White Paper

Three Myths of Modern Integration

Why Integration Should Be Low-Code, Cloud-Native, and Full Lifecycle

Jason Bloomberg

President, Intellyx

September 2021

The world of enterprise IT has entirely transformed from the way it was a mere decade ago. Today, we expect and require the ability to change at scale.

IT infrastructure must be inherently dynamic without any limitations on the scope or scale of any particular application. Cloud-native computing has arisen to meet these needs.

Enterprises, however, must continually deal with existing, legacy assets in the context of this modern IT landscape. Cloud-native computing raises the bar on enterprise integration, as organizations can no longer afford to slow down when connecting old assets to new.

To meet this need, a new class of integration technology has emerged that is low-code, cloud-native, and supports change across the full software lifecycle.

The Modern IT Landscape: Change at Scale

Commenting on the ubiquity of change, it was the ancient Greek philosopher Heraclitus who said that it was impossible to step into the same river twice. The fellow may have been prescient, considering the state of enterprise IT today.

Not only is change constant in the world of technology, but it affects all aspects of IT, from the expectations of customers to the underlying infrastructure. In many situations, change is even accelerating.

The ability to support change, however, is not sufficient. IT organizations must be able to support *change at scale*.

This pervasive requirement has given rise to *cloud-native computing*: a paradigm shift in how organizations build and leverage technology to support this kind of change.

The ability to support change, however, is not sufficient. IT organizations must be able to support *change at scale*. This pervasive requirement has given rise to cloud-native computing.

Cloud-native computing, however, can be complicated and difficult – and wouldn't be worth the effort if customers weren't demanding better, more dynamic technology-centric capabilities from the companies they do business with.

The Digital Transformation Context for Cloud-Native

Putting customers at the center of the IT effort is at the core of digital transformation. Such transformation is both *customer-driven* and *technology-empowered* – and is becoming increasingly urgent now that the Covid pandemic is nearing its end.

Impeding this mandate to leverage cloud-native computing to deliver on digital transformation is the full spectrum of technical debt, primarily within IT. Decades of expedient decisions and generations of now-legacy technologies present a formidable barrier to the change at scale that today's enterprises crave.

Ripping out old tech is rarely if ever an answer. To move forward in the face of technical debt, IT leaders must bring to bear a deliberate combination of modernization, migration, and integration – updating, moving, and connecting existing assets to new ones, respectively.

Integration, in particular, requires a new way of thinking in order to support the cloud-native 'change at scale' mantra. We must bust the old myths of integration in order to move forward with cloud-native computing, and by extension, our digital transformation priorities.

Integration requires a new way of thinking in order to support the cloud-native 'change at scale' mantra. We must bust the old myths of integration in order to move forward with cloud-native computing, and by extension, our digital transformation priorities.

Integration Myth #1: Low-code is only suitable for simple integration scenarios

Low-code technologies have been around for decades, but the modern generation of low-code tools and platforms focuses on reducing the burden on professional developers as it fosters greater collaboration with stakeholders while accelerating software development.

Low-code, however, is for more than building standalone applications. Within organizations, creating software never stands alone. There is always a database to query, an API to connect to, or a connector to integrate with.

Some low-code tools struggle to address this reality. Many low-code tools can help application creators tackle many of these basic integration scenarios, but leave the more complex ones to hand-coding.

This limitation, however, is quickly becoming out of date, as modern low-code tools help application creators with increasingly sophisticated integration scenarios without requiring the writing of a single line of code.

Integration Myth #2: Cloud-native computing is about greenfield applications, but integration addresses challenges of brownfield

The need for change at scale has given rise to microservices and containers, which in turn brought the container orchestrator Kubernetes to the forefront of the cloud-native movement.

Many early adopters of Kubernetes leverage its ability to support change at scale to build new microservices that assemble into greenfield applications.

Cloud-native computing, however, is broader than Kubernetes and supports far more than greenfield apps. In fact, the typical enterprise scenario is a hybrid mixture of existing, 'brownfield' assets as well as the newer capabilities that we're calling 'greenfield.'

The bigger picture of cloud-native computing is the presence of a control plane abstraction that supports and enforces policy-based behavior across even the most complex of hybrid IT environments.

Greenfield is a component of this bigger picture to be sure, but connecting new to old – green to brown – is the more common and indeed more valuable scenario.

Traditional, hard-coded integration, however, is too inflexible to support this hybrid IT vision for cloud-native computing. Instead, integration must itself be policy-based and inherently flexible to support the cloud-native applications of today and into the future.

Traditional, hard-coded integration is too inflexible to support this hybrid IT vision for cloud-native computing. Instead, integration must itself be policy-based and inherently flexible to support the cloud-native applications of today and into the future.

Integration Myth #3: Integration is a pre-deployment activity

In order to achieve the change at scale benefits of cloud-native computing, we must free ourselves from the shackles of traditional, hard-coded integrations.

Even as integration technologies have evolved, from the enterprise application integration tools of the 1990s to the enterprise service buses (ESBs) of the 2000s to the RESTful approaches of the 2010s, IT professionals have always treated integration as something they do before deployment. In other words, hook things up before you run them.

The goal of being able to change integrations dynamically at runtime was a dream of the ESBs and Web Services days. Early implementations of Web Services supported automatic discovery and integration, where endpoints could negotiate with each other automatically at runtime.

In practice, however, this type of dynamic integration proved impractical – and Web Services became yet another endpoint for hard-coding integrations.

Even today, they use cases for fully automated integrations that can change dynamically at runtime are limited. The sheer variety of different kinds of changes are simply too much for automated integrations to handle.

To balance these priorities of changing integrations at scale, we must consider integration to be a *full-lifecycle activity*.

In other words, we must break free from the waterfall-centric belief that the pre-deployment and post-deployment phases for software are somehow immutable. Instead, we must take a page out of the Agile playbook, and approach the software lifecycle in an iterative manner.

> We must consider integration to be a full-lifecycle activity. In other words, we must break free from the waterfall-centric belief that the pre-deployment and post-deployment phases for software are somehow immutable.

The cloud-native way of thinking pushes this iterative approach to its extreme with modern development practices like DevOps, CI/CD, and GitOps.

We're expecting constant change without slowing down the deployment of new and updated software, the theory goes, so we allow and plan for changes in the production environment.

This modern, dynamic approach to building software that supports change at scale applies to integration as well. We must be willing and able to build and change integrations post-deployment in an iterative, Agile manner.

The policy-based strictures of CI/CD and GitOps certainly help with this new way of integrating software. Given the wide variety of integration scenarios and the diversity of changes that might impact them, however, a low-code approach to integration becomes absolutely essential.

Remember, low-code empowers development teams and stakeholders to be more collaborative as well as iterative – not just while building applications, but while running and updating them as well.

Integrations won't always be perfect in the face of changing requirements and dynamic applications, but with low-code, the people responsible for those integrations have the power and control to keep them up to date across the software lifecycle.

The Intellyx Take: Solving Modern Integration Challenges

There are many integration products on the market, and it seems that analysts keep coming up with new names for the various categories they ostensibly fall into.

We have Integration Platform-as-a-Service (iPaaS), *Enterprise* Integration Platform-as-a-Service (eiPaaS), and Hybrid Integration Platform (HIP). We also have overlapping product categories the big analyst firms have a harder time tracking, including low-code integration tools and cloud-native integration tools.

Don't let the analysts and their market terminology confuse you. Instead, look for the core capabilities you require to support integration change at scale.

Integration platforms like Digibee, first and foremost, are cloud-native. Not only do they leverage microservices, containers, and Kubernetes, but they also provide a policy-based control plane that enables IT personnel to manage integrations across the software lifecycle.

Digibee's platform is also low-code, providing straightforward visual interfaces that simplify and accelerate the work of professional developers while fostering greater collaboration and support for change across the lifecycle.

In fact, Digibee's support for the full integration lifecycle is another must-have capability of any modern integration tool. No longer can we relegate integration activities to the pre-deployment phase of a software effort.

Cloud-native computing delivers change at scale across the software lifecycle, and integration technology must also rise to this challenge.

About the Author: Jason Bloomberg

Jason Bloomberg is a leading IT industry analyst, author, keynote speaker, and globally recognized expert on multiple disruptive trends in enterprise technology and digital transformation.

He is founder and president of Digital Transformation analyst firm Intellyx. He is ranked #5 on Thinkers360's Top 50 Global Thought Leaders and Influencers on Cloud Computing for 2020, among the top low-code analysts on the Influencer50 Low-Code50 Study for 2019, #5 on Onalytica's list of top Digital Transformation influencers for 2018, and #15 on Jax's list of top DevOps influencers for 2017.

Mr. Bloomberg is the author or coauthor of five books, including *Low-Code for Dummies*, published in October 2019.

About Digibee

Digibee developed their hybrid integration platform to quickly connect enterprises to the digital world leveraging Kubernetes, micro-services, low-code, and a unique delivery model. Their innovative and flexible platform solves the integration challenge faster and more efficiently than any other integration approach in the market. The innovative approach allows clients to unlock innovation by simplifying integration and treating integration as an abstracted capability versus a one-off undertaking.

Their holistic and agile approach to integration, allows enterprises to accelerate decision making, enhance customer experience and achieve greater business outcomes in a fast and scalable way without a major investment. Digibee has offices in São Paulo, Brazil; Weston, Florida and Denver, Colorado.