



THE BOSTON CONSULTING GROUP

# TAKING THE RISK OUT OF DIGITAL PROJECTS

By Jon Brock, Rash Gandhi, and Sesh Iyer

**A**GILE HAS REVOLUTIONIZED THE software development world. And for good reason. With its characteristic rapid-fire iteration and continuous-feedback loops, agile has helped companies compress the length of time from development to release. What once took years now takes only months—even weeks. But if companies don't have the right conditions in place before they adopt agile, digital projects can go awry—at great cost. A pure agile approach may not be sufficient if there are too many moving parts: a technology architecture that isn't decoupled, significant project dependencies (such as hard deadlines), or development processes that are not automated. In such situations, the risk level escalates and along with it, the potential for delay or failure.

Through a hybrid approach called programmatic agile, companies that are not burdened with these preconditions can avoid such negative outcomes. This approach overlays traditional program management practices on agile methods. Programmatic

agile, as part of a five-lever approach, helps companies take the risk out of digital projects and transition to modern applications, systems, and platforms deftly and flexibly. Once the levers take effect, risks abate, and delivery dependencies shrink, companies can relax the program management controls and shift to a pure agile approach and self-managing teams.

## Why Digital Leaders Succeed with Agile

Agile by itself is not sufficient for digital neophytes. Their technology is coupled with the data, there are too many dependencies, and they have not adequately invested in automation. Digital leaders, however, have overcome these conditions.

**Decoupled Technology.** Digital leaders decouple the data from the business logic. They make their data available with pre-built application programming interfaces (APIs). Developers that are building new digital applications don't need to know how the underlying applications or databases

work; they simply develop to standardized APIs to access the data. Decoupling allows for the independent—and much faster—development, testing, and deployment of the data and logic. It also means that developers can later add new features, release applications, and issue software patches in a rapid, plug-and-play manner.

**Minimal Dependencies.** The activities, events, and deadlines of the various IT and business teams do not interfere with one another. Because projects have few dependencies, either technical (such as having to create new interfaces to access data in systems) or activity based (such as a go-live date for a new web-based product), they can succeed—or fail—fast.

Decoupled technology and the lack of dependencies foster a pure agile approach and set the stage for investment in automation.

**Investment in Automation.** Digital leaders invest heavily in automating their development, testing, and operations activities in order to dramatically reduce the timescales for releases.

By contrast, when the technology is coupled or significant dependencies exist, an agile approach is insufficient. The effort is often vastly underestimated. Risks crop up along the way, and teams miss critical deadlines. Consequently, projects suffer lengthy delays, require extensive rework, and incur escalating costs.

## The Five Levers

So, when there are too many moving parts, how can a company, regardless of its digital dexterity, make sure to strip risk from its digital projects? We've identified five levers that are essential for eliminating risk. Rather than employing a step-by-step approach, organizations should apply the five levers in parallel.

### LEVER ONE: PROGRAMMATIC AGILE

For digital neophytes, derisking digital projects calls for a gradual transition to agile development. Programmatic agile blends aspects of agile development (such as user

stories, scrums, and sprints) with elements of traditional program management, including detailed planning, the definition of requirements and design up front, and active project management, reporting, and oversight that are linked to the plans, timetables, and organization dependencies.

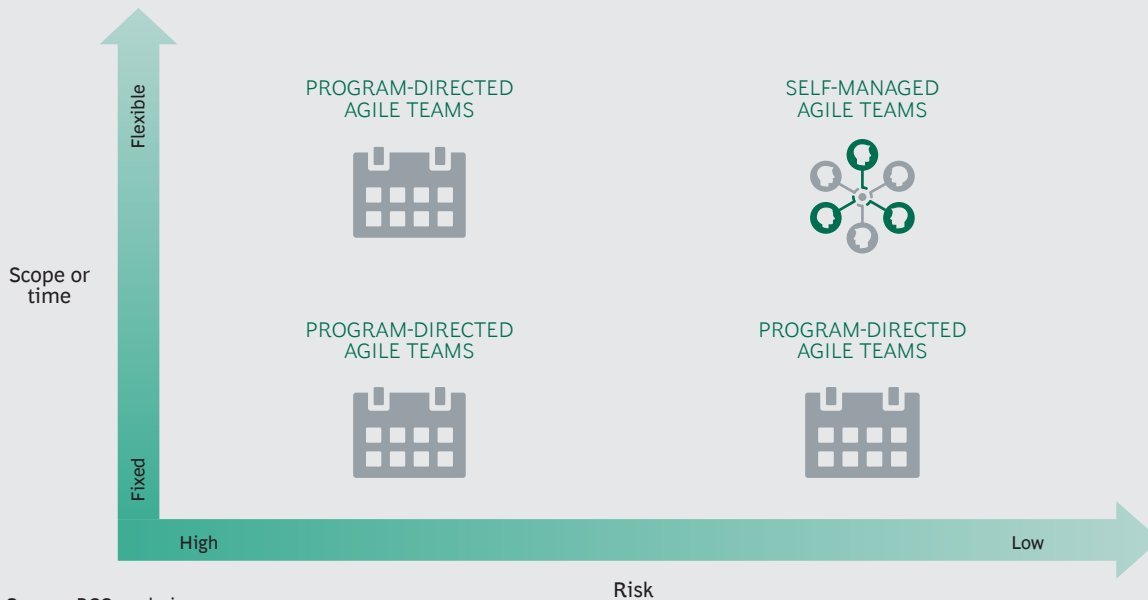
The governance and oversight that programmatic agile provides make it, in effect, an important insurance policy that gives executives the confidence to aim for and hit their milestones and the reassurance that in doing so, neither the technology nor the budget will blow up. Programmatic agile is designed to manage the execution risks that exist as a result of having many dependencies, manual development and testing processes, and little experience building a minimum viable product (MVP). This means that when these conditions are present, it is useful not only for digital neophytes but also for experienced digital professionals. (See Exhibit 1.)

Digital projects that, for example, require coordination of multiple development streams and new technical integration or that have hard deadlines demand more rigorous management and oversight. Instead of relying on self-managed agile teams to plan and execute one sprint at a time—with hopes of achieving the complex deliverables—organizations can use programmatic agile to ensure that all the dependencies and risks are managed over multiple sprints and aimed at achieving a fixed outcome by a specific date.

A good candidate for programmatic agile might be, for example, new software that requires migrating and integrating data with other systems before the release. In such a case, aiming to avoid rework or integration problems down the line, multiple sprint and scrum teams would work out the design early on. Meanwhile, release managers would plan incremental releases on the basis of dependencies, including, for example, deadlines for the new product release and associated deadlines for marketing and advertising campaigns, as well as integration with internal order fulfillment software.

## EXHIBIT 1 | When to Use Programmatic Agile

THE GREATER THE RISKS AND THE MORE FIXED THE SCOPE OR TIME, THE MORE APPROPRIATE PROGRAMMATIC AGILE IS



Source: BCG analysis.

Two other examples are a project with an externally imposed deadline, such as educational software for which there's a set release date, and software with a fixed scope that includes a functional feature required by regulation. Such projects require considerable planning months in advance in order to meet their deadlines. With programmatic agile, a company can create detailed plans to reach major milestones and plan for having the most critical features built halfway or two-thirds of the way through the plan. Doing so will ensure sufficient latitude to hit delivery dates with key features in place.

### LEVER TWO: STRONG MULTIDISCIPLINARY BUSINESS AND IT TEAMS

A crucial ingredient in digital development is a strong multidisciplinary business with IT teams that coordinate closely. Unlike the old way of working—in which the IT team was off on its own, and the business team sat on the sidelines awaiting the results—the two teams work together from start to finish to design, develop, and deliver products and services.

Ideally, the two teams should be “joined at the hip,” working at the same location and

communicating continuously so that they can change course quickly should project objectives shift. Developers should regularly demonstrate workable software to their business team peers to show that the project is on track and to set priorities for future releases.

In a well-functioning joint team, the product owner—the business leader, ever-mindful of the ultimate goal—articulates the project needs and acceptance criteria, approves the application functions that the scrum teams produce, and generally prioritizes tasks and goals. In parallel, the development leader ensures that his or her team is in constant communication with the product owner and flags problems and opportunities as they arise.

Cohesion isn't the only requirement. Joint teams must have the right people in place to ensure the appropriate mix of deep business knowledge, technical skills, and managerial acumen. A product owner who lacks the ability to prioritize will slow project momentum and miss opportunities. One who micromanages runs the risk of annoying the IT team members (who prize autonomy) and, worse, chasing away valuable

tech talent. At the same time, an IT team leader who lacks control or communication skills can end up delivering subpar work that requires rework and leads to delays and possibly product deficiencies. The subsequent and inevitable finger-pointing further undermines cohesion.

Team leaders must understand how to work together and respect their counterparts' knowledge and abilities. Product owners need to be able to push back on developers. Development leaders and their teams, for their part, need to understand the customer in order to prioritize features and create an appropriate development plan. All of this can pose major challenges to organizations and cultures that are not used to working in this way.

### **LEVER THREE: A MINIMUM VIABLE PRODUCT**

In many cases, large projects fail because of their scale and complexity. It takes exhaustive planning to identify and control risk, and even the best of plans cannot anticipate the unforeseeable. The bigger the project, the more time it will take. Some critical problems don't surface until the eleventh hour. And at that point, the resulting application can easily end up being bloated with unnecessary features.

Leading digital companies approach scope in a fundamentally different way. Early on, they invest in the effort to define an MVP: an acceptable product with the fewest-possible and most essential features that will ensure basic functionality. After an MVP is released, the company can incrementally, and in order of priority, add functions on the basis of a product roadmap informed by customer feedback. In reducing the product's scope, companies can contain the risk of delay. They can change the scope as priorities shift.

But defining an MVP isn't easy. It bears repeating that the product must be both minimum and viable, and achieving that is no small feat. Only if a team has established a solid understanding of customer needs can it have confidence that the minimum features it has identified are indeed those that

matter most to the most users. Determining what those features are should involve vigorous discussion and debate among team members from both the business and the IT sides. The teams should sketch a skeleton application and then rapidly code, release, test, and integrate it as a proof of concept. They should also craft a roadmap of new functionality for successive releases.

### **LEVER FOUR: DEVOPS AUTOMATION**

Automation is a critical evolutionary step in a company's digital transformation. DevOps—a way to develop and deliver high-quality digital projects quickly and with minimal risk—is the practice of having development and operations engineers work together throughout the service life-cycle, from design through support. It involves the automation of the development, integration, testing, and deployment processes—processes that once involved scores of manual tasks—and makes it possible to conduct these processes continually (generally, at least once a day) as development progresses. Cycle times are accelerated by orders of magnitude: Google, for instance, can deploy new software to a production system many times a day with minimal risk, keeping costs dramatically low. By freeing up resources and reducing time spent on labor-intensive, low-skill tasks, DevOps fosters a fruitful, test-driven culture. (See “Leaner, Faster, and Better with DevOps,” BCG article, March 2017.)

DevOps allows automation technologies and services to provision and configure infrastructure rapidly for each environment, from development and testing to preproduction and production. Developers code, release, and fix problems nearly in real time without the worry of glitches cascading into other work streams.

DevOps automation calls for a strong owner—clearly defined early in the process—and a team with the resources, engineering expertise, and authority to act. Because it demands high levels of skill, DevOps can require bringing in new talent. Testers become quality engineers, and working with scrum teams, they code automated tests that track errors and collect data that al-

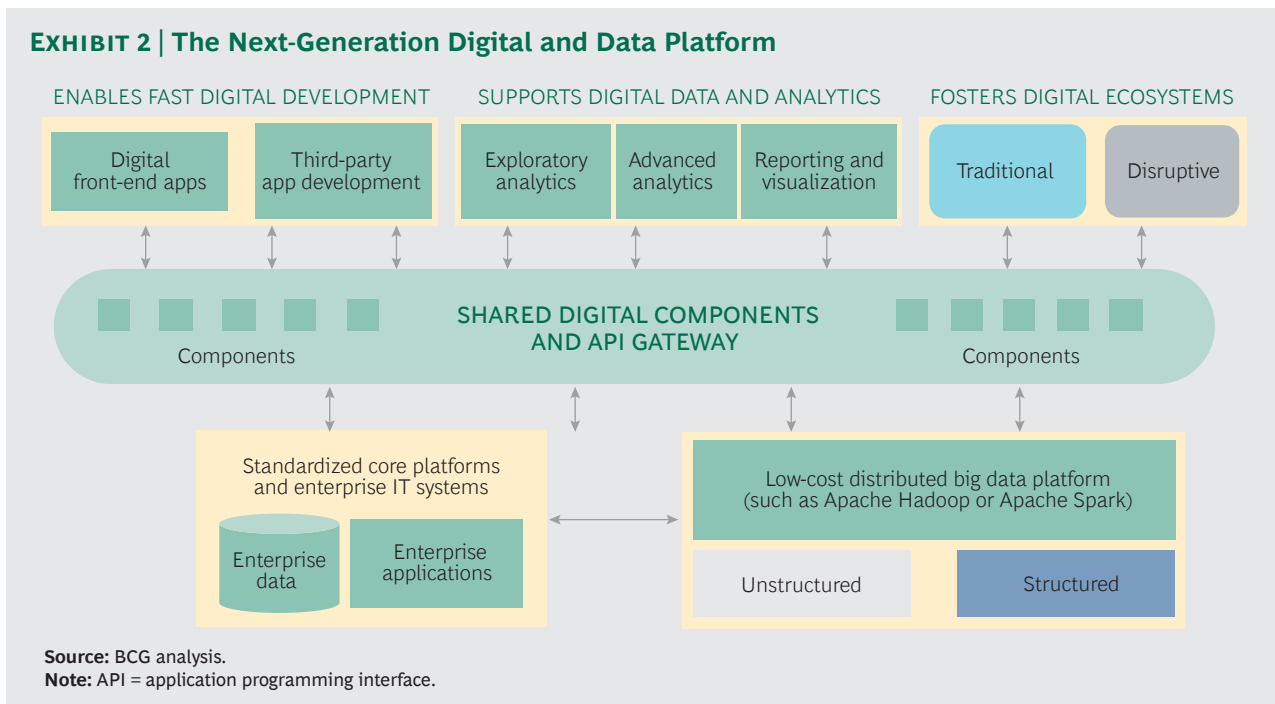
lows for fixing errors. Operators—traditionally the least skilled on the IT team—take on the role of reliability engineers, adding monitoring code and metrics to applications and conducting advanced analytics to spot and fix problems early. Operators not only improve an application’s long-term reliability but also reduce maintenance costs. (For more on how one company successfully applied the derisking levers, see the sidebar, “Meeting a Launch Deadline with a Hybrid Agile Approach.”)

**LEVER FIVE: A NEXT-GENERATION DIGITAL AND DATA PLATFORM**

Few today would doubt the necessity of a modern digital and data platform. But many overlook its overwhelming importance in taking the risks out of digital projects. A modern digital and data platform is a crucial lever because it facilitates development of multiple digital products—such as a bank balance inquiry tool, mortgage tracker, and retail store payment app—that can be created, tested, and launched in one place rather than through discrete, separate infrastructures. DevOps tools are embedded in the platform, so it is easy to begin developing new products. And companies can reuse components, thus minimizing error and vastly accelerating their overall digital transformation.

Exhibit 2 illustrates the following defining characteristics of a next-generation digital and data platform:

- **It enables rapid digital development.** A modern platform relies heavily on standardized APIs (Representational State Transfer, or RESTful, services) to separate digital product creation from the data and functions in traditional IT systems. Using such a system, companies can create many new microservice-based components—for example, customer identity management and promotional campaign management components—that can be reused across many different products and front-end devices. The standardization allows for the development of new components by third parties, thus expanding and accelerating innovation.
- **It supports digital data and analytics.** A platform stores (and makes accessible) many new forms of data, such as unstructured data and data from digital devices, in massive volumes. In effect, it serves as “data central,” capturing data from all channels and making it possible to perform analytics on it, cheaply and powerfully, through technologies such as Apache Hadoop.



## MEETING A LAUNCH DEADLINE WITH A HYBRID AGILE APPROACH

Risk Management Solutions (RMS), a company that models catastrophe-linked risk for insurers and the public sector, decided to transform its software application into a cloud-based platform. The plan was to give customers capabilities that they had been clamoring for: integrated risk modeling along with scalable exposure and loss analytics. Ultimately, the move would also spawn new possibilities for the company. In short, the new product, RMS(one), meant a transformation of the company's business.

According to CEO Hemant Shah, demonstrating RMS(one) to customers represented "a critical milestone." When company leaders decided to launch the new platform at the RMS annual customer conference, the event was only ten months in the future. Given the many dependencies and integration requirements, Shah and his deputies agreed on several measures that would lower the risks. This entailed planning all the upfront work leading to the conference date—essentially, adopting programmatic agile.

The first step was getting the modeling, data science, and software engineers to agree on the most important application features. Starting with several scenarios, the team hammered out the elements of a minimum viable product (MVP). Once comfortable with their technical choices, they were ready to begin development of an integrated prototype.

Next, RMS tackled the complex technical integration of its application components. The teams validated each of the individually chosen technologies and then built and tested the technical integration, connecting the company's original RiskLink software to the platform using application program interfaces (APIs). They next added the minimum

functionality needed to deliver a coherent product at the customer conference, prioritizing the development of the most critical features during the first seven months of the development plan and allowing for any contingency.

For team members who were used to working in silos—and pointing fingers when problems arose—collaboration was a completely new experience. Team leaders opted for a loosely coupled model execution framework that allowed them to harness the existing engines. Team members worked together on design and implementation, learning from one another and leveraging one another's distinct strengths.

In another critical move, RMS replaced manual testing with automated tests conducted in parallel with the MVP development. The company recruited new talent and, to build automated deployment, formed a DevOps team within the development group.

When it was time for the customer conference, RMS was ready to debut fully working software. Release 1.0 was in customers' hands shortly afterward. And because all the components worked together, it was possible to add new features later, sequentially, on the basis of the product roadmap and in response to customer feedback. RMS was also able to move quickly to monthly code releases. (See "How a Software Company Derisked Its Move to the Cloud," a BCG interview with RMS executives, February 2018.)



- **It fosters digital ecosystems.** A modern platform permits secure and consistent interaction with third-party systems. It can connect to and foster new digital ecosystems of customers, partners, suppliers, and third-party developers. It thus goes far beyond a set of applications and can foster collaboration at new levels, creating new sources of value and disrupting traditional value chains.

At the very least, the platform connects participants. Ultimately, however, it enables organizations to own a new ecosystem. Think: Apple, which owns the iPhone app ecosystem; Uber, which now offers takeout-food delivery; and Amazon, which not only sells but now also delivers groceries, thus changing consumer behavior and transforming the retail grocery landscape. By retaining control of its data, a company has the potential to create new offerings (for example, selling aggregate customer

data) while utilizing the data to continually improve and refine its existing offerings. Arguably, ecosystems are what digital is all about: companies working with an extended network to tap opportunity and create more value. (See “The Age of Digital Ecosystems: Thriving in a World of Big Data,” BCG article, July 2013.)

**I**F DIGITAL IS the new imperative, going digital in the right way is crucial. By employing the five levers outlined here—adopting programmatic agile when appropriate, creating strong business-IT teams, defining MVP features, investing in DevOps, and establishing a next-generation digital and data platform—companies can speed development while minimizing risk. Eliminating delays, rework, and cost overruns, they can meet commitments to their customers, maintain goodwill, and keep their sanity as they harness bigger and ever-bolder value-creating opportunities.

### About the Authors

**Jon Brock** is an associate director in the London office of The Boston Consulting Group. He leads the large-project derisking and recovery business of the Technology Advantage practice. You may contact him by email at [brock.jon@bcg.com](mailto:brock.jon@bcg.com).

**Rash Gandhi** is a principal in the firm’s London office and a member of BCG’s Technology Advantage practice. He focuses on the technology and architecture of big data. You may contact him by email at [gandhi.rash@bcg.com](mailto:gandhi.rash@bcg.com).

**Sesh Iyer** is a partner and managing director in BCG’s Washington office and a member of the Technology Advantage practice; the Technology, Media & Telecommunications practice; and the Energy practice. In the Americas, he is the leader of lean services and operations in technology and IT and cloud computing, and he works with the IT Capability Maturity Framework. You may contact him by email at [iyer.sesh@bcg.com](mailto:iyer.sesh@bcg.com).

The Boston Consulting Group (BCG) is a global management consulting firm and the world’s leading advisor on business strategy. We partner with clients from the private, public, and not-for-profit sectors in all regions to identify their highest-value opportunities, address their most critical challenges, and transform their enterprises. Our customized approach combines deep insight into the dynamics of companies and markets with close collaboration at all levels of the client organization. This ensures that our clients achieve sustainable competitive advantage, build more capable organizations, and secure lasting results. Founded in 1963, BCG is a private company with offices in more than 90 cities in 50 countries. For more information, please visit [bcg.com](http://bcg.com).

© The Boston Consulting Group, Inc. 2018. All rights reserved. 2/18