

Top API Trends That Will Change The World Beyond 2021

SEPTEMBER 2, 2020 BY SHYAM PURKAYASTHA <[HTTPS://BLOG.API.RAKUTEN.NET/AUTHOR/SHYAM/](https://blog.api.rakuten.net/author/shyam/)>

2020 was a watershed year, as the highly contagious Coronavirus forced people worldwide to avoid all forms of physical touchpoints. This has made the world more connected and various channels of online touchpoints have snowballed like never before in history. But what does this shift present for *API trends*? Some experts believe that the pandemic presents an opportunity to introspect [new ways that APIs can be used](https://cloud.google.com/blog/products/api-management/top-5-trends-for-api-powered-digital-transformation-in-2021) <<https://cloud.google.com/blog/products/api-management/top-5-trends-for-api-powered-digital-transformation-in-2021>> not only for security but also offer better user experiences while on these platforms, yet others are less optimistic about how such large-scale change will affect things moving forward into 2030

However, APIs are already ruling the world of enterprise software. They enable countless use cases around cloud technologies. Their role is also pertinent in achieving modularity within distributed computing applications. Therefore innovation around cloud computing plays a pivotal role in **API trends**.

This blog post is an amalgamation of various API trends resulting from these introspections and the continuous innovation around the cloud. Let us unravel these trends based on the critical areas of API-centric application deployment.

[<https://english.api.rakuten.net/enterprise?s=ctab>](https://english.api.rakuten.net/enterprise?s=ctab)

Table of Contents [hide]

- 1 Trends in API Development
- 2 Trends in API Technology
- 3 The emergence of API Marketplace
- 4 Trends in API Testing
- 5 Trends in API Usage
- 6 Trends in API Management
- 7 Future of APIs Beyond 2020
 - 7.1 Share this:

Trends in API Development

There is good news for all API developers and publishers. The effort of developing an API is getting progressively easier. It is the result of standardization and improvements in tooling. At the same time, APIs are called upon for solving more complex problems. In effect, their underlying architecture and hosting infrastructure is becoming complex too.

- **OAS3 (Open API Specification version 3):** OAS <<https://www.openapis.org/>> version 3 promises to simplify the API specification. It brings in more security features and reusable components in the API definition. It makes life easier for API developers, testers, and APIOps teams. With these improvements, we are moving towards a homogeneous API development toolchain. This approach offers better control over the entire lifecycle, from design, development, publishing, operationalization, and consumption to APIs' retirement.
- **Low Code API Development:** Low code development platforms are cropping up across many technology domains, such as AI and IoT. APIs are not far behind. IT managers and system administrators can design the API business logic with a low code platform without getting into the nitty-gritty of coding. These platforms help in the design and deployment of APIs with less effort for automating business processes. They provide a UI interface to design the API development and

deployment workflows visually. This approach makes them intuitive to use and suitable for people who do not fancy themselves as coders but want to get the job done quickly.

- **Event-Driven APIs:** APIs are moving beyond the two-way synchronous communication mechanism between a client and a server. With the increasing adoption of data-driven and notification-centric applications, APIs have adopted newer communication patterns. With event-driven APIs, clients no longer poll the servers for data or state changes of the application. In turn, the server sends asynchronous events back to the client upon any state change that the client may be interested in. This pattern is contrary to the REST architecture, where the client is responsible for maintaining the application's state.
- **Service Mesh:** With their increasing adoption, APIs are envisaged as part of an API product or ecosystem, offering a suite of related APIs. In such cases, the underlying business logic and semantics have many things in common across APIs. This begs the question of the reusability of the business logic and consistency in the API behavior. A service mesh architecture allows API developers and infrastructure teams to deploy internal bridges between the underlying microservices environment that hosts the APIs. In this way, API developers can tap into common functionality hosted on an internal service or another API. This approach fosters better code reuse and modularity of the entire API suite.

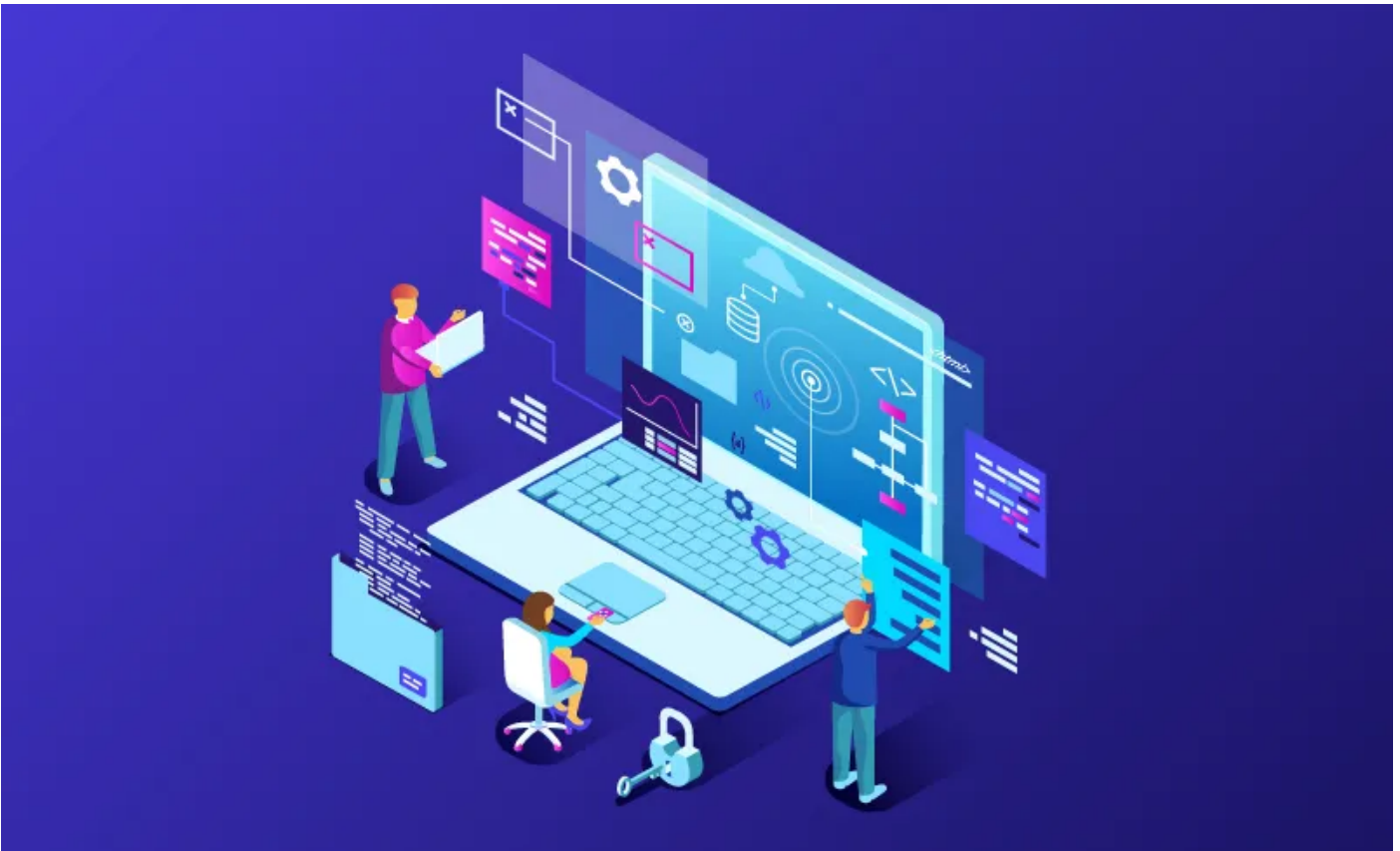
Explore Enterprise Hub

<https://english.api.rakuten.net/enterprise?s=ctab>

Trends in API Technology

APIs have been around for about two decades. That's a long time. However, for the most part, nothing has significantly changed under the hood. If you think of the underlying protocols driving the APIs, it is HTTP. Further, if you consider the architectural patterns governing the design of APIs, it's primarily REST.

Nevertheless, the trends indicate a shift towards better protocols and newer architectural patterns for supporting a better user experience in the future.



- **HTTP 3:** HTTP has been the de facto application layer protocol enabling the APIs. It has undergone a few enhancements from HTTP version 1 to 2, but things have been slow. But there has been an upsurge in recent years. With the upcoming standardization efforts for HTTP 3 <https://www.ably.io/topic/http3>, the existing transport protocol, TCP, will be replaced with QUIC. By design, QUIC offers a better mobile Internet experience.
- **WebSub:** WebSub is a protocol that derives its functioning from the Publish-Subscribe pattern <https://www.ably.io/topic/pub-sub> of distributed messaging across multiple targets. It is ideal for building event-driven APIs. Such APIs are increasingly becoming relevant for many data-driven and notification-centric applications built on the SaaS model.
- **GraphQL:** GraphQL <https://graphql.org/> has been around for a few years now. Many API developers are familiar with it. It is a concept which defines a new way of interacting with APIs using a query and manipulation language. It is more efficient than REST. It can fetch data in a compound form with one API request, where REST requires multiple API requests. GraphQL augurs well with event-driven APIs. Therefore in the coming years, we will see GraphQL being adopted more, eventually displacing the REST style of API definition.

The emergence of API Marketplace

An API marketplace is an online platform that lets developers find and use APIs. It also gives companies the opportunity to make money by allowing them to create a subscription plan for their product or service, usually through monthly fees (or other arrangements). The key benefit of listing your company's API on this type of site is it provides you with more exposure among potential clients who are looking for what you have in mind.

Rakuten RapidAPI is a leading API marketplace. Our platform allows API providers to ease their go-to-market journey. Rakuten RapidAPI API marketplace provides a 360° solution across the entire API management value chain.

We should expect to see exciting development in the API marketplace segment this decade.

Trends in API Testing

Every API performs a well-defined function. Under the hood, this translates to various technical and performance parameters. They must be idempotent across all API invocations. All of these aspects need thorough testing.

However, beyond this, APIs have a larger purpose in attaining particular business objectives for the end-users, the API consumers. This is API user experience. It adds a few additional dimensions for testing.

- **Developer Experience for APIs:** Since most API consumers are developers, providing an enhanced developer experience is the key to increased API adoption. Therefore, API testing must also include developer experience test plans beyond the usual software testing. Testing for developer experience is mostly subjective. It is a good practice to involve multiple testers to rate the different API parameters that affect the overall developer experience. Some of the critical elements of good developer experience revolve around intuitiveness, structuring, and consistency. An intuitive API elicits a better turnaround time for developers to get started with it. Well-structured API

documentation is easy to follow. Consistency leads to improved handling of the API calls, eventually leading to better readability and code robustness.

- **Testing for Newer Architectural Patterns:** The REST architecture is synchronous in nature. With the

rise of event-driven and streaming APIs, the trend is shifting towards asynchronous API operations. Therefore, API testers have to develop the skills for testing asynchronous APIs. Thankfully, there is a standardization effort underway to define the specification for such APIs. The AsyncAPI consortium <<https://www.asyncapi.com>> is working towards this direction. It is adapted from the OpenAPI specification and adds specific extensions for defining the server-side event parameters.

<<https://english.api.rakuten.net/enterprise?s=ctab>>

Trends in API Usage

From humble beginnings as data collectors, APIs are now evolving to spearhead data-driven applications. They are becoming the interface for data exchange. In some cases, APIs are the custodians of data for a larger social purpose.

- **Crowdsourced APIs:** For a long time, APIs were designed to work with an organization's internal data. However, some use cases, such as training a machine learning model, rely on a vast and diverse external data corpus. A better approach is to adopt crowdsourcing tactics to let the API consumers contribute data in such cases. This approach also helps in sharing data through APIs for the public good. This trend is emerging in 2020 concerning Covid-19. Several universities, non-profit organizations, and individuals have launched COVID-19 related APIs [≤https://english.api.rakuten.net/collection/coronavirus-covid-19>](https://english.api.rakuten.net/collection/coronavirus-covid-19) to provide pandemic-related analytics, statistical data, country-wise reports, and other research findings.
- **Inter Technology Integration:** APIs have a larger purpose in unifying digital transformation initiatives across an enterprise. They play a critical role while integrating disparate systems supporting different technologies. For instance, APIs play a significant role in tying together the

various steps in an RPA (Robotic Process Automation) driven process automation. APIs also play a part in integrating hardware assets, with the help of IoT. Similarly, APIs are aiding artificial intelligence systems by becoming the bridge between the human and machines. The adoption of newer technologies and architecture patterns beneath the APIs makes all this possible. Such integrations will continue to evolve for other emerging technologies, such as Blockchain, Web3.0, and beyond.

Trends in API Management

The [API management <https://blog.api.rakuten.net/api-management/>](https://blog.api.rakuten.net/api-management/) market has gained traction in the recent past, thanks to the need for stringent control on APIs. However, the scope is not limited to usage and monitoring alone. [API management <https://blog.api.rakuten.net/essential-components-of-api-management-tech-stack/>](https://blog.api.rakuten.net/essential-components-of-api-management-tech-stack/) now governs all aspects of an API's lifecycle.

Notwithstanding the need for managing APIs, there is also a growing desire to automate the API management processes. New features around automation and hyper customization of APIs will be the latest trend in [API platforms <https://blog.api.rakuten.net/api-platform/>](https://blog.api.rakuten.net/api-platform/).

- **Adaptive API Governance:** At its core, API governance <<https://blog.api.rakuten.net/api-governance/>> is about setting up processes and policies to manage the API life cycle. Historically, these processes were set statically, as rules and exceptions. With the growing use of APIs within an enterprise, these configurations are always changing. Added to that, the dynamics and seasonality of API usage make it even more cumbersome. An adaptive governance approach is the need of the hour. It adapts to different business contexts to devise appropriate governance styles. Backed by artificial intelligence, it can tweak those governance styles based on API usage patterns and historical data.
- **Custom API Platforms:** When it comes to building API platforms <<https://blog.api.rakuten.net/api-platform/>>, individual APIs with their specific capabilities become the building blocks. As an example, you may want to build an API platform for machine learning-based recognition. By

combining the best APIs that individually perform face and optical character recognition, you can build a custom API platform. What if you get an image with a face as well as text in it? Instead, you would want this to be handled in a single API call, rather than triggering two separate calls, each for face and character recognition. With a custom API wrapper along with a query language like [RapidQL <https://rapidql.com/>](https://rapidql.com/), you can build a new API from existing ones. That's the power of custom API platforms. We are going to witness more innovative ways of merging, aggregating, or reusing the existing API semantics to build new APIs, that too, without rewriting the underlying logic or deploying the API afresh.

[Rakuten RapidAPI Enterprise Hub <https://english.api.rakuten.net/enterprise>](https://english.api.rakuten.net/enterprise) is a one-stop solution that enables enterprises to define and implement successful API strategies. Whether they want to implement governance policies for discovering and connecting to internal APIs, build their API ecosystem, or subscribe to third-party APIs, the Enterprise Hub gives them total control over the API governance, access, and monitoring of the API lifecycle. It also leverages the Rakuten RapidAPI marketplace, the world's largest API marketplace with over ten thousand APIs.

You can read more about the Enterprise Hub [here <https://blog.api.rakuten.net/enterprise-hub-introduction/>](https://blog.api.rakuten.net/enterprise-hub-introduction/). If your enterprise needs assistance in setting up an easy-to-use and powerful API management platform, send us an [inquiry](https://api.rakuten.co.jp/en/inquiry/service_inquiry.html). [<https://api.rakuten.co.jp/en/inquiry/service_inquiry.html>](https://api.rakuten.co.jp/en/inquiry/service_inquiry.html) right now.

[<https://english.api.rakuten.net/enterprise?s=ctab>](https://english.api.rakuten.net/enterprise?s=ctab)

Future of APIs Beyond 2020

From being a user-to-system interface, APIs have come a long way. Now APIs are powering inter-subsystem interfaces as well as system-to-system interfaces. How far can we possibly stretch it?

We can envisage the future of APIs in two ways.

APIs of the future will drive more intelligent forms of communication. This trend is evident, given the fan-out pattern of communication in many distributed applications. So you can imagine smart, event-driven APIs fanning out information to multiple systems.

Yet another area of impact for APIs is in the advent of the future Internet

[<https://en.wikipedia.org/wiki/Future_Internet>](https://en.wikipedia.org/wiki/Future_Internet). With multiple research initiatives to improve the Internet, APIs will find many synergies. One such initiative is the concept of Named Data Networking [<https://en.wikipedia.org/wiki/Named_data_networking>](https://en.wikipedia.org/wiki/Named_data_networking) (NDN). NDN proposes a different semantics for Internet packet format with unique names instead of IP addresses to identify endpoints and

applications. Although still in its infancy, NDN and similar ideas want to radically transform the Internet's addressability to make it more user-friendly, based on names. APIs with their name-based URL and resource formats have the potential to become the higher-level abstractions for these underlying

network semantics.

In this way, we can envision the API trends evolving with the future of the Internet. APIs will be the unifier of all diverse technologies and systems, achieving truly distributed, inter-system communication. They will form the building blocks of the communication fabric for the Internet of the future.

Let us witness the future of APIs unfold over the next few years.

[<https://english.api.rakuten.net/enterprise?s=ctab>](https://english.api.rakuten.net/enterprise?s=ctab)

5 / 5 (1 vote)

Share this:

 [<https://blog.api.rakuten.net/api-trends/?share=facebook&nb=1>](https://blog.api.rakuten.net/api-trends/?share=facebook&nb=1)

 [<https://blog.api.rakuten.net/api-trends/?share=twitter&nb=1>](https://blog.api.rakuten.net/api-trends/?share=twitter&nb=1)

 [<https://blog.api.rakuten.net/api-trends/?share=linkedin&nb=1>](https://blog.api.rakuten.net/api-trends/?share=linkedin&nb=1)

 [<https://blog.api.rakuten.net/api-trends/?share=reddit&nb=1>](https://blog.api.rakuten.net/api-trends/?share=reddit&nb=1)

© 2021 Rakuten RapidAPI. All rights reserved.