THE 4 PHASES OF AN OPEN API MANAGEMENT PROJECT ROADMAP

In a previous release of “What the Tech?” we discussed why you should integrate API management into your business strategy. The benefits are plentiful, key among them is that when you open APIs to third-party developers your organization can derive new revenue, reach new audiences, and realize greater innovation.

Despite the clear benefits and successes of an API project, the actual process of opening and managing your API can be challenging. Oftentimes an organization’s goals for the project are vague. Perhaps they want to realize more revenue or reach new audiences. That’s great, but such a large investment of time and resources warrants more definably articulated goals. Without that laser focus, you have nothing to measure or track progress against.

Another important factor for API success is user adoption. You can theorize as much as you want about the market opportunity for your API, but only actual usage data will help you better understand your user’s needs and help you craft an API strategy that is clearly aligned with those needs.

The good news is that a simple and trusted approach, borrowed from product development best practices, can help you diligently determine the value of your API project as well as its addressable market. And that’s a good way to look at it—APIs should be treated as products. So it follows that during the research phase you should:

1. Test your market (often with a limited pilot).
2. Track usage and feedback.
3. Adjust the API model (i.e. your product) based on what your market or user data is telling you.
4. Rinse and repeat. If a feature is not getting traction, redefine your API product, i.e. bundles of APIs.

Let’s assume you have fifty APIs that have the potential to be opened up to various user groups. In all likelihood, a large amount of those APIs will fail to provide much in terms of useful data, content, or business logic. But by implementing a pilot open API project you can quickly determine which are relevant and which can be sidelined (or continuously iterated)—ultimately saving yourself the time and cost of maintaining a bunch of extraneous APIs.

Of course, the research phase is only one piece of the product development pie. Let’s dig a little deeper into each phase of a tried and tested open API project roadmap:

Phase 1: Pilot—Get to Know Your Users

As mentioned earlier, a critical part of any open API project is understanding your users. What APIs are important to them? How are they using them and how often?
The right tools can help with this. Choose an API management platform that offers an inexpensive tier for market testing. As with the early stages of product development, try to keep your investments reasonable during this exploratory phase. You can easily switch to another platform down the line, so don’t get too hung up on the bells and whistles of your selected platform at this point. The most pertinent things to focus on in phase one are your data, content, and business logic, not your choice of platform.

During the pilot, use built-in analytics in the platform to relentlessly track users (where requests are coming from, how long users are coming to your application, the number of requests per day). You may discover that your users prefer a certain authentication method, that requests come from a specific type of device or geography. Twee your APIs and policies accordingly. You might want to consider offering your users free sets of basic APIs on a trial basis to help boost your testing pool.

The data gathered during this phase will tell you where to focus your effort based on which APIs to weed out, which ones are gaining traction, how they’re being used, and so on. It will also help you eliminate under-performing/under-utilized APIs, adjust them, or even mashup your APIs.*

**Phase 2: Define your Open API Model**

Now that you’ve formed a better understanding of your API user base you can begin to package API offerings that meet their needs and address your business objectives.

If the goal of your API is revenue generation then your initial task in phase two is to determine which API offerings can help you maximize that revenue. In other words, defining your API model.

To specify your API model, ask yourself several questions:

- Which APIs do you need to open up to third-parties (partners, suppliers, vendors, franchises, etc.)?
- Which ones will you open up to the wider public?
- What API packages and limit policies make the most sense?
- Which payment methods create the least amount of friction with the developer community?

Another important consideration during this phase is the total cost of ownership (TCO) of the API management solution you intend to deploy (which may or may not be the same as the platform used for the pilot project), and how this will impact the profitability of your API project. Your calculation should also include the cost of the payment system. Note: Built-in payment systems are convenient but costly, so be sure to factor this into the profitability equation.

Once defined, your API model must be articulated as a requirements document. Be sure to loop your IT team or vendor for this stage of the project to ensure a seamless execution of your new product implementation on the chosen API management platform.

**Phase 3: Configure the API Management Platform**

Now that you’ve pruned redundant APIs from your inventory, optimized your APIs, and identified the TCO of your API management solution, it’s time to set up the APIs on the platform. Think of this stage as your product packaging phase. You need to “gift-wrap” your APIs in a way that will excite and entice the development community. Some of your to-dos at this stage include:

- Define the logic transformation for any legacy APIs. You don’t need to re-write your original back-end SOAP services, but be sure to set-up a REST façade on the API management platform for those legacy web services.
- Define authentication between proxies and the back-end system. For example, set-up mutual certificates or other forms of authentication.
- Define your API packages and their limit policy (e.g. API call quotas and rates), prices, subscription periods, etc.

At this point your initial API configuration is complete. All your APIs, technical documentation, pricing, etc. are available on your developer portal (e.g. http://API.YourCompany.com/).

*A mashup occurs when complementary APIs are combined to create a new application that power boosts the combination by drawing on content and functionality from multiple sources. For example, Google and Bing Maps are heavily used in mashups and often combined with other data sources to power consumer apps such as weather, real estate


Phase 4: Integrate

Your API project is almost ready for external developers. But first you need to ensure your revenue model is in place so that you can begin charging for your data, content, or logic. Some API management platforms don’t include built-in payment gateways—so that’s a consideration. Or perhaps you’ve chosen not to use your platform provider’s payment system. Either way, you’ll need to integrate your preferred system into your API platform.

In fact, phase four involves several integrations and this can get tricky. Typical integrations include user management systems, subscription management systems, as well as the payment gateway. If you lack the expertise in-house, consider outsourcing to a vendor with API integration experience.

Finally, it’s time to support and manage your APIs. Ongoing tasks include small tweaks to your APIs, usage and performance trend analysis, and seeking new opportunities to expand your user base.

Reaching phase four is a significant achievement. Few organizations exhibit such strong commitment and discipline to API management. By establishing deep integration between their API platform and user/subscription systems, these organizations have recognized and realized the potential for solution cross-sell among their user pool and the additional revenue opportunities this brings.

In Summary

More than simply a piece of programming, an API is a powerful business tool. Which is why it’s important to manage the lifecycle of your API as you would any other software product. Hence the significance of having goals, understanding your audience(s) needs, testing, fine-tuning, calculating profitability, and even adding a marketing plan to the mix—each of which combine to form a roadmap for your API.

In the final part of our series on API management we’ll discuss the key ingredients of a profitable API. Stay tuned.

To learn how EastBanc Technologies’ API experts can help your organization thrive, contact:

Wolf Ruzicka
Chairman
wruzicka@eastbanctech.com

Natalia Tsymbalenko
Solution Consultant
ntsymbalenko@eastbanctech.com