

The Unissant logo is a blue circle containing the word "Unissant" in a dark blue, sans-serif font. The background of the entire page is a light blue network of interconnected nodes and lines, with a large green circular graphic on the right side containing icons for a computer monitor, a person, and a curved arrow.

Unissant White Paper Series

Application Programming Interface (API) – Drivers of Digital Transformation

Application Programming Interface (API)–Drivers of Digital Transformation

One of the keys to modernizing legacy applications is to take advantage of Cloud Computing by breaking up of monolithic application/services into micro services that can be re-used and “glued together” into new applications/services. These services require a mechanism to communicate with each other and Application Programming Interfaces (APIs) fill that need. They act as the “Glue” to tie these the services to together to create new applications/services. APIs have been a key driver for digital transformation and led to terms such as “API Economy” and “API First Approach.” There are multibillion-dollar companies who are built on “API Economy.” This paper focuses on a key aspect required for successful adoption of APIs – API Governance.

What are APIs?

The need for communication between software components has been around for a long time in different forms and shapes. However, it has taken on a whole new dimension with the advent of cloud computing and microservices. Application Programming Interface (API), in very simple terms, provide mechanism for two software applications to communicate with each other. They define how the applications can talk to each other, the language, the naming convention, the operations, and the access information.

Some consider APIs as an evolution of Service Oriented Architecture (SOA) which defined the concepts such as “Service Consumers” and “Service Producers.” SOA introduced the concept of breaking functionality of an application into independent services that are well-defined, re-usable and discoverable by “consumers” to accomplish a given task. This concept of usage of SOA is very much applicable to the API world and it impacts how an organization manages designs, implements, and manages these APIs. One can also see the roots of this in structured programming languages where functions/procedures were defined to perform a specific task and in Remote Procedure Calls (RPCs). Thus, APIs can also be considered as an evolution of Legacy Web Services, RPC, FTP, and other messaging formats.

So, what has changed? It is the boundary between the consumers and providers thus leading to the term “API Connectivity.” The systems no longer reside in a single environment but across multiple distributed environments and across multiple computing environments and, platforms such as cloud and mobile. In addition, as organizations embrace the concept of breaking monolithic applications into micro-services, APIs become the key enablers for exposing the micro-services to other applications for consumption. A simple example of an API comes from the travel industry. Travel aggregators such as Expedia or Google Flights are calling the APIs provided by the airline reservation systems to show the availability of tickets to meet the criteria entered by the user. Aggregators do not have to be concerned about the underlying implementation of the APIs or the where the service is executed, as long as they follow the interface definition prescribed by the API. Another great example of how businesses are built on providing APIs is Twilio. The company provides APIs to engage customers across channels - SMS, voice, video, WhatsApp, email and more. Recently Twilio’s APIs are powering the communication channels required to programmatically send and receive text messages. In fact, this platform is being used currently by many Govt. agencies to co-ordinate Covid-19 distribution activities!

Types of APIs

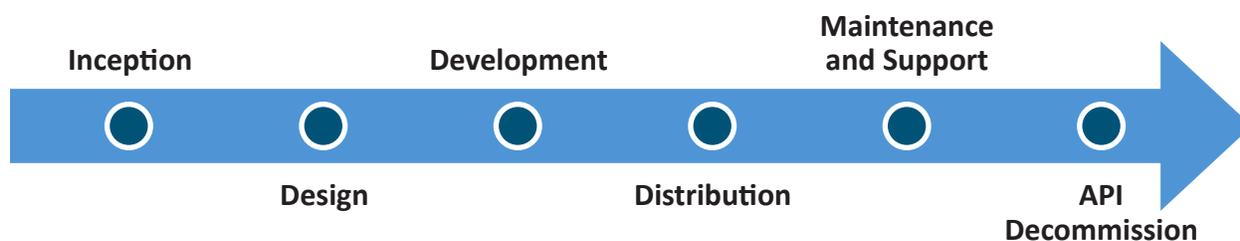
The proliferation of APIs poses some interesting challenges for management and governance. Before we look at the challenges let us look at the various types of APIs as that will determine the types of challenges and provide a better understanding of the scope of these challenges.

APIs can be grouped under two main categories. The first category is based on the types of applications or components that can communicate with each other: (a) Process APIs – for orchestration of APIs (b) System APIs – for communicating with the Systems of Record or backend systems, and (c) Experience APIs – for external clients to consume the API. The second group is based on the location of the consumers with respect to the providers of the API: (a) Internal – used exclusively within an organization or company (b) External - external consumers (c) Partners – business partners.

API Governance

Each organization that is adopting APIs, irrespective of whether they are using it for internal digital transformation efforts (e.g., Center for Medicaid and Medicare Services) or for external partners basing their whole business model on selling APIs (e.g., Twilio), need to manage APIs just as they would manage any products. Each API has its own life cycle and needs to be defined, designed, and developed to meet specific customer needs. The customer can be end users (e.g., Mobile Application Developers) or internal Systems of Record (e.g., HR) or other business partners (e.g., Credit Card company). This approach to design is what makes APIs most attractive as a key enabler for Digital Transformation.

APIs should not be developed with the anticipation of someone using it when it is built and published. The model of “Build and they will come”, will not work! Instead, the APIs should be driven by the needs of the consumers belonging to the categories described above. The consumers should be able and willing to use them confidently without the need to understand how they are interacting with backend systems. Achieving this goal as an API provider necessitate a robust API Governance Model. So, what is API governance? It is a combination of people, process and technology put in place to manage the complete lifecycle of an API to ensure the maximum utilization of the APIs and thus maximize return on investment. It covers all aspects of a typical product life cycle:



The scope of API Governance includes the following items:

- Identification of need for APIs and the APIs to be developed
- API Architecture, Design and Development Decisions
- Interface Standards to be used
- API Security models
- API Deployment and Publishing including mechanisms for discovery
- API Maintenance and Support that includes API Usage monitoring, maintenance, upgrades,

- and version control
- API Decommissioning
- API Utilization Metrics
- API User Community Support

A combination of people, process and technology is required to address each aspect of this governance.

People

Several roles are required to create an API governance structure. A good set of roles to begin with are: Leadership team, Product Manager, Architect, Designer, Developers and testers, Trainers/Community and Marketing/Support group. Each of these roles has their own roles and responsibilities that should be clearly defined.

Process

At the end of the day, Governance needs to be supported by organizational processes depending on the definition of the API business model. The roles defined earlier will ensure the execution of these processes. One example of a process will be API version control. Any updates to an existing API will affect multiple steps in the API lifecycle and, other supporting functions such as marketing and customer support. Ideally the change to an API must have minimum effect on the consumers of the API. In other words, it should be as transparent as possible. Imagine what the impact would be if an organization decided to de-commission an API! Having good governance process to manage version control will minimize and mitigate any before they new version of an API is deployed.

Technology

Role of technology in API Governance is obvious for both the development of an API and the management of API spanning from the technology we use to design and implement the API to management and discovery of the API after it is deployed to production. Ideally, the implementation details should be completely transparent to the user other than the way the API End points are exposed, and the mechanisms provided for API Discovery. Fortunately, there are multiple vendors to support API Management and Discovery. Many a times, the term API Management is used synonymously with API Governance. However, we treat API management as a subset of API Governance to distinguish between the technology and the process used for design and development processes, and the execution of these processes.

Summary

APIs are the key enablers of digital transformation that is leveraging advances in data analytics, cloud computing and microservices in almost every major sector of the economy. APIs are becoming a core component of the business model of many companies and a new term has been coined around that model – “API Economy.” API Governance brings the discipline and efficiency needed to maximize the return on investment.

About Unissant

Unissant is an advanced data analytics and business transformation services provider with expertise in healthcare and health IT, finance, national security, and energy. Unissant delivers innovative solutions to assist Government agencies and private sector businesses in tackling their biggest challenges. Founded in 2006, Unissant is a prime contractor on various Government vehicles such as CIO-SP3 SB, GSA PSS, GSA HealthIT SIN, and GSA 8(a) STARS II. We are assessed at a CMMI DEV-SVC Level 3; and an ISO 9001 and 27001 certified company. Headquartered in Herndon, Virginia, we also have a satellite office in San Antonio, Texas.

Unissant has been a leader in helping our customer in their digital transformation journey. We work with our customers in leveraging their investments, and modernizing their infrastructure when necessary, to take advantage of the emerging cloud computing platforms by moving from monolithic architecture to micro-services and connecting them using APIs.

Copyright

Copyright © 2020 Unissant, Inc. All Rights Reserved.

This document may not, in whole or in part, be copied photocopied, reproduced, translated, or reduced to any electronic medium or machine-readable form without prior consent, in writing, from Unissant, Inc.

Information in this document is subject to change without notice and does not represent a commitment on the part of Unissant. This document is provided “as is” without warranty of any kind including without limitation, any warranty of merchantability or fitness for a particular purpose. Further, Unissant does not warrant, guarantee, or make any representations regarding the use, or the results of the use, of the written material in terms of correctness, accuracy, reliability, or otherwise.

All other brand and product names are trademarks of their respective companies.

For additional information on Unissant full range of services, please visit our website at www.unissant.com or call us at 703.889.8500.