



An API Journey for Velocity and Lower Cost

Introduction

With digital business and market competition intensifying, many companies are shifting to APIs to expose business processes and data for easy and better integration, enabling business expansion, and driving lower cost and faster time to market (TTM) through reusable services and APIs. APIs are being deployed on an unprecedented scale in broad ecosystems for social, mobile, cloud, and analytical applications as well as for the Internet of Things (IOTs). This article shares what Intel IT has learned in adopting service oriented architecture (SOA) and APIs.

Intel IT's Approach to APIs

APIs are now used by most global businesses to interconnect and integrate with each other's businesses, forming a trend called an API Economy. Since 2013, Intel IT has been promoting the adoption of API based application development practice along with our SOA effort. We standardized APIs, design patterns, and code libraries. We developed an API based application framework to help Intel IT deliver solutions and integrations faster and more flexible.

Prior we started our API journey, most of our applications were built as monolithic solutions with tight integrated architecture. A typical application was built as a closed system, with some modules compiled and linked together to run as a single process. With little ability for reuse, every capability or integration needed some new design and development effort. As we started our API journey, we created our API strategy, and developed standards, technical guidance, training, and an application framework based on the API and service concept, as illustrated in Figure 1.

We used the standards, guidance, and the API based application framework to guide our application architecture design and development. The framework helped teams shift their application development approach from building tightly coupled applications to building APIs and SOA service based applications.

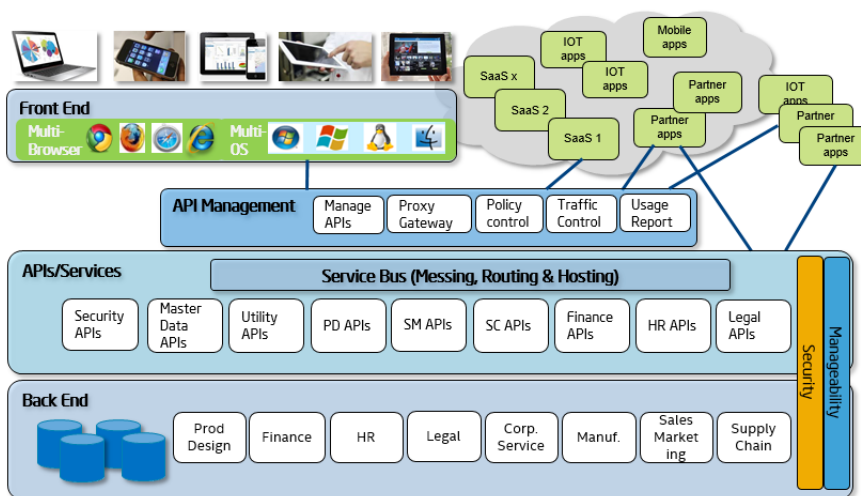


Figure 1. *API based application framework.*

We also learned that we need to define the API taxonomy and metadata for APIs up front, see Figure 2. Having clearly defined taxonomy and metadata would allow us to communicate among business stakeholders and development teams, and help us analyze the business processes and data, and properly model the services and APIs. With a defined taxonomy and related metadata, we were able to break large business processes and data into smaller function pieces, and model them as APIs. We then implemented them as services, and exposed them as APIs.

Category	Category Values	Definition
Scope	Enterprise	The services that can be broadly used by multiple business domains. Example: user logon service
	Domain	The services that can be used by multiple areas within a business domain.
	Project	Services with a capability specific to some project.
Function Area	Business Process	Handles business process logics. May also deal with some data, but focuses on business logics.
	Master Data	Processes (CRUD) any Master data defined by the Enterprise Data Taxonomy.
	Transactional Data	Processes (CRUD) any transactional data defined by the Enterprise Data Taxonomy.
	UI Component	User interface components. Example: font blocks, style sheets.
	Security	Services provide security related functionalities.
	Utility	Reusable assets for Utility. Example: email, IM, IT support tools.
Form	Intel API	API provided by Intel IT
	Vendor API	API provided by third-party vendor.
	Code Library	Reusable code libraries.
	Code Block	Reusable code blocks.

Figure 2. *Intel IT's API taxonomy.*

Next we knew it was important to model core APIs that could be used by many enterprise applications. To do this, we developed three categories of core APIs: Master Data APIs, Security APIs, and Utility APIs. These core APIs provided immediate value through broad reuse and shortened application development time. We have built more than 80 core APIs since developing the categories. As a result of developing the applications based on these reusable APIs, TTM has been greatly reduced. Figure 3 shows some sample of our core API usage.

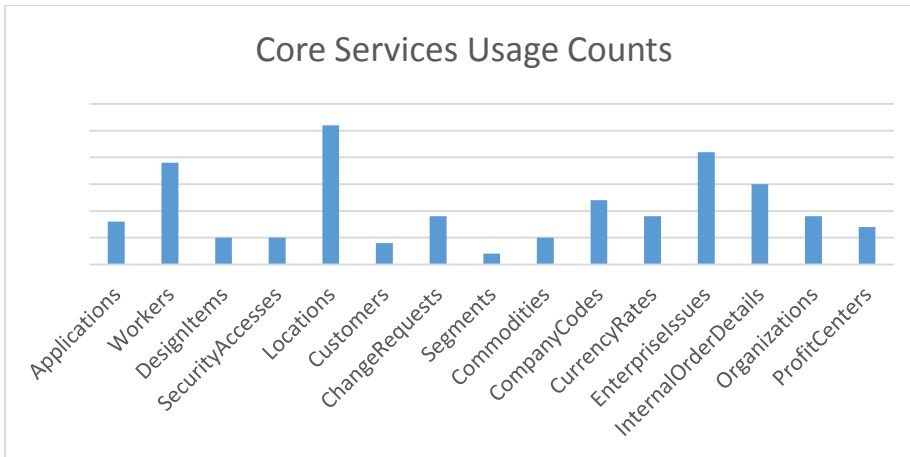


Figure 3. Intel IT's Sample core API usage.

Finally, we realized the importance of having an API management capability for registering, finding, and managing APIs. With the number of APIs increasing, we needed a better ability to control API calls, throttle the API call volume, track the call traffic, and measure and report the API calls. Of course, we also needed to properly secure the APIs. A good API management product would provide all these key API management capabilities. Figure 4 illustrates a conceptual diagram of API management and usage.

Intel IT developed the Enterprise API Registration Application (EARA) and the IT Developer Zone (ITDZ) Service Portal to provide some of the API management capabilities. API owners use the EARA to register their APIs. Application developers use the ITDZ Service Portal to search for, and request to use the APIs. As we continue on our API journey, we are deploying a vendor API Management tool to meet our growing needs.

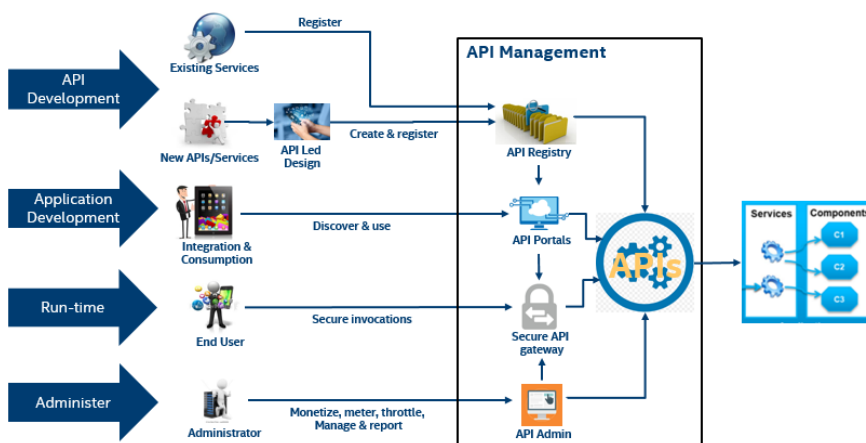


Figure 4. API management and usage concept.

Lessons Learned

To embark on the API Economy journey, and start API based application development, we had to change our mindset from the traditional way of building applications to the API

approach. Instead of thinking of an application as one big thing, we had to think of it as an assembly of smaller APIs and services. From there, we designed the capabilities as reusable APIs integrated together using a loosely coupled framework.

Once we built the component APIs, we designed applications as containers, and orchestrated the actions and responses of APIs to deliver the desired results for the business and customers. We also exposed selected APIs to external partners for business expansion and integration when necessary.

Changing the mindset or our development culture was much more difficult than simply planning the development process. Intel IT had chartered a dedicated program to drive our API adoption. We not only drove the architecture change, API management capability, and technology toolsets, but more importantly we also developed standards, guidance, and training courses, providing guidance and training to our development community to help anyone be successful on their API journey.

Summary

The API based application development and integration approach is useful for any company. APIs enable easier integration with customers and partners business applications, and support mobile applications better. The API approach helps reduce the cost of application development and integration through reuse, and gains the TTM advantage, which enables a business to get to market quicker. The approach is especially important for companies that want to expand business onto the Internet, and participate in the growing API Economy. Intel IT is actively using APIs in our business application integrations with internal and external partners. As our journey continues, we are beginning work on our Microservice and Container strategy. Our API architecture framework, experience, and learnings can help other companies start their own API Economy journey.

References

Welcome to The API Economy:

<http://www.forbes.com/sites/ciocentral/2012/08/29/welcome-to-the-api-economy/#5076f836d399>

<http://www.forbes.com/sites/ciocentral/2012/08/29/welcome-to-the-api-economy/#5076f836d399>

API management definition:

<http://searchcloudapplications.techtarget.com/definition/API-management>

Top API management vendors:

<http://www.informationweek.com/cloud/platform-as-a-service/forrester-names-top-api-management-vendors/d/d-id/1316520>

Authors:

Jian Wu has over 30 years of industry experience. His career spans cross the Factory Automation, Process Control, Supply Chain, Product Engineering and IT enterprise application architecture design, development and integrations. Currently, Jian is responsible for setting and governing Intel IT enterprise application integration capability strategy, standards, and technical guidance.

Mark Boucher is a 30 year veteran in large scale enterprise software. Currently Mark is a principal engineer setting the software architecture direction for sales and marketing organizations.

Don Meyers has over 30 years of industry experience leading the architecture, design, and deployment of enterprise applications, IT infrastructure, and collaboration systems. He has patent filings for various computer systems, including: Network, Data management, Collaboration, eMail, Search security, and Perceptual computing.

Notices

No license (express or implied, by estoppel or otherwise) to any intellectual property rights is granted by this document.

Intel disclaims all express and implied warranties, including without limitation, the implied warranties of merchantability, fitness for a particular purpose, and non-infringement, as well as any warranty arising from course of performance, course of dealing, or usage in trade.

This document contains information on products, services and/or processes in development. All information provided here is subject to change without notice. Contact your Intel representative to obtain the latest forecast, schedule, specifications and roadmaps.

The products and services described may contain defects or errors known as errata which may cause deviations from published specifications. Current characterized errata are available on request.

Copies of documents which have an order number and are referenced in this document may be obtained by calling 1-800-548-4725 or by visiting www.intel.com/design/literature.htm.

Intel and the Intel logo are trademarks of Intel Corporation in the U.S. and/or other countries.

*Other names and brands may be claimed as the property of others.

© 2016 Intel Corporation.