

Morgan Stanley



Using API's to open up and integrate your mainframe environment and applications

Calder Sacks, Executive Director

February 2026

Bio Page

CALDER SACKS

Executive Director

Enterprise Z

Phone: +1 212 276-1952

Calder.Sacks@morganstanley.com

Years of Experience: 33 Years

Years at Morgan Stanley: 26 Years

Biography

Worked in many roles within the mainframe technology landscape. Started as a Natural mainframe programmer and moved into the Adabas DBA role. Lead the database engineering team at Morgan Stanley for 10+ years and now lead the team working on Mainframe integration strategy and technologies for our Cobol/DB2 mainframe environment.

Been working on technologies between mainframe and distributed since 2003.

- One of the first companies to call Java RPC services from our mainframe Batch and CICS processes back in 2005
- Setup Sybase access from the mainframe.
- Implemented SQL & ACI access to mainframe Adabas and CICS
- Running Kafka Connectors running on the mainframe to distributed Kafka

Our goal is integrating mainframe to be just another platform and to leverage what has already been built where possible

Table of Contents

Why API's for the mainframe ?

Types of API's

API Gateways and providers

Summary

Why API's for the mainframe ?

- Get to the data at the source of the data
- Leverage what has already been built without reinventing everything
- Creating a set of standard interfaces for developers to discover and use to stitch together new workflows and processes is key to many organizations
- With the advent of AI gateway services and discovery of MCP servers, API's and micro services will become more important as Agentic AI is used to build new processes and workflows

Types of API's

- There are many API's that have been built over the years to bridge applications
- Many are custom built and the standards and interfaces agreed upon by a few development teams
- Direct connection to applications on mainframes have generally been purpose built

DB2 REST services

- Taken all of our DB2 Stored procedures and exposed them as REST services

CICS REST services

- Creating REST services for access to our CICS transactions

ZOWE and the mediation layer

- Leveraging ZOWE and the mediation layer to expose and use mainframe components

Vendor interfaces (Sysview etc)

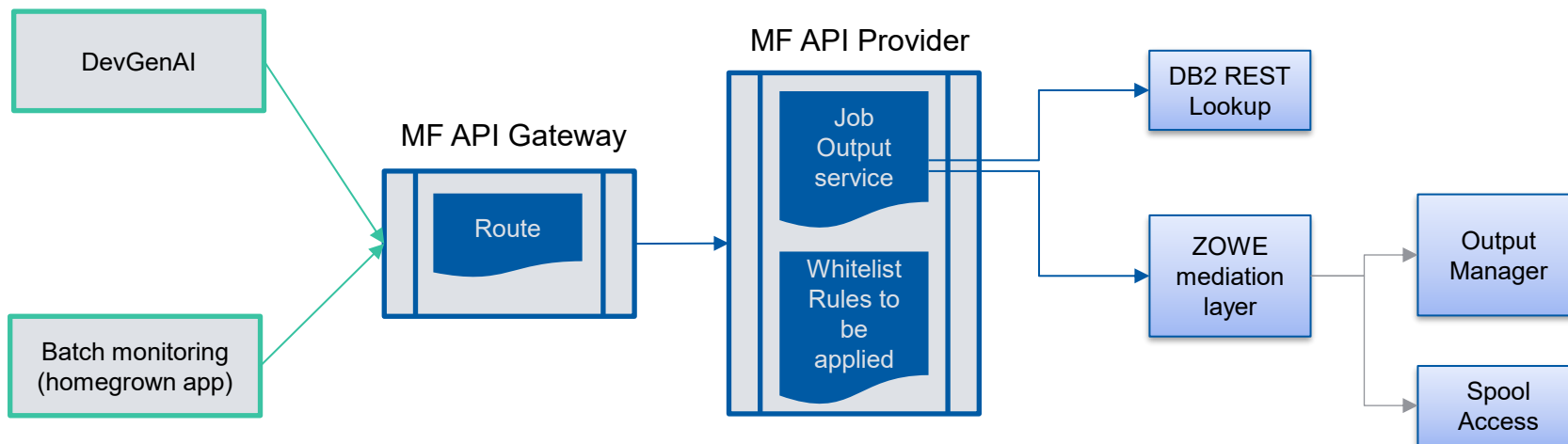
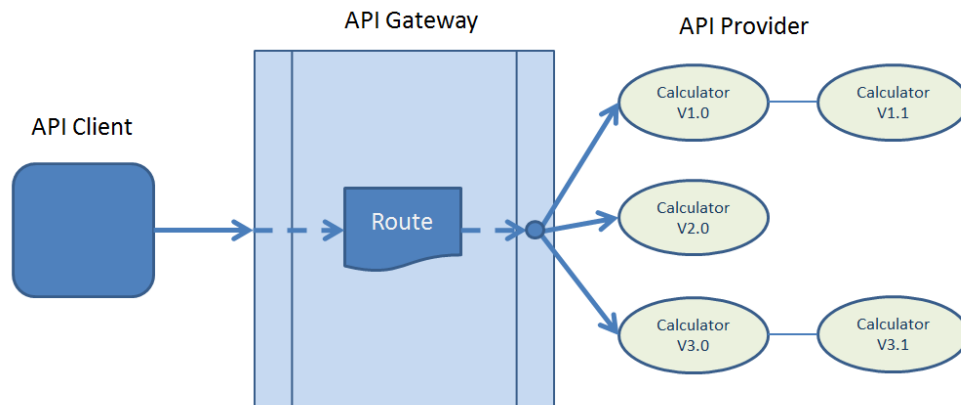
- Using API's provided by vendor products

Python to build services on the mainframe

- Building scripts and services to allow interaction with the mainframe

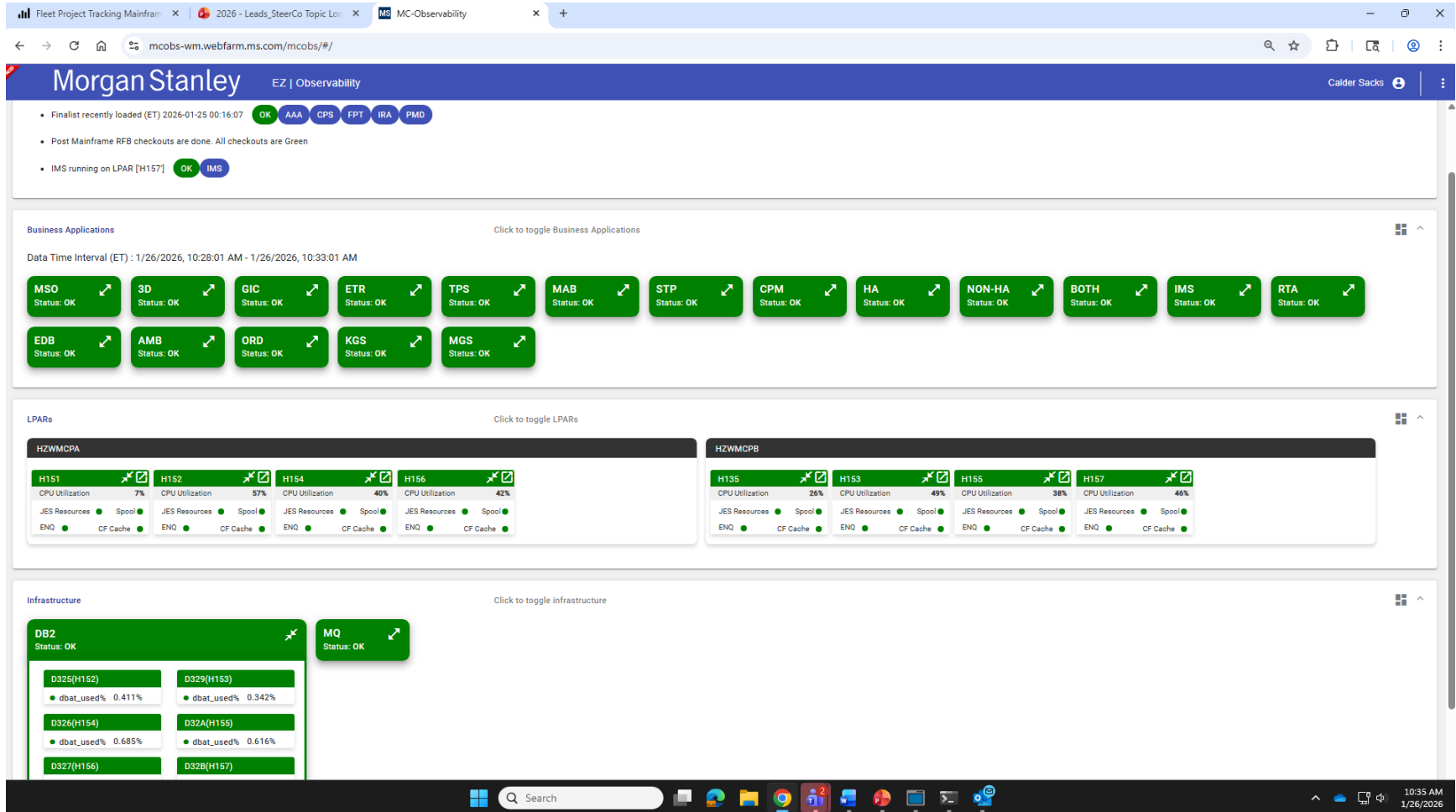
API Gateways and providers

- Gateways can be used in many contexts
 - Allowing access to different versions
 - Controlling volume flow
 - Simplifying connectivity
 - Ensuring security and compliance
 - Reducing complexity for consumers
 - Combining sources to give a single result



Building integrated dashboards

- Using multiple API's across Sysview, zOSMF and Splunk
 - Leveraging data directly from mainframe sources where possible



Summary

- Leverage microservices technology and REST services across products
- Plan to integrate with the larger Services catalog in order to make these API's and services discoverable
- Plan for mechanisms to managing these API's and services in terms of
 - Version control
 - Performance and connectivity
 - Access control

Questions ?