

Tiger Analytics

Tiger Data Fabric



Tiger Data Fabric - Introduction



"Tiger's Accelerator for Implementing Data Lake & Data Fabric on Microsoft Azure in a highly scalable manner with powerful Governance Capabilities at it's core".



AGILITY WITH SELF-SERVICE

Cut-down the time it takes for setting up new Data Pipelines from **Days & Weeks to Few Minutes**.



SUPPORT MULTIPLE DATA SOURCES

Supports Extracting and Ingesting Massive Volumes of Multi-Structured Data from RDBMS, File System, SFTP & different Cloud Storages.



MEDALLION DATA LAKE

Follows Data Engineering Best Practices & based on Medallion Data Lake
Architecture (Bronze Silver Gold).



TRANSFORMATION CAPABILITIES

Capabilities to perform **Transformation Operations** like data cleansing, validation, quarantine, profile & merge.



SCALABLE CLOUD ARCHITECTURE

Highly Scalable & Service Oriented

Architecture – developed using different

Azure PaaS & Open Source Technologies.



GOVERNANCE CAPABILITIES

In-built capabilities for metadata & lineage Cataloging, Logging, Notification & Operational Reporting.

POWERED BY:

















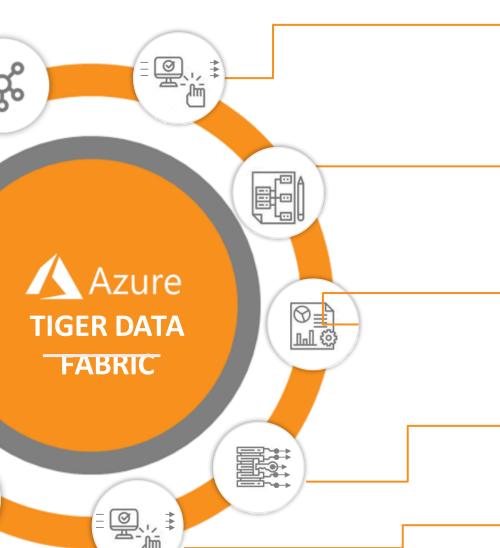




SUPPORTED INTERFACES:

- 1. REST API (JSON)
- 3. Web UI
- 2. EXCEL Upload
- 4. Chatbot / VA

Tiger Data Fabric – Key Features



SELF SERVICE DATA LAKE MANAGEMENT

- Highly scalable, End to end Data Lake management accelerator developed using Azure PaaS and Open Source technologies.
- Powerful Self-Service capabilities for setting up new data ingestion pipelines & transformation feeds, perform data life cycle management and monitor operations using **Chatbot**.

CONFIGURATION DRIVEN EXTRACTION & INGESTION PIPELINES

- Configurable extraction and ingestion pipelines, that can be reused for various data sources
- Supports Data extraction from File, RDBMS sources with Full, Incremental modes using Truncate and Reload, Upsert and Append merge strategies
- Support for Event , Schedule triggers

AUTOMATED DATA QUALITY & PROFILING

- Solution for implementing configuration driven data testing and automated data profiling to surface data quality issues and assess data quality confidence.
- Library of more than 50+ testing rules and ability to add more custom rules.
- Integrated DQ Reporting and alerting capabilities.

SERVICE FABRIC BUILT USING API

- Highly scalable, reusable services that can talk to each other using API
- Flexible, Robust API's that makes integration between various services easier

GOVERNANCE CAPABILITIES

Provide Data / Platform governance capabilities like Data catalogues, metadata management, Data lineage etc.

Data Pipeline Development Lifecycle - Without Accelerator

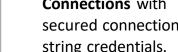


- **Understand** Dataset Metadata (Format, Channel, Table, Columns, Data type, Primary Key, Business Key, possible acquisition strategy etc.).
- Configure & Test DB **Connections** with secured connection string credentials.

- Prepare **Target** structure in Data Lake (Pipeline Metastore entries, Hive Database and Tables, Data Lake Folder structures etc.)
- Understand consumption pattern to design data partitioning strategy.
- Define **transformation** logic (data cleansing, validation, profiling, standardization, merge etc.).

- **Implement governance** features to improve maintainability & support (logging, alert & notification, checkpointing etc.).
- **Unit Test** all the Pipeline code (ADF, Databricks, SQL Database, ADLS, Synapse etc.).
- Schedule & run the pipelines in dev environment.

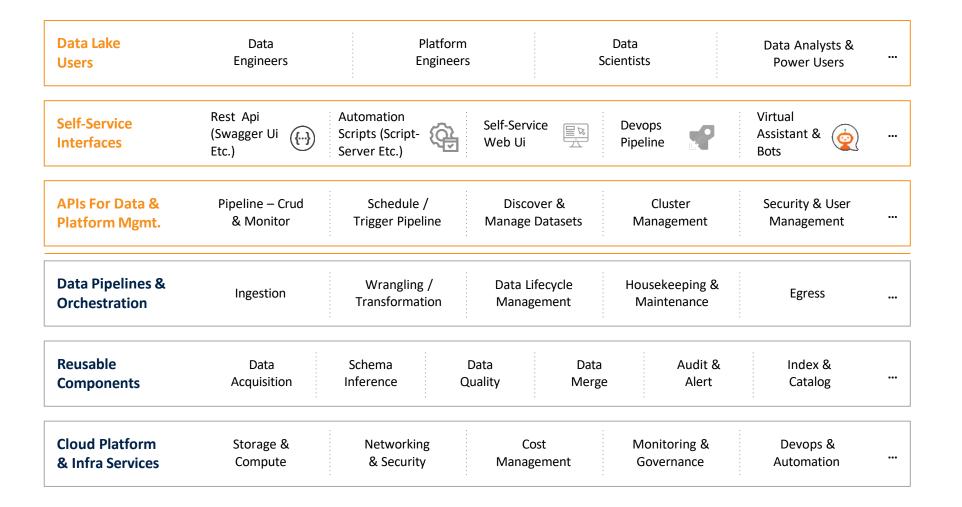
- 10. Deploy, Monitor and **Test** the pipelines in higher environments like QA and UAT / Staging.
- 10. Deploy to **Production** and monitor.
- 10. Perform various data lifecycle management & house-keeping activities like clean-up in case failure, restore, archive, reprocess error records etc.



Data Ops



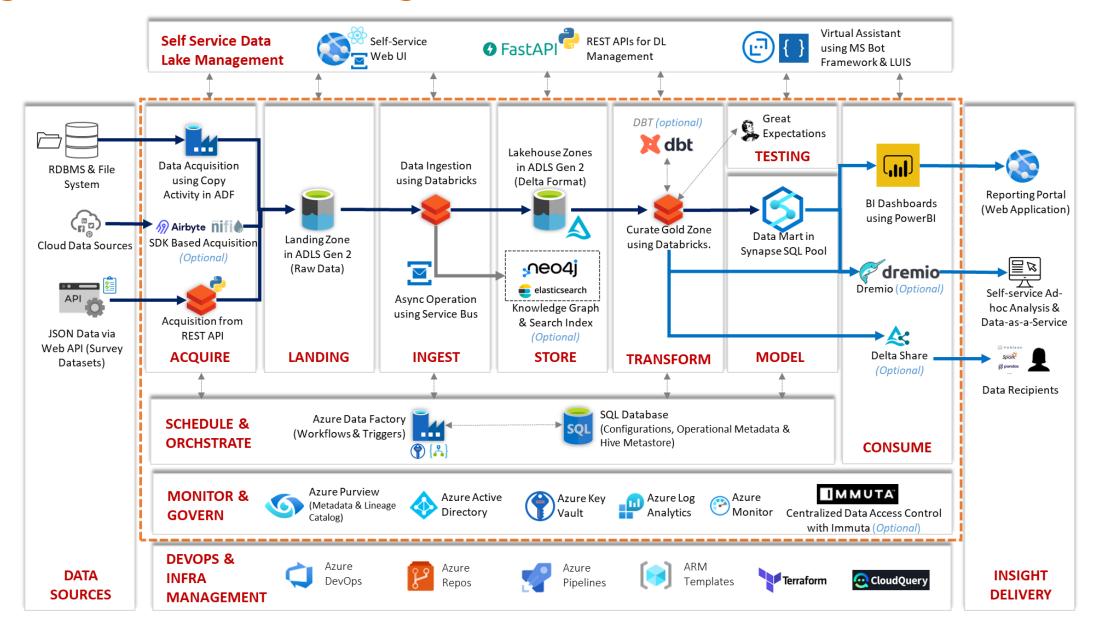
How Self-Service is Enabled in Tiger Data Fabric



Automation and Self-services capabilities added by Tiger Data Fabric

Typical Cloud
Data Lake Platform
Implementations

High Level Architecture - Tiger Data Fabric (Azure)



Integrated Knowledge Graph Based Data Catalog

