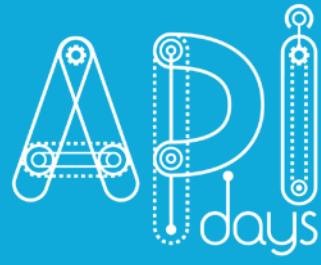




NOVEMBER 2015



OPENBANKPROJECT

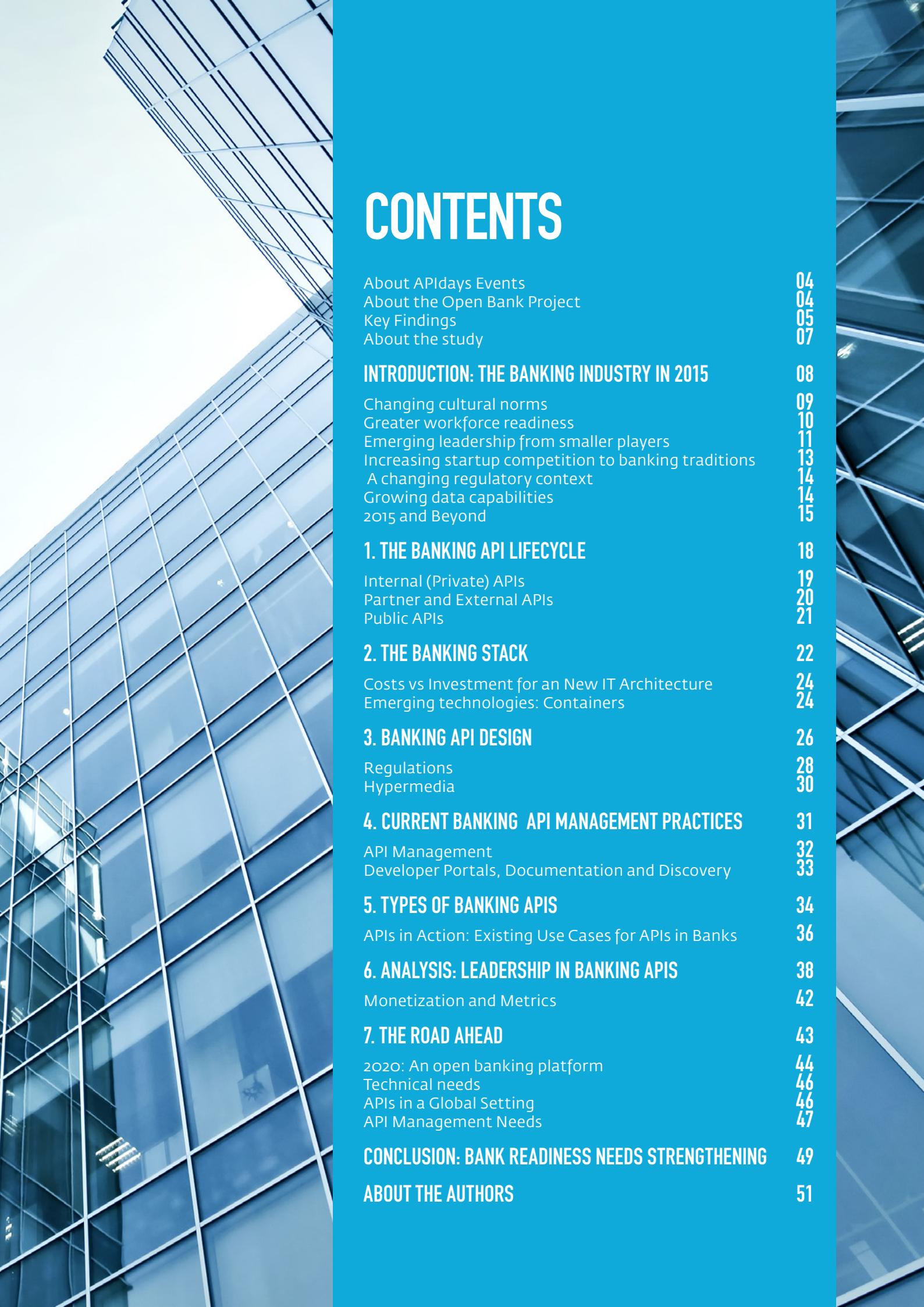
BANKING APIs STATE OF THE MARKET

INDEPTH INTERVIEWS FROM 22 LEADING BANKS AND FINTECH
INSTITUTIONS ON THEIR CURRENT AND FUTURE USE OF APIS



ca
technologies





CONTENTS

About APIdays Events	04
About the Open Bank Project	04
Key Findings	05
About the study	07
INTRODUCTION: THE BANKING INDUSTRY IN 2015	08
Changing cultural norms	09
Greater workforce readiness	10
Emerging leadership from smaller players	11
Increasing startup competition to banking traditions	13
A changing regulatory context	14
Growing data capabilities	14
2015 and Beyond	15
1. THE BANKING API LIFECYCLE	18
Internal (Private) APIs	19
Partner and External APIs	20
Public APIs	21
2. THE BANKING STACK	22
Costs vs Investment for an New IT Architecture	24
Emerging technologies: Containers	24
3. BANKING API DESIGN	26
Regulations	28
Hypermedia	30
4. CURRENT BANKING API MANAGEMENT PRACTICES	31
API Management	32
Developer Portals, Documentation and Discovery	33
5. TYPES OF BANKING APIS	34
APIs in Action: Existing Use Cases for APIs in Banks	36
6. ANALYSIS: LEADERSHIP IN BANKING APIS	38
Monetization and Metrics	42
7. THE ROAD AHEAD	43
2020: An open banking platform	44
Technical needs	46
APIs in a Global Setting	46
API Management Needs	47
CONCLUSION: BANK READINESS NEEDS STRENGTHENING	49
ABOUT THE AUTHORS	51

ABOUT APIDAYS EVENTS

APIdays is a series of open international events about APIs and the programmable web. We bring together innovators, decision makers and technicians from both traditional industries and API businesses.

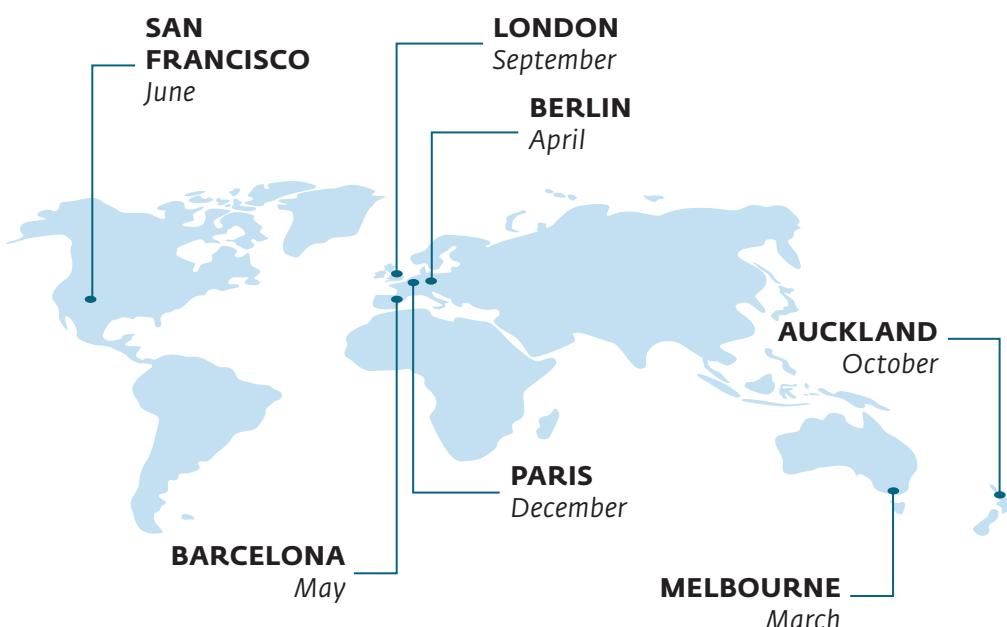
In 2015, we have extended the APIdays global conference series to London, where we partner with Open Bank Project to host an annual APIdays conference specifically focused on banking and fintech APIs.

For more information, please email contact@apidays.io or visit apidays.io

ABOUT THE OPEN BANK PROJECT

Led by Berlin based TESOBE, the Open Bank Project is an open source API and App store for banks that empowers financial institutions to securely and rapidly enhance their digital offerings using an ecosystem of 3rd party applications and services. We assist banks in executing effective API strategies by providing a proven API platform supported by an active community of developers and partners.

For more information, please visit www.openbankproject.com or email contact@openbankproject.com



KEY FINDINGS

2015 is seeing a significant cultural shift within banks and a greater readiness to utilize APIs and related technologies to assist banks meet current challenges in transforming to a digital landscape



Banks are concentrating on internal (private) APIs at present, with some recognition that once their architecture is in place, they will move towards opening partner and public APIs.

Internal APIs take two forms: an API to enable functionality for a bank's customerfacing mobile app product, and APIs to enable access to internal services as part of reorienting of legacy architecture.

Banks are at various stages of reorienting their IT architecture to enable APIs. This is being done in three main ways:

- Some banks have a build/replace program where SOAP services are being swapped for REST APIs
- Some banks are adding an abstraction layer (a REST interface) on top of their SOAP services
- Some banks are adding an API gateway to their ESB/pointtopoint communications architecture.

Overwhelmingly, the majority of banks interviewed are focused on getting internal APIs right for:

- Payments
- Customer information
- Customer transaction history

In API design, definition formats are being increasingly used, but two in five banks have yet to choose which definition format they will use as a standard within the bank. Swagger is the most popular at present, followed by RAML, with the remaining equally split between custom solutions, API blueprint, and using templates that come with their API management solution.

Containers are being adopted quickly across the banking industry, with most banks already implementing pilot projects, and some having fully integrated container technology into their production architecture.

**THE MAJORITY OF BANKS
THINK CONTAINERS
WILL BE A PART OF THEIR
PRODUCTION APPLICATION
ARCHITECTURE WITHIN
THE NEXT THREE YEARS**

Security remains a difficult hurdle for many banks. The majority are looking at multifactor protocols, with OAuth2 and enhancements to two-factor authentication being commonly cited as part of the security layers banks are using. Others have added proxies, API keys for each device, tokens, and SSO.



Banks are increasingly turning to API management solutions to help them manage their API architecture. Some have chosen to build their own solutions, but several of these have acknowledged that they have now grown sufficiently where an external provider may be warranted. The ability to manage security is the greatest concern when choosing an API management provider. Reliability is important: enterprise customers want to buy enterprise level offerings from established players.

The study found that amongst those interviewed, there has been a major shift this year in the acceptance of an API strategy amongst Executive level decision makers within a bank.

INCREASINGLY, BANK LEADERSHIP ARE SEEING THAT APIs CAN PLAY AN INTEGRAL ROLE IN SUPPORTING THEM TO QUICKLY PROVIDE CUSTOMERS WITH THE SERVICES AND PRODUCTS NEEDED

An API strategy within a bank is being defined in terms of helping a bank remain customer focused.

The bank leaders we spoke with have identified an open banking platform as a future goal. They see this as a particular opportunity in helping

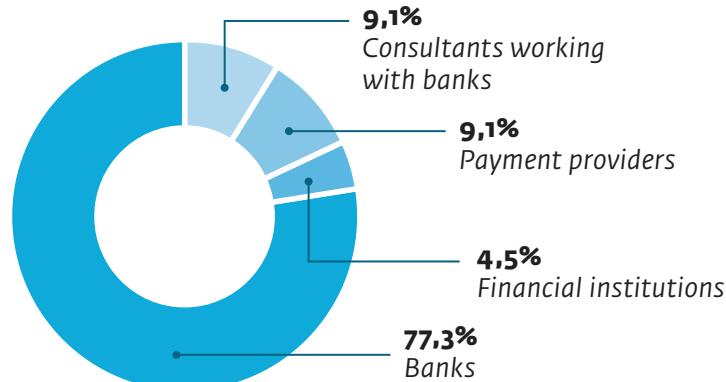
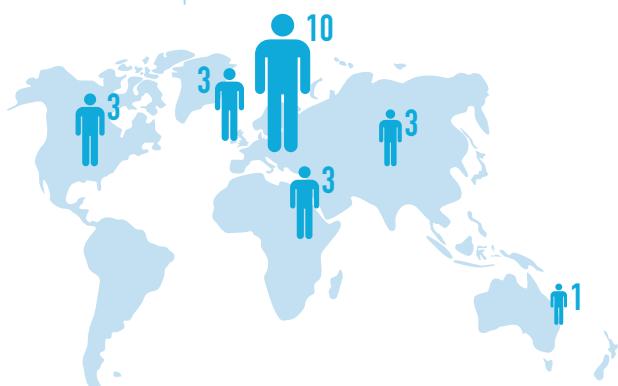
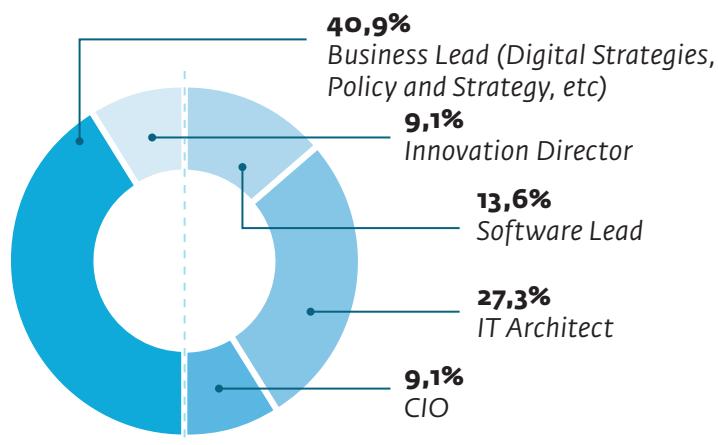
meet emerging regulatory requirements including the suggested UK Open Banking API standards and, more importantly, the second iteration of the European Union's Payment Services Directive (PSD2). Banks are seeing this as an opportunity to become a platform for collaboration with fintech third party providers, and while the bank respondents we spoke to are keen on this future horizon, they acknowledged that the decision to open a bank's platform has yet to be won internally and may take several years to convince management of the benefits.

Going forward, API leaders within banks need further resources and tools to help them build internal support for an API strategy. As banks complete the process of reorientation towards a REST-based architecture, new API-enabled products will need to be developed to demonstrate the value of APIs. These pilot products will need to be measured in terms of demonstrating increased customer engagement, and faster development and timetomarket cycles. The use of APIs to improve internal processes will need to be documented and shared as success stories internally to encourage further buy-in.

A range of specific API management services for the banking industry, including enhanced security and data protection features, the ability to test APIs prior to production, internal developer feedback mechanisms, and the capacity to measure engagement will help banks advance their API agenda faster.

ABOUT THE STUDY

The Banking APIs: State of the Market study was conducted between July 15 and September 7, 2015. 22 respondents with leadership roles in API strategy and management were interviewed in an hourlong interview. Respondents then also completed a short data questionnaire.



WHO WE INTERVIEWED

Participants in our survey were evenly split between business roles (including Innovation Leads, and Business Leads such as those with a Digital Strategy title) and technical roles (CIO, IT architects and Software Leads).

WHERE WE INTERVIEWED

The majority (45%) of those interviewed were working in banks in Europe. The remaining interviewees were fairly evenly split between the UK, U.S., the Middle East, and Eastern Europe with one respondent from the Asia Pacific region.

ENTERPRISES WE INTERVIEWED

The majority of those interviewed (77.3%) worked within tier-one banks, some with both retail and commercial arms and others focused on retail banking. The remaining respondents worked as consultants, with payment providers or other financial institutions, such as credit unions.

INTRODUCTION

THE BANKING INDUSTRY IN 2015

2015 is seeing a significant cultural shift within banks and a greater readiness to utilize APIs and related technologies to assist banks meet current challenges in transforming to a digital landscape

The key six trends signaling this cultural shift are:

- ① **Changing cultural norms** means banks are more ready than ever to implement an API strategy.
- ② **Greater workforce readiness** amongst some of the largest banks and financial enterprises.
- ③ **Emerging leadership from smaller players** that are demonstrating to larger banks how to utilize APIs.
- ④ **Increasing startup competition** is driving the need for agility amongst larger, more traditional enterprise players.
- ⑤ **A changing regulatory context** is forcing all banks to consider how to modernize and manage new compliance requirements.
- ⑥ **Growing data capabilities** means that banks and fintech are thinking of new ideas, products and solutions.



1. CHANGING CULTURAL NORMS

For at least three years now, leading industry analysts have been arguing for banks to seize a digital transformation agenda. For example, in her work with Capgemini and the MIT Center for Digital Business, Claire Calmejane (Director of Innovation at Lloyds Banking Group) looked at longterm strategies and new banking functionalities that had the potential for a quickwin adoption path.

KEY TRENDS IN DIGITAL BANKING IN 2012

SOURCE: QUICK-WIN ADOPTION STRATEGIES FROM CLAIRE CALMEJANE'S 2012 RESEARCH INTO DIGITAL BANKING

There are long-term strategies and new banking functionalities with a quick-win adoption path:

1. Customer centricity
2. Mobile banking
3. Smart data
4. Conversation / Social media
5. Core system flexibility

Quick wins:
P2P, PFM, Multiscreen, ID&V

Many of these ideas (customer centricity, mobile banking, smart data, social media conversations, and core system flexibility) rely on APIs. While most banks have made some headway with mobile apps, using APIfirst design principles to create ongoing mobile experiences is in its infancy. In fact, it is only since the start of 2015 that many banks are truly starting to consider APIs as the key enabler that allows a modernization and transformation in banking to occur.

Banks like ING are leading the cultural shift and showing that when it comes from the Executive level, it can at least open up new opportunities. (In many cases, IT Architects and Divisional Business Managers still have the hard task of convincing the midlevel that APIs are the answer, but having Executive leadership is definitely progressing the conversation faster.) In an interview in the *Financial Times*, ING CEO Ralph Hamers reframed the whole banking industry as being inherently a tech industry. He told FT's Banking Editor Martin Arnold:

We all have to recognize that technology is what banks do...Banking is immaterial. Our products are immaterial. I give you a mortgage and it is important to you, but it is nothing more than a digital number because you don't see the money or even the paper any more.

This sort of mindset has made the transition easier for banks like ING who have actually grown their customer numbers at a time when other banks are shrinking.

One of their latest flagship initiatives is their forthcoming InsideBusiness product. Aimed at commercial customers, this website and app will draw in multiple datapoints initially from within the bank's own data sources but could conceivably be mined with external data sets as well. "InsideBusiness will bring together all our channels, products and services on one platform with a single signon and a single contract for our clients to truly create a 'One Bank' experience. This will allow customers greater access in real time to the critical information that affects their companies' financial positions and empower clients to conduct more transactions themselves," explains Steven van Rijswijk, ING's global head of Client Coverage.

Providing a single source of truth and single platform for bank account customers and businesses is one of the key product drivers that is leading banks to investigate APIs. Many are hopeful that APIs will give them a way to integrate various data sources, and create new customer-centric products with a faster time to market.

2. GREATER WORKFORCE READINESS

The recent launch of Hygieia , an open source DevOps dashboard, by U.S. banking leader Capital One is a key example of the banking sector's developer and leadership workforce being more ready than ever to implement an API strategy. Hygieia is a similar idea to ING's InsideBusiness, but with the datapoints all being key metrics for an IT development-to-production workflow.

"We at Capital One are committed to continuous delivery and continuous integration," says Tapabrata 'Topo' Pal (Director, Enterprise Arch, Cloud Computing, Capital One).

Pal's goal in leading the development of Hygieia was to "shorten the feedback loop and amplify it". (It is a similar goal many enterprises have when using APIs overall: the idea is that by integrating various datasets and banking functionalities via API, enterprises can speed up development and more quickly identify when customers and partners experience problems working with banking data and services.)

Pal says that initially the Capital One team looked at open source and commercial tools and could not find one that could collate performance metrics across the various tools they use in their full stack development pipeline.

BELOW: DATA SOURCED VIA A VARIETY OF APIS IS INTEGRATED INTO THE HYGIEIA DASHBOARD (VISUAL FROM THE CAPITAL ONE BLOG)



Building the dashboard themselves, the open source project makes use of APIs from dev tools like Jira, GitHub, Hudson, Jenkins, Sonar and others. The project uses APIs to plug data from those primary source tools into a realtime visualization display. Pal explains:

The overall architecture is a MongoDB database. We created a generic data model that doesn't tie us to any tools. We make a wrapper around a particular tools' API and then we store the data from it in MongoDB in generic JSON.

Out of respect for the open source community that has made so much of their enterprise software a possibility — and no doubt to win kudos with their own developer workforce and potential new employees — Capital One has made the whole project open source. Pal says:

So one of the cultural things that Capital One has already gone through is that the enterprise has been using open source for some time, and it is time to give back to the community. Our main focus has been agile and cloud, we have mobile apps, our thoughts are going towards innovation and how to speed up development. We encourage innovation.

WHILE THE PROJECT FOCUSES ON DEVOPS METRICS, WHAT IS AT THE HEART OF IT IS AN API-DRIVEN ARCHITECTURE

The creation of the tool means their inhouse developers had to familiarize themselves with other APIs and got a chance to see some industry best practices around building API wrappers and integration tools.

Through the project, Capital One has ensured a developer workforce that is confident with integrating with external APIs and building data flows that draw from disparate data sources: a key capability that will be required as banks seize the API advantage. Pal says:

APIs are central to our development all the way across: we have a big focus on API driven application and microservices architecture and that is the current mindset in our digital platform. We do have a managed API platform. Most of our applications are API driven, and the whole culture sits around that architecture. We believe all APIs should be publicly consumable, but not necessarily all be exposed, but if someone wants to use it that way, we want to be ready for it.

3. EMERGING LEADERSHIP FROM SMALLER PLAYERS

Smaller financial institutions are demonstrating how banks can implement API strategies and are acting as examples that banks can follow.

Loren Paulsen is Software Development Manager at Maps Credit Union in the U.S., and led a work program that saw the Oregon-based credit union recently receive the prestigious Celent Financial Services Industry Award.

"The award was a reflection of our culture and our use of APIs," says Paulsen. Paulsen says the journey started several years ago with the enterprise's core conversion to DNA from Fiserv, a banking IT hub that includes an open API allowing users to access its data and banking functionalities.

"A major decision criteria for us was the presence of an open API. It is a mindset that is not terribly common in the credit union industry," says Paulsen. (While such an option may not be available



to banks, many are seeking ways to create an abstraction layer on their legacy systems that builds an API gateway to their banking functionalities, in much the same way that DNA channels access to banking functions through its open API.)

Maps Credit Union then initiated an Idea Lab to help various departments identify new opportunities to leverage the power of APIs.

PAULSEN HAS BEEN ABLE TO LEAD SOLUTIONS THAT COUPLE INTERNAL BANKING DATA WITH EXTERNAL APIs TO CREATE SEAMLESS WORKFLOWS THAT END UP IN SIGNIFICANT ADMINISTRATIVE SAVINGS FOR THE CREDIT UNION

Paulsen describes:

Through the open architecture, we have been able to automate very easily. Drawing on multiple APIs and composing them together is becoming a reality for us.

We have these insufficient funds notices, and we were able to use the Sendgrid API to send out these notices, paperless. In another case, where we have our own text banking platform, we are using the Twilio API as the backbone of that service. Another API we use is Formstack that allows our marketing and other departments to make their own forms and we use the Formstack webhook to draw stuff back to us to use programmatically and start automatic workflows based on those form submissions.

Paulsen says that one of the advantages of this is that the credit union is not locked in to using software that it doesn't need, but can instead be nimble and gain "*a more powerful result just using the API*".

To encourage other departments to understand what is possible, Paulsen's Idea Lab team would meet with various internal stakeholders and identify tasks that required spreadsheets or email reports, or that tended to involve repetitive administration tasks. He gives an example where at the start of each day, accounts that were overdrawn would be marked with a warning flag. Before their API infrastructure was in place, someone would manually mark each of the day's accounts to be locked, based on the day's report. Now all of that is done automatically via API.

The approach reflects a key API enabler that Matt Graney, VP of Product Management at API Management Provider Axway says is at the core of how banks and other enterprise need to operate :

The organizations that will succeed in the modern business era are the ones that don't approach digital business as just a set of mobile applications, but rather the ones that use an API-first approach to build their digital platform as they secure, manage and control all aspects of new data flows and business.



4. INCREASING STARTUP COMPETITION TO BANKING TRADITIONS

In *Inc 500*'s recent article on "Why fintech is one of the most promising industries of 2015", Senior Editor Maria Aspan writes:

Long seen as a highly technical, highly regulated industry dominated by giant banks that resist disruption — other than the occasional global meltdown — finance is now riding an entrepreneurial wave.

It is an increasingly recognized perspective that is the central tenet of James Haycock's *Bye Bye Banks?* (written with Shane Richmond). Haycock, the Managing Director of Adaptive Lab, writes:

The financial services playing field has been changed irreversibly in recent years by a new generation of companies and leaders who have torn the rulebook to pieces, adopting new technology, introducing new working practices, and serving customers whose lives are increasingly orientated around their mobile phones.

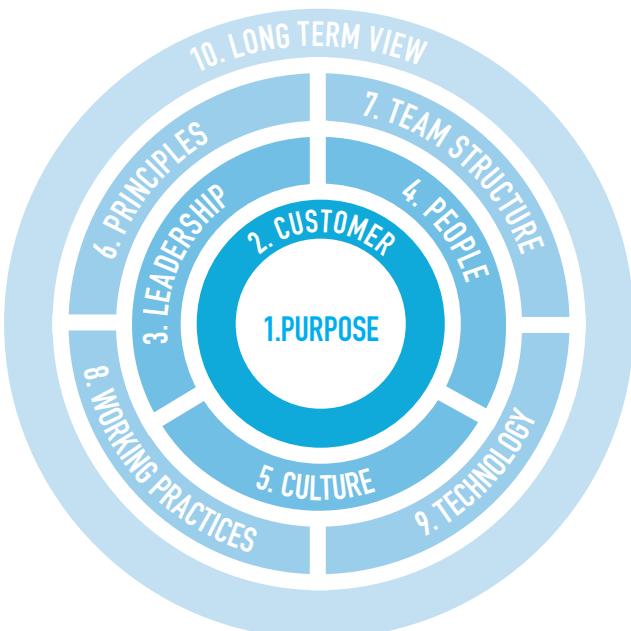
The growth in startups offering payments, consumer and business lending, investment decisionmaking, money transfers, currency conversions, and a plethora of auxiliary fintech data services has added pressure on banks to innovate as well as expanded the finance industry landscape with a whole new generation of agile product and service providers.

Haycock says one of the advantages of these fintech startups is that they have built an API architecture from the start, in tandem with a customercentric focus overall. Banks are tied to legacy systems that slow down their agility to adapt to new customer needs and market opportunities.

One of the solutions Haycock and his Adaptive Lab team advocates, then, is a Beta Bank model built on ten principles of agility and lean management. Clearly, the principle to "design technology and data that enables agility" speaks to the idea of an APIfirst approach in order to create new bank products that can reach the market quickly, and respond to customer demands for realtime insights into all of their financial assets in the one place.

Some leading banks wanting to compete against these new players have already begun to implement a Beta Bank vision internally. The idea is evident in new intrapreneurial models being trialled within larger banks, to prevent them from being weighed down by their own legacies, and to cultivate the capacity to implement more lean, agile models that draw on customer feedback and product iteration.

Innovation hubs like BBVA's Open4U and BNP Paribas' L'Atelier Lab have been specifically created within those existing banking institutions to allow banks to compete more effectively (and, at times, collaborate more openly) with the strengthening fintech sector.



ABOVE: BETA BANK OPERATING MODEL FROM *BYE BYE BANKS?* BY JAMES HAYCOCK WITH SHANE RICHMOND

5. A CHANGING REGULATORY CONTEXT

APIs are also expected to help banks and fintech companies to meet new regulatory requirements in Europe, the UK and around the world.

Already, many banks in Europe are looking at the potential impact of the European Commission's Directive on Payment Services (PSD2), and in particular the Access to Account (XS2A) provision. The existing PSD is the legal context for an EUwide single market for payments, and PSD 2 is set to be voted on by the European Parliament in the first week of October.

The new Directive will introduce some compliance requirements for banks and other financial institutions, including the enforcement of new security requirements and interoperability standards aimed at reducing barriers to entry for nonbank card and internet payment providers.

The PSD2 XS2A proposition looks set to enforce the right of third parties to be able to access bank account providers in order to provide payment and other financial services.

Innopay consultants Mounaim Cortet and Douwe Lycklama recently explained the potential impact:

PSD2 XS2A can be considered an accelerator for technology driven disruption of incumbent banks by flexible and innovative service providers that target not only the payments value chain, but every single 'piece' of the universal banking model. These innovative players threaten to capture revenues long taken for granted by incumbents. This development of digital transformation will disrupt the complete banking sector as we know it today and will require incumbents to adapt their business and operating model.

Meanwhile in the UK, plans for a mandatory open banking API standard were included in this year's March National Government budget, after having been introduced into the UK Treasury's Autumn Statement last year. The move will require a level of API standardization so that third party providers like application developers can securely access account data from any UK bank (with the customer's permission).

Banks will need to be able to have the infrastructure in place to implement such a standard, but will also have to innovate and act quickly to provide competitive products. The race will be on when other banks — and other fintech players — start using the data flow that such a standard could make available to everyone. There are also opportunities for banks to partner with third party providers to create new business models based on collaboration and that combine bank's traditional strengths with fintech startup's agility in product development.

6. GROWING DATA CAPABILITIES

Amongst all of these pressures and changing landscape, banks and fintech startups are discovering yet another key truth: data is currency.

Banks sit on a wealth of data that can be aggregated to provide crucial insights into finance markets, supply chains, and consumer behavior. Transactional data at an aggregated level can help logistics and warehouses optimize their distribution and supply chains and reduce waste; can generate insights into population flows and travel patterns for cities, transport and travel companies; support retailers to





manage stock and push sales timed to consumer demand; and assist lenders and insurers to better identify and calculate potential risks.

Banks are beginning to see how APIs can help them to draw this data out of their own systems, clean and analyze it, mix it with external sources, and build analytical tools for internal use and as new business-facing products.

Chris Haddad, VP of Technology Evangelism at WSO2 describes the new business model opportunity for banks in a white paper on an API management technical evaluation framework:

Instead of delivering static, monolithic products and services, APIs create an opportunity to whitelabel, embed, and monetize business capabilities. Business teams are offering content brokers and information hubs that share data aggregated from millions of devices and billions of customer interactions. The aggregate dataset can help identify trends, structural market shifts, and economic patterns.

Adrian Hausser, from global consultancy firm PayX works with top tier banks and payment businesses to help reorient major institutions to the new global financial industry marketplace. He sees APIs as being the key technology that can help banks position themselves, and that will allow new fintech players to enter the market. Speaking at CA Technologies' 360 Summit late last year, he explained that APIs are the enabler that helps financial institutions create value in their relationships with customers. Opening up transactional data is a key example of the sort of new value that Hausser says can be created. It is a model that is already being experimented with by some of the world's leading banks.

2015 AND BEYOND

This report looks at how these six trends are playing out this year in how banks are reorienting towards an API-enabled future. The report is based off confidential interviews with 22 banking experts from around the globe and discusses the current API landscape within banks, where banks want to head, and what tools and strategies they think will help them get there.

OVERALL, ONE OF THE MOST INTERESTING FINDINGS OF THE STUDY WAS THE LACK OF WILLINGNESS TO PREDICT WHERE APIs ARE HEADING BEYOND THE NEXT YEAR

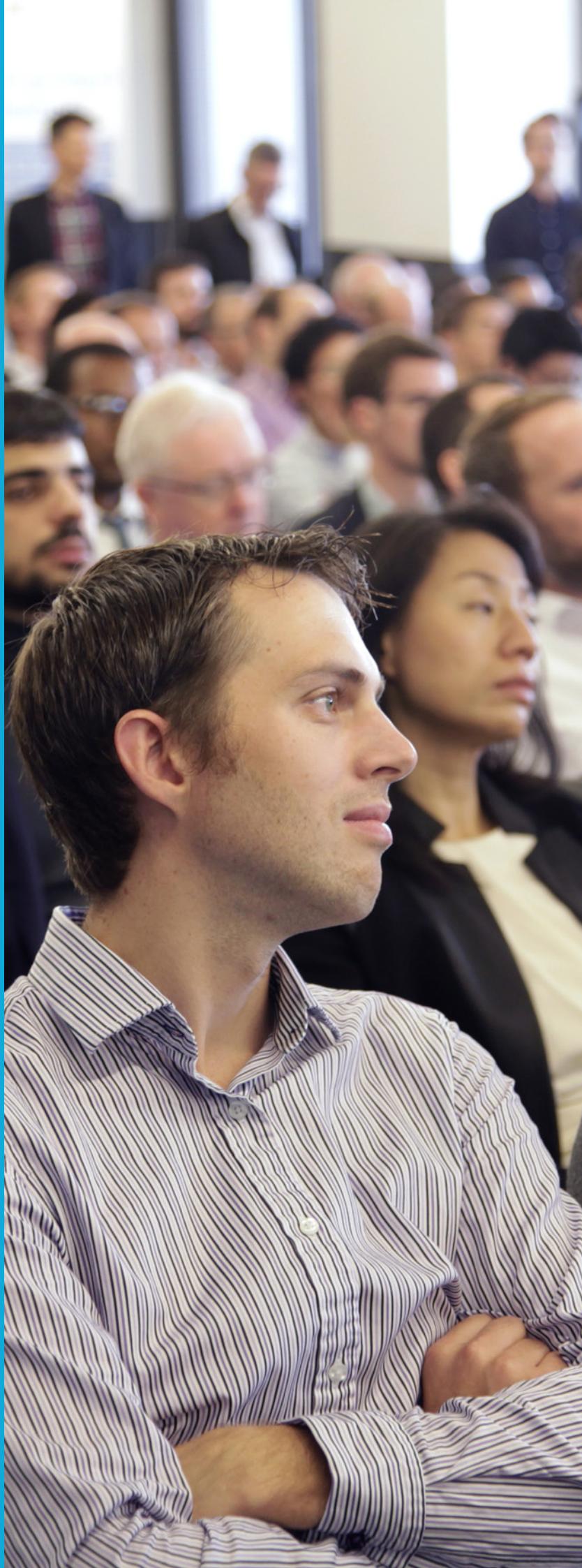
The leaders we spoke to have worked hard over the past several years to build internal support for an API strategy and are now seeing a cultural change from the Executive level that is now committed to implementing an API agenda for their bank. As such, the leaders we spoke to are now in the process of building midlevel management support and in laying the foundational architecture that will make it possible to take advantage of the agility, customer opportunity, partnership reach, new business, and efficiencies that APIs promise.

01

THE BANKING API LIFECYCLE

KEY FINDINGS

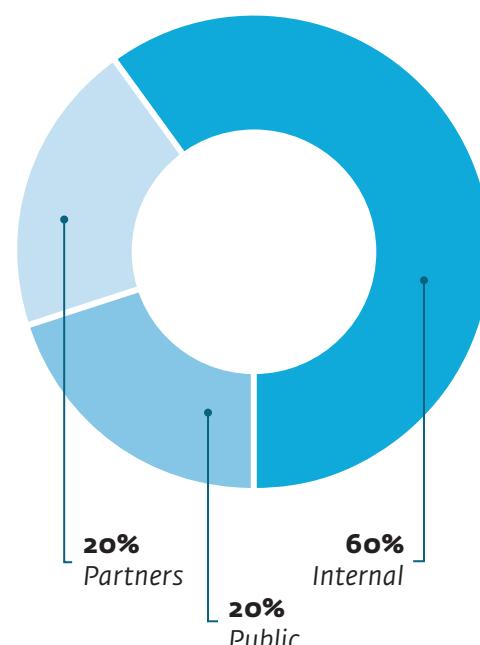
- ✓ For the majority of the banks interviewed, internal APIs are the key focus for the next six to twelve months.
- ✓ Internal APIs are being used to enable a bank's mobile application, and to enable access to bank functionalities in a legacy infrastructure



Almost all banks interviewed are concentrating on internal (private) APIs at present, with some recognition that once their architecture is in place, they will move towards opening partner and public APIs. This matches similar research conducted by the Open Bank Project and *Bank Innovation*, which found that 88% of banks surveyed think internal APIs are essential for regulation and compliance, backoffice systems management, and for leveraging big data.

88% OF BANKS SURVEYED THINK INTERNAL APIs ARE ESSENTIAL FOR REGULATION AND COMPLIANCE, BACKOFFICE SYSTEMS MANAGEMENT, AND FOR LEVERAGING BIG DATA

Split between Internal, Partner and Public APIs



INTERNAL (PRIVATE) APIs

Internal APIs took two forms:

- An API for a mobile's customer facing mobile app.
- APIs to enable access to internal services as part of reorienting of legacy architecture.

An API to enable a bank's customer-facing mobile app

In this use case, banks had begun building a mobile application and after seeing the diversity of devices and changing mobile landscape, realized that building their mobile product on an API would allow them some flexibility moving forward:

We basically learned this during our last application. Now we make them all API-first: mobile app, the web version, and the web management console. So we avoid having to make a web application that targets the database directly and then make a separate mobile API. It is easier and simpler to just make the API first, it makes deployment simpler.

APIs to enable access to internal services

In most cases, APIs were being created internally in order to reduce complexity and overcome legacy systems that had previously relied on point-to-point integration.

We started creating internal but global APIs. We try to work with internal developers as if they are external developers. So define a workflow: API definition, implementation, quality, and operation. We are working with a big architect and engineering team at the core of the bank to implement the workflow. We have a best practice document, a style guide for defining APIs, and we are now using RAML for definition.



PARTNER AND EXTERNAL APIs

Overall, the majority of banks were not yet using APIs to share data or functionalities with external partners.

A very small number of banks had opened up some APIs to partners to enable some third party service delivery. One example use case mentioned was to provide a bank's commercial banking data to a stock market pricewatch service.

Some banks however, are already being asked for partner APIs and can see a benefit in reaching customers where the customer wants to be, and not forcing customers to come to a bank's physical or virtual door:

👤 People don't need banks, they need bankers: we see that they need the services but they don't want to have to visit the website anymore. So we need to integrate all services into the first interaction of the customer. for example, we need to offer payments via credit card service. Or if you go to a funds institution, you need to be able to buy funds directly from there. It is the same thing with subscriptions. If you subscribe to a newspaper specializing in the stock market, you want your customers to be able to become a client while still on the newspaper information website. Customers don't want to have to switch interfaces. When we trace it, we have a lower transformation rate if we do not provide an API to these partners.

A number of banks are fostering fintech growth by supporting incubator and accelerator programs aimed at building fintech startup partners. While not explicit at present, these startups may be encouraged to use partner APIs provided by the bank in future. In some

cases, hackathons with access to a bank's transactional data are used to recruit startups to these programs.

As described in more detail in *Section 3: The Banking Stack*, bank leaders in charge of designing APIs are doing so in a way that will make it possible to open them to third party providers in the future. Several respondents noted they are also designing their internal APIs with a view to opening them to prevetted partners in future.

Partner APIs: Consuming Third Party APIs

There is some consumption within a bank of external APIs. Most commonly used external APIs were:

- Maps
- Payment systems and providers
- External transactional big data
- Exchange rates and currency converter APIs
- Email APIs
- GitHub API

Some IT teams were meeting with business units and as part of training and information around the value of APIs to the organization, would help teams map workflows and processes that could be automated with APIs. Other innovation departments had established internal crowdsourcing ideas platforms where staff could share details of manual processes that they felt could be automated.



PUBLIC APIs

Workflow processes identified through these consultations were then used to prioritize internal API development, but also to identify external APIs that could help reduce data entry or manual workloads. For example, SendGrid and Twilio were named as two APIs that could help automate internal processes such as informing customers when they were overdrawn or to confirm large credit card purchases.

The most common use case for external APIs was where banks are using mapping APIs on their websites to show customers where branches and ATMs are located.

The second most common use case is payment provider APIs. One bank mentioned using PayPal's API specifically, while others indicating they are consuming external payment system APIs.

In a small number of cases, banks are using external data sources via API to enhance their own data analysis. This was the case in calculating mortgage loans by using real estate data via API, and in another case, external data was harnessed via API to help calculate loan risks.

While not prescribing what external APIs could be used, IT architects and API lead strategists were looking to add external APIs to their internal developer portal to help teams identify available external services.

Using external APIs also influence internal API design. Several of the banks and financial institutions that were using external APIs noted that doing so had influenced their own API design, particularly encouraging them to use API definition languages.

Overall, banks are not yet ready to open up their data or services externally via API.

In a small number of cases, public APIs are being made available to provide static information to external developers via API. This is data that traditionally would be available from a bank's website: branch and ATM locations, descriptions of bank products, and foreign exchange data.

IN A SMALL NUMBER OF CASES, PUBLIC APIs ARE BEING MADE AVAILABLE TO PROVIDE STATIC INFORMATION TO EXTERNAL DEVELOPERS VIA API

In all but one other use case, public APIs were created for shortterm hackathons and other developer evangelist projects. In these cases, bank data would be placed in a sandbox or replicated in anonymized form. Then the public APIs made available would make calls to this mock data, so there was no connection between these public API endpoints and any live bank data. The public APIs would then be decommissioned after the hackathon or developer competition was held.

02

THE BANKING STACK

KEY FINDINGS

- ✓ Banks were either building new REST APIs and replacing SOAP services as they go.
- ✓ Creating REST interface on top of their SOAP services.
- ✓ Reorienting legacy systems to enable access to services via an API gateway or ESB message bus.
- ✓ Half the banks were piloting containers, with the majority of banks interested in the technology.



Banks are facing significant hurdles in adjusting their IT infrastructure to be able to manage APIs.

Amongst those banks who have generated internal buyin for an API strategy, the following architecture reorientation approach is most typical:

1. CATALOG INTERNAL SERVICES

- Internal services available via SOAP interface
- Legacy system with message bus and ESB
- Point-to-point services

2. BUILD ABSTRACTION LAYER

- Apply API gateway
- Message ESB interface
- REST interface

3. BUILD REST SERVICES

- Built on top of SOA
- API gateway
- New build REST services

While this was the most common strategy for orienting legacy architecture to prepare for an API strategy, there were still three overall approaches within this orientation:

- Those who are seeking to build/ replace SOAP services
- Those who are adding an abstraction layer on top of SOAP
- Those who are adding an API gateway to their ESB/pointtopoint communications architecture.

Bank leaders interviewed who are now ready to progress their API strategy aggressively have already transformed their internal services to a SOAPbased architecture with RESTful interfaces on top of that. Going forward, these banks plan to build REST APIs directly, but apart from one or two leadership outliers who have spent the last several years rebuilding their infrastructure from the ground up, others have no plans at present to tear down their legacy systems and start again with a complete REST API framework:

 *We are creating APIs on top of our SOA architecture, and we eat our own dog food: we only offer our APIs for us internally. Because we know that our partners are our new type of customers, if we retain our SOA services, they are very technical, we don't think they will serve the needs of other companies. So we need to put them out as an API product to the market.*

 *The idea is not to have too many SOAP services. It is not the target so when we have something new to build, we do it through a REST interface. If we build new applications, we will expose most of the SOAP services through a REST API, but we don't have any big plans to put everything in a trashcan. But we do hide the existing SOAP services behind REST APIs, and for new projects, we build new REST services.*

 *We didn't want to just plop a REST interface on top of our SOAP services. Tactically, that may have happened a number of times, but the principle of it was you are rethinking what you are doing with these services. So it wasn't a straight translation of SOAP to REST, but it was a full replacement of our SOAP services with new and different REST services. So there were zero REST APIs two years ago, and now we are beyond feature parity at this point.*

The number of internal services defined by banks ranged from 5 to thousands, with the majority estimating between 200-500 services. Of these, APIs are attached to about 2% of the internal services, although a couple of API leaders had completed their internal reorientation and had reached 100% of all internal services having API endpoints.

While some respondents had achieved 100% REST APIs, the majority were working with a split of around 75% SOAP services and 25% REST APIs.

Costs vs Investment for an New IT Architecture

For business and technical leads responsible for reorienting a bank's architecture, costs remain one of the key obstacles for building out an APIenabled architecture.

Data Sharing and Open Data for Banks: A Report for HM Treasury and Cabinet Office by Fingleton Associates and Open Data Institute in the UK found that the majority of estimates for moving to an APIbased infrastructure "were consistently below £1m, and tended to cluster in the lowtomid hundreds of thousands." Fingleton Associates'

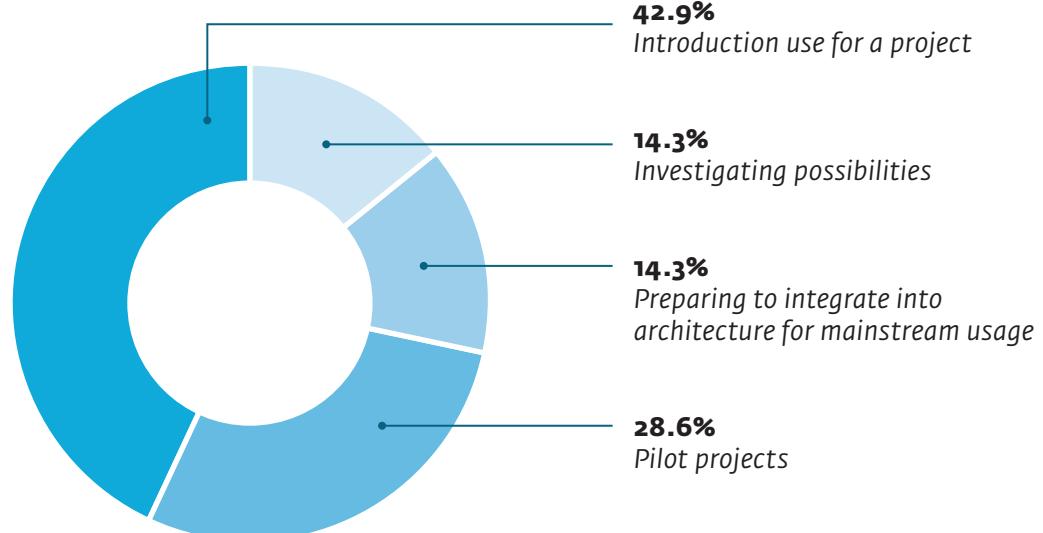
report notes that third party suppliers like Open Bank Project and digital consultancies with experience building API-based mobile apps estimated costs to be between £300-400,000. Banks interviewed generally estimated costs to be much higher:

Only one bank gave us a number, and this was couched as a 'rough guesstimate'. They told us that it would probably take 'tens of millions of pounds' for full implementation of third party API access, but that most of this cost would be compliance and legal rather than technical.

Another difficulty for many banks is the technical debt their current systems have on their IT budgets. A report by global financial consulting service, Celent , analyzes global spending trends of banks. While global banking IT expenditure is expected to grow 4.6% to reach US\$196.7 billion this year, 76% of this budget is spent on maintaining legacy systems.

Overall, reorienting a bank's legacy system was considered costly, and amongst those who had not started a reorientation process, it appeared that the fear of exorbitant costs was one of the key factors that led to a paralysis of action. Amongst interviewees, the banks that had shown the greatest progress in

Container technology in banks: Current trends



building out their API architecture were also most advanced in harnessing the benefits of container technology to implement a microservices approach.

Microservices is seen within banks as part of the overall reorientation towards an API infrastructure: this year more than ever before, IT architects are finding support for reorienting legacy systems towards defining bank functionalities and datasets into composable, independent components that can then be orchestrated via APIs into new products or delivered in new workflow processes in order to improve efficiency, reduce costs and speedup product development or analytical insight.

Several respondents recognized the benefits of a container approach:

- Isolation of services
- Optimized resource management: limiting CPU and memory usage at a granular level
- In containers
- Faster delivery of new software
- Less dependency on other software
- Automate with continuous integration tooling
- More productive

Others said that it would be impossible for banks to implement:

• We are an enterprise company: we have bank production systems that cannot be containerized. They have different principles of work, and we cannot change that. It is one of our limits that we cannot break.

Amongst those who have successfully introduced an API enabled architecture across their legacy systems, many were aware of container technology and had begun experimenting with its benefits. This ranged from pilot projects, some implementation in testing and deployment environments, some production usage, and full implementation of containers to manage a bank's online products:

• We have implemented containerization in pilot projects, it is not a bank standard today.

• We have webbased containers, and are building mobile containers. Containers will use features (microservices), bolted on to our existing web applications. There are challenges with the framework: one is in the mobile space, you will never get the best performance in usability in containers vs native, but we are working through that. The upside is that you can develop it and develop features across rather than doing them in each natively.

• We are already using containers right now for our newer products. In the last year, we have had about 12 or so internally.

The biggest challenge in using container technology was the uncertainty around security strength. Within the container ecosystem, banks were reticent to use Docker Hub (the public Docker registry) as they felt this was not secure enough for their needs. Those banks experimenting with containers were using their own private registry to build and maintain images.

Other containerrelated tools banks are using in conjunction with their containers includes:

- Jenkins CI
- Ansible
- Jfrog
- Kubernetes
- Mesosphere
- ElasticSearch
- Consul

However, in the main, the responsibilities for introducing containers into a bank's IT architecture was being done by a separate team within a bank's IT department and this was not always closely aligned with the API agenda: those steering API IT strategies were often aware of container work within their enterprise, but not directly involved.

03

BANKING API DESIGN

KEY FINDINGS

- ✓ In Europe and the UK, the PSD2 initiative was encouraging banks to take action and invest in an API orientation strategy.
- ✓ The majority of banks were moving towards a Swagger or RAML API description language, however, most were yet to start using this as a primary tool in their API design.
- ✓ Hypermedia APIs are being used by some banks but in the main hypermedia has been rejected as adding greater complexity.



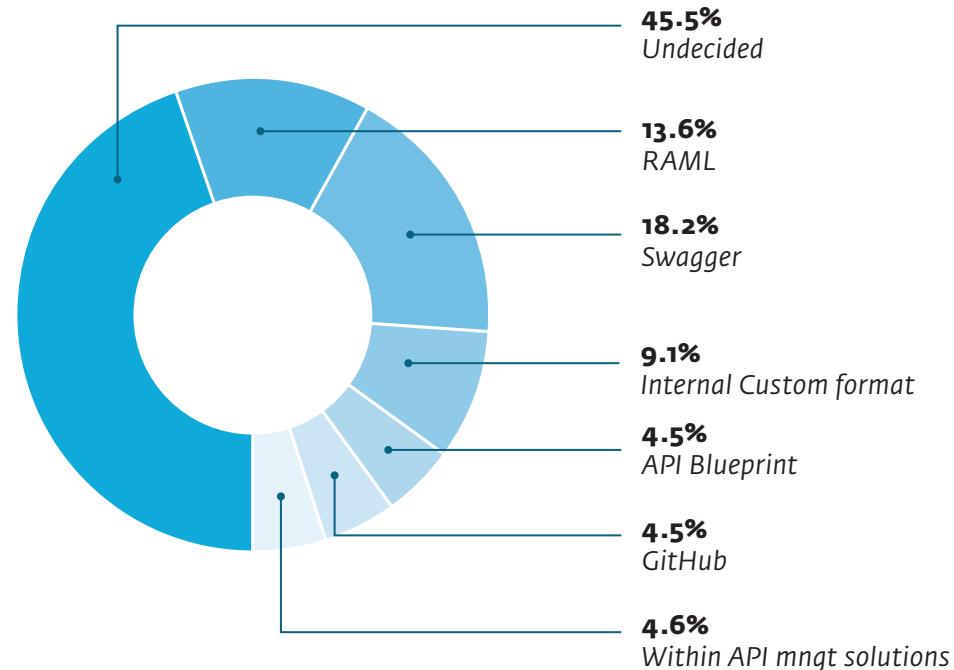
The majority of banks interviewed were adopting an APIfirst approach to designing new APIs. As outlined by Accenture, an industrial API maturity model recognizes that in the enterprise, it is not uncommon for APIs to be initially created for individual use cases. As the benefits of an API approach are then identified within business units and shared across the company, IT architects and business innovation leads tend to be involved in order to identify service components and data sources that can be bundled together and made available via API.

In some instances, banks are then thinking in terms of complementary services, for example, all of the functionalities that are related to a customer transferring money may be bundled together as an API.

An emerging dominant mindset amongst those interviewed is the idea of envisioning the API as an eventual product: it may start internally but more often than not, they are now being designed with a view to potentially being opened up to partners or third party providers later on. API design is considering the need to have endpoints and functionalities so that the one API can stand alone and not be dependent on other information or functionality, and not tightly coupled with any one channel (such as mobile, web, or on a teller's machine) so that it can be utilized by third parties.

 Every time we build an API, we think in terms of it being a public API, so we build the necessary authorization mechanisms.

Current or planed API definition language to be used by a bank



Those with a more advanced API agenda are then using API definition languages (Swagger or RAML in particular), description tools (Google Discovery docs or even just a internal word template) to map out the API design, or the descriptive tools embedded in their API management solution. Almost half (45.5%) had not yet decided.

Once this conversation on creating an API with a future view to it being a product occurs, API design becomes a crossdepartmental process involving a number of decisionmakers.

• An API may be reused in other products as well, if this product needs that API. For example, if we want to build other products in future, maybe this product needs customer information and this product may use the customer profile API that we have now for this new business product.

• We work with the digital business architecture and work in conjunction with them to further the API stack. We certainly try to challenge the design so it can be used by multiple applications. The idea is that different applications can get their data by filtering the data in the API.

• In the last 3-4 years, we have been working with agile methodologies. So we have digital banking working with the product owner from the business department, and the business analyst with the engineering dept, all working to define the API product.

• We are reinventing our whole product development way of working. We are transforming to agile, usercentric innovation. We really connect to our users and really engage with them and see if it works with them, in a lean way.

Some banks were having success after reorienting their teams so that business and engineering worked sidebyside:

• It is now easier to develop, every squad is able to develop APIs.

But a small number are still pushing back against such an approach:

• Creating an API when you develop a product is a more lean way for development that you start to do. You must solve business problems, not build some API layer for all immediate use cases. Many different projects have different needs, and have different requirements and different data. If you build an API immediately for everything, you don't know what requirements will be needed in projects in future, so you spend your time maybe developing for things that never happen. It is not a good way for software development. But when you build an API, you should think about future usage of this and do things that could make it reusable in the future, if you can predict that it may be useful in future products.

REGULATIONS

By and large, regulations were not greatly influential in setting a bank's API design practices.

At least, not any more so than regulations and compliance are taken into account in any case:

• We constantly deal with regulatory requirements, especially for external APIs but there is nothing in the legislation that will accelerate or stop our strategy right now. For a bank, compliance is always the first and last thing you deal with.

As can be expected from jurisdictional legislation, the impact of regulations on API design is often set by location. In the U.S., banks mostly spoke of being required to use multifactor security in the management of personal and financial data. In the UK, the potential introduction of an Open API Banking Standard began to influence discussions around API design late last year and in the lead up to this year's election, but for the most





part it appears that has now given way to a more expansive discussion regarding alignment with the European Commission's PSD2 directive.

At present, PSD2 is driving policy and business decisions within banks, and it has begun to influence some architectural decisions within European and UK banks. While some banks are using the PSD2 to leverage discussion around the role banks will play in a more competitive, open environment (*see the discussion on internal policy in Section 6 below*), others have begun to take a somewhat technical approach and are reorienting their architecture to be ready for implementation.

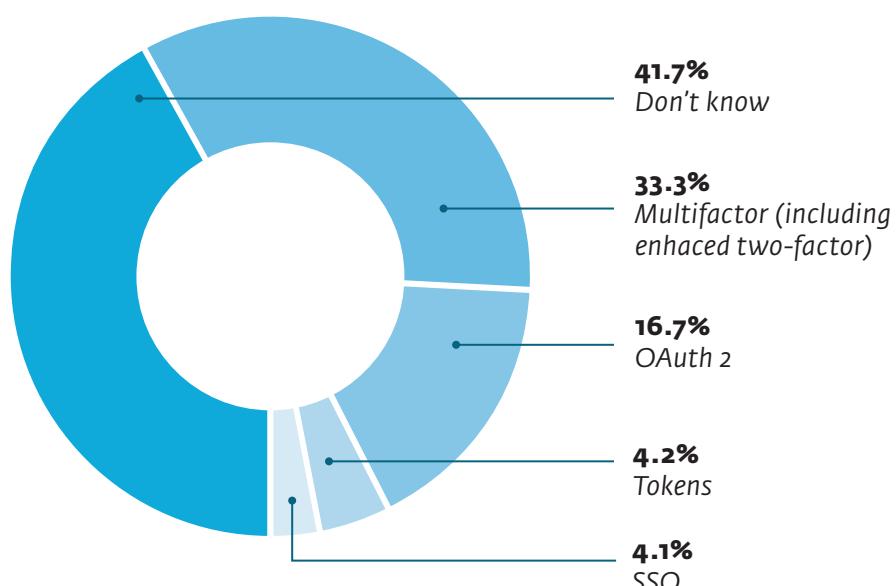
With new regulations coming into force, we are seeing a huge driver in European banking for being able to describe APIs. Banks across Continental Europe are very serious about how they are going to design and create APIs. There is also a huge push by multinational banks. This is not just being driven by regulation, but also by wanting to have really good APIs.

Banks are also looking for some greater clarity as to what the architectural requirements the PSD2 will require:

The regulator needs to move on a few areas as well: we need clarity on some of the legal issues. There is still a liability question mark: who is liable if we open up our data and a third party provider gets hacked? To get complete buyin, the legal framework needs a bit of work.

Amongst European and UK banks, the PSD2 is already having some influence, although more often on overall business direction than API design. Some specific country regulations have a much greater influence. For example, Belgium has regulatory requirements that require banks to support two-factor authentication whenever a customer logs into their account (a one-time password needs to be sent and validated by a customer after they have logged in with their user code and PIN), while Germany has open API regulatory standards.

SECURITY PROTOCOLS AND MANAGEMENT



Overall security considerations are being given greater weight by the majority of banks interviewed. For the most part, banks are seeking to implement hybrid, multilevel security approaches that have multiple checks throughout the routing of data and the provision of services. This includes:

- OAuth2 being seen as a key authentication model, particularly in preparation for
- when banks open up APIs to their parties
- HTTPS encryption for APIs
- Endpoint provisioning
- API keys being issued for each process

One of the most challenging areas facing banks is twofactor authentication. For the most part, the banks interviewed feel there are too many weaknesses in relying on twofactor authentication as it is now set up and are introducing a number of their own models to fortify access.

"These providers are working through a small number of partners: it is a massive volume of data but a small number of partners integrating with it. Now this is expanding. They want to bring in more people and will need classic onboarding. SSO and OAuth2 is hugely important to these analytics and data providers, they are working their way through all these issues at the moment. At present, they have very rudimentary schemes and basic authentication, but this is not scaling for them, and they are mostly exploring the SSO route from what I have seen. There is a big move towards two-factor authentication in the finance world."

Two in five (41.7%) respondents reported that decisions regarding security protocols were an ongoing discussion, and choice of security protocols had not yet been decided. Only a very small number of respondents discussed interest in biometrics as a form of security for customers access to their banking data or to bank products and services.

HYPERMEDIA

For the most part, hypermedia APIs are either not on the roadmap, or have been considered and rejected.

One comment from a respondent sums it up best:

-  *Generally there are two reactions: First, is 'hey, this is really neat', and then secondly there is 'hey, this is really chatty'.*

Some believe that hypermedia will be a necessity for the banking industry. With so much data flowing through APIs, the ability for an API to evolve and "follow the links" rather than have everything hardcoded in documentation is seen as a necessity facing the banking APIs of the future. Those who believe this is the road ahead have started to make some progress with implementing hypermedia APIs across their internal catalog. In some cases, hypermedia APIs can make up to 50% of a bank's internal catalog.

Others have tried to apply hypermedia APIs in their work, but have found that it creates more complexity than it solves:

-  *It's not a priority. It is not something that will move the needle in the near term. Applying it at an enterprise level is a nonstarter. We have too many different factions and two many different use cases to build the kind of affordances that actually make sense when clients are navigating. So you would end up constantly developing and changing the mediatypes and contracts. The way I would like to see that happen is if you start carving out individual capabilities as API products (for example, the ability to move money as a capability, that might be a set of APIs), then I would love to work with the team developing that and discussing how to implement a hypermedia approach but not mandate it across all our APIs.*

04

CURRENT BANKING API MANAGEMENT PRACTICES

KEY FINDINGS

- ☑ Banks are fairly evenly split between those using an API management solution, those who have built their own solution, and those who are in the process of selecting a service.
- ☑ The majority of survey respondents felt that current API management solutions are overly complex, have a range of features that are not needed, and that no solution currently available has been specifically designed for banking needs.

API MANAGEMENT

Amongst those banks interviewed, about one third were using an API management provider, a third had built their own API management toolkit, and a third were in the process of selecting a provider.

Banks that have an API management solution in place

The key features that banks were looking for when they chose their API management solution included:

- Tooling to enable conversion of SOAP services to REST
- Having a monetization component (although those who had wanted this in their solution were yet to make use of it at present)
- Identity access management
- Rate limiting
- The ability to work with a large enterprise (therefore were looking for established commercial products, with good maintenance and worldwide support).

ADDITIONALLY, SOME BANKS WERE MORE COMFORTABLE CHOOSING AN API MANAGEMENT PROVIDER WHO HAD A NETWORK OF LOCAL RESELLERS AND PARTNERS WITH REGIONAL EXPERIENCE USING THE PLATFORM WITH OTHER BANKING CUSTOMERS

Some larger banks were also using multiple API management providers for discrete pilot projects and evaluating the capabilities under a production environment.

Banks that have built their own API management toolkit

There were two main drivers that led some banks to decide to use their own custom solution for API management:

- Need for an on-premises solution: Some banks needed a complete on-premises API management solution where their data and all functionalities could operate from within an on-premises data center, rather than have any component of their API management in the cloud. This was a requirement based on organizationwide internal policy decisions in some cases, and based on interpretations of regulatory requirements in others. A related issue for some banks was that the security protocols in their legacy systems were overly complex and were not easily integrated into a third party API gateway providers' solution.
- Price point did not match services required: Several banks that chose to create their own API management solutions did so because they felt that product offerings were overly complex and often had too many additional features that they did not need or want to use. High costs of an API management solution were mentioned by several respondents. Some banks at the start of their API journey only had a small number of API consumers and did not see the need for a whole-of-platform solution at present.

- From what I have seen at the moment, we will stay on our custom solutions...*
- The number of features is a bit overwhelming and way more complex than what we needed.*
- We need to run it. Another problem is how our security systems are made, we have a lot of legacy systems and it is very difficult to integrate with commercial products.*

Several of those banks who had created their own API management tooling did note that the decision to do so was made two years ago and that they would be interested in reconsidering their future direction if they were confident that products had matured enough, simplified, and were more tightly targeted at the specific needs of the banking industry.

Banks in the process of selecting an API management provider

Overwhelmingly, the ability to meet a bank's security levels was the major criteria influencing banks that were currently in the process of selecting an API management solution.

 *Security is the main concern for API management. You are redefining the boundary of your organization so how do you manage it security wise? This is a new phenomenon for a bank.*

Here, banks predominantly referred to their security protocols and the need for a third party API management provider to be able to adhere to the enterprise's security framework. But additionally, several banks also referred to the need for security in that their chosen API management provider should be able to coordinate API usage as part of a disaster recovery plan. For example, data management problems are something that banks face on a biannual timeline: some respondents noted that every couple of years there is some problem with data recovery needing to be performed at a services level. Another aspect of disaster recovery needs is the ability to be able to switch suppliers automatically via API when a problem is identified with one API supplier's data or service delivery.

SECURITY AND RELIABILITY WERE IDENTIFIED AS THE TWO KEY REQUIREMENTS WHEN SELECTING AN API MANAGEMENT PROVIDER

DEVELOPER PORTALS, DOCUMENTATION AND DISCOVERY

For the most part, banks were fairly early in their developer portal maturity.

Those with smaller inhouse developer teams were relying predominantly on wordofmouth.

They may have identified a developer evangelist to work internally, but for the most part, internal API discovery was promulgated through a series of knowledge awareness and professional development activities. For example, API leadership teams or innovation leaders would hold seminars and training courses across a series of business units to promote the value of APIs, and the potential value that can be created for clients/customers.

Through this process, teams with some readiness to use APIs would be identified and assisted to integrate with the bank's internal APIs or be supported to utilize external APIs to automate parts of their workflows.

For banks with a midsize number of developers, the IT architecture team acted as the developer portal. They respond to internal developer requests where engineers were seeking an understanding of an API's potential capabilities and to request support in best practice coding when using an API internally.

For banks with larger numbers of developers and a bigger catalog of internal APIs and SOAP services, some form of API catalog was created internally to aid in discovery.

Sometimes this was done within the API management solution being used by the bank. In other cases, it was on their intranet or they created a purposebuilt catalog. Where such a catalog did exist, some IT architects were also including a suggested list of external APIs to encourage developers to make use of third party data and service providers in their product and workflow design. The idea here was not to prescribe what external APIs were prevetted or approved by the IT department, but instead to reduce the search time for some high value external APIs, and to promote the idea of using external APIs to automate processes where possible.

05

TYPES OF BANKING API

KEY FINDINGS

- ✓ Payments, customer accounts and a mobile/web app API are often the key APIs currently available or under development.
- ✓ A multi-account, 360 view of the customer's financial portfolio is seen as the most desirable API-as-a-product to be created by a bank.



Respondents shared a wide variety of potential use cases for APIs. Overall, APIs are being used internally to enable access to functionalities and data for internal uses. The majority of banks interviewed are in the process of 'getting their own house in order' first before moving to a partner or public API strategy. Increasingly, however, API leads are mapping their internal API strategy with a view to opening up their APIs to vetted partners or to third party providers in the future.

APIs being used now by banks include:

- Customer profile data (name, address, demographics, etc of customers)
- Transactional data of customers accounts
- Static data that would be otherwise scraped from a bank's website: ATM and branch locations, product offerings
- Mobile and web app API
- Functionality to modify bank card limit.

"We have several APIs that have small boundaries and business contexts that let the APIs do a small number of things, but do them well for your business tasks. For example, an API for customer profile."

The majority of banks are currently in the process of fasttracking API development over the next six months.

 *Current accounts data is the first set of APIs being looked at. Bank products are very complex: it is not like a bus timetable API. These are products that have a lot of added value benefits and trying to code that in a uniform way is a huge challenge.*

 *Our priority APIs for development are current accounts, and daily transactions. They are the most common needs. Most of the API solutions we want to build speak to customer experience, there is some data science APIs (big transactional data) but that is in the minority.*

APIS CURRENTLY UNDER DEVELOPMENT AT BANKS INTERVIEWED

Most popular

- Transaction history/customer transaction APIs
- Payment APIs

Common

- Customer profile APIs
- Bank products catalog
- Transfers
- Credit scoring

Specific banks

- Modify bank card limits
- Authentication
- API endpoint for data on location of branches/ATMs
- Big data of anonymized transactions



APIS IN ACTION: EXISTING USE CASES FOR APIS IN BANKS

Two main examples of current and underdevelopment APIs were shared by numerous respondents:

- An API that drives the functionality of a bank's mobile app
- Work towards building a 360 view of the customer, starting with bankheld customer data.

An API that drives the functionality of a bank's mobile app

Several banks were already using an API to power their mobile app product. Often, the initial intention of the bank was to create a mobile application. In beginning this body of work, the software development team usually discovered a number of complexities that meant duplicated effort:

- They couldn't predict what device the app would be used on
- They wanted to make sure the app backend could be used for different use cases in future
- They needed a way to decouple their complex inhouse services from the mobile app frontend.

👤 We have an API that powers our mobile app, it was created with the intention of opening it up. The mobile app API is a full API, our intention was to integrate with mobile apps and address all those special needs like load balancing, performance, and security.

👤 The main goal was to create an app for our customers, and an API came as byproduct of that. The total dev time was 6 months, followed by a 3 month rollout period. The API and all the connections were built inhouse but the app itself

was built by a small local company. Today, for all our customer touchpoints, mobile is already number one.

👤 Originally the API was built for the application, but things happened pretty fast, you don't know what devices we will have in a few years, and you can't foresee what channels customers may use in future, that was our first point for taking an API approach. Then secondly, whatever serves the app as a backend will have to serve other uses, most of which we don't see. Thirdly, we can then open that up to the world. While this design was a technical decision, the decision that this was something we want to reuse was a business one.

Work towards building a 360 view of the customer, starting with bankheld customer data

Several respondents discussed a potential future scenario where a bank would create an API product that connected with a customer's complete financial accounts. While many did see this as advantageous, they recognized that this would be an advocacy battle internally.

This reflects findings from a recent Open Bank Project and Bank Innovation study that found that amongst those banks interviewed for that study, around a third of respondents said that the bank thought using APIs to link to a customer's other financial accounts and products was only slightly important or not important at all.

For those who were interested in thinking through such a product, the majority believed the first step would be to make customer bank account and creditdebit data information via API. A customer accounts data API was a common API many banks are currently working on developing.



Several banks interviewed saw the development of a 360view API of a bank customer's financial information as the ideal customerfocused financial products. The view was that customers were beginning to demand this type of product, so any bank that is able to be the first to market amongst their local competition would be in a privileged position.

Additionally, such an API product would be needed as the foundation for offering valueadded services such as an account switching tool, or predictive calculators that could provide comparisons of what the customer's account with a third party financial provider would be worth if they used one of the bank's own products.

The first product banks want to implement is always multibanking features to aggregate all their customers banking accounts from multiple banks. Initially, the reaction was no way, banks will never do this. The top level reason tended to be security and data protection, but generally it was about not wanting to give competitors any space here. But they are beginning to understand, the customer already has those accounts with competitors anyway. So at the moment, there is not something like that sort of API services: there is not a portal into a customer's complete accounts. It will be interesting to see when the first banks start offering this type of service, to see how all the other banks react. Challenger banks and innovator banks are really interested in this. They might not have the biggest customer base, but they feel they have the greatest opportunity to be a hub for online banking. And with this sort of product comes all sorts of smarter banking features: you can be a hub of financial services. You can build new services on top of that, there are new use cases. It is a great chance for banks to build much more information and data about the customer.

One proposed model would be to have a data lake with all the customers transactions data and then have tools to extract a customer's events to build a 360 view of that customer. The data lake could also enable use of aggregated, anonymized data via API for other internal and possible partner purposes. It could also store external data sources so that a bank could enhance theory data by comparing a client's behavior against benchmarks or averages, or swap out external data sources depending on the analytics need. (For example, if data was used for credit scoring, one external source may be used, while assessing insurance needs might require different datasets to be applied). A data lake could also help banks meet regulatory standards where requirements to store customer data for a certain period of time are set. Finally, a data lake of this sort would be the early version of an API-enable load balancing efficiency mechanism for the bank, where different data sets could be switched out depending on the supplier's costs of using particular datasets.

Several bank respondents also noted that the culture of discussions around such a product had changed significantly in the past year. Prior to that, many executive and midlevel management leaders refused to consider such a product. They saw it as a threat to their customer loyalty as it would give their customers the information they needed to leave the bank for another supplier. This view had changed 180 degrees, so that now bank leaders were recognizing that such a product could be at the center of customer loyalty.

While mostly thought of in terms of retail banking, some banks are considering similar 360-view products for their corporate banking customers.

06

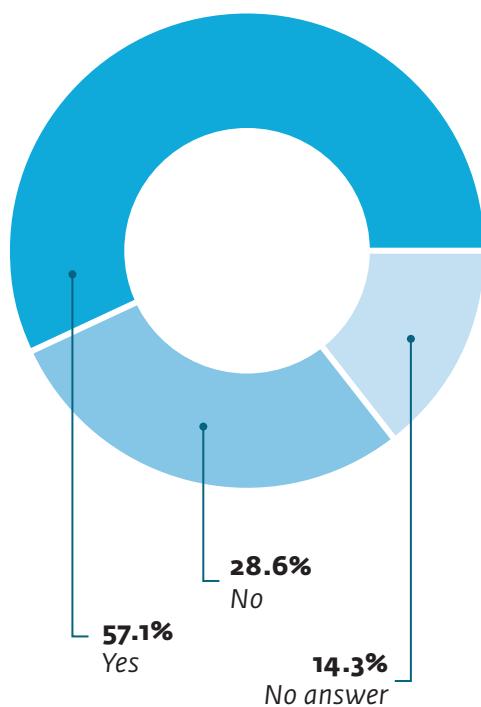
ANALYSIS: LEADERSHIP IN BANKING APIs

KEY FINDINGS

- ✓ Four bank cultures were identified in the study: API pioneers, API drivers, API talkers and API slow movers.
- ✓ The major two policy drivers for banks to commence and invest in an API strategy are:
 - ✓ to be more customer-focused/ customer-centric, and
 - ✓ to prepare the road ahead for a banking platform (including meeting PSD2 requirements).
- ✓ Successful strategies used by bank leads responsible for an API strategy involved building cross-departmental support through crowd sourced ideas hubs, mapping of processes with the potential to automate via APIs, and departmental presentations that included case studies from other traditional industries and introductions to APIs.



BANK IS DOING PROOF OF CONCEPT AND PILOT PROJECTS WITH APIs

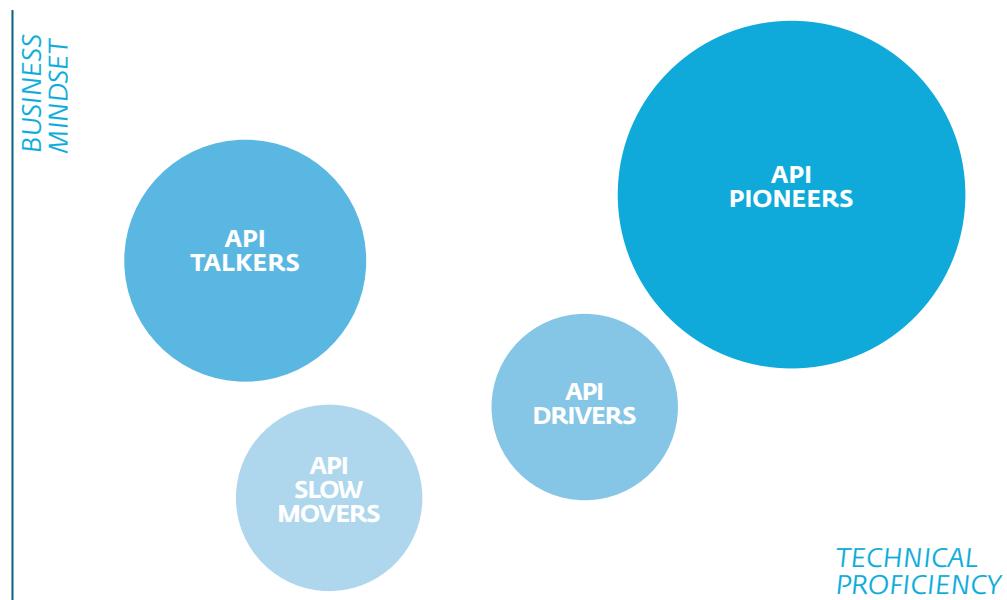


The majority of bank in a leadership position with their API strategies were using proof of concept and pilot projects to introduce the value of APIs internally.

This research uncovered four types of bank cultures around APIs:

- API Pioneers: banks that have prepared their architecture and are looking at a whole of organization API strategy
- API Drivers: banks that have Executive buyin and have created a role to lead an API Strategy
- API Talkers: banks with an API Strategy leader who is charged with encouraging crossdepartmental buyin
- API Slow Movers: banks where IT architects are introducing an API architecture and building solutions on their own, with the hope of demonstrating value through their own actions.

BUSINESS MINDSET AND TECHNICAL PROFICIENCY



While the majority of respondents were API pioneers (8 of 22), the second largest group of respondents were API talkers (6 of 22), followed by API Drivers (5 of 22), and API Slow Movers (3 each of 22). Of course, as API Days is an event series focused on APIs, many of our bank contacts reached for this survey skew towards being API pioneers.

Amongst those API pioneers that respond to our survey, we are seeing high levels of action on both the business and technical side, with strong enthusiasm and leadership in implementing an enterprise-wide API strategy. These survey respondents recognized that the wider bank industry is more typically placed in the API slow mover and API talkers categories, where the majority of banks do not have organizationalwide buyin and are building the technical capabilities to reorient their architecture to enable APIs.

Characteristics of API Pioneers (strong thought leadership and technical capabilities)

- Executive level support and budget for innovation
- Leaders engaging all of organization in activities to understand and make use of APIs
- Focus is on reorienting architecture quickly: building REST on SOAP services and committed to building new APIs as REST
- Consolidating internal catalog of services (decommissioning duplicate services and seeking to create composable functionalities)
- Using containers in production and looking for ways to integrate into organizational architecture
- Have chosen an API definition language or are using a common format to build new APIs

Characteristics of API Drivers (strong thought leadership and growing technical capabilities)

- Executive level support but leaders need to convince budget and departmental buyin
 - Leaders engaging in some crossagency training
 - Focus is on getting a REST interface up and running on top of SOAP services
 - Piloting container technology in a number of projects
-  Yes, the top management layer is in favor of innovation, but the reality is I am working within a business unit with a tough middle layer where it is difficult to sell the solution. It took me at least nine months to convince them that this is new API driven product is a good product to invest in.

API Talkers (strong thought leadership but limited technical proficiency)

- Executive level support in general but unclear as to what APIs are, just that something has to be done
- API Strategy operating from within one unit
- APIs discussed by wordofmouth or direct between IT department and end user dev teams
- Investigating the potential of container technologies to develop, deploy and distribute applications

API SlowMovers (no Executive buyin and limited action on technical reorientation)

- API Strategy is seen as an ITled activity, little awareness across rest of agency
 - Focus on abstraction layers, building APIs that connect to ESB and point to point communications
 - Aware that containers exist
- We don't have business involved in this, not yet. They don't care that much at the moment. We will show them the easy way to do this so it will spark their interest. Everything that will come out for external consumers will be API based, any new services in the bank will be APIbased. This way it will reach the right person and be less costly, go faster, and be a more grounded approach.**

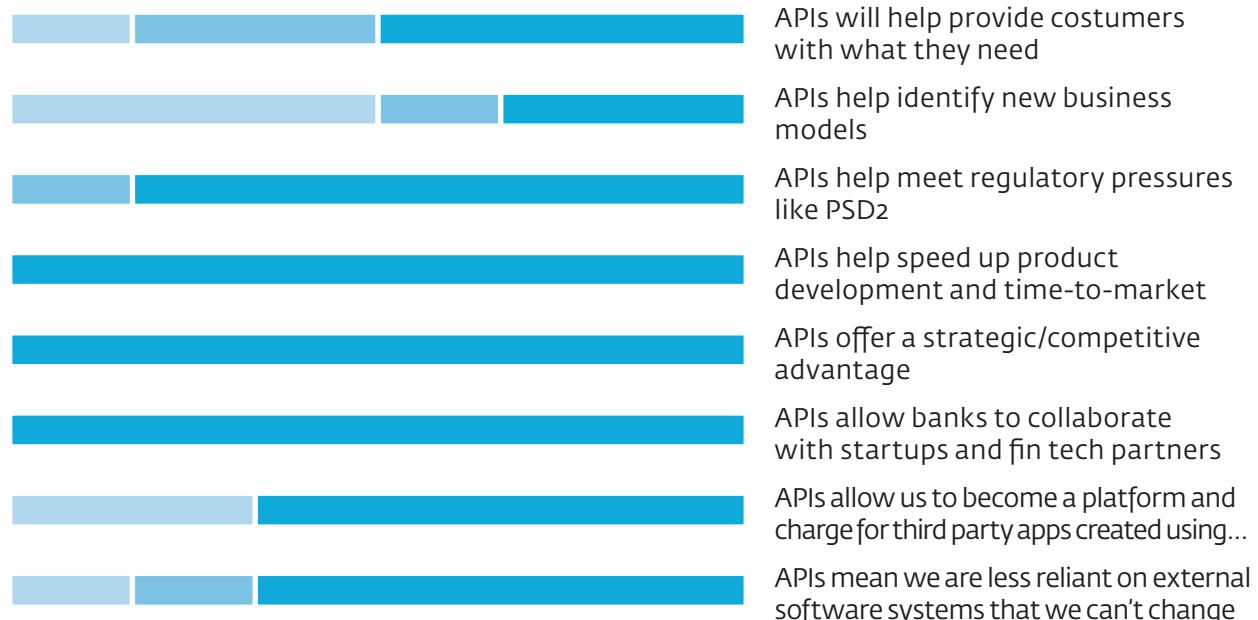
Internal Drivers and Strategies Used to Foster API Buyin

There were two major policy drivers that have been influential this year in encouraging Executive and departmental buyin for an API strategy with banks:

- Providing customers with the services they need and are demanding (that is, APIs are seen as a way for banks to reorient towards a greater customercentric focus).
- APIs are seen to help banks collaborate with startups and fintech partners.

API STRATEGY: MAJOR POLICY DRIVERS FOR BANKS

Major influence
Average
Not an important factor



In general, APIs are seen as helping banks obtain a strategic, competitive advantage, particularly by speeding up product development and reducing the timetomarket for new initiatives.

But demonstrating the internal arguments that are often still occurring in banks, while the majority of those interviewed saw APIs as providing banks with an opportunity to collaborate with startups and fintech partners, there was not complete agreement on this being done through providing an open bank platform. Interestingly, the new PSD2 regulations were not seen as a major policy driver, again reflecting the current internal debates occurring as to whether PSD2 provides a new opportunity to become a bank platform:

"One of the main insights especially with the PSD2 regulation is that this could be a huge opportunity to create a new business model, or it could be a big threat (where you become the backend for speedy fintech and Apple: they could take over the client relationship). It is a strategic decision: do you want to be the ultimate backend for all of that or do you want to continue with the customer relationships?"

For the API Pioneers and API Drivers who had built internal support for API strategies, they often implemented the following strategies:

- Slide decks of examples from other industries, eg. BMW, energy companies, telcos (more relevant than Amazon and Twitter)
- Multiple presentations to different management boards and departmental groups
- Crowdsourcing ideas platform encouraging staff to identify opportunities for new products or integrations
- Process mapping by innovation teams so they could help identify how to integrate and automate manual processes with APIs for various business units
- An "API game" that staff were encouraged to play to see how APIs could change the bank's way of doing things.

"A lot of presentations for higher management to educate them and created a program you could join. An internal platform to activate employee participation and gather ideas. We opened the

platform, did a lot of workshops. We had an API game which we forced people to play around. That actually worked, the whole company started talking about APIs."

 *We started by visiting departments and watching their business processes and learning about them. Then we helped them bridge the gap between technology and business processes, and we were able to automate the first few processes rather easily. It reduced 23 hours of their work a week, and there was a revelation there. After the first couple of those, we publicized them internally at staff meetings and on the intranet. Now we have an online forum on our intranet that we run on top of UserVoice so we can gather that information, and we have overall business goals driving things, and business process improvement, so it is a great way to prioritize all of that.*

MONETIZATION AND METRICS

No banks were implementing any form of monetization metrics at present, with few even using rate limiting thresholds (outside of mobile apps) to identify heavy users of their APIs and potential business models that might be indicative of usage rates.

While several banks had chosen an API management provider (see Section 5) that provided a monetization component in the product, none were using this feature as yet. In the main, where metrics were part of an API project, engagement was seen as the main measure of success.

 *Profitability is a key metric, but it really starts with user engagement. If there is added value to our customers, then in the end they will pay for it.*

 *"Our metrics for new API-enabled products are:*

Client engagement: total number of clients that are working with the product, tracked every minute.

Then validated by customer interviews.

Finally, can we create a business model behind it?"

 *For monetization, we are expecting app developers to make money off the app themselves directly.*



07

THE ROAD AHEAD

KEY FINDINGS

- ✓ Banks are now ready to invest and implement in API strategies.
- ✓ The main focus for the next 6-12 months is to reorient architecture and legacy systems and to build capacity to enable access to internal REST APIs.
- ✓ Banks are going through a process of identifying the highest priority internal APIs and are starting to build them.
- ✓ An open banking platform is seen on the horizon by many banks, but they believe that this will only be achievable in a 3-5 year timeframe.
- ✓ Banks have expressed interest in a number of specific features from API management solutions, including automating REST APIs from SOAP services, providing a product roadmap, integrating with rapid application development, and providing some middleware orchestration.

2015 IS TRULY THE YEAR THAT THE API REVOLUTION HAS COME TO THE BANKING INDUSTRY

A common theme amongst our interviews for this study was that the leaders charged with implementing an API strategy within their bank enterprise are finding a shift in the level of organizational buyin and willingness to participate in APIfocused discussions.

Several of the survey respondents shared that last year, Executive level decision makers were still hesitant to embark on an API strategy. Others with a technical leadership role also felt paralyzed by the enormity of the IT architecture redesign task ahead of them, amidst an organizational culture that was either reluctant or uninterested to discuss the potential of APIs.

This year, C-Level Executives have approved API strategies within their banks, and have assigned innovation leadership roles to help their bank steer them through to an organizationwide API implementation strategy.

In the main, bank leaders responsible for managing an API implementation see the work this year as being about putting their API house in order.

This involves:

- Completing architecture reorientation projects aimed at exposing bank functionalities via APIs (see section 2)
- Fostering a supportive internal culture and facilitating new internal partnerships in which API development is seen as product focused and not restrained by department structures (see section 6)
- Working through a prioritysetting process in which high value internal APIs are identified and built

- Testing several internal APIs (often in the retail banking arm of a bank)
- Building customer products with the newly created internal APIs
- Measuring customer engagement in new products as a way to determine potential to monetize.

Beyond this work program, the majority of those interviewed had an iterative mindset, in which they planned to map the next stage in building out their API strategy after initial feedback to the current work program. The majority of those interviewed believed in an open platform future for banks, in which 60% of a bank's APIs would be made available to partners and third party providers. But this was seen as a 3-5 year agenda.

2020: AN OPEN BANKING PLATFORM

A future of open APIs in which the bank plays a platform role was recognized as a future goal for many banks. A small number were working towards this goal already. Others had a personal ambition for this future track, but felt a long way from convincing their Executive or other management teams that this was desirable.

Geographic differences also played some part in how immediate an open banking platform future was being considered. Amongst UK and German banks, where open APIs are either being proposed by regulation (UK), or already required (Germany), an open banking platform was seen as an opportunity to enable new partnerships and business models to emerge.

Amongst European banks more broadly, several were viewing the introduction of the PSD2 regulations — in which banks would need to open APIs to allow third party payment providers to directly transact with consenting customers' accounts — as





an opportunity to build a competitive open banking platform. In some cases, banks were still considering how much they wanted to be customerfacing in this landscape: some were calculating whether it was beneficial to be the infrastructure that allowed third parties to extend new, agile products to customers. Others saw a potential in leveraging the regulatory environment to be firsttomarket with new payments products and to extend their open API catalog by forming new partnerships with fintech beyond the payments sphere.

IN THE U.S., THE MAJORITY OF BANKS INTERVIEWED DID SEE A BENEFIT IN PROVIDING AN OPEN API PLATFORM

They saw this work as achievable within the next 35 years. Several interviewed had previously tried opening up APIs — or had watched as other banks attempted to open APIs and build an ecosystem — but as this had not been successful, they were now more cautiously advancing this agenda, and only after first rebuilding their architecture and managing their internal API strategy.

However, banks were also uncertain as to what business models would be created from such a platform. Several asked if bank app stores had been shown to accelerate third party app development. Others were looking for traditional monetizing opportunities where app makers would pay a commission for every app they sold, rather than seeing the third party apps as an opportunity to reach new potential customers.

One potential model discussed would be that a bank would have an external facing developer portal. Each third party registering for an API key would have their own account dashboard.

Application starter kits with SDKs, documentation and API keys would be made available, along with approval processes where third parties would map what bank functionalities and data they would require in their app. Banks could then approve the application model and provide API usage data through the dashboard showing how each third party provider's end users were accessing the banks open APIs through the provider's app.

• Convincing the CEO was the toughest and the argument was: people build apps for Apple for free. This is leveraging external innovation: people are developing for free. So for him, it was about branding and innovation: being out there and communicating with customers and developers.

Several banks are also working on building innovation labs with an incubator arm that is funding and supporting new fintech startups. While these incubator models are focused on building a new set of fintech partners for each bank, at present there is little to no discussion on banks sharing open APIs with these partners, although it is imagined that this would be encouraged when an open banking platform is more fully developed by each enterprise.

Apart from regulatory requirements for open payments APIs, banks were more often thinking in terms of opening up their transactional data to third parties and partners as a new product, however the business model and end product were still to be tested:

• We have a lot of data that is interesting to our customers to gain insight into what their industry are doing, looking at several models where we can provide bodies with extra data maybe even in an API way. We are challenged by whether new user interfaces will even be necessary for this, or whether you should provide them with a state of the art API and let them do the user interface themselves.

TECHNICAL NEEDS

Security

For APIs being built with a view to opening them to partners and providers in future, OAuth2 was identified as a key security protocol that banks were looking to use.

But all banks saw a need to build a multifactor security setting. Many were currently still testing security protocols and identifying which ones would be strong enough for to protect their customers. In particular, the majority of banks were dissatisfied with the current twofactor authentication model and were implementing enhanced versions of twofactor authentication that might require API keys for individual devices, more stringent proof that a device was in the hands of a customer, or additional authorization steps

(threefactor authentication?) to ensure that each login session was being completed by the bank customer.

API Design

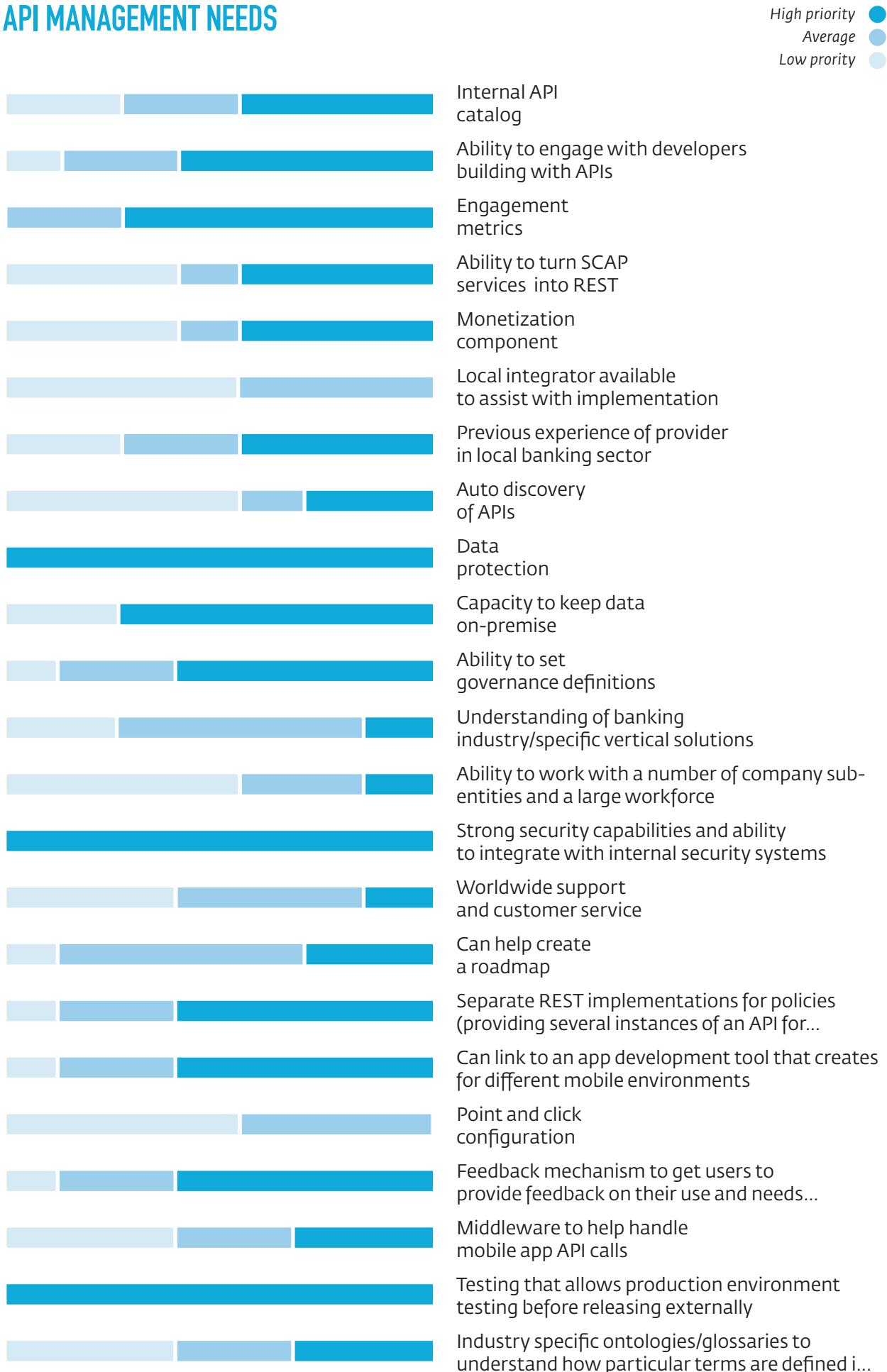
The majority of respondents expected to have 100% of their internal services available via REST API within three years. The majority of banks interviewed were moving towards use of Swagger or RAML as the starting point for designing an API. In addition, this documentation was also beginning to be used by innovation leads as a process map for bringing business and engineering teams together in the conceptualization and design of an API. It was also seen as an important tool to encourage a reconfiguring of internal business units so that APIs are designed as products that may span several use cases.

APIS IN A GLOBAL SETTING

Onboarding internal developers was seen as a challenge for banks working across national borders. This is a problem expected to grow amongst European banks as banks face issues in onboarding payment merchants when the PSD2 regulations come into effect. While APIs can be built in a standardized way for about 80% of its functionality and data usage, each nation may have a regulatory environment that requires specific data storage or security arrangements. This could account for up to 20% of how an API manages data transfer or carries out bank functions.



API MANAGEMENT NEEDS



In addition to banking regulation differences, industry codes and definitions also change across various national borders. For example, if a bank is using credit scoring data or provides insurance services, nomenclature and coding of buildings can differ between various national industry associations, creating difficulties in managing an universal API for the bank.

The most important API Management features for banks are:

- Data protection
- Strong security capabilities and ability to integrate with internal security systems
- Testing that allows production environment testing before releasing externally

The second most important set of features are:

- Engagement metrics
- Capacity to keep data onpremise
- Ability to engage with developers building with APIs
- Ability to set governance definitions
- Separate REST implementations for policies (providing several instances of an API for separate private/partner/public use)
- Can link to an app development tool that creates for different mobile environments
- Feedback mechanism to get users to provide feedback on their use and needs eg. internally.

Overall, banks interviewed pointed to several key features they would like to see from API management providers:

- **SOAP to REST:** Overwhelmingly, banks were looking for tools that could help automate the transformation of SOAP services to REST. Along with this, they were seeking tools to help auto discoverability of services in their legacy systems, the ability to autodocument the mapping of SOAP to REST (including mapping between subservices and the REST API). One respondent suggested the need to be able to create separate, multiple implementation policies for a REST API to meet internal, partner and public contexts. The ability to be able to create several instances of a REST API and apply different policies was seen as an implementation advantage.

• **A product roadmap:** Several banks would like to see a process workflow that could help them create and manage an API product roadmap. Desirability for this type of functionality reflected a wider request for wholeofapplication lifecycle tooling within their API management solution.

• **Application development tooling:** Following on from the product roadmap requests, banks wanted their API management solution to include application development platforms. At a minimum, this may include the autogeneration of SDKs for their APIs, but more often was described as a more complete rapid application development product.

• **Node.JS integration:** Several banks had adopted Node.JS as a language to facilitate integration between their legacy systems and APIs, and were expecting API management solutions to be able to cope with Node usage. Alongside this, banks were looking for API management solutions that could integrate with and respect complex security legacy systems.

• **Support for data to be kept onpremises:** Several banks had regulatory requirements of internal organizational policies that required all customer data to be maintained onpremises in the bank's data servers. They expected API management solutions to respect this requirement and to be able to manage API endpoints from behind the firewall.

• **Rate limiting** was seen as an important feature as a security measure to help identify DDoS attacks and to help manage data access and functionality, and as a data source for identifying future new business model design.

• **Middleware orchestration:** One issue is that APIs that empower mobile apps often need to make a series of calls, and an API middleware layer could help with orchestration and more easily manage multiple calls.

• **Metadata schema alignment:** Along with this was the desire for API management solutions to help break down intraindustry and national barriers that may mean that an API endpoints and attributes use one nomenclature in one location and something slightly different in another. Some way to automate matching up of codes from various schema was seen as a valuable feature.

CONCLUSION

BANK READINESS NEEDS STRENGTHENING

This study shows 2015 is finally the year that banks are ready to implement more strategic API approaches

To best harness this newfound support for APIs within the banking industry, enterprises need to pursue five strategies:

- ✓ Banks must reorient architecture towards enabling a build and replace program, where when new services or API needs are identified, these are built as REST APIs to replace existing services.
- ✓ API leaders within banks must build crossdepartmental support and interest in an API strategy.
- ✓ As new APIs are created, these are best done with collaboration between business and engineering and adopt an API design approach that ensures that internal APIs may later be opened up to partners or third party providers. This includes the need to set appropriate identity management and security protocols to enable later opening up of internal APIs to a wider audience.
- ✓ Banks must build expertise in developer experience. This includes implementing best practices in developer engagement for both internal developer teams as well as paving the way to ensure banks can leverage external innovation with developers in the future.
- ✓ Banks must measure customer engagement of any products and services that rely on, or are built with, the bank's API stack.

The Executive level is ready to commit resources and budgets to an APIfirst model. Leading IT architects have helped steer the enterprise into a position where it either has the architecture in place or is preparing it. This internal API layer is essential for enterprise readiness to create new products, deploy applications in new channels and partner with third party providers in new markets.

To ensure their readiness, banks are in the early stages of using APIs internally: the first indicator of success for an APIfocused bank.

While the Executive level has lent support for API strategies within banks, those charged with making it happen often still need to build internal collaboration and willingness.

Sharing the advantages that APIs create in helping a bank focus on customer needs and reorient towards a stronger customercentric perspective is the most persuasive policy driver internally for banks. Departmental managers and leadership teams often need presentations and training to better understand the potential of APIs. API leaders are having greater success when they can help departments map processes that could be more efficient with the use of internal or external APIs. Encouraging ideas via an intranet crowdsourcing platform



has been effective for many innovation leaders in charge of building crossorganizational support.

Like many industries coping with disruption on a massive scale, the main focus for banks at present is on taking some initial steps rather than justifying return on investment from their actions. Evaluating effectiveness in terms of monetization or mapping out a three to five year strategy is not a priority. Instead, banks are taking six or twelve month implementation cycles to test new strategies and to identify the best progress to take after seeing the impact of their initial actions.

To move effectively, banks are needing to engage and support internal developers to be able to work more agilely, access and contribute to open source projects, and to be inspired through a developer experience that encourages them to make use of internal APIs and to build with a view to an open banking platform future.

API leaders with clear vision of what is possible with APIs see the next two years as about completing the reorientation phase: establishing their internal architecture and using APIs internally to manage customer engagement, partnerships and build new products with agility.

In the next three to five years, many of these API leaders expect banks to be taking steps into becoming an open bank platform, where APIs are made available to third party providers, and where banks are consuming more data and other services via API themselves.

ABOUT THE AUTHORS



MEHDI MEDJAOUI

Mehdi Medjaoui is the founder of APIdays Conferences, the main conference series on APIs worldwide, and also founder of OAuth.io, an API integration middleware already powering more than 25,000 applications.

Mehdi has evangelized APIs since 2011, and strongly believes APIs define a new supply chain: on the business, the technical, and the legal side of the web. Mehdi is a regular speaker and industry influencer at many API and developer conferences, where he is known for sharing his point of view on the industry and where the API economy is going, always with opinionated ideas that help drive the debate forward.

In his free time, Mehdi is writing a Book on Man versus Software : The Great Substitution which will be available Q1 2016.



MARK BOYD

Mark Boyd is a writer and API industry analyst. He has previously written about the growth of the API economy for ProgrammableWeb, The New Stack, and Venture Beat and chairs several API conferences.

He has published a number of ebooks on the API lifecycle, container technologies, and best practices for API adoption. He also regularly conducts industry research and manages data-driven projects that draw on open data and proprietary sources.

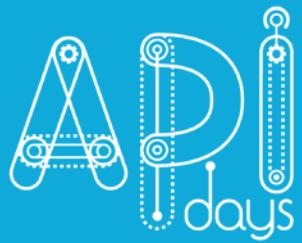
With a public health and urban planning background, Mark is especially interested in how APIs can enhance citizen and local business engagement and foster economic development in smart cities.

REPORT DESIGNER: www.bernatfont.cat/en

PHOTOS ON PAGES 4, 12, 16, 26 AND 44: Maria Dabow

APIDAYS

contact@apidays.io
apidays.io
<https://medium.com/@APIdays/>
@APIdaysGlobal



OPENBANKPROJECT

