4

Rearchitecting the Firm

> **FROM:** Jeff Bezos
> **TO:** All Development
> **SUBJECT**: Bezos Mandate
>
> All teams will henceforth expose their data and functionality through service interfaces. Teams must communicate with each other through these interfaces.
>
> There will be no other form of inter-process communication allowed: no direct linking, no direct reads of another team's data store, no shared-memory model, no back-doors whatsoever. The only communication allowed is via service interface calls over the network.
>
> It doesn't matter what technology they use.
>
> All service interfaces, without exception, must be designed from the ground up to be externalizable. That is to say, the team must plan and design to be able to expose the interface to developers in the outside world. No exceptions.
>
> Anyone who doesn't do this will be fired. Thank you; have a nice day!
>
> Jeff Bezos

In 2002, when Amazon's CEO wrote this email, the online retailer had hit a wall.[1] The company was having trouble supporting its own growth. Its processes were breaking down as the software infrastructure powering Amazon's operations was cracking under pressure. Too much volume, too many products, too many different businesses—books, office supplies, electronics, apparel—were being sold on a network largely cobbled together through acquisitions and only weakly connected by a common home page. Without consistency in technology or data architecture, and without a consistent view of the customer, Amazon was coming apart at the seams.

The Bezos memo is one of the seminal documents in the digital transformation of business. In the preceding chapters, we highlight the birth and growth of a new breed of firm. The twenty-first-century firm is not only about leveraging the internet, or implementing mobile technology, or being a "digital native." Plenty of recently founded, software-intensive firms are built in the wrong way. Rather, it's about being architected differently, about being built on fundamentally different business and operating foundations.

Rather than rest on a traditional organizational model and operate through a variety of specialized and siloed organizational processes, digital firms rest on an integrated, highly modular digital foundation. Information technology is no longer merely an enabler and optimizer of traditional processes and methods; instead, software makes up the actual operating core of the firm. Replacing traditional labor- and asset-intensive organizations, fueled by a pipeline of data and powered by algorithms, software constitutes the critical path in delivering value to the firm's customers. And because of these digital foundations, the firm is capable of generating increasing returns to scale, scope, and learning—and of overwhelming traditional business models.

Even the most advanced AI factory in the world will not deliver the promised value if it is not embedded in an operating model that leverages its strengths. Bezos's intuition on this front was remarkable. In our language, he saw that the key to Amazon's sustainable growth lay in transforming its *operating architecture*, which defines the boundaries and linkages between the components of the operating model. He understood that a digital firm requires a different kind of operating model—one that is architected to take an integrated core of software, data, and AI and use it to power a new breed of organization.

To unpack the importance of the Bezos memo and its implications for the modern design of firms, we next take a bit of a detour into the history of operating models and their relationship to the architecture of organizations and technology.

Bezos and the Mirroring Hypothesis

One of the more intriguing fields of management study focuses on the relationship between the structure of an organization and the architecture of the technological systems that the organization works with. In short, the organization reflects the system, and the system reflects the organization. This simple observation carries important implications for the evolution of firms.

In 1967, a computer scientist named Melvin Conway noted that an organization is constrained to design systems that reflect the communication patterns prevalent in the organization.[2] Conway's law is based on the reasoning (supported by abundant empirical evidence) that for an integrated technology component to be designed properly, its designers must communicate frequently. Thus, it's now generally accepted that interrelated tasks are best performed by integrated teams, ideally located within a few feet of each other.[3] This is why software development projects are organized into agile feature teams rather than functional teams, and why manufacturing plants, and even financial and professional services, are organized into departments performing related tasks.

This framework is summarized as the *mirroring hypothesis*, which states that "organizational ties within a project, firm, or group of firms … will correspond to the technical patterns of dependency in the work being performed."[4] Well beyond design tasks, the architecture of systems mirrors the architecture of the organizations that depend on them.

These mutually reinforcing connections can become a significant asset for a firm, enhancing both quality and efficiency in the execution of work. As organizations perform similar tasks—say, designing and producing car door handles for various models and generations of vehicles—they develop productive ways of performing work. These techniques, embedded in technology, processes, and routines, enable organizations to build uniqueness and distinctiveness over time. After many years of dedicated practice, Toyota's production system (TPS) is embedded in the organization. Reinforced by incentive and performance measurement systems, these patterns serve to improve the performance of daily activities.

While making similar tasks increasingly efficient over time, these patterns can also constrain an organization, building inertia that hampers the response to change. Our Harvard colleagues Rebecca Henderson and Kim Clark argued in a 1990 paper that architectural innovations—ones that require changing the architecture between technological components—are a particular danger for established firms.[5] Their insights are relevant to many examples, including RCA's failure to rearchitect and miniaturize its tabletop radios and music devices even in the face of competition from Sony (which licensed RCA's technology!). Other examples include IBM's failure to transition from mainframe to PC, and Microsoft's failure to rearchitect PCs into smartphones. The concept of architectural inertia—the resistance to adaptation—in turn informs Clayton Christensen's disruption theory.[6] According to disruption theory, it is the architectural inertia established by the links with existing customers that prevents an organization from responding effectively to disruptive change.[7]

The bottom line in many of these perspectives and theories is similar: as organizations become good at doing something in a certain way, they develop routines and systems that reinforce each other and make it difficult to do things differently. Architectural inertia thus makes it difficult to achieve transformations that require organizing work in new ways.

Critically, architectural inertia is woven into the story of enterprise information technology over the past three or four decades. Enterprise IT has been largely deployed along traditional operating and organizational boundaries. We have general ledger systems, marketing "automation" software, customer relationship management software, product life cycle management, and enterprise resource planning, each fitting neatly into the established components of a traditional firm. Although improving efficiency on the margin, this componentization has limited the systemic impact of information technology and has constrained the scale, scope, and learning potential of the traditional firm.

In writing such a clear and provocative memo, Bezos was trying to break architectural inertia and change the architecture not only of Amazon's technology but also of Amazon's organization. Bezos was determined to transform Amazon's operating architecture and build the foundations for a software-, data-, and AI-driven firm.

Before exploring the new model, let's take a quick step back to understand the historical roots of operating models and see why traditional operating architectures look the way they do and are as entrenched as they are.

The Historical View

Long before we had information technology, firms evolved into siloed operating architectures consisting of specialized, largely autonomous functions and operating units. Dating back to at least the Italian Renaissance, operating models managed operational complexity by breaking an organization into smaller, separate units, each focused on an individual task and discipline.[8] Each unit was given a large degree of independence to maximize flexibility and minimize the load on the (excruciatingly slow) lines of communication.

One of the earliest known examples of distributed commercial operating architecture dates back to the fifteenth century. In Prato, Italy, the wool and textile trades distributed operations across many specialized production, distribution, banking, and insurance facilities.[9] This operating model functioned as a loosely connected ensemble of specialized organizations. In some cases, the relationships between organizations were established by family ties. In other cases, they were structured more formally, with joint ownership of assets among business partners, effectively creating holding companies with a multifunctional structure. These "primitive" organizations evolved a highly effective operating model and developed leading positions in Europe.

The First Firms

The first modern corporation may have been the Dutch East India Company, founded in 1602. From its inception as the consolidation of seven rival trading companies, the company achieved economies of scale by integrating various shipping portfolios and managing the considerable risk involved in individual voyages. But to manage its extensive operations, the company evolved into a multi-unit structure. By subdividing the organization into a number of specialized, geographically separate, and largely autonomous units, the company managed multinational, multidiscipline operations without drowning in communication delays and managerial complexity. Its siloed operating architecture and flexible managerial approaches worked well to accommodate the requirements of its dispersed geographic locations.

The company grew into an economic powerhouse, first monopolizing trade in spices like nutmeg, mace, and cloves from ports across Asia and Africa, and subsequently moving into silk, cotton, porcelain, and textiles. By 1670, the company was possibly the richest the world had ever seen, deploying almost two hundred ships and employing more than fifty thousand people (along with a relatively large private army) to make up a complex operating model that came to dominate global trade.[10]

Although trade and financial services continued to become more sophisticated through the seventeenth and eighteenth centuries, manufacturing processes did not evolve very much. Traditional craft methods, also known as "filing and fitting," relied on expert crafters working on one artifact at a time, producing all components and making each adjustment by hand—"filing" each part so that it would "fit" into the assembly.

The Rise of Mass Production

The Industrial Revolution transformed production techniques. From England to the United States, the emergence of mass production drove a wave of specialization and standardization. Unlike filing and fitting methods, mass production meant that each worker focused on a single component or a single stage of the production process. In this way, operating models could benefit from specialization and repetition to increase the advantage of scale and the speed of learning. This approach led to specialization within the organization by the nature or discipline of work, something that further subdivided the operating architecture of corporations.

The true icon of mass production and industrialization is found in the automotive industry—most of all, in the Ford Motor Company. Henry Ford founded the carmaker in 1903 in Dearborn, Michigan, with $28,000 in cash from twelve investors. Ford's vision was to make automobile transportation practical, affordable, and accessible to the average person. Ford sensed an opportunity to design and produce a car that could be sold at a price that met the potentially vast demand of middle-class customers.

With its introduction in 1908, the Model T (affectionately called the Tin Lizzie) was explicitly designed for mass production. Efficient, durable, reliable, and easily maintained, it was generally considered the first automobile within reach of most American consumers. Overwhelmed by the demand for its new car, Ford had to find a new way to deliver value.

Ford introduced the first moving chassis assembly line at the Highland Park Plant in 1913 and transformed manufacturing. Traditionally, cars had been assembled in fixed stalls, with workers coming to each vehicle to deliver and mount the required components. On the assembly line, vehicles would move through a series of stationary workers, with each worker performing highly specialized, increasingly narrow assembly tasks. With the help of the legendary Frederick Taylor, the Ford assembly line cut the Model T assembly time by a factor of 10, a change that in turn dramatically reduced costs. Prices came down by more than a factor of 2, and by 1918 half of all cars in America were Model Ts.

Ford had become the largest manufacturer in the United States by deploying unprecedented levels of standardization and specialization. Its operating model broke down functional specialties and associated organizational silos to the smallest, most specialized, standardized human tasks.

Twentieth-Century Operating Models

Ford's operating model led the automotive industry for decades. Over time, General Motors started to win share from Ford by offering a much broader range of cars at a broader range of prices. To increase the scope of offerings delivered by its operating architecture, General Motors created dedicated organizational units—among them Chevrolet, Buick, GMC, and Cadillac—each focused on a different product line with its own specialized assembly lines. These largely autonomous product units enabled GM to focus on the specific needs of different customer segments.[11] Now organizational silos were broken down not only by narrowly defined function but also by product.

The GM model reigned supreme through the 1950s and 1960s, until a new generation of competitors, many from Japan, introduced more efficient and higher-quality cars. Their success emerged from additional refinements in the design of operating models and operating architecture. The Toyota TPS operating model added a dedication to learning and problem solving at all levels of the organization. Toyota's model pushed back on traditional narrow specialization common in the industry, but it was notoriously difficult for others to imitate it and deploy it successfully. This remained true even when Toyota fully opened its factory floors to outsiders, wrote many books about the process, and undertook joint ventures with other auto companies.

Beyond the automotive sector, mass production took off rapidly in the twentieth century in most other manufacturing industries in the United States and Europe. As workers and organizations specialized, and as production generated more output, manufacturing operating models enjoyed increasing economies of scale, with efficiencies (and quality, as specialization improved the work) greatly increasing with the volume of the operation. Additionally, the production volumes enabled learning, further increasing production efficiencies. These economies virtually wiped out traditional crafts in a broad range of manufacturing and service industries, from weapons to textiles and from agriculture to insurance.

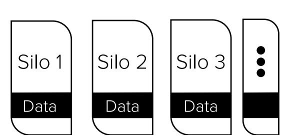
Over time, mass production concepts like specialization, focus, and standardization also spread widely across the service industries. Notably, the growth of supermarkets relied on significant process standardization as well as economies of scale in purchasing and delivery, and fast food franchises like McDonald's relied on highly repeatable work and scale efficiencies in both supply chain and food preparation. Specialization and standardization led to efficiencies in hotel chains and banks, energy companies and insurance providers, hospitals and airlines.

Highly specialized, siloed operating models remain essential in manufacturing and service delivery today. Take the iPhone, assembled in China by Foxconn Technology Group. Foxconn's facilities in Zhengzhou cover 2.2 square miles and can employ up to 350,000 workers, whose work is narrowly specialized, meticulously specified, and highly optimized. There are ninety-four production lines, and it takes about four hundred steps to assemble the iPhone, including polishing, soldering, drilling, and fitting screws. The facility can produce more than 500,000 iPhones a day, or roughly 350 a minute. Although modern manufacturing lines like these are enabled by information technology—tracking parts and products, analyzing problems, or enabling robotic assembly—modern operating models still drive scale by designing standard, repeatable work in both product and process development.

We emphasize, again, that the deployment of enterprise IT did not transform the trajectory of operating models. During several waves of adoption—from the mainframes of the 1960s and 1970s, to the client-server models that took off in the 1980s, to early internet-based systems deployed in the 1990s—IT systems such as Oracle financials and SAP product life cycle management improved the performance of many traditional operating processes; but these IT systems generally mirrored the firm's siloed and specialized architecture. Although often improving efficiency and responsiveness and driving additional economies of scale, scope, and learning across operating units, technology did not change the structure of the enterprise.

FIGURE 4-1

Siloed architecture



In company after company, processes, software applications, and data are still embedded in individual, largely autonomous and siloed organizational units (as sketched in figure 4-1). As we look across most major enterprises, we see that IT—and, most critically, data—are most often gathered in a distributed and inconsistent fashion, separated and isolated by existing organizational subdivisions and by generations of highly specialized and often incompatible legacy technology. Large firms often use thousands of enterprise applications and IT systems, working with a variety of scattered databases and supporting diverse data models and structures. Integrating data across different functional silos (without rearchitecting the entire system) is a long, horrifically complicated, unreliable process, requiring significant dedicated investment and extensive custom code. It's no wonder that many such projects are plagued by painful delays and cost overruns.

Traditional Operating Limits

From the East India Company to GM to McDonald's, operating models reinforced autonomy and specialization and led to new levels of productivity and innovation. In each case, there is evidence of great success. However, there is also clear evidence of limits, as the complexity of expanding operations eventually outpaced the capacity of every organization and opened opportunities for competition. Traditional operating architecture created serious constraints to firm growth and value. Ford's mass production methods ran into problems when confronted by General Motors' product variety and differentiation and Toyota's process improvement and quality mindset. Even the Toyota production system had a hard time handling rapid growth and increasing complexity, as demonstrated by its many product recalls during the mid-2000s.[12] Ultimately, as traditional organizations grow, they suffer *dis*economies of scale, scope, and learning.

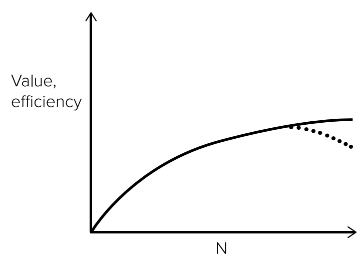
When organizations expand, they become increasingly complex and difficult to manage, so they build bureaucracies and inefficiencies, and they embed norms, incentives, and rewards—and each of these fosters inertia. With too much scale, too much scope (variety), or too much demand for learning and innovation, any managerial process will eventually stop working well, leading to inefficiency and even failure.

Plants reach an optimal size and then become unwieldy to organize and manage. Restaurants reach a maximum size and scope, as their customers and menus begin to overwhelm the staff's capabilities and systems. Even R&D organizations and product development teams can grow too big, and their productivity and innovativeness are known to suffer as a result. These considerations shape the maximum efficient scale of an organization and impose overall limits to its growth.

Notably, traditional information technology has not significantly loosened these constraints. As a traditional enterprise creates ever more functional silos, it deploys a myriad of IT systems, from CRM to general ledger software, each meeting the demands of the specific function to which it is assigned. Integrating and aggregating various applications and connecting potentially valuable data is a long and painful endeavor, as disparate disconnected legacy systems need to be carefully pieced together through custom software, which, over time, will itself cause its own inertia and resistance to change.

FIGURE 4-2

A traditional organization's ability to deliver value faces a curve of diminishing returns



N is a parameter that stands for a variety of variables, such as the number of users or the number of complementors on a platform.

In a nutshell, firms are shaped and limited by their operating models. These models help manage complexity and growth—but only up to a point. Traditional functional structures and operating silos have also caused firms to hit limits and diminishing returns to scale, scope, and learning. Despite several generations of widespread improvements in management and operations, and even despite the extensive deployment of enterprise IT, the complexity of operating models has constrained the value that can be delivered by the traditional firm, as depicted in figure 4-2.

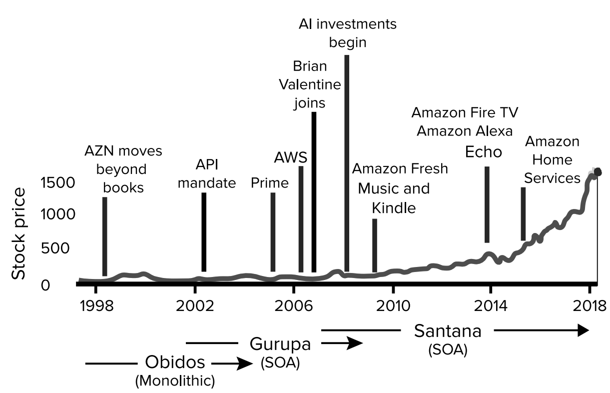
A Critical, Difficult Transition

Before Bezos wrote his memo, Amazon was starting to look like a traditional firm. Its organization, data, and technology had evolved into silos, with disparate retail focus areas largely contained in separate, disconnected units. Connections between silos were haphazard and often unpredictable, motivated by meeting immediate needs and fighting fires. Amazon was dealing with limits on the scalability and scope of its business. It needed a major architectural change.

Bezos knew well that in the software business, working with multiple versions of the same code is a nightmare. Also, scattering data across systems and functions prevents aggregation, destroys the integrity of any data pipeline, and hampers the development of a comprehensive view of the customer. His brilliant insight was that while supporting traditional operating tasks (e.g., supply chain, retail operations) Amazon could rearchitect those tasks, starting with the software. His vision was to build the best software- and data-driven operating model in order to expand his retail operations to unprecedented levels of scale, scope, and learning. But he also realized that to scale a software- and data-driven organization, he had to break organizational and technological silos. Figure 4-3 traces the progress of the transformation.

FIGURE 4-3

An Amazon time line



The curve depicts the Amazon stock price. Obidos, Gurupa, and Santana are the systems Amazon built to enable its operating capability and to meet its scale, scope, and learning objectives.

Bezos sought to rearchitect Amazon's technology and organization at the same time. Recognizing that software capabilities were now sophisticated enough to run significant parts of Amazon's operating model, Bezos rebuilt Amazon's retail operation on top of a software platform, which gradually evolved to embed a state-of-the-art AI factory. The organization was simultaneously transformed according to the new architectural boundaries, with emphasis on the broad deployment of agile teams working within clearly established interfaces.

Starting in the early 2000s, Amazon's transformation produced as many challenges as successes. When the first platform redesign did not meet expectations, the company brought in Brian Valentine, then a software executive at Microsoft. Valentine brought deep platform experience, having overseen successful releases of Microsoft Exchange, Windows 2000, and Windows XP. It is significant that a software platform leader—not a traditional IT professional—was charged with rebuilding Amazon's IT infrastructure. The goal was to go from siloed, disconnected IT into a true software and data platform, a common set of building blocks that could be deployed to drive scale and scope economies across Amazon's rapidly expanding roster of businesses.

The third version of the Amazon platform was code-named Santana, and even though it took a long time to complete, it propelled the firm to its current leadership position. Valentine created a real software platform, with a central, standardized set of services and clear APIs for interacting with those services. This shift required Amazon to rewrite virtually all of its e-commerce services, and the new platform, while vastly superior, took longer to build and implement than originally expected.[13]

With the redesign of its retail platform, Amazon's development organization evolved into a modular, distributed structure. Sharing a common foundation in the Santana technology, "two-pizza" agile teams (to cut down on pointless meetings, Bezos decreed that they never be so big that two pizzas wouldn't feed the entire group) can work independently while respecting clear architectural rules that enable teams to share common code and aggregate data across applications. The Amazon structure thus preserves common foundations and, crucially, aggregates the data that fuels machine learning and artificial intelligence, all while preserving the agility of small teams.

Santana enabled Bezos to get to the next stage and rapidly build data pipelines and a slew of world-class AI applications. From its recommendation engine to Amazon Echo and Alexa, the company has become a powerhouse in deploying AI across its enterprise. Although Amazon never led the pack in basic AI research (Google and Microsoft were ahead), the company has become expert in deploying the latest advances across all aspects of its business and deriving enormous operational impact.

Amazon's not-so-secret secret weapon on the AI front is its cloud services division, Amazon Web Services (AWS). Serving a million-plus customers, AWS has a mission to democratize access to information services, including compute, storage, and database, and the AI toolkit is heading in the same direction. In 2015, AWS began offering Amazon machine learning to its customers and quickly used innovations from Alexa to offer voice recognition, text-to-speech services, and a natural language processing interface.

In no time, its customers—large organizations like NASA and Pinterest, along with multitudes of startups—started to deploy AI tools on their own problems and made advances across the board. The company is now offering SageMaker, a software toolkit that enables its customers to go from data to insights by using prepackaged Amazon-developed systems, algorithms, and tools. The scope of the AI reinvention is so broad that Amazon's own internal machine learning conference has gone from hundreds of attendees to many thousands and is on track to become the largest internal company event.

The Amazon transition in operating architecture was among the first in a much broader trend across the economy. From Ant Financial to Google, a generation of AI-driven firms is being designed with this kind of operating model, driving scale, scope, and learning by aggregating software, data, and analytics and driving agile teams to focus on specific applications across the organization. These operating models depart radically from hundreds of years of corporate evolution and exhibit a profoundly different architecture, posing an existential threat to traditional firms.

Architecture for an AI-Powered Firm

How do you construct an organization founded on code instead of human labor? We must first remember that unlike humans, a digital system (let's call it a digital "agent") can communicate at zero marginal cost with a virtually infinite number of other digital agents performing similar tasks, anywhere in the world. Moreover, the same digital agent can be easily connected to the complementary activities of many other agents, providing a huge number of potential combinations. Finally, digital agents can embed processing instructions—algorithms that not only can execute logic but also might learn and improve themselves—as they process data.

Digital agents may not (yet) be as smart or as creative as humans. However, unlike humans, digital agents have no need of autonomy or isolation to reduce perceived complexity or scale, or to limit the variety of interactions. As long as digital systems use a well-designed, common interface they can connect and combine capabilities, dramatically enriching the range of possibilities.

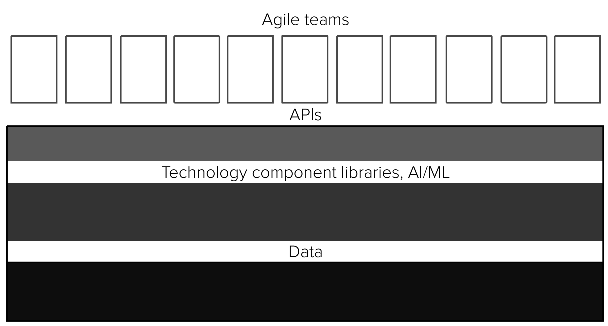
We are not talking about a few connections but potentially a limitless set. Think of the World Wide Web, connecting untold numbers of websites through an extremely flexible and general set of networks and interfaces. Many of the websites interact frequently with each other, in ways their original designers never dreamed of. Similarly, the iOS and Android platforms connect millions of disparate apps and services, from health and fitness to financial services. The aggregate functionality they deliver is virtually boundless. Digital operating architectures thus have little need of isolated functional silos or hard separations between individual subunits. They instead benefit from unlimited connectivity and data aggregation, driving increasingly powerful analytics.

With a digital operating model, the organization should be designed to unleash the potential of the digital technology it is built on, as shown in figure 4-4. This means creating a foundation (or platform) encompassing data and technology, a platform that can be deployed easily and rapidly to create or connect to new digital agents in the form of applications that address any of a broad variety of use cases.

The ideal is to have a common foundation of data inputs, software technology, and algorithms, all provided by an AI factory, as described in chapter 3. This foundation provides easily accessible (but carefully designed and secure) interfaces that teams developing individual applications can use. The applications connect the foundation to enable operating tasks, from customer relationship management to supply chains. The process used to develop these applications is driven by small, agile teams equipped with data science, engineering, and product management capabilities. Agile processes and digital operating architectures go hand in hand.

FIGURE 4-4

Operating architecture for an AI-powered firm



Modern operating models are also characterized by a relentless focus on improving performance through learning. Although some of that learning happens in real time—for example, as data fine-tunes algorithms for suggestions and pricing—much learning also happens on dedicated experimentation platforms, as described in chapter 3. Every day, employees might engage in hundreds, or even thousands, of A/B tests or randomized controlled trials to understand how various tweaks to the service prompt action by consumers, increase satisfaction, and ultimately lead to more revenue. While the data is centralized, the company's experimentation capability is highly decentralized; almost anyone with a hypothesis can launch a live experiment and use the results to implement meaningful changes.

Finally, digital operating models should promote modularity and reuse of the software and algorithms that are developed to perform various operating tasks. This requires adopting consistent frameworks for building functionality, such as React for user interfaces or Apache Storm for data processing. Interestingly, much of the software can be drawn from (and contributed to) the open domain, because the competitive advantage will move to the data that is accumulated by the firm. With this new breed of firm, we go from a focus on proprietary technologies and software to an emphasis on shared development and open source.

Breaking Traditional Constraints

In a digital operating model, the employees do not deliver the product or service; instead, they design and oversee a software-automated, algorithm-driven digital "organization" that actually delivers the goods. This transforms the growth process by removing the traditional operating bottlenecks constraining the scale, scope, and learning potential of a firm.

Removing human interaction from the critical path has a crucial impact on the operating model. The marginal cost of serving an additional user by digital agents becomes negligible, transforming the process of increasing capacity and making it much easier to scale. Furthermore, much of the operational complexity is solved through software and analytics or is outsourced to the external nodes of the firm's operating network. Algorithm-driven operating models are thus almost infinitely scalable, as long as you can continue to add computing and storage capacity to the technology infrastructure (which is now predominantly cloud based and available on demand) and add data to the AI factory pipeline.

Digital technologies are also intrinsically modular and can easily enable many more business connections. When fully digitized, a process can readily plug in to an external network of partners and providers or even into external communities of individuals to provide additional, complementary value. Digitized processes are thus intrinsically multisided and can greatly increase the scope of the operation. After value is delivered in one domain (e.g., accumulating data about a set of consumers), that same process can be connected to drive value in other applications, adding a multiplicative factor to the number of services and overall value it's delivering to the customer. This is how Ant Financial and Amazon work.

The value created by a digital operating model can also grow rapidly as learning effects lead to increasing returns to scale. This is where analytics and AI can shine. AI and ML thrive on data, and as machine learning models have evolved, the amount of data they can learn from has increased quickly. As they accumulate data by increasing scale (or even scope), the algorithms get better and the business creates greater value, something that enables more usage and thus the generation of even more data. The impact of machine learning on digitally enabled businesses, such as Amazon Echo or the Facebook ad network, effectively turbocharges the way a business delivers value to its users.

Finally, this new breed of organization transforms the role of management. Management as supervision, especially of employees performing routine tasks, is finally over. In an AI-powered operating model, managers are designers, shaping, improving and (hopefully) controlling the digital systems that sense customer needs and respond by delivering value. Managers are innovators, as they envision how these digital systems will need to evolve over time. Managers are integrators, as they work to connect disparate digital systems and identify new connections between the firm's operating model and the customers it serves. And managers are guardians, as they work to preserve the quality, reliability, security, and responsibility of the digital systems they control. Digital, AI-centered operating models challenge virtually every traditional managerial and operational assumption, forcing us to fundamentally rethink the nature of firms and of their management teams, their ability to grow, and the constraints on their impact and power.

But despite the massive business potential of the data-centric operating architectures driving AI-powered firms, many traditional firms hesitate. Their impulse is to protect capabilities, routines, and organizational boundaries, sometimes built over decades. They either do not see their architectural problem or are not willing to fully commit to the organizational transformation that is required to solve it. Frankly, the technology is the easy part. As many others have noted, organizational change is really hard.

The next chapter examines what it takes to become an AI company.