SmartAPIStreamX - Phase 1 Documentation By Paresh Ranjan Rout

SmartAPIStreamX is a metadata-driven, dynamic API ingestion framework built using Azure Data Factory. This document outlines the architecture, datasets, linked services, and steps implemented for Phase 1.

# 📌 Pipeline: pl\_api\_ingestion\_controller

The pipeline performs metadata-driven REST API ingestion with support for incremental loads and watermarking.

## 🔗 Linked Services

1. ls\_http\_generic - For non-auth REST APIs

2. ls\_http\_token - For APIs requiring Bearer token in header

## 📦 Datasets

1. ds\_generic\_api - For public APIs (param: relativeUrl)

2. ds\_token\_api - For token-protected APIs (param: relativeUrl)

3. ds\_metadata\_json - Loads metadata config from JSON

4. ds\_last\_load\_json - Reads last watermark from JSON

5. ds\_sink\_bronze - Writes raw API data into ADLS Bronze Zone

6. ds\_watermark\_template - Used to overwrite watermark file

## 🧾 Metadata Files

• api\_metadata.json - Controls API selection, authType, sink path, incremental field, etc.

• last\_load.json - Stores the last successful load timestamp

## 📊 Control Flow

1. Lookup api\_metadata

2. foreach\_api loop

3. lookup\_last\_load

4. set\_current\_load (using @utcnow())

5. switch\_authType → none | bearer | queryKey

6. copy\_generic\_api\_data | copy\_token\_api\_data | copy\_querykey\_api\_data

7. copy\_update\_lastload\_json - Overwrites last\_load.json with new value

## ✅ Incremental Logic

• Checks supportsIncremental = true in metadata

• Replaces {{last\_load}} in URL using lookup\_last\_load

• Updates watermark after success using set\_current\_load

## 🧠 Notes

• REST dataset used with HTTP linked service for flexibility

• Bearer token APIs dynamically configured via metadata

• Sink path and file name set dynamically using item().sinkPath

Generated on: 2025-07-15 18:14:43 UTC

A screenshot of a computer

AI-generated content may be incorrect.