APIs@Scale

Setting the Foundations for Success

June 2021

By Régis Martin, Jilco Schuurmans,
Norbert Faure and Jakub Fila

BCG PLATINION

To access the range of benefits offered by APIs, many organizations are evolving towards a new model where digital capabilities are based on shared and reusable components.

By Régis Martin, Jilco Schuurmans, Norbert Faure and Jakub Fila

June 2021

BCG PLATINION

As a part of Boston Consulting Group (BCG), BCG Platinion provides consulting & engineering services in human centric design, IT architecture, development, and implementation of advanced technology solutions that fuel critical transformation and creation of new-generation business models. Today, our presence spans the globe with offices in Europe, North and South America, South Africa, and Asia Pacific.

Contents

05 Introduction

06 The Challenges

08 Where to Begin?

10 Setting the Foundations

12 Activating the Model

15 API Resources

17 API Discovery

18 Governance

20 Design with Business Units in Mind

22 APIs as Products

24 A Bold and Comprehensive Approach

25 Appendix

- TomTom Case Study
- The Importance of a Developer Portal
- Pricing its APIs
- The Technical Architecture of TomTom
- Conclusion

APIs@Scale

By Régis Martin, Jilco Schuurmans, Norbert Faure and Jakub Fila

After years of development and battle-testing, APIs are rapidly becoming a catalyst for collaboration. The process of building digital ecosystems has encouraged organisations to work with suppliers, customers, and other like-minded companies. The result of this has been the emergence of ecosystems based on shared data and digital resources.

Today, many of the world's biggest and most disruptive companies have built their businesses through scalable developer services. For example, more than 60% of the companies included in the CNBC Disruptor 50 selection for 2020 provide APIs to third parties, enabling them to influence everything from payments to mapping, from instant messaging to email. BCG research also shows that 43 of the 50 most innovative companies of 2020 were operating with discoverable APIs, amounting to a towering 86%.

According to programmableweb.com, in the region of 24,000 APIs are in operation across 500 categories, including open banking, telephony, and electronic signatures to name a few. As businesses of all sizes leverage the API economy, they are discovering new, impactful business opportunities.

In a recent BCG analysis of digital ecosystems across all industries, one-quarter of the companies surveyed said that API ecosystems will account for more than 60% of sales within three years. In addition to this, close to 30% of those surveyed expect digital ecosystems to impact annual profit by at least 25% in the same timeframe. We are also seeing organizations leveraging APIs to improve their internal communications, particularly in the case of enterprises with multiple business units and subsidiaries.

To access the range of benefits offered by APIs, many organizations are evolving towards a new model where digital capabilities are based on shared and reusable components. This trend goes hand in hand with the move from on-premise monolithic legacy IT systems, to microservices in the cloud.

As part of this evolution, an increasing number of developers are becoming skilled in understanding the nuances of coding and integrating APIs into applications. On account of this influx, the global API management infrastructure market is expected to reach a valuation of $1.78 billion by 2026, growing at a CAGR of 20.3% according to Allied Market Research.

From a business perspective, the increasing rate of API adoption will enable organisations to respond to growing market demands and successfully benefit from new opportunities. To achieve this in an effective way, it is vital that organisations are aware of the critical challenges to adopting APIs, and the right foundations that must be put in place to overcome them.

The Challenges

APIs are a key enabler of data ecosystems, whereby combining one's data with others' significantly increases the number and value of potential available use cases. The stakes are high when it comes to data ecosystems, and there are significant barriers to data sharing that only a few successful ecosystems have overcome. For example, HERE Technologies and Airbus Skywise managed to do this by leveraging APIs.

Having analysed the barriers, BCG recommends the adoption of governance that is based on the rules of Smart Simplicity, promoting co-operation within complex data ecosystems. The rules include understanding what people really do, reinforcing the integrators, increasing the total quantity of power and reciprocity, while rewarding those who cooperate.

APIs must be adopted with those rules in mind, with a particular focus on understanding what people really do. Alongside this there must be a clear understanding of what the API provides, and what the developer experience is like. Reinforcing the integrators is also essential, with APIs acting as a common and battle-tested technical means to exchange data.

Based on our work with clients, there are six common dimensions challenges must focus on when building out their APIs:

1 Value creation – Modeling the value created through APIs can be complex, and the root of this complexity is in the diversity of the business models they support. These range from top line impacts to internal IT efficiency. Understanding the business model of APIs enables the creation of a strong business case, factoring in how they will impact customer relationships, and whether a pricing model is required. Successful early adopters of APIs have gone through the journey of learning how to model and assess the benefits of APIs, with the topic being discussed among the c-suite and at board level. An example of such a journey was experienced by the insurance company, Euler Hermes, which monetized its APIs and benefitted from the innovation they generated.

2 Developer Engagement – On the ground, software developers, and increasingly citizen developers leveraging Low Code/No Code platforms, are the ones who test and consume the APIs by integrating them into their applications. Those developers can be internal or belong to third party organisations. It is critical for the success of APIs to engage with these developers in an appropriate manner, in contrast to a classical marketing and sales target. Successful API-based companies have learnt how to engage with developers, leverage appropriate communication channels, and invest in developer portals that offer documentation, tooling and support.

3 Operating model maturity – The transition to becoming an API-driven business presents a challenge for many industry incumbents. This relatively new way of doing business leaves many internal teams unclear on how to leverage the value derived from APIs, and how to efficiently begin delivering them. Remedying this requires organisations to instill a culture that is comfortable with the management of digital products, data and software architecture. Successful execution of widespread change within a business requires a supporting framework, and in this case one is required that explains how APIs are to be organized and managed. This blueprint must then be aligned with agile ways of working.

4 Data Isolation – In the world of data, speed is critical, and accuracy is everything. As organisations collect countless data points from different locations, systems must be capable of processing the information and generating real-time insights. In many cases this ideal situation is made impossible by legacy systems, due to their inability to make locked-in data easily accessible with the scale and elasticity required by APIs.

A recent BCG survey of some 600 companies found that 60% of respondents rated their data governance capabilities at various levels of underdevelopment. APIs present an opportunity to support data governance and future proof data quality without compromising on speed.

5 Security – Security presents a major challenge when developing APIs, with cloud-native apps often built with thousands of microservices. This makes them difficult to protect at scale, and Gartner has identified this as a key concern with a prediction that by 2022 API abuses will be the most frequent attack vector.

6 Scaling Investments – Agility and flexibility are crucial to digital initiatives. Because of this, APIs must be able to change and adapt over time to meet the changing demands of both partners and users. They must also be able to scale to accommodate changes to the size of the business or unpredicted usages.

To leverage the full potential of APIs, businesses need to go beyond viewing them as just another tool inside an IT architect's toolbox. Instead, organisations should embrace APIs from the outset as enablers of evolving business objectives and digital transformation.

Where to Begin?

If an organisation puts the cart before the horse by developing APIs as a quick technical solution, momentum will soon be lost. It is essential that companies commit to continuous learning and study the market through a business lens. The development of an API strategy is secondary and supportive of a business, its vision, mission and customer experience.

Case Study - EasyJet

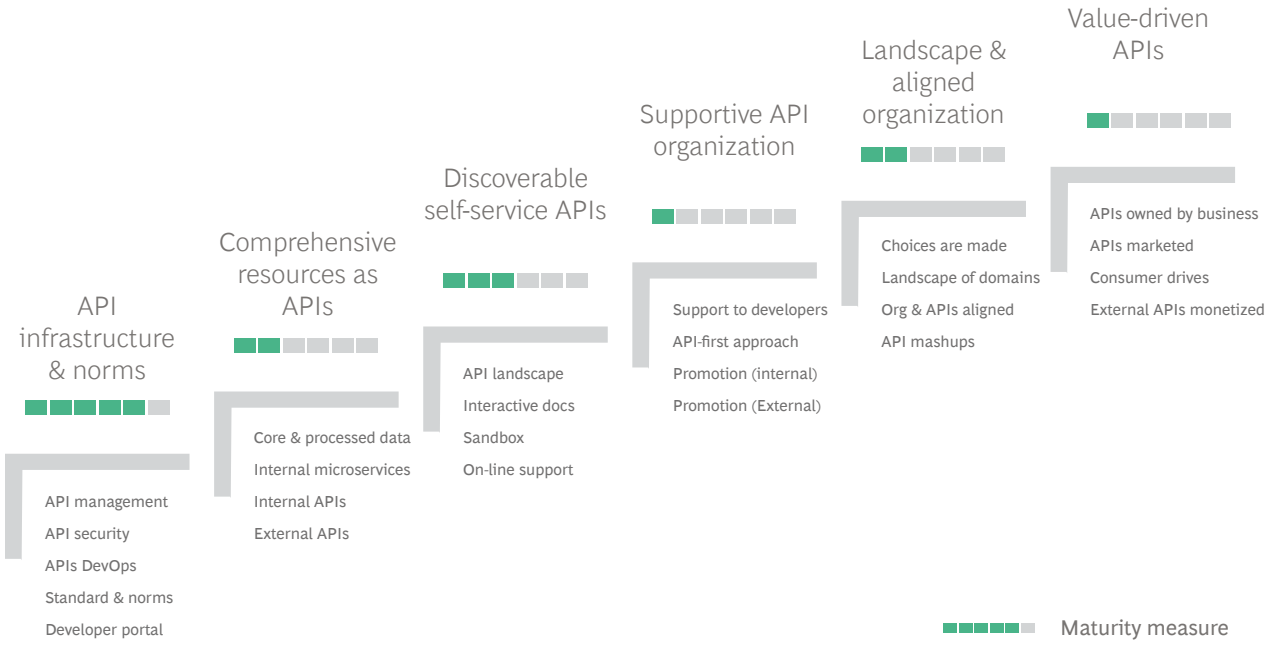
Its strategy was to diversify the way it targets corporate travel in reaction to companies across the world, prohibiting business-class tickets for short-haul flights. The solution was the development of an API-led ticketing function which allows travel agencies to book easyJet like any other carrier, whilst the airline connects to technology platforms such as Amadeus. This was the first time a major European low-fares airline made its inventory available to Global Distribution Systems (GDS) providers, and as a consequence business travel now represents 20% of its total bookings.

By setting out with both a business strategy and a plan for the customer journey and experience, organizations can determine which API-specific capabilities are needed in order to deliver measurable outcomes.

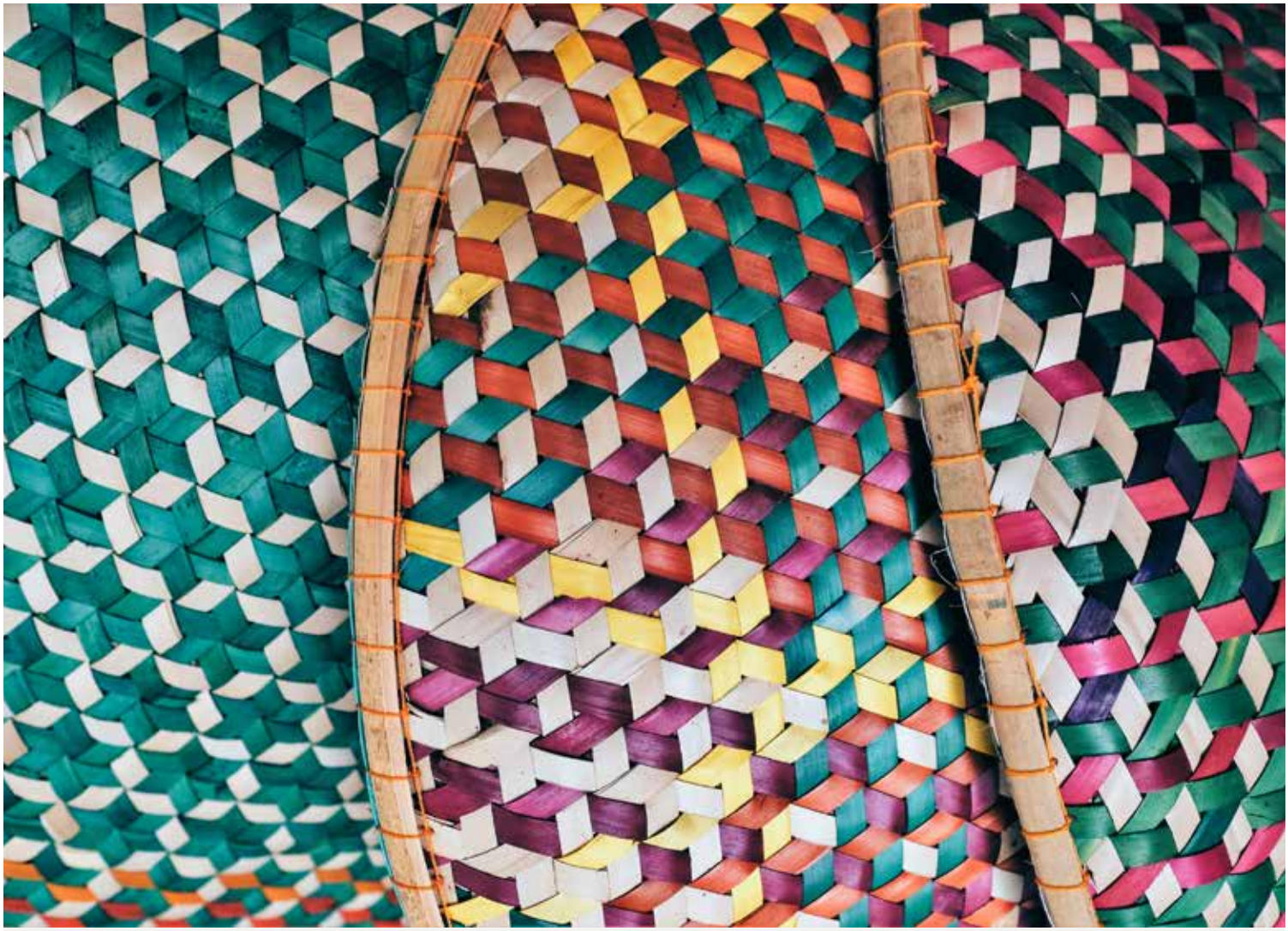
Starting out with a success story can also be a powerful strategy. This involves taking a robust, thorough approach to the creation of a single API, which will serve as a template for further implementations at scale across your organisation.

At BCG Platinion, our API capability framework sets the standard for succeeding with APIs at scale while managing them as products. We take different capabilities like the technology platform, self-service, organisation and value creation into account, with a holistic view for preventing the common challenges associated with scaling an API ecosystem.

Exhibit - API-specific capabilities support APIs as product



Source: BCG Platinion analysis



Setting the Foundations

There are a range of open platforms available that can host, build and deploy API services, leveraging middleware and governance tools to help orchestrate them. Whether you need an API to provide internal data or are planning to offer an external API to partners, there will be a platform supporting your approach.

API management solutions

In recent years, numerous vendors have developed solutions to expose, protect and manage the full lifecycle of APIs, both in the public cloud and for hybrid setups.

One example of a collaborative API design platform is Stoplight. This company offers a service that integrates into existing workflows with the intention of boosting the productivity of all stakeholders working with APIs.

Classical API management solutions focus mostly on the publication and operation of APIs, providing capabilities such as gateways, developer portals, support governance of the lifecycle of the APIs, and administration of the API management itself.

There are three key types of vendors to consider in regard to classical API management solutions:

1. Vendors with a background in enterprise integration tooling, which are typically ESBs, like Mulesoft (part of Salesforce), IBM, Software AG or Axway.

2. Hyperscalers, which often have acquired emerging specialised vendors, including examples like Google/Apigee, Azure/Apiphany or AWS.

3. Specialised API management vendors themselves, with Kong or Tyk being prime examples.

The first step towards selecting the right API management architecture is to frame your requirements, here is a summary:

- **Understand** the full scope of functional capabilities of API management solutions and related stakeholders. Use a reference architecture like the one exhibited below to do this.

- **Describe** the most likely use cases, including the typology of APIs to be exposed. Examples of these include internal, external or those between microservices.

- **Define** the user journey for the various teams involved with the APIs, including API owners, API developers, consumers and associated teams like security and finance. In large organisations, particularly when decentralized, the solution should enable diverse roles and permissions for working with the API management solution.

- **Assess** the physical infrastructure options adapted to your hosting strategy, including options like cloud, on-premise and hybrid.

- **Set** clear performance and scalability expectations.

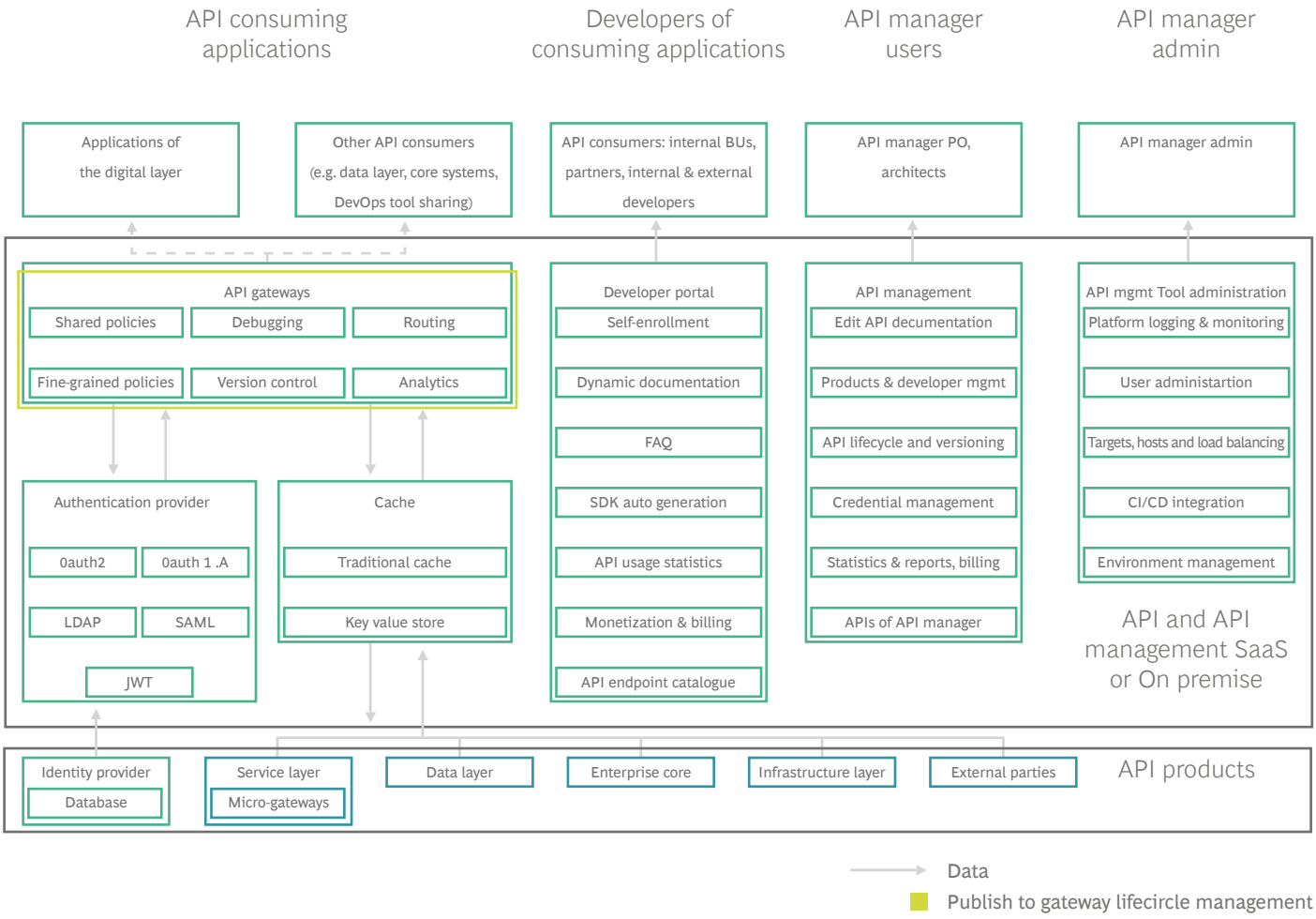
When identifying the right API management architecture, an important part of the process involves testing the API management solutions in order to better understand their features. Once these steps have been carried out, next it is time to review the tools, standards and security measures you have chosen, then begin establishing a roadmap.

Security model

From a risk perspective, the best approach is to consider what the relevant, inherent vulnerabilities are, taking your organisation and industry into account. Since APIs serve as conduits that reveal applications for third-party integration, they are susceptible to cyberattacks.

The Open Web Application Security Project® (OWASP), a nonprofit foundation that works to improve the security of software, lists and describes the top 10 vulnerabilities that must be taken care of, including exposed endpoints, incorrect authentication and insufficient monitoring.

Exhibit - Simplified API management reference architecture



Activating the Model

The digital ecosystems that are emerging due to new business models and market forces require solid technical foundations to be successful in the long run. It is also important that developer-friendly data platforms are established at the outset, including microservices, analytics, containers and application architectures. This approach supports an effective roll-out of APIs at scale, as well as future architecture development.

Standards should not be set with a sole focus on the design architecture. Developers need clear guidelines for robust security, successful data integration and optimized connection, as well as an understanding of the commercial terms. A public example of API standards has been established by the UK government's digital service.

It is also important to set the standards at the outset because it is the first step to engaging with developer communities. When many companies begin this journey, they establish a developer portal, through which technical documentation can be hosted and accessed.

In the case of TomTom, the developer portal set up by the company serves as an ecosystem and distribution channel, integrating sales and support into a single platform.

Case Study - TomTom

When TomTom made the strategic decision to make its developer portal a sales and support channel, it recruited and assigned dedicated teams (rather than re-purposing the TomTom consumer marketing team or TomTom product people) and asked them to adopt an agile way of working that focused exclusively on the developer community.

The team were developer-literate and understood the unique marketing tactics and language that would engage and support developers. The design of subsequent workflows, self-service functionality and dashboards were all informed by developer requirements, which meant a community was already established before the portal went live.

APIs play a key role in acting as contracts between the producers of digital resources and their consumers, both internal and external.

API Resources

Once organisations have set up an effective API platform, product owners can communicate the objectives of new APIs to the relevant architects and engineers. This will enable them to build the right application and integrate it in a way that supports both internal and external data sources.

Many advanced organisations are taking full advantage of APIs, adopting an API-first approach to evolving their technology landscapes:

- **At the infrastructure level, the best examples of the power of APIs are in public cloud applications, as well as in Platform-as-a-service (PaaS) and Container-as-a-Service (CaaS) offerings. As an example, the AWS cloud offering counted 175 services exposed as APIs in 2020.**

- **In the case of legacy systems such as ERPs, APIs can be used as façades to facilitate the connection of digital applications without having to launch huge ERP transformation programs.**

- **In modern data architectures, processed data is often exposed through APIs to facilitate the build-up of analytics and digital use cases.**

- **In the upper layers of the architecture, business modules and digital components are exposed through APIs, enabling the modularity of the architecture. This is a key quality required in the context of digital.**

Exhibit - APIs bring clarity and simplicity at all levels of the enterprise architecture

Digital applications
User-centric environment coupled with modular components, leveraging APIs for data access

Data environments
Collection, storage, analysis and synchronization of information across applications & processes to leverage data

Core applications
Enterprise backbone production systems connected in real-time

Infrastructure
lCloud services, hardware, network and DevOps toolchain to run application landscape

API Discovery

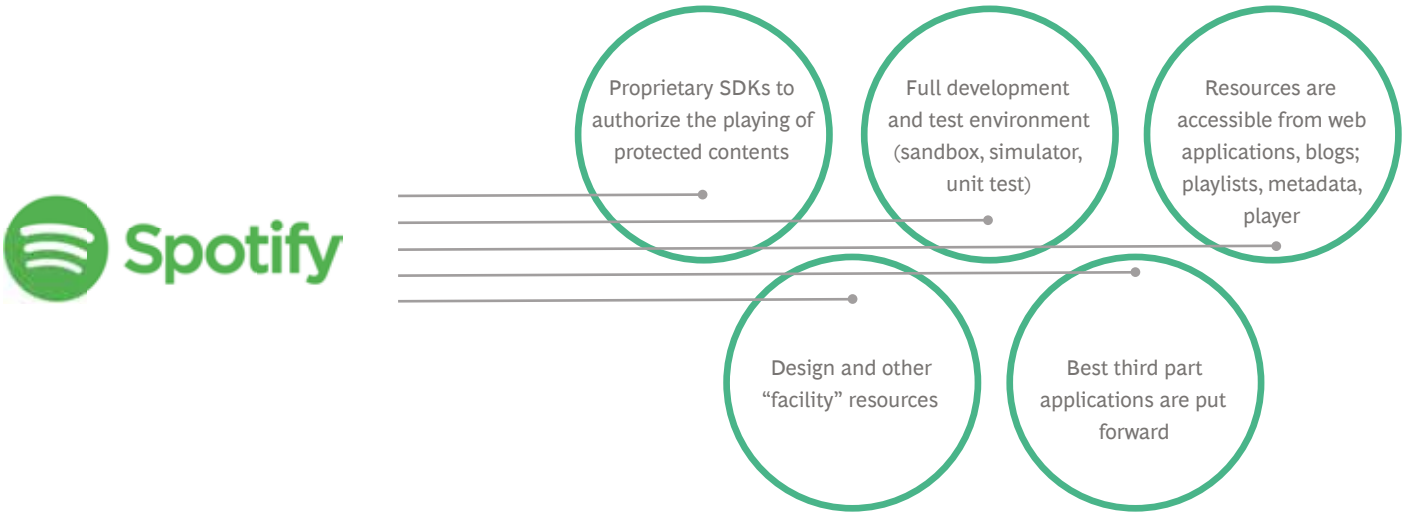
The next key phase is to enable API discovery, where developers get to understand the capabilities of the API, and how the resources it provides can match their needs. If required, this step can involve combining an API with others to derive the desired capabilities.

Today, the process of API discovery is usually controlled using a central repository of the organisation, where documentation for resources is made available to developers. In the case of APIs that are only made available to selected partners, just the authorized developers will be able to view the documentation.

Once discovered, the developer portals also need to have a well thought through developer experience (DX). This is to ensure the core capabilities and code are easily tested and sourced, and that features can be seamlessly integrated.

The Spotify developer portal is an excellent example of this, having recently released a new open-source tool called Backstage. The tool collects all the required resources into one browser-based user interface, including design patterns, SDKs for popular programming languages, and deployment instructions.

Exhibit - Portal supporting developers in the integration of APIs in their platforms

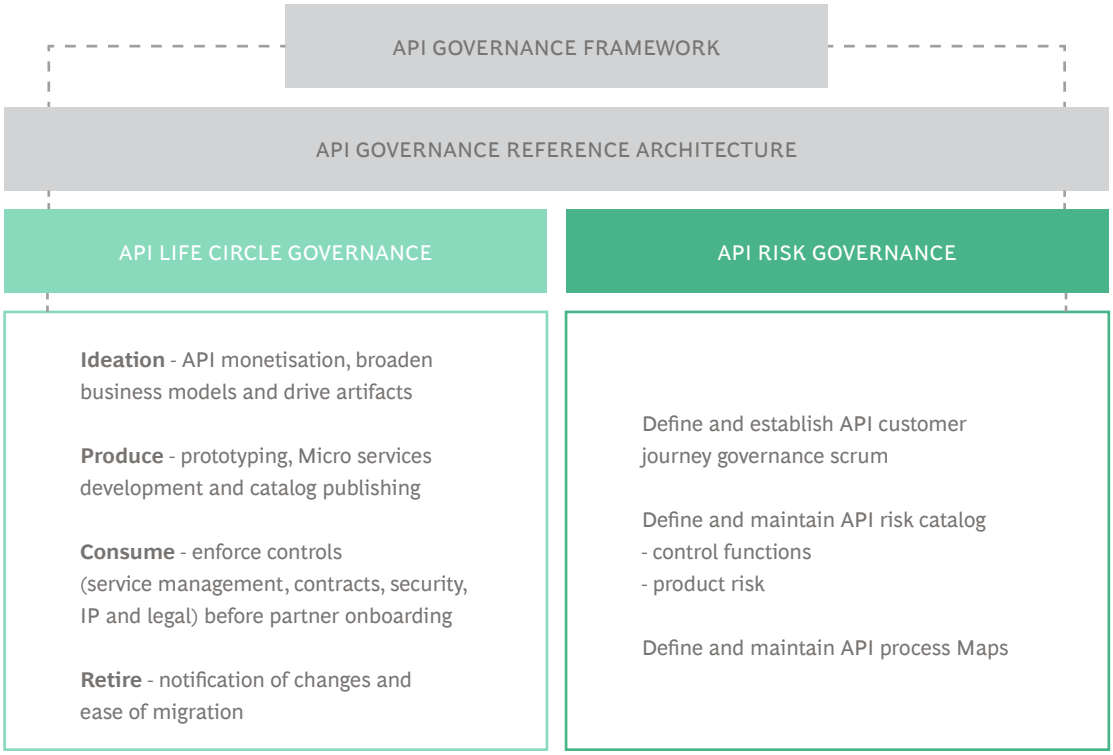


Governance

With a robust API governance framework in place, developers will be equipped with the necessary rules to ensure that they adhere to industry standards and security policies. Another benefit of effective governance is that it enables APIs to be enriched with metadata, making it easy for both internal and external stakeholders to use them.

A strong API governance framework balances two dimensions; API lifecycle governance and API risk governance:

Exhibit - API Governance framework



Many organisations choose to create a Center for Enablement (C4E) to support the design and development of APIs, as well as their governance. This approach catalogues APIs in a way that allows developers to reuse the code and avoid duplication, this information is then made accessible via an intuitive dashboard. Above all, a C4E is designed to accelerate the process, rather than becoming a central bottleneck.

A C4E should be clearly positioned as an accelerator for the development of APIs, supporting operations, development teams and agile ways of working.

Not only is it crucial that API governance is effective, but it is also essential that it is connected to your organization's overarching data governance framework. Linking the two forms of governance is vital to effectively implementing APIs at scale. By doing this, you ensure that full visibility is achieved, and quality outputs are generated through proper data management.

Case Study - Experian

The API Center for Enablement for Experian Information Technology Services is responsible for overall API Experian strategy, including ensuring applications meet the needs of the business as a whole. The C4E oversees the technical and business fitness of the application portfolio and their associated costs and risks, as well as development of the application governance process. This provides consensus on a prioritised set of measurable business outcomes that applications must support.

Design with Business in Mind

When APIs are delivered as products the process requires a close interaction between business and tech stakeholders, working together to develop a common semantic for APIs.

By leveraging Domain-Driven Design as an approach to architecture and software development, you can mitigate the risk of mistakes caused by the inherent ambiguity of data in large organisations. In the automotive industry for example, a "vehicle" can mean quite different things in a R&D department to what it means in aftersales.

In its most rudimentary definition, Domain-Driven Design is a method for developing a team's understanding of the wider business objectives and establishing a common language with their business counterparts.

Adopting a Domain-Driven Design approach makes it possible to:

- **Identify the most important subdomains on which the design effort must be focused.**
- **Base the modeling of the business domain on a language shared between different business units and development teams.**
- **Reuse the concepts of a common language in the code of services and applications (Model-Driven Design).**
- **Use techniques of decomposition, well-defined subdomains and communicate through well-defined interfaces.**
- **Model services from current Bounded Contexts in subdomains and define areas of responsibility.**
- **Separate data by context to ensure any given context cannot directly access the database of another subdomain.**

Modern architectures that are based on modular microservices, combined with small agile teams, make it possible to align the structure of software with the structure of the teams building and maintaining it.

This reality was foreseen in 1967, when the computer scientist Melvin Conway stated that "any organization that designs a system will produce a design whose structure is a copy of the organization's communication structure." This statement is now widely known as "Conway's Law."

APIs play a key role in acting as contracts between the producers of digital resources and their consumers, both internal and external. They are also essential to communication at the border of domains and subdomains. They must convey clarity and simplicity in exposing resources and hide the internal complexities of IT systems. To achieve this level of clarity, Domain-Driven Design is a popular approach.

As an acid test, it should be enough to have a high-level knowledge of the business to understand what an API provides, no business expertise or tech background should be needed. On a more technical note, an additional dimension to consider when building an API is the communication architecture, which can follow synchronous or asynchronous patterns.

Exhibit - Designing with business in mind

PRINCIPLE	DESCRIPTION	EXAMPLES	VALUE
Domain driven design	How services, APIs and components are grouped into domains that are relevant to business areas	TomTom organizes its API in 5 domains: maps, directions, places, traffic, tracking The Stripe APIs are re-grouped in 10 categories reflecting business processes or objects	This provides a more business relevant way to develop services. It also attributes clear ownership and defined meaning to objects
Microservice architecture	How systems will be composed into focused, well-defined, purpose-built services	Spotify, a pioneer of microservices, proposes 15 APIs in various domains such as album, artist, episode…	This enhances scalability and adaptability as well as unification
Communication architecture supporting the API	This process involves the identification of the two main communication patterns: synchronous and asynchronous	The synchronous pattern is the most common, with REST over HTTP Slack proposes the asynchronous Events API, making use of WebSockets IQVIA Technologies is a supporter of the AsyncAPI project	When using synchronous communication combined with the request-response style (e.g. with REST APIs), it enables a lightweight, standardized and simple implementation Asynchronous communication offers the advantage of decoupling services as required, adding a degree of flexibility, but additional infrastructure and complexity

APIs as Products

When an organisation explores the prospect of APIs as products, decisions around the API architecture and data infrastructure must be aligned to the business vision. It is also important to align the architecture to the target customer experience while embracing innovation.

Technology innovations such as VR, 5G networks, 3D-printing, blockchain, quantum computing and artificial intelligence support the emergence of innovative business models. Examples include crowdsourcing, decentralized finance, the circular economy, the free data economy and the transformation of products into platforms.

Whatever the technological catalyst, regardless of industry, it is likely that APIs will be the foundation upon which a digital model is built. This allows developers to create apps that can react quickly to rapid disruption.

It is still difficult to predict with absolute certainty how the future will affect any one organisation, this is why it is paramount that scalability and evolvability must be instilled into the development of an API and the overall architecture. With this taken into account, when the future does catch up with a business, or vice versa, the API can process increasing amounts of data and value can be maximized.

Today, the API economy value chain has three main stakeholders:

- **API owner:** The owner opens up an asset or platform under certain terms and conditions and is responsible for sales and marketing.

- **Consuming application developer:** The developer uses the API under these terms and conditions to supply a digital resource to an application for an end-user.

- **End user:** The end user does not directly see the API, but benefits from its use in the app that is provided.

While a free option may help to drive adoption of APIs among developers, or build brand loyalty amongst end-users, this may prove to be a limiting factor further down the line.

If the developer of the consuming app incurs the cost, then there must be a return on investment for them further down the line. A classic example might be a credit-check API that would help a business assess a customer's creditworthiness. In this scenario, there are various sub-models that could come into play, including a pay-as-you-go model whereby the developer pays for what is used.

Another example could be a freemium model that has a basic level that is free but a more feature-heavy option that requires a subscription. Tiered models are similar but more structured around the number of anticipated API calls the business will require.

In contrast, when the developer gets paid, an API owner will provide a monetary incentive for that developer to leverage a web-based option. An example of this would be a third-party insurance agent selling a product and receiving a share of the final revenue.

Whatever the decision, pricing needs to be kept simple for any stakeholder, otherwise developers will favor competition. Developers need confidence that time and effort will be rewarded, which invariably means allowing them to test their code before paying for it. Free usage levels and sandboxes are the most common methods that will allow developers to experiment.

As well as charging directly for APIs or requests, there are also emerging revenue models designed around the acquisition of customers using APIs as inbound funnels. Similar to web sites acquiring new customers through referral traffic and affiliates, APIs can take on a similar role. Companies on the customer acquisition trail can either use their API to make third-party widgets available inside a desktop application or create stand-alone applications that make use of it.

Once customers are on board, APIs can also increase the Customer Lifetime Value (CLTV) by offering an API that is integrated with third-party apps. Enabling third-party apps to obtain your content or products also makes it easy to open up new revenue streams with partners.

A Bold and Comprehensive Approach

A recent BCG survey called "Flipping the odds of Digital Transformation success" evidences the commitment required to succeed in achieving digital transformation at scale:

1. Craft a clear, integrated strategy
2. Commit to leadership from the top down
3. Put the best people in the right places
4. Adopt an agile governance mindset
5. Monitor and measure your progress
6. Rethink and modernize your tech stack

Succeeding with APIs involves the same commitment as is required to succeed in digital transformation. APIs are multifaceted and each of their aspects must be leveraged and supported with a clear and integrated strategy.

APIs have been a prominent theme in the Gartner Hype Cycle and are considered a catalyst for high levels of business value. To bring this to fruition, the appropriate commitment and talent to deliver value through business ecosystems is essential.

Among the key benefits of APIs are their abilities to mask complexity, provide clarity and be consumed easily. They facilitate the adoption of agile and empowered teams, enabling the acceleration of digital product delivery and measurable outcomes. Additionally, APIs contribute to the modernization and scaling up of your tech stack, due to factors like high quality coding and the ability to serve as a robust, industrialized integration pattern.

To sum up the potential value and impact of APIs, it is worth reflecting on a famous 2002 mandate from Amazon founder, Jeff Bezos. Opening with the words, 'All teams will henceforth expose their data and functionality through service interfaces,' the note would prove to be central to Amazon's remarkable success in the modern web space, and in the emergence of Amazon Web Services as a powerful offering.

In 2006, Amazon S3 was launched, providing a storage service that would equip developers with a scalable, reliable data storage option. With the emergence of this API, it started to become clear that the Jeff Bezos mandate would ultimately do more than just influence Amazon's success. It would in fact go on to shape modern thinking around APIs and microservices to a great extent.

Before embarking on or continuing on your journey to leveraging APIs at scale, there is a key consideration to remember. Just as any element of digital transformation requires a bold, comprehensive approach, so do APIs.

Appendix

TomTom Case Study

TomTom has been synonymous with navigation since the days its mapping devices were mounted to windscreens around the world. But today built-in systems and mobile devices have made the necessity for separate units redundant.

And whilst TomTom might not enjoy the same level of brand recall as a result, it is guiding more travelers from A to B, and via C, than any other provider of navigational data in the world.

They are currently the mapping provider for Microsoft, Apple and Huawei, and provide data for the majority of automotive manufacturers including Renault, Peugeot, Citroën, Nissan, Fiat, Chrysler, Volkswagen, Ford, Audi, BMW and Mercedes. They also supply the maps and traffic data for Uber drivers in over 10,000 cities worldwide.

Its successful pivot into a global technology enterprise is thanks to APIs. With TomTom being an original mapmaker, they have been able to encourage developers to innovate new solutions via their APIs. This includes functionality that can calculate the range of electric cars and identify recharging points, geofences that track assets in fleet management, optimized routing for ride hailing applications, traffic data from over 600 million sources and search capabilities that reveal real-time reviews on places of interest.

By allowing its partners and customers to bring the TomTom products inside of their own applications, TomTom has expanded the scale of possible markets exponentially. And that has been the catalyst to help the company future-proof its digital transformation.

The Importance of a Developer Portal

When TomTom set out to build a portal that did more than simply host technical information, it assigned dedicated teams rather than re-purposing the TomTom consumer marketing team or product people - it recruited specialists and asked them to adopt an entrepreneurial and agile way of working that focused exclusively on the developer community, learning their language and the tactics needed to best support them. This resulted in a seamless onboarding experience for new developers, agile workflows, automated self-service and human-centric dashboards.

The importance of the developer portal continues to be critical for TomTom as the company works hard building MVPs with developers as a way of creating two-way learning. And from that it can understand new use-cases and target different verticals. It's all part of a mission to design a developer journey similar to how TomTom might have previously considered a customer journey when it was a manufacturer.

Pricing its APIs

TomTom considers its online pricing models a key competitive differentiator, since it provides maps and traffic flow tiles for free in its mobile map SDKs and unlike many companies building an API ecosystem, for TomTom, the API is the service that it sells. And that's visible in its developer portal where you have longtail pricing.

With free maps, developers can add a location to their application without incurring any additional charge, such as displaying the location of your business on a map.

This is a free service via the TomTom Maps Mobile SDKs, as are the traffic tiles. As a result, developers can also obtain accurate traffic-based data and develop relevant mobile applications without additional cost.

Meanwhile, the TomTom Maps API pricing structure is transactional, and customers only pay for the API requests they consume, without any additional subscription fees.

The Technical Architecture of TomTom

At TomTom there are technical teams that each have their own competencies and expertise in map display, search, routing, geocoding, traffic and backend development.

But there is a centralized team who is responsible for the API Management System and a strong alignment on the architecture of APIs to ensure consistency across an API ecosystem and the management of data.

This team sets the dynamic rules for accessing the mapping and traffic services to ensure that data is scalable and accurate, which is imperative because its database can be consumed at any moment in time.

TomTom is also focusing on building APIs and SDKs as products in their own right, with software development kits that solve very specific problems for the industries that the company serves.

This is the catalyst towards migrating everything to the cloud where TomTom can have enough computing power to be scalable. By transforming the back-end architecture, TomTom can support its strategic pivot from a provider of static and historical data, to an intelligence company that can receive and expose data in collaboration with its partners and in real time. It allows TomTom to focus on new micro-services and functionalities for telematics, mobility and delivery services.

And by having such a strong developer portal, TomTom can work with customers on very specific problems or solutions respective to an industry, for example components for autonomous cars.

Conclusion

Every day, over 125 million drivers rely on location-based apps built with TomTom. But even then, TomTom is not resting on its laurels and through APIs is thinking about how it's continually going to disrupt itself to stay ahead of the curve in terms of the services that it provides to businesses.

About the Authors

Régis Martin
IT Architecture Director
Paris
Martin.Regis@bcgplatinion.com

Norbert Faure
Managing Director
Paris
Faure.Norbertbcgplatinion.com

Jilco Schuurmans
Managing Director
Amsterdam
Schuurmans.Jilco@bcgplatinion.com

Jakub Fila
Chief IT Architect
Warsaw
Fila.Jakub@bcgplatinion.com

For information or permission to reprint, please contact BCG Platinion at **platinionwesamarketing@bcg.com**

To find the latest BCG Platinion content, please visit **bcgplatinion.com**

Follow BCG Platinion on **LinkedIn**

© BCG Platinion 2021. All rights reserved.

bcgplatinion.com