Not all APIs are created equal

Curtis Evans
Managing Director EMEA
Jacobi

Investment technologies are quick to boast of their API credentials. But a close inspection reveals that many leave a lot to be desired.

Having an API is table stakes for investment technology in today's world. An Application Programming Interface (API) defines the methods and data formats with which other applications can request and exchange information with the API platform. APIs can be used to build processes that span multiple platforms, efficiently distribute and process data across many systems, and achieve scalability not possible with human interactions alone.

For investment managers, APIs are important systems that allow different platforms and technologies to coexist and connect seamlessly. Unfortunately, it is not always that simple; the reality is not all APIs are created equal. There's enormous dispersion in how APIs are implemented across the industry. There are numerous platforms that rely on legacy systems yet there are also many new technologies and paradigms rapidly gaining share as the field develops. In addition, there are differences due to how providers may choose to balance their system's security, reliability and profitability.

CTOs and COOs need a strong grasp of what limitations an API may impart on users. A poor choice of API can result in frustrated, unproductive users. So what are some of the signs that an API may not be so useful?

First, limitations and restrictions on usage.

Many incumbent platforms impose restrictions on the usage of their API. This may be in the form of usage limits such as restricting the number of actions in a given timeframe or limits on the raw size of data transfers.

These limitations can present real problems for both developers and investment managers. It's hard to anticipate exactly where limits may be breached, especially in advance of development work commencing. The financial industry has many workloads that are "bursty"; large amounts of data may need to be quickly processed with no tolerance for faults or delays due to rate limits.

Often, these restrictions are used by vendors to manage resources across multiple clients. Allowing a single client to submit an unlimited number of requests can potentially result in a poor experience due to slow speeds or connectivity issues. However this risk can often be mitigated in less disruptive ways. It is important to balance the reliability of the system against the needs of the client.

More perversely, there may be charges on the basis of the volume of calls or data transferred. In some cases, endpoints may be gated behind paywalls. This prevents firms from making best use of an API - undermining the flexibility they were created for in the first place.

Second, thinking of APIs solely as a data transfer mechanism.

The APIs of many systems only concern themselves with data flows in a prescribed format. Yet if that's their only focus, it's barely an improvement on the days of batch file transfers. APIs should help the speed of data flows and add flexibility to accommodate different data formats, but that isn't all that they can do.

Good APIs should allow developers to programmatically perform the majority of actions and tasks that are otherwise done in the platform by users. For example, making changes to a portfolio, running analytics, generating displays and exporting a report. When such tasks can be performed programmatically via API at scale, a world of possibilities opens up to industrialize, connect and automate existing and entirely new processes.

Third, poor user documentation, validation and reliability.

For developers using an API, the documentation to describe how it's used can often fall short. Frequently seen issues are limited information on higher level concepts, few practical examples or inconsistent documentation. When testing work, developers may find validation errors are unclear or worse, missing entirely.

In the case of opaque error messages, users can be inhibited from self-diagnosing problems resulting in wasted time. On the other hand, missing validation can also result in operations using invalid or incomplete data. If this data is used in important contexts, users can unknowingly be basing their decisions on faulty data.

When in regular operation, ensuring reliability in the forms of uptime or scaling large requests jump to the top of developer API complaints. Too often there is a gulf between those who select vendor technology versus the developers left to use it.

Fourth, not facilitating enough third party connections.

Despite all the talk of 'open architecture', the industry has a long way to go until there is widespread integration across systems, both incumbent and new. Yet third-party API integrations are hugely valuable to investment managers. It negates the need to manage connections in-house, saving precious resources and cost, and can make a system much more useful.

CTOs and COOs should ask their vendors why there are not more connections. If you look behind the marketing gloss, you will likely find that both technical and commercial obstacles are slowing progress.

Many systems in the industry are "old". This may mean they utilize obsolete technologies, or come with many years of accrued technical debt and embedded business logic that only a select few can understand. Either situation can make integrating and developing new connections slow and difficult.

The bigger elephant in the room is that for some incumbents there is no compelling commercial incentive to embrace open architecture given the cannibalisation risks. Newer vendors tend to see only an upside to more open architecture. It can be difficult to balance the risk of competitors achieving competitive advantage due to exposed implementation details, versus the benefits of extending the functionality of the platform with these integrations.

Prioritizing technologies that put APIs first.

For CTOs and COOs that aspire for an integrated investment ecosystem, they must prioritize technologies that truly put APIs first. Old inflexible systems may boast of having an API but miss the true value of having a more open architecture. Choosing to "Play it safe" by sticking with these systems may end up being a constraint. Indeed, many of those incumbents need a complete overhaul of their underlying technology to deliver on the API flexibility that the industry needs. The risk is that this may be years away, or worse, a pipe dream.

About Jacobi

Jacobi was founded in 2014 with a vision to transform technology used for multi-asset portfolio design, analytics and client engagement. Jacobi provides its services to top-tier investors across the globe with a client base representing assets under management of over US$7 trillion. Its award-winning technology has its roots in institutional investment management and uniquely incorporates a market leading software development kit. This allows firms to build their own models, tools and applications on the platform.

Contact

United States/Canada
Liz Gartsman
+1 (757) 510-1330
liz@jacobistrategies.com
www.jacobistrategies.com

Australia/Asia
Rebecca Chong
+61 (0) 480 350 279
rebecca@jacobistrategies.com
www.jacobistrategies.com

Europe/Middle East/Africa
Curtis Evans
+44 79-2873-1094
curtis@jacobistrategies.com
www.jacobistrategies.com

www.jacobistrategies.com