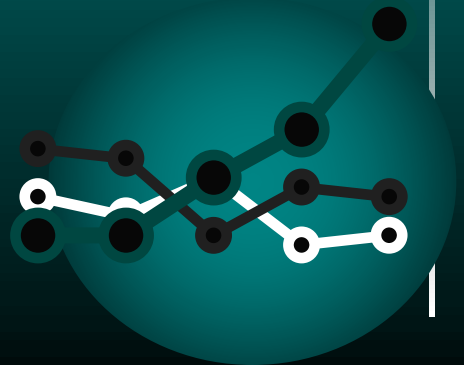
2 Cost-efficient OCR routing.

3 Metadata-driven design ensures scalability.

4 Strong lineage & logging offers auditability and traceability.

3 Supports analytics and ML growth.

4 This design is simple, scalable, auditable, and practical for real-world operations.



Q&A?

Thank You