Data Engineering and the Rise of the API

The changing landscape of data sources and systems interaction.

Data left
unattended

Speaker Info

John Q. Martin

Nearly twenty years of experience working with data platform and cloud technologies. Working with SQL Server, Azure, AWS, Snowflake and Databricks to deliver OLTP and Analytics solutions.

/johnmart82

john@jqmartin.co.uk

/in/johnqmartin

https://blog.jqmartin.co.uk

Data left unattended

What are we discussing?

Looking at how we used to get data, what APIs were used for and how that has changed. Then looking forward and thinking about what might change, and what the needs of a data engineer are for APIs.

How we got Data

Working with direct database access to get data via SQL queries or through views and direct object mapping in specialist tools.

Screen scraping from terminal emulators from mainframe or things like AS400 based ERP platforms.

Ingesting files which were produced by applications on a schedule or manually by people running them manually.

What were APIs used for?

APIs were mainly used for transferring data between applications in a transactional pattern, normally using a service bus or messaging system.

Standardised formats under the banner of EDI used to transfer data between organisations and systems.

How we get data from applications

Still, some direct access to databases where software is installed on owned systems, or by agreement with suppliers.

Externally hosted or SaaS applications via APIs.

SOAP, REST, GraphQL

REST APIs

Introduced by Roy Fielding and colleagues via his dissertation in 2000.

Uniform interface, Stateless, Client-Server, Cacheable.

Multiple data response options including XML and JSON.

Data left unattended

GraphQL

Developed by Facebook in 2012, made public in 2015 as a language to query APIs.

Pull back the information that you need from the API and leave the stuff you don’t. Solves the over/under-fetching.

GraphQL is strongly typed as opposed to REST which is weakly typed.

Data left unattended

Demo

How do we interact with APIs?

Considerations for API sources

Authentication and Authorisation mechanisms. Different APIs will support different options and even then, there will variations on OAuth processes.

Rate Limits for APIs exist to help minimise impact of DoS attacks as well as helping maintain a consistent level of performance at the service level.

Pagination of results is common when dealing with larger volumes of data.

Data left
unattended

What does the future hold.

More use of GraphQL for data engineering activities, supporting more granular queries than API endpoints.

APIs on analytics services like Microsoft Fabric which are used for deployment of artefacts as opposed to ARM/Bicep for Synapse or similar.

When developing new APIs and capabilities, take into account the need for bulk operations.

Data left unattended

Summary

Read the docs, each API is different and will have its own nuance around authorization, token expiry, rate limits etc.

Understand where you plan to do the transformations once you have the data, at landing or in platform processing.

Data left unattended