Gartner

G00278085

Hype Cycle for Open Banking APIs, Apps and App Stores, 2015

Published: 6 July 2015

Analyst(s): Kristin R. Moyer

The focus of this Hype Cycle is the self-service discovery, provisioning and creation of new banking services by ecosystems that are inside and outside the bank. APIs, apps, app stores and partners are key to enabling open banking.

Table of Contents

List of Tables

List of Figures

Analysis

What You Need to Know

Banking is becoming open, despite security and compliance concerns about the use of technologies like Web APIs in such a heavily regulated industry (see "How to Make APIs a Banking CEO and Board Priority"). CIOs previously had three main business drivers to create and deliver an open banking strategy:

- **CEO business priorities:** Open banking enables several of the CEO's top business priorities, including revenue growth, efficiency, customer experience and talent management (see "The Business Value of Banking APIs").
- **Digital transformation:** Open banking is a key element of digital transformation because it creates a new channel, and engages new ecosystems of internal and external partners (see "Citi Uses Hackathons to Accelerate Digital Innovation").
- **Competitive pressure:** Financial technology (fintech) firms are using open banking aggressively to enable digital transformation (see "Anyone Can Build a Bank"), and a number of licensed/ regulated banks are beginning to use them in similar ways.

Government and regulatory body standards will now provide CIOs with an additional and potentially even more powerful driver to adopt open banking strategies and technologies. Several government and regulatory bodies are considering, developing or have already developed API standards or gateways for banking (for example, the National Payments Corporation of India [NPCI], the U.K. government and the European Commission [Payment Services Directive/PSD2]). We expect to see regulators in 50% of G20 countries create open banking API standards or gateways by 2018, and we believe this will lead to the rapid adoption of open banking strategies and technologies (see "How to Make APIs a Banking CEO and Board Priority").

Open banking is the self-service discovery, provisioning, and creation of new business models and services by ecosystems that are inside and outside the bank (see "Reference Model for Open Banking APIs, Apps and App Stores"). APIs, apps, app stores, developer/partner ecosystems and other technologies provide CIOs with the ability to do the following (see "The Business Value of Banking APIs"):

- **Enable mobility and innovation:** For example, creating a third-party developer community that can access APIs to build mobile apps, and using a banking store to enable customers to discover/download apps and share ideas.
- **Increase product and service accessibility:** For example, providing APIs that make products, services and pricing more accessible.
- **Create new business models:** For example, providing APIs and platforms that enable users to create and market their own services.

Open banking is not without risk. Software-defined application services (SDAS) and application services governance are needed to enable security, integration, monitoring, load balancing, composition, orchestration and others. Strong collaboration with enterprise risk management and legal departments is needed to ensure regulatory compliance. Intellectual property issues, which are already thorny when it comes to banking software, become extremely complex in a more open environment. Many open banking initiatives also are adding to an already highly complex bank architecture, infrastructure and operations.

The opportunities with open banking, along with the threat of becoming irrelevant to customers in the digital world, are pushing banks to confront these risks. If these risks are addressed, then open banking will enable CIOs and line-of-business leaders to transform what it means to be a bank.

The Hype Cycle

The Four Main Categories of Open Banking

This Hype Cycle examines the four main categories that are relevant to pursuing open banking:

- **Strategies:** These include an Open Banking Strategy, Open Business Model and Programmable Business Model.
- **Architecture:** This includes APIs (Commercial Banking APIs, Retail Banking APIs, Internal APIs, Fintech APIs and Open Bank Project), Software-Defined Application Services and Application Services Governance.
- **Technology:** This includes API Marketplaces, Apps (Commercial Banking Apps and Bank Employee Mobile Apps), Banking App Stores, Open-Source Banking Systems, Community Source Banking Systems, OpenID Connect and OAuth.
- **Ecosystems:** These include API-Enabled Ecosystems, Internal Hackathons, External Hackathons and Crowdsourcing.

Off the Hype Cycle

- **Business Models for Open Banking:** This year, we have replaced the term "Business Models for Open Banking" with "Open Business Model."
- **Vendor Readiness for Open Banking:** We removed this because it is covered in "Open Bank Systems" in the "Hype Cycle for Digital Banking Transformation, 2015."
- **Developer and Partner Ecosystems:** This year, we have replaced the term "Developer and Partner Ecosystems" with "API-Enabled Ecosystems."

New on the Hype Cycle

- Programmable Business Model

Why This Hype Cycle Is Front-Loaded

The early adoption of open banking strategies and technologies has already occurred, but early mainstream adoption is just beginning. The vast majority of emerging technologies on this Hype Cycle are positioned between the Innovation Trigger and just past the Peak of Inflated Expectations (see Figure 1). This is because open banking is an emerging trend in banking and investment services, and many of the architectures, technologies, strategies and business models in this Hype Cycle are relatively new. For these reasons, open banking implementations in the 2015 time frame will be somewhat more expensive, complicated and risky for banking CIOs to implement. Building a solid foundation in terms of business strategy, business models and SDAS will help CIOs reduce the risk of deployments in the next 12 months.

Fast Movers

More than 50% of the strategies, architectures, technologies and ecosystems on this Hype Cycle will become mainstream within the next two to five years. Some are especially fast-moving, even though they are not yet in the Trough of Disillusionment:

- **Internal and External Hackathons:** Banks have accelerated their usage of Internal and External Hackathons this year, and we expect this to continue over the next 12 months as they approach the Trough of Disillusionment.
- **Internal APIs:** Many early adopter banks began their open banking journeys with Internal APIs. This is why they have a solid foundation to begin using APIs externally (for example, to expand product accessibility). We expect to see this trend accelerate in the next 12 months.
- **Retail Banking APIs:** We expect to see these become mainstream within two to five years, given their ability to improve customer experience, expand product accessibility, and attract and retain talent.
- **Commercial Banking APIs:** Despite being just past the Innovation Trigger, we expect to see these become mainstream within two to five years, given their ability to enable better partner integration and product/service accessibility.
- **Commercial Banking Apps:** Although most banks have deployed a first-generation Commercial Banking App, we believe usability will improve over the next two years as a next generation of app functionality becomes more focused on the role of the user.
- **Banking App Stores:** Many banks are making their banking apps more narrowly focused, which increases the total number of apps they are providing. Banking App Stores make it easier for customers to find banks' apps.
- **API-Enabled Ecosystems:** Many banks already have run hackathons and created or participated in fintech accelerators, but few have purposefully created a third-party ecosystem the way banks like BBVA and Citi have (see "Citi Uses Hackathons to Accelerate Digital Innovation"). We expect to see the purposeful creation of API-Enabled Ecosystems increase rapidly over the next two to five years.

- **Bank Employee Mobile Apps:** We expect these to achieve mainstream adoption in two to five years, given their ability to improve efficiency, communication, collaboration, access to information, and presentation with face-to-face customer experiences.
- **Software-Defined Application Services** and **Application Services Governance:** We expect SDAS and Application Services Governance to become mainstream within two to five years because they help to enable governance, monitoring, security, load balancing and management of services.

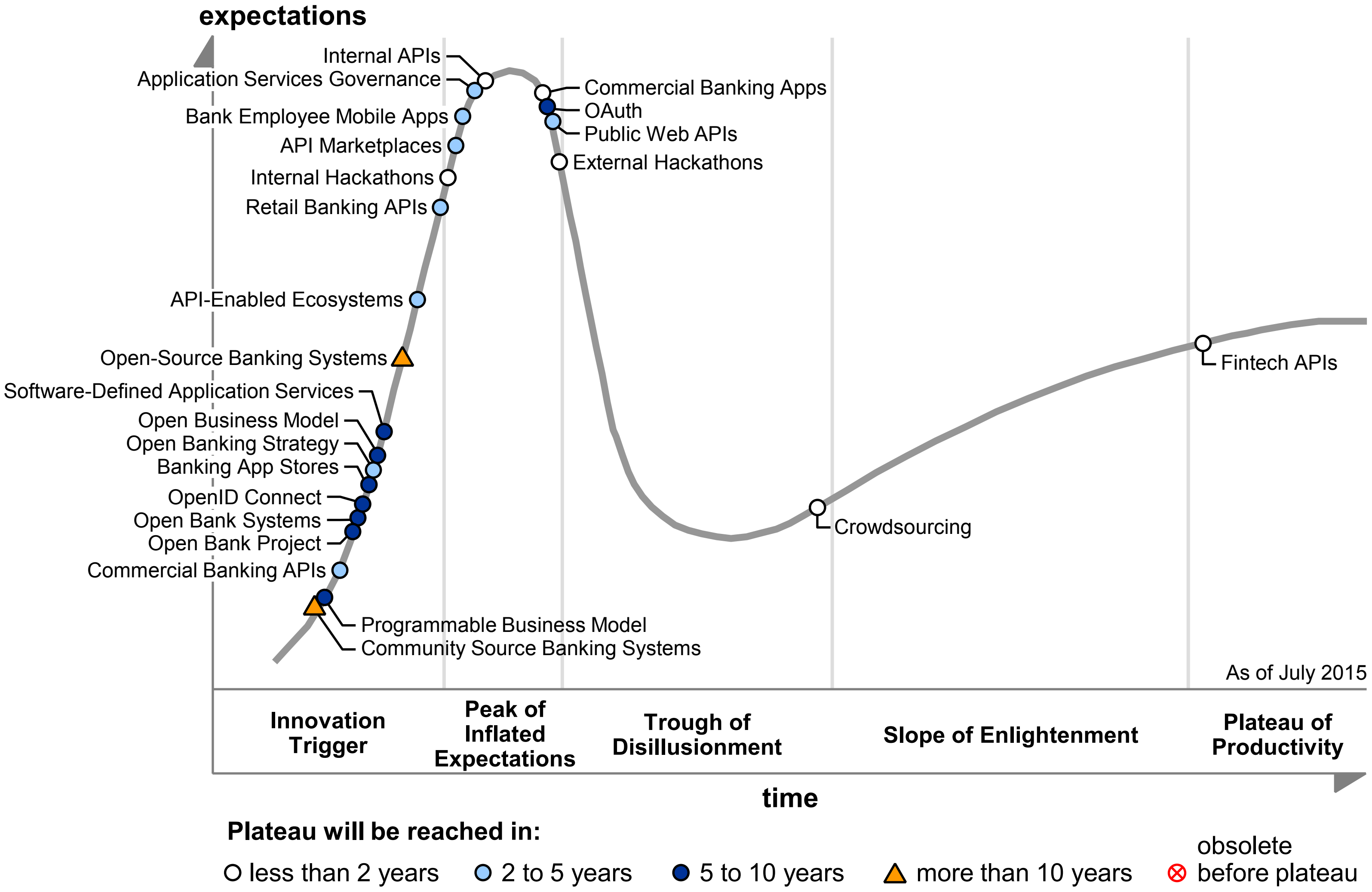
Who Can Benefit From This Hype Cycle?

The focus of this Hype Cycle is targeted specifically at banking CIOs. However, it is also relevant for other CxOs, IT leaders, and digital marketing and line-of-business executives.

Multigeography, Multicontext Analysis

This is a multigeography, multicontext analysis. Individual bank use cases are likely to have different speeds of adoption and maturity. Client inquiry can be used to identify which strategies, architectures, technologies and ecosystems are optimal for individual bank objectives.

Figure 1. Hype Cycle for Open Banking APIs, Apps and App Stores, 2015



Source: Gartner (July 2015)

The Priority Matrix

In the next two to five years, APIs will have the most profound ability to be transformational because they can be used to:

- Reduce time and cost to market for new business capabilities by up to 90%.
- Increase net revenue growth by up to 30% year over year.
- Enable co-creation between employees, developers, partners and even other banks.

Over the next five to 10 years, the Programmable Business Model will enable internal and external users to create their own business models through building blocks such as APIs, open data and marketplaces. Programmable Business Models are disruptive in nature because the bank is no longer the only one that creates and controls its business models. This multiplies revenue opportunities for banks, but also multiplies the ability of individuals, organizations and other industries to create and deliver banking services directly to customers.

Most of the strategies, architectures, technologies and ecosystems included in this Hype Cycle will become mainstream within the two- to five-year time frame. This is unusual for a Hype Cycle, and reflects the urgency of how quickly banks need to transform the way they do banking — before someone else does it for them (see Figure 2).

Figure 2. Priority Matrix for Open Banking APIs, Apps and App Stores, 2015

benefit	**years to mainstream adoption**			
	less than 2 years	2 to 5 years	5 to 10 years	more than 10 years
transformational	**Fintech APIs** **Internal APIs**	**Commercial Banking APIs** **Public Web APIs** **Retail Banking APIs**	Open Bank Systems Programmable Business Model	
high	**Commercial Banking Apps** **Crowdsourcing**	API-Enabled Ecosystems Application Services Governance Bank Employee Mobile Apps Open Banking Strategy	Banking App Stores Open Business Model Software-Defined Application Services	
moderate	External Hackathons Internal Hackathons	API Marketplaces	OAuth Open Bank Project OpenID Connect	
low				Community Source Banking Systems Open-Source Banking Systems

As of July 2015

Source: Gartner (July 2015)

On the Rise

Community Source Banking Systems

Analysis By: Kristin R. Moyer

Definition: Community source banking systems are banking-specific applications or components. They are either: (1) developed by banks or commercial companies that sign an agreement and provide financial and/or human resources, and get exclusivity in influencing the development of the banking system during an initial closed stage; or (2) a marketplace that is operated through a mediator that matches buyers and sellers together.

Position and Adoption Speed Justification: Community source banking systems (based on a definition provided by OSS Watch, "Community Source vs. Open Source," by Gabriel Hanganu, 13 March 2012) are different from open-source banking systems. With open source, software code is made available for free to everyone for inspection and modification. One way that community source can operate is through a consortium of companies that contributes resources (financial and human) during an initial closed stage. After the initial closed stage, code is made available, as is the case with open source. Another way that community source can operate is through a mediator, who matches buyers and sellers of source code (from homegrown applications) together.

Community source at an infrastructure layer and for commoditized functions (like public reference data access, cleansing and validation) may be especially appealing to banks. However, operational risk is a concern, since banks have less control over community source banking systems than homegrown or on-premises applications, which can be customized through parameterization and configuration. Given a slow rate of adoption of community source in the banking industry, we expect that it will take more than 10 years for it to become mainstream.

User Advice:

- Monitor the emergence of community source banking systems and consider participation if the objective is to: (1) reduce standard application, component and infrastructure efforts; and (2) extend application development bandwidth for bank applications, as opposed to developing apps for customers.
- Evaluate opportunities to leverage community source components such as user interface widgets for spot liquidity, orders or trade repositories as they become available. These may enable faster time to market and lower costs than are involved in evaluating, selecting and implementing a packaged application, which includes more functionality than needed.
- Identify community source banking system alternatives as part of the RFI process when facing the replacement or retirement of a component or application.

Business Impact: Community source banking systems can enable efficiency and faster time to market for commoditized functions.

Benefit Rating: Low

Market Penetration: Less than 1% of target audience

Maturity: Embryonic

Recommended Reading: "Community Source vs. Open Source," by Gabriel Hanganu, OSS Watch, 13 March 2012

Programmable Business Model

Analysis By: Kristin R. Moyer

Definition: A programmable business model enables internal and external ecosystems to create, manage and adapt their own business models. Programmable business models are dynamic and disposable. They enable enterprises to not only survive, but to thrive within the volatility that is inherent in digital transformation by competing with hundreds of business models — rather than just one.

Position and Adoption Speed Justification: Most enterprises have one (or a small number of) fixed, centrally planned and controlled business model to ideate, create, engage, offer, monetize and adapt the way value is created and retained. At times, the enterprise is leveraging users outside the enterprise to create value (for example, using hackathons to support ideation), but it still controls the overall business model. This is an "inside-out" approach to value creation that may not change how the company makes money.

Programmable business models expose assets like intellectual property, business functionality and business processes so that users inside or outside of the enterprise can create, manage, adapt and retire their own business models. Enterprises, people and "things" can provide and/or use programmable business models. Programmable business models are dynamic and disposable because they are easily created, changed and retired through self-service platforms.

This means enterprises no longer need to rely on one business model to create sufficient shareholder value. Rather, sustainability comes from an outside-in approach with multiple business models that are created, managed and maintained by internal and external ecosystems. This enables enterprises to multiply value-creation through two-sided markets (with one side being customers and the other side being ecosystems that create business models and value) and network effects (where value increases as the number of customers and ecosystems increase).

Some of the technologies used to enable programmable business models (like Web APIs, open data and marketplaces) are widely available today. The programmable business model platform can, therefore, be built and used today by referencing existing/legacy financial instruments and the Internet of Things. In the longer-term, the programmable business model will be an enabler of the programmable economy, which will emerge from metacoin platforms and a generalized notion of value exchange that goes beyond monetary exchange.

We do not expect the programmable business model to become mainstream for five to 10 years due to barriers like closed industry structures and operating models, and concerns about intellectual property, regulatory compliance, security and operational and reputational risk.

Gary Olliffe and Eric Knipp are co-authors of this analysis.

User Advice:

- Identify the existing business models used by your enterprise or organization.
- Make a catalog of the emerging business models that are being used in your industry at each stage of the Gartner Business Model Framework.
- Evaluate gaps between your existing business models and the emerging business models in your industry.
- Pick one of the gaps at a specific stage, such as create, and do an internal pilot that gives more control to employees to adapt and change existing business models.
- Pick another gap and do an external pilot with a trusted group of existing partners to customize one stage of your existing business model.
- Identify new internal and external ecosystems of partners that can leverage your programmable business model.
- Define and differentiate the individual business model components that can be leveraged by others as raw materials instead of understanding the end-to-end business process and how it can be differentiated.
- Become a designer of building blocks or platforms, rather than business models.

Business Impact:

- Revenue growth — by enabling the creation of entirely new solutions and monetization strategies from a potentially infinite number of ecosystems.
- Cost reduction — by reducing the time and cost of bringing new business capabilities to market.
- Customer experience — by putting more control in the hands of users (sometimes the actual customer) that may be closer to the actual customer than the provider of the programmable business model.
- Agility — by enabling rapid time to market and the ability to adapt to volatile business conditions.

Benefit Rating: Transformational

Market Penetration: Less than 1% of target audience

Maturity: Embryonic

Recommended Reading: "Introducing the Gartner Business Model Framework"

"The Business Value of Banking APIs"

"Industry 2020: Open and Horizontally Focused"

Commercial Banking APIs

Analysis By: Kristin R. Moyer; Fabio Chesini

Definition: Commercial banking APIs expose data, business processes and application capabilities through interfaces typically based on Web-oriented architecture (WOA) principles. Commercial banking APIs in this context are for use outside the bank, and are often semiprivate or private, depending on the type of users (such as employees, customers, partners or third-party developers) that will be accessing the APIs.

Position and Adoption Speed Justification: The API architecture pattern has been around for 40 years — it is a foundation of code development. Most commercial banking CIOs have hundreds or even thousands of APIs within various silos of their organizations. However, very few of those APIs will be helpful in creating new value as part of digital transformation. Most were not built using modern Web API practices or for modern customer use cases. As an example of a missing modern practice, they tend not to be managed with disciplined service governance techniques. This means their use is inconsistent and not well-understood/traceable. As another example, most existing commercial banking APIs are one-off APIs that were created for specific customers and focused on the application, which makes them difficult to use. The defining characteristic of a good commercial banking Web API is that it is created for developers and is very easy to use. For example, Stripe enables merchants to accept bitcoins with one line of code.

Some banks are already using commercial banking Web APIs. For example, external APIs started within commercial banking at Barclays. Mobile Checkout enables Pingit customers to order goods from merchants and pay for them using Pingit from non-Barclays mobile apps through an API. Commonwealth Bank of Australia has an API platform — Pi — and an app store for merchants. New market entrants using APIs as a key part of their strategies are emerging. For example, Seed is a bank (U.S. Federal Deposit Insurance Corp. [FDIC]-insured up to $50 million) that provides an API that enables unlimited customization and rich notifications. Standard Treasury is working with the Financial Conduct Authority (FCA) and Prudential Regulation Authority (PRA) to become an authorized bank.

Commercial banking APIs are currently less mature than retail banking APIs. However, we expect to see APIs used to support commercial banking processes such as account opening/onboarding, information (consolidated account, reconciliation and transaction information), foreign exchange (quotes and execution) and others. While some public Web APIs may be provided in commercial banking, many will likely be semiprivate (for all bank customers) or private (for specific bank customers).

We expect to see more than 75% of the top 20 commercial banks in the world (by assets) deploy a semiprivate or public Web API within the next two to five years. We also expect to see 50% of all APIs across industries, including the banking industry, be B2B-related. This is a very rapid rate of adoption for commercial banking APIs, given that they are so close to the trigger point of the Hype Cycle. However, the benefits for commercial banking, such as customer integration, are transformational, and we believe this will drive rapid adoption.

User Advice:

- Provide commercial banking APIs to integrate information, onboard customers, enable mobility, pursue new revenue opportunities and create new business models.
- Don't assume that "if we build it, they will come." Establishing traction for a semiprivate or public Web API requires a clear strategic direction, business model and application service governance (including API management). Additionally, it requires creating awareness of its availability and making the API easy to consume via a self-service developer portal.
- Before releasing your commercial banking API to the outside world, consider the traffic and resource implications for your data center. Complement your infrastructure with Web API management products and cloud services as required, and ensure that you have an adequate funding model in place for its ongoing operation.
- Put application service governance in place, including API management, prior to launching an open banking strategy. Also involve the enterprise architecture team and security team — for example, taking into consideration dynamic application security testing (DAST) and static application security testing (SAST; see "Magic Quadrant for Dynamic Application Security Testing" and "Magic Quadrant for Static Application Security Testing").

Business Impact: Web APIs can be used in commercial banking to improve mobility, increase product and service accessibility and create new business models. The business value of commercial banking APIs address key CEO business priorities, including revenue growth, cost reduction and customer experience.

Benefit Rating: Transformational

Market Penetration: Less than 1% of target audience

Maturity: Emerging

Sample Vendors: Standard Treasury

Recommended Reading: "The Business Value of Banking APIs"

"How to Make APIs a Banking CEO and Board Priority"

Open Bank Project

Analysis By: Kristin R. Moyer

Definition: The Open Bank Project is an open-source API management software and app store created specifically for the banking industry that provides data APIs, governance, intermediation, metadata, registry and repository, and security. The Open Bank Project operates under the Affero General Public License (AGPL), which permits free use, access to the source code, modification and redistribution.

Position and Adoption Speed Justification: API management is available through proprietary and open-source software. A number of nonindustry-specific, open-source API management platforms have emerged — for example, apiGrove and WSO2 API Manager.

The Open Bank Project focuses specifically on the banking industry, and provides and manages REST APIs using OAuth authentication so that bank customers do not disclose their credentials to third parties. The Open Bank Project provides a form of canonical message/data model that sits in between applications and the bank's internal IT systems. It abstracts the complexities of different core banking systems so that developers and vendors can securely connect their applications to multiple banks.

The Open Bank Project revenue model includes a commercial licensing option through which the Open Bank Project can manage APIs, leverage its developer community, execute hackathons and build new apps for banking clients. Commercial licenses, community building and support are also available from the project founder, TESOBE (Germany and the U.K.), and its partners (such as Vanso in Nigeria, Versent in Australia and DataArt in the U.S.).

During the past 12 months, the Open Bank Project has:

- Added a common payment interface and endpoints for customers, customer messages, financial products, branches and ATMs
- Grown to approximately 300 developers
- Been used by more than 350 apps and prototypes
- Organized 11 hackathons (for such institutions as BNP Paribas, Ulster Bank/Royal Bank of Scotland and Rabobank)
- Begun working more closely with API management vendors and core banking vendors (notably Axway and Temenos)
- Renewed license agreements for banking customers as they prepare for production rollouts — and deployed its sandbox API into new Tier 1 banks in Europe (including Ulster Bank and BNP Paribas)
- Added two partners in Australia and the U.K.
- Supported the U.K. Treasury in drafting a framework for an open API standard for U.K. banks.

We believe that the Open Bank Project will be appealing to developers because of the ability to build an app that can be used by any bank or customer through the use of connectors that interface between the open-source API management platform and the bank's internal systems. However, we expect that the Open Bank Project will be less appealing initially to large commercial banks that desire higher levels of control and differentiation, compared with nongovernmental organizations and microfinance institutions. Because of this, we believe that it will take five to 10 years for the Open Bank Project or other banking-specific open-source API management platforms that emerge to reach the Plateau of Productivity.

User Advice:

- Evaluate opportunities to use the Open Bank Project for nondifferentiated functions.
- Use the Open Bank Project and apps built from this platform to close competitive gaps quickly.
- Create a common catalog of APIs shared among banks as another approach to openness and an alternative to the Open Bank Project.
- Consider partnering with the Open Bank Project to run hackathons or enable accelerators/ innovation labs.
- Use participation in open platforms as an opportunity to identify and recruit development talent.

Business Impact: The Open Bank Project can enable developers to write an app once and have it used by any bank. This enables open innovation, easier integration, and the ability to monitor and monetize APIs.

Benefit Rating: Moderate

Market Penetration: Less than 1% of target audience

Maturity: Embryonic

Recommended Reading: "Govern Your Services and Manage Your APIs With Application Services Governance"

Open Bank Systems

Analysis By: Mary Knox; Don Free

Definition: An open bank system's assets and functions are accessible and usable in the context of the end user across intra- and interenterprise boundaries. They are also capable of incorporating functionality and data from many sources spanning internal and external ecosystems. Open bank systems are characterized by adherence to service-oriented architecture principles, the exposure of APIs, the ability to consume APIs, incorporation of workflow and business rules capabilities, and supporting technologies and organizational structures and practices.

Position and Adoption Speed Justification: The position of open bank systems is based on the achievement of accessibility and usability in the context of the end user. While some aspects of bank systems, such as componentization, or incorporation of business rules or workflow, may be farther along, the achievement of open bank systems is emerging, with a minimum of eight years expected before they reach the plateau. This is because most banks and vendors are saddled with legacy systems, including core systems with long replacement cycles; the need for new organizational models and practices; and immaturity of best practices and industry standard approaches to the creation and operation of open bank systems. Work across banks in industry associations such as Banking Industry Architecture Network (BIAN) help to accelerate open bank system achievement, and startup banks and vendors unsaddled with legacy may achieve open bank systems in a shorter time frame.

User Advice: CIOs:

- Assess the level of openness of bank systems required to support your bank's digital banking and other business strategies.
- Identify the audiences (such as internal bank IT staff, internal bank business staff, third-party developers or customers) for open bank systems, and the associated requirements.
- Use the openness assessment and audience identification to pinpoint weaknesses in your current systems for the support of digital banking, prioritize investments, and evaluate upcoming projects and vendor solutions.
- Don't confuse componentization with openness. Debunk vendor hype by realizing that just because a system is componentized doesn't mean it is open. Accessibility must be granted to the components and the functionality, and data they contain in a manner that is suited to the skill level (business as well as technical) and intent of the end user.

Business Impact: Open bank systems are essential for supporting digital banking strategies that rely, in part, on the ability to expose bank functionality and information to end users — including customers and counterparties — through APIs and apps. They also require the ability to present customers and other third parties with personalized services. Bank system openness also is important for meeting a bank's internal operational goals. The degree of bank system openness will significantly impact the ability to introduce differentiating customization, while ensuring the ability to accept future upgrades and reducing replacement risk if switching solutions. Similarly, open bank systems support "assembly" approaches to application design that blend functionality from multiple vendor solutions, including public, community and managed cloud services, and in-house development including private cloud.

Benefit Rating: Transformational

Market Penetration: Less than 1% of target audience

Maturity: Emerging

Recommended Reading: "Questions to Ask Vendors When Evaluating Bank System Openness"

"Digital Banking Requires Open Bank Systems"

"Market Guide for Financial Messaging Platforms for Banks and Other Financial Services Firms"

"Critical Capabilities for International Retail Core Banking"

OpenID Connect

Analysis By: Gregg Kreizman

Definition: OpenID Connect (OIDC) is an identity layer on top of the OAuth 2.0 protocol. It enables clients to verify the identity of an end user, based on the authentication performed by an authorization server, as well as to obtain basic profile information about the end user in an interoperable and REST-like manner.

Position and Adoption Speed Justification: Four of OIDC's component specifications, including the core specification, were ratified by the OpenID Foundation in February 2014. Despite having the slightly more mature Internet Engineering Task Force (IETF) draft standard OAuth 2.0 as its underpinning, OIDC is in the early stages of development and adoption. OIDC's promise is that it will meet the needs of organizations looking to protect APIs. It is also designed to provide service discovery and session management using a more lightweight RESTful approach than XML-based standards, such as SAML and WS-Federation. RESTful approaches are increasingly favored by developers, because REST uses familiar HTTP GET, PUT and POST operations, and its encoding scheme is less verbose.

OIDC will improve security, compared with earlier versions of OpenID, and OIDC includes functional specifications to gain users' consent before allowing access to their resources. Several component specifications are in draft or in implementer's draft stages of development. These include session management and native app agent specifications. The native app specification should provide a standardized approach for mobile app developers to incorporate identity security.

OIDC is beginning to replace earlier versions of OpenID. Partial implementations of the standard have been added by several access management vendors, and prominent SaaS platform and application developers such as Google, Salesforce and Microsoft have implementations. Proven interoperable product implementations are relatively scarce.

User Advice: Most enterprises should take an opportunistic approach to implementing OIDC for proofs of concept (POCs) and closed implementations. OIDC will be best for use cases in which organizations use third-party identity providers that support the standard by default, when identity and access management as a service (IDaaS) vendors support the standard, and when the implementation meets organizational security requirements.

Business Impact: OIDC will provide users with convenience and control. It will reduce the data entry burden on users registering and revisiting service providers. If interoperable implementations increase, then enterprises should be able to conduct identity interactions more seamlessly and with less friction for developers than today's XML-based standards or purely proprietary implementations, and with greater security than earlier versions of OpenID.

Benefit Rating: Moderate

Market Penetration: Less than 1% of target audience

Maturity: Emerging

Sample Vendors: CA Technologies; ForgeRock; Google; Microsoft; Ping Identity; Salesforce

Recommended Reading: "Magic Quadrant for Identity and Access Management as a Service"

"Decision Point for Federated Identity and Cross-Domain Single Sign-On"

Banking App Stores

Analysis By: Kristin R. Moyer

Definition: Banking app stores are similar to public app stores (such as iTunes and Google Play), but are deployed and branded by the bank. Like the public stores, banking app stores support app discovery, downloads and collaboration between customers and developers through a local storefront client or browser, and include ratings and comments.

Position and Adoption Speed Justification: Banks use app stores in at least four different ways:

- Public stores (for example, iTunes, Google Play, BlackBerry World and Windows Store) — Banks and nonbanks use public stores to deploy mobile apps they have built for banking customers.
- Vendor app stores — Cloud-based or on-premises stores that are sponsored and branded by a banking technology vendor to enable licensed customers to discover and download banking apps.
- Enterprise app stores for employees — Private, cloud-based or on-premises stores enable banks to deploy apps for employees and partners.
- Banking app stores for customers — Cloud-based or on-premises stores that are sponsored and branded by a bank and enable customers to discover and download banking apps.

Banking app stores are usually provided by a bank that has deployed public Web APIs and software development kits (SDKs) for third-party developers to custom-build apps. Examples of banking app stores include Crédit Agricole (CA Store), Deutsche Bank (Autobahn App Market) and Commonwealth Bank of Australia (App Catalogue).

The focus of this Hype Cycle entry is banking app stores for customers. Banking app visibility in public stores is decreasing due to rapid growth in the total number of apps available (see "Where's That App? The Rise of Banking App Stores"). Bank websites often make it difficult for customers to find and discover apps, because they are usually listed by line of business rather than in a central location. Banking app stores can address these and other challenges, but add cost and complexity that may not always be warranted.

The two biggest drivers for a banking app store are the total number of apps and the presence (or lack) of an API platform (which usually drives app volume). The more apps a bank has deployed, the more challenging it becomes for users to discover the right app for them. Public Web APIs deployed to third-party developers, customers, partners, employees and others will serve as a multiplier to the number of apps deployed by the bank itself. However, cost, additional complexity and operational risk will serve as barriers to the deployment of banking app stores. In addition, the deployment of banking app stores has slowed and not gone beyond a handful of early adopters. We, therefore, believe banking app stores will not reach the Plateau of Productivity before five to 10 years (we originally projected two to five years).

User Advice:

- Evaluate the potential need for a banking app store to improve app discovery, user experience and collaboration.
- As an alternative to implementing a banking app store, improve website organization to support app discovery and user experience. Provide a central location for apps and highlight them on the bank's home page.
- Continue to make banking apps available through public app stores. It's important to be where customers are, and many will continue to discover apps through the public stores.

Business Impact: Banking app stores will enable easier app discovery, a better user experience, bank control (for example, to test apps for compliance and security before posting to the store, and to quickly remove an app from the store when necessary) and co-creation/crowdsourcing. Bank-branded app stores can trigger a wave of innovation by creating a new, highly competitive market for banking apps. However, banking app stores are not right for all banks. A relatively small number of apps (less than five), cost, additional complexity and operational risk are all reasons not to deploy a banking app store.

Benefit Rating: High

Market Penetration: Less than 1% of target audience

Maturity: Emerging

Sample Vendors: AirWatch; Apperian; BMC Software; Citrix; Good Technology; MobileIron; Symantec

Recommended Reading: "Where's That App? The Rise of Banking App Stores"

Open Banking Strategy

Analysis By: Kristin R. Moyer

Definition: An open banking strategy includes the vision, mission, goals, governance, incentives, value network, priorities, approaches, skills and tools for achieving business value (for example, new user interfaces and extending product accessibility) through sustainable Web API programs, apps and app stores.

Position and Adoption Speed Justification: The banking industry has been adopting private and public Web APIs rapidly during the past 12 months, and some early adopters have launched their own banking app stores. However, the industry has generally failed to develop a clear, enterprisewide business strategy for open banking. The focus has primarily been on technology — for example, how to deploy a gateway (build or buy); whether to use REST, XML or SOAP; and the pros and cons of OAuth 2.0. These types of technology issues, and APIs themselves, are currently much more mature than a clearly defined open banking strategy. The identification of specific goals and performance metrics is rare.

The problem with this approach is that it is not tied to specific goals and performance metrics, vision, and incentives regarding why internal or external users would want to participate, or a roadmap that includes priorities and actions required to achieve goals or how value will be created and captured. Other hidden risks have yet to be addressed as well — for example, intellectual property and impact on reputation risk. In addition, open banking has so far mainly been additive to already complex bank architectures, infrastructures and operations. This lack of alignment with overall architectural and business goals is a problem.

The risk with open banking is that CIOs and line-of-business leaders will build something that fails to gain traction or improve net profits — wasting precious IT budgets and hurting the bank's reputation. We believe that creating a strategic direction for open banking is an urgent imperative. We expect to see some visible failures for banks that have launched open banking initiatives without an open banking strategy that is in alignment with broader enterprise architectural, application and enterprise strategies. We would define these as open banking initiatives that are launched and then later retracted due to a perceived lack of results or potentially due to security problems or system outages. We, therefore, believe that the creation of a well-defined and articulated open banking strategy will take two to five years to reach the Plateau of Productivity.

User Advice:

- Identify the mission and goals of an open banking initiative. Identify what will be achieved, as well as specific goals and performance metrics. Examples of what will be achieved may include things like increasing revenue through expanding product accessibility, reducing the time and cost to build new mobile apps, and harnessing the power of open innovation and crowdsourcing. Examples of goals might be to increase net profits by 1% to 2% within two years, the creation of a developer community consisting of 1,000 people within 12 months, and a reduction of internal mobile app development and maintenance costs by 50% within 24 months.
- Develop vision and incentives for why retail or commercial banking customers, third-party partners, citizen developers, bank employees, or even other banks would want to leverage your open banking capabilities. An example of a vision for open banking might be to shift power from the bank to the individual to deliver needs-based financial services that increase net profits.
- Define the value network and ecosystem, and with whom value will be created and captured. Examples of who will create the value may be partners or a third-party developer community. Examples of how the value will be captured may include things like hackathons and an app store for retail and commercial customers to access apps built by third-party developers.
- Create an open banking strategy at the enterprise level, and align it with architecture, infrastructure and operations innovation. Siloed approaches to open banking will result in increased costs, operational risk and an inconsistent experience for customers.

Business Impact: CIOs will fail to deliver noticeable benefit from the use of private and public APIs without an open banking strategy. It will help CIOs and line-of-business leaders reduce risk and align open banking strategies with enterprise architectures, infrastructures and operations innovation.

Benefit Rating: High

Market Penetration: Less than 1% of target audience

Maturity: Emerging

Recommended Reading: "Introducing the Gartner Business Model Framework"

"The Business Value of Banking APIs"

"How to Make APIs a Banking CEO and Board Priority"

Open Business Model

Analysis By: Kristin R. Moyer

Definition: Open business models leverage internal and external users to support ideate, create, engage, offer, monetize and adapt. The bank provides the business model; business model users can be internal or external. Open business models are monetized through traditional value exchange and global, decentralized digital representations of value.

Position and Adoption Speed Justification: A business model for open banking must be created to deliver open banking strategies that drive financial results. However, business models for open banking are currently at a much earlier stage of adoption than the deployment of open banking technologies, such as public Web APIs, hackathons and app stores. The state of the business model is a matter that urgently needs to be addressed for open banking strategies to be successful. We believe that CIOs and business leaders will address strategic direction first, but may continue to fail to address business model innovation, given how difficult it is to achieve this level of change and transformation within banks (especially large banks). Therefore, we believe that business models for open banking are still five to 10 years from the Plateau of Productivity.

We have seen a lot of focus on specific parts of the business model (e.g., traditional IT in the "create" area; many industry innovations in the "offering"), but it is important to consider all aspects of the business model to maximize the opportunities, minimize the risks and create a coherent model. The risk is that CIOs will build something that fails to gain traction or improve net profits — wasting precious IT budgets and hurting a bank's reputation.

User Advice:

- Leverage the business model framework (see "Introducing the Gartner Business Model Framework") while creating strategic direction for open banking, and *before* deploying an open banking strategy.
- Articulate the business value of open banking to CEOs and boards in terms of revenue growth, cost efficiency, user experience and talent management. Use a business model for open banking as the basis for an ongoing CEO dialogue regarding the business opportunities inherent in open banking strategies and technologies, and how you will capture them.

- Exploit lateral innovation — use business model analogies that can be used to "steal" innovation from other industries.

Business Impact: Creating a business model for open banking is an urgent imperative for CIOs, IT leaders, digital marketing executives and line-of-business executives who have launched, or will launch, an open banking initiative to fully realize business value. Failure to do so will result in limited traction with the value network, and, therefore, dramatically increase the risk of wasted IT budgets, negative impact on net profits, reputation risk and customer trust.

Benefit Rating: High

Market Penetration: Less than 1% of target audience

Maturity: Embryonic

Recommended Reading: "Introducing the Gartner Business Model Framework"

"Business Model Innovation: Unleashing Digital Value Everywhere"

"The Business Value of Banking APIs"

Software-Defined Application Services

Analysis By: Anne Thomas; Yefim V. Natis

Definition: Software-defined architecture (SDA) is a design pattern in which IT resources are virtualized, managed, protected and enriched by deploying a software mediation layer between consumers and the resources. Software-defined application services (SDAS) apply this pattern to application services and their APIs. The SDAS mediation layer (a service control gateway) can inject numerous capabilities into the interaction for increased agility, usability, performance and control of application services.

Position and Adoption Speed Justification: SDA is a mature pattern in a number of technology domains, such as software-defined networking and storage, and it is gaining traction as a general architectural pattern that applies to many contexts that suffer from high complexity and challenging scale. The application services domain is one such context. Many organizations employ API gateways in a tactical way to protect and enrich their externally facing APIs and services. SDAS is a strategic approach for managing all APIs. Gartner believes that in the next five to seven years, most IT organizations will adopt SDAS for both internally and externally facing APIs. As relevant technology vendors recognize the scope of opportunity that SDAS represents, Gartner expects a rapid growth of promotion, investment and hype before mainstream adoption brings it to reality and eventually to its Plateau of Productivity.

User Advice: SDAS can add a host of benefits to both external and internal API interactions: monitoring, scalability, resiliency, security, privacy, compliance, metering, protocol bridging, integration, orchestration, data filtering and transformations, data cleansing, context injection, personalization and more. SDAS also virtualizes APIs and thereby increases the flexibility and agility of service-oriented architecture (SOA) environments. An SDAS gateway encapsulates a service's "inner" API — the API that reflects the service's internal domain model — and can then expose one or more "outer" APIs that more directly suit the needs of the service consumers. SDAS applies to both incoming and outgoing API interactions. This virtualization enables consumers and services to evolve independently.

You can implement an SDAS mediation layer today using technologies such as API managers, API gateways and SOA governance platforms. Enterprise service buses (ESBs) and integration platform as a service (iPaaS) can also be used to implement some of the SDAS gateway capabilities (see "Selecting Technology to Support an SDAS Service Control Gateway"). The gateway layer should be distributed, replicated and federated to avoid creation of performance bottlenecks or a single point of failure. When selecting gateway technologies, give preference to vendors that have a strategic view for the role of this technology in the operations of an IT organization, and that have a roadmap for increasing the scope of their technical capabilities.

Business Impact: SDAS delivers many business benefits: improved agility, scalability, performance, resiliency, security, privacy and regulatory compliance. SDAS enables Web-scale SOA. Introduction of the guiding architectural principles for management of large-scale SOA environments will enable mainstream IT organizations to scale and innovate, while avoiding increased risks to integrity or accountability of their operations. Web-scale digital business in mainstream IT can't materialize without an emerging system of management and control. With it, mainstream businesses can expect higher rates of innovation, easier scaling and advanced cooperation between business, IT, customers and partners of the enterprise.

Benefit Rating: High

Market Penetration: 1% to 5% of target audience

Maturity: Emerging

Sample Vendors: Apigee; Axway (Vordel); CA (Layer 7); IBM; Intel (Mashery); MuleSoft; Torry Harris; WSO2

Recommended Reading: "An Introduction to How Software-Defined Application Services Enable the Apps and Services Architecture"

"Software-Defined Architecture for Applications in Digital Business"

"Evaluate Gateway Capabilities Required to Deploy Software-Defined Architecture for Application Services"

"Selecting Technology to Support an SDAS Service Control Gateway"

Open-Source Banking Systems

Analysis By: Kristin R. Moyer

Definition: Open-source banking systems are banking-specific applications (like card management software or core banking), components (like relationship pricing) and infrastructure (like user interface widgets) that are made available through an open-source license process. Open-source software is available under license and distribution conditions specified by the Open Source Initiative. An open-source license typically permits free use, access to the source code, modification and redistribution, subject to the conditions of the entity distributing it.

Position and Adoption Speed Justification: Open-source banking systems make code available to everyone, and let CIOs take advantage of open innovation, which capitalizes on knowledge and expertise that lie beyond an enterprise's boundaries. Open-source banking systems may be especially appealing in areas of commoditized functionality, components and infrastructure — as is the case with packaged applications relative to homegrown applications. Some examples of open-source banking solutions include Cyclos (online and mobile banking), MyBanco (microfinance), Micro-finance Open Architecture Project (which also includes software, not just standards) and Mifos (core banking and APIs).

We have not seen an increase in the use of open-source banking systems during the past 12 months, and, therefore, have not changed its position on the Hype Cycle. Lack of structured product governance (on the part of the open-source platform), support, standards (impacting interoperability), cost of skills training/maintaining and security vulnerabilities are barriers to adoption. On the supply side, software vendors benefit from the prevalence of the traditional licensing model, which provides them with an annuity stream for customization and support. These factors are likely to delay growth in open-source banking and result in not reaching the Plateau of Productivity for more than 10 years.

User Advice:

- Do not adopt open-source software to execute an open-banking strategy if the primary objective is to tap into third-party developers for the development and deployment of apps to customers. Public Web APIs and software development kits (SDKs) are easier for third-party developers to use, because they will usually include documentation, sample code and other support not available via open-source software.
- Evaluate opportunities to leverage open-source components like microloans or smart cards that may enable faster time to market and lower costs than are involved in evaluating, selecting and implementing a packaged application that includes more functionality than needed.

Business Impact: Open-source banking systems enable banks to have direct customization control over applications, components and infrastructure, as well as leveraging the power of open innovation. In theory, open-source banking systems could reduce the total cost of ownership of applications, components and infrastructure because it is free. The continuing focus on cost reduction on the part of CEOs will create an opportunity for increasing the interest in open-source banking systems.

However, corporations often desire maintenance and support services for open-source software (as is the case for Linux and providers like Red Hat and others), which then reduces the cost benefit of open source and can create the same type of vendor dependence involved with proprietary applications. It also means that there is no vendor that is actually supporting the customizations that have been made, which can increase maintenance costs.

Benefit Rating: Low

Market Penetration: Less than 1% of target audience

Maturity: Embryonic

Sample Vendors: Cyclos; jPOS; Micro-finance Open Architecture Project; Mifos; MyBanco; Open Smart Card Development Platform

API-Enabled Ecosystems

Analysis By: Kristin R. Moyer

Definition: Internal (such as for employees) and external (such as for third-party developers) ecosystems are dynamic communities that extend product and service accessibility, and create new business models by taking full advantage of APIs. They can be formal (such as a long-term partnership that is purposefully formed) or informal (such as a developer participating in a hackathon or a digital firm mashing up a bank API with one of its own APIs).

Position and Adoption Speed Justification: Digital business multiplies complexity for banking CIOs. It has been difficult enough to keep up with building mobile apps for smartphones and tablets. Wearable technology, the Internet of Things, smart machines and autonomous vehicles will increase complexity on an almost exponential scale. There is no way CIOs can deliver against all of these new opportunities themselves. New internal and external ecosystems enabled by APIs through self-service developer portals are needed to create ideas, user interfaces and product/service offerings.

Reaching more broadly across lines of business/internal ecosystems of employees (for example, through an internal hackathon) can enable CIOs to tap into new talent within the enterprise and capture innovative ideas that otherwise may not be realized or even identified. Much of this is business empowerment and making the tools, data and functionality available in a usable fashion. The same is true of external ecosystems of partners and third-party developers. Internal and external ecosystems can help CIOs deliver effectively in the face of complexity. Web APIs are of little use to CIOs without internal and/or external ecosystems that use the APIs to drive business value.

Some banks have been purposeful about creating external ecosystems. For example, BBVA, Citi and Credit Agricole have API events (or "meetups" as Citi calls them) that often include networking, education and feedback on the bank's programs. We have seen an increase in the number of banks purposefully creating internal and/or external ecosystems during the past 12 months, and believe this will reach the Plateau of Productivity within two to five years.

User Advice:

- Create and nurture internal and external ecosystems to enable open innovation. This requires providing strong support and building trust with the ecosystem, not just a good API.
- Experiment with APIs internally before deploying them externally to gain experience and refine approaches.
- Deploy APIs that are interesting and easy to use. Having access to interesting data previously unavailable can sometimes be more important to developers than cash prizes.

Business Impact: Internal and external ecosystems enable a far broader set of relationships, products, services, information and transactions. This translates to faster time to market, the ability to deliver very specific and needs-based solutions (based on the user's need, technology and location), lower cost of development and delivery, and increased acceptance of the product and its price via functional enhancement.

Benefit Rating: High

Market Penetration: Less than 1% of target audience

Maturity: Emerging

Recommended Reading: "Citi Uses Hackathons to Accelerate Digital Innovation"

Retail Banking APIs

Analysis By: Kristin R. Moyer

Definition: Retail banking APIs expose data, business processes and application capabilities through interfaces typically based on Web-oriented architecture (WOA) principles. Retail banking APIs can be public, semiprivate or private, depending on the type of users (such as partners or third-party developers) who will be accessing the APIs.

Position and Adoption Speed Justification: An API is an interface to an application capability that can be used programmatically (see "Choosing an API and SOA Governance Architecture"). A single component or service can expose many interfaces to support different interaction models or protocols. APIs have been used for a long time by bank IT, but, during the past 10 years, the Web incarnation of APIs has emerged and enabled e-commerce, social media, mobile and cloud services.

Four business drivers are leading to the rapid adoption of retail banking Web APIs:

- CEO business priorities — Web APIs enable several of the CEO's top business priorities, including revenue growth, efficiency, customer experience and talent management (see "The Business Value of Banking APIs").
- Digital transformation — Web APIs are a key element of digital transformation because they are a new channel, and they engage new ecosystems of internal and external partners (see "Citi Uses Hackathons to Accelerate Digital Innovation").

- Competitive pressure — Financial technology firms (fintechs) are using APIs aggressively to enable digital transformation (see "Anyone Can Build a Bank"), and a number of licensed/regulated banks are beginning to use them in similar ways.
- Government/regulatory body standards — Several government and regulatory bodies have created, or are in the process of creating, banking API standards or gateways (see "How to Make APIs a Banking CEO and Board Priority").

Barriers to the adoption of retail banking APIs are still very real at many banks, however. Many boards and CEOs in the banking industry are concerned about using APIs (especially externally) due to security and regulatory compliance concerns. Those concerns are valid and reasonable, but can be addressed by proper API usage policies and application infrastructure investments. If you build it (a retail banking API), they (users and developers) may not come. For example, Axa Banque launched a retail banking public Web API and was unimpressed with the results (see www.economistinsights.com/analysis/future-factors).

Despite these barriers, we expect to see public Web APIs reach the Plateau of Productivity within two to five years.

User Advice:

- Provide retail banking APIs to quickly pursue new revenue opportunities (for example, social commerce), provide deeper localization specificity, facilitate market expansion, harness the power of open innovation and create a "stickier" customer relationship.
- Don't assume that, if you build it, they will come. Establishing traction for a public Web API requires a clear strategic direction, business model and application service governance (including API management). Additionally, it requires creating internal and external developer awareness and making the API easy to consume via a self-service developer portal.
- Before releasing your Web API to the outside world, consider the traffic and resource implications for your data center. Complement your infrastructure with Web API management products and cloud services as required, and ensure that you have an adequate funding model in place for its ongoing operation.
- Put in place application service governance, including API management, prior to launching an open banking strategy.

Business Impact: Retail banking APIs can provide the foundation for a vibrant ecosystem of employees, business partners and independent developers applying their own ingenuity and effort to expand the use of the bank's services and data. As a result, IT organizations can become central players in helping their organizations expand top-line growth, improve service delivery, drive competitive advantage and reimagine the banking industry.

Benefit Rating: Transformational

Market Penetration: 1% to 5% of target audience

Maturity: Emerging

Sample Vendors: Apigee; Axway; CA Technologies; Mashery; MuleSoft

Recommended Reading: "The Business Value of Banking APIs"

"How to Make APIs a Banking CEO and Board Priority"

At the Peak

Internal Hackathons

Analysis By: Kristin R. Moyer

Definition: Internal hackathons are contests during which bank employees develop and showcase their ideas. Internal hackathons can be focused on creating new ideas, but they can also involve employee developers using APIs and other tools and services to develop prototypes, apps and data visualization.

Position and Adoption Speed Justification: The purpose of an internal hackathon is for employees to develop new ideas and prototypes. One way banks are beginning to use internal hackathons is to experiment with using already-available internal Web APIs. Insight from these hackathons will enable banks to design and deploy Web APIs that are more tailored to internal developer personas and that are easier to use. Another way banks are beginning to use internal hackathons is to pilot and test Web APIs that they plan to deploy to public users, such as third-party developers or partners. Some banks are using internal hackathons to develop new apps more quickly and less expensively than would be the case with typical bank software development life cycles. For example, Citi ran several internal hackathons before launching the Citi Mobile Challenge, a global hackathon program that aims to accelerate digital banking innovation by bringing together developers and designers to create financial technology solutions.

Internal hackathons should be run differently than external hackathons. Banks usually have a pretty diverse group of employees (some people have kids, long commutes, etc.), and may decide to not allow coding after hours. This way, everyone would have an equal chance to participate and win the event. Providing employees with some unstructured time to hack could be well worth the time and may improve team bonding.

We believe that internal hackathons will continue to increase in number, therefore reaching the Plateau of Productivity within two years. Organizers will need to ensure clear goals and financial benefits for participants, or risk lackluster attendance and participation.

User Advice:

- Build fun projects that enable employee developers to compete against each other (similar to public hackathons).
- Give employee developers time to hack for competitive hackathons and, if possible, to drive innovation beyond just hackathons.
- Make team bonding one of the objectives of the hackathon.

- Use internal hackathons to quickly stimulate new ideas, bring new apps to market more quickly and cost-effectively and experiment with APIs.

Business Impact: Internal hackathons can be used to test existing or new APIs before going public, and enable faster time to market and lower costs for new apps. They can also get the entire staff thinking differently and more deeply about digitally enabled business opportunities.

Benefit Rating: Moderate

Market Penetration: 5% to 20% of target audience

Maturity: Adolescent

Recommended Reading: "Citi Uses Hackathons to Accelerate Digital Innovation"

API Marketplaces

Analysis By: Paolo Malinverno

Definition: An API marketplace is a portal where API providers publish APIs for developer communities. Developers can use an API marketplace to discover, access, test, sign up for and embed APIs in their new Web or mobile applications. These APIs are public or semiprivate (typically limited to business partners in a B2B community) and can be consumed outside the firewall of a company.

Position and Adoption Speed Justification: API marketplaces can be implemented via the developers' portal functionality offered in an API management platform. Because the developers using the marketplace are typically outside the company that provides the APIs, the developers' portals are typically in the cloud (the APIs themselves can be on-premises or in the cloud). API marketplaces enable API providers to distribute Web APIs and developers to include published Web APIs in the applications they are implementing.

The number of public Web APIs is still rising very steadily, and shows no signs of slowing down. To the contrary, as more and more companies realize the value of the data they hold, and the functionality offered by their own internal systems, they will try to promote the Web APIs they publish through an API marketplace. We have just seen the beginning of this phenomenon, which will be further fueled by the needs of digital business applications and the Internet of Things.

User Advice:

- **For developers:** Expect API marketplaces to proliferate, and the number of Web APIs available in the current, more established API marketplaces to increase steadily. Exploit "hackathons" and your social connections to find the APIs you need, and choose the API marketplace that most closely matches your development style and needs: The uniformity of access to APIs within that marketplace, and the consistency of the documentation offered for each of the APIs, will rule your loyalty (or lack thereof) to that marketplace.

- **For API providers:** Evaluate the business opportunities a public Web API could offer as a new channel of distribution for your products, and as secured and disciplined external access to the functionality in your internal systems or your data, by making it available through an API marketplace. Your choice of which one will depend on the number and profile of its users, whether those users match the developer communities your API program wants to target, and the real business value your API efforts are ultimately targeted to deliver.

Business Impact: Web APIs that are made available for public reuse can be provided for free or monetized through basic or premium-type plans, depending on the type of API provided and its role within the business model the provider wants to apply. Developers use API marketplaces to discover, access, test, sign up for and include Web APIs that are available for use, according to the policies set by the API provider. API marketplaces earn revenue by hosting the developers' portal functionality on behalf of the API provider. Frequently — especially when the functionality or the data provided by the API is sensitive or of high value — developers have to go through some process of authentication and screening before they can test or use the API, which is enforced through policies in API management platforms. Further screening might happen just before the final app/application is made available on an app store, which in turn might be public or hosted by the API provider.

Benefit Rating: Moderate

Market Penetration: 5% to 20% of target audience

Maturity: Emerging

Sample Vendors: 3scale; Apigee; GitHub; Intel (Mashery); Mashape; MuleSoft

Recommended Reading: "Extend the Enterprise With Public Web APIs"

"Basic API Management Will Grow Into Application Services Governance"

"A Guidance Framework for Designing a Great Web API"

"Govern Your Services and Manage Your APIs With Application Services Governance"

"Magic Quadrant for Application Services Governance"

Bank Employee Mobile Apps

Analysis By: Stessa B Cohen; Kristin R. Moyer

Definition: A bank employee mobile app is a software packaging construct designed for bank employees to serve bank customers or operate specific bank processes using a mobile device. These activities may include handling an in-branch query, onboarding a client in the field or collaborating with a customer in creating an investment portfolio. Employee mobile apps should be narrowly focused on doing one complete thing well.

Position and Adoption Speed Justification: Bank CIOs are deploying mobile apps to employees on smartphones and tablets, using native, Web and social media apps in a multitude of ways to enable employees to work "any place, any time" with customers. In particular, bank staff are using these apps in the areas of communication, creation, information and productivity (see "What Consumer Apps Mean for Workplace Engagement Strategies"). Banking-specific employee apps have bank-business-specific functionality and are proprietary to the bank or are provided by a vendor or third-party developer that enables banking-specific business processes. Employee apps are used to interact with customers (whether consumer, wealth, private, investment, business or corporate) at some point during the financial services supply chain — that is, employee apps enable bank employees to work with customers face to face or behind the scenes to perform a task or transaction that affects customers in some way. The apps can be used at a branch, in the field (as the ICICI example), with customers or not. These apps do not do corporate work such as IT, HR or finance.

We expect employee apps to reach the Plateau of Productivity within two to five years (closer to five years) as CIOs more aggressively seek to enable branch transformation, increase operational efficiencies and sales productivity, reach the unbanked and improve general employee productivity.

User Advice:

- Create an app roadmap for employee apps that enable sales, service, communication, creation, information and productivity. For example, an app for underwriters could decrease time to closing by reducing bottlenecks.
- Deploy employee apps for relationship managers, loan officers, branch sales employees and others that go into the field and have face-to-face interactions with customers. Apps can provide a more compelling and efficient user experience and reduce sales cycles, especially via tablets with larger screen sizes that enable customer collaboration and interaction. Banks must seek to do more than just digitalize existing processes and desktop applications, and leverage the capabilities of tablets and mobile devices for employees.
- Design an appropriate app deployment strategy that could include enterprise app stores, managed apps or others.
- Align the employee app strategy with the bank's tablet and mobile strategies for customers. Apps and distribution will be impacted by bring your own device and other policies.

Business Impact: Employee apps improve efficiency, communication, collaboration, access to information and presentation with face-to-face customer experiences. For example, employee apps can enable branch transformation by eliminating teller counters and moving to bank employees with tablets that enable onboarding and other sales activities. As another example, loan officers often go into the field with a rolling suitcase that includes paper brochures, a laptop, scanner, cables and other equipment to sell and originate a loan. An employee app accessed through a tablet can eliminate all this equipment when coupled with other technologies, such as image capture and electronic signatures. It can also shorten sales cycles and reduce errors due to manual data entry.

Benefit Rating: High

Market Penetration: 5% to 20% of target audience

Maturity: Adolescent

Sample Vendors: CMC; Epam Systems; FIS; Lexmark; Mitek; Newgen Software; Nucleus Software Exports; Top Image Systems

Recommended Reading: "What Consumer Apps Mean for Workplace Engagement Strategies"

Application Services Governance

Analysis By: Paolo Malinverno

Definition: Application services governance is about the planning, design, implementation, publication, operation, consumption, maintenance and retirement of APIs and services within an application construct. It includes a developer's portal to target, assist and govern communities of developers who embed the APIs and services in the application constructs they are implementing.

Position and Adoption Speed Justification: The market has changed fundamentally in the last two years. Projects have become smaller, focused on very few APIs at one time, more business-oriented, in need of very quick execution and with generally simpler interactions, frequently driven by mobile apps. Buying centers are shifting rapidly from IT departments to business units, anticipating a general trend. Application rationalization, the API economy, bimodal IT, the evolving notion of applications/apps and changing API consumers (both human and electronic) will significantly drive the application portfolio of the future, and APIs are right in the middle of all that.

The current positioning of the application services governance technology profile (and its maturity) reflects the irresistible surge of the API economy, the hype around which still has a lot to climb before reaching the Peak of Inflated Expectations, and the maturity of SOA endeavors, which are square on the Plateau of Productivity.

User Advice: Application services governance is the functionality organizations need to run successful API programs, and thrive in the API economy. API programs and initiatives don't necessarily need the full set of application services governance functionality from the start, but their new functional requirements come by very rapidly (and will need to be addressed very quickly) as they evolve — so we recommend that CIOs, IT leaders and business executives in charge of API programs consider offerings that have the potential to address their needs well beyond the first project or two. Extending the current platform will be much easier and less expensive than replacing it.

Application services governance is your insurance policy to get value out of your APIs and services. Without it, you won't even be able to measure what value you and your company are getting out of them (if anything). Application services governance also plays a fundamental role in XaaS (mainly PaaS, iPaaS and MBaaS), and in nascent application design patterns like software-defined architecture (SDA), where an IT capability is virtualized by exposing a set of virtual ("outer") APIs that decouple the implementation of the capability from its consumers.

Acquisitions in this space will continue to be commonplace, as larger players will want to buy an initial share in a rapidly growing market, and use API management as part of their larger XaaS offerings.

Business Impact: One of the taglines of Gartner Symposium keynotes is "Every company is a technology company." APIs make this even more true. Starting in media and high tech, APIs have now moved into financial services, government and retail. During the past two years, a critical mass of companies and government institutions has been publishing APIs into their data and some functions of their IT systems to fuel business-to-consumer (B2C) innovation, enable mobile apps and take advantage of more-direct B2B interactions with their business partners. This will continue to fuel the so-called "API economy." However, that's not all; in fact, the best is still to come.

APIs are increasingly key to application strategies and effective application governance. However, most future API use will come from digital business applications and things (as in the Internet of Things), and APIs will be firmly tied to business scenarios.

Benefit Rating: High

Market Penetration: 5% to 20% of target audience

Maturity: Adolescent

Sample Vendors: 3scale; Akana; Apigee; Axway (Vordel); CA Technologies (Layer 7); Cloud Elements; Dell Boomi; IBM; Intel (Mashery); MuleSoft; SAP; Software AG; TIBCO Software; Torry Harris Business Solutions; WSO2

Recommended Reading: "It's Time to Gradually Give Up Developing, Buying and Maintaining Applications"

"Use the Demand for New Applications to Drive Application Rationalization"

"Which New and Old Applications Will Enable Digital Business?"

"Magic Quadrant for Application Services Governance"

"Choosing an API and SOA Governance Architecture"

Internal APIs

Analysis By: Kristin R. Moyer

Definition: Internal Web APIs expose data, business processes and application capabilities through interfaces that are typically based on Web-oriented architecture principles. Internal APIs in this context are for use inside the bank.

Position and Adoption Speed Justification: Banks have a substantial amount of data and application capabilities that are proprietary, and cannot or should not be shared externally. Some Web APIs can be used internally and externally, but others (like customer spending patterns down to the individual level) are more appropriate for internal use only.

Most banking CIOs have hundreds or even thousands of APIs within various silos of their organizations, whether or not they are aware of them. However, very few of those APIs will be helpful in creating new value as part of digital transformation because most were not built using modern Web API practices for modern customer use cases or for use outside the development group they were implemented in (see "A Guidance Framework for Designing a Great Web API"). It is possible that some existing APIs will be suitable, but demand should drive internal API deployment — not supply (for example, we have x number of APIs; does anyone want to play with them?). Ease of use for the developer is also a defining characteristic of modern Web APIs, which is in contrast to APIs that are more focused on applications and integration (and are, therefore, not easy for developers to use).

The use of internal Web APIs has increased rapidly during the past 12 months. Many banks that have started their Web API journeys during the past 12 months began with an internal focus. For example, one large global bank is in the process of deploying an API layer across its country operations to enable faster and less expensive mobile app development. Some banks are running internal hackathons using existing APIs. This can be a good way to gain experience with pushing APIs out to users, and identifying potential developer personas and scenarios.

Despite the benefits of APIs, there are barriers. CIOs and their counterparts will be concerned with operational issues like security, performance and control of what information is available. Although these are issues that a control plane and policies can address, they require associated technologies (such as an API management gateway) and application services governance — all of which take time and money to deploy. We believe that it will take less than two years for internal Web APIs to reach the Plateau of Productivity.

User Advice:

- Create a strategy for internal API use.
- Use a demand-driven approach to drive Web API design.
- Run an internal hackathon to gain experience with pushing APIs out to users.

Business Impact: Internal APIs can increase innovation by making data, business processes and application capabilities more accessible to bank employees. This can better enable mobility, extend product accessibility, and create new business models and revenue opportunities. Internal APIs can also reduce cost and time to market by pushing control to users closer to the end customer, without needing to wait for IT to have the bandwidth to execute.

Benefit Rating: Transformational

Market Penetration: 5% to 20% of target audience

Maturity: Adolescent

Recommended Reading: "Reference Model for Open Banking APIs, Apps and App Stores"

"Designing a Great Web API"

"Digital Banking Requires Open Bank Systems"

Commercial Banking Apps

Analysis By: Kristin R. Moyer; Fabio Chesini; Stessa B Cohen

Definition: A commercial banking app is a software packaging construct where value is derived from purposefulness. The more narrowly focused an app is on doing one thing well, the more valuable it's deemed to be. This contrasts with the traditional "application" packaging construct, where value is tied to capabilities — the more things it does, the better it's perceived to be.

Position and Adoption Speed Justification: The corporate banking environment has consistently lagged behind the technological evolution in retail banking. Historically, corporate banking has been heavily dependent on personal human interaction and noncommoditized, brand-related services and relationships. However, because business is increasingly being transformed by the development of digital capabilities, long-standing preconceptions about the corporate banking business are being questioned (see "Survey Analysis: Digitalizing Corporate Banking"). Technology — particularly software — is becoming a critical differentiator, not just in the context of transactional activities — such as electronic funds transfer or foreign exchange execution. Banks are now being associated with their technology innovations and the way they use those innovations to maintain brand presence and reinforce customer loyalty. Unlike retail banking, CIOs cannot merely reorient their technology strategy to offer commoditized apps, mobile banking and mobile payment services. For corporate banking, technology — especially digital technology — should be seen more as an augmenter of human interactions in the banking relationship, as opposed to a replacement. The provisioning of mobile apps is the second-most important technology design and selection criteria.

The first generation of commercial mobile banking apps was focused on replicating a broad range of online banking functionality to a mobile device — in most cases, smartphones first and then tablets. While this extensive functionality is somewhat manageable with a tablet, it is too broad for a smartphone app and does not meet the true definition of an app (for example, narrow, focused functionality). We believe the next phase of commercial banking app maturity will embrace minimalist design and more narrow functionality for smartphones. For example, the Deutsche Bank Autobahn App Market has more than 150 apps (see "Where's That App? The Rise of Banking App Stores").

We believe the next phase of app maturity will also include apps built specifically for tablets with more-detailed information and functionality than is appropriate for smartphones. Given the rapid growth of banking apps, we expect to see commercial banking apps move to the Plateau of Productivity (and off the Hype Cycle) in less than two years.

User Advice:

- Streamline existing smartphone mobile banking apps by eliminating functionality that is rarely used.
- Create apps for specific business user roles, such as accounts receivable and accounts payable.

- Build apps that address businesses not currently served by traditional business banking.
- Identify ways that fintechs are disrupting commercial banking, for example providing simple, attractive apps with very specific use cases.
- Deploy narrowly focused apps that digitalize existing processes for frequently used smartphone mobile banking functionalities.
- Deploy more-full-functioning apps specifically for tablets, rather than one app that can be used on smartphones and tablets (which is what much of the industry does today).
- Use API platforms and third-party developers or partners to meet the long tail of app development needs.
- Retool the application development process so that more time is spent identifying and refining personas and meeting unmet needs than is spent asking for feature requests.

Business Impact: Commercial banking apps have a high business impact because they enable banks to deliver services that revolve around the digital life of the customer. For example, it is largely through apps that banks will deliver need-based services based on the context of the user (for example, the user's need, technology and location). Apps enable faster time to market and lower costs, given their narrow focus, and are, therefore, disposable. This is in contrast to banking applications that require months to years to implement and are, therefore, often used for five to seven years or more (in some cases, for decades).

Benefit Rating: High

Market Penetration: 5% to 20% of target audience

Maturity: Adolescent

Sample Vendors: Comarch; Crealogix Group; Finanteq; Newgen Software; Technisys

Recommended Reading: "Survey Analysis: Digitalizing Corporate Banking"

OAuth

Analysis By: Gregg Kreizman

Definition: The OAuth authorization framework orchestrates an approval interaction between an end user and the application, thus making it possible to grant third-party applications limited access to resources on behalf of end users. The OAuth authorization framework separates the access request from the resource being requested, thereby allowing an independent authorization service to manage the use of credentials on behalf of the client and the resource.

Position and Adoption Speed Justification: The OAuth specification originally was developed to enable users to grant others access to the users' Web-based information resources, without the users being required to divulge passwords (the sharing of password-protected photographs was an early example). However, OAuth has evolved to support many more use cases. This authorization framework is needed to support the increasing requirements for multiparty interactions and the use of native mobile applications. OAuth 2.0 allows for any HTTP-based authentication protocol to be used.

OAuth is an Internet Engineering Task Force (IETF) Internet draft specification. OAuth has considerable support from and adoption by large Internet technology providers, such as Facebook, Google, Microsoft, Salesforce and Yahoo. Cloud vendors use OAuth heavily for enabling service-to-service interactions. Also, identity and access management (IAM) software and IAM as a service (IDaaS) providers are increasingly implementing OAuth support in their products and services. In addition, enterprise adoption of OAuth for business-to-consumer (B2C) applications is increasing. While we believe that business-to-employee (B2E) adoption remains relatively low, due to investment in established legacy standards and proprietary authorization technologies, we are seeing OAuth being used to protect new RESTful APIs.

Gartner has observed limited implementations of OAuth to support reduced sign-on with mobile-device-resident application clients. Native apps (NAPPS) OpenID Connect profiles of OAuth eventually may replace the need to independently manage OAuth flows for native apps single sign-on (SSO).

User Advice: Enterprises that have consumer-facing applications or are developing Web services should evaluate OAuth 2.0 as a means of abstracting authorization. These enterprises also should query applicable IAM providers to determine where they are in executing their OAuth roadmaps.

Business Impact: Applications that invoke embedded browsers can use proprietary authentication methods and SAML federation protocols. However, as native apps continue to make inroads into different mobile devices and OSs — each with its own access methods — enterprise users and consumers may be faced with an ever-growing number of stand-alone proprietary access methods. Standard authentication and authorization capabilities could help enterprises avoid lock-in for any given access method. OAuth will be relevant as a method of supporting authorization for consumer-facing and partner-facing applications, and it may be particularly suitable for developing native mobile apps. Within one to three years, adoption of the OpenID Connect profile of OAuth should help to ensure more consistent implementations with security baked in.

Benefit Rating: Moderate

Market Penetration: 5% to 20% of target audience

Maturity: Adolescent

Sample Vendors: CA Technologies; ForgeRock; Gigya; IBM; Janrain; Micro Focus (NetIQ); Okta; Oracle; Ping Identity; SecureAuth

Recommended Reading: "Magic Quadrant for Identity and Access Management as a Service, Worldwide"

"Decision Point for Federated Identity and Cross-Domain Single Sign-On"

Public Web APIs

Analysis By: Paolo Malinverno

Definition: Public Web APIs expose data, business processes and application capabilities to be used in innovative application constructs. Enterprises use public Web APIs to expand their reach through new delivery channels, such as third-party websites and mobile apps, as well as for quicker and simpler B2B connectivity. A number of business models for API monetization are in use, both direct and indirect.

Position and Adoption Speed Justification: By 2017, more than 80% of large global enterprises will offer public Web APIs. Web APIs are required for mobile and social applications, as well as for digital business initiatives, integration with the Internet of Things (IoT), and innovative forms of e-commerce that combine Web services and contextual information to deliver a compelling user experience.

By 2018, more than 50% of B2B collaboration will take place through Web APIs. Although a prominent majority of B2B organizations today still employ well-established B2B approaches (such as electronic data interchange [EDI], FTP and managed file transfer [MFT]), the nascent power of digital business is poised to drive adoption of new approaches to B2B based on APIs.

So Web APIs have just started to make a difference, their visibility amplified by very public initial business-to-consumer (B2C) usage scenarios, but they still have a long way to go, and will move slowly through the Hype Cycle curve.

User Advice:

- Treat public Web APIs as products, and a key component of your organization's business strategy. Carefully consider how they can be exploited through Web, digital business, IoT, mobile and social strategies.
- Design your public Web API according to the needs of the intended API consumer, which could be a person (i.e., using an app) or a thing, as in IoT.
- Don't assume that "If we build it, they will come"; in general, the reverse is true. Establishing traction for a public Web API requires creating developer awareness, targeting and cultivating a community of developers and tapping into nearby communities, as well as making the richly documented API easy to consume via a self-service developer portal.
- Before releasing your Web API to the outside world, consider the traffic, resource and security implications for your data center and the data encapsulated by the API. Complement your infrastructure with API management and application services governance products and cloud services, as required (i.e., to protect assets through security policies), and ensure that you have an adequate funding model in place for its ongoing operation.
- Put application services governance in place, including API management, prior to launching a public Web API. Also involve the enterprise architecture team and security team by, for example, taking into consideration dynamic application security testing (DAST) and static application security testing.

Business Impact: Public Web APIs provide the opportunity for IT organizations to become central players in helping their organizations expand top-line growth, improve service delivery, drive competitive advantage and reimagine their industries.

A wide variety of organizations have already begun to see measurable business growth by leveraging APIs to encourage and support the introduction of their products, services and content into adjacent markets, new channels and new delivery contexts by third parties.

Web APIs support mobile app development, foster innovative applications, drive customer engagement and extend business models. Many organizations, such as Netflix, Best Buy, BBVA, ESPN, Hoovers and The New York Times, have seen significant success with their public API programs. However, most organizations begin their API programs with an internal focus, such as in support of mobile or multichannel development initiatives. These organizations frequently plan to extend their API initiatives to external parties as they hone their skills through experience and establish the requisite business propositions for their funding.

Benefit Rating: Transformational

Market Penetration: 5% to 20% of target audience

Maturity: Early mainstream

Sample Vendors: 3scale; Apigee; Axway (Vordel); CA Technologies (Layer 7); IBM; Intel (Mashery); MuleSoft; WSO2

Recommended Reading: "Extend the Enterprise With Public Web APIs"

"Basic API Management Will Grow Into Application Services Governance"

"Designing a Great Web API"

"Govern Your Services and Manage Your APIs With Application Services Governance"

"Magic Quadrant for Application Services Governance"

External Hackathons

Analysis By: Kristin R. Moyer

Definition: External hackathons are focused events during which many individual or team participants from outside of the bank quickly develop and showcase their ideas. In the context of open banking APIs, external hackathons typically involve third-party developers using APIs, and other tools and services, to develop apps, data visualization and other outputs for areas such as mobile banking, social media, big data, trading, risk management, retail banking and commercial banking.

Position and Adoption Speed Justification: The purpose of internal and external hackathons is to rapidly develop new ideas and prototypes. External hackathons usually begin with a breakfast or other meal, followed by networking and a presentation of the objectives of the event. Sometimes there are workshops and other educational opportunities, but often the hacking begins next. Eating is usually informal, although sometimes meals are provided. Sleeping can also be informal, with some participants working long into the night with their sleeping bags nearby. When the time designated for hacking is completed, the participants demonstrate their apps. The demonstrations are usually very short — for example, a three-minute demo of the app and then two minutes for questions. A panel of judges evaluates the demos and selects winners. Winners receive prizes such as cash, iPads, Kindle Fires and gift cards.

The number of hackathons run by banks during the past 12 months are too numerous to list here. We believe that external hackathons will continue to increase in number, therefore reaching the Plateau of Productivity within two years. However, we expect hackathons are beginning to slide into the Trough of Disillusionment because CIOs sometimes struggle to take the ideas generated in a hackathon into a production environment (70% or more of the effort to bring a new app to market is around security and compliance) and as developers become fatigued by their use.

User Advice:

- Use external hackathons to develop and nurture third-party developer communities and quickly stimulate new ideas, apps, data visualization and other outputs.
- Use external hackathons to bring new apps to market much more quickly and cost-effectively.
- Ensure clear goals and financial benefits for participants, or risk lackluster attendance and participation.
- Use well-defined and monetized hackathons as a way to nurture and motivate the developer community to participate and stay engaged.
- Provide a predetermined level of technical support for the top three to five ideas and prototypes generated in the hackathon, so that they may become commercially viable.
- Review hackathon demos internally to stimulate innovative thinking.

Business Impact: Hackathons have been shown to reduce the time and cost to bring new mobile banking apps to market by up to 90%. Hackathons can also generate new, innovative solutions and ideas. For example, external hackathons usually result in location or context-specific solutions and ideas; a good idea from a hackathon in Los Angeles is not necessarily a good idea from a hackathon in India or even from another location in the U.S. Hackathons might jump-start some innovative thinking within the bank. They can also generate interest in the technology community for working in financial technology (e.g., "Banking development is not just boring technology and programming"). This could help banks attract new employees with fresh skills.

Benefit Rating: Moderate

Market Penetration: 5% to 20% of target audience

Maturity: Adolescent

Recommended Reading: "Citi Uses Hackathons to Accelerate Digital Innovation"

Sliding Into the Trough

Crowdsourcing

Analysis By: Carol Rozwell

Definition: Crowdsourcing is the process of sourcing a task or challenge to a broad, distributed set of contributors using the Web and social collaboration techniques. Crowdsourcing applications typically include mechanisms to attract the desired participants, stimulate relevant contributions and select winning ideas or solutions.

Position and Adoption Speed Justification: Crowdsourcing is being successfully applied to narrowly defined tasks, open-ended challenges and requests for ideas. It can be used internally or externally and, in either case, can be available to any participant or confined to a group of experts.

Successful crowdsourcing requires the sponsoring organization to:

1. Specify the task or challenge (including the time frame for responses, guidelines and rules) and notify potential contributors.
2. Establish rules for recognition or payment.
3. Assess intellectual property (IP) implications.
4. Ensure some level of quality control regarding access, contributions (particularly if prizes or payments are involved) and voting.

Crowdsourcing has been applied to a range of areas in government and private-sector organizations for nearly a decade. During the past two or three years, its use for idea generation in organizational innovation programs has accelerated rapidly. Innovation activities where customers create and rank ideas, or design marketing campaigns, are the most popular. These crowdsourcing activities generate ideas that are increasingly selected through voting by participants — with the final selection being made by the organization's employees.

As more people interact on the social Web, often by providing frequent updates on their location and activities, crowdsourcing involving all types of status information — mixed with context information — becomes practical. For example, drivers can alert fellow "road warriors" to traffic conditions; travelers can offer suggestions about good restaurants nearby; and knowledgeable people can answer other people's questions.

There is a large, untapped opportunity to apply crowdsourcing to a much broader range of tasks and goals. However, there is still more to be learned about where crowdsourcing is most effective, compared with other approaches. The tools to establish a crowdsourcing environment, particularly those involving recognition incentives or micropayments, are becoming widely available. But, as with any technology, effective practices are the critical success factor.

User Advice: Look for creative ways to use crowdsourcing, beyond idea generation. Consider tasks that can be broken down into smaller chunks and attacked in parallel — such as the classification of images — as candidates for crowdsourcing. Use crowdsourcing for innovation in areas that can be focused and well-defined as challenges. Look, in particular, for opportunities to crowdsource tasks to which volunteers (internal or external) would be prepared to contribute. Also, extend your ability to innovate or resolve these tasks in a resource-constrained environment. Government organizations are particularly well-positioned to take advantage of the willingness of citizens to help out in areas that affect their local environment or special interests. Alternatively, consider the use of crowdsourced communities or application-sourcing communities for application development and testing work, whereby crowdsourcing companies or your incumbent application service provider controls the crowd on your behalf.

When crowdsourcing with nonemployees, be prepared to grapple with IP issues. In some cases, you will reveal your own IP (for example, your internal or future plans) to outsiders. In other cases, you will need to take account of the terms of IP ownership for any relevant contributions from external participants.

Business Impact: There are multiple areas of business impact:

- There is potential to open up innovation efforts by stimulating and capturing creative ideas from outside an organization.
- Crowdsourcing offers the ability to dramatically increase the available human insight that can be applied to a task or challenge — well-designed crowdsourcing efforts will attract interest and creativity to a task.
- Crowdsourcing can be used for collaborative decision making, opening up the process to multiple groups within an organization.
- There are opportunities for organizations to crowdsource core business competencies, extend their access to key capabilities and change their associated cost structures.

Benefit Rating: High

Market Penetration: 5% to 20% of target audience

Maturity: Adolescent

Sample Vendors: Amazon; clickworker; CrowdFlower; CrowdSource; IdeaScale; InnoCentive; Quora; topcoder

Recommended Reading: "Leading Enterprise Architects Leverage Gartner's Innovation Crowdsourcing Framework"

"Market Guide for Crowdsourced Application Testing Services"

"Market Guide for Innovation Management"

Entering the Plateau

Fintech APIs

Analysis By: Kristin R. Moyer; Stessa B Cohen

Definition: Fintechs are startup companies providing a banking service, often through mobile, social and/or cloud technologies. Fintech APIs are private or public APIs based on Web-oriented architecture principles to integrate, create new user interfaces, share data, expand product accessibility, authenticate and enable transactions.

Position and Adoption Speed Justification: Creating and launching a banking service has never been easier. Mobile, social, cloud, and information and architectural advances such as APIs can enable anyone to deliver a banking service quickly and inexpensively. Billions of dollars of venture capital were invested in fintechs in 2014 (see "Anyone Can Build a Bank"). Fintechs are aggressively using APIs and ecosystems to enable openness. For example, Zillow has more than 30 APIs that let partners integrate to its platform. Apple has paid more than $9 billion (www.apple.com/about/job-creation) in royalties to its developer ecosystem. The Stripe API is used more frequently for mashups than all bank-provided APIs put together. APIs are enabling fintechs to enable mobility, expand addressable market share and create new business models rapidly and inexpensively. We expect to see fintechs accelerate their use of APIs during the next 12 months, given this high level of success. This most likely means that fintech APIs will reach the Plateau of Productivity by 2016, even though many may not have profitable or sustainable business models.

User Advice:

- Identify customer pain points that can be resolved with fintech API platforms. Partner or acquire as appropriate.
- Deploy an API that enables an internal and/or external ecosystem to build apps that provide personalized data.

Fintechs should pursue banks as customers or partners more aggressively. Many banks are frustrated with the lack of innovation from incumbent banking vendors, and desire a work-around solution that will provide them with greater levels of flexibility.

Business Impact: Fintechs with API platforms are partners with and competitors to banking CIOs (or, in other words, "frienemies" and "co-opetition"). Fintechs can enable CIOs to go to market much more quickly and at a lower cost, but they can also become financial data and transaction powerhouses at a time when data is becoming a currency.

Fintechs that provide API platforms could displace incumbent banking vendors. Many nonbank providers are coming to market with focused APIs today (for example, automated clearinghouse [ACH] send and receive functions), but they plan to build out fully functioning banking platforms over time, and as their customer bases grow.

Benefit Rating: Transformational

Market Penetration: 20% to 50% of target audience

Maturity: Early mainstream

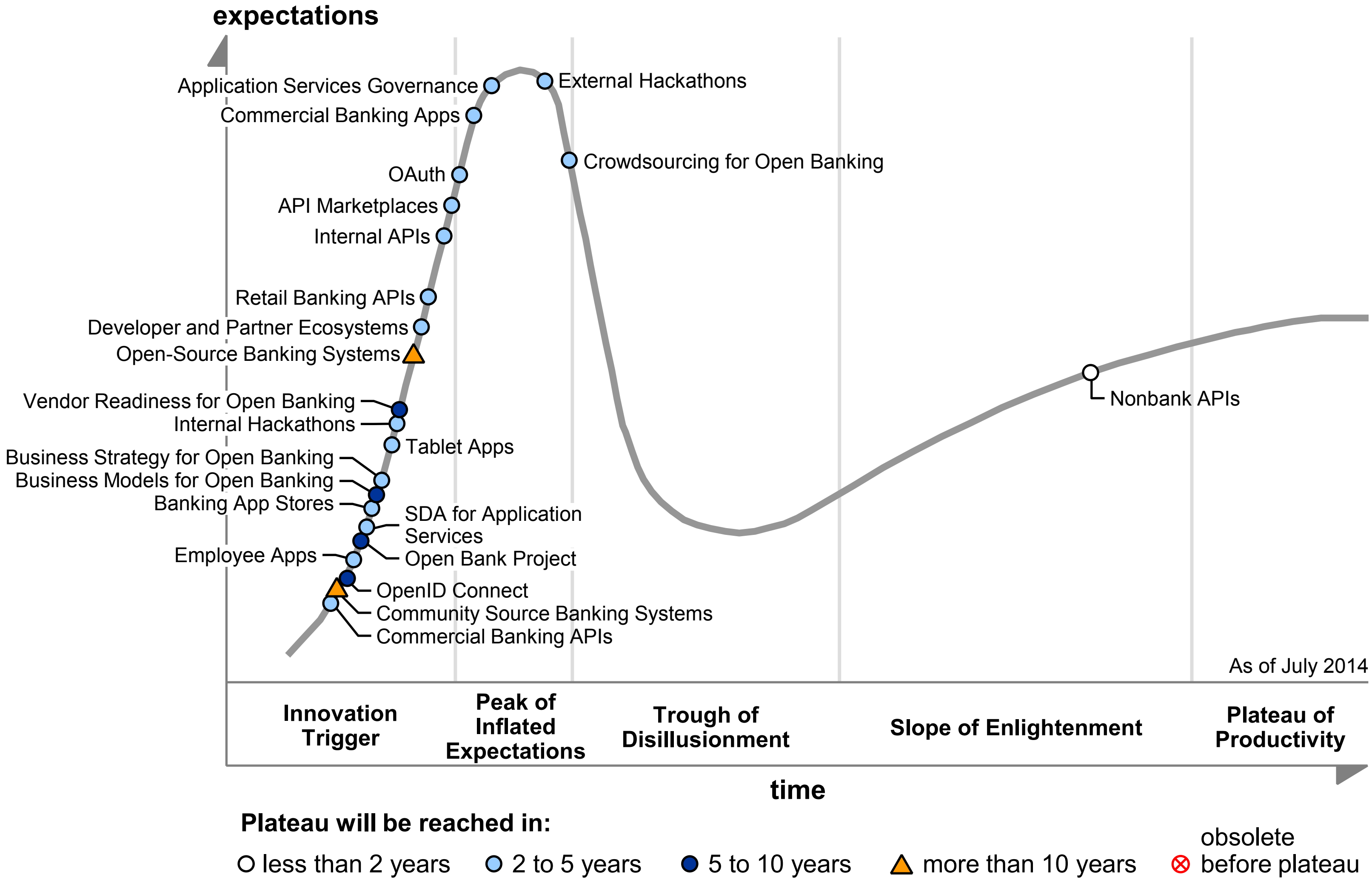
Sample Vendors: Amazon; Bancbox; Billtrust; Bill.com; FreedomPay; iZettle; Intuit; LevelUp; PayPal; Plaid; Square; Standard Treasury; WePay; Yodlee

Recommended Reading: "Anyone Can Build a Bank"

"Digital Disrupters and Customer Demand Force Bank CIOs to Find New Sources of Innovation"

Appendixes

Figure 3. Hype Cycle for Open Banking APIs, Apps and App Stores, 2014



Source: Gartner (July 2014)

Hype Cycle Phases, Benefit Ratings and Maturity Levels

Table 1. Hype Cycle Phases

Phase	Definition
Innovation Trigger	A breakthrough, public demonstration, product launch or other event generates significant press and industry interest.
Peak of Inflated Expectations	During this phase of overenthusiasm and unrealistic projections, a flurry of well-publicized activity by technology leaders results in some successes, but more failures, as the technology is pushed to its limits. The only enterprises making money are conference organizers and magazine publishers.
Trough of Disillusionment	Because the technology does not live up to its overinflated expectations, it rapidly becomes unfashionable. Media interest wanes, except for a few cautionary tales.
Slope of Enlightenment	Focused experimentation and solid hard work by an increasingly diverse range of organizations lead to a true understanding of the technology's applicability, risks and benefits. Commercial off-the-shelf methodologies and tools ease the development process.
Plateau of Productivity	The real-world benefits of the technology are demonstrated and accepted. Tools and methodologies are increasingly stable as they enter their second and third generations. Growing numbers of organizations feel comfortable with the reduced level of risk; the rapid growth phase of adoption begins. Approximately 20% of the technology's target audience has adopted or is adopting the technology as it enters this phase.
Years to Mainstream Adoption	The time required for the technology to reach the Plateau of Productivity.

Source: Gartner (July 2015)

Table 2. Benefit Ratings

Benefit Rating	Definition
Transformational	Enables new ways of doing business across industries that will result in major shifts in industry dynamics
High	Enables new ways of performing horizontal or vertical processes that will result in significantly increased revenue or cost savings for an enterprise
Moderate	Provides incremental improvements to established processes that will result in increased revenue or cost savings for an enterprise
Low	Slightly improves processes (for example, improved user experience) that will be difficult to translate into increased revenue or cost savings

Source: Gartner (July 2015)

Table 3. Maturity Levels

Maturity Level	Status	Products/Vendors
Embryonic	▪ In labs	▪ None
Emerging	▪ Commercialization by vendors ▪ Pilots and deployments by industry leaders	▪ First generation ▪ High price ▪ Much customization
Adolescent	▪ Maturing technology capabilities and process understanding ▪ Uptake beyond early adopters	▪ Second generation ▪ Less customization
Early mainstream	▪ Proven technology ▪ Vendors, technology and adoption rapidly evolving	▪ Third generation ▪ More out of box ▪ Methodologies
Mature mainstream	▪ Robust technology ▪ Not much evolution in vendors or technology	▪ Several dominant vendors
Legacy	▪ Not appropriate for new developments ▪ Cost of migration constrains replacement	▪ Maintenance revenue focus
Obsolete	▪ Rarely used	▪ Used/resale market only

Source: Gartner (July 2015)

Gartner Recommended Reading

Some documents may not be available as part of your current Gartner subscription.

"Understanding Gartner's Hype Cycles"

"The Business Value of Banking APIs"

"How to Make APIs a Banking CEO and Board Priority"

"Citi Uses Hackathons to Accelerate Digital Innovation"

More on This Topic

This is part of an in-depth collection of research. See the collection:

- Gartner's Hype Cycles for 2015: Five Megatrends Shift the Computing Landscape

GARTNER HEADQUARTERS

Corporate Headquarters
56 Top Gallant Road
Stamford, CT 06902-7700
USA
+1 203 964 0096

Regional Headquarters
AUSTRALIA
BRAZIL
JAPAN
UNITED KINGDOM

For a complete list of worldwide locations,
visit http://www.gartner.com/technology/about.jsp