What is API?

API stands for application programming interface. As it's a rather complicated concept, let's break it down by looking at each of its parts.

Application

If you have a smartphone, you are well acquainted with what applications are, i.e., the tools, games, social networks and other software that we use everyday.

Programming

Programming is how engineers create all the software that make our lives so much easier.

Interface

An interface is a common boundary shared by two applications or programs that allow both to communicate with one another.

So an API is essentially a way for programmers to communicate with a certain application.

What is API Management?

API management is the process of publishing, documenting and overseeing application programming interfaces (APIs) in a secure, scalable environment. The goal of API management is to allow an organization that publishes an API to monitor the interface's lifecycle and make sure the needs of developers and applications using the API are being met.

API management tools

Deciding on an API Management Tool, you are faced with lots of choices. Available solutions may focus in one or two or cover many of the features discussed above and vary greatly in price. There are tools that were acquired by bigger vendors like Intel or CA or Microsoft. Open source tools are also available. Last but not least, some tools are heavy enterprise focused and other much less so.

Name	Type	License	Stackoverflow questions	Market segment	Strong Points
3scale	Agent, Proxy	Commercial	15	Startups to Enterprises	Wide range of tools
ApiAxle	Proxy	GPL	9	SMBs to Enterprises	
Apigee	Proxy	Commercial	598	SMBs to Enterprises	Powerful Analytics
Axway	Proxy	Commercial	9	SMBs to Enterprises	
CA Layer7	Proxy	Commercial	35	Enterprises	Advanced support for

Name	Type	License	Stackoverflow questions	Market segment	Strong Points
					mobile applications
IBM API Management	Agent, Proxy	Commercial	17	Enterprises	Large Scale, User friendly
Mashape	Proxy	Commercial	106	Startups to Enterprises	Monetization, discoverabilty
Mashery	Agent, Proxy	Commercial	57	SMBs to Enterprises	API strategy services
Microsoft' Azure API Management	Agent, Proxy	Commercial	262	Startups to Enterprises	

Name	Type	License	Stackoverflow questions	Market segment	Strong Points
MuleSoft	Proxy	Commercial	134	Enterprises	Based on proven open source technology, programmableweb
Oracle SOA	Proxy	Commercial	213	Enterprises	Large scale, SOA
Akana (formely SOA Software)	Proxy	Commercial	3	Enterprises	
WSO2	Agent, Proxy	Apache	4421	Startups to Enterprises	Open source

How does an API work?

An API is a messenger that takes requests and tells the system what you want to do, then returns the response back to you.

To explain this better, let us take a familiar example.

Imagine you're sitting at a table in a restaurant with a menu of choices to order from. The kitchen is the part of the "system" that will prepare your order. What is missing is the critical link to communicate your order to the kitchen and deliver your food back to your table. That's where the waiter or API comes in. The waiter is the messenger – or API – that takes your request or order and tells the kitchen – the system – what to do.. Then the waiter delivers the response back to you; in this case, it is the food.

Let us apply the above metaphor to a real-life API example. You may be familiar of the process of searching flights online. Just like the restaurant, you have a variety of options to choose from, including different cities, departure and return dates, and more. Let us imagine that you're booking you are flight on an airline website. You choose a departure city and date, a return city and date, cabin class, as well as other variables. In order to book your flight, you interact with the airline's website to access their database and see if any seats are available on those dates and what the costs might be.

However, what if you are not using the airline's website—a channel that has direct access to the information? What if you are using an online travel service, such as Kayak or Expedia, which aggregates information from a number of airline databases?

The travel service, in this case, interacts with the airline's API. The API is the interface that, like your helpful waiter, can be asked by that online travel service to get information from the airline's database to book seats, baggage options, etc. The API then takes the airline's response to your request and delivers it right back to the online travel service, which then shows you the most updated, relevant information.

As you can see, APIs make it possible to use travel sites—and the same goes for all interactions between applications, data, and devices that we have everyday. These interactions are all powered by APIs, and that is what ultimately creates connectivity.