

A Developer's Guide To Forrester's Strategies For API Success

Take A Guided Tour Of Forrester's Research On APIs And API Strategy

by Randy Heffner
September 19, 2019

Why Read This Report

APIs are critical for digital transformation. Beyond their architectural role in app integration, APIs enable new business strategies, rapid business change, broad ecosystem connectivity, and improved customer engagement. To help application development and delivery (AD&D) clients establish and evolve a well-grounded API strategy, this report ties together Forrester's body of research on APIs, making a cohesive whole from more than 85 reports.

This is an update to a previously published report. Major changes include the addition of digital bonding (Forrester's vision beyond REST APIs), more research on platform businesses, deeper detail on the relationship between APIs and microservices, and updated Forrester Wave™ evaluations of API management, API strategy services, and related areas. Significant changes since the prior edition of this report are marked with "[NEW]" or "[UPDATED]" as appropriate.

Key Takeaways

APIs Are Critical For Digital Business Success

APIs change business strategy by opening up new ways for organizations to reach new ecosystems and customers. Business and technology executives must recognize that APIs represent direct digital access to core business competencies and assets, enabling their organizations to play dynamically in multiple business ecosystems. These connections need more than the request-reply model of REST APIs, so the better framing is digital bonding, which encompasses a broader array of interaction models and technologies.

Strategy For APIs Must Be Incremental, Architected, And Governed

Street-level strategy is the right approach: Establish a lightweight vision for your API business strategy and then leverage each business change initiative to build your API taxonomy, architecture, platform, and governance.

Mature API Platforms Cover Six Major Areas

A strong platform for APIs and service-oriented architecture (SOA) leverages six major areas of technology: API design and documentation, API design and delivery infrastructure (e.g., application and integration servers and API gateways); service testing and virtualization; API management solutions; runtime service management; and service lifecycle management.

A Developer's Guide To Forrester's Strategies For API Success

Take A Guided Tour Of Forrester's Research On APIs And API Strategy



by [Randy Heffner](#)

with [Christopher Mines](#), Abigail Livingston, and Kara Hartig

September 19, 2019

Table Of Contents

- 2 APIs Are Critical For Digital Business Success
 - 7 How To Set Strategy For APIs
 - 9 Architecture Sets The Proper Context For Design Of APIs
 - 13 Effective API Strategy Requires Agile Processes And Governance
 - 15 Mature API Platforms Cover Six Major Areas
-
- Recommendations
- 19 Make Agile-Plus-Architecture The Foundation Of API Strategy

Related Research Documents

[A Developer's Guide To Forrester's Strategies For Integration And Digital Business Platforms](#)

[Digital Bonding: Expand Your API Strategy Beyond REST APIs](#)

[How APIs Reframe Business Strategy](#)



Share reports with colleagues.
Enhance your membership with Research Share.

FORRESTER

Forrester Research, Inc., 60 Acorn Park Drive, Cambridge, MA 02140 USA
+1 617-613-6000 | Fax: +1 617-613-5000 | [forrester.com](#)

© 2019 Forrester Research, Inc. Opinions reflect judgment at the time and are subject to change. Forrester®, Technographics®, Forrester Wave, TechRadar, and Total Economic Impact are trademarks of Forrester Research, Inc. All other trademarks are the property of their respective companies. Unauthorized copying or distributing is a violation of copyright law. [Citations@forrester.com](#) or +1 866-367-7378

A Developer's Guide To Forrester's Strategies For API Success

Take A Guided Tour Of Forrester's Research On APIs And API Strategy

APIs Are Critical For Digital Business Success

[UPDATED] Forrester often advises AD&D clients about APIs in the context of microservices, open banking, or some other targeted initiative. Often, clients ask about coordinating API strategy across multiple departments to improve solution delivery. Sometimes they focus on open web APIs, such as those available from big players such as Facebook, Google, and Pinterest. But our conversations don't often enough focus on APIs as a strategic investment to help enterprises thrive in a world of unpredictable digital disruption and transformation. This is a problem because:

- › **[UPDATED] API business strategy can create new business possibilities.** API business strategy is not just using APIs to update a traditional business strategy; API-infused business models and strategies create new ways of doing business, including platform business models that can extend an enterprise's reach even into new industries. The creativity and insight to conceive and develop API-infused ways of doing business require a nuanced mix of both business and tech savvy. In the same way that cardiology is a specialty area within medicine, API business strategy is best regarded as a specialty area within business strategy. The broader business context is the most important API discussion between business and technology execs, as we describe in these reports.

[How APIs Reframe Business Strategy](#)[APIs Turn Disruptions Into Business Opportunities](#)[\[NEW\] Four Ways APIs Are Changing Banking](#)[Brief: Four Ways APIs Are Changing Your Business](#)

- › **[UPDATED] API business strategy requires ecosystem thinking.** Priorities for digital transformation mean that executives — both business and technology — must put their organizations in play in multiple ecosystems of value, digitally connecting capabilities, assets, processes, and resources inside and outside their organizations. But it's not the old way of thinking of ecosystems as just the collection of partners and customers a firm works with; it requires 1) a much more open way of imagining new ways for digital connections to flow end to end across multiple enterprises to achieve customers' goals and 2) new ways to add value by plugging into that flow. Most industry conversation is about *providing* APIs, but ecosystem thinking also brings the value of *using* APIs from external providers, as discussed within the first three reports in this group.

[Tap Into Your Services Partner's Ecosystem](#)[\[NEW\] Assess Your Digital Insurance Capabilities](#)[APIs Foster Innovation At The Largest Global Firms](#)[The Digital Business Imperative](#)[Unlock The Business Value Of Your API Strategy](#)

A Developer's Guide To Forrester's Strategies For API Success

Take A Guided Tour Of Forrester's Research On APIs And API Strategy

- › **[NEW] API business strategy centers on relationships, not REST APIs.** Even in compelling conversations about API business strategy, the focus is too narrowly on REST APIs and their request-reply model rather than the full range of interaction styles and value flowing across enterprises. We call this broader view “digital bonding,” which centers on business relationships and encompasses many more options than REST.

[\[NEW\] Digital Bonding: Expand Your API Strategy Beyond REST APIs](#)**[NEW] APIs Enable Platform Business Models**

Platform business models are among the newer angles on business strategy that APIs enable. Although Alibaba's, Amazon's, and eBay's marketplaces, which are business platforms, have been around for more than a decade, it's only recently that more than a few other enterprises have moved in that direction (see Figure 1).

- › **[NEW] Platform strategy enables both new business models and core business agility.** Beyond marketplaces, platform business models can take numerous forms including running a platform that others use (e.g., GE Predix), having a platform that allows others to provide integrated value-add to your customers (e.g., Salesforce AppExchange), or even making your own core business capabilities available as a platform (e.g., Saxo Bank, Three Ireland).¹ Beyond these and other business opportunities, platform business strategy, by fostering creation of modular, pluggable business capabilities, opens wide the possibilities for directions your firm can go in the future. When disruptions get really big, you can rapidly reconfigure these modular capabilities and migrate completely away from outmoded business models. These reports describe the landscape of platform business models.

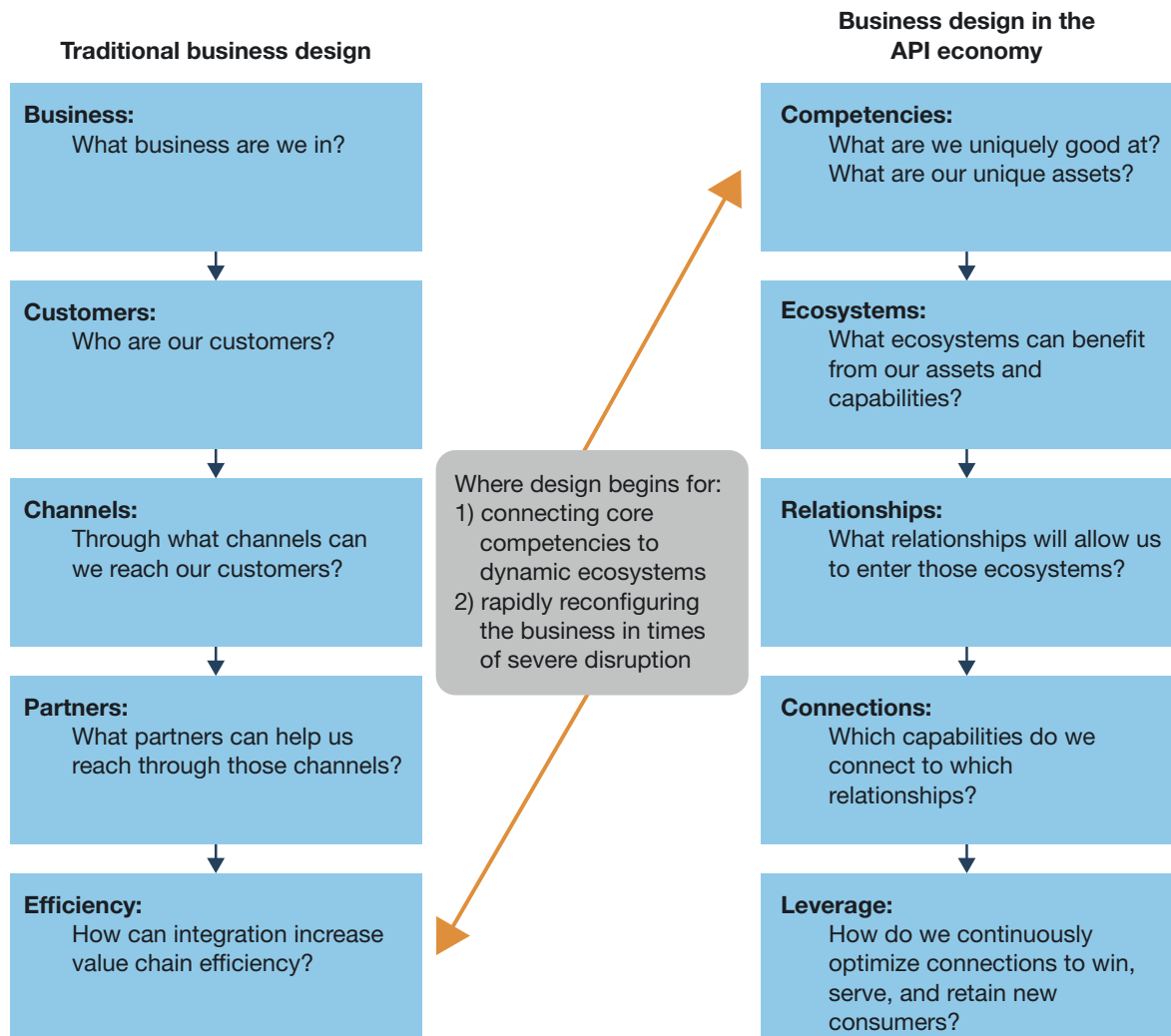
[\[NEW\] Earn Your Place In The Platform Economy](#)[\[NEW\] How To Build A Platform Business](#)

- › **[NEW] Platform businesses require a platform-to-the-core architecture and approach.** A platform business model without an underlying platform architecture is not really a platform. Achieving the modularity for rapid business reconfiguration demands modularity and alignment from business design to the major structures of the solution architecture and technology platform underneath. This begins by switching how executives think about the business. Instead of thinking of the whole business, which is then broken down into separate departments, start by designing business capabilities and then assembling them into a business — or part of a business — that suits a variety of ecosystems, perhaps even across vertical sectors. APIs play a central role as the main access mechanism for business capabilities. It takes more than APIs to make a platform business, but without APIs, you can't really have one at all.

[\[NEW\] Four Steps For Building A Platform Business](#)[APIs Underpin A Digital Business Platform](#)[\[NEW\] Build Technology Platforms To Accelerate Growth And Agility](#)

A Developer's Guide To Forrester's Strategies For API Success

Take A Guided Tour Of Forrester's Research On APIs And API Strategy

FIGURE 1 The Best Platform Business Strategies Require Conceiving The Business As A Competency Collection**Distinguish API Business Strategy From API Strategy**

Clearly, changing business strategy requires getting business executives onboard. The first step in doing so is for AD&D and other tech leaders to think of APIs as more than technology strategy and implementation, strategize like a businessperson, and:

- › **Drive the right conversations with business and technology executives.** A simple question like, “What’s our API business strategy?” may help frame strategy and justification conversations with business execs. However, the term “API” may get in the way; these reports offer guidelines for working with the executives.

A Developer's Guide To Forrester's Strategies For API Success

Take A Guided Tour Of Forrester's Research On APIs And API Strategy

[Brief: Don't Talk To Me About APIs!](#)[How To Sell API Strategy To Executives](#)

- › **[UPDATED] Make it clear that business strategy must interfuse with digital strategy.** Long past are the days when one could lead the market by developing an interesting business strategy and then tossing it over the wall for the techies to implement. Tech strategy must instead be an integral part of forming a business strategy. In API business conversations, AD&D pros and businesspeople must understand that almost anything can be a digital resource because either it is already digital (e.g., data, applications, and connected devices) or it can have a digital proxy (e.g., the location code posted by a public transit stop and a Twitter handle that serves as an entry point to the customer service team). And an organization's most important digital resources are its unique core competencies and business assets, which are the foundation of API-enabled business agility. It's not two things (i.e., business and technology); it's one thing: digital business, as these reports describe.

[Digitize Your Business Strategy With A Three-Phase Road Map](#)[\[UPDATED\] Embrace Next-Generation Digital Organization Structures](#)[Monetizing APIs: Help Execs Think Bigger, And Drive More Revenue](#)[Microservices And External APIs Underpin Digital Business](#)

- › **[UPDATED] Show how digital business requires agility for business capabilities.** APIs create business agility not so much through the raw merits of the technology but through the design models with which developers create them. At their best, API designs open access to business capabilities (e.g., business transactions and business data views), not technical applications. Thus, API enablement is key for creating agility for everything from new business models to optimization of everyday operations, both of which are part of a strong digital business road map. A foundation of API-based business capabilities fosters multiple angles into rapid business change and innovative business possibilities, as these reports describe.

[\[NEW\] The New Commerce Revolution: Off With Their Heads \(Or Not!\)](#)[AI APIs In The Cloud Are Here](#)[Use APIs And Components To Prepare For The Banking Industry's Increasing Pace](#)[Developing Tomorrow's Digital Experiences](#)[Selecting Tools That Enable Agility](#)[The Rise Of The Headless Content Management System](#)[\[UPDATED\] Build A Four-Tier Digital Engagement Platform](#)

A Developer's Guide To Forrester's Strategies For API Success

Take A Guided Tour Of Forrester's Research On APIs And API Strategy

Business APIs Embody Business Design In Modern Applications

To understand and pursue a mature and effective enterprise strategy for APIs, AD&D pros should begin by getting three key things crystal clear:

1. **[UPDATED] API strategy and design pull heavily from SOA best practices.** Despite continuing to hear industry voices discounting SOA as old and irrelevant, SOA best practices are still very much alive, well, and necessary. Don't be distracted by negative SOA comments — they're based on poor definitions of SOA.² Most SOA best practices have corollaries in API best practices, so use them to boost your API strategy. Often, clients find it best to retire the term "SOA" and simply use "APIs" as a blanket term for both. Alternatively, they could treat SOA as the part of their strategy for building core business agility and APIs as the strategy to extend their business agility to new contexts.³
2. **[UPDATED] APIs require a business design perspective first and foremost.** While you can (and should) use APIs for technical scenarios such as integration and application delivery, it is critical to understand that your most valuable and strategic APIs are those that embody business transactions and queries. These services provide business building blocks (rather than merely technical ones) for consistent business results, no matter from which customer touchpoint, B2B partner integration, or internal business process a transaction originates. Increasingly, we observe organizations using concepts of domain-driven design as part of their API and microservices strategy. This may extend to organizing teams around business domains, combined with applying trends toward product management discipline.

[\[NEW\] Put Product Management To Work In Software Development](#)

3. **An API taxonomy allows adapting lifecycles and governance based on API type.** The critical focus on business design means that an effective approach to services will differentiate between different types of services and be quite deliberate about design, coordination, and governance for business APIs. The following reports emphasize the critical need to distinguish between different types of APIs, with business APIs being the most important. It is also important to understand the types of APIs that your off-the-shelf applications offer.

[Keep API Strategy On Track With An API Taxonomy](#)[\[NEW\] Banks Need To Prepare For Banking Application Ecosystems](#)[Build SOA Success With A Business-Focused Approach To SOA Design And Governance](#)

A Developer's Guide To Forrester's Strategies For API Success

Take A Guided Tour Of Forrester's Research On APIs And API Strategy

How To Set Strategy For APIs

Armed with a business-focused and broad-based view of APIs and digital bonding, AD&D pros have the right perspectives to begin evolving the rest of their enterprise API strategy. Central to success is an incremental three-step approach that Forrester calls street-level strategy:⁴

1. **Craft a high-level vision — and stop there.** This is the strategy part of a street-level strategy. Don't write a 300-page architectural treatise about how to do APIs. Instead, outline key concepts and aspects of strategy at the "mile-wide, inch-deep" level. Create only enough detail to 1) make stakeholders aware of the breadth of business and technical considerations and possible investments; 2) identify (but not design) major API design and delivery processes, patterns for building APIs, and governance structures; and 3) guide just-in-time drilldowns and elaborations as part of step 2.
2. **Do projects, leveraging them for incremental strategy implementation.** This is the street-level part. Use each project to advance and mature the implementation of one or more aspects of API strategy. Use Agile-plus-architecture practices to collaboratively decide which aspects are most important to each project's success or which practices the project provides an excellent opportunity to develop.⁵ Don't worry too much if street-level investments don't take you straight toward the vision — some zigging and zagging is inevitable along the way.
3. **Use project experience to adjust the vision.** As you gain real-world experience with each project, you become much smarter about how to structure the vision and design the architecture to achieve it. So keep adjusting the vision and strategy by returning to step 1.

[UPDATED] Understand Distinct Starting Points For API Strategy

To help accomplish a first iteration of step 1, this report provides a comprehensive overview of eight major areas of maturity for APIs (the report uses "SOA" to refer to elements of API strategy aimed at building core business agility and "APIs" to refer to extension of the reach of that agility):

[Drive Business Agility And Value By Increasing Your API And SOA Maturity](#)

The two most important starting points for API strategy are:

- › **Business opportunity, four API categories, and product APIs.** To structure the business opportunity side of the strategy, start by understanding how four major categories of APIs create opportunity in different ways: open web, B2B, internal, and product APIs. The first three are commonly discussed in the industry (AKA public, partner, and private APIs). The fourth category, product APIs, is not often discussed but is critical as an alternate perspective for brainstorming possible APIs and business ecosystems. These reports define and describe the four categories and call out major aspects of API design for each.

[Establish Your API Design Strategy](#)

[Brief: Product APIs Create Distinct Customer Value And Opportunity](#)

A Developer's Guide To Forrester's Strategies For API Success

Take A Guided Tour Of Forrester's Research On APIs And API Strategy

- › **Business agility.** Continuing a major best practice theme from SOA, business agility is a critical focal point for API strategy. This centers mostly on the notion of business APIs that embody major business transactions and queries, without regard to the applications or implementations behind the API. This report tells the story of how business APIs (built when “SOA” was the term de jure) were a critical element of success in the merger that created EE, a UK-based telco, including key aspects of how EE approached Forrester's eight major areas of maturity for APIs and SOA.

[SOA Plays An Important Role In A Telco Merger](#)**Systems Integrators (SIs) And Consultants Can Help With API Strategy And Delivery**

[UPDATED] API business strategy is an entirely new dimension above and beyond typical SOA strategies, and an SI or consulting firm can help. Multiple factors complicate API strategy, making it more important to consider the potential benefits of outside help. Organizations without a strong SOA base have particularly good reason to do so — especially if they lack technical skill with API technology and design. Our research on API strategy and delivery services centers on providers that go beyond simply using APIs in building systems for clients; we focus on providers that intentionally organize and prepare to help clients build their own competency for API strategy. As clients seek to evaluate API services, two major points of guidance are critical:

- › **[UPDATED] Carefully assess providers' API business strategy competency.** Even if your firm does not now have big plans for external partnering, ecosystems, products, or customer integration via APIs, we recommend favoring providers that are strong in API business strategy. We do so because the design practices and governance models that foster great support for API business strategy more generally support good API practice. But in our evaluations, we find that API business strategy is also new to many service providers. In some cases, the term “API business strategy” means little more than technical API strategy based on good business requirements. This is a much lower bar than Forrester's view of APIs as a specialty area within business strategy, so Forrester advises careful vetting to ensure that your provider has the level of creativity you need for anticipating and inventing API-infused business models and strategies.
- › **[UPDATED] Carefully assess providers' API design competency.** API strategies may vary widely. Some may have immediate needs for external integration, others may center first on renewal of vintage applications, and others may focus on mobile apps, customer experience, internet of things (IoT), or something else. But all API strategies should center on business design. Rich API taxonomies and business APIs should guide your provider's design approach. Beyond that, there is no single set of cookie-cutter guidelines and best practices for an enterprise API program because APIs apply to a broad range of business scenarios. Even within one organization, there will be several threads of API strategy. Furthermore, one's API strategy should consider not only providing one's own APIs but also using APIs from a variety of other organizations and API providers.

These reports provide profiles of 22 key SIs and consulting firms for APIs and comparative analysis for 16 of those vendors.⁶

A Developer's Guide To Forrester's Strategies For API Success

Take A Guided Tour Of Forrester's Research On APIs And API Strategy

[\[UPDATED\] Now Tech: API Strategy And Delivery Service Providers, Q1 2018](#)[\[NEW\] The Forrester Wave™: Global API Strategy And Delivery Service Providers, Q2 2019](#)[\[NEW\] The Forrester Wave™: Specialist API Strategy And Delivery Service Providers, Q2 2019](#)

Architecture Sets The Proper Context For Design Of APIs

Business-focused design is the most critical design aspect for AD&D pros to concentrate on for APIs. However, because not all services are business APIs, a complete discussion of API design best practices must identify and account for the architecture context around various services. This sets the right foundation for detailed API design.

Put Strong Architecture Context Around APIs

Forrester's business-centered vision for the future of solution architecture shows how to organize the constantly expanding universe of technology infrastructure and options around business design concepts. At the center of Forrester's vision, business APIs embody an enterprise's core digital business capabilities. The architecture context begins by putting a finer point on this central position for APIs:

- › **Understand that interface design is the fulcrum of the architecture with APIs.** The very center of the architecture is the interface design for an API. Interface design, separate from the details of how one implements an API, is the fulcrum of the architecture. Interfaces are the leverage point, and, as with a mechanical lever, their placement (i.e., their designs) relative to other aspects of services is the single most important factor determining whether your API strategy machine will accomplish what you intended it to and how much work it will require to accomplish the strategy's goals.
- › **Set a business-centered solution architecture context around business APIs.** Around business APIs, which are the digital embodiment of an organization's business capabilities, Forrester's solution architecture vision positions other business-centered design points. Two key design points are user roles that multitouchpoint role-based workspaces serve and automated, monitored business processes that control the flow of customer engagement and work across an organization's internal and external ecosystems. Others are virtualized business views of data that align across siloed legacy and cloud solutions and business insight, control, and optimization that technologies like analytics, rules, collaboration, and events provide.

This report describes the vision; note that as a vision report intended to reframe enterprise thinking, it centers on the strategic endpoint, not the street-level path to get there.⁷

[The Future Of Solution Architecture: Six Business Design Focal Points](#)

- › **Extend the business-centered context into the integration architecture.** When developers apply APIs to integration scenarios, they often neglect to focus on business design. To counter this tendency, we have articulated an alternate vision for solution architecture from the perspective of integration strategy. We use the term “digital business design” for it to emphasize that integration

A Developer's Guide To Forrester's Strategies For API Success

Take A Guided Tour Of Forrester's Research On APIs And API Strategy

developers need to move away from a central focus on application silos and integration software to a central focus on designing business outcomes and then applying integration strategy to achieve them. Again, business APIs are key to this. This report provides the strategic vision without discussing the street-level path to get there.⁸

[Digital Business Design Is The New Integration](#)

- › **[UPDATED] Look for vertical sector API standards, but be patient.** Historically, technology standards drive and facilitate ecosystems. This could be true for APIs, except that few such standards exist and custom APIs can be quite useful for creating unique competitive advantage. Although healthcare has the HL7 FHIR API, telcos have TM Forum's open APIs, and, in banking, BIAN has even created sample implementations and a portal for its API standards. The general pattern is that industry groups such as ACORD (insurance) and Open Travel Alliance are still moving very slowly on API standards.⁹ Many verticals have message formats that may provide input to API design, but there are few that go as far as actual API design. In some sectors, consultancies, software firms, or collaborative groups publish vertical API specs, hoping they'll catch on (e.g., AgTech API and Open Bank Project). For European banks, the updated Payment Services Directive (PSD2) should have provided clear impetus for definitive vertical standards, but it didn't play out that way.

[\[NEW\] BIAN Takes A Big Step Toward The Promise Of Standard Banking APIs](#)[Standard APIs: A Key To Agile And Open Banking](#)[\[NEW\] Europe Lays The Foundations Of Open Banking](#)**Establish Strong Design Guidance For A Variety Of API Types**

Many AD&D pros put too much of their API design effort into theoretical discussions about REST, including HATEOAS, nouns versus verbs, HTTP error codes, and the like.¹⁰ These are useful discussions, but they miss more important concerns about designing comprehensible APIs, ensuring that developers understand the different types of APIs, and evolving coherent portfolios of APIs. Forrester's guidance begins with the layering needed when designing APIs for mobile and then continues with a comprehensive API design series:

- › **APIs for mobile apps require three major layers, each with different design concerns.** It's clear that APIs are needed for mobile, but beyond that, design guidance is often hard to find. The central concept is to plan for three major layers: 1) business APIs, as described earlier in this report; 2) multitouchpoint APIs, which provide common and familiar customer, employee, and stakeholder engagement across all of an organization's touchpoints; and 3) touchpoint-specific APIs, which handle specific requirements for presentation and engagement through individual touchpoints.

Multitouchpoint and touchpoint-specific APIs are best thought of as being part of the user experience layer of one's solution architecture. This report provides guidance and clarification, using a point-counterpoint structure to balance API design considerations.

A Developer's Guide To Forrester's Strategies For API Success

Take A Guided Tour Of Forrester's Research On APIs And API Strategy

How To Design APIs For Mobile

- › **[UPDATED] Understand that REST APIs are not the only option.** To help clients round out their comprehensive API design guidelines, Forrester has created a four-part series on API design. Although the industry is currently favoring REST-based APIs, the need for the broader view of digital bonding triggers the use of WebSockets, GraphQL, and event APIs even for external APIs.¹¹ Forrester clients shouldn't fear using SOAP when appropriate.¹² Our API design series begins by positioning various styles of REST-based messaging against SOAP and message-oriented middleware (MOM), including a comparison of pragmatic REST and high-end REST theory. When talking with practitioners that use APIs in context with microservices, we also encounter APIs based on using gRPC, Kafka, messaging queuing, or even Redis to communicate between microservices.¹³

API Design, Part 1: REST Is The Leading But Not Only Option For Your APIs

- › **Design APIs for the audience that will use them.** The second part of the series covers a broad set of basic considerations for API design, focusing heavily on REST while touching on scenarios where SOAP or MOM may be a better choice. A key part of Forrester's guidance includes consideration of the target audience for an API, including the possibility that multiple audiences may require different messaging styles. The report also notes multiple alternative styles for APIs, including JavaScript, web components, and language bindings via software development kits (SDKs).

API Design, Part 2: Design Messaging Styles By Balancing Reach With Your Other Design Goals

- › **Design APIs for high quality of service.** With all the discussion about REST being simple and easy, and with so many REST APIs being created for simple, quick, and low quality-of-service (QoS) data access, there is a dearth of guidance in the industry on how to achieve high QoS with REST. To address this gap, transaction management and error handling are key parts of the third report in Forrester's API design series.

API Design, Part 3: Make Transactions And Error Handling Clear In Your API Designs

- › **Round out design guidelines with security and future proofing.** The last part in the API design series outlines five major scenarios for trust enablement with APIs (i.e., authentication and authorization), including ones for third-party authorization using OAuth2. The other major topic that the report addresses is designing APIs for future change, which requires balancing open-ended design with data integrity. OAuth2 requires careful consideration, planning, and administration, especially when used in the classic three-legged scenario (e.g., involving customer, enterprise, and app developer) it was originally created for.¹⁴ API designers can learn from security practices for mobile apps as well.

API Design, Part 4: Future-Proof And Secure Your APIs To Fit Your Usage Scenarios

Mobile Solution Architecture: User Authentication Is The Foundation Of Mobile Trust

A Developer's Guide To Forrester's Strategies For API Success

Take A Guided Tour Of Forrester's Research On APIs And API Strategy

[UPDATED] Define And Drive The Right Relationships Between Microservices And APIs

As a final note on API design, the industry conversation about microservices is growing, but great confusion persists over what microservices are, how to design them, and how they relate to APIs. Often developers use “API” and “microservice” interchangeably, but we find it best to think of them separately. We observe two ways that practitioners think about microservices:

- › **Microservices as a component-based and sometimes container-based structure.** In this definition of microservices, which Forrester strongly favors, container-based technologies such as Kubernetes and Docker or serverless technologies like function-as-a-service (FaaS) structure an application as a collection of relatively small separately deployable units.¹⁵ This use of microservices borrows heavily from early-2000s concepts of component-based development, including the potential for marketplaces for components.¹⁶ A microservice may be part of the UI layer (i.e., a visual component) or part of the business logic layer (i.e., nonvisual), and it may be accessed using a number of different messaging styles, of which REST APIs are but one option. In our Forrester Analytics Global Business Technographics® Developer Survey, 2019, 43% of developers said their firms were currently using microservices, and of these, 22% said their microservice-based applications were composed of many container images.
- › **Microservices as an API design concept.** In this definition, a microservice is thought of as a small API. We strongly discourage this usage, primarily because of the points above that a microservice can be visual or nonvisual and accessed in many ways. Nonetheless, with this usage, we commonly hear the phrase, “An API/microservice should do one thing and do one thing well” — especially in contrast to worst-practice ways that SOA services were created. In best practice, it should be “one thing done well *from the perspective of the service user*” — meaning that one will have coarse-grained business APIs (e.g., submitOrder) to achieve consistent results across all customer touchpoints, very fine-grained services (e.g., a type-ahead API), and all sizes in between.

These two notions of microservices may be used together or separately, but we find it best to use only the first definition. The second definition causes confusion by conflating APIs and microservices when the two concepts should be separate. These reports elaborate on the definitions of microservices and place them in the context of APIs and other major shifts in application delivery.

[Microservices Have An Important Role In The Future Of Solution Architecture](#)[Designing Microservice Apps For Containers And Cloud Platforms](#)[Application Modernization, Service By Microservice](#)[How To Capture The Benefits Of Microservice Design](#)

A Developer's Guide To Forrester's Strategies For API Success

Take A Guided Tour Of Forrester's Research On APIs And API Strategy

[NEW] Use Domain Design Concepts To Align APIs, Microservices, And Service Meshes

Service mesh technology is a rising part of the landscape for microservice architectures (e.g., Istio, Linkerd, and Consul Connect).¹⁷ It is so early in the cycle of maturity for service meshes that vendors have a difficult time finding customers in production that Forrester can talk to, but a couple of key emerging practices are becoming clear in our research:

- › **[NEW] Service mesh planning and design is bounded using domain-driven design.** A conversation about service mesh technology may start with a question like, “How will thousands of microservices find each other?” The question is off the mark because the landscape of microservices should be divided into domains (e.g., revenue management, order fulfillment, and platform management), which dramatically reduces the size of the problem to 1) microservices finding other services within a single domain (so-called “east-west” traffic) and 2) key microservices representing the domain’s touchpoints with other domains (so-called “north-south” traffic).
- › **[NEW] Domain edges are the primary locale where APIs and microservices touch.** Although individual microservices within a domain may or may not communicate via REST APIs, the more important intersection point for APIs, microservice architecture, service meshes, and API management solutions is at the domain edge (i.e., the services, events, and other interactions between domains). Thus, for example, for an API at the edge of a domain, developers would highly likely publish it via an API management solution to permit access from other internal (or external) domains.¹⁸
- › **[NEW] Domain edges also provide bounding for relating data models and APIs.** To understand how data definitions, microservices, and API definitions relate, consider an example of an invoice in telecommunications. In the billing domain, an invoice is a very complex entity that must address a variety of concerns from regulatory and taxing entities to rate plans, product structures, bundling, and discounts — all of which help determine specific line items on a bill. In the customer service domain, an invoice is a simpler entity that shows the final results of these calculations. What does this mean? Rather than having a single data model across both domains, each needs its own specific model. However, the models must align at the edges of the domains, and business APIs (i.e., data in motion) are a primary embodiment of this alignment. Inside a domain, microservice designers can innovate and optimize.

Effective API Strategy Requires Agile Processes And Governance

A key reason to have strong design guidance for APIs is that understanding the various types of APIs acts as a foundation for AD&D pros to set the right governance strategies, whether they use formal Agile methods or their own flavor of DevOps and continuous integration/continuous delivery. Some APIs require less governance (e.g., touchpoint-specific APIs for mobile), while others require more (e.g., core business APIs). Either way, governance is important because:

A Developer's Guide To Forrester's Strategies For API Success

Take A Guided Tour Of Forrester's Research On APIs And API Strategy

- › **No governance means an incoherent collection of APIs and negative customer impact.** If application delivery teams simply toss together whatever APIs seem good for their immediate purpose, without collaborating across teams, they may achieve good results for their isolated applications, but the enterprise as a whole will achieve little, if any, synergy with APIs. This is a worst practice for SOA, and with APIs, it still hurts the enterprise by leading to disconnected and confusing customer engagement across touchpoints, unreliable transaction handling, inefficient and duplicative back-end processes, and unnecessary costs for developing and maintaining duplicate APIs.
- › **Agile-plus-architecture provides a foundation for collaborative success.** Streamlined governance structures, developer-architect connections, and multilayered architecture collaboration are all best practices for infusing architecture governance into Agile and continuous delivery methods, which can prevent bad results. With Agile-plus-architecture, delivery teams gain the requisite context to design APIs that fit within a broader portfolio, and architects' activities better focus on the near-term context and needs of delivery teams. These reports describe more than 30 Agile-plus-architecture best practices across four major categories: business architecture and project context, project delivery guidance and governance, architecture management and technology selection, and organization and culture. And they correlate with top-line business outcomes.

[Best Practices For Agile-Plus-Architecture](#)[\[NEW\] Forrester Infographic: Drive Digital Business Success By Applying Agile-Plus-Architecture](#)

These two reports provide background and support for Agile-plus-architecture with Forrester survey results that highlight ways that architects and developers work better together.

[Brief: Developers Get More Architecture Oversight — And They Like It](#)[A Guide To More Effective Developer-Architect Relations](#)

- › **[UPDATED] API portfolio management is a key governance discipline.** Among the SOA best practices that help structure Agile-plus-architecture collaboration for APIs is API portfolio management, which guides design and evolution using lightweight definitions of target API portfolios. For business APIs, portfolios typically center on business domains or capability areas (e.g., billing and customer engagement). Teams use API portfolios to identify when to build new APIs and to more reliably identify which APIs to use on any given project. API portfolio management is a valuable but not frequently used practice. So AD&D leaders may draw from broader strategic portfolio management disciplines and scope them down to APIs.

[\[NEW\] Use BT Road Maps To Drive Strategic Portfolio Management](#)

In addition to being an overall API best practice, API portfolio management is often a key responsibility of a center of excellence, as described in these earlier reports on SOA.

[Survey Results Show SOA Governance Improves SOA Benefit Realization](#)

A Developer's Guide To Forrester's Strategies For API Success

Take A Guided Tour Of Forrester's Research On APIs And API Strategy

[The Five Most Valuable SOA Governance Practices](#)[SOA Centers Of Excellence: The Five Most Valuable Practices That Keep SOA On Track](#)

Mature API Platforms Cover Six Major Areas

Although API management is the hot new product category for APIs, a mature platform for APIs requires much more than API management:

[Defining A Platform For API Success](#)[How To Manage APIs For Customer Engagement](#)

Forrester identifies six major elements of a mature API platform — API security is a pervasive concern across all six areas (see Figure 2):

- › **API design and documentation.** API interface design is the single most important aspect of API strategy. But even as standalone API design tools emerge, AD&D pros must step back and take a solution approach to the tooling and processes they use for API design and documentation. Why? Because multiple product categories embed API design and documentation features and because six major disciplines intersect here: API design and documentation; API creation with integration tools; API creation with app dev tools; API management solutions; API product management; and API portfolio management.

[Create Great API Designs And Documentation With Integration Across The API Life Cycle](#)

- › **API creation and delivery.** This area covers a broad array of alternatives for where and how an API implementation runs — vintage mainframe applications, Node.js, microservices, containers, cloud platforms, hybrid integration, Java EE, .NET, SaaS apps, or anywhere else.¹⁹ APIs from these sources may permit access directly or through integration software or an API gateway.²⁰ Diverse options for implementing APIs mean that a variety of old and new products may combine to form an organization's API creation and delivery platform, and these may include APIs consumed from an extended ecosystem of partners. Besides the runtime platforms where API implementations operate, API creation requires DevOps tooling and processes. API security touches all parts of an API platform. These reports provide a sampling of the many connections among API strategy, API implementation, and other technology domains and trends.

[How To Set The Right Strategy For SaaS Integration](#)[\[UPDATED\] The Forrester Wave™: Strategic iPaaS And Hybrid Integration Platforms, Q1 2019](#)[\[UPDATED\] Now Tech: iPaaS And Hybrid Integration Platforms, Q3 2018](#)[\[UPDATED\] Emerging Technology Spotlight: Serverless Computing](#)[The Three Faces Of Platform-As-A-Service](#)

A Developer's Guide To Forrester's Strategies For API Success

Take A Guided Tour Of Forrester's Research On APIs And API Strategy

[\[NEW\] The State Of Application Security, 2019](#)[\[UPDATED\] Digital Transformation Requires Development Transformation](#)[Demystifying Hybrid Solutions And Architectures](#)[The Software-Defined Data Center Comes Of Age](#)[The Forrester Tech Tide™: Identity And Access Management, Q4 2017](#)[Reform Legacy Operations For Composable Infrastructure](#)

- › **API testing and virtualization.** Every API must be thoroughly tested on its own, separately from any application that calls it.²¹ This includes functional testing (including verification that authentication and authorization function properly), performance testing, and service virtualization. Service virtualization allows developers to test using a simulated implementation of an API, letting API users proceed with development without waiting for API development or enhancement. This parallel development facilitates delivery speed. However, Forrester often observes that clients do not place enough priority on tools for repeatable automated API testing. These two reports provide a foundation for API testing tool selection.²²

[\[NEW\] Now Tech: Omnichannel Functional Test Automation Tools, Q1 2018](#)[\[NEW\] The Forrester Wave™: Global Continuous Testing Service Providers, Q1 2019](#)[Vendor Landscape: Application Security Testing](#)[The Forrester Wave™: Modern Application Functional Test Automation Tools, Q4 2016](#)

- › **API management solution.** This element centers on the relationships between API users and API providers. Users may be internal or external to the provider's organization. The core elements of an API management solution are an API user portal (AKA developer portal), an API product admin portal, and an API gateway. Although the gateway enforces security (and other) policies, it is *not* the most important element of API management. Instead, think of API user relationship management as the center point, with the gateway there mainly to enforce usage agreements between API users and API providers. (Later in this report we provide links to our vendor research on API management solutions.)
- › **[UPDATED] Runtime service management.** This element ensures top-quality API operations by monitoring APIs across the various layers of implementation behind API interfaces. API management solutions typically provide limited features, with only interface-level visibility. Runtime service management provides deeper visibility to pinpoint a root cause at a specific point in an API call chain. Industry developments for monitoring microservices bring focused attention to observability and visibility across call chains. AD&D pros and operations teams should evaluate three major alternatives: log analytics, using open source tools like an ElasticSearch, Logstash, and Kibana (ELK) stack or commercial tools like Splunk (open tracing and other microservices topics

A Developer's Guide To Forrester's Strategies For API Success

Take A Guided Tour Of Forrester's Research On APIs And API Strategy

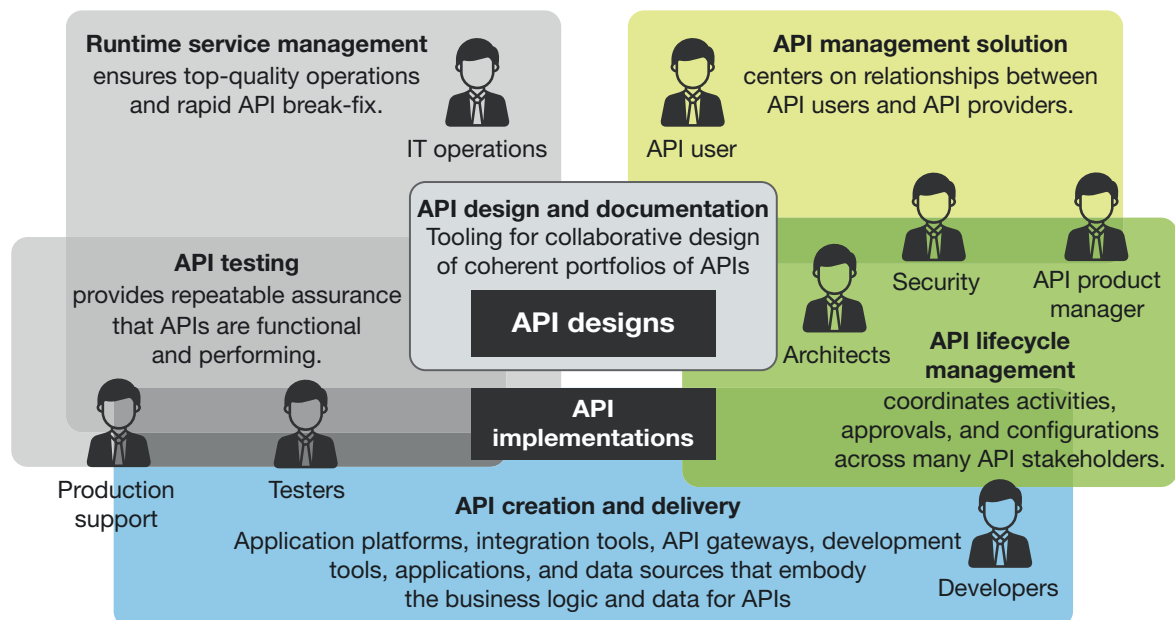
fit here); application performance management tools like Dynatrace or AppDynamics; or traditional SOA management tools like Software AG's webMethods Insight.²³ Early in the SOA days, runtime service management showed its value as a product category for services.

[NEW] The Forrester Wave™: Intelligent Application And Service Monitoring, Q2 2019

SOA Product Adoption: SOA Management Solutions Provide The Strongest Benefit

- › **Formal API lifecycle management.** Aside from managing usage and operations of APIs, a mature API program has defined and managed lifecycles for creating various types of APIs and services. For example, based on an API's type or categorization, lifecycle management might ensure teams perform required security reviews. API management, runtime service management, and API lifecycle management combine to provide comprehensive management of APIs. In contrast to Forrester's formal use of the term to refer to strong tooling for process automation and control, vendors often use "lifecycle management" very loosely. Formal lifecycle management requires strong organizational maturity and discipline, but it can deliver critical value and control for organizations that employ it properly. API management solution vendors with strong investments in formal lifecycle management include Oracle, Perforce (via its acquisition of Rogue Wave Akana), Sensedia, Software AG, and WSO2. Google and IBM have lightweight lifecycle management features.²⁴

FIGURE 2 A Comprehensive API Platform Has Six Major Areas Of Tools And Infrastructure



A Developer's Guide To Forrester's Strategies For API Success

Take A Guided Tour Of Forrester's Research On APIs And API Strategy

Diverse Requirements Feed A Diverse Market For API Management And API Gateways

Among these six elements of API platforms, API management solutions currently have the greatest current interest among Forrester's clients. Because API use occurs in many diverse scenarios, there is room in the market for a variety of styles of API management solutions from a variety of vendors. Forrester fields many client inquiries on the space because:

- › **[UPDATED] API management is a rapidly growing market with five major profiles.** API management solution vendors have varying approaches to the needs and requirements surrounding API user relationship management. We organize these into five major functionality profiles, and solutions may provide one or more of them. The narrowest profile, focused API publishing, is the foundation — all solutions can support it, but some will be overkill if that's all an enterprise wants. The other four profiles are API user relationship management, API products and billing, federated API ecosystems, and enterprise API governance. The landscape includes integration and platform vendors (e.g., Axway, Google, IBM, Microsoft, MuleSoft, Oracle, Red Hat, SAP, Software AG, and TIBCO Software) and open source solutions among a total of 22 vendors. Since our Q4 2018 Now Tech analysis, Amazon Web Services has released an API user portal companion to its API gateway service as an open source project.

[\[UPDATED\] Now Tech: API Management Solutions, Q4 2018](#)

- › **[UPDATED] API management solutions vary widely.** Our detailed look at 15 API management vendors shows a widely diverse market. This is appropriate because enterprise strategies for APIs vary widely. Clients should construct their shortlists by carefully examining and cross-comparing their API strategy and the characteristics of both high- and low-scoring vendors in our Forrester Wave™ analysis of the space. For example, clients whose API strategies demand high customization and convenient billing for APIs may find that a low-scoring vendor in our analysis provides a simpler base to work from than a high-scoring vendor. Since our last Forrester Wave analysis, the most significant developments include IBM's completion of its acquisition of Red Hat (IBM has said it will maintain both of the two vendors' API management solutions). Also, MuleSoft integrated Salesforce's community features, resulting in stronger API user relationship features; Red Hat 3scale is now a fully open source solution; and Perforce acquired Rogue Wave.

[\[UPDATED\] The Forrester Wave™: API Management Solutions, Q4 2018](#)

- › **[UPDATED] Some misuse “API management” for integration and API gateways.** Some players in the market misuse the term “API management.” As Forrester defines it, an API management solution must have three elements: 1) an admin portal for API providers to define available APIs and policies for their use; 2) an API user portal suitable for engaging an external audience; and 3) an API gateway to enforce the agreements between API users and API providers. Thinking that rate-limiting and security or an API catalog for internal developers are all that effective management of relationships between API users and API providers needs, some vendors (e.g., Dell Boomi and SnapLogic) refer

A Developer's Guide To Forrester's Strategies For API Success

Take A Guided Tour Of Forrester's Research On APIs And API Strategy

to integration products or API gateways as API management. Forrester strongly disagrees with this view because the complexities of managing API user-provider relationships demand a richer business application feature set — particularly for relationships with external API users.

- › **API/messaging gateways continue to have market space on their own.** Every API management solution includes embedded API gateway functionality, but gateways also continue on as a standalone market space, protecting not only APIs but also application messaging, WebSockets, file transfers, and more. Traditional edge-of-the-enterprise scenarios continue — albeit increasingly served by an API management solution — but the variety and usage patterns for standalone gateways are expanding. Gateways may be embedded inside an application — especially a microservice-based application. Focused special-purpose gateways (e.g., Amazon API Gateway and Apache Knox) serve narrow ranges of endpoints (e.g., a given cloud platform and Hadoop). The market for API gateways has four concentric rings of increasing message style support (REST, SOAP, application messaging, and file transfer). These reports describe the market for API and messaging gateways.

[Vendor Landscape: API Gateways](#)[Brief: Amazon API Gateway Is Limited But Useful](#)**Recommendations**

Make Agile-Plus-Architecture The Foundation Of API Strategy

[UPDATED] When Forrester reviews clients' API strategies, the three most often missing elements are API business strategy, API taxonomy, and API portfolio management. Besides ensuring that no one misses these, across all of Forrester's advice for APIs, SOA, and service-based strategy, AD&D pros who build an API strategy must ensure an Agile-plus-architecture foundation. It is only by combining the two that an enterprise can meet and sustain the speed-of-delivery promises of APIs and services. Furthermore, only by combining the two can an enterprise evolve coherent portfolios of APIs and services within each major business domain. Each report we've referenced here has other important recommendations for AD&D pros for specific topic areas, and their recommendations will deliver better results with a street-level strategy that has Agile-plus-architecture best practices as the foundation.

A Developer's Guide To Forrester's Strategies For API Success

Take A Guided Tour Of Forrester's Research On APIs And API Strategy

Engage With An Analyst

Gain greater confidence in your decisions by working with Forrester thought leaders to apply our research to your specific business and technology initiatives.

Analyst Inquiry

To help you put research into practice, connect with an analyst to discuss your questions in a 30-minute phone session — or opt for a response via email.

[Learn more.](#)

Analyst Advisory

Translate research into action by working with an analyst on a specific engagement in the form of custom strategy sessions, workshops, or speeches.

[Learn more.](#)

Webinar

Join our online sessions on the latest research affecting your business. Each call includes analyst Q&A and slides and is available on-demand.

[Learn more.](#)



Forrester's research apps for iOS and Android.

Stay ahead of your competition no matter where you are.

Endnotes

- ¹ Saxo Bank started its platform business model with a trading platform; O2 Ireland (later acquired by Three Ireland) offered its billing capabilities as a service separately from its telco capabilities. See the Forrester report "[How APIs Reframe Business Strategy](#)" and see the Forrester report "[Digital Business Design Sharpens Organizations' Competitive Posture](#)."
- ² More evidence that SOA is alive and well is the number of conference sessions on or references to SOA at major vendor events. After one IBM event, Forrester described what we saw. Source: Randy Heffner, "Sorry, Kids: APIs Have Not And Will Not Kill SOA," Forrester Blogs, May 2, 2014 (https://go.forrester.com/blogs/14-05-02-sorry_kids_apis_have_not_and_will_not_kill_soa/).
- ³ Forrester describes the different ways to relate the terms "SOA" and "APIs" as a foundation for understanding API and SOA maturity. See the Forrester report "[Drive Business Agility And Value By Increasing Your API And SOA Maturity](#)."
- ⁴ Rather than use a broad-based, top-down plan for implementing architecture changes like APIs, Forrester recommends a success-first approach: Identify an important upcoming business decision and learn just enough of the right aspects of API strategy to make a significant move toward API maturity. Then build from your street-level situation toward a long-term API vision and strategy. See the Forrester report "[Build Your Stepwise Strategy For Business-Centered EA](#)."

A Developer's Guide To Forrester's Strategies For API Success

Take A Guided Tour Of Forrester's Research On APIs And API Strategy

- ⁵ Organizations must be able to sustain and increase their rate of change over time. Agile development practices and continuous delivery are essential tools, but so is an architecture that enables resilience. Combining Agile and architecture is challenging because their respective goals of delivering now and preparing for the future often appear to be at odds. Forrester outlines the challenges that developers and architects face in trying to collaborate and identifies important resources that both sides can draw upon. See the Forrester report "[Agile-Plus-Architecture: Embrace The Oxymoron.](#)"
- ⁶ On updating and expanding our overview of key service providers that can help with API strategy and delivery, see the Forrester report "[Now Tech: API Strategy And Delivery Service Providers, Q1 2018.](#)"
- ⁷ Two separate reports provide perspectives and decision models for street-level strategy for evolving toward Forrester's vision for the future of solution architecture. See the Forrester report "[The Future Of Solution Architecture, Part 1: Business Processes Within A Capability](#)" and see the Forrester report "[The Future Of Solution Architecture, Part 2: User Roles Within A Business Capability.](#)"
- ⁸ A separate report provides a perspective on street-level strategy for evolving toward Forrester's vision for digital business design. See the Forrester report "[How To Implement Digital Business Design.](#)"
- ⁹ FHIR: Fast Healthcare Interoperability Resources. Source: FHIR (<https://www.hl7.org/fhir/http.html>).
- BIAN: Banking Industry Architecture Network. Source: BIAN (<https://portal.bian.org/landing>).
- ACORD; Association for Cooperative Operations Research and Development.
- Telemanagement Forum's open APIs may be found at the following website. Source: TM Forum (<https://www.tmforum.org/open-apis/>), and ACORD (<https://www.acord.org/>).
- ¹⁰ HATEOAS: hypermedia as the engine of application state.
- ¹¹ REST-only strategies don't match the full range of business relationship dynamics. Digital bonding strategy works better than REST-only by encompassing a broader array of possibilities and interaction models. Early adopters in finance use WebSockets in their digital bonding strategies for market data and cryptocurrencies. GraphQL has broader vertical usage, although many GraphQL APIs are marked as beta or experimental. A wide range of new and old styles and mechanisms are possible including events, streaming, web components, AsyncAPI, EDI, B2B portals, and file transfer. See the Forrester report "[Digital Bonding: Expand Your API Strategy Beyond REST APIs.](#)"
- ¹² SOAP: Simple Object Access Protocol.
- ¹³ Communication between microservices is one of 30-plus emerging areas of best practice for microservice development. See the Forrester report "[Designing Microservice Apps For Containers And Cloud Platforms.](#)"
- ¹⁴ Forrester notes two particular major concerns with OAuth2. First, in a typical OAuth2 scenario, customers will have a security dialog suddenly pop up in front of them asking them to provide access to their data. This may have the unintended side effect of training customers to click "yes" on things that randomly pop up on their screen. An alternative is to provide a page in customers' self-service portal where they can manage which apps have access to their data. You should provide such a page in any case so that customers can revoke access they previously granted. Second, only a few API management solutions provide strong administration of the specification of OAuth privileges for APIs. Forrester defines a best practice we call "closed loop OAuth," which we strongly recommend for firms that use OAuth2 with their APIs. See the Forrester report "[The API Management Buyer's Guide, Q4 2016.](#)"
- ¹⁵ There is a quiet revolution underway in software development that leverages openly available services fronted by APIs, service-rich platforms, and deployment technologies like microservices and containers. See the Forrester report "[From Application Design To Application Composition.](#)"
- ¹⁶ It is clear that OS-level containers are beneficial for application architectures and lifecycles, but several important questions remain about Docker. Forrester clarifies for app developers the most significant things that are known about Docker and the critical questions that remain. See the Forrester report "[Nine Questions To Ask About Docker.](#)"

A Developer's Guide To Forrester's Strategies For API Success

Take A Guided Tour Of Forrester's Research On APIs And API Strategy

¹⁷ Other major service mesh offerings include AWS App Mesh, Decipher Technology Studios Grey Matter, Pivotal Service Mesh, Red Hat OpenShift Service Mesh, Solo.io Gloo, F5 Networks Aspen Mesh, and VMware NSX Service Mesh. Other open source service meshes include Kuma, Mesher, Meshery, and SOFAMesh.

¹⁸ In this view, there are two major granularities of microservices and two corresponding levels of APIs. Individual microservices are the low-level, individually deployable units, while large-grained microservices (or domain services) package the domain's capabilities for the world outside the domain. Each of these may be accessed via APIs, but the lower-level APIs are typically private to the team that owns the domain and thus not published broadly to other teams. Check Figure 2 in the following report. See the Forrester report "[Designing Microservice Apps For Containers And Cloud Platforms](#)."

¹⁹ JavaScript and the Node.js runtime environment in particular are becoming an increasingly important part of an enterprise environment. See the Forrester report "[The Dawn Of Enterprise JavaScript](#)."

²⁰ Historically, the terms "enterprise service bus" (ESB) and "application integration server" have been used interchangeably. As cloud-based integration platforms have gained momentum, Forrester has begun using two different terms to refer to the space: integration platform-as-a-service (iPaaS) and hybrid integration. Most of the vendors we classify as iPaaS originated as cloud-based integration products. Most of the traditional ESB/integration server products have evolved into hybrid integration products. Thus, Forrester reports tend to use "ESB" only when referring to architectural patterns for creating APIs and SOA services using an integration product (iPaaS or hybrid).

²¹ With increasingly complex applications and layering, it is important to automate tests at all layers of an architecture by going beyond GUI automation testing to testing at the API, service, and process levels. See the Forrester report "[Five Must-Do's For Testing Quality At Speed](#)."

²² Our most recent Forrester Wave for service virtualization was completed in 2014. See the Forrester report "[The Forrester Wave™: Service Virtualization And Testing Solutions, Q1 2014](#)."

²³ Although application performance management and log analytics can provide deeper visibility for monitoring APIs, neither is API-centric the way SOA management tools are, nor are they built for rich participation in comprehensive API governance. In addition to webMethods Insight, key players in traditional SOA management include Rogue Wave (via its acquisition of Akana) and WSO2. Because there is a slow rate of change in SOA management products, we will not be updating our Forrester Wave in the space. Since our most recent Forrester Wave, the most significant change is that Progress Software sold its Actional product (and other products) to Aurea. Software AG continues to OEM Aurea Actional as webMethods Insight. SOA Software renamed itself Akana and then was acquired by Rogue Wave. See the Forrester report "[The Forrester Wave™: Standalone SOA Management Solutions, Q4 2011](#)." Our most recent Forrester Wave for service virtualization was completed in 2014. See the Forrester report "[The Forrester Wave™: Service Virtualization And Testing Solutions, Q1 2014](#)."

²⁴ We will not be updating our most recent Forrester Wave on service lifecycle management, but the functionality appears as a part of our API management solutions Forrester Wave. The changes to the vendor landscape are that 1) Sensedia entered the market and later integrated its product into its API management solution; 2) SOA Software was renamed to Akana and then acquired by Rogue Wave; 3) IBM deemphasized its WebSphere Registry Repository product, choosing to not carry it forward as a key part of its API Connect API management solution; 4) HP sold its SOA Systinet product to Micro Focus; and 5) Progress Software's solution was discontinued. See the Forrester report "[The Forrester Wave™: SOA Service Life-Cycle Management, Q1 2012](#)."

We work with business and technology leaders to develop customer-obsessed strategies that drive growth.

PRODUCTS AND SERVICES

- › Core research and tools
- › Data and analytics
- › Peer collaboration
- › Analyst engagement
- › Consulting
- › Events

Forrester's research and insights are tailored to your role and critical business initiatives.

ROLES WE SERVE

Marketing & Strategy Professionals

CMO
B2B Marketing
B2C Marketing
Customer Experience
Customer Insights
eBusiness & Channel Strategy

Technology Management Professionals

CIO
› Application Development & Delivery
Enterprise Architecture
Infrastructure & Operations
Security & Risk
Sourcing & Vendor Management

Technology Industry Professionals

Analyst Relations

CLIENT SUPPORT

For information on hard-copy or electronic reprints, please contact Client Support at +1 866-367-7378, +1 617-613-5730, or clientsupport@forrester.com. We offer quantity discounts and special pricing for academic and nonprofit institutions.