



## DEFINING THE ORGANIZATIONAL API MATURITY MODEL

### Abstract

In today's highly digitalized world, product and services companies are under pressure to deliver highly customized experiences to customers at speed. Traditional coding approaches are slow and do not scale easily. Building APIs enables businesses to improve workforce collaboration, automate workflows, and connect applications to talk to each other to deliver customer experiences quickly. To ensure a best-of-breed API foundation, organizations must examine their current API capability and establish an API maturity road map.

This white paper discusses how to conceptualize an API Maturity Model and the parameters to consider while adapting the API Maturity Model for your organization.

“

“A maturity model is a tool that helps people assess the current effectiveness of a person or group and supports figuring out what capabilities they need to acquire next in order to improve their performance.”

— Martin Fowler

”

## Introduction

We live in an API economy, building a digital society where the physical world and the virtual world merge and everything is connected. *Application programming interfaces (APIs)* make integration easier. APIs help connect

people, places, systems, data, and algorithms, enable new digital products/ services and business models, and create new business channels.

When we speak about API maturity, viewpoints differ. As developers, when we talk about it, we refer to the [Richardson](#)

[Maturity Model](#), where classification is based on architectural maturity towards the REST API design. However, when enterprise architects and CXOs look at API maturity, it is viewed at the organizational level and applies to the golden triangle by balancing the three elements — people, process, and technology (PPT framework).



“

“Working with a maturity model begins with assessment, determining which level the subject is currently performing in. Once you’ve carried out an assessment to determine your level, then you use the level above your own to prioritize what capabilities you need to learn next.”

— Martin Fowler

”

With this understanding, let me explain a few essential measurements to build the pillars of the API Maturity Model. These can serve as guidelines for your API adoption program. You need to determine where you stand today and the next-level learning capabilities that you may need to achieve.

## Define Maturity Levels

*"Maturity models are structured as a series of levels of effectiveness."*

– Martin Fowler.

Let me explain the determinants to establish the outline of maturity levels.

**Level definition:** A clear name plus a definition is mandatory for all interested stakeholders in your organization to understand a milestone. Refer to the Capability Maturity Model (CMM) to define all levels that best suit your organization.

**Level weightage:** Each level may not represent the same weightage. The organization needs to take a call on what percentage needs to be achieved at each

level. The leaders and architects in the organization must work closely to define these levels and weightages.

As an example, I have proposed five level definitions with weightage:

- **Chaotic (15%):** Use cases are mostly based on attempts by driven individuals. There is no evidence of an organized initiative. There is no system in place to capture data to measure, articulate, and improve business value.
- **Organized (25%):** There is some amount of process awareness and sufficient documentation. Basic system-centric parameters such as performance, traffic, and error logging data are being collected. There is awareness about the need to demonstrate measurable business value to stakeholders.

- **Strategic (30%):** Businesses accept APIs as digital products that strengthen the organization's foothold in the digital ecosystem. Business metrics are well defined and measured to replicate the performance in business terms.
- **Evolve (20%):** API performance analytics capabilities enable an organization to capture, analyze, and manage data at their business unit and enterprise levels to meet their intended business objectives.
- **Economy (10%):** Enterprise collaboration becomes the key focus. The organizational focus shifts towards continually improving process performance through incremental and innovative technological changes that offer the best value for the money.

## Define Maturity Dimensions

*"A maturity model may have only a single dimension or may have multiple dimensions."*

– Martin Fowler.

Dimensions are the pillars that your organization needs to focus on and track closely for the next several years. The

definition of these dimensions should be self-explanatory and easily understood by everyone in the organization. Adding dimensions makes the model more nuanced, but also more complex. In the early stages, you will need to work closely with your leaders and enterprise architects to determine which dimensions are the best fit for your organization.

I have chosen five dimensions for my explanation:

- Development process
- Architecture
- Developer community
- Optimization
- Strategy and governance

Figure 1 demonstrates how these dimensions stack up against the identified maturity levels

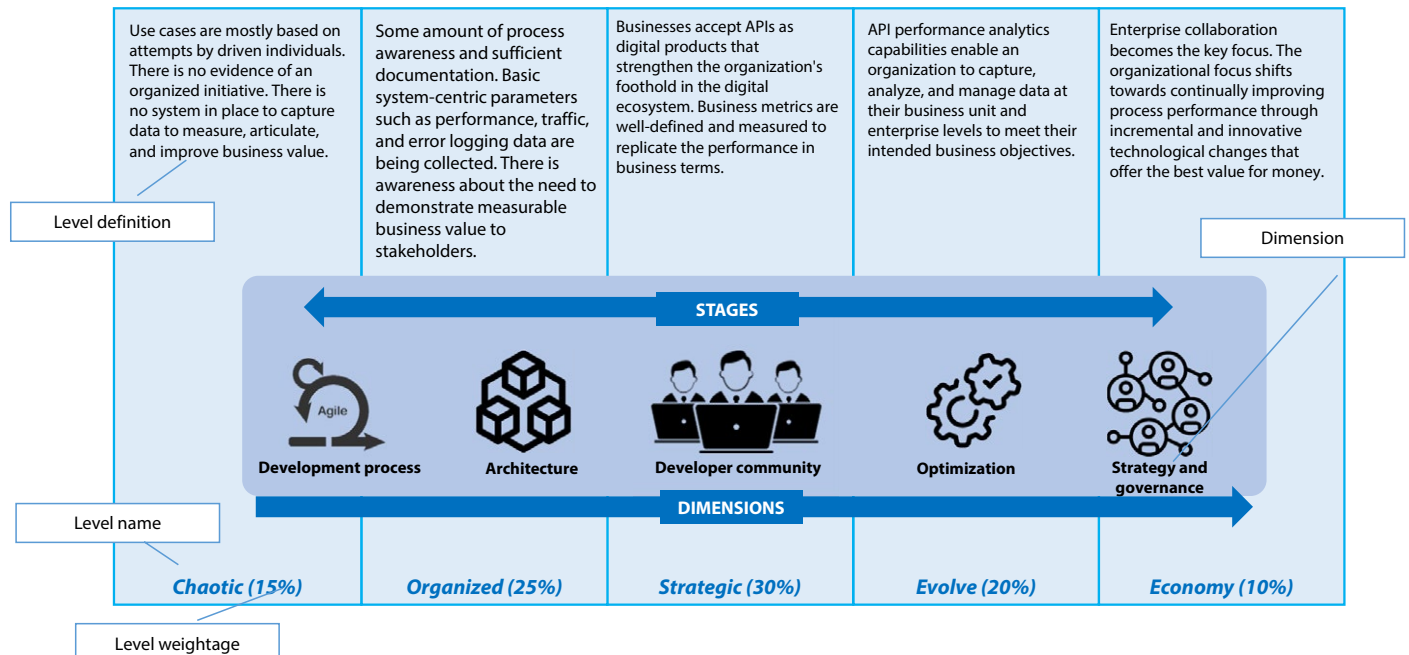


Figure 1 - API Maturity Model: five dimensions vs. five levels

## Define the Maturity Matrix

*“The true outcome of a maturity model assessment isn’t what level you are but the list of things you need to work on to improve.”*  
 – Martin Fowler.

Therefore, identifying parameters required at each level for each dimension is essential. The identification needs to be a collaborative discussion between your

enterprise architects, domain architects, and SMEs.

### Factors to keep in mind while identifying parameters

- The choice of parameters at each stage is driven completely by the organization’s roadmap.
- Each parameter must be associated with the golden triangle elements — people, process, and technology, based on

the PPT framework.

- Prepare a set of questions against each identified parameter to interview the domain-level project architects and leaders to determine where they exist in the matrix.
- Federate the process across domains and sub-domains. You may want to classify the maturity level applicable for each domain and sub-domain layer to obtain a complete picture.

## Define a Target Goal

*“A maturity model can contain generalized estimates of progress.”*  
 – Martin Fowler.

An organization may not be able to achieve its defined top level of any maturity dimension on day one. Therefore, identifying the right initial target goal, and planning and budgeting to accomplish that is important. Certain investment decisions may be involved. A minimum viable product

(MVP) in one domain or sub-domain might give you a clear picture before allocating a big budget and timelines for the entire organization.

Figure 2 is a sample API maturity matrix model prepared using the factors discussed earlier.

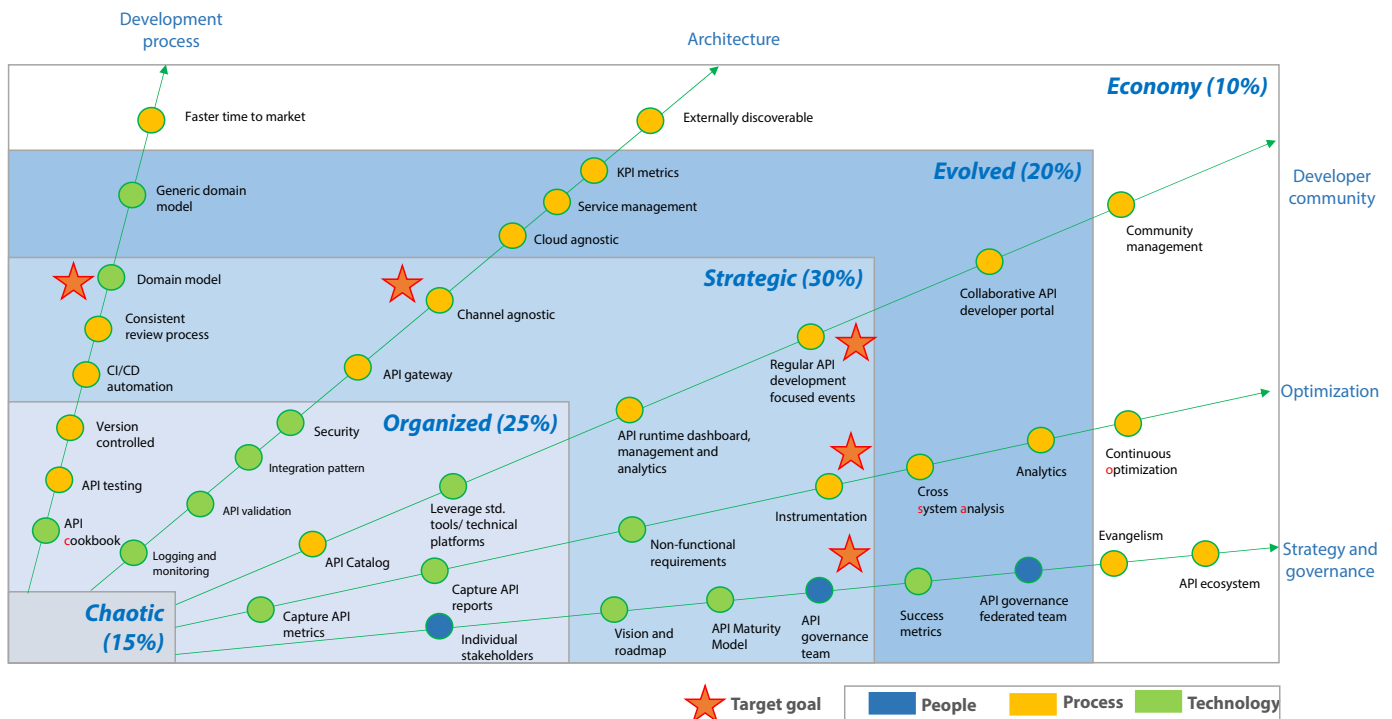


Figure 2 - API Maturity Model with matrix

## Collect Data

Dimensions	Current Maturity	Target Maturity	Levels	Parameters	Question	Answer	Question Weight	"Answer Score (Max 100)"
Development process	15.50%	70.00%	Organized	API cookbook	Do you have a developer's reference manual or guidelines to develop an API/microservice fast?	No	16.67%	0
			Organized	API testing	Do you have any API testing processes and guidelines?	Yes	16.67%	25
			Organized	Version controlled	Do you follow any API version guidelines that include API major and minor versions?	Yes	16.67%	30
			Strategic	CI/CD automation	Do you have a fully automated CI/CD pipeline integrated to deploy APIs?	No	16.67%	0
			Strategic	Consistent review process	Do you have API design review meetings that follow guidelines and processes against the pre-defined API ontology, specification, NFRs, etc.?	No	16.67%	0
			Strategic	Domain Model	Do you have a modeling tool, modeling standards, and exercises to design an API?	No	16.67%	0
Developer community	27.85%	70.00%	Organized	API catalog	Do you publish your APIs in the organization-level API catalog, available across lines of businesses?	Yes	25.00%	20
			Organized	Leverage std. tools/technical platforms	Do you standardize your API tools and platforms such as Mulesoft, Spring Boot, and the use of containers like Docker, Kubernetes, etc.?	Yes	25.00%	50
			Strategic	API runtime dashboard, management and analytics	Does your API portal include API runtime dashboard, management, and analytics?	Yes	35.00%	35
			Strategic	Regular API development focused events	Do you conduct regular events such as a brown-bag session or API day?	No	25.00%	0
Optimization	34.00%	70.00%	Organized	Capture API metrics	Do you capture API performance (such as availability, CPU/memory usage, transaction per second/tps, response time/latency, errors per minute, usage growth, unique and top customers etc), security and compliance?	Yes	25.00%	55
			Organized	Capture API reports	Do you generate API optimization reports based on pre-defined API optimization parameters such as caching, content compression, serializing JSON, etc	Yes	25.00%	30
			Strategic	NFR	Do you monitor and control API performance? (such as throttling bandwidth, assigning CPU for particular transactions, etc)	Yes	25.00%	75
			Strategic	Instrumentation	Do you monitor and control API optimization? (such as enforcing the use of cache, content compression, serializing JSON, etc)	No	25.00%	0
Strategy and governance	13.75%	70.00%	Organized	Individual stakeholder	Have you identified domain-level individual API stakeholders?	Yes	25.00%	10
			Strategic	Vision and roadmap	Have you defined API vision and roadmap?	No	25.00%	0
				API Maturity Model	Have you defined your API Maturity Model based on identified metrics?	Yes	25.00%	15
				API governance team	Do you have any identified team responsible for defining and executing the API strategy, governance, and best practices?	No	25.00%	0

Table 1 - A sample question set for the development process dimension against targeted levels.

Now plot all collected data from different projects on each parameter and generate a matrix. The initial matrix may look not so great, and periodic interviews and data collection must follow to ensure improvement and better representation in the future. Figure 3 shows a spiral matrix representation of the as-is API maturity for a specific domain or sub-domain plotted using the sample interview data collected.

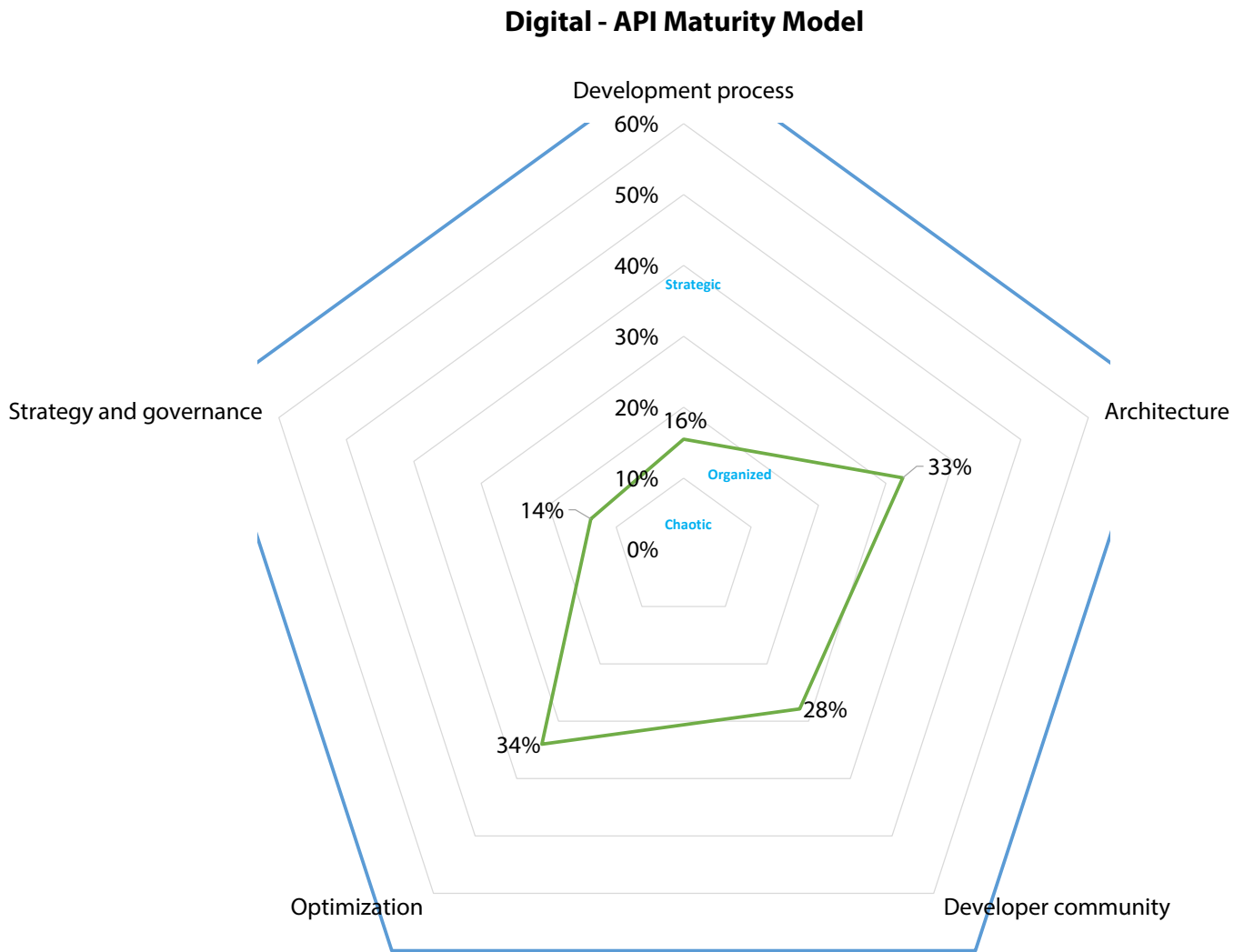


Figure 3 - API Maturity Model — current stage

A photograph of two women in a warehouse setting. The woman on the left has her hair in braids and is wearing a dark polo shirt. The woman on the right has curly hair and is wearing a light blue button-down shirt. They are both looking at a tablet held by the woman on the right. In the background, there are blue metal shelving units and a large water bottle on a table.

## Conclusion

Choose an API maturity model that works for your business. While input is gathered from architects and other subject experts, it is also important to align with the overall business goals of the organization. When you use any of the maturity models to understand a situation and draw inferences, you need to be careful and ensure that the model is the right fit for your specific requirements. The stated model could be your base version and you need to go from there. Finally, the model and the metrics of progress to the required level must be arrived at collaboratively with leaders and other stakeholders in the organization.

## About the author



**Debasish Dalui**, *Principal - Enterprise Applications, Infosys*

Deb is a passionate technologist with a zest for solving large and complex business-critical challenges leveraging his experience and expertise. At Infosys, he drives the digital transformation initiative by taking ownership of strategic technical engagements across multiple domains and geographies, delivering market-moving engagements, meeting customer success targets, and building trust for lasting impact.

## About Infosys API and Microservices

Infosys' API Economy and Microservices, part of **Infosys Cobalt**, is focusing on business transformation through ecosystem connectivity, unlocking the value of data, and simplification and agility in digital initiatives. To learn more please visit – [www.infosys.com/api](http://www.infosys.com/api)

**Infosys Cobalt** is a set of services, solutions and platforms for enterprises to accelerate their cloud journey. It offers over 14,000 cloud assets, over 200 industry cloud solution blueprints and a thriving community of cloud business and technology practitioners to drive increased business value. With Infosys Cobalt, regulatory and security compliance, along with technical and financial governance comes baked into every solution delivered.

For more information, contact [askus@infosys.com](mailto:askus@infosys.com)

**Infosys**<sup>®</sup>  
Navigate your next

© 2022 Infosys Limited, Bengaluru, India. All Rights Reserved. Infosys believes the information in this document is accurate as of its publication date; such information is subject to change without notice. Infosys acknowledges the proprietary rights of other companies to the trademarks, product names and such other intellectual property rights mentioned in this document. Except as expressly permitted, neither this documentation nor any part of it may be reproduced, stored in a retrieval system, or transmitted in any form or by any means, electronic, mechanical, printing, photocopying, recording or otherwise, without the prior permission of Infosys Limited and/ or any named intellectual property rights holders under this document.

[Infosys.com](http://Infosys.com) | NYSE: INFY

Stay Connected   