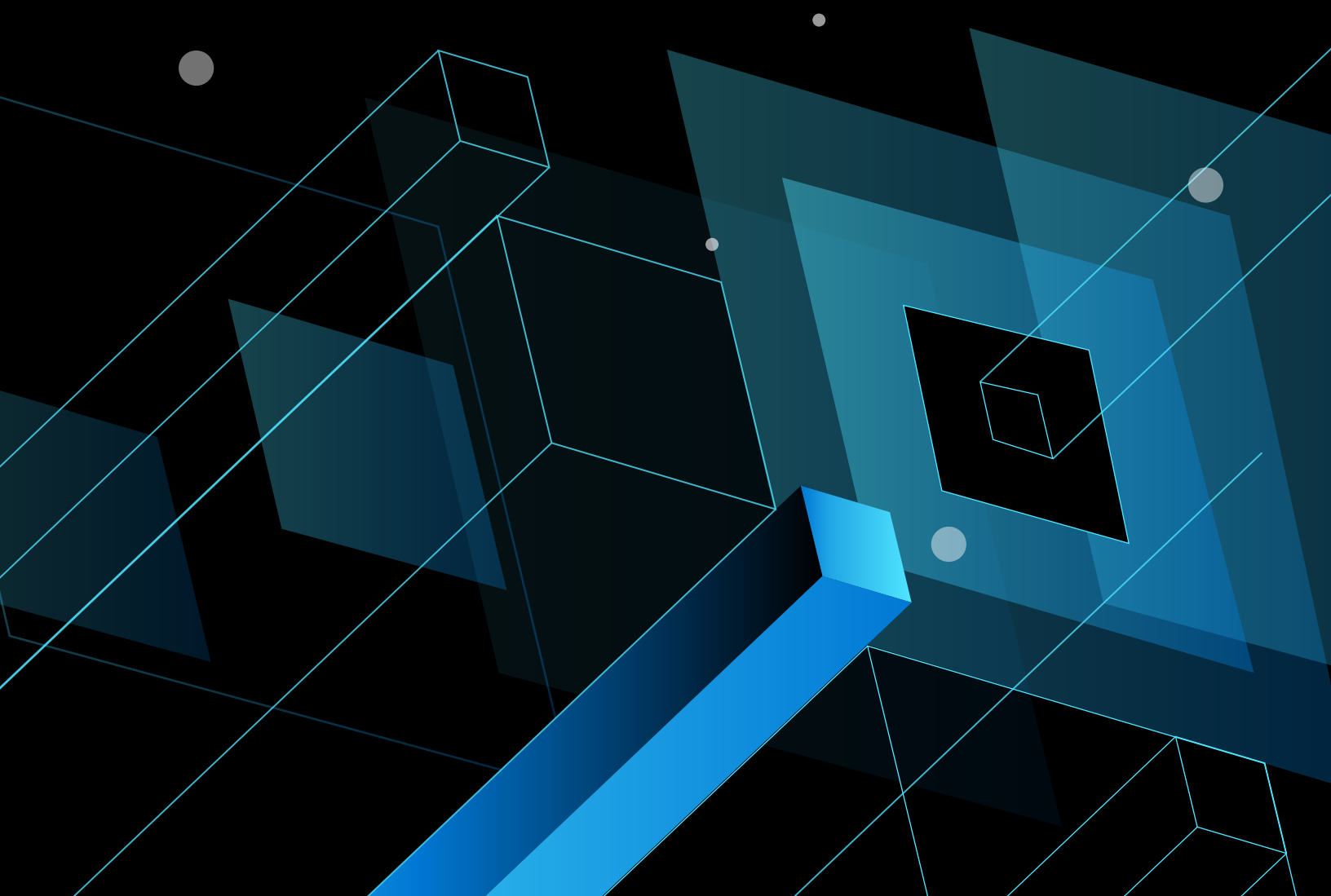


# Developer Velocity

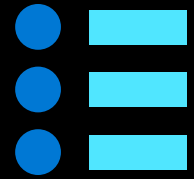
Lessons from Digital Leaders on  
Accelerating Business Performance  
through Software Excellence



# Contents

- Executive summary ..... 3
- About the research ..... 7
- Deep dive on learnings in banking ..... 10
- Deep dive on learnings in retail ..... 21
- Learnings that apply across companies ..... 30
- Implications for technology leaders ..... 36
- How to increase Developer Velocity in your organization ..... 37
- Building a more resilient future together ..... 38
- About the Authors ..... 40
- Resources ..... 41

# Executive summary



**Companies across industries are pursuing digital strategies and transforming their business through software development. Doing this effectively requires companies to achieve superior [Developer Velocity Index \(DVI\)](#) scores.**

This enables companies to tap into the full range of skills and creativity of developer teams by empowering them, creating the right environment to innovate, removing points of friction, and turning ideas into software that supports customer needs and business objectives.

In early 2020, we conducted in-depth research with more than 400 companies to both quantify the impact of Developer Velocity on business performance and identify which drivers matter more than others. Following our initial research, significant durable shifts have occurred driven by the COVID-19 pandemic. These

shifts have accelerated digital transformation timelines, leading to a greater urgency for companies to invest in software development to adapt and thrive.

To understand how companies build high-performing development organizations while facing the challenges of COVID-19 and remote development, we recently completed a second phase of research on Developer Velocity. This research focused on digital leaders in retail and banking—two industries that have seen high impact and acceleration of software investments through 2020.<sup>1,2</sup> For example, in retail, e-commerce spending was up 30 percent between March and April,<sup>3</sup> and many customers expect to continue using online channels post-pandemic. Additionally, both industries have unique challenges when building software. For example, banks need to build systems that can handle complex regulatory and compliance requirements and retailers need to build systems that can handle millions of transactions daily across multiple channels.

<sup>1</sup>[Retail Reimagined report \(McKinsey, 2020\)](#)

<sup>2</sup>[Transforming the US consumer bank for the next normal \(McKinsey, 2020\)](#)

<sup>3</sup>[The reinvention of retail report \(McKinsey, 2020\)](#)

Specifically in this phase, we studied five leading companies in depth:<sup>4</sup> ABN AMRO, Capital One, Albertsons, a leading global Quick Service Restaurant (QSR) chain, and a large global US-based bank. In addition, we interviewed a few additional leading retailers with global footprints. Each of these companies have made it a core focus to build world-class software and build a culture that attracts technical talent. In this whitepaper, we detail the steps they have taken and the Developer Velocity drivers they have deployed. While these insights contain several learnings for other retail and banking leaders, many learnings also apply across industries.

We observed common patterns that predict success in the three banks we profiled. Included among them is a strong push on software and cloud that functions as a joint partnership across business and technology. Each company also took a phased and multi-year approach to building software capabilities. This starts with the foundations of adopting Agile, rearchitecting the code base to be loosely coupled, up-leveling talent, and migrating to public or hybrid cloud. While doing this, each of the banks took a thoughtful approach to security and risk—automating and embedding controls within the pipeline versus driving manual processes.

In retail, we found Albertsons and a global Quick Service Restaurant (QSR) faced a different set of challenges and have taken slightly different approaches to building software capabilities. However, some common themes emerged including: organizing around autonomous teams that can own a product end-to-end (checkout, POS, etc.), and an aggressive push to cloud for business applications—often prioritizing customer-facing workloads (e.g., the e-commerce stack) and the core backend systems (e.g., merchandising). For both retailers, the impact has been high, enabling them to increase digital revenue growth. For example, e-commerce was a key growth driver for achieving its third straight quarter of more than 200 percent sales growth, up 225 percent in Q3 of FY 2020.

<sup>4</sup>We included 3 banks in this research, but in this document we will only discuss two banking case studies in-depth.

As mentioned above, there are themes that can be universally applied across industries.

### In technology:



**Successful cloud adoption requires a joint partnership across business and technology** with a focus on accelerating business outcomes. In successful companies, business and technology often align on specific timelines and outcomes that enable and empower development teams to identify the optimum cloud adoption path.



**For tools, companies often look to standardize their software development toolchain** in key areas, typically: CI/CD pipeline, planning, and code management tools, leaving flexibility in other areas to empower individual teams and developers.




**Data architecture and data platform modernization are the next evolution of application platform modernization** efforts for even the most successful companies. Data architecture is inseparable from companies' efforts to capture application modernization benefits such as enhanced personalization, improved fraud detection, and real-time alerts.




**Security is a major area of ongoing investments**, including shift-left, automation, and embedding controls in the pipeline itself. Companies see cloud as an enabler to improve security.


## In organizational enablement:


 Unsurprisingly, **world-class talent differentiates the top performers from the rest.** Companies across industries compete with leading tech companies for the best talent through culture and emphasizing their unique value proposition (i.e., the ability to work on products that “impact the lives of millions of people every day”).


 **Product Management (PM) and adoption of a product and platform-centric operating model is the next frontier for many organizations.** Common challenges include; balancing PM skills across business domain knowledge, experience design and technical skills, lack of a well-defined operating model across “business” and “technology,” and lack of a strong Technical Product Management (TPM) function.

 Finally, **cloud adoption and modern tools differentiate companies that have parity between in-person and remote developer productivity.**

## In working practices:

 **Autonomous scope and loosely coupled architecture are must-haves for Agile companies to succeed.**

 **Open source software is a differentiator between the top performers versus the rest, particularly in attracting and retaining top talent and increasing speed of innovation.**

 **Inner source adoption is early across the board, but a top priority for companies that are looking to achieve top-quartile Developer Velocity.**

This research has significant implications for technology leaders. In the early stages of transformation, leaders should look to invest in foundations such as adoption of Agile practices, integration of business and technology, cloud adoption, and re-organization into small autonomous teams. It is critical for leaders to guide the organization through this early phase, constantly adopting the latest cloud technologies, rearchitecting into microservices, driving organizational agility, adopting open source, and continuously integrating and deploying (CI/CD) processes. The most successful organizations are taking this continuous improvement journey further by adopting DevSecOps and building true product management and design capabilities.



# About the research

**We define Developer Velocity as the ability to drive transformative business performance through software development. This is achieved by empowering developers, creating the right environment for them to innovate, and removing points of friction.**

In phase one of our research (early 2020), we surveyed 440 large enterprises and quantified the impact of Developer Velocity on business performance, identifying which drivers have more impact than others. As a result, we created the Developer Velocity Index (DVI), which pinpoints the most critical factors in achieving Developer Velocity, as well as some factors that are not as important as many executives and observers might believe. One of the key findings from the research was that across industries, companies with top quartile Developer Velocity are growing revenue four to five times faster than their counterparts. Top-quartile companies also have 60 percent higher total shareholder returns and 20 percent higher operating margins. However, financial performance differences between

the second, third, and fourth quartiles is marginal. In addition, top-quartile players appear to be more innovative, scoring 55 percent higher on innovation than bottom-quartile companies. Similar patterns hold within specific industries and sectors. For example, in financial services and retail, top-quartile companies saw positive revenue growth while average revenue declined in the other quartiles. Additionally, of the more than 40 drivers tested, four universally show up as the most important for business performance: product management capabilities, culture, developer tools, and talent management.

However, depending on where you are in the journey or your industry, what matters can vary. For example, for companies in the bottom quartile, cloud adoption and Agile team practices matter significantly. For companies in the top quartile, open source adoption is the biggest driver of business performance. In retail and banking, cloud adoption is a major driver of performance. In banking specifically, security and compliance practices are a major driver of Developer Velocity.

In late 2020, we launched phase two of the developer research focused on a select set of digital leaders to better understand how they have accelerated Developer Velocity. This work investigates what has and has not worked well, and the impact on business performance. These companies have made significant investments in software over the past several years. Within each of these companies we interviewed senior technology leaders, product managers, software architects, and developers across a range of product areas. This focus allows us to present insights for organizations that have committed to technology transformation with specific details best practices.

**We focused our investigation in three areas:**

### 1. Overall software journey

How the company has invested in software development capabilities and top business priorities.

### 2. Developer Velocity drivers deployed

Which capabilities have been most impactful in driving performance of software development teams to achieve business outcomes. In these areas, how did they drive adoption and scale changes. We asked what key learnings other software leaders could take away from their experience, and what two to three areas they are looking to focus on next.

### 3. Impact of investments

The quantified impact of investments the companies made in Developer Velocity drivers on business performance.

While each company is unique, the research is meant both to distill the lessons from these companies' experiences and to identify patterns that lead to success. Additionally, in this second phase of research we also added a set of new drivers focused on remote work to address changes with COVID-19 and application modernization.

# Developer Velocity involves 57 drivers across 14 dimensions

Technology



## Architecture

- Software architecture
- Data architecture



## Testing

- Test automation
- Test-driven development



## Infrastructure and platform

- Public cloud adoption (IaaS, PaaS)
- Infrastructure as code
- Application modernization



## Tools

- Planning tools
- Collaboration tools
- Development tools
- DevOps tools
- Low- or no-code tools
- AI assistance in development

Working practices



## Engineering practices

- Technology debt management practices
- Coding guidelines
- Code reviews
- CI/CD practices



## Security and compliance

- Security practices
- Compliance practices



## Open source and inner source

- Open source usage and contribution
- Inner source adoption



## Agile team practices

- Work-in-progress management
- Agile ceremonies
- Definition of done

Organizational enablement



## Team characteristics

- Autonomous scope
- Limited context switching
- Cross-functional teams
- Colocation of teams



## Culture

- Psychological safety
- Collaboration and knowledge sharing
- Continuous improvement culture
- Servant leadership
- Culture of customer obsession



## Talent management

- Incentives
- Capability building
- Recruiting
- Team health management
- Employee value proposition
- Engineering career paths



## Product management

- Product management capabilities
- Product telemetry
- Product vision
- Linkage between strategy and team metrics
- Rapid prototyping



## Organizational agility

- Dependency management
- Funding mechanisms
- Portfolio management



## Remote work

- Remote work model
- Mental health
- Manager coaching
- Asynchronous communication
- Video based communication
- Remote collaboration tools
- Single source of truth
- Core working hours
- Security
- Remote working toolchain

# Deep dive on learnings in banking



**Financial services technology is currently in the midst of a profound transformation. Customer preferences are shifting to include digital and omnichannel experiences.**

Additionally, the importance of data to drive initiatives such as new products and underwriting capabilities, personalization, fraud detection, and automation of core banking processes is higher than ever. In a competitive environment of rising cost pressures, where rapid action and response is imperative, financial institutions must modernize their technology function to support expanded digitization of both the front and back ends of their businesses.<sup>5</sup> Software has accelerated some critical use cases for banks on both the front end (e.g., digital banking, next-product-to-buy recommendations) and the back end (e.g., credit risk analytics, fraud detection).

Furthermore, the durable shifts driven by COVID-19 are putting immense pressure on

technology capabilities (e.g., significant shifts in demand for digital channels, remote working, new cybersecurity threats), requiring executives to anticipate and prepare for the future.<sup>5</sup> For example, 15 to 20 percent of US customers said they expect to increase their use of digital channels once the pandemic has passed.<sup>6</sup>

However, driving a software transformation in banking also has unique challenges. For example, banks operate in an environment of complex regulations with stringent security and compliance requirements. These are also rapidly evolving to incorporate more complex risks as adoption of cloud, AI/ML techniques, and new technologies like crypto expand. This requires banks to automate governance and controls to ensure business and technology teams have ready access to appropriate datasets, with the necessary controls for security and permission where needed. Additionally, many banks recognize that their feature development and release cycles are longer and more complex due to

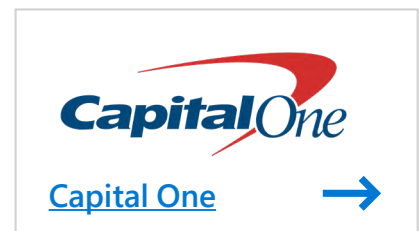
<sup>5</sup>[Next-gen technology transformation in financial services report \(McKinsey, 2020\)](#)

<sup>6</sup>[Transforming the US consumer bank for the next normal \(McKinsey, 2020\)](#)

regulatory compliance issues, and thus are adapting a fit-for-purpose Agile methodology. Long-term planning methods are used to create feature plans and to manage dependencies. These plans are then broken into checkpoints and MVPs for development teams to build using Agile methodologies.

To understand how banks are adapting and using software to accelerate business performance, we studied three banks in-depth: ABN AMRO, Capital One, and a leading North American bank.

While we included three banks in this research, for this document we will discuss only two case studies in-depth. All three are large global institutions that have taken very different paths to becoming world-class software organizations. However, we observed some common patterns with banking; a strong, all-in mindset with regards to software that is jointly driven by business and technology, and a phased, multi-year approach starting with building the foundations before advancing to more mature capabilities. They have also taken a very thoughtful and balanced approach to driving speed and agility while meeting the highest standard around security and compliance, and viewing cloud adoption as an opportunity rather than a barrier.



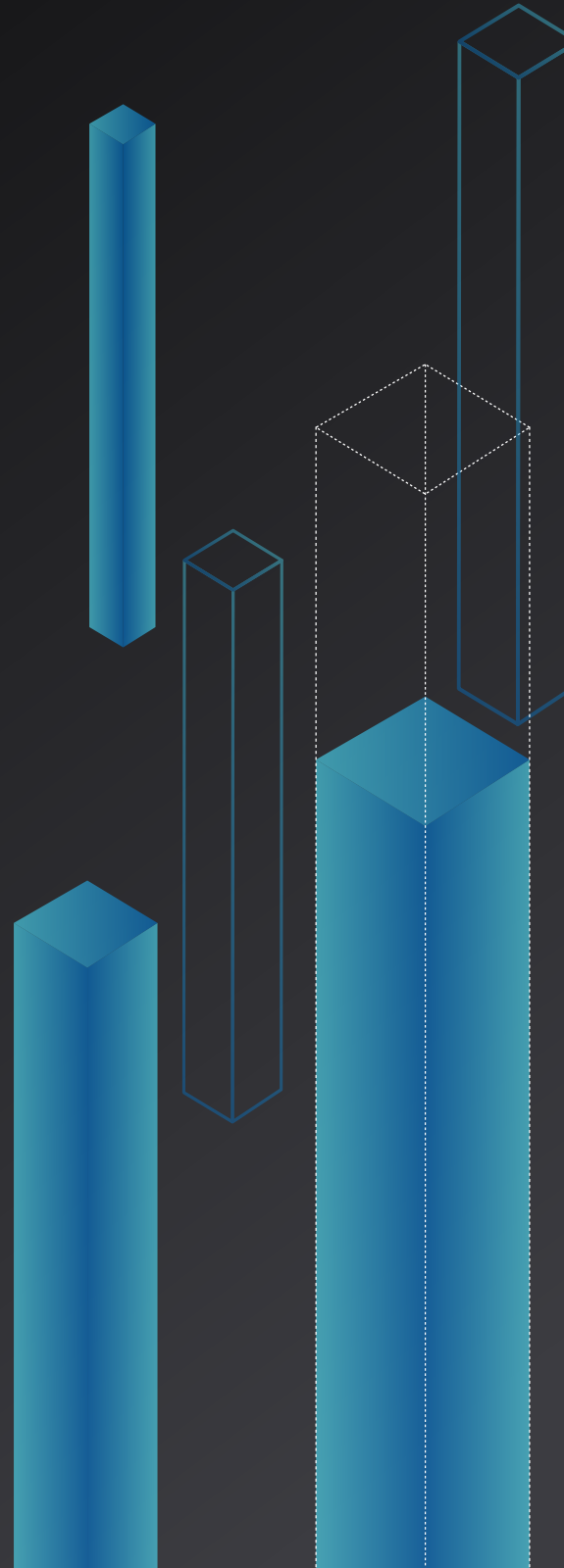
# ABN AMRO

## ABN·AMRO

Strong focus on software development to address growing demand for digital banking.

Increased productivity by standardizing development tools and the CI/CD pipeline, and reinventing its developer team operating model.

Today, ABN AMRO's investments have led to dramatic improvements in software development: testing time has reduced by 95 percent, 45 percent of workloads are in cloud (with Microsoft Azure as primary cloud vendor), and 100 percent of teams use Agile methodology.



**ABN AMRO is a leading global bank that focuses on commercial, private, and retail banking. As digital banking grew in the early 2010s, ABN AMRO recognized the need for greater control over its digital products and faster time-to-market.**

It embarked on an ambitious software development transformation through two concurrent strategic initiatives: standardizing development tools and practices and reinventing the team working model.



## **Standardizing development tools and practices**

2016–2017

ABN AMRO quickly realized that standardizing toolsets across teams would help ensure that developers could easily move among teams, and that employees from one org could easily work with employees from another—whether developers or business leaders. The company created a Center of Excellence (CoE) to develop tooling standards, enforce adoption, and provide training to all teams (e.g., Jenkins and Azure DevOps for deployment). Importantly, standardized tooling extended to business functions and external vendors. For example, everyone uses JIRA for product management, scheduling feature requests, and release schedules. Teams may request exceptions when standardized tools do not meet their requirements.

ABN AMRO then focused on consolidating CI/CD pipelines into one pipeline per platform, each of which enforces numerous quality checks. Every release candidate must pass stringent requirements, including security, automated unit/functional/regression testing, and coding style. Standardized controls ensured that quality expectations were fully transparent to developers, while also providing easier oversight and management to technical leads.



**When we standardized and enforced controls, the quality improved dramatically, especially in security ... everyone knew the standard they were held to."**





## Reinventing team operating models

2016–2017

ABN AMRO wanted the core unit of development to be a single Agile team, with each team accountable for a product end-to-end. Technical leaders thus created standardized teams of 8 to 10 people, each with developers who specialized in specific areas such as tooling, security, and testing. Teams often included an embedded business product owner who guided planning and ensured sprint cycles aligned with business priorities.

Further, a core ABN AMRO value is that any third party it works with is considered a partner, not a contractor. Partners were included in broader decisions (such as technical roadmaps) and internal trainings. Some ABN AMRO employees even relocated to offshore development facilities to form closer bonds with partner teams. Their efforts were so successful that many partner teams hung pictures of Amsterdam on their walls and adopted ABN AMRO's values as their own.

### Impact

The investments to standardize tools, engineering practices and reinvent the team operating model have paid significant dividends. Productivity increased by 33 percent after the changes, all teams use Agile methodology, testing time has declined by 95 percent, and 45 percent of workloads are cloud-based, with Microsoft Azure as primary cloud vendor.



# 33%

Productivity increased



# 95%

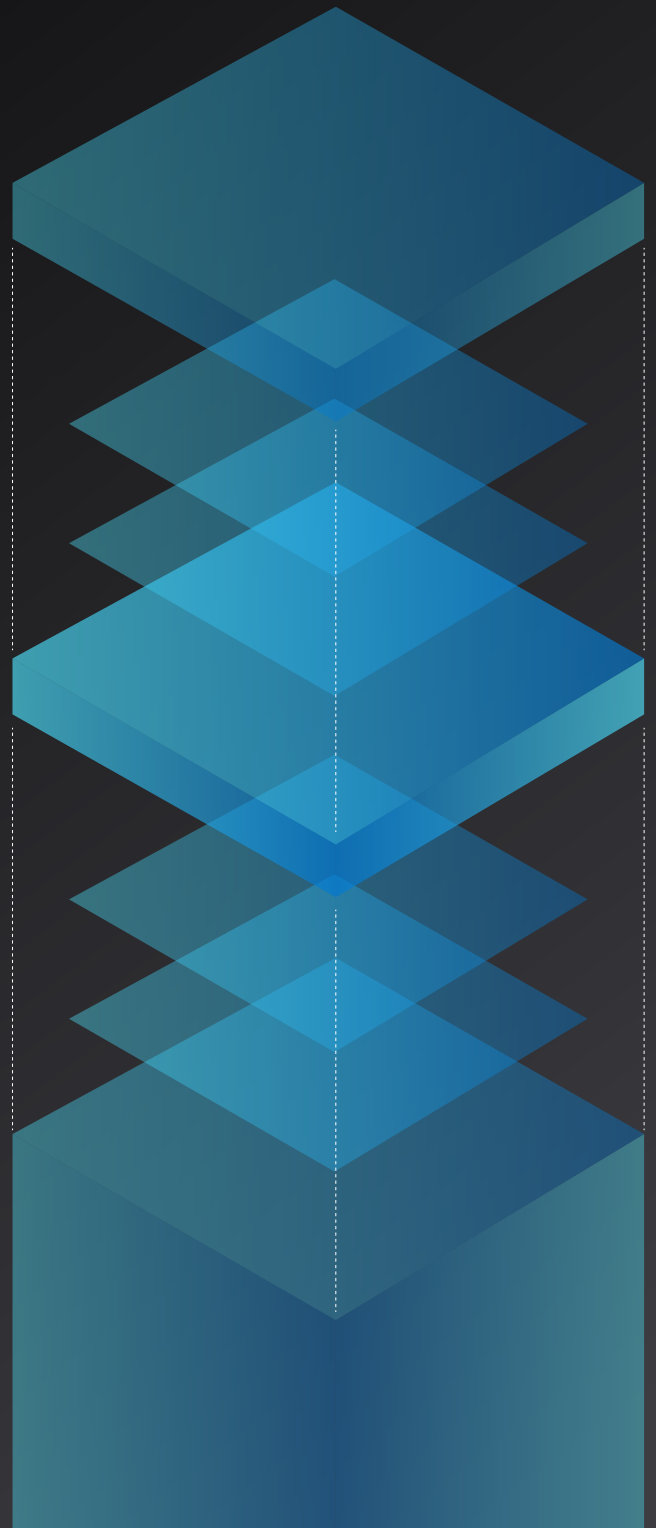
Less testing time

# Capital One



The bank is an early adopter in digital and achieved software success through a well-planned, three-phase journey beginning in 2011.

Some of the highlights include 80 percent of workloads are now cloud-based and 75 percent of customer interactions are digital.



**Capital One is a leading bank focused on credit card, consumer banking, and commercial banking. Its CEO and executive team have for more than a decade recognized that an all-in commitment to technology, and specifically cloud technology, was the best way to ensure Capital One could meet the growing customer demand for digital banking and stay ahead of fintech disruptors.**

Through this period, it has successfully moved the majority of workloads to public cloud hosting and the majority of software development in-house. It also has cloud-native development, attracted top development talent, and built a strong technology culture. Capital One's transformation was executed in the three phases outlined below.



Phase 1:

## Laying the foundation

2011–2014

The journey started with owning the technology stack end-to-end to enable the flexibility and speed that full ownership provides. This included in-sourcing tech talent, adoption of Agile and DevOps, and modernizing tech infrastructure.

### In-sourcing talent:

Capital One values in-house talent across the company to drive employee ownership and quality. Hiring technical talent became a key focus, especially concentrating talent on primary areas that drive business growth. Building partnerships with universities was especially important, as was training and growing early-in-career talent with both banking domain expertise and technical knowledge. Capital One attracted talent with a unique value proposition: solving real world customer problems at a scale of millions of users per day.

### Agile development practices:

In early stages of the transformation, every team adopted Agile with consistent team set-up and sprint cycles. Shifting the culture to thinking in terms of continuous delivery and improvement enabled Capital One to break ambitious projects into concrete, incremental steps that delivered on time. Teams also adopted shared collaboration tools to ensure Agile teams could work together easily.



**Early adoption of agile was key. We had to shift the developers' mindset before we changed anything else."**



### Moving to a microservice oriented architecture and APIs:

Capital One wanted each Agile team to have end-to-end ownership of a specific service, and thus architects decomposed the monolithic code base into a service-oriented architecture. Giving each team ownership of an independent service enabled Capital One to prioritize upgrading individual services according to business needs using a single accountable team. Furthermore, embedded business specialists helped translate business priorities into technical upgrades scheduled for specific Agile sprints, ensuring complete alignment between business and technical priorities.



#### Phase 2:

### Migrating to the cloud

2014–2018

### All-in, joint business and technical commitment to migration:

The CEO announced an all-in commitment to a cloud-based model, and the senior business and technical leaders pushed progress to quickly remove roadblocks through daily stand-up meetings.



**The aggressive, top-down commitment to the long-term vision was remarkable...the fearlessness of an all-in commitment to cloud cannot be underestimated."**

### Development teams choose their own migration schedule and strategy:

Tech leaders gave teams control over their migration schedule and strategy, but provided an assessment framework to aid decision-making. The framework included both business and technical needs, from anticipated usage and cost to security needs and software dependencies on other teams. Teams then made their own migrations plans, with light-touch coordination from a central infrastructure team.

### Early adopter teams share best practices:

Early adopter teams pioneered cloud migration. The earliest teams experimented with different migration strategies, cloud architectures, and tools until they found a set that worked for the bank's needs. Those teams then helped guide the next waves of adopters by establishing best practices, toolsets, and step-by-step guides.

**Phase 3:****Advanced phases of maturity**

2018–present

The current phase focuses on enforcing consistently high-quality delivery across all teams. Historically, teams had the autonomy to develop their own CI/CD practices with light guardrails to ensure they were meeting company standards. When technical leaders started seeing diminishing returns from team-level investments in DevOps tooling, they adjusted the approach, moving to a shared enterprise pipeline framework that would enable more rapid automation of controls and standardization of critical processes. All code must pass more than 20 compliance checks enforced in the CI/CD pipeline, including unit, functional, integration, and regression

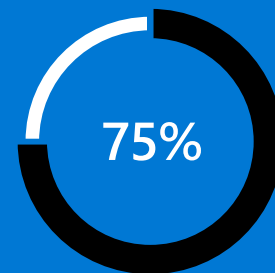
tests, security compliance checks, style-guide conformity, and more. Only code that passes all tests is approved for release. Capital One therefore preserved teams' autonomy to develop and innovate, while also gaining efficiency and ensuring all software meets the same stringent quality standards for release.



**Aligned autonomy ... it is giving our teams enough shared context and clarity on where to go, and then let them be autonomous to accomplish it."**

**Impact**

Some of the highlights of impact include majority of workloads being cloud-based, majority of software development in-house, cloud-native development and talent, and a strong tech culture. More than 75 percent customer interactions are digital today.<sup>7</sup>



Digital customer interactions

<sup>7</sup>[Doing business the digital way: How Capital One fundamentally disrupted the financial services industry \(Capgemini Consulting, 2017\)](#)

# Deep dive on learnings in retail



**For retailers, software has increasingly become the basis for growth, including hyper-personalized experiences and omnichannel adoption. These changes drive high-priority digital use cases on both the front-end (which affects customer experiences) and the backend (which helps manage core retail systems such as merchandising and supply chain).**

The rewards for those who get this right can be significant. For example, McKinsey research suggests that effective personalization can increase store revenues by 20 to 30 percent.<sup>8</sup>

In this context, our earlier research highlighted the importance of Developer Velocity in retail. In the light of the COVID-19 crisis, there is even more urgency for companies to invest in effective and impactful software development to adapt and thrive. COVID-19 drove significant durable changes for retail; by some estimates,<sup>9</sup> retailers have vaulted 10 years ahead in consumer and business digital penetration in less than three months. Forty-six percent of US consumers switched

brands or retailers through the pandemic and e-commerce spending was up 30 percent from the beginning of March through mid-April of 2020 as the appetite for digital and contactless ways of shopping intensified.

However, driving a software transformation in retail is not without its unique challenges. Large retailers handle millions of transactions across diverse channels, and need to maintain consistency of experience and data across these transactions and channels. Additionally, retailers that are driving growth across physical and digital channels can have a fragmented technology landscape. For example, they may be using legacy on-prem technology for point-of-sale and store management systems at their physical locations and cloud applications for their e-commerce stores. It may only take a few minutes to publish new features to their mobile apps, yet updating physical store systems may take weeks.

Finally, many retailers own multiple brands across multiple geographies, often as the result of acquisitions. This can also produce a fragmented technology stack, with disparate systems that do not communicate.

<sup>8</sup>A transformation in store (McKinsey, 2019)

<sup>9</sup>The reinvention of retail report (McKinsey,2020)

Many retailers recognize this challenge and are investing in the consolidation of their technology stack data platforms, and are building loosely coupled services that are available as a platform for developers.

To understand the ways that successful retailers are overcoming these constraints, and leveraging software to differentiate themselves, we deeply studied two leading

**Finally, many retailers own multiple brands across multiple geographies, often as the result of acquisitions. This can also produce a fragmented technology stack, with disparate systems that do not communicate. Many retailers recognize this challenge and are investing in the consolidation of their technology stack data platforms, and are building loosely coupled services that are available as a platform for developers.**

companies that operate in very different parts of retail: Albertsons—a grocery company—and a global Quick Service Restaurant (QSR) chain. Both are large global brands that have software as a company priority, and are on a mission to become a world-class place for developers to grow and innovate. Aside from in-depth research into these two retailers, we also interviewed two more leading retailers with global footprints for additional insights.

For each of these digital leaders, we observed a methodical and phased approach to driving a multi-year software program. They have invested in foundational capabilities with Agile, DevOps, and close integration of business and technology. They have also made an ambitious push to public cloud starting with the customer facing experiences and the e-commerce stack, with an eye to driving speed.

The companies we studied have seen massive impact from their investments in software. Their digital businesses have grown one to three times in recent years, driving business success even during the COVID-19 pandemic.



[Albertsons](#)



**Global Quick Service  
Restaurant (QSR) chain**

[QSR chain](#)



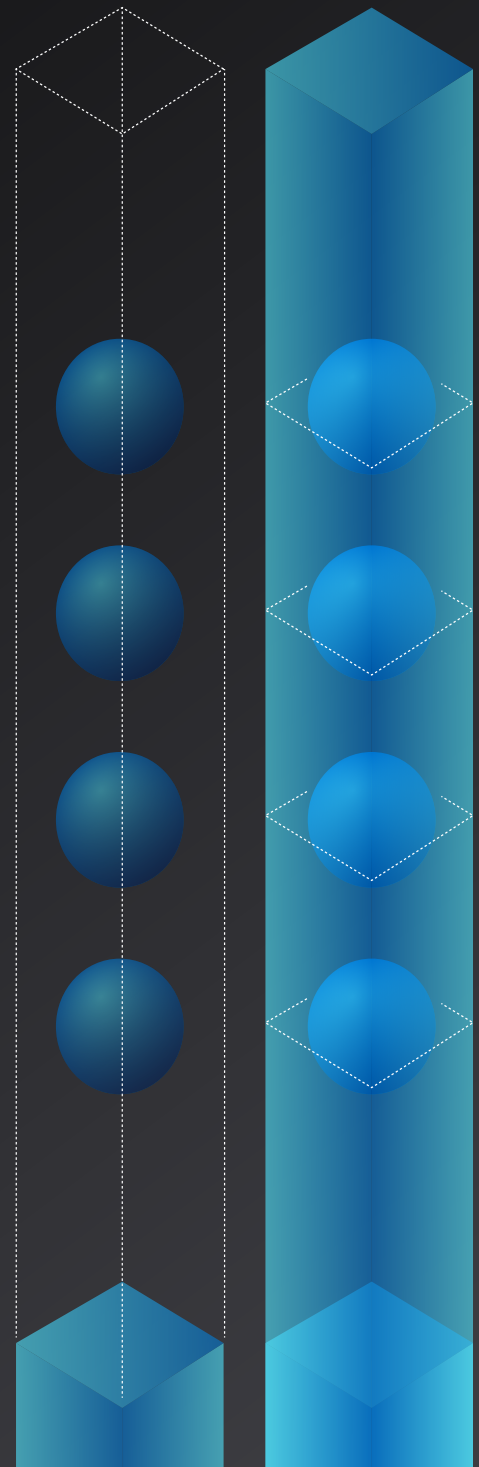
# Albertsons



Transformed software development to meet rapidly growing demand for online orders.

Focused on having cross-functional Agile teams, developing a loosely coupled architecture, and migrating to public cloud.

Impact of the investments has been high: Albertsons e-commerce is a key growth driver, as it recently achieved its third straight quarter of more than 200% sales growth, up 225% in Q3 of FY 2020.



**Albertsons Companies is a leading US grocer with 2,000+ physical stores and more than 20 store brands including Safeway, Albertsons, Vons, Jewel-Osco, Shaw's, Acme, Tom Thumb, Randalls, and United Supermarkets.**

Albertsons transformed its software development capabilities in multiple phases. First, Albertsons adopted a structure for its development teams that enabled autonomy and agility. Teams would be cross-functional and own the product end-to-end. Simultaneously, Albertsons drove an aggressive migration to public cloud and developed a loosely coupled architecture.



Laying the foundation  
**Building standardized Agile teams that own unique services**  
2017–2018

The first challenge the Albertsons team took on was making the organization more responsive to customer needs. Albertsons rearranged teams into cross-functional, independent teams with 6–10 developers, PMs, architects, and testers. Each team now owns a service end-to-end. Teams also began using a set of standard collaboration and code management tools, including GitHub.

A central team of enterprise architects then led the effort to develop loosely coupled services (e.g., in e-commerce: shopping cart, checkout.) Each team then owned a single service over which it had full accountability and ownership.



Moving beyond basics

## Migrating the e-commerce stack to Azure and rearchitecting individual services

2018–2019

Albertsons chose a two-step approach to migrate the e-commerce technology stack to Azure. First, they adopted a lift and shift approach to migrate the e-commerce stack to cloud. The leadership team ensured there was a clear execution plan along the way that was communicated to both business and tech to

ensure momentum. Individual services were rearchitected independently with a test-and-learn approach to ensure zero downtime. Services were migrated sequentially; each individual service was upgraded, stabilized, and incorporated into the e-commerce stack before moving to the next service. The central architect team sequenced services based on a combination of business and technical needs, with critical customer features or foundational tech dependencies (e.g., the data model) going first. This sequential, progressive modernization also ensured the team was able to learn best practices from early upgrades that were useful for later services.

### Impact

Impact of investments at Albertsons has been high. E-commerce is a key growth driver for the company, as it recently achieved its third straight quarter of more than 200% sales growth—up 225% in Q3 of FY 2020. Developers are also proud of their work. Launch time for new customer features has reduced by more than 85 percent, and testing time by more than 50 percent, while maintaining quality. About 70 percent of the customer-facing e-commerce stack is in the public cloud, compared to zero percent just three years ago.



# 85%

Launch time decreased



# 50%

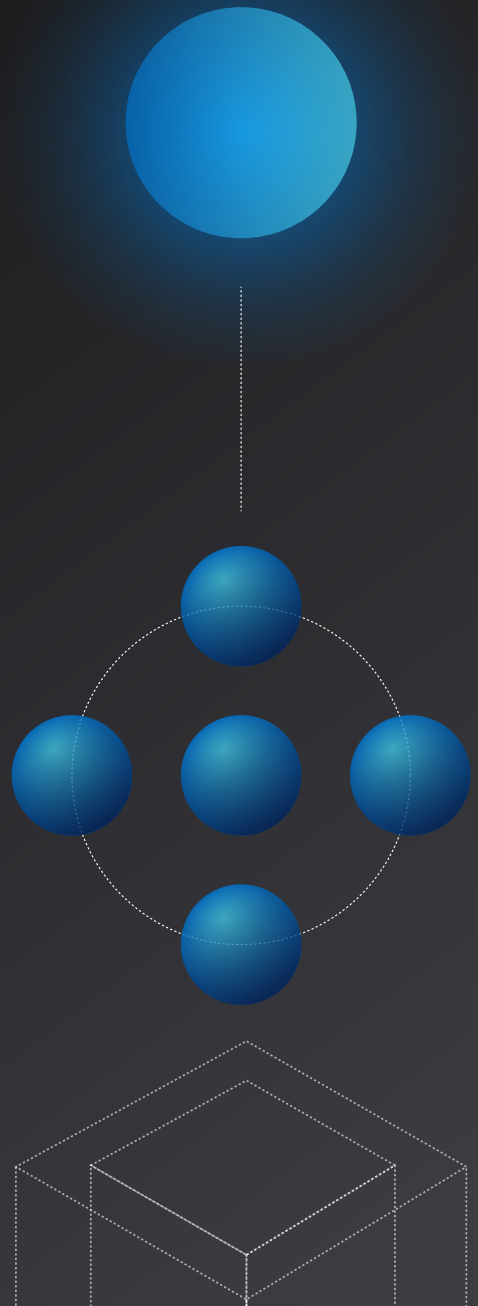
Less testing time

# Global Quick Service Restaurant (QSR) chain

Digital transformation with a goal to meet customers' increasing demand for digital experiences, especially mobile ordering, loyalty, and AI-driven apps.

Transformed its development org with investments in cloud, standardization of engineering practices, and focus on talent and culture.

One achievement that stands out is that digital sales have grown 100 percent in the last two years.



**This retailer is a global, fast, casual restaurant chain with \$10B+ revenues in 2019. It has transformed its software development organization over the years via a strong focus in three areas: cloud migration, adoption of world class tools and engineering practices, and a focus on talent and culture. These key investments helped it dramatically improve digital performance. In the last two years, digital orders have grown 100 percent.**



## Migration to public cloud

In 2017, the company drove an aggressive push to public cloud. It adopted a three-pronged approach to making this transition successful; a joint partnership across business and technology, ownership and autonomy to teams, and a cloud technical director leading a cloud Center of Excellence (CoE).

As a starting point, the company decided a cloud-migration initiative supported with executive commitment was the best way to ensure a successful migration. Executives were given specific cloud-adoption initiatives and targets to incentivize migration. They also recognized that each department had its own technical needs, maturity, and business case for cloud migration. Each department was given target timelines to migrate but empowered to choose the right strategy. For example, one team strategically migrated services according to the business case, optimizing for cost and rapid scalability to meet growing demand. In contrast, another team migrated more opportunistically. Many strategies can lead to a successful cloud migration, and the retailer recognized that its orgs were best suited to decide their own paths.

Lastly, technical leaders created a cloud Center of Excellence (CoE) to accelerate the migration. Their focus was on evangelizing the technical benefits of cloud migration, leading education and training efforts on cloud technology, creating learning curricula for developing in the cloud, and disseminating best practices. The CoE ensured that every organization had access to the expertise and coaching needed to make the migration successful.



## Adoption of world class tools and engineering practices across the entire organization

On tools and engineering practices, the company focused in three areas: standardized tools and processes for cross-functional collaboration while maintaining autonomy for teams, full-stack DevOps teams using standardized CI/CD pipelines, and security embedded into day-to-day development.

As a foundational capability, the QSR drove high levels of consistency in tools and processes, but empowered the teams to have the flexibility to select tools and ways of working that are unique to the individual team. JIRA planning tools, Confluence documentation, and GitHub repositories are required for working across teams. Cross-training and standardized learning journeys ensure that developers are well trained on all toolsets, enabling developers to move among teams with minimal friction.

Additionally, the company has made a significant push on adopting the DevOps model. All teams are full-stack DevOps teams; responsibility for the service end-to-end instills a sense of ownership and enables teams to see the impact they have on customers daily. However, granting every team responsibility

for its DevOps requires them to enforce strict quality and compliance controls to ensure high standards are met across all teams. The QSR requires all code to be released through standardized, automated CI/CD pipelines to enforce quality at each step. At code check-in, security and compliance checks run automatically. Tests, including unit, functional, integration, regression, and capacity tests, automatically run every night and during deployment. Finally, the production code is constantly monitored, and alerts sent whenever issues are detected.

Lastly, embedding security in the day-to-day development cycles is a priority for teams. All developers are trained in secure-by-design methodology and incorporate best practices into the code. They also enforce robust data classification systems, ensuring that data receives differential treatment and privacy depending on its sensitivity. A central security team is also available to help developers refine their work, from providing risk assessments to consulting on how to improve service security.



## Investments in talent and culture

The retailer recognized early on that developer talent was critical to its success. When competing for talent, the focus is on clearly articulating the mission and what makes it a unique place to work. For example, the ability to work on systems that transact millions of orders per day globally, across both digital channels and through thousands of brick-and-mortar stores. The demands of

meeting that scale and crossing the physical-digital divide attracts developers looking for unique challenges. The company also focuses on engaging with the developer community via conferences and events. Most importantly, the retailer is well known for deeply caring for its employees, and that reputation alone attracts thousands of applicants every year.

### Impact

The investments in cloud, engineering practices, and talent have paid significant dividends. Almost all code is deployed through standardized CI/CD pipelines, and a significant portion of infrastructure is on the public cloud. Digital sales have grown 100 percent in two years.

↑ **100%**  
Digital sales growth

# Learnings that apply across companies



Even though every company is unique, technology leaders at each of the companies we studied faced similar challenges in their software transformation. Their responses and lessons are valuable for other companies regardless of industry.


## On Technology

1. For cloud adoption and modernization of business applications, two elements emerge as enablers for success:



**Joint partnership across business and technology with focus on accelerating business outcomes**

Successful companies, often have business and technology come together to agree on specific timelines and outcomes and then enable and empower development teams to identify the optimum migration path. For example, one US banking CEO set hard timelines for teams to move resulting in a successful transition. At another bank, effort floundered for a long period as there was no top-down mandate and migration was seen as an “engineering effort.” In coming together, business and technology can agree on specific timelines and outcomes and then enable and empower development teams to identify the optimum migration path.



**Cloud Center of Excellence (CoE) that builds a “scaffolding” and provides support for teams as they migrate**

Successful companies create a recommended path for cloud adoption and application modernization that makes it easy for a vast majority of teams to thrive. They also give teams with advanced track records or unique needs the flexibility to choose their own tools, architecture, and services as long as they adhere to baseline standards on security, quality, availability, performance, etc. in the cloud.

- 2. For tools, companies often look to standardize the software development toolchain in a few key areas—typically CI/CD pipeline, planning, and code management tools—and leave flexibility in other areas to individual teams and developers.**

Earlier research showed that tools are the top driver of Developer Velocity; companies with top-quartile scores on tooling have developer satisfaction scores that are 47 percent higher than companies in the bottom quartile. However, many companies struggle with the balance between maintaining a level of standardization versus empowering teams with the flexibility to use tools of their choice.

We observe that standardization in a few key areas such as the CI/CD pipelines can drive higher levels of autonomy and psychological safety for developers. Standardization helps increase the confidence of developers to push code to production while reducing friction and manual quality, risk, and compliance reviews. Standardization of planning and code-management tools helps organizations coordinate and manage dependencies more easily, allowing developers to distribute knowledge and share learnings. In other areas, organizations should look to be flexible and allow developers to pick tools that support success. For example, flexibility in dev tools such as an IDE is a critical factor that contributes to developer happiness in many organizations.

Some of our case studies found that early software transformations were fragmented by organizations that experimented to identify the best tools. This fragmentation can make the adoption of standard tools difficult. After ensuring uptake is supported by changes in culture, decision makers can clearly and positively communicate the tools' benefits for the individual developer.

- 3. Data architecture and data platform modernization are the next evolution of application platform modernization efforts for even the most successful companies.**

In most cases, application platform modernization—through application rearchitecture and cloud adoption—is a high priority at the outset of a software transformation. Many organizations have varied investments for different categories of apps. For example, while moving apps to the cloud, many companies incrementally rearchitect applications using microservice-based architecture and Kubernetes as a standard way for customers to orchestrate containers at scale.

However, data architecture and platforms are often under-invested areas when addressing the full scope of application modernization. Data is inseparable from a company's effort to capture benefits such as enhanced personalization, improved fraud detection, and real-time alerts. The data lakes and customer data platforms required to support such technologies have increased the complexity of a company's data architecture and created more heterogeneity and duplication. The most successful companies recognize the challenges in data, and are starting to prioritize and invest in this area. Specifically, these companies have begun to modernize their data ecosystems and architecture, including consolidating data pipelines, low-to-no duplication of databases, and standardized tools and data structures. They are also building supporting capabilities

such as data governance tools and processes, incorporating data scientists and engineers into Agile teams, and training teams on responsible usage of AI/ML.

4. **Security is a major area of ongoing investments** (including shift-left, automation, and embedding controls in the pipelines itself). Companies approach cloud as an opportunity when doing this.

In a world where customers infrequently contact staff, but rather interact almost entirely through digital channels, "digital trust" has fast become a significant differentiator of customer experience. Especially with cloud adoption, security is fundamentally shifting from being centralized and run by a handful of security experts, to decentralized and in the hands of thousands of individual developers. Successful companies are building security at the most fundamental level in their tech stack. They are embedding security in their pipelines, driving high levels of automation, and embedding secure development practices into the Agile teams themselves. As they do this, companies approach cloud as an enabler, through ability to automate, embed controls and provide access to the latest tools and technologies.

## On working practices

### 5. **Autonomous scope and loosely coupled architecture are must-haves for Agile to succeed.**

Agile methodologies, where teams are not autonomous and cannot deliver business value independently, are unlikely to deliver favorable results. Successful companies tackle these topics concurrently. For example, one executive, when asked about what drove success early in their company's software journey, told us "Agile is equal to architecture." On the other hand, the lack of architectural changes alongside an Agile transformation can result in a high degree of interdependences in backlog, significant time spent on "stabilization" or "integration" of codebase, and a general culture of lack of ownership, empowerment, and accountability.

In addition to architectural changes, companies also reorganize teams to have a more autonomous scope. In this model, teams have end-to-end ownership of a product, are able to push to production independently, and can be accountable for delivering business outcomes.

### 6. **Open source software is a differentiator between the top performers versus the rest**—particularly for attracting and retaining top talent and increasing speed of innovation.

We found in our earlier research that open source adoption is highly correlated with outperformance in business results and innovation. Companies with high open source scores realize three times more impact on innovation and 20 percent higher developer satisfaction compared to peers, assuming foundational capabilities are strong.

Using and contributing to open source projects is a significant way to attract and retain development talent. Indeed, the ability to contribute to open source projects is an important way for developers to build their network and create a stronger technical reputation. Usage and contribution to open source code is an important way for companies to signal that they embrace innovation, no matter the source.

### 7. **Inner source adoption is early across the board, but a top priority for more mature companies.**

Inner source is a topic that we found to be top of the agenda for all companies as they look to move into more advanced stages of software development excellence. This is an area that is still nascent in its adoption, yet companies are seeing strong results even in early stages of adoption. Like open source, inner source adoption creates a culture of openness, transparency, and knowledge sharing, increasing the speed of innovation

and bringing follow-on benefits in culture and the ability to retain the best talent. Some successful companies are increasingly shifting to a platform-oriented model, encouraging teams to both contribute services to the platform and leverage services wherever possible.

## On organizational enablement

### 8. Unsurprisingly, world-class talent differentiates the top performers from the rest.

Companies across industries compete with leading tech companies for the best talent through culture and emphasizing their unique value proposition (i.e., “the ability to work on products that impact the lives of millions of people every day”).

In our previous research, talent showed up as top four driver for business performance. Successful companies make technical talent a C-suite priority and articulate a distinctive value proposition to appeal to technical talent. For instance, many retailers tout the opportunity for developers to solve technical problems that affect millions of consumers, especially in the context of the COVID-19 crisis. The top performers in attracting talent also invest in campus programs and leverage their brand recognition to attract talent from top local and national programs.

Another approach we see companies take is to focus on a smaller percentage of top talent that can be embedded in the most important roles. They use this critical pool of talent to both accelerate the most important projects and build capabilities in the rest of the organization. The focus should not just be on leaders, but also strong individual contributors who can have disproportionate impact. The hiring, retention, performance management, and succession planning in these critical roles are often led directly by CxOs.<sup>10</sup>

### 9. Product Management (PM) is the next frontier for many organizations. Common challenges include: balancing PM skills across business domain knowledge, experience design and technical skills, lack of a well-defined product- and platform-centric operating model across “business” and “technology,” and lack of a strong TPM function.

Based on our previous research, a well-rounded product management function is critical. A strong product management function is even more critical with remote and distributed teams, where clarity and alignment on outcomes matter more than ever. Far too often, product managers in banks and retailers have been merely a re-branding of an existing business role, resulting in gaps in capabilities of areas

<sup>10</sup>[Linking talent to value \(McKinsey, 2018\)](#)

such as experience design and technical skills. Successful companies realize that there are no shortcuts, and they are investing significantly in hiring and developing PMs who have balanced skills across business domain knowledge, experience design, and technical skills.

Another common failure model we observed is the presence of an unclear product- and platform-centric operating model. This often starts with the lack of a clear definition of a product or platform. An effective operating model should include a holistic definition of product and platform that is not constrained to monetized experiences, but rather any software that delivers value to an internal or external stakeholder. Effective operating models should also include clear product roadmaps (versus Short-term projects), Agile funding mechanisms, outcome-based goal setting (leveraging OKRs for example), quarterly and monthly product reviews, and well-defined roles and accountabilities across the entire development lifecycle.

Finally, the challenge of applying a product management discipline is even greater for complex or large engineering projects, such as modernizing the tech stack. At software companies, technical product managers (TPMs) tend to fulfill these roles, but many companies outside the software industry lack

a comparable role and struggle to achieve their goals from these efforts.

**10. Finally, cloud adoption and modern tools differentiate companies that have parity between in-person and remote developer productivity.**

The context of the COVID-19 pandemic accelerated the shift to remote around the globe. Public cloud adoption and tooling sophistication enabled software development teams in retailers and banks to have developer productivity parity between in-person and remote. For example, adoption of collaboration tools such as Microsoft Teams skyrocketed for organizations to connect remotely. Existing CI/CD toolchains enabled development teams to collaborate effectively on code base, and public cloud adoption enabled companies to quickly scale resources to support an influx of employees working remotely—and millions of customers performing transactions online.

Many employees are likely to continue to work remotely after the COVID-19 crisis. An April 2020 Gallup poll found that three in five US workers who have been working from home during the COVID-19 pandemic would prefer to continue to work remotely as much as possible, even after public health restrictions are lifted.<sup>11</sup>

<sup>11</sup>[U.S. workers discovering affinity for remote work \(Gallup, 2020\)](#)

# Implications for technology leaders



**Becoming world-class at developing software is a strategic imperative for companies across industries. As we have seen from the case studies, getting there is a multi-year journey. The most successful companies, ones that are almost indistinguishable from a native software company, have been on this journey for close to a decade. And in the true spirit of continuous improvement, they are continuing to explore new horizons.**

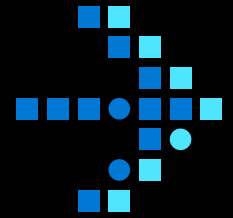
In the early stages of transformation, leaders should look to invest in adopting Agile practices and restructuring into small, autonomous teams with persistent missions. Leaders should bring business and technology together on a day-to-day level, attracting top-tier talent in-house, and investing in cloud adoption and application modernization.

Leaders should help guide the organization through ongoing challenges: the continued adopting of the latest cloud technologies; rearchitecting into microservices; adopting open source; embracing organizational

agility (that is, going beyond Agile at the team level to agility in enterprise-wide planning, dependency management, risk management, etc.); continuously integrating; and continuously refining deployment processes. If done well, the above capabilities will deliver significant results through the ability to reach customer and market inputs quickly and go from idea to release in days and weeks versus months and years. However, leaders cannot rest on their achievements. The most successful organizations are taking their continuous improvement journey and investing in areas such as DevSecOps, building true product management and design capabilities (similar to software companies, where product managers can own and deliver end to end business outcomes), adoption of low-code tools, and transforming their data and analytics architecture.

The individual journey, pace, and sequence of how the different levers are deployed will be unique to every company. However, there is one element that we believe is a constant: transformation requires a close partnership across business and technology, with the focus of the entire leadership team, over a long period of time.

# How to increase Developer Velocity in your organization



Developer Velocity helps you unleash the full potential of your developer teams, drive innovation, and boost business performance. Today, more than ever, investing in software excellence and building a culture that empowers developer teams will continue to be critical for every organization's success.

The digital transformation journey often begins when a company realizes it increasingly needs to operate like a software company. Developer Velocity is essential for a company to drive better business results through empowered developer teams.

The Developer Velocity Index (DVI) is a score calculated by assessing a company in three categories: Technology, Working Practices, and Organizational Enablement. You can benchmark the maturity DVI score for your organization relative to the peers in your industry. The Developer Velocity assessment provides key learnings and actionable guidance to boost your Developer Velocity and drive business outcomes.

Calculating your Developer Velocity Index is straightforward. For this purpose, Microsoft is helping many of our customers by providing unique guidance and recommendations to increase Developer Velocity through the Developer Velocity Assessment tool.

Take a self-guided Developer Velocity Assessment by visiting <https://aka.ms/dva> or by clicking on "Take the assessment" at [azure.com/developervelocity](https://azure.com/developervelocity).

Microsoft

## Developer Velocity Assessment

Power up your developer teams and take your business to the next level

Gain deeper insights into your team's performance

Discover your organization's score on the Developer Velocity Index (DVI) by taking your first Developer Velocity Assessment.

By answering a few simple questions, you can benchmark your maturity score relative to peers in your industry. Then, get actionable guidance to boost Developer Velocity and drive better business results.

Get started

Are you a registered user?  
Sign in here to check your results.

See improved business outcomes like other companies with high Developer Velocity<sup>2</sup>, such as:

- 4-5x faster revenue growth
- 20% higher operating margin
- 55% higher innovation

McKinsey & Company Developer Velocity: How Software Excellence Can Fuel Business Performance

# Building a more resilient future together



**Our mission at Microsoft is to empower every person and organization on the planet to achieve more. And our goal is to empower developers with the platforms and tools that will make every organization incredibly successful.**

Over the past few years, we've seen many developers around the world building amazing customer applications and internal line-of-business applications.

In recent months, every organization has been affected by COVID-19 with unprecedented disruption. We've seen massive shifts in the way many organizations connect with employees and customers. Businesses and organizations of all sizes are adopting solutions to enable remote work and are creating brand-new, cloud-based applications while staying secure.

Superior tools can enable developers to be more resilient and help them code, collaborate, and ship from anywhere, securely. They prefer developer tools and cloud platforms that offer ready access to open source ecosystems.

Regardless of coding ability, low-code tools continue to grow for professional developers and business users. Best-in-class tools empower developers within every organization to keep developer talent happy and motivated and accelerate digital skilling.

To help developers build productively, collaborate securely, and scale innovation—no matter where they are—Microsoft offers the world's most comprehensive developer toolkit with Visual Studio, GitHub, Power Apps, and Azure. This unique toolset makes developer teams incredibly productive when building cloud-based solutions.

Developers are also playing a key role in accelerated innovation, building and shipping global and secure cloud-based applications faster.

Our research shows that companies often realize large improvements in business performance when cloud adoption is high. Public-cloud adoption as a catalyst of Developer Velocity is especially strong for non-software companies, having up to four times the impact on their business performance than it does for software companies.

Application migration and modernization is top of mind for many organizations, and more recently, public cloud adoption has enabled companies to quickly scale resources to support an influx of employees working remotely—and millions of customers shifting to online.

Increasingly, we are seeing customers who value the idea of innovating at a faster pace by using cloud-native architectures with loosely coupled microservices, managed databases, AI, DevOps support, and built-in monitoring to detect failures and problems before they happen.

Such apps are typically powered by elastic infrastructure that adjusts to the varying load, supports zero downtime rollouts, and provides low latency access to data worldwide.

Microsoft is working closely with thousands of organizations around the world to help them use the Microsoft cloud to address today's biggest challenges. One of the things that makes the Microsoft cloud unique is how comprehensive it is. With Azure, we provide an end-to-end cloud platform that enables you to develop custom applications and leverage cloud infrastructure. You can also take advantage of our SaaS solutions

like Microsoft 365, LinkedIn, Dynamics 365, and the Power Platform to connect your employees and business processes to your custom applications.

As we enter a new post-COVID era, we see massive improvements on application time-to-market thanks to shifts to remote developer productivity and public cloud adoption.

Whether your goal is to modernize and build new apps, continuously and securely deploy your code to the cloud, contribute to open source, or rapidly build apps with low-code platforms, Microsoft is pleased to play a small part supporting developer teams and thousands of organizations around the world to increase Developer Velocity.

# About the Authors



**Erika Ehrli is a Director of Product Marketing** with Microsoft Azure focused on cloud application development with Visual Studio, GitHub, and Azure. She serves as a cross-functional business leader working across engineering, product marketing, and sales leadership on defining go-to-market strategy for application development in Azure and open source in Azure. She leads a technical product marketing team focused on narratives, customer stories, technical content, as well as flagship event content strategy to support high-impact product launches. In 2019 she launched the cross-company Developer Velocity Thought Leadership program, research, and go-to-market including envisioning and launching the Microsoft Developer Velocity Assessment Tool. Erika has worked at Microsoft for 17 years with different roles in consulting, engineering, and developer marketing.

**Larry Larsen is a Senior Product Marketing Manager** with Microsoft Azure focused on cloud application development and customer engagement. He serves as a cross-team storyteller and produces event content and Developer Stories that explore the challenges and successes of cloud customers. Larry previously led the Channel 9 developer network hosting hundreds of technical shows and Microsoft's live developer events. Prior to Microsoft Larry worked at The Poynter Institute where he founded Poynter Labs, an experimental hands-on technology center that taught thousands of working journalists from around the world how to embrace and leverage new technologies.



Microsoft would like to thank Dr. Nicole Forsgren, PhD as research advisor and reviewer for this paper.

Microsoft would like to thank McKinsey & Company, GitHub, and HashiCorp for their contributions to parts of the Developer Velocity research.

# Resources



[Developer Velocity](#)



[Developer Velocity Assessment](#)



McKinsey & Company Report: [Developer Velocity: How software excellence fuels business performance](#)



McKinsey & Company Article: [Developer Velocity at work: Key lessons from industry digital leaders](#)



© 2021 Microsoft Corporation. All rights reserved.

This document is provided "as is." Information and views expressed in this document, including URL and other internet website references, may change without notice. You bear the risk of using it. This document does not provide you with any legal rights to any intellectual property in any Microsoft product. You may copy and use this document for your internal, reference purposes.