WHITEPAPER

MuleSoft

CLOSING THE IT DELIVERY GAP

MAKE YOUR IT TEAM MORE PRODUCTIVE AND INNOVATIVE

EXECUTIVE SUMMARY

Challenges

- Responding to technological change and digital disruption places a massive burden on enterprise IT teams, creating a gap between what the business needs and what the IT team can actually deliver.
- Resourcing for IT is expected to stay the same or rise very slightly, meaning the IT delivery gap cannot be solved through additional resourcing alone.
- If enterprises don't respond quickly to these changes, it is believed negative repercussions could occur in six months or fewer.

Recommendations

- Rethink IT's role in the enterprise from a sole deliverer of projects to a strategic partner and enabler of the rest of the business self-serving its own technology deliverables.
- Consider established business models, like franchising, when determining how IT can scale its delivery capacity.
- Establish both a strategic discipline of integration, led by APIs, and a capacity within the business to encourage and make visible reusable assets on which the entire organization can collaborate and utilize to deliver new products and services.

INTRODUCTION

Many business leaders think that we are currently going through the "fourth industrial revolution." Our current age, marked by a mix of technologies that are blurring the boundaries between the physical, digital, and biological, is considered to be just as disruptive — if not more so — as the agricultural, manufacturing, and electronic revolutions before it. Klaus Schwab, the founder of the World Economic Forum, outlined the hallmarks of this revolution:

"The possibilities of billions of people connected by mobile devices, with unprecedented processing power, storage capacity, and access to knowledge, are unlimited. And these possibilities will be multiplied by emerging technology breakthroughs in fields such as artificial intelligence, robotics, the Internet of Things, autonomous vehicles, 3-D printing, nanotechnology, biotechnology, materials science, energy storage, and quantum computing."

While these technological advantages have the potential to offer significant benefits to the world, they present extraordinary challenges as well. For the enterprise, the challenge of the emergence of all these technologies is particularly profound. These forces are not just a question of a seismic change in business model, like the way e-commerce disrupted the retail industry in the 1990s. The emergence of all of these forces at once is upending the way industries do business itself; it is starting to create a sea change in enterprise culture.

The qualities necessary to survive in today's enterprise environment are different from before. Organizations today have to contend with demanding stakeholders that engage through new channels, competition coming from everywhere due to lower barriers to entry, and a truly global marketplace. No longer does the big eat the small, as in years past; today, the fast eat the slow. The qualities that organizations need to succeed today are speed, agility, inventiveness and the ability to try many experiments rapidly and fail fast.

How can you make sure that your business has these necessary qualities to survive and thrive in today's competitive environment? How can you make sure that the IT delivery gap in your organization is as small as possible, enabling your technology team to step away from the reactive hamster wheel of projects and developing a truly nimble, innovative infrastructure for change?

The solution lies in our unique vision of enterprise architecture, the application network. An application network emerges from our approach to enterprise integration, called API-led connectivity, and our equally unique approach to organizational structure and application delivery, the Center for Enablement. The two go hand in hand. We will explore these concepts in further detail, and provide solutions to create a powerful capability for change in your organization that will close the IT delivery gap.

What is the IT delivery gap?

In January 2017 MuleSoft conducted a study of over 900 IT decision makers, and the trends they revealed were startling. Only half said that they were able to complete all the projects they were asked to do in the last 12 months. 60% said they were not adequately resourced to meet their company's digital goals. 60% also said that if digital transformation goals were not met, revenues would be negatively impacted in six months or less. And the single biggest obstacle to meeting those goals was time constraints.

It's not particularly difficult to understand why. As businesses are pivoting to respond to more and more changes in strategy and interaction forced by technological trends like cloud computing, mobile, and SaaS, IT has to shoulder the responsibility of implementing those responses. And because there are so many forces, and IT only has a finite amount of resources, what will ultimately occur is a gap between what IT needs to do and what it can do.

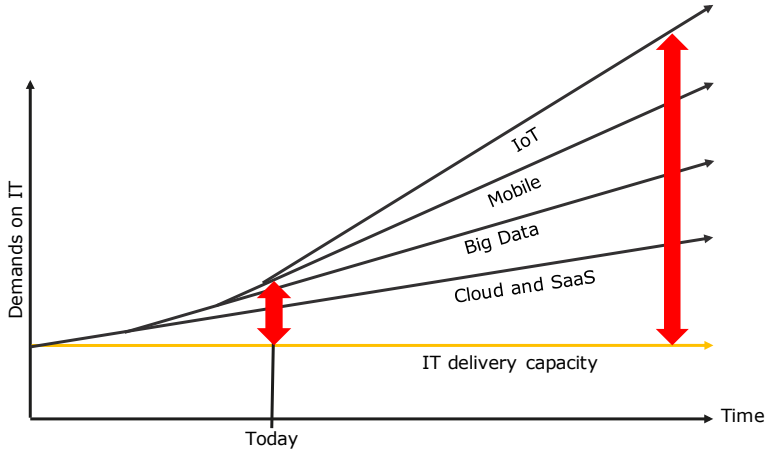

Figure 1: The IT delivery gap

With each new digital pressure coming on and with IT remaining unable to add any more resources or specialized capacity, the gap is widening exponentially and is accelerating. Moreover, under pressure, IT often will take short cuts and create technical debt, even while complexity is increasing, which compounds the problem. And this only addresses the technological change that businesses have to respond to today. IT decision makers believe that their businesses will change more in the next five years than they have in the last five. How will the IT team be able to accommodate changes that you may not be able to predict today?

Clearly a new approach to project delivery and technology strategy is required, one that can scale IT's capacity. It is clear that it would be impossible to keep adding resources to close the gap — what's needed is a way to exponentially increase the delivery capacity of existing resources. It seems like an impossible task, but it can be done through a rethink of IT's role in the business. In short, IT needs a new operating model.

The new IT operating model depends on established business principles

The new IT operating model employs an approach by which Central IT drives both the production of reusable assets and, importantly, enables the consumption of those assets by teams within the lines of business, to deliver digital initiatives. The model addresses production and consumption, but what is new is what is being produced as well as what is being consumed. This new IT operating model relies on business principles that have been around for decades, though they haven't often been applied to enterprise IT project delivery. And that model is franchising.

Think about a business like McDonald's, who, in many ways, were pioneers of the franchise business model. The central company has core assets such as recipes, retail layouts, and marketing offers which they package up for reuse by their franchisees, who execute to those plans in a self-service way. This encourages innovation at the edges by federating reusable assets and intellectual property for self-service and consumption by each individual franchisee. The company enables their franchisees or vendors by educating them with best practices and processes, and promotes quality both through established and exacting standards and through continual consumer feedback. Finally, the parent company retains control over the assets and has full visibility into the entire operation but federates work and innovation. This business model has two distinct advantages. First, the franchise model allowed McDonald's to scale and expand globally at a much faster pace than it would have otherwise. Secondly, even though every McDonald's all over the world recognizably belongs to the same business, franchisees in local markets can innovate according to their own business needs (e.g. localized items on the menu), increasing the chance of business success in markets in which it has expanded to based on local knowledge.

For IT organizations to scale to the level required by digital disruption, they must think about how they can adopt the principles of franchising. Franchise businesses have both core reusable assets (the what) and organizing principles (the how) that franchisers unlock and package for self-service consumption by their franchisees to drive scale and innovation, while maintaining quality and governance to exacting standards. In the new IT operating model, Central IT needs to morph its role into one that unlocks assets and enables consumption by lines of business and developers throughout the organization, to create scale, innovation, and speed, while retaining control.

How the new IT operating model should work inside the organization

The idea is that, much like a franchise business, Central IT should federate innovation across the organization by building reusable assets and enabling self-service rather than working on and delivering entire projects themselves. By leveraging the resources and capabilities outside of central IT, organizations can make a step change in delivery speed and capacity.

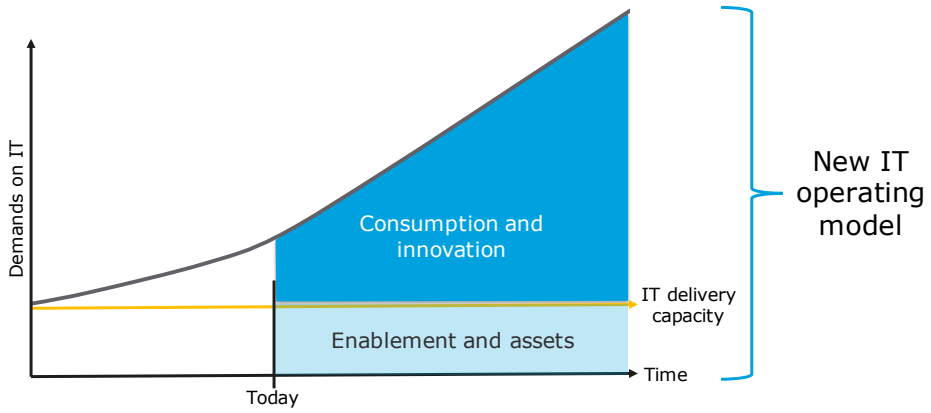

Figure 2. How IT scales with a new operating model

But what this actually means is that a change in the way that IT operates is necessary; rather than just delivering on projects, Central IT will spend its time creating reusable assets and enabling the rest of the organization to use them.

How does this actually happen in practice? There are three things that tech leaders have to consider as they work on scaling the IT organization. These seem like big changes, but the work done to put these initiatives in place will actually pay dividends in terms of increased productivity and innovation in the future.

- Changing your delivery model
- Changing your approach to integration
- Changing the way your IT team is organized

Changing the delivery model from production to production + consumption

As we now know, Central IT needs to move away from trying to deliver all IT projects themselves and start building reusable assets to enable the business to produce some of their own projects — to self-serve rather than waiting for IT to deliver. The key to this strategy is to emphasize consumption as much as production.

Traditional IT approaches (for example, SOA) focused exclusively on production for the delivery of projects. In the new IT operating model, IT will change its mindset to think about producing assets intended to be consumed by others throughout the business. The assets need to be discoverable, and developers need to be enabled to self-serve them in projects through a visible, usable repository and best practice documentation.

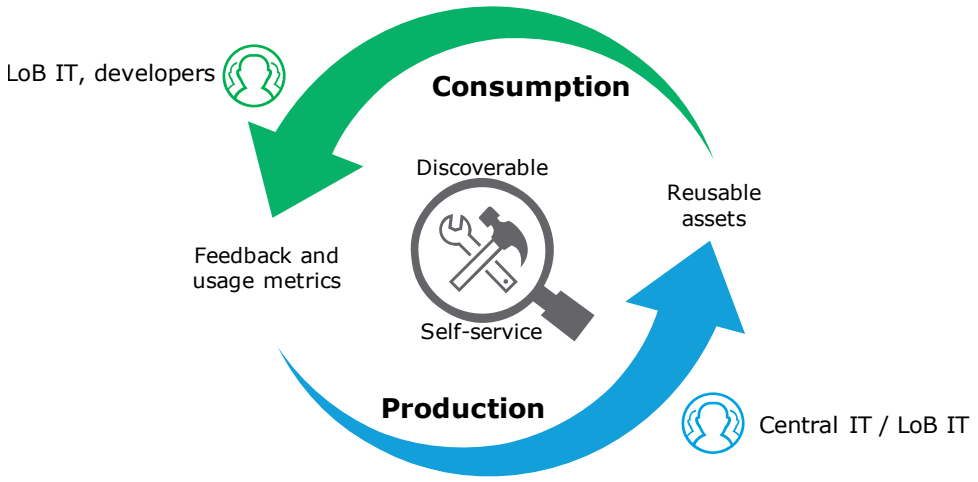

Figure 3. The production + consumption model

One of the best parts of this new operating model is that the most useful assets to the rest of the company will rise to the top thanks to active feedback from the consumption model along with usage metrics to inform the production model. The reusable assets that are most useful become the basis of numerous project deliverables that can be completed more quickly. Mike Hamilton, the director of IT at MuleSoft, explains how this works in practice:

> We took the entire project list of business processes that needed automation and decided to break each one down into pieces and then group the pieces into common services. What we found in doing this was that quite a lot of the projects had many needs in common. A key common microservice that came up was related to sending notifications via email, Slack, or SMS. Since most integrations had this requirement, the "notifications" microservice came up early as a quick win. The function of this microservice is to be able post information about who to notify, how to notify them, and the content to send in the notification and then have that delivered to the intended recipients. By creating this as a service, every project can leverage the capability of the microservice rather than implementing notifications as a part of each individual project. This saves development time on every project and provides us with a consistent way to implement notifications in all of our work.

By figuring out what all the delivery requests had in common, Mike and his team were able to develop a useful, reusable asset that will save both his team and any other team that has access to it time and effort in delivering any project necessitating notifications. The model has successfully been changed from pure production to consuming a reusable asset and then using that to speed production.

Changing your approach to integration

Julian Burnett, the CIO of House of Fraser, a premium department store in the UK, has noted that for many businesses, integration is not really a strategic discipline. "Most integrations between business systems, if I'm honest, occur point-to-point in a really rudimentary way. We hadn't really thought about integration as a capability." But as House of Fraser's technological capabilities needed to evolve in order to accommodate the massively changed and disrupted retail business, he realized that he would have to rethink his company's approach to integration beyond the usual point-to-point connections, in order to increase the speed of change and to de-risk the effect to current business operations.

What many businesses find, as they pivot to developing apps and other ways of responding to digital disruption, is that the point-to-point integrations required to make these apps work pile up over time, creating technical debt, a brittle architecture, and the inability to respond to changing business needs quickly.

Here's an example. You might wish to develop a web app to provide real-time order status and order history for sales teams to engage with customers. For this example, let's assume you have customer data in SAP and in Salesforce; inventory data in SAP; and order data in an ecommerce system. What might be commonly done at this point is that you would get your IT team to jump in and create aggregated customer data by wiring together customer data from both systems – with code. Then, the aggregated customer data is further combined with order data in the ecommerce system to produce both the order status and order history data – with more code. Now these two sources of data are hooked into a Web app API which can be leveraged by the web app.

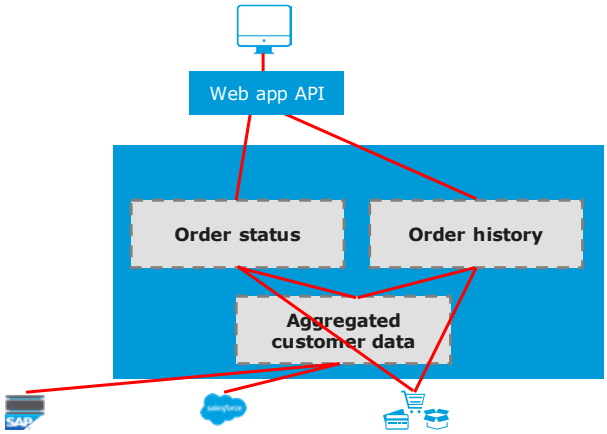

Figure 4. A project-based approach to IT delivery

This project might be considered a success; it was launched on time, on budget, and has the correct functionality. But then the sales team, who are often on the road, are demanding that this functionality be available on their mobile phones. So, the IT team is now tasked with building a mobile app. But the developers building the app aren't able to use any of the work that was done for previous projects. So they have to redo all the work, which in itself is not a great outcome.

Even though the developers know this is likely a short-sighted approach, they justify it given the typically intense time pressures. If there are consultants involved (as is typical), the problem gets worse, as they have little incentive to think about the long term. Over time, changes become very expensive or near impossible to make. But as change is constant, agility is now made very difficult. As you can see below, the familiar "spaghetti code" pattern begins to take shape.

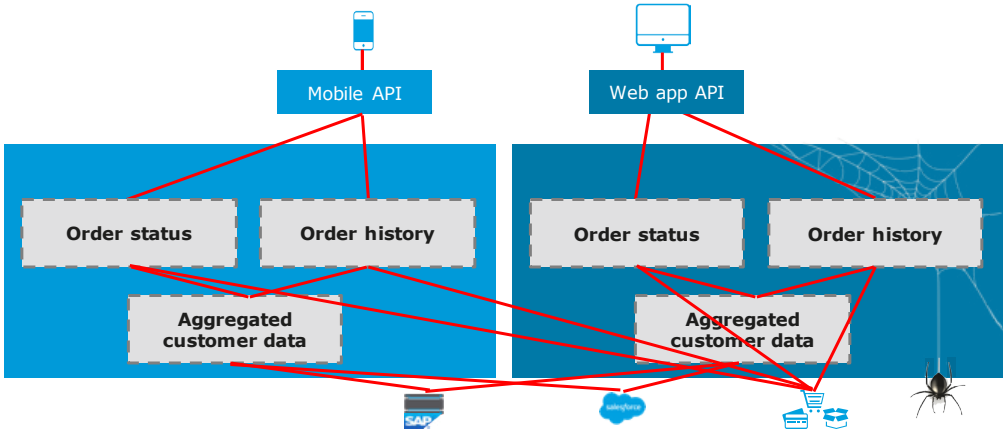

Figure 5. Too many shortcuts lead to the emergence of spaghetti code

With all subsequent projects, more non-reusable code is built anew. This deprecates all the learning and IP from the first project, demotivates developers, creates an unmanageable, duplicative, and expensive code base, and lays the foundation for strangling agility. The apps are now tightly coupled to the underlying endpoints; if anything changes with the endpoints or the business requirements, the entire app or major portions of the code need to be rewritten.

The next application to be built that uses similar data will have to be built from scratch, which means development teams will need the same system experts to get the data out of the source systems, which can add time to project delivery. And, as nothing is registered for security or governance or management, you have lost any visibility as to what is actually going on, which over time, can create security vulnerabilities.

A better way of going about this is through an strategy called API-led connectivity, which enables the new IT operating model. When we think about transitioning to a production + consumption model, core capabilities must be packaged up for consumption. Those core IT assets and capabilities are packaged in a well-designed API.

API-led connectivity is a methodical way to connect data to applications through a series of reusable and purposeful modern

API-led connectivity is a methodical way to connect data to applications through a series of reusable and purposeful modern APIs that are each developed to play a specific role – unlock data from systems, compose data into processes, or deliver an experience. With this approach, rather than connecting things point-to-point, every asset becomes a managed API – a modern API, which makes it discoverable through self-service without losing control.

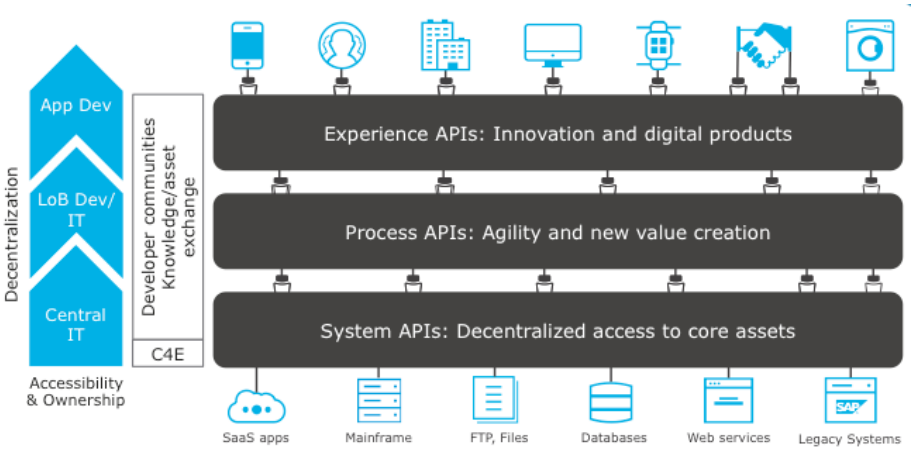

Figure 6. The API-led approach to connectivity

This approach to connectivity classifies APIs in three types:

Systems APIs: Your core systems of record (e.g. ERP, customer and billing systems, proprietary databases etc.) are often not easily accessible because they have proprietary connectivity interfaces and require highly specific skills, which are not readily available within the company (typically a major bottleneck). System APIs provide a means of insulating the user from the complexity or any changes to the underlying systems. Once built, many users, can access data without any need to learn the underlying systems and can reuse these APIs in multiple projects, such as a customer API to be leverage in a mobile app and a partner portal. System experts are not needed for every project that needs access to that system and central IT maintains and governs these APIs, given the importance of the underlying systems. The scope of a System API is to open up and provide consistent managed access to the underlying data. System APIs do not try to normalize the data too much or combine data from other sources.

Process APIs: Naturally, you would have a view of the customer, and order status, prescription status, etc. in this layer. These APIs interact with and shape data within a single system or across systems (breaking down data silos) and are created here without a dependence on the source systems from which that data originates, as well as the target channels through which that data is to be delivered. For example, customer information may be exposed across 2 or 3 system APIs, and you would create a view of the customer by composing fields from those 3 system APIs. This view of the customer might be a canonical model if you have one or it may be specific for the domain the API lives in. This customer API is more consumable because it narrows down all the possible information to just the things need for a set of applications. It's also easy to add new fields or create a different view of the customer for a different domain.

Experience APIs: Data is now consumed across a broad set of channels, each of which wants access to the same data but in a variety of different forms. For example, a retail branch POS system, eCommerce site, and mobile shopping application may all want to access the same customer information fields, but each will require that information in very different formats. Experience APIs are the means by which data can be reconfigured so that it is most easily consumed by its intended audience, all from a common data source, rather than setting up separate point-to-point integrations for each channel. An Experience API is usually created with API-first design principles where the API is designed for the specific use experience in mind.

With this approach, you have essentially "Lego-ified" your business, allowing you to compose, recompose and adapt these building blocks (APIs) to address the changing needs of the business.

So, with API-led connectivity, when your teams are tasked with building a new mobile app, you now have reusable assets to build from, eliminating all of the work needed to build them. And it is now much easier to innovate — in this case, adding shipment status information — in much the same way as you accessed order status and order history.

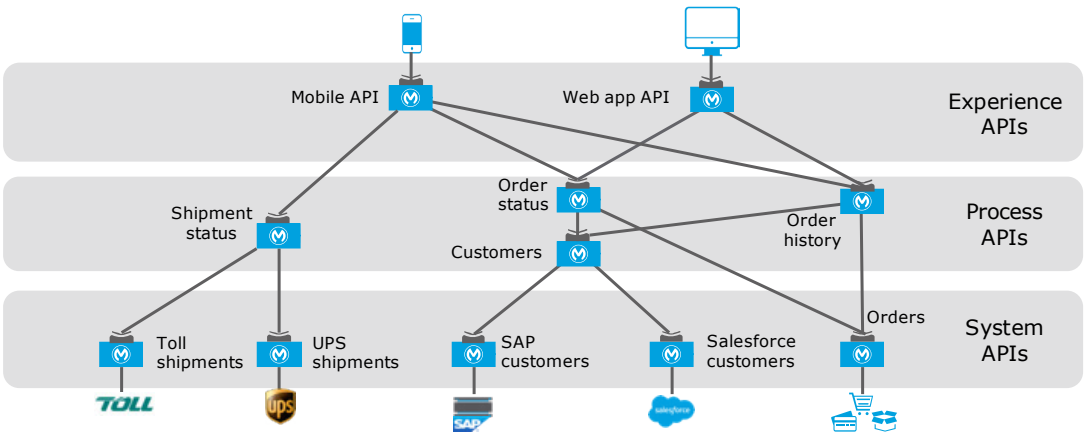

Figure 7. Web and mobile apps built with an API-led approach

An API-led connectivity approach to delivering IT projects ensures you are not only on time and budget with your first projects but you have created reusable assets that will save your company time and money, created an infrastructure which is designed for change, built in visibility, compliance and governance and, most importantly, met the needs of the business, which is long term sustained agility. It enables you to move fast on your first project, but then actually accelerate further from your second project onwards, due to reusable assets and a built-up organizational capability; API-led connectivity liberates resources, allowing you to innovate and to move quickly.

Changing the way your IT team is organized

We have discussed the delivery and technology infrastructure parts of the new IT operating model, but those on their own aren't enough. With a major culture shift that the new model requires, there needs to be a people and business process component as well. We suggest doing this by establishing a cross-functional team called a Center for Enablement, or C4E.

The Center for Enablement (C4E) is a team dedicated to enabling the organization to consume and get full value from the assets. It is typically staffed with members from Central IT, Line-of-business departments, and digital innovation teams and is charged with productizing, publishing and harvesting reusable assets and best practices. They promote consumption and collaboration, and help drives self-reliance while improving delivery through feedback and metrics. It captures the notion that there is balanced deployment approach to each project. The team decides if the project can use a reusable asset and if it should create reusable assets for future projects or if it is a one-off project.

The name might sound similar to an older concept — the Center of Excellence (CoE). However, C4E is a radically different idea. A traditional CoE would feature centralized expertise and knowledge, which would result in information being protected and rationed. CoEs often become bottlenecks which make developers and architects work around them. In contrast, the C4E model focuses on putting valuable assets in the hands of development teams across the organization, encouraging them to not only utilize the assets but add and improve assets as well.

As projects are initiated and developed, a core set of assets such as APIs, templates, common platform capabilities, like logging, caching and error handling as well as documentation, code samples, videos, become the initial asset base for the C4E. These assets must be productized for self-service consumption and published to a central repository like Anypoint Exchange so the assets can be discovered and utilized.

Then, as new projects come along (like the order status web application described earlier), the C4E will engage those project teams and encourage collaboration and reuse of existing assets. As the asset base grows, developers on the innovation teams throughout the organization can be self-reliant. They can find those assets, accelerate the development of their mobile app, add their own API assets for shipping information and feed that capability back to the C4E. This shows the multiplicative power of the community and its concomitant effect on IT delivery capacity.

Unilever, one of the world's largest CPG companies, has used a C4E and achieved great business outcomes. With its large and ever expanding portfolio of products, it was becoming an increasingly difficult process for Unilever to integrate systems across seamlessly for business needs like supply chain management and sale processes. During the last three years, the company started taking steps to decommission its legacy tools while speeding up its integration processes in parallel. Frank Brandes, Unilever's director of global enterprise business integration, started off by creating a Center of Excellence

it served as a bottleneck to business units. It came to be understood that Central IT would need to maintain a degree of control while giving business organizations enough freedom to innovate. The challenge here, Brandes said, is that "If you have too much control, you kill innovation. If you don't have enough control, you don't get the reuse, and you have people moving off in different directions."

Under those circumstances, an Adaptive Integration Capability team (a C4E) was created. It functioned like an internal integration consulting group, going around business units and solving their integration pain points. The team also aimed to provide reusable services on API platforms to shift the focus away from individual assignments to reusable platform capabilities. Not only do business groups have resources for API-led solutions in their possessions now, but the entire company also reaps the benefits of this flexible infrastructure. For example, after Brandes' Adaptive Integration Capability team was created, Unilever completed its SAP ERP integration project and reduced its cost by 26%.

The result is an application network

Making these three changes results in something very powerful emerging — an application network. An application network is a network of applications, data and devices connected with APIs to create reusable services and making them pluggable and unpluggable easily and as business needs require. APIs allow for any digital asset to be quickly and securely discovered and reused by consumers on the application network.
The salient features of an application network are that it:

- Emerges bottoms-up via self-service
- Creates visibility, security and governability at every API node
- Is composable and recomposable: it bends, not breaks. It is built for change.

Ultimately, an application network sets organizations up for speed, agility, and innovation.

Building an application network is not an activity that must take place outside of normal project delivery. Instead, it takes shape through adopting the API-led approach. From the very first project, the nodes of the network are built with all the intrinsic qualities of the network in them – secure, easy to change, discoverable, self-served, ready for reuse, modular, and composable.

Every new node in the network adds exponentially increasing value. Each node is secured by design, ready for self-service and discoverability. For your second project, and every subsequent project, you are delivering against project requirements while at the same time creating more of these reusable self-serve assets on the network.

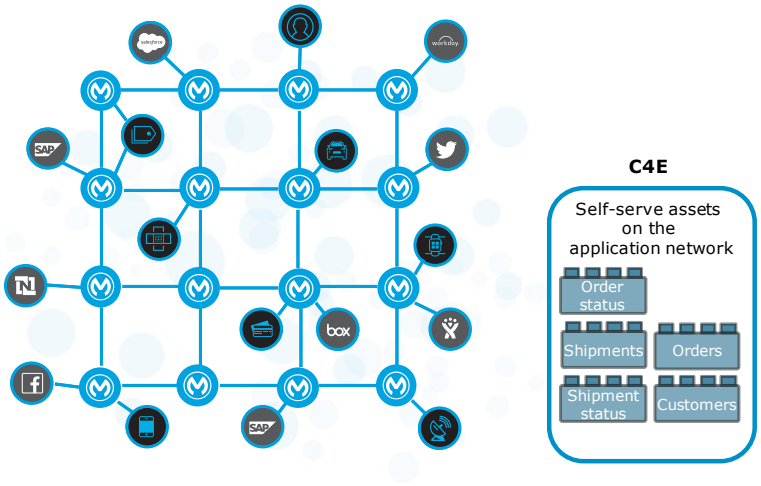

Figure 8. Emergence of an Application Network

These nodes get added from each new project and the value of your application network grows exponentially as more of your data is unlocked. What emerges now is your application network in which the nodes are inherently secure by design, they are easy to add and change. With this the speed with which every subsequent project is delivered begins to accelerate.

The application network is built as a natural yet powerful byproduct of day to day project work done using the API-led connectivity approach with an intention to expose assets that are discoverable, self-served, and consumable by the broader

organization. The nodes are highly valuable pieces of a composable enterprise. They are like Lego blocks, which can be used to create an enterprise IT self-serve delivery model which drives speed and creates agility. Given that changes can be accommodated at the node level without breaking the entire architecture, the application network encourages innovation and a mentality to experiment and fail fast.

Why MuleSoft?

Many of these concepts are easy to imitate, and the messages are often repeated by other companies in the market, but MuleSoft is the only company that puts all these pieces together in a comprehensive and proven way. We are the only vendor in this space that offers the capabilities to achieve the new IT operating model that will allow our customers to achieve speed, agility, and innovation.

MuleSoft's Anypoint Platform is the only solution that enables end-to-end connectivity across API, service orchestration, and application integration needs through a single unified platform. This allows developers to rapidly connect, orchestrate and enable any internal or external endpoint. The result is a 2x to 5x faster time to launch new initiatives, connect systems, and unlock data across the enterprise and a 30% reduction in integration costs.

Furthermore, unlike alternatives, MuleSoft's Anypoint Platform can be rapidly deployed on-premises, or accessed as a cloud solution. Since MuleSoft's solutions are easy to use and understand, any developer can quickly become productive without lengthy training in vendor-specific technology resulting in 10% higher employee productivity and 70% higher productivity for app development teams.

Finally, MuleSoft's experience in partnering with our customers to drive digital transformation initiatives gives us unparalleled expertise in change management, organizational design and IT development best practices to complement our technology offerings and truly drive lasting success.

Conclusion

Our research reveals that the IT delivery gap is a real problem for every enterprise, and it isn't going away anytime soon. Yet if organizations don't figure out ways to respond to changing technologies, there will be consequences. Everyone knows they have to change, but the questions remain: What are the best practices? How do I get started?

Organizations ranging from Unilever to Coca-Cola to major banks have all used MuleSoft as a partner in their digital transformation journeys; MuleSoft enables the disciplines of reuse and the consumption model, allowing increased efficiency, productivity, and scalability at never-before-seen levels.

Contact us to find out how MuleSoft can help your company digitally transform.