Mediated APIs: An Essential Application Architecture for Digital Business

Published 30 May 2018 - ID G00351557 - 14 min read

By Analysts Aashish Gupta, Anne Thomas, Mark O'Neill

Supporting Key Initiative is Modernizing Application Architecture and Infrastructure

APIs are fundamental to your digital business technology platform, API economy, mesh app and service architecture, and integration. Application leaders responsible for architecture and infrastructure should adopt the mediated API pattern to protect, control and enhance these APIs.

Document Revision History

Mediated APIs: An Essential Application Architecture for Digital Business - 26 August 2016 (https://www.gartner.com/document/code/310378?ref=ddrec)

Overview

Key Challenges

- Application leaders need to expose their application and data services via APIs, while simultaneously protecting and controlling those services.
- Shared general-purpose APIs don't always support the specific needs of their consumers.
- API-enabling legacy and packaged applications are not enough to connect them with modern applications. They are likely to be semantically and syntactically inconsistent with each other.
- Organizations cannot build a thriving ecosystem based on APIs without proper governance.

Recommendations

To modernize application architecture and infrastructure, application leaders should:

- Direct your teams to protect and manage APIs by providing a mediation layer to enforce policies for managing security and traffic.
- Ask API product managers to extend, enhance and enable customization of APIs by using the mediated APIs pattern.
- Explore with their teams the technologies used in the mediation pattern to connect legacy and packaged applications, and manage complex integration flows.
- Make sure that the mediation layer is present in the platform on which the API ecosystem will be implemented.

Strategic Planning Assumption

By 2022, API abuses will be the most-frequent attack vector, resulting in data breaches of enterprise web applications.

Introduction

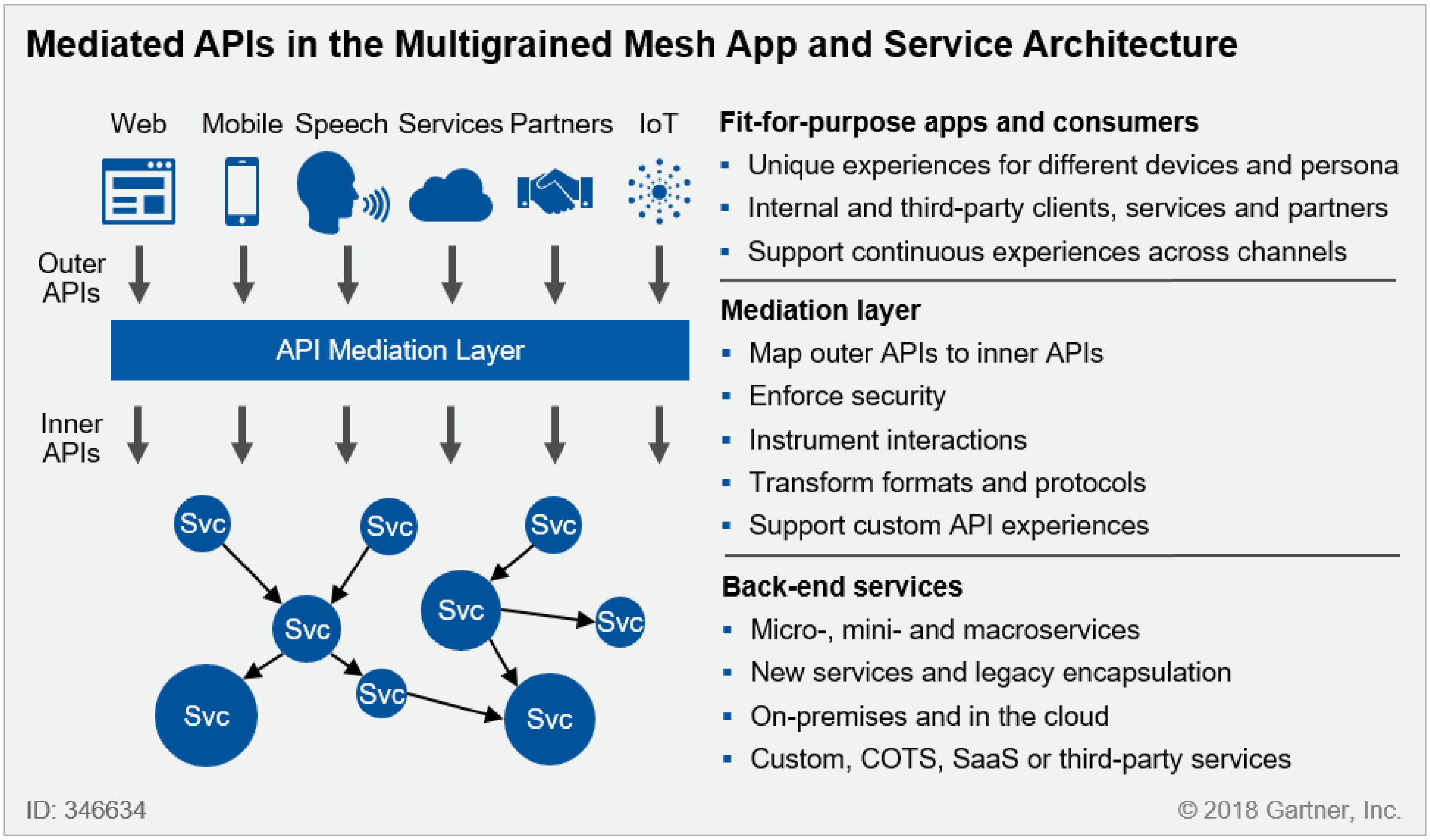
APIs are fundamental in every effort to modernize application architecture and integration. They provide essential access to application and data services that support:

- The creation of your digital business technology platform (DBTP; see "2017 Strategic Roadmap for Application Architecture, Infrastructure and Integration" (https://www.gartner.com/document/code/331056?ref=grbody&refval=3877163)).
- Multichannel and cloud-native applications (see "Adopt a Multigrained Mesh App and Service Architecture to Enable Your Digital Business Technology Platform" (https://www.gartner.com/document/code/338573?ref=grbody&refval=3877163)).
- Participation in an API economy (see "From APIs to Ecosystems: API Economy Best Practices for Building a Digital Platform" (https://www.gartner.com/document/code/331662?ref=grbody&refval=3877163)).
- Pervasive integration (see "How Pervasive Integration Enables Your API Initiatives (and Vice Versa)" (https://www.gartner.com/document/code/328687?ref=grbody&refval=3877163)).

Extensive use of APIs for distributed services, however, also generates new challenges for application leaders. It exposes sensitive functionality and information that must be protected, increases network traffic that must be managed, and increases application complexity, making monitoring and service-level management more challenging. Shared, general-purpose APIs also often don't support the specific needs of all potential consumers. And, in many cases, existing APIs to legacy and packaged applications are semantically and syntactically inconsistent with modern applications. All these challenges lead to a portfolio of applications and services that face security issues, have low traction in the ecosystem, and lack interoperability, much to the disappointment of the various stakeholders — customers, partners and product teams.

API mediation is the recommended solution for addressing these challenges. It places one or more mediators between an API consumer and the API service implementation. These API mediators protect and control access to APIs, and they enhance and enable utilization of APIs. Figure 1 illustrates the role of mediated APIs in modern application architecture — the mesh app and service architecture (MASA) — which supports multichannel, cloud-native applications. A MASA application provides multiple apps that support fit-for-purpose experiences for multiple personas and client devices. Those apps use mediated APIs to communicate with a mesh of distributed services that implement the back-end functionality. Back-end services many be of any granularity; they may be existing services or newly developed to support this particular application; and they may be internally managed, deployed in the cloud, or supplied by a third party. API mediation manages both inbound and outbound API traffic (see "Managing the Consumption of Third-Party APIs" (https://www.gartner.com/document/code/348312?ref=grbody&refval=3877163)).

Figure 1. Mediated APIs in the Multigrained Mesh App and Service Architecture



Mediated APIs in the Multigrained Mesh App and Service Architecture

Fit-for-purpose apps and consumers

- Unique experiences for different devices and persona
- Internal and third-party clients, services and partners
- Support continuous experiences across channels

Mediation layer

- Map outer APIs to inner APIs
- Enforce security
- Instrument interactions
- Transform formats and protocols
- Support custom API experiences

Back-end services

- Micro-, mini- and macroservices
- New services and legacy encapsulation
- On-premises and in the cloud
- Custom, COTS, SaaS or third-party services

ID: 346634

Source: Gartner (May 2018)

Application leaders responsible for application architecture and infrastructure should:

- Invest in a mediation layer to enforce security, traffic management and transformation policies.
- Guide their teams to take advantage of its multilayer pattern to provide customized experiences to API consumers.

- Leverage the mediation capabilities offered by integration platforms to integrate API-enabled legacy applications.

Analysis

Direct Your Teams to Implement a Mediation Layer for Enforcing Policies

The mediated API model gives you a consistent model for defining and enforcing policies on all your API interactions, including internal, external and third-party interactions, and also including request/response, message-oriented and event-driven interactions. It is also important to note that it applies to APIs in general and is not limited to just REST APIs. Policies can be defined declaratively and applied to one, many or all APIs. Updates to policies can be made once and automatically applied to all associated APIs. Application leaders should direct their teams to adopt this architectural model for all published APIs — that is, all APIs that can be consumed by application components that operate outside the scope of a single application service.

API mediation allows you to ensure that security and compliance policies are properly enforced. It also improves performance and scalability by supporting routing and load balancing, and increases your agility by reducing the brittleness of API contracts. And it improves the usability of APIs by supporting custom API experiences. The mediation layer allows a service to expose an "inner API" that directly reflects its domain model, and one or more "outer APIs" tailored to support specific client requirements. A mediator can inject additional capabilities that can enrich the interaction. The model is, by definition, extensible, permitting any type of injected capability.

The mediated APIs pattern provides numerous benefits, including:

- A standard way to implement security and access control on all APIs (see "How to Build an Effective API Security Strategy" (https://www.gartner.com/document/code/342236?ref=grbody&refval=3877163)).
- A mechanism that improves overall system performance, scalability and resiliency.
- A mechanism for managing and throttling inbound and outbound traffic.
- A mechanism that supports API versioning and custom APIs.
- A standard way to instrument API interactions so that you can monitor distributed applications and service-level agreements.
- Collects essential information/metrics that can be utilized in the monetization of APIs and in judging the effectiveness of the API strategy.

A mediation layer can be implemented using a variety of technologies, although API gateways are the most common mediation technology. Many types of products supply API gateways, including:

- Full life cycle API management systems
- Stand-alone enterprise gateways

- Microgateways
- Integration platform as a service (iPaaS) offerings
- Mobile back-end services (mBaaS)
- Service mesh for microservices

Other types of mediation technologies include:

- Enterprise service buses (ESBs), which specialize in routing, load balancing and transformation (as well as integration, which typically implements business logic and is outside the scope of API mediation).
- Network security mediators, such as application delivery controllers (ADCs), web application firewalls (WAFs) and cloud access security brokers (CASBs).
- Homegrown custom mediators, which can be especially useful when building complex mediations for custom experiences.

Always use a capabilities approach to determine which technology, or combination of technologies, is best suited to your requirements. For organizations that are still in the early stages of their API program, the built-in API gateway in tools they may already be using, such as iPaaS or mBaaS, may be sufficient to support your needs. More advanced organizations will typically bring in a full life cycle API management solution to address related API management requirements. But, in some cases, an API gateway may not be sufficient to address all your requirements. For example, you may need advanced bot mitigation, in which case you may want to deploy additional security mediators to your infrastructure (see "Design an API Mediation Layer to Underpin Your Digital Business Technology Platform" (https://www.gartner.com/document/code/323828?ref=grbody&refval=3877163)).

Ask API Product Managers to Leverage the Mediated APIs Pattern to Offer Customized Experiences

APIs have moved beyond being just programmable interfaces that provide access to data or to an application. In the API economy, they are capable of generating revenue, directly through monetization or indirectly by providing a new channel to traditional business. Therefore, organizations need to think about APIs as products, and the users of those APIs as their customers. A proper API strategy requires having an API product manager that ensures that the APIs meet consumer requirements, which often means having a strategy to support customized APIs (see "Create the Role of API Product Manager as Part of Treating APIs as Products" (https://www.gartner.com/document/code/320767?ref=grbody&refval=3877163)).

A service typically exposes an API that directly reflects its internal domain model, process workflow and native communication model (format, protocol and interaction model). But consumers may require various levels of customization of that API. For example, a mobile client may require an API that allows the app to aggregate information to send via a single request rather than using a chatty API that walks through a step-by-step process. The mobile client may also need the API to return a subset of the information model. In other cases, a client may require a specific communications protocol (such as SOAP versus REST), or the client may want to interact with the service using an asynchronous pattern when the existing API only supports request/response. You don't want to burden the service development team with building custom APIs to support each custom API request. And because APIs need to evolve over time, versioning can be a significant challenge for API product managers. You want to add new features, but you don't want to break the API for existing consumers.

API mediation provides a convenient method to support customization and versioning. It enables a service to work with its inner API, retaining its focus on its native domain model and workflow, while the mediation layer maps that inner API to any number of outer APIs that support custom experiences for its consumers. The outer/inner API model can also map older versions of the API to a new version of the API, ensuring a manageable transition for existing API consumers.

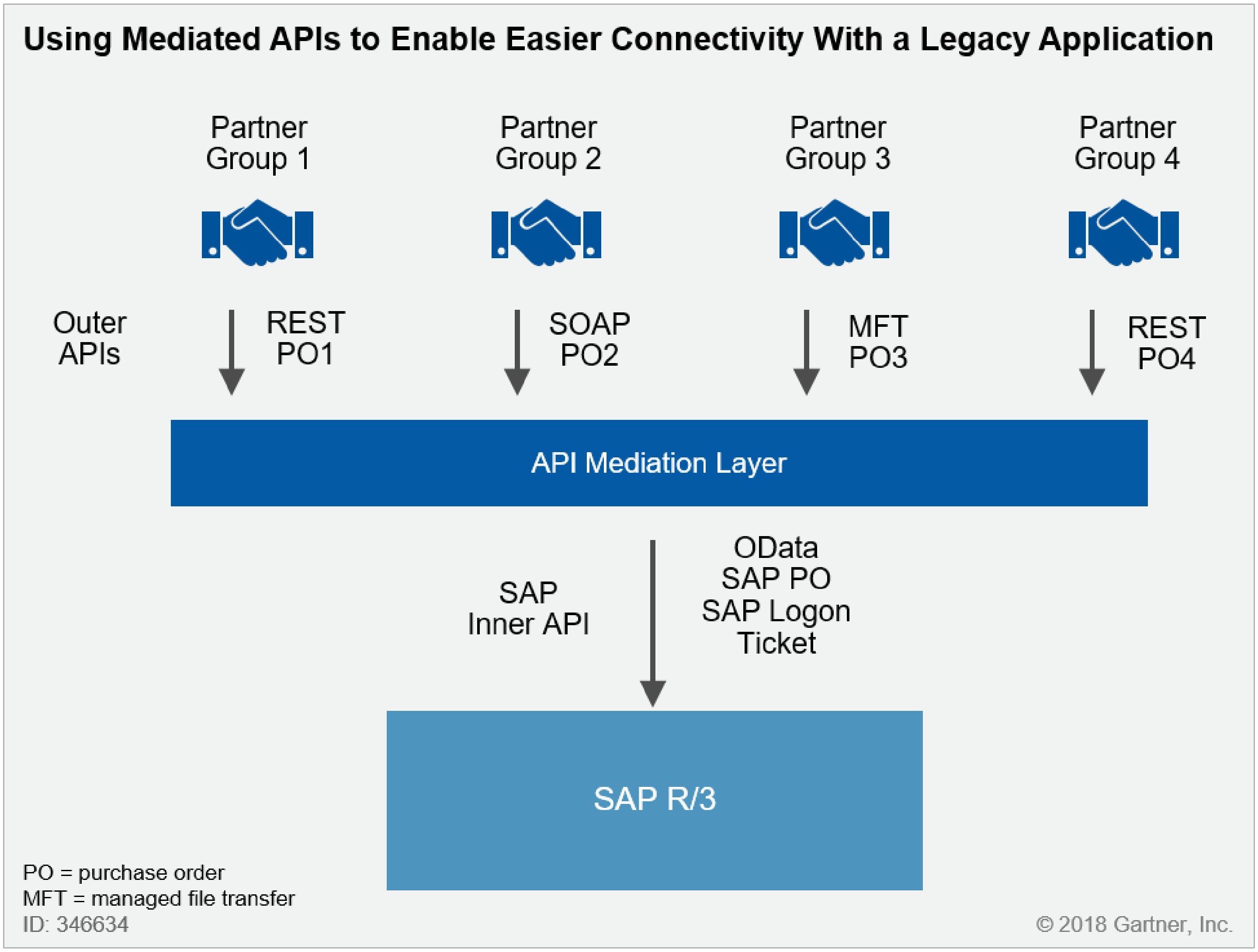
Be careful not to implement business logic in the API mediation layer though. Complex orchestrations and transformations should be implemented in separate services.

Encourage Your Teams to Explore Mediation Platforms to Connect Legacy and Packaged Applications

APIs have become the most popular means to enable integration with legacy and packaged applications. However, in many cases, an API alone is inadequate for enabling connectivity for all potential consumers. APIs to legacy and packaged applications often use proprietary formats, protocols, and authentication and authorization methods that make connectivity more difficult.

A mediation layer can simplify legacy connectivity by supporting capabilities such as credential mapping, protocol translation and message transformation. For example (see Figure 2), an organization wants to make business easier for its partners by supporting multiple purchase order (PO) formats submitted through multiple protocols (REST, SOAP, MFT, etc). However, its SAP ECC system requires a specific PO format delivered via a specific protocol (OData) to its native API. An API mediation layer can translate the various partner PO formats and protocols into the required SAP PO format, and it can convert various types of credentials into an SAP logon ticket.

Figure 2. Using Mediated APIs to Enable Easier Connectivity With a Legacy Application



Source: Gartner (May 2018)

Important things to consider when connecting to legacy and packaged applications through a mediation layer:

- If the applications don't expose any APIs, basic service enablement of these applications will be required. You will need to build or buy some type of adapter to the application (which may require some modification to the application). Once the adapter is in place, an API mediator can abstract it and expose outer APIs that make it easier to consume.
- The mediation layer could be built in-house or implemented via an API gateway for simple and minimal integration scenarios. However, technologies such as ESBs and iPaaSs are better suited for complex integration workflows.
- Modern applications/application services use more current access control methods like SAML, OAuth, and OpenID Connect. These current methods need to be mapped to the old, proprietary authentication and authorization mechanisms (some of which are embedded inside the application code). API mediation can provide the necessary bridge.

API mediation is a critical capability for implementing a pervasive integration strategy, which is an organizational practice to build enterprisewide integration competencies across all types of endpoints, domains, deployment models and integration personas. Gartner's hybrid integration platform (HIP) capability framework describes the competencies and technology capabilities that organizations need in order to implement a pervasive integration strategy and build an HIP. See "How to Implement a Hybrid Integration Platform to Tackle Pervasive Integration" (https://www.gartner.com/document/code/300867?ref=grbody&refval=3877163) to learn more about this framework.

The Mediation Layer Should Be Present in the Platform on Which the API Ecosystem Will Be Implemented

API ecosystems offer new business models and revenue (direct and indirect) opportunities, accelerated innovation, and greater control for ecosystem providers over transactions happening between themselves and their customers and partners. Once you have successfully evaluated the ecosystem's business model, cost justification and value promotion (see "To Create a Successful API-Based Ecosystem, Look Before You Leap" (https://www.gartner.com/document/code/351667?ref=grbody&refval=3877163)), you should then consider appropriate mediation strategies for providing the ecosystem a strong backbone.

Why Do Application Leaders Need to Include Mediation in Their Ecosystem Strategy?

When businesses rely on your ecosystem or access sensitive information generated within it, the need for mediation becomes crucial. Just having a developer portal or an API marketplace will not suffice if there are no adequate governance mechanisms behind them. The recent case of Cambridge Analytica and Facebook has brought the issue of API governance into the spotlight. In an ecosystem, a breach or a misuse of the provider's APIs could have a domino effect on all the participants, who may become wary and start looking for alternatives. The urgent remediation measures, such as deprecation of the affected API, adopted by the ecosystem provider may break access for highly dependent participants, who may have built their entire business model on the API. This breaks down trust in the system. In such scenarios, application leaders and senior leadership find themselves answerable to customers, partners and internal stakeholders.Having a mediation layer helps in the following ways:

- It helps minimize security incidents through greater control with declarative policies.
- It provides greater visibility into the situation through usage and value metrics.
- It enables deeper insight and control to help establish more effective remediation measures.

Case Study

Pitney Bowes (PB) is an international e-commerce and fulfillment technology provider. Its application/products teams needed to connect their mailing and shipping hardware products with the back-end services provided by APIs from their commerce cloud, as well as use adjacent third-party APIs within their products. Application leaders were concerned about the impacts of such an undertaking on security, product usage and the customer experience.

This was further complicated as PB's own services leveraged custom-built capabilities and those exposed by third-party APIs to deliver a complete solution. For example, its SendPro C-Series postage meter machine prints labels for packages. It calls its GeoSearch API to autocomplete addresses, which further connects with an external location service to retrieve IP address information. In such a scenario, there is a need to manage and govern both the outgoing and incoming API calls that terminate and originate from various channels and devices.

Application leaders guided their teams to design and implement a mediation layer to govern all communication between their products and back-end services, as well as requests to third-party APIs (see Figure 3). The technology their teams used to implement the API mediation layer was a full life cycle API management solution (Google Apigee).

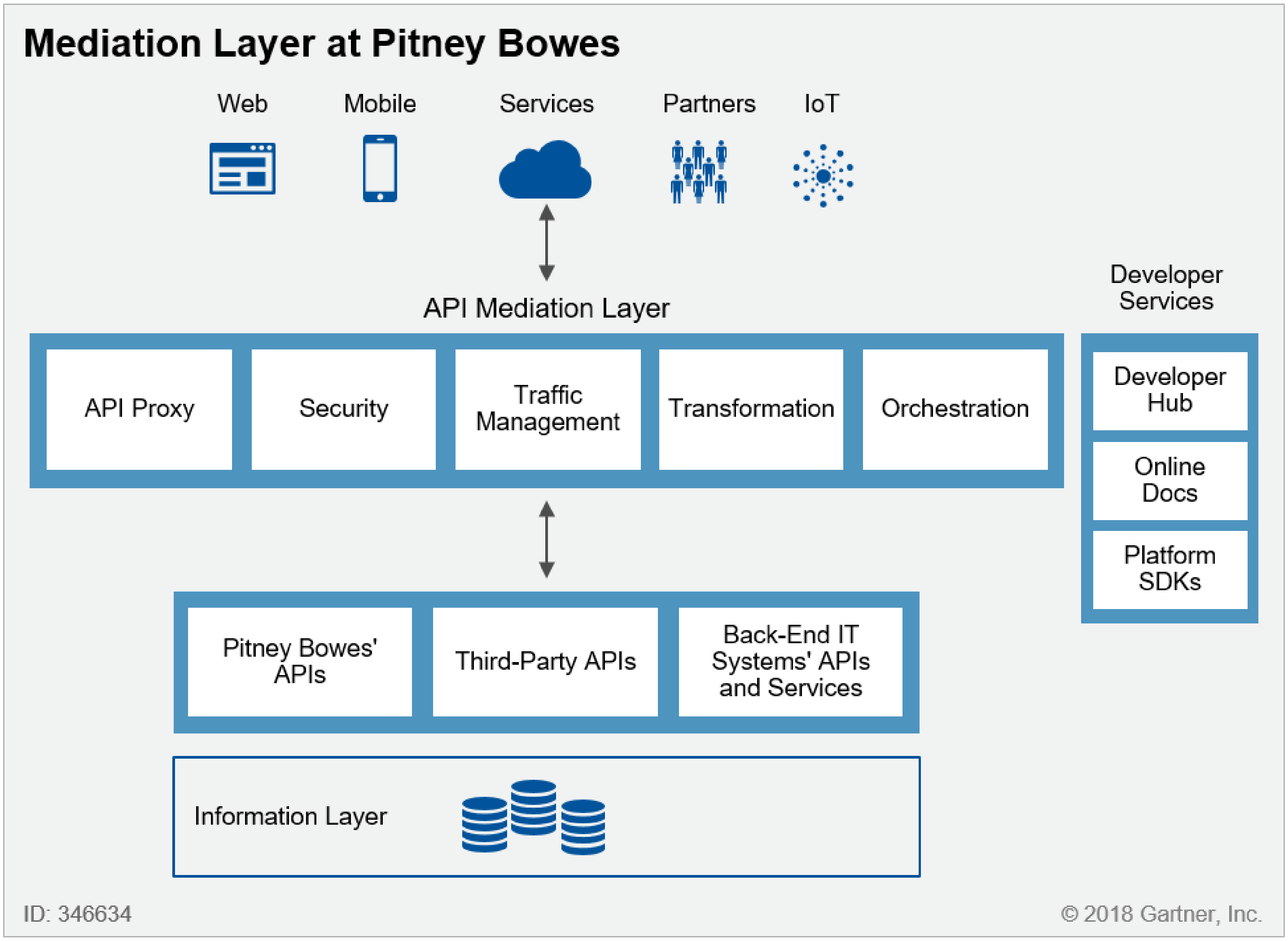
PB now uses the mediation layer in three ways:

1. **Managing the outbound calls:** The back-end services call various third-party APIs for a variety of capabilities, including resolving physical addresses from IPs and MAC addresses, retrieving personal identity information, and gathering real-time weather observations. These calls are proxied by the gateway to manage their consumption.

2. **Refining data returned on inbound requests:** The mediation layer refines and enriches the data received from third-party APIs before presenting it to the requesting consumer service/client. This ensures coherence between APIs and abstracts proprietary formats used by other service providers. This refinement and abstraction enable developers to communicate with all APIs in the same way without worrying about the changes in downstream services.

3. **Tailoring authentication based on the end user:** Authentication and authorization flow vary based on whether the client is a web interface or a machine.Creating a separate authentication service for each edge case would be inefficient and cumbersome to maintain.With the help of the mediation layer, PB can use a single authentication system in the back end, but provide tailored authentication mechanisms for different devices.

Figure 3. Mediation Layer at Pitney Bowes



Source: Gartner (May 2018)

Document Revision History

Mediated APIs: An Essential Application Architecture for Digital Business - 26 August 2016 (https://www.gartner.com/document/code/310378?ref=ddrec)

Recommended by the Authors

Adopt a Multigrained Mesh App and Service Architecture to Enable Your Digital Business Technology Platform (https://www.gartner.com/document/code/338573?ref=ggrec&refval=3877163)

Design an API Mediation Layer to Underpin Your Digital Business Technology Platform (https://www.gartner.com/document/code/323828?ref=ggrec&refval=3877163)

Managing the Consumption of Third-Party APIs (https://www.gartner.com/document/code/348312?ref=ggrec&refval=3877163)

How to Build an Effective API Security Strategy (https://www.gartner.com/document/code/342236?ref=ggrec&refval=3877163)

How Pervasive Integration Enables Your API Initiatives (and Vice Versa) (https://www.gartner.com/document/code/328687?ref=ggrec&refval=3877163)

A Guidance Framework for Evaluating API Management Solutions (https://www.gartner.com/document/code/337691?ref=ggrec&refval=3877163)

Magic Quadrant for Full Life Cycle API Management (https://www.gartner.com/document/code/319327?ref=ggrec&refval=3877163)

Recommended For You

2017 Strategic Roadmap for Application Architecture, Infrastructure and Integration (https://www.gartner.com/document/3805464?ref=ddrec&refval=3877163)

Design an API Mediation Layer to Underpin Your Digital Business Technology Platform (https://www.gartner.com/document/3698953?ref=ddrec&refval=3877163)

Innovation Insight: The Digital Integration Hub Turbocharges Your API Strategy (https://www.gartner.com/document/3880263?ref=ddrec&refval=3877163)

How to Achieve Digital Business Excellence by Mastering Pervasive Integration (https://www.gartner.com/document/3795671?ref=ddrec&refval=3877163)

How to Implement a Truly Hybrid Integration Platform (https://www.gartner.com/document/3858363?ref=ddrec&refval=3877163)

It consists of the opinions of Gartner's research organization, which should not be construed as statements of fact. While the information contained in this publication has been obtained from sources believed to be reliable, Gartner disclaims all warranties as to the accuracy, completeness or adequacy of such information. Although Gartner research may address legal and financial issues, Gartner does not provide legal or investment advice and its research should not be construed or used as such. Your access and use of this publication are governed by Gartner's Usage Policy. Gartner prides itself on its reputation for independence and objectivity. Its research is produced independently by its research organization without input or influence from any third party. For further information, see "Guiding Principles on Independence and Objectivity."