

SaaS, open source, and serverless: A winning combination to build and scale new businesses

Three tech approaches can rapidly accelerate business building for established companies that learn how to use them.

by Stéphane Bout, Philipp Hillenbrand, and Henning Soller



Speed and scalability are critical when building new businesses, with the accelerating pace of change creating a winner-take-all dynamic in many markets. Most digital-first start-ups have rapidly gained a lasting competitive advantage by incorporating new technologies that enable them to innovate and scale quickly.

Specifically, a trifecta of technology approaches has emerged, providing a formidable arsenal for companies looking to launch new businesses more quickly, securely, and effectively at lower costs:

- *Software as a service (SaaS)*, which allows companies to consume all software services they need without having to create the software themselves
- *Serverless architecture*, which enables companies to focus on *writing* code rather than *running* it
- *Open-source code*, which gives businesses access to existing, free-to-use software libraries that can easily be integrated into a company's own code

Each of these can offer significant benefits. But the real breakthrough is in combining all three, flexibly combining services (such as a login) from a third-party SaaS provider and integrating them with other third-party services (such as recruiting management services) and company-specific functions (such as salary calculations) in a fully serverless architecture. This tech fusion enables companies to radically simplify and accelerate the development process for both launching and scaling new businesses.

For companies looking to launch new businesses, the core principle of combining these approaches is to both maximize the value of what already exists and reduce the burdens of development and maintenance. Other benefits are also emerging. As companies develop greater expertise at using serverless and build up their architecture

capabilities, they are able to take greater advantage of the cyberservices offered by cloud service providers (CSPs)—although they will always need to take practical steps to manage their individual risk.

Companies seeking to build new businesses, however, too often misunderstand the full extent of the benefits of SaaS, serverless, and open source, or they implement them ineffectively. In many cases, that's because they believe their own legacy systems and approaches offer a more reliable foundation for launching a new business. Unfortunately, these companies soon learn that this approach not only brings with it the inefficiencies tied to legacy systems but also limits their ability to think ambitiously and creatively enough to architect the applications needed to completely enable the new business.

To make SaaS, serverless, and open source work for new businesses, incumbent companies must completely rethink their IT strategy, redesign processes, embed new approaches and mindsets, redesign security, and attract a new type of talent. The benefits of serverless in reduced time to market alone, for example, more than outweigh the challenges of making the transition. Leaders who can manage these five tasks in tandem will gain the organizational capabilities to better support business building and jump-start growth.

How SaaS, serverless, and open source accelerate business development

The benefits of serverless and SaaS specifically are the result of competition among major CSPs to offer more—and more-compatible—applications of the best tech platforms “as a service.” This trend underscores the value of cloud as an enabler and driver of innovation, rather than just as a way to optimize IT costs. Recent McKinsey research, in fact, has shown that as much as 75 percent of the more than \$1 trillion of value at stake in cloud comes from business innovation rather than from managing IT costs.¹

¹Will Forrest, Mark Gu, James Kaplan, Michael Liebow, Raghav Sharma, Kate Smaje, and Steve Van Kuiken, “Cloud's trillion-dollar prize is up for grabs,” February 2021, McKinsey.com.

A closer look at serverless: Interview with Fabrice Bernhard

What is serverless architecture, and what role can it play for incumbent companies looking to build new businesses?

Serverless is the future of cloud and is the result of major cloud service providers trying to compete by offering more and more of the building blocks of the best tech platforms “as a service.” Leveraging these building blocks allows companies to offload more of the heavy lifting to the cloud provider, thereby making it faster, cheaper, easier, and safer for teams to create scalable applications. Serverless also makes architectures more event driven, which means easier real-time integration with other services, whether from internal software or external service providers.

With the right talent and serverless platforms, any business can now build what only the best-funded start-ups could afford to build just a few years ago. We therefore see serverless as a game-changer for the next decade, similar to what cloud has been in the past one—in effect, a second cloud revolution. It’s not about updating the current cloud infrastructures; it is as transformative as the cloud was ten years ago, requiring new skills, new roles, and—at scale—changes to the whole IT organization.

What are typical failure modes in migrating to “serverless,” and how can you avoid them?

Education is identified as the biggest challenge in O’Reilly’s 2019 serverless survey.¹ The technology is still relatively new, and the mindset shift is probably also underestimated by technical teams at the beginning. The key here is to ensure an autonomous capability is built through the delivery of an initial greenfield project.

Once a successful greenfield project has been completed, organizations can start scaling serverless and looking into migrating legacy systems. Here, one failure mode is inconsistent approaches and duplicating experimentation between different silos. The best success is seen when organizations invest in a platform team to develop the serverless best practices and provide them via in-house building blocks “as a service” to the rest of the organization. One of the benefits of serverless over previous cloud technologies is the granularity of the encoding of best practices. With the right tooling and approach, many common application components can be standardized, audited, and reused with ease, further improving the productivity of the tech teams.

Real-world examples consistently confirm that serverless is cheaper. However, trying to “lift and shift” large systems to serverless is absurd. To reap the benefits of serverless, you need to transform your application to a service-oriented and event-driven architecture.

Fabrice Bernhard is the cofounder and Group CTO of Theodo, an agile software-development specialty consulting firm that works with clients to build disruptive tech platforms.

¹ Roger Magoulas and Chris Guzikowski, “O’Reilly serverless survey 2019: Concerns, what works, and what to expect,” O’Reilly, November 12, 2019, oreilly.com.

For companies looking to get a new business off the ground, SaaS is a way to quickly and cheaply access a robust array of existing services. When using just a single service or combining a number of them, SaaS has proven capable of meeting as much as 90 percent of the needs of a given function. The business essentially just consumes the software it needs, often paying through a predictable subscription model, saving it from having to write new code—and if additional code is still needed, there are large libraries of standardized, open-source code that developers can access quickly.

Serverless is unmatched in rapidly deploying new tech stacks and digital technologies in modular formats that are completely adaptable, even at the code level. Traditionally, IT teams deploying new software had to specify, order, install, and maintain







the hardware, operating systems, and security of the servers. Even with the emergence of virtual machines and containers, IT teams still needed to patch for security, configure dependencies, scale the provisioned capacity, and make sure that the container was always running.² With serverless, the provider handles all of these tasks and removes the operations overhead (exhibit). Serverless allows IT teams to offload infrastructure management entirely to the provider, therefore making it faster, cheaper, easier, and safer for teams to create scalable applications.

Serverless also promotes more event-driven architectural patterns—that is, rather than running persistent workloads in the cloud, event-driven architectures in combination with serverless enable organizations to pay only when a function

² Recently, CSPs have introduced serverless container services to address some of these issues.

Exhibit

Serverless reduces overhead and speeds new-business development.

	Containerized	Serverless
 Persistence	Always on and transient	Transient <i>only</i> with maximum allowed execution time
 Cost structure	Pay as you go or provisioned capacity	Pay as you consume
 Deployment packaging	Must be packaged as part of a container image	Minimal packing—source code is directly run by serverless execution environment
 State management of applications	Can be either stateful or stateless	Stateless by nature
 Scalability	On demand, but with configuration and compatible coding Application must be stateless and container platform configured to support autoscaling	Can support hyperscaling subject to design to manage max concurrency, quotas, and dependent limits on gateways
 Load balancing	Typically needs to be configured manually	Deployed based on access gateways

is actively being used and consuming compute resources. While this approach needs to be carefully considered in low-level architectural decisions (for example, synchronous versus asynchronous functions) to control costs, it enables easier real-time integration with other offerings, whether from internal software or external service providers.

There are additional benefits of SaaS, serverless, and open source that are particularly useful when building and launching new businesses, among them:

- *Reduced initial investment outlay.* Moving existing software to the cloud requires investments, especially in automation and often to a lesser extent in development and licensing. As companies “move up the stack” (in other words, consume more generalized services), they can increasingly take advantage of CSP offerings. For companies looking to set up quickly, build functionality efficiently, and test the market cheaply, serverless is an attractive option.
- *Elimination of infrastructure management.* Serverless requires zero infrastructure management (and hence no operational overhead) and significantly eases support for version upgrades. This frees product teams from hard dependencies on infrastructure provisioning, allowing businesses to shift resources to building products and services that directly generate value. Developers can take advantage of the fact that serverless architecture simplifies access to software functions as a service, reducing effort wasted on creating solutions that already exist. These functions can be flexibly recombined. With the right tooling and approach, many common application components can be standardized, audited, and easily reused, massively improving tech teams’ productivity. That can allow a company to quickly and easily test new-business functions with customers and adapt them as needed.
- *Code brought “closer to the business.”* Using SaaS and serverless to free IT from

infrastructure management greatly reduces the complexity of app development and deployment. This, in turn, allows tech teams to organize around products—for example, “cards” or “loans”—which brings code “closer to the business.” As part of a commitment to agile and product-driven engineering, this approach makes it much easier for business leaders, product owners, and analysts to understand solutions and work with engineers in cross-functional teams.

For all the benefits of SaaS, serverless, and open source, it’s important for companies to recognize and address their challenges. Initial obstacles might include the need for training; a lack of automation tooling, granular observability, and self-healing applications (for example, tools to monitor application health in a decoupled environment); and the need to build stateless software. Lastly, vendor lock-in over time may be a concern, if native CSP serverless development frameworks are used. In our experience, all of these challenges are manageable and greatly outweighed by the upside of SaaS and serverless.

Real-world applications

Two recent examples demonstrate the impact of these tech approaches on application development.

An Asian oil and gas company spun out a series of products that have now become stand-alone businesses. By choosing to use the same serverless architecture across all of them, the company needed only 12 weeks and a small central team to stand up the new tech infrastructure, which was already supporting its core applications. Now the company can run a range of tasks, from complex visual-data upload and compression to drone steering, on its serverless architecture. As these businesses continue to grow, needed processing performance enhancements can be made rapidly.

A leading private-equity firm used serverless to develop an entirely new and highly configurable investor information and administration site. An application-programming-interface (API) gateway connects the new serverless architecture to the

company's legacy systems and customer data. The site can be flexibly adapted and enhanced, giving the firm the ability to easily boost the site's performance. By using serverless, the firm was able to design, build, and launch this new digital customer tool within a few months, compared with more than a year using the traditional approach.

Five keys to unlocking the value in serverless, SaaS, and open source

While the advantages of serverless are significant, so too are the underlying changes required to harness it. The greatest success comes when companies focus on five areas.

Adopt 'modular' mindsets

Serverless is still relatively new, and technical teams can underestimate the mindset shifts it requires. Indeed, IT leaders often erroneously think serverless is "just a few more cloud features." In reality, serverless involves not just building differently but also using technology assets differently—from bringing a new, more modular approach that functions in a stateless architecture design (a microservice deployed in a container will translate to ten to 30 functions) to reconfiguring how database services are accessed.

Reimagine IT architecture

To get the full value from serverless, companies need to completely rethink how they approach IT architecture in support of business building. It's crucial to invest the time to make the best low-level architecture decisions when going serverless, such as asynchronous versus synchronous functions. Poor architecture decisions can be extremely costly. For example, the oil and gas company mentioned previously was able to set up the basic technology infrastructure in 12 weeks. It then added services to support tasks—for example, acquiring visual data from autonomously moving cameras, uploading the data to storage, and extracting relevant insights layers—by deploying several readily available services and a few proprietary modifications.

Building functionality as needed will allow companies to reengineer processes at a completely

different speed than is currently possible, where IT would need to plan in changes to customize the IT architecture to specific tasks and business processes, a time-consuming and resource-intensive task in most cases.

The change toward more industrialized and standardized software development will have a massive organizational impact. The complete adoption of SaaS and serverless architecture could mark the end of high-cost compute for commercial off-the-shelf applications, eliminating operational overhead associated with infrastructure management while enabling a massive increase in both time to market and organizational agility as well as greatly improved integration and security. Grasping this transformational change at the outset is necessary for IT and business leaders to successfully undertake the scope of changes needed.

Invest in tooling and tech product teams

Teams that have not restructured to support cloud-native operating models must be reconfigured to focus more on architectural interplay, functionality development, and new capabilities for self-healing and on-demand scaling. The most successful organizations invest in tooling and in tech product teams to harness SaaS and serverless application components and integrate additional functions as required. The decision on whether to upskill existing in-house development teams or bring in new talent depends on a company's starting point. Once IT functions embrace the new architectural-design paradigm and complete successful pilots, they will need added capacity to roll out serverless across the organization.

Build out explicit integration plans and processes

A serverless approach requires teams to have a clear view of how to integrate and manage the interplay of existing assets and providers. An up-front integration plan needs to be developed to make sure the tech requirements that enable existing assets to work well with serverless are met. This calls for the automated provision of additional provider services to ensure that endpoints, events, and fine-grained access controls are consistently provisioned.

Design for security

Accidental risk of exposure is high for serverless if companies do not adopt mature secure-by-design cloud principles on day one. There are four categories of serverless considerations:

- securing serverless in public cloud, perhaps by isolating serverless workloads in public cloud with granular account-level segmentation, and limiting exposure through the use of blast-radius architecture
- rethinking authentication for transient serverless workloads by using ephemeral credentials and short-lived tokens, which are key risk mitigators for credential exposure
- protecting your availability in a serverless landscape with robust perimeter security that deploys public and internal functions at discrete gateways

- upgrading risk assessment, governance, and awareness by, for example, adopting policy as code for the codification of organizational policies; using regulatory frameworks in automated governance pipelines for cloud-service provisioning; and deploying all serverless workloads using an embedded DevSecOps pipeline

SaaS, serverless, and open source are emerging industry standards. But only by understanding how to use them together—and align tech operations more closely with the business—can companies take full advantage of these tools to launch better businesses faster.

Stéphane Bout is a senior partner in McKinsey's Paris office, **Philipp Hillenbrand** is a partner in the Berlin office, and **Henning Soller** is a partner in the Frankfurt office.

The authors wish to thank Fabrice Bernhard, cofounder and CTO of Theodo, for sharing his experience and perspective. The authors also wish to thank Aaron Bawcom, Jayne Giemzo, and Lars Vinter for their contributions to this article.

Copyright © 2021 McKinsey & Company. All rights reserved.