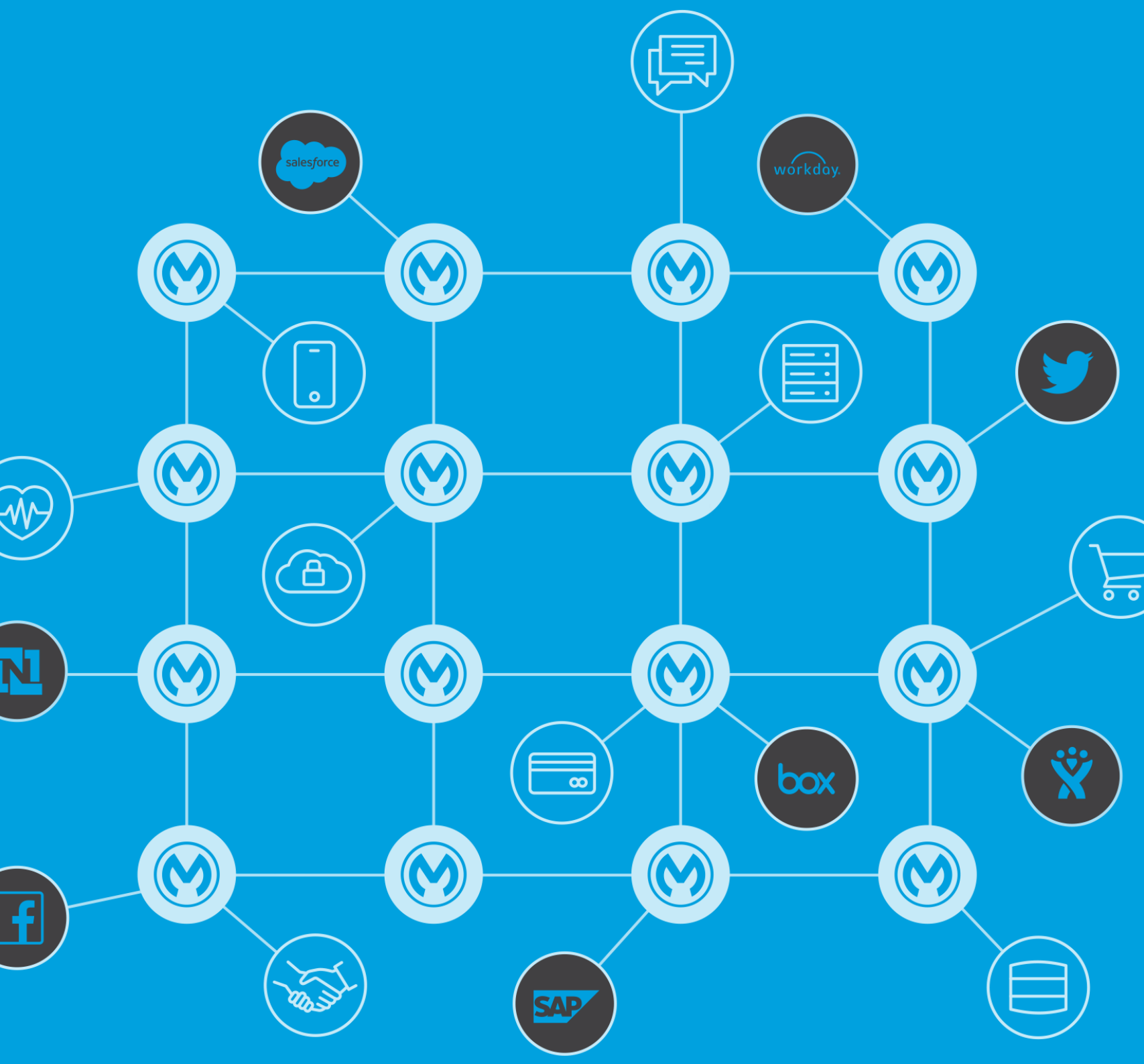
MuleSoft

WHITEPAPER

The application network

What it is and how it works



Contents

Introduction 3

What is an application network? 5

The key aspect of an application network: Reuse 7

Success depends on self-interest 9

Benefits of an application network 12

Capabilities of an application network 14

The key design principles of an application network 16

How is an application network different from SOA? 18

Conclusion 20

About MuleSoft 21

Introduction

We are in the "fourth industrial revolution"[1] where the forces of digital disruption deeply affect businesses of every size and in every industry. It's an era where all parts of the enterprise function using software, SaaS, and custom apps. More than ever, consumers have lofty expectations and are willing to shop around. And if that wasn't enough, the barriers to entry for each industry have evaporated. Startups everywhere are disrupting all types of products from financial services to crowd-sourced insurance to customizable health care to unique shopping experiences. A new idea can be built in a hackathon and developed in the garage using major building blocks like unlimited cloud compute power, free dev tools, open APIs, and open source components. These new entrants build fast, move fast, and continue to educate consumers on the type of experience they should be getting. Consumers, employees and partners now expect everything to be on demand and fully connected.

There is not a single business in the world unaffected by these changes, but large organizations with calcified processes and monolithic infrastructures are hit the hardest. Fortunately, the means for real business transformation are within reach. An application network can help an organization be more agile, change the clock speed of business, and respond to changes in the market, the business, and the consumer without having to break things.

Companies need to launch new solutions that innovate in order to provide new products and experiences to meet customer demand, get ahead of the competition, and make their businesses more efficient. Actually producing these

[1] Wladawsky-Berger, Irving. "Preparing for the Fourth Industrial Revolution." The Wall Street Journal, February 26, 2016.

solutions, particularly in larger, older organizations with a technological or cultural legacy, can be difficult. Customers, employees, and partners are demanding ever faster innovation; if a business can't deliver, there is a real risk of losing customers, surrendering market share, and missing growth targets. MuleSoft conducted a recent survey of 800 IT professionals and found that 66 percent say they are under "drastic" pressure to deliver technology projects across the business; a similar number say that change has to occur[2] in order to meet business demand.

The problems are compounded further by the increasing number of people operationalizing change, particularly those not in IT. Businesses who are using and deploying technology in traditional ways cannot respond quickly to change. In addition, the creation of and alterations to monolithic business processes is time consuming and complex, creating paralysis rather than agility.

IT cannot keep up with changing business needs if it continues on a traditional path. If IT continues to work as a sole technology provider for the business, it can neither integrate systems quickly nor leverage the big data now coming from all parts of the company's operations.

Radical transformation can happen if you create smaller, agile units for business, processes and underlying technology, while still maintaining stability and control. To effect the kind of change that will allow organizations to compete in a truly disruptive business environment, we propose a new vision of enterprise IT: the application network.

[2] Mulesoft.com. "MuleSoft survey finds top 5 reasons APIs are taking centre stage for businesses." January 12, 2016.

What is an application network?

Application networks seamlessly connect applications, data and devices. They take a different approach to the methods used to connect applications, data, and devices today. Instead of utilizing point-to-point connections or isolated architectures, the application network provides an infrastructure for information exchange by allowing applications to be “plugged” into the network. The network can be as simple as two nodes that enable two applications to share information, or it could span across the enterprise as well as to external ecosystems. Designed to honor Metcalfe’s law, every new node added to the network will increase the network’s value since the data and capabilities of that node are made discoverable and consumable by others on the network.

Every new node added to the network will increase the network’s value since the data and capabilities of that node are made discoverable and consumable by others on the network.

Unlike prior approaches to connecting applications in the enterprise, an application network is designed to allow many people inside and outside the enterprise to have controlled access to valuable business data. This will occur by allowing anyone in the business to use consumption models they are familiar with. In other words, it makes it easier for someone in the organization to create a useful application, use of data, or an API creating a particular experience, and then expose that value to the network.

For example, someone might have found a way to expose the organizational structure of a group of people in the company, perhaps by tapping into the HR services provider system the

company uses to manage its employees. To do this, they may have used the HR services provider's API and some credentials to pull in the org chart below a particular executive or by function. Or they may have wrapped that HR services provider's native API with a friendlier API optimized for consumption in certain use cases. They likely have done so as a part of a bigger project: perhaps to create a portal of skillsets, or to manage access to source code repositories, or to synchronize with a disaster recovery contacts database that needs to be available offline.

If this work is made available beyond the scope of this project — e.g. by making the friendlier API available to others, or allowing the connection credentials to be reused, or by capturing the project in a template — then the provider's service is exposed to the network and may be leveraged in other scenarios. Some of those use cases might be a team needing to create dynamic visualizations of resources applied to company objectives, whereas another team might need to automate the laptop requisitioning request process. Through the application network, they can more easily access the HR services provider's employee subsystem through the work of the original team, without the explicit help of that team.

The key aspect of an application network: Reuse

The key aspect of an application network is reuse; a node of the network must be reusable elsewhere via its connection to the network. For instance, in a social network, the information of a member — whether in the form of the content they post, or in their rating of existing content (explicit or implicit), or in various other signals they provide — is then reused across the network.

A social network member's content is viewed by many others; that content can then be shared with others on the network. Those ratings and other signals inform what content is promoted to yet more members; in other words, they are reused by the larger subset of the network. The network enables reuse by other members whether those members understand their work has been reused or not. This induced cooperative behavior provides the classic network effect, and is the core source of value of the network itself, atop the value of each application to its direct consumers.

As in other networks, the value of exposing information from and allowing access to the application network comes at a price; not all information should be exposed to everyone, and not all capabilities should be made available to just anyone. In a social network, members may restrict certain content to be viewed by only some of their connections; they may only allow certain people to follow them, or to post to their timeline. Attention must be paid to the transitive nature of this access. If I grant access to a friend, and they grant access to another friend, does that friend-of-a-friend (FOAF) also have access, perhaps on a more limited basis?

For an application network in the enterprise, this manifests via APIs and the transitivity of API access. For example, an

employee management system API allows access to employees and their data based on entitlements assigned to the API's clients, and perhaps to the end user on whose behalf the clients are accessing that API. The team that integrates with the system API will need to register as a client of that API, and they will need to figure out what entitlements to set up and establish which end user(s) on whose behalf it will access that system API. And if this team in question offers its own API which makes use of the system API, it will need to think carefully what access clients of its API should have to the employee management system API.

Success depends on self-interest

Another key aspect of the application network, and one that affects the viral growth and stickiness of the network, is the degree to which it caters to the self-interests of each node. Facebook, for example, depends entirely on its ability to satisfy the innate curiosity of humans about other humans; in particular, it satiates their need to know what their friends (real or otherwise) are doing and offers engaging content to each member based on their taste as inferred from their friends' tastes. Any given member is engaging with the network primarily because of their self-interest, not because of an interest in contributing to the network as a whole. Members curate content for their friends knowing their friends will see it; they will judge their friends' reactions and rate their friends' content, which establishes reputation and self image. It is then up to the network and its dynamics to translate that to the value of the network as a whole.

A successful application network must find ways to align the teams' self-interest with the creation of reusable assets that benefit everyone.

This is equally true for an application network, in which the teams attaching the applications and creating valuable new services are usually most concerned about getting specific projects done, rather than creating reusable assets for the whole organization. A successful application network must find ways to align the teams' self-interest with the creation of reusable assets that benefit everyone. Just as a social network must allow members to keep some content private and some access limited, an application network must also allow some services, design assets, and APIs to be kept private, without

necessarily imposing on the teams the burden of maintaining them for a broad population of other teams to reuse.

Many factors can influence this balance between the autonomy and independence of each team and the value of their work to the broader community. There may be incentives placed to encourage building reusable assets, such as recognition by the community or management; constructive criticism that goes to increasing the quality of the asset; access to other services ("you can use my service if I can use yours"); funding of some sort to offset the cost of maintaining reusable assets, etc. In some companies, the reusability of assets may be imposed as a requirement. For example, a review process may be instituted to ensure all integrations and APIs that access an HR services provider are documented and registered as potentially reusable assets, and are reviewed for compliance with security and privacy concerns, usage limits, and so forth.

Whether assets are in fact reused will also depend on the self-interest of teams engaging with the network and consuming or contributing to it. In traditional enterprise IT initiatives based on Service-Oriented Architecture (SOA) principles, reuse was mandated from the top. Certain patterns and service interfaces and architectures were predetermined by a small group of enterprise architects and were required to be reused by all teams. While this approach may still make sense in some scenarios, increasingly it is being replaced by "merit-based reuse": potentially reusable assets are created by a broad population of teams in the course of implementing projects; those assets are then consumed or not by that same population based on the merit each team finds in them. This approach prefers the "wisdom of the crowd" over that of the central architects, who often participate by guiding or enabling the teams and potentially pruning assets rather than pre-determining everything.

The result is a much healthier, more valuable, and self-sustaining ecosystem of apps than would have ever been produced by a single controlling provider.

An analogy may be found in the world of mobile consumer apps. Rather than the traditional app stores based on a single telecom provider's creation and curation of mobile apps, the modern app stores allow apps from a broad base of developers and let consumers "vote with their feet" to determine which apps have more merit. Usually, as long as there is a vibrant community of both consumers and producers, the best apps rise to the top rather quickly and are reinforced in a non-linear way. The result is a much healthier, more valuable, and self-sustaining ecosystem of apps than would have ever been produced by a single controlling provider. The role of the provider, and of the application network architects, is to provide just enough enablement to seed the network and to enable and encourage best practices and just enough governance to create a trusted ecosystem and safety net for the organization.

Benefits of an application network

In order to serve the needs of the business and provide the benefits that organizations need to thrive in today's connected, competitive environment, an application network must be:

- **Clearly defined:** A well-defined operating model enables the business to get the most value from the IT assets on the network.
- **Broadly deployed:** It empowers the whole organization — not just IT — to discover and access the data and assets to fuel applications, reporting, analytics, business processes and automation.
- **Accessible and modular:** Every application is accessible and recomposable to fit the rapidly changing needs of the business.
- **Seamless:** Is a seamless network of applications, data sources, devices and APIs, whether in the cloud or on-premises, home-grown or acquired.
- **Elastic:** The application network is elastic and can grow or shrink depending on the demands of its consumers.
- **Scalable:** It allows companies to introduce new applications more easily because the application network already defines how to access, expose and share new application data.
- **Secure:** It can be organized to reflect the security and governance constraints in a business of any size.
- **Monitored end to end:** Data running through the network can be tracked end-to-end, monitored and analyzed. The impact of a request from one node in the network can be traced back to all the other nodes that take part in that request. Analytical: The network understands dependencies

between applications and can perform impact analyses on changes made to applications on the network. This gives organizations a real-time enterprise wiring diagram that can be inspected. Measurable: The value of an application network can be explicitly measured through a set of built-in KPIs that track reuse, utilization, adoption and redundancy.

- **Organic growth and reuse:** As the first applications are connected to the application, data and functionality is made available for reuse. IT and the business can plug in more applications over time. Each application adds more value to the network, more reusable building blocks for building applications, services, and business processes. The business and IT can use the existing assets to build new assets which are also made available through the application network.

Capabilities of an application network

When an application network is in place, it allows businesses to break down information silos as well as build and deploy apps and processes faster, all in a highly scalable manner. It creates room to govern and control while allowing the business to reconfigure the architecture easily.

The qualities of an application network allow for these capabilities:

- **Standard communication:** The base architecture for the application network provides a standard communication model which simplifies data exchange between applications on the network.
- **Security:** Any application connected to the application network will be subject to the same security policies and access controls. Different domains can have different security policies, which allows enterprises to segment and provide access to that data depending on the data source, consumers, geographic location, or other factors.
- **Reusable, modular components:** All data resources on the application network are made available through reusable interfaces, meaning that any application on the network is composable.
- **Layered governance:** The application network mandates well-defined interfaces to access resources. It also provides a layered governance model that makes it possible to track data consumption from consumer applications (such as mobile apps, web apps, dashboards, analytics) all the way through to the backend data. It can also track dependencies between applications and even perform change impact analysis.

- **Discoverability:** The application network is designed to exchange information between applications and people. This means that assets on the network are discoverable. In addition, different consumption models can be designed around different groups of users within or outside the enterprise.
- **Custom consumption models:** It provides a set of core APIs, allowing solutions to be built that enable different consumption models for different groups of people across the enterprise (such as a “data marketplace”). This might be useful for business analysts to power their BI reporting.
- **Self-service enabled:** It provides a core set of services to enable consumers (developers, analysts, data scientists, creative teams, mobile developers, ops and admins) to access the network in ways that makes sense for them through tools they understand. It also has a set of APIs that allow new consumer models to be built.

The key design principles of an application network

When considering the genesis of an application network, it's important to define the design principles that shape how its components are built:

- **The network is opinionated:** The network has built-in, well-defined and systematic mechanisms for attaching new nodes and enabling them within the network. These include publishing APIs and their specifications, registering client access grants and hence also client dependencies, bringing them within a security context, etc. It also provides built-in, opinionated ways of operating the nodes and tending to their life cycle, of collaboration between users on the assets and their lifecycle, and of analyzing and securing the network. The network is not intrinsically opinionated about the specific architecture of which applications must be connected in what ways, as that is often determined in an incremental, merit-based way, vs a top-down traditional SOA approach. However, architects may suggest or impose certain architectures, API patterns, standardized data schemas, etc. to enable and potentially constrain how the network is built.

- **The network is consistent:** Rules and policies are enforced within a domain. Artifacts such as APIs, templates and portals inherit the services and restrictions of the network. This means that APIs are subject to the policy enforcement of the network; templates use common error handling defined by the network.

- **Applications are connected to the network:** Conversely, applications are mostly not connected directly with each other. The connection into the network contains rich

metadata about each application which the network uses to expose that application inside the network.

- **The network adheres to Metcalfe's law:** Adding a new node to the network must add n+1 value to the network. That value must be a set of KPIs that are built in and can be tracked.
- **All data connected to the network is discoverable and addressable:** Every resource is exposed through through system APIs. Not all data will be exposed to consumers — this is an administration choice.
- **The consumer is king on the network:** The fundamental value of the network is to unlock and enable consumption of data and resources. Every person in an enterprise can connect and get value from the network. This means the network must cater to different consumer profiles across technical and non-technical roles.

How is an application network different from SOA?

The principles of SOA still are valid today, but the needs of the organization have changed drastically, requiring an equally drastic shift in how IT works with the rest of the business. SOA had to be built from the ground up for every company; it therefore became a reflection of the complexity and custom applications that defined the enterprise's IT landscape. In contrast, the application network is designed to enable the whole business, not just IT. This is key, as innovation cannot occur at the necessary speed if IT is the sole technology provider to the business. The needs of the business will grow faster than the capability of IT to meet them. Finally, SOA is an architectural concept, while an application network is a tangible product that connects not just applications, but people and processes as well.

SOA is an architectural concept, while an application network is a tangible product that connects not just applications, but people and processes as well.

It's important to note that services must be loosely coupled to provide the flexibility and scalability that businesses need. Tightly coupled networks become brittle and expensive to maintain, which defeats the self-service ethos of the application network. Simply completing a lot of integration projects in a traditional way does not result in an application network nor does it provide an application network's benefits. Instead of connecting applications through point to point connections or isolated architectures, application networks provide an

infrastructure that can achieve reuse of services, knowledge, access and best practices.

This means that IT has to shift its mindset from complete control to a distributed authority over technology in the business. It requires a federated approach to governing and controlling data, applications, and systems. This allows IT to be a platform enabler for the organization, will increase productivity through reuse, and will make change more predictable and easier to manage.

Conclusion

With the massive number of applications, data, and devices that need connecting in the modern enterprise, and the incredible amount of time and resources that companies spend trying to tie everything together, an application network can provide the agility, flexibility, and speed that businesses in today's environment urgently need. New applications can be plugged into the application network as easily as you can plug in a printer. The application network can deliver unified vision and control and offer intelligent data about the relationships between different applications. A new vision of what your IT organization can be and do can allow your business to harness the digital revolution; an application network is designed to make that happen.

About MuleSoft

MuleSoft's mission is to help organizations change and innovate faster by making it easy to connect the world's applications, data and devices. With its API-led approach to connectivity, MuleSoft's market-leading Anypoint Platform™ is enabling over 1,400 organizations in approximately 60 countries to build application networks.

For more information, visit **mulesoft.com**

MuleSoft is a registered trademark of MuleSoft, Inc., a Salesforce company.
All other marks are those of respective owners.