Differences Between API Documentation, Specifications, & Definitions

By Janet Wagner

We are often asked about the differences between API specifications and API documentation. So today, we thought we would outline the differences and the importance of API documentation, API specifications, and API definitions.

What is API Documentation?

API documentation is essentially the reference manual for an API – It tells API consumers how to use the API. API documentation is meant for humans, usually developers, to read and understand. Providing documentation that is well-designed, comprehensive, and easy to follow is crucial when it comes to ensuring developers have a great experience with the API. And a great developer experience (DX) means a better chance for API success. If developers can't follow your API documentation, they'll likely find another API with better documentation and would be easier to integrate. Good documentation also helps decrease the time it takes to onboard new API consumers. If developers have all the information they need to use your API, then they don't have to send your company emails, call your customer service department, or submit posts in forums to get help.

Great documentation includes many things, far too many to list here. But some of the things that make for great API documentation include a quick start guide, tutorials, and interactive documentation so developers can try out API calls. There are many tools available for generating and maintaining API documentation. Many of these tools can automatically generate both static and interactive API documentation from an API definition file (more about this later in the article).

API documentation should provide an example of every call, every parameter, and responses for each call. It should include code samples for commonly used languages such as Java, JavaScript, PHP, and Python. Documentation should provide an explanation for each API request and examples of error messages. It's also important that API documentation is actively maintained and always up to date.

What is an API Specification?

API specification is a term that is often used interchangeably with API definition. While these terms have many similarities, they are different entities. An API specification provides a broad understanding of how an API behaves and how the API links with other APIs. It explains how the API functions and the results to expect when using the API. A good example of an API specification is the OpenAPI Specification. You can view the latest version of this specification (3.0.1) on GitHub.

In some ways, the OpenAPI 3.0.1 document is also API documentation, but an API specification explains how the API behaves and what to expect from the API. And the OpenAPI Specification document on GitHub does just that. In the document, there are many API objects, values and parameters, how the objects are called, and what each object does. We also see the relationships between the objects and how each object can be used.

For example, if you look at the Request Body Object section, you'll find information as to what this object does, the descriptions and types for its fixed fields, and request body examples. You will see how the Request Body Object is supposed to function and what to expect when using this function. An API specification is also an indication of how the API is designed and the data types the API supports.

OpenAPI (formerly the Swagger specification) is one of several API specification languages. There's also RAML and API Blueprint. It should be noted that there is momentum towards these API specification languages converging into one API specification language, OpenAPI.

What is an API Definition?

An API definition is similar to an API specification in that it provides an understanding of how an API is organized and how the API functions. But the API definition is aimed at machine consumption instead of human consumption of APIs. An API definition provides information about how the API functions, how it links with other APIs, and the expected results in a machine-readable format. It focuses on defining the API and outlining the structure of the API.

An API definition is often used as a baseline for automated tools. API definitions can be used to generate API documentation, code samples, and SDKs automatically. A few examples of tools that generate API documentation (static and interactive) from an API definition file are SwaggerHub and Swagger Inspector. A few examples of tools that can automatically generate sample code and SDKs from API definitions include REST United and SwaggerHub.

API definitions can also be imported into a mock server for virtual API testing. Among the many tools for mock server and API testing that allow import of an API definition file are SoapUI and SwaggerHub.

An API definition is important in that it can be used to power automated tools that can ensure the success of your API like interactive documentation and SDKs. Some developers even advocate a schema-first API design which means creating the API definition first based on one of the specification languages and then writing the code for the API.

API documentation, API specifications, and API definitions are all related, but they are different entities. And each of these serves an important purpose.

The documentation tells developers and other API consumers how to use the API. After all, how can your API be successful if developers don’t know how to use it?

An API specification provides a broad understanding of the functionality of the API and the expected results. The specification is largely about the design of the API or your design philosophy. API design and functionality are key factors when choosing to integrate an API with an application.

And finally, the API definition is about machine consumption of an API and providing a machine-readable format for use by automated tools like automatic API documentation and SDK generators.

API documentation, API specifications, and API definitions are all key to the success of an API.

Thanks for reading! Looking for more API resources? Subscribe to the Swagger newsletter. Receive a monthly email with our best API articles, trainings, tutorials, and more. Subscribe