COMMSCOPE®

iTRACS® Converged Infrastructure (iTRACS CI)—An Open API approach to unlocking your data

Contents

Summary

The ProgrammableWeb directory of public application programming interfaces (APIs) has recorded a total of 22,000 APIs as of June 2019. Their findings have shown an increase of over 30 percent over the previous four years (Santos, Wendell. "APIs Show Faster Growth Rate in 2019 than Previous Years." ProgrammableWeb, 12 July 2019). A similar trend in growth is also prevalent within organizations. APIs enable more efficient development as well as consumption by external organizations and developers. With such rapid growth, many organizations have turned to various frameworks and formats to help manage, build and track their APIs. One such method is the OpenAPI Specification (OAS), formerly known as Swagger, and developed by Tony Tam. In recent years Swagger has split into Swagger and the OpenAPI Specification, the former being a software tool to design, build, and document RESTful web services.

These key benefits make up OAS and will be discussed throughout this paper: ease and speed of data access; lower development time and cost; less need for qualified SMEs; human readable, more accessible, RESTful.

What is an API?

API is the acronym for application programming interface, which is a software intermediary that allows two applications to talk to each other. We use APIs in our everyday life without realizing it. For example, browsing the Netflix catalog for your Friday movie night is passing a request to a server which is then interpreted by the receiving API, and the appropriate data is returned: in this case, a film of choice. As you might imagine, APIs can perform very complex tasks for us. In the development world complexity is a bit of a curse, as the more complex a task is, the more complex the underlying code—or in this case the API—may become. Over time this process can become a large burden on organizations as their API becomes more and more bloated with features and, even worse, lacks proper documentation or design standards. Well, in 2011, the first release of the Swagger specification was released, later to be known as the OpenAPI Specification.

What is the OpenAPI Specification?

"The OpenAPI Specification (OAS) defines a standard, programming language-agnostic interface description for HTTP APIs, which allows both humans and computers to discover and understand the capabilities of a web service without requiring access to source code, additional documentation, or inspection of network traffic. When properly defined via OpenAPI, a consumer can understand and interact with the remote web service with a minimal amount of implementation logic. Similar to what interface descriptions have done for lower-level programming, the OpenAPI Specification removes guesswork in calling a service" (OAI/OpenAPI-Specification: The OpenAPI Specification Repository).

In other words, the OAS is a standard that developers can use to design their APIs. On the other end, the OpenAPI provides a standard in which developers can consume these APIs without understanding all the intricate code that powers the API.

1 Ease and speed of data access

The sole role of any API is to access data. Typically this is data from distinct services or applications. APIs inherently attempt to speed up data access by creating a standard within the application to do just that. OpenAPI takes this a step further by providing these benefits not just to those developing the API but to developers that may be working with said API and other applications or APIs jointly.

By leveraging OAS, developers know—without seeing a single line of code—what to expect to find and where to find it. This benefits the "discovery" or research phase many developers use to learn about a new technology. OAS also allows a developer to seamlessly generate documentation that details how the API operates. This dramatically shortens time to access data behind the API and the effort/prerequisite knowledge needed to do so.

2 Less development time and cost

As we might expect, standardization in any industry has major cost saving benefits as the requirements are often static through this standardization. Those that work with standardized technology and the overlying organization can appropriately predict and replicate processes needed to fulfill a task. This is also very true in software development, which is often a world where time scales are hard to predict and many unknowns can exist. Referencing the statistic quoted by ProgrammableWeb: Of the 22,000 public APIs, if each were built using their own standard, there would be countless hours lost simply in trying to understand how these APIs function.

When discussing APIs, the cost savings can apply to those that plan to consume the API (e.g., other developers, end users, external organizations). These savings can make an application and its API—or even the organization as a whole—more attractive for potential users, developers. This can fuel greater adoption of a given technology that incorporates an API.

With OAS, API owners can generate human-readable documentation with little to no overhead. Strong documentation is often the bane of the software development lifecycle, as every code change risks impacting how a software feature operates and thus may require updates to documentation for said feature. Over the life of an API, many hours of documentation are now automated and immediately ready for consumption using OAS and OAS tools such as Swagger.

3 Accessibility and SMEs

In many IT fields, subject matter experts (SMEs) are key to the success of their respective subject. Strictly speaking in terms of APIs, OAS can often remove the need for a dedicated SME or, at the very least, reduce the time needed to become an SME. A developer who understands the specification is agnostic to what the API is doing behind the scenes; rather, they often are only concerned with what is contained in the API. With the provided benefit of great documentation via OAS tools, no longer are developers and organizations limited by the amount of time and resources available to understand a service with which they wish to interface. Rather, in a very short time, the developer is very much an SME without the traditional overhead or experience.

Organizations often have to vet applications and API offerings or possibly involve themselves in long sales cycles to ensure said software fulfills their needs. This process no longer requires deep-rooted development knowledge and can often be demonstrated on short notice with clearly documented examples and explanations of what an API is offering. This is key to providing necessary insight into how an API may benefit an organization without all the sales fluff.

4 RESTful API

At the heart of it, a digital twin is data. Data should be able to be incorporated into existing business workflows as easily as possible. It is much easier for an organization to implement and begin drawing useful information from a digital twin when they can access and use that data in other areas of the business workflow.

Having a common, open, well-documented, and easy-to-use API achieves this goal. Ideally, the API should follow standards already used commonly throughout the industry, such as a RESTful API. A well-documented RESTful API should come with examples that make it simple to retrieve, update, and delete data against a digital twin.

Being able to extract the necessary data held against a digital twin allows that data to be loaded into other existing business workflow applications. This removes the effort that may be required to implement entirely new software.

Extracting and adding the data to existing business systems through straightforward APIs reduces the overhead of additional time and money spent on training and workflow changes.

Additionally, simple, well-documented APIs make it easier to extract data that is useful for reporting purposes. When the data is easier to extract and manipulate into the desired format, reporting becomes much easier, takes less time, and provides more opportunities for spotting areas of improvement.

Learn more about how digital twins and how iTRACS can be one.

Appendix

OpenAPI Initiative. "OAI/OpenAPI-Specification: The OpenAPI Specification Repository." GitHub, OpenAPI Initiative, https://github.com/OAI/OpenAPI-Specification

Santos, Wendell. "APIs Show Faster Growth Rate in 2019 than Previous Years." ProgrammableWeb, 17 July 2019

Red Hat. "What is a REST API?" Red Hat, 8 May 2020, https://www.redhat.com/en/topics/api/what-is-a-rest-api.

Trademarks

All trademarks identified by ® or ™ are registered trademarks or trademarks, respectively, of iTRACS LLC or CommScope, Inc. This document is for planning purposes only and is not intended to modify or supplement any specifications or warranties relating to iTRACS or CommScope products or services. iTRACS and CommScope are committed to the highest standards of business integrity and environmental sustainability.

Further information regarding CommScope's commitment can be found at www.commscope.com/About-Us/Corporate-Responsibility-and-Sustainability

Contact information

Visit our website or contact your local iTRACS representative for more information.

https://www.iTRACS.com

For technical assistance or customer service, visit us at:

https://www.iTRACS.com/CustomerCare

CommScope pushes the boundaries of communications technology with game-changing ideas and ground-breaking discoveries that spark profound human achievement. We collaborate with our customers and partners to design, create and build the world's most advanced networks. It is our passion and commitment to identify the next opportunity and realize a better tomorrow. Discover more at commscope.com.

COMMSCOPE®

commscope.com
Visit our website or contact your local CommScope representative for more information.