FRANKLIN TEMPLETON

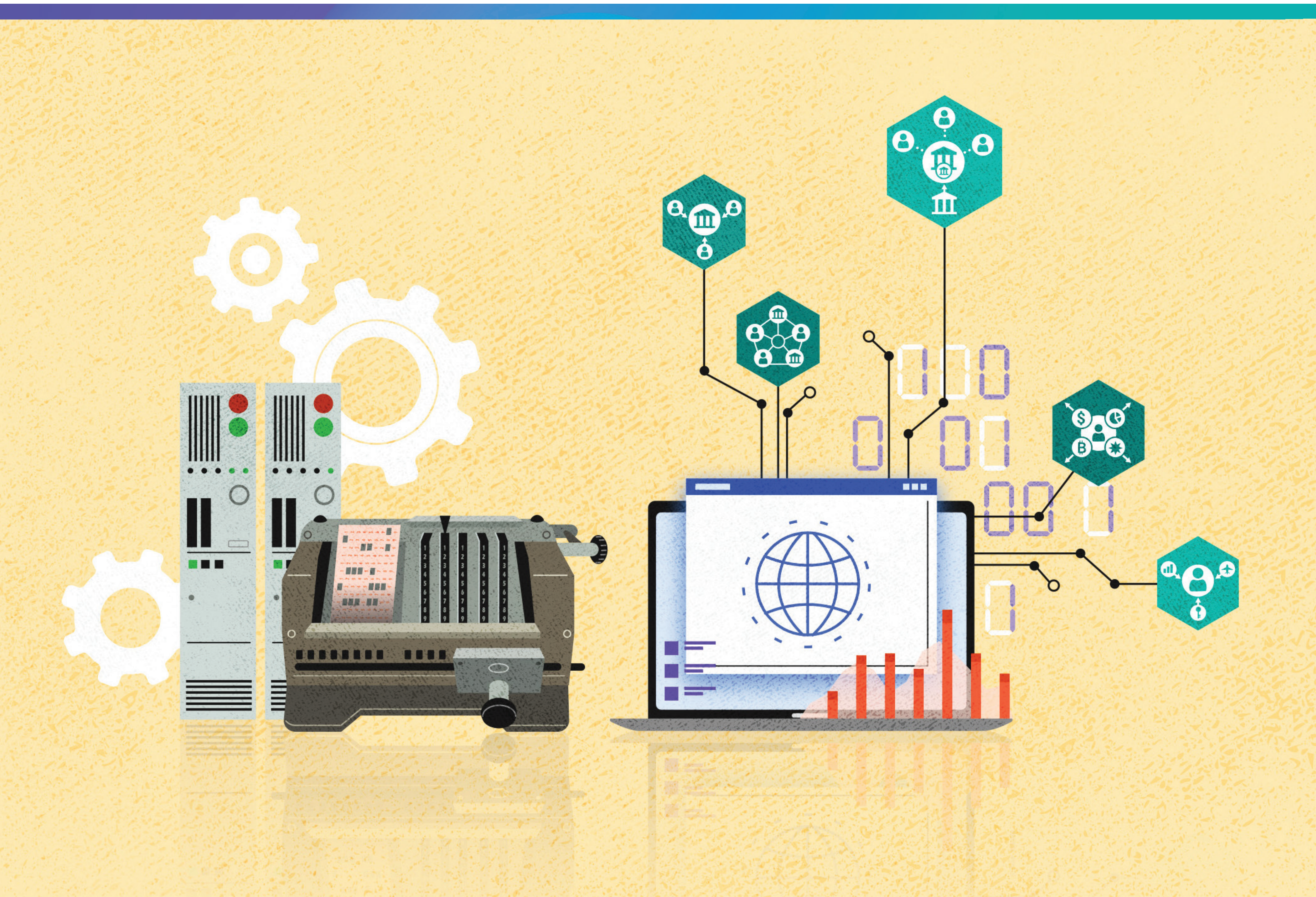
Institute | Technology disruption

Franklin Templeton Thinks™

September 2022

Evolution of commercial technologies and impact on business delivery

Part I of five technology-driven megatrends impacting societies and what to expect with Web3



About Franklin Templeton Institute

Our mission is to provide our clients with research that meets their needs and concerns. We do this by listening, understanding, and then harnessing the resources of our firm to answer the challenge. We organize around areas of exploration to develop distinct insights and their practical applications.

Introduction

Sandy Kaul
Head of Digital Asset and Industry Advisory Services
Franklin Templeton

The foundational technologies used to build the worlds of business, government and academia have already undergone three distinct periods of evolution since World War II, each of which has resulted in a re-architecting of the commercial landscape. A fourth cycle is now emerging. Each time the underlying technologies have changed, a new set of capabilities has been enabled and a resulting shift in behaviors has occurred.

Such shifts in behaviors are building upon each other, becoming amplified as new technologies replace obsolete versions. Viewed across several decades, certain behavioral changes are emerging as megatrends. These are not fads that can have a temporary influence on the trajectory of a society and then fade. Instead, these are deep-rooted transformations that are unfolding around us, reshaping our lives, and redefining how we act and operate.

In many ways, the history over the past eight decades has been one of rapid innovation and disruptive change. What was once deemed incomprehensible in terms of behavioral shifts has quickly become common place and normalized. Acknowledging just how much societies have already been impacted is critical because the upcoming cycle of technological change may be the most disruptive yet. The megatrends that have been building are likely to see their ultimate expressions in societal changes that are soon to be enabled.

This paper is Part I of a three-part series that will lay out the emergence and progression of these megatrends. This work seeks to build a layperson's understanding of the technology cycles that are driving these megatrends. The extent of innovation that has occurred since computing technologies began to move from the military and academic realm into the commercial domain has been impressive and iterative. Each epoch of technology advancement has shifted the way in which businesses operate and required enterprises to build and re-build their core infrastructure to keep up with the pace of change.

Understanding how these technologies and architectures have advanced is foundational to tracing the behaviors that give rise to the megatrends and tracking how each successive cycle of innovation helped to amplify their impact, transforming how people work, engage, communicate and pursue entertainment.

Key findings

Many technologies have contributed to the innovations that have transformed societies since World War II, but perhaps none has been as impactful for commercial enterprises as the emergence of the computer and its related infrastructure, hardware, software, networks and peripherals.

The processes required to run and deliver business functions have been encoded into software programs and systems; the work documents required to operate a commercial enterprise have been transformed into electronic records that reside on computer databases; the connections that enterprises rely upon to facilitate communications, travel, manufacturing and distribution have been grafted onto a vast array of computer-driven networks; the data and analytics that provide crucial intelligence about the health of a business and the strength of its customer relationships have been enhanced by complex algorithms that rely upon computer processing power and speed; the logic and predictions required to drive personalization engines and run sensors inside the machines that govern our daily lives have been enabled by artificial intelligence (AI) tools that learn through massive data ingestion carried out across huge arrays of networked computers; and the cryptographic protections required for us to share our personal and business information require high-level mathematical calculations made possible by lightning-fast computation.

Since World War II, we have progressed from societies that use machines as tools to societies that forge "human+machine" relationships that are transforming how we work, live and entertain ourselves. The pace of innovation has been relentless and shows no signs of slowing.

Yet, these capabilities have not come about all at once, and the impacts that they have had on how we design and deliver commerce have been iterative. Indeed, it is possible to identify a cycle that governs how these dynamics unfold: 1) New innovations occur and result in upgraded technology, 2) which in turn requires enterprises to rethink their business practices and 3) rebuild the infrastructure they utilize to deliver their goods and services. In this paper, we lay out three distinct periods in which this cycle has already repeated itself, and we make the case that a fourth cycle is emerging.

> **"Each cycle has had profound impacts on the way that a business works and on the relationship between a commercial enterprise and the individuals that make up its employee and customer base."**

Each cycle has had profound impacts on the way that a business works and on the relationship between a commercial enterprise and the individuals that make up its employee and customer base. Understanding how significantly these processes and relationships have changed in each period of technological progression already completed or underway is crucial to appreciate how disruptive the next set of changes may be and to anticipate what may soon lie ahead.

First cycle of commercial technology—automation

The first cycle of commercial technology emerged in the 1960s. Innovations in the way that computers functioned—processed information, stored data and programs, and enabled data retrieval—allowed for the size of computers to become more compact (no longer requiring vast air-conditioned facilities)—and for the way that computers operated to become more efficient (using hard-coded software programs instead of relying on dedicated teams of punch card operators to perform business functions).

Whereas only governments, the largest universities and manufacturing giants such as General Electric could afford computers and the teams required to run them in earlier years, the new miniaturized, chip-built computers became affordable and accessible to a far broader array of commercial enterprises. New peripherals—items that could plug into the computer to extend its functionality, such as work terminals and printers—became useful business tools as the outputs they were able to produce moved beyond huge computer printouts to more recognizable documents. Word-processing solutions that stored their outputs on magnetic tapes allowed for far more data to be captured and made retrievable electronically. These outputs were stored in hierarchical databases that used file-based systems to associate and organize information. By the end of the

> **“Business practices changed as automation occurred. Organizations described what they were experiencing as a ‘knowledge explosion.’ The time required to complete business tasks fell sharply, productivity soared, and the amounts of data being collected grew exponentially. However, there was almost immediately an emerging concern about the societal impacts of the new technologies and how they might replace the need for human workers.”**

period, microcomputers that were small enough to fit into individual offices were emerging. These devices could be networked via hard-wired solutions to share outputs across a set of local users.

Business practices changed as automation occurred. Organizations described what they were experiencing as a “knowledge explosion.” The time required to complete business tasks fell sharply, productivity soared, and the amounts of data being collected grew exponentially. However, there was almost immediately an emerging concern about the societal impacts of the new technologies and how they might replace the need for human workers.

A trend toward decentralization of enterprises that had been unfolding over prior years reversed. Data and work outputs were no longer dispersed across a robust network of powerful branch offices. Businesses centralized their data collection and processing efforts in the hub that housed their mainframe. This allowed them to reduce or redeploy a significant percent of their workforce. Those that controlled the access and dissemination of information at the central hub gained organizational power. Enterprises hired and built information technology (IT) teams to support these business leaders, further centralizing the resources of the organization.

Ambitious managers no longer sought out jobs in more remote locations, instead wanting to be as close as possible to the central hub and data to further their career options.

A monolithic architectural approach dominated in these early years of computing. The infrastructure, hardware, software, network technologies, databases and peripherals required to run the enterprise were typically supplied by a single provider that had its own service and support teams. Internal technology teams facilitated the set-up of individual users, the maintenance of the networks and the execution of data inquiries.

Second cycle of commercial technology—digitization

The second cycle of commercial technology innovation began in the late 1970s as microcomputers became increasingly compact, morphing into personal computers (PCs). What started as an offering for hobbyists quickly began to grow as the smaller computers were able to run programs on portable software and, later, use a mouse to give x-y directional commands that could interact with icons on a screen to launch and control programs using a graphical user interface (GUI)—thus removing the need for a user to utilize program commands.

The line between business and personal usage of computers began to blur with the entry of IBM into the personal-computing space in the early 1980s. To speed time to market and reduce the cost of its PCs to be competitive, IBM veered away from their prior monolithic approach and used an open-sourcing model to construct their offering, contracting with several suppliers for various components, software and peripherals, and making its machines open architecture so that other providers could begin to build and launch IBM-compatible products. The marketplace exploded with offerings from multiple manufacturers, creating more competition and allowing costs to come down so that smaller as well as larger enterprises could begin to consider automating their white-collar worker functions.

Commercial off-the-shelf technology systems emerged, focused on providing discrete functions such as content management, enterprise resource management and accounting. These offerings made the deployment of new capabilities simpler. An evolution in the design of storage technology shifted the standard from hierarchical to relational databases and made it easier for information to be extracted and distributed to business users without the need for centralized IT teams to run queries. The growing expansion of commercial, off-the-shelf software gave business users the tools to enter data, create spreadsheets, run macros, author, and save documents and presentations—thus allowing decades of paper files to be digitized and starting a new epoch of office automation. The creation of intellectual property inside organizations surged.

Proprietary systems development also emerged as a necessity. The variety of commercial off-the-shelf products being brought into the organization made it difficult to deliver complex, multistep business processes that utilized discrete functions and data from several

> **“Though computer networking had been possible through local area networks and organizations had tied into wide-area networks that allowed data to be transmitted over telecommunication lines, the introduction of the internet and the World Wide Web significantly changed the way that businesses operated.”**

underlying applications. Enterprises began to build their own user interfaces, business logic and data mappings to tie together these various offerings. All of these increased the complexity of the organization’s technology infrastructure.

A new practice called enterprise architecture emerged to better align business needs and technology delivery. Early architectures followed a client-server approach—separating out the interface and user commands (the presentation layer) from the back-end business logic and data that delivered the required functionality and was stored on the organization’s processing backbone. These architectures began as two-tier—client and server—but moved quickly to three-tier as a new type of facilitative technology called middleware was integrated, making it easier to knit together multiple systems. The emergence of web technologies changed this approach further, adding a fourth or n-tier as web browsers and interfaces had to be integrated.

Though computer networking had been possible through local area networks and organizations had tied into wide-area networks that allowed data to be transmitted over telecommunication lines, the introduction of the internet and the World Wide Web significantly changed the way that businesses operated. Public domain internet protocol suites (TCP/IPs) were created and began to be publicly disseminated in 1989 by the University of California, Berkeley. The growth in the number of website offerings exploded by 2000. New web-based businesses began to emerge, many of which sought to disrupt traditional business models and give consumers direct access to transactional opportunities via online channels. Soaring valuations for these companies pressured traditional business participants to speed their own launch onto the web.

This created massive pressures on the organization’s technology as already complex enterprise architectures had to be rebuilt to expose certain functions to website offerings, and lapses in system response time became a brand issue as clients could now access the organization 24/7/365 (24 hours a day, 7 days a week, 365 days a year). A new approach called a service-oriented architecture (SOA) was developed that broke down system functionality into a set of services that combined the business logic and data required to fulfill that discrete task and “containerized” them, creating an index of such offerings and making them callable through a new messaging layer called an enterprise service bus.

Third cycle of commercial technology—virtualization

The third cycle of commercial technology innovation began in the early 2000s. Three computing trends that began in earlier years came together at the turn of the century to help lay the foundation for distributed processing and cloud computing.

Time-sharing began as a way for multiple programs to run simultaneously on large mainframes to improve the efficiency of their operation, leading to a vision of computation being shared, and offered as a utility like telephony and electricity. Technologies emerged that allowed computers to create computer workstations and virtual terminals, allowing multiple distinct computing environments—each with its own operating system—to exist within one physical environment. This allowed for scaling the number of users that could utilize a common architecture. Finally, networking technologies matured, first allowing for the transfer of data packets over telecommunication lines, then incorporating TCP/IPs, and later allowing for the development of application processing interfaces (APIs).

By the early 2000s, Salesforce.com became the first commercial enterprise to offer its software as a service (SaaS) rather than requiring customers to purchase a commercial off-the-shelf system. Google had demonstrated the power of distributed processing, using vast arrays of low-cost computer servers networked together to increase their computational abilities. Amazon began to provide merchants a development toolkit that allowed them to plug into Amazon’s shopping carts and fulfillment infrastructure and, later, realized a broader vision of providing the “operating system for the internet” through Amazon Web Services (AWS), which launched in 2006.

For the first time, a new shared infrastructure was enabled that allowed software to be developed or accessed on a pay-as-you-go basis, and hosting of an organization’s data, networks and processing capacity to be rented on an as-needed basis.

A new architectural approach emerged called microservices. Although microservice architectures had similarities to

For the first time, a new shared infrastructure was enabled that allowed software to be developed or accessed on a pay-as-you-go basis, and hosting of an organization's data, networks and processing capacity to be rented on an as-needed basis.

service-oriented architectures, there were some important differences that made interoperating with shared development platforms more economical and effective—namely, adding the data required to run a service into the same container as the service rather than having the service call the data from a central location, and using APIs to have services communicate with each other rather than using an enterprise-messaging bus to send point-to-point instructions.

New types of businesses could be built upon these cloud-based architectures as even start-up firms could access the development resources, bandwidth, scalability and reliability that were previously only available to the most successful enterprises that had built their own proprietary infrastructures. Simultaneous advancements in the scripting languages used for web design and the development of new streaming capabilities allowed for a massive upgrade in the delivery of online services.

Whereas Web1 was characterized by static websites that relied on hierarchical navigation and page links as well as plug-ins to run video or audio content, new Web2 platforms were dynamic offerings that could read and write, built around user-centric views and engagement where content could stream directly. Individuals could contribute their own content, define their own metadata tags to classify their content, and form their own networks. Advancements in mobile technologies—particularly the launch of Apple's iPhone in 2007—freed users from the confines of their desktop computers to engage with these Web2 platforms anytime, anywhere.

The result of these advancements was that individuals began to embed the way they live into these tech-driven networks they use every day for a growing number of personal and business tasks and engagements. In tandem, the machines that societies use to operate—from wearable technology to home appliances, traffic, shipping and satellite communications—are also being enabled with "smart" technology that allow them to communicate their status and operational data via a growing array of sensors delivered via the internet of things (IoT).

Advances in AI are enabling this machine-to-machine communication as well as allowing for the creation of algorithms to make Web2 platforms stickier and more compelling through personalization and behavioral profiling. Key to the creation of AI tools has been the ability to ingest and process massive amounts of data to provide sufficient training sets for computers to apply deep learning techniques. The result has been a set of machine learning, natural language processing, predictive analytics and interactive voice advancements that are transforming the design and delivery of goods and services.

The framework that allows for massive amounts of data consumption and computation is often categorized under the broad term "big data." Based on blueprints shared by Google in 2003–2004, big-data processing reverses the approach to performing data analysis that had previously dictated the way that inquiries were run and data was utilized. In relational database management systems, the data is imported into the application to perform the analysis. This requires structured, tagged and mapped data stored in tables. In a big-data approach, the computational instructions are sent to the data and the analytic calculations performed in that distributed environment. This allows data inquiries to be run across structured, semi-structured and unstructured data.

Both big data and AI tools have become additional capabilities offered within cloud-based development platforms like AWS and its competitors that have emerged since 2006. Innovations being designed within these ecosystems continue apace with emerging offerings like self-driving cars, private spaceflights and the metaverse, demonstrating the potential that these platforms enable and preview new models yet to come.

Fourth cycle of commercial technology—decentralization

The fourth cycle of commercial-technology innovation is just emerging, and yet it already marks a significant departure from earlier advancements. The goal of decentralized technologies is to create a free-standing, peer-to-peer economy where the users of networks that enable commercial transactions are also the *owners* of those networks and can share in the financial rewards they

"Both big data and AI tools have become additional capabilities offered within cloud-based development platforms like AWS and its competitors that have emerged since 2006. Innovations being designed within these ecosystems continue apace with emerging offerings like self-driving cars, private spaceflights and the metaverse, demonstrating the potential that these platforms enable and preview new models yet to come."

generate. To accomplish this aim, decentralized technologies look to enable a new system of commerce, not just a new set of technologies.

The roots of this vision spring from several seminal works on cryptography that describe how to enable information to be encoded and transmitted over public networks. Much of the early work on cryptography was classified, but by the mid-1970s, the technology and mathematical algorithms to enable encryption were becoming better understood. In tandem—as the earlier commercial cycles discussed above were playing out—many involved in the cryptographic realm were becoming increasingly concerned about privacy and the ill effects that might result from allowing organizations to have exclusive access to huge amounts of data on individuals, their communications and their transactions.

A group of like-minded individuals with technology and cryptographic expertise came together in the late 1980s–early 1990s and dubbed their movement the "cypherpunks." The group put forward two published declarations—the *Crypto Anarchist Manifesto* and the *Cypherpunk Manifesto*—that laid out their goal of creating a new system that allowed individuals to operate pseudo-anonymously by relying on cryptographic protections, digital signatures and electronic money to enable any two individuals to directly transact with one another in a trustless manner without the need for intermediaries.

Several innovations designed to fulfill those goals emerged in the following years. Hashcash introduced the concept of "proof of work"—forcing one party to use a significant amount of computer time and resources to encode a set of content and create a cryptographic puzzle that another party would need to solve—using a lesser amount of computer time and resources to unlock the encoded content. A proposal for a digital cash payment system, called B-money, would create electronic coins and apply consensus mechanisms, such as proof-of-work, to get independent third parties to validate transactions. Reusable proof of work (RPOW) was a third innovation that showed how tracking digital transactions sequentially and making all the details available to a group of third-party observers to validate could prevent double-spending of an electronically created coin.

Each of these innovations came together with the announcement of Bitcoin in October 2008 and the launch of the network in 2009. Bitcoin is a peer-to-peer electronic cash payment system that operates in the pseudo-anonymous manner envisioned by the cypherpunks, allowing anyone with access to an internet connection to participate in the network and use Bitcoin as a payment mechanism.

> **"In addition to deploying the new technologies that emerged from the cypherpunk movement, Bitcoin also introduced two important innovations of its own. The first was blockchain—a new type of distributed ledger technology (DLT) that recorded transactions and held the data of those transactions in a novel way that made such transactions highly transparent and immutable. The second was the concept of digital scarcity."**

In addition to deploying the new technologies that emerged from the cypherpunk movement, Bitcoin also introduced two important innovations of its own. The first was blockchain—a new type of distributed ledger technology (DLT) that recorded transactions and held the data of those transactions in a novel way that made such transactions highly transparent and immutable. The second was the concept of digital scarcity. Up until this time, there was no way to ensure that a digital asset being sent by one party to another was not just a copy. By coupling the innovation proposed in RPOW's approach to sequential-transaction tracking and third-party validation with the new blockchain ledger that offered full-transparency across a distributed set of network nodes, it became possible to ensure that a payment coin was removed from one wallet before being sent to another wallet—ensuring that it could not be duplicated or double-spent.

In 2015, the Ethereum network launched, expanding the cryptocurrency arena of bitcoins and similar payment coins. Ethereum created more than a payment network, it offered a completely new digital ecosystem and infrastructure tools. Ethereum offers a new open-source development platform that allows for the creation and deployment of decentralized applications that run and have their business logic and transactions housed within a virtual computer that sits on top of a decentralized payment network.

Decentralized applications developed in the ecosystem are based on smart contracts—self-executing bits of code that describe the specifics of a transaction and cause the transaction to automatically take place when an authorizing message is received or a specific data-trigger is activated. Smart contracts

“Transactions occur differently in Web3—participants pay to have their transactions recorded in tokens native to the L1 platform that the decentralized app sits within, rather than paying fees in a government-backed fiat currency to third-party processors that record their transactions using government-sponsored bank and payment rails.”

use similar or compatible programming languages and are based on templates, giving all developers on the platform a common building block. As such, these contracts are interoperable and the code they contain is composable—meaning that one developer can use another developer's code and build upon it. This is a completely new architectural approach that works differently than service-oriented architectures or microservices.

Ethereum and the other digital ecosystem platforms (L1s) that have launched subsequently have introduced a new Web3 dynamic that is likely to reshape behaviors and societies.

Transactions occur differently in Web3—participants pay to have their transactions recorded in tokens native to the L1 platform that the decentralized app sits within, rather than paying fees in a government-backed fiat currency to third-party processors that record their transactions using government-sponsored bank and payment rails. Decentralized applications can issue their own tokens that serve different purposes—utilization tokens allow the holder to partake in a service, asset tokens represent ownership in a physical or digital item, security tokens convey ownership in a decentralized project or protocol, and governance tokens grant the holder the right to vote on matters affecting the strategic and financial development of the underlying protocol. These tokens have their own individual value based on the relative success of the decentralized application they are associated with, and because they are built on the same template, they can be recognized and used by other protocols as forms of collateral.

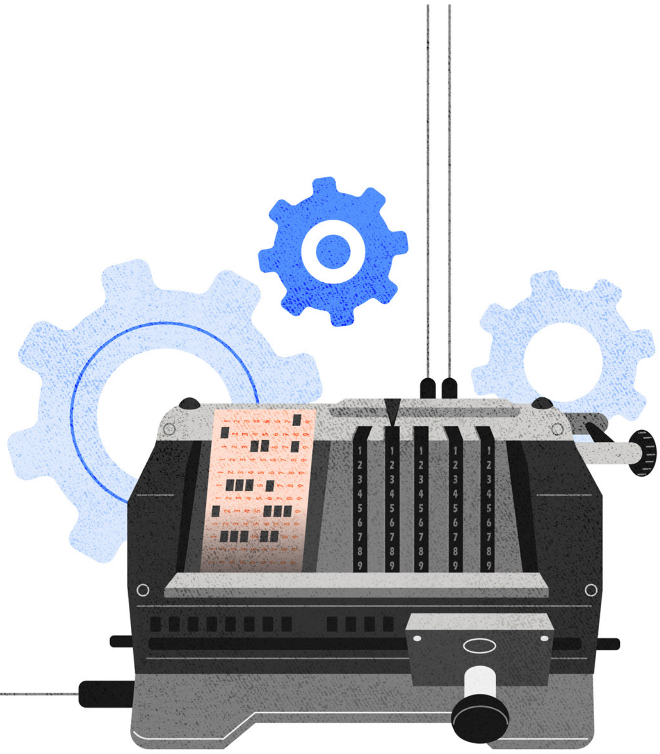
Because these novel token types are built on smart contracts, they can also deliver certain rights and privileges to the holders. These include the ability to administer copyright protections and automatically collect and distribute royalty payments; encapsulate ownership rights and collect or disperse payments and register title; control access to one's digital identity and personal data and collect payments if an individual chooses to share data; and, grant admittance to special communities, events, products and content. These contracts are embedded into the token, and each time a token is transferred from one owner to another, the rights are automatically transferred and re-registered to the new owner.

The build-out of the Web3 space is still in its proof-of-concept stage, but unlike any prior technological innovation, this test phase is running 24/7/365 and is fully transparent to the entire world. Indeed, it is critical to remember that digital ecosystems like Ethereum and the other platforms that have been subsequently launched have been around for less than 10 years. As these offerings look to rewrite the rules of commercial engagement, it will be critical to watch the traction that they can obtain as there is likely going to be a tipping point that starts to see a wholesale re-architecting of the way we live, work and engage.

Section I

Origins of commercial computing

Mid 1800s to 1950s



Humans have been obsessed about capturing and sharing information since our earliest history. From petroglyphs carved into caves to the invention of alphabets and numbering systems, humankind has been seeking to transmit, store and share information. Efforts in this regard have been accelerating as more technological options become available.

The mechanical age—when people began to process information in an automated manner—unfolded over a span of nearly 400 years between 1450 and 1840.[1] This period covered such diverse innovations as the slide rule to the mechanical computer. The next era, the electromechanical age, lasted only 100 years from the 1840s to the 1940s.[2] Innovations since then—considered to be part of the electronic age—are occurring even more rapidly. Using the ability to deploy computers into commercial enterprises as our starting point, it is possible to identify four distinct stages of evolution in just the past 80 years.

Understanding the backdrop for this period of unprecedented change requires revisiting some of the key discoveries and innovations that preceded this commercial milestone. These advances laid the foundation for our current world, and they help to illustrate how rapidly the pace of change is accelerating.

Mechanical calculators

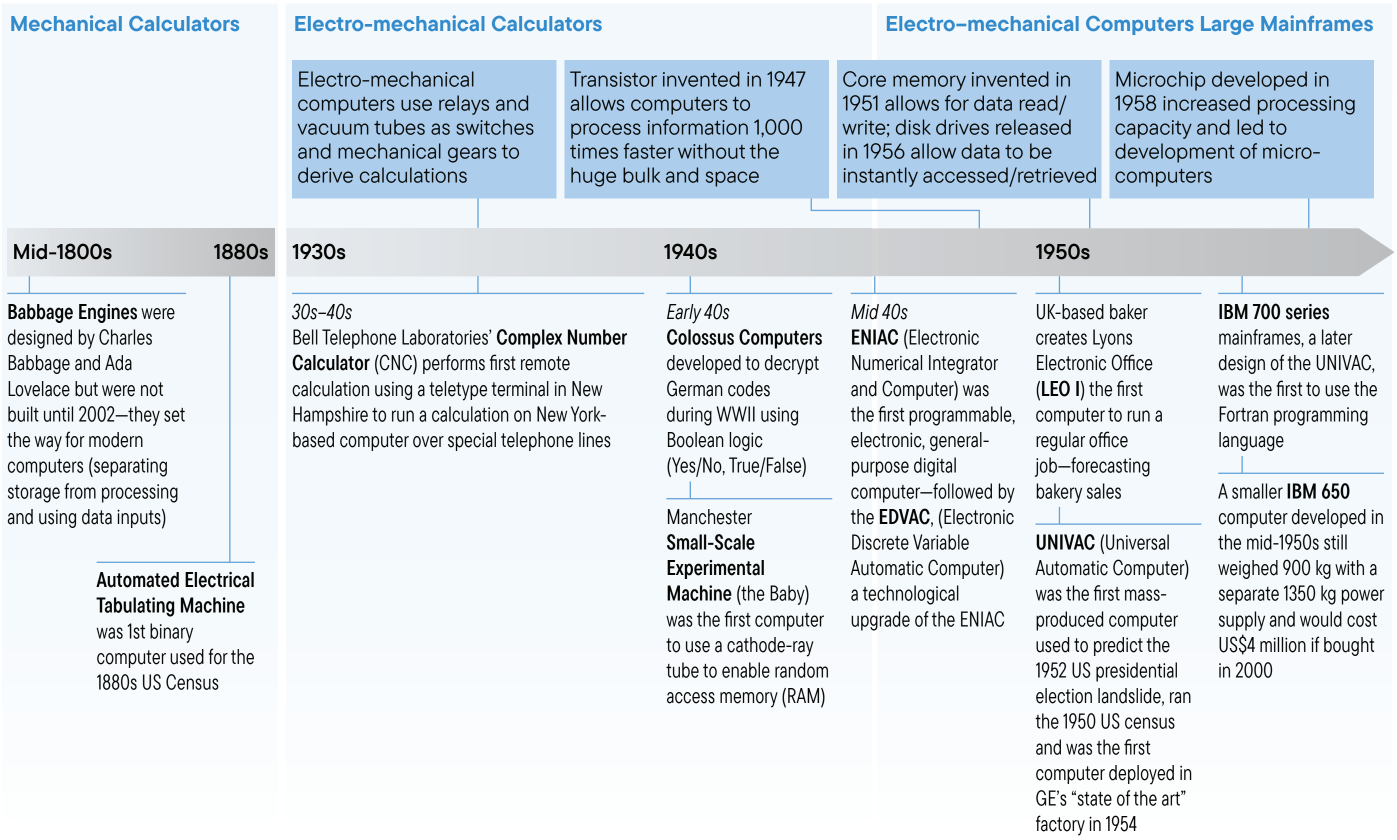
The invention of the Babbage engines in the mid-1800s marked the beginning of the mechanical calculator age.

Charles Babbage (1791–1871) was a computer pioneer who designed two classes of engines. The first, his Difference engines, were able to add numbers mechanically. Babbage was able to assemble one-seventh of his Difference engine #1 in 1832—a demonstration piece consisting of about 2,000 parts that still works and represents the first successful automatic calculating device.[3] Upon seeing a demonstration of the analytic engine, a mathematician Ada Lovelace began a collaboration with Babbage, helping to contribute the idea of computational algorithms and computer programs.[4] This resulted in the second class of Babbage engines—Analytical engines—that were much more than just a calculator and marked "the progression from the mechanized arithmetic of calculation to full-fledged general-purpose computation."[5]

While these early mechanical computers bore little resemblance to computers in use today, they "paved the way for a number of technologies that are used by modern computers or were instrumental in their development."[6] These innovations included separating storage from processing and the way that data and instructions are inputted and outputted.

This age of mechanical computers continued for many more decades. As shown in Exhibit 1, by the 1880s, the automated electrical tabulating machine was used in the 1880s US census to handle data from 62 million Americans. Additionally, the first binary computer—which used only two numbers,

Exhibit 1: Early History of Computers: Mid-1800s to 1950s



Source: Franklin Templeton Industry Advisory Services analysis based on Craig, William, "The History of Computers in a Nutshell," WebFX web site, July 21, 2021; " Timeline of Computer History," Computer History Museum web site, 2022; and "The History of Harvard Mark 1: A Complete Guide," History Computer web site, October 25, 2021. For illustrative purposes only.

0 and 1, to code instructions—was developed by Konrad Zuse and released in 1938. The Z1 computer is commonly referred to as the precursor to the next cycle of early computational development—the age of electro-mechanical computers.[7]

Electro-mechanical calculators

Electro-mechanical calculators that emerged in the 1930s generally worked with relays, variable-toothed gears, and vacuum tubes that could be used as switches to create logic circuits. These were extremely large and unwieldy constructions, and the vacuum tubes themselves used a lot of power and got very hot, making them unreliable.[8] The first large-scale automatic digital computer developed in the United States—the Mark 1 created at Harvard University in 1937—was 8 feet high (2.4 meters) by 50 feet long (15.2 meters) by 3 feet wide (0.9 meters). It weighed five tons and used 18,000 vacuum tubes. The machine took one second to perform three mathematical calculations.[9]

Another pivotal electro-mechanical computer was the complex number calculator (CNC) developed in 1939 by Bell Telephone Laboratories. In 1940, its designer, George Stibitz, demonstrated the CNC at the American Mathematical Society conference at Dartmouth College, stunning the group by performing a remote calculation using a teletype terminal connected to the CNC in New York City over special telephone lines.[10]

World War II was a galvanizing event in the development of these technologies. The Colossus computer was designed to decrypt German codes and was the first machine to perform select Boolean search and operations—the use of variables and simple keywords, terms and symbols for searches and formulas such as "true or false" or "yes or no."[11] Another wartime electric computer, the Manchester Small Scale Experimental Machine, also known as "the Baby," was the first computer to use a Williams tube filled with cathode gas to facilitate random access memory (RAM) storage.[12]

It was also around this time that the terminology began to change away from "calculators." In the defense industry, "computers" referred to the people who worked on complex math equations such as determining a ballistic shell's predicted path accounting for atmospheric variables like air density, temperature and wind. This was a slow and tedious process, not fit for wartime footing. Electro-mechanical calculators available at the time were not much help in this regard. The approach to calculating was to use differential analyzers that were mechanical wheel-and-disc devices to perform integrations. "Setting up a problem involved putting gears of the right size together, and once the problem was set-up on a differential analyzer, it was very hard to change."[13]

Electro-mechanical computers—large lainframes

To address these challenges, the US Army granted funds to the Moore School of Engineering at the University of Pennsylvania to build an electronic computer capable of analyzing data in a timeframe suitable for war efforts. Though it was finished after World War II had ended, the result of this work was the electronic numerical integrator and computer (ENIAC). As the first general purpose, programmable digital computer, it was the fastest computational device of its time, able to do 5,000 additions, 357 multiplications or 38 divisions per second. Problems that took a human mathematician 20 hours to solve took only 30 seconds for the ENIAC.[14]

Because it had no internal storage, it had to be programmed manually for each new set of calculations. Programming was done by a group of six women working on plug boards and banks of switches. Because of the classified nature of their work, the "ENIAC Girls" [sic]—who had all previously been working as computers at the Moore School—only had access to blueprints and were not even allowed into the same room as the device. They used schematics and interviews with its engineers to figure out how to design algorithms and adjust ENIAC's switches for programming calculations.[15] By the time it was retired in 1955, ENIAC had done more calculations by itself than all of humankind had done up until 1945.[16]

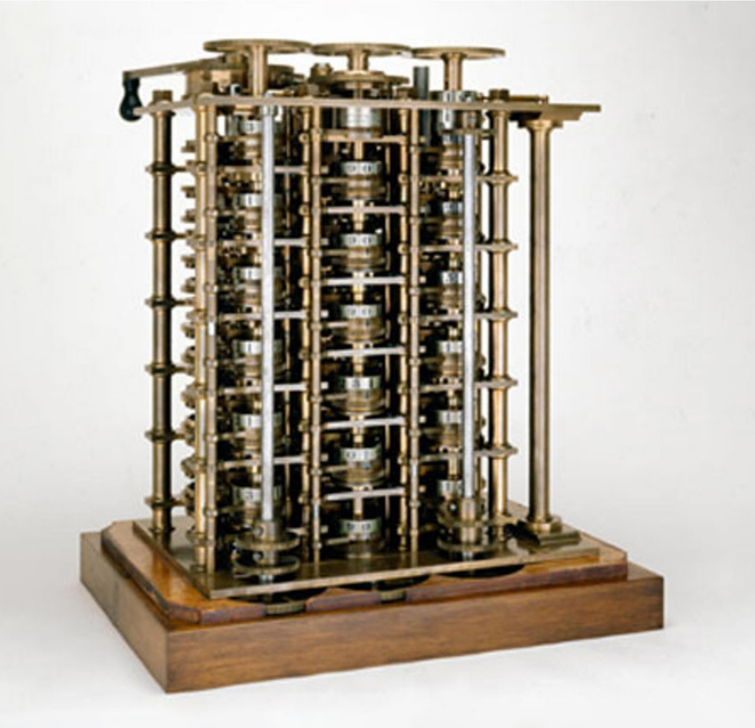
"Despite its success, issues with the ENIAC were apparent from the outset. As such, the designers sought to address its shortcomings through the creation of another computer called the electronic discrete variable automatic computer (EDVAC). The EDVAC used binary computation versus the decimal computation used in the ENIAC. Instead of rewiring the machine each time a user wanted to change the program, the EDVAC introduced the concept of storing a program in memory, just as if it were data."

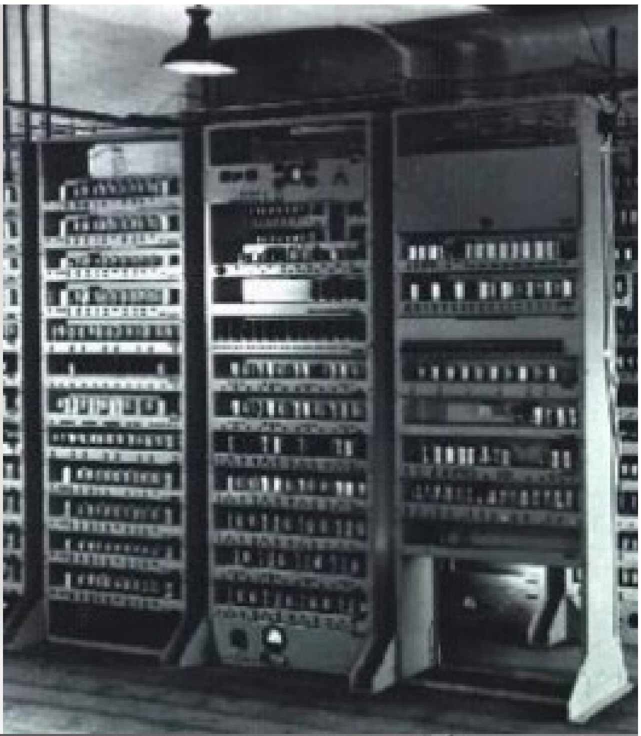
Despite its success, issues with the ENIAC were apparent from the outset. As such, the designers sought to address its shortcomings through the creation of another computer called the electronic discrete variable automatic computer (EDVAC). The EDVAC used binary computation versus the decimal computation used in the ENIAC. Instead of rewiring the machine each time a user wanted to change the program, the EDVAC introduced the concept of storing a program in memory, just as if it were data. Moreover, the memory no longer consisted of vacuum tubes but was stored as electrical impulses in mercury—an advancement that was 100 times more efficient in terms of the electronics necessary to store data, making much larger amounts of memory feasible and more reliable.[17]

The EDVAC was the machine that impressed two senior managers visiting the United States from J. Lyons and Co.—one of the United Kingdom's leading catering and food manufacturers—that came to United States to look at new business methods developed during World War II. These executives saw the potential of using computers to help solve the problem of administering complex business processes and learned from the EDVAC developers that peers at the University of Cambridge were pioneering their own general-purpose computer aligned to the EDVAC design—the electronic delay storage automatic calculator (EDSAC).[18]

Upon the executives' return to the United Kingdom, J. Lyons and Co. provided funding to the Cambridge scientists to help speed their efforts, and once the EDSAC computer was up and running, the company started construction of its own machine. The resulting Lyons Electronic Office I, or LEO I, was modeled on the EDSAC. The first business application to be run on the LEO I was bakery valuations that computed the cost of ingredients used in breads and cakes. After the first successful run of the application on September 5, 1951, the LEO I took over the firm's bakery valuation calculations completely by the end of November 1951.[19]

Though the LEO I marked a key milestone for the commercial application of computing, it was still a behemoth in terms of size, taking up 2,500 square feet of floor space and cumbersome in terms of programming as it relied on paper tape readers and punches.[20]





TOP LEFT: Difference Engine 1. Source: Computer History Museum web site
TOP RIGHT: The EDVAC Computer. Source: Generation of Computer web site
BOTTOM: IBM 650 at Texas A&M University. Source: Wikipedia.

> “Despite its success, issues with the IBM’s entry into the computing realm started concurrently with the UNIVAC and its work on two classes of computer. Up until this time, IBM was known for its punched-card accounting machines. Thomas Watson Jr., IBM’s president, informed the company’s shareholders at the annual meeting in 1952 that IBM was “building the most advanced, most flexible high-speed computer in the world.”

In 1951, the UNIVAC—the universal automatic computer—was released. It was a stored-program computer that could tabulate 4,000 items per minute and calculate a complicated payroll for 10,000 employees in only 40 minutes.[21] This computer was designed by the same team that had developed the ENIAC and the EDSAC, but the team had left academia and had moved into the commercial realm, working on a requisition from the US Census Bureau. They were able to convert information from punch cards to magnetic tape, allowing the new offering to come in at only half the size of the ENIAC.

The UNIVAC ended up being the first mass-produced computer, with 46 units eventually sold.[22] It made a big splash in the 1952 US presidential election by predicting the Eisenhower landslide before the polls closed in California, using statistical sampling techniques from previous elections.[23] The first commercial sale of the UNIVAC was to General Electric (GE) in 1954 to use in its brand new “state-of-the-art” Major Appliance Division plant in Louisville, Kentucky. As one GE executive put it, “If a computer could predict election results, why couldn’t it forecast sales, lay out production schedules, simulate factory operations, perform “what-if” financial analyses and solve a whole range of engineering, scientific and operations research problems.”[24]

IBM’s entry into the computing realm started concurrently with the UNIVAC and its work on two classes of computer. Up until this time, IBM was known for its punched-card accounting machines. Thomas Watson Jr., IBM’s president, informed the company’s shareholders at the annual meeting in 1952 that IBM was “building the most advanced, most flexible high-speed computer in the world.”[25] This marked a shift in company strategy, transitioning the company from punched-card machines to electronic computers.

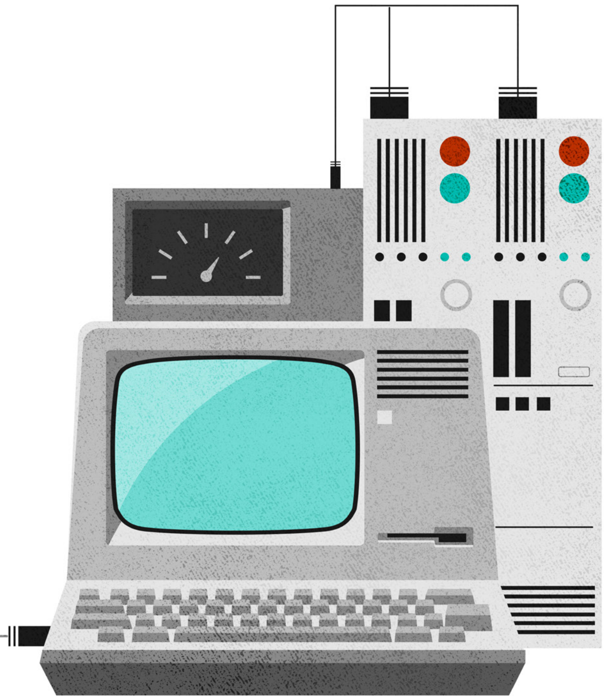
IBM’s 700 series, announced in 1952, was a direct competitor to UNIVAC in 1952. IBM’s central processing unit (CPU) was much faster than UNIVAC’s—it was able to process 2,200 multiplications per second versus only 455 for UNIVAC’s. It could also execute almost 17,000 additions and subtractions. The 701s’ 8-million-byte tape drive could also start and stop much faster than UNIVAC’s and it was capable of reading or writing 12,500 digits per second. IBM produced 19 units of the 701 series, offering access at US$16,000 per month since IBM did not sell computers at the time but only leased them.[26]

IBM’s 650 magnetic drum processing machine came out in the mid-1950s as a lower-end offering than the 701 (and the later 700 series releases). It was positioned somewhere between the big mainframes like the 701 and the UNIVAC, as well as the punched-card machines used at the time, which were still dominating the market. The 650 still weighed over 900 kilograms with a separate 1350-kilogram power supply,[27] but it only cost US$3,250 per month. Just over 2,000 of these machines were built and leased. Although very reliable by computing standards, it still used vacuum tubes and was thus inherently less reliable than IBM’s electromechanical accounting machines.[28]

IBM’s replacement for the 650, which came out in October 1959, marked the end of the early computer era. The IBM 1401 Data Processing System “was the computer that made punched-card machines obsolete.”[29] Its combination of functionality and a relatively low cost allowed many businesses to start using computer technology. It allowed IBM to become the dominant computer company of the era.

Section II

First cycle of commercial technology—automation

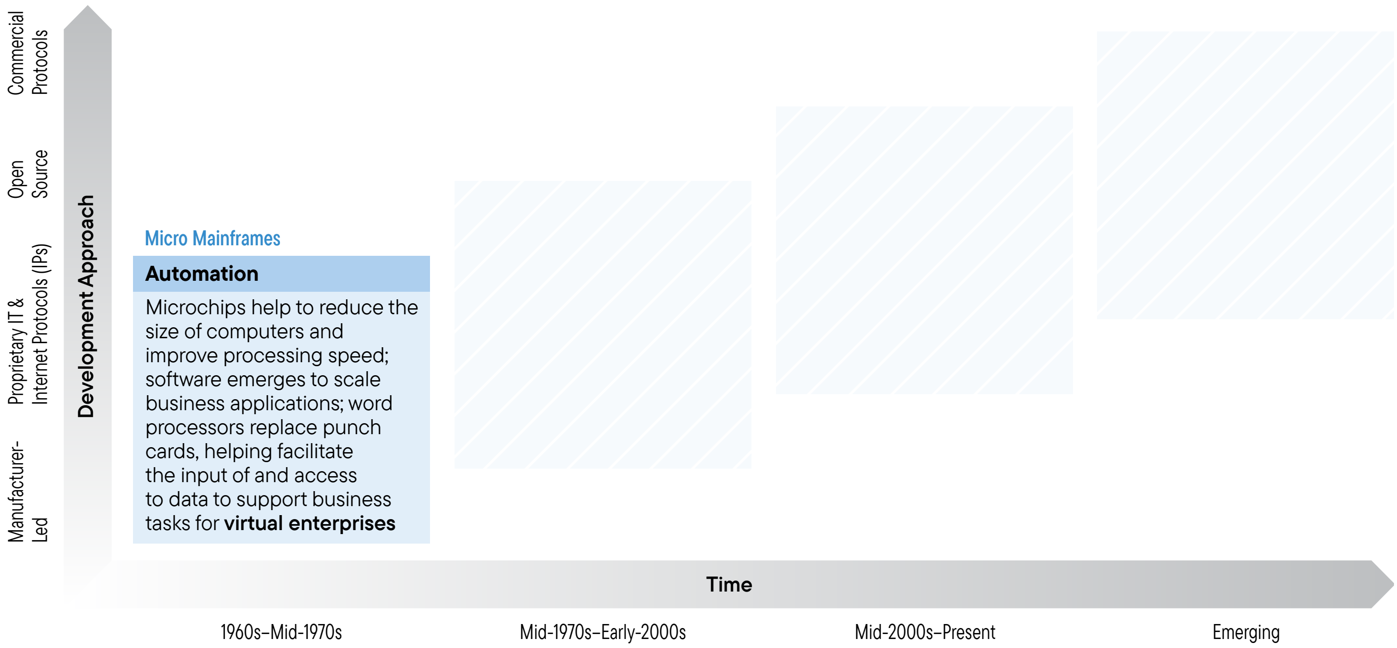


The nature of work changed as new, affordable mainframes moved into the commercial realm. Tasks that had been performed manually began to be automated, particularly those that involved extensive amounts of data and large numbers of calculations. Soon after, solutions emerged to facilitate the creation and storage of documents. For the first time, many business functions that could take several days or more to complete could be accomplished within a single business day. More intelligence could also be gleaned from the timely delivery of information, enhancing a firm's decision-making process and enabling the creation of new types of data-driven products.

Exhibit 2 illustrates that this period of "Automation" began in the 1960s and continued through the mid-1970s.

From a technological standpoint, there were five innovations that helped set the stage for the commercialization of computer technology.

Exhibit 2: Cycle 1 of Modern Commercial Technology



Source: Franklin Templeton Industry Advisory Services. For illustrative purposes only.

- **Transistors:** In 1947, the transistor made up of semiconductors was invented. The transistor was the first device designed to act as both a transmitter (converting sound cycles into electronic cycles) and a resistor (controlling electronic current).[30] This invention allowed for computers to be developed that could process information 1,000 times faster than previous computers without the bulk and huge space once needed. Transistors allowed for the replacement of vacuum tubes and resulted in much smaller computers. Transistors offered reliability, improved power use, heat dissipation and were lower cost than vacuum tubes.

- **Core memory:** In 1951, a patent was filed by An Wang (Wang Laboratories) for the creation of a magnetic core memory—an early form of random-access computer memory. The patent right was then sold to IBM in 1955. Prior to this, memory was stored in cathode ray tubes, but these devices were temperamental and unreliable.[31] Core memory allowed for the introduction of a read-and-write cycle. In computer operations, a read cycle causes the memory contents to be lost, whereas the write cycle restores the content to the memory location. Moreover, core memory allowed for non-volatility—that is, an ability for the computer to maintain its memory when the power supply is lost.[32]

- **Disk drives:** In 1956, IBM introduced a technology known as the random access method of accounting and control (RAMAC), the first magnetic disk drive for computers and a progenitor to today's hard drives. For the first time, RAMAC enabled instantly accessible information. Prior to its launch, information was entered into a computer by running a stack of cards through a punched-card machine. Answers to inquiries would arrive in hours or days. RAMAC could find data in seconds, alter it, and move on to another piece of data. "It let enterprises think about data in new ways, mixing and matching it on the fly."[33] Random access made the relational database possible. The day after IBM's release of its RAMAC computer, a newspaper report noted that, "Card-sorting, one of the most time-consuming office-machine processes, is eliminated or greatly reduced."[34]

- **Software development:** Software is the interface between computer systems and users. It consists of programming instructions and data that tell the computer how to execute various tasks. Fortran (or formula translation), one of the very first higher-level programming languages, was originally published in 1957.[35] By 1964, BASIC (beginner's all-purpose symbolic instruction code) was created in response to the exclusivity and extreme difficulty of software development.[36] Early software was "hard-wired" instructions built into the memory of the machine, but it allowed for those tasks to be performed upon demand by all buyers of the computer systems rather than having to be custom-programmed by each buyer.

"These innovations all came together in the IBM 1401 Data Processing System. The development of the IBM 1401 system was shaped by industry and government need to process rapidly growing amounts of information quickly. For many customers, it was the first computer they owned. In many ways, it was a bridge computer."

- **Microchips:** In 1958, the integrated circuit chip was developed. Until that time, transistors were manufactured as discrete devices and wired together into circuits. Such circuits were still bulky and fragile.[37] Integrated circuit chips (also called silicon chips) are small, thin, rectangular chips or tiles of crystalline semiconductor that are layered with large numbers of microscopic transistors and other electronic devices. This invention made it possible to miniaturize computers, communication devices, controllers and hundreds of other devices. By 1971, the size of these circuit chips had shrunk down to the equivalent of a grain of rice—giving rise to the microchip. For about 40 years, the number of electronic components that could fit on an individual microchip at a certain cost doubled every few years. This trend, identified by US engineer Gordon Moore, became known as Moore's Law.[38]

Computing goes mainstream

These innovations all came together in the IBM 1401 Data Processing System. The development of the IBM 1401 system was shaped by industry and government need to process rapidly growing amounts of information quickly. For many customers, it was the first computer they owned. In many ways, it was a bridge computer. Optional tape and disk attachments "allowed customers to start moving away from the storerooms (sometimes warehouses) of punched cards they had accumulated onto more compact formats. A single reel of magnetic tape could store the equivalent of tens of thousands of punched cards while disks allowed in-line data processing and the rapid random access of data that punched card and tape systems lacked."[39]

The IBM 1401 was the world's most popular computer during much of the 1960s. By 1965, worldwide installations of the 1401 and its family of machines represented half of all computers being used and peaked in 1967 at about 15,000 systems.[40] The reliability of the 1401 "was renowned and many systems operated around the clock. IBM had a large organization of customer engineers that worked closely with users to maintain their systems.[41]

Enhancements enabled by the 1401 and competitor offerings made electronic data processing work more effectively, bringing the activities into a business day timeframe. Up until this time, in the early decades of the 20th century, "business data processing—inventory, billing, receivables, payroll—was accomplished by passing decks of punched cards through various electro-mechanical accounting machines to sort, calculate, collate, print and punch. Each machine was controlled by a hand-wired plugboard tailored for a particular job."[42]

Stored-program computers, such as the UNIVAC discussed earlier, were considerably more flexible and adaptable, but were too expensive for all but the largest corporations, renting for about US$30,000 per month (US$200,000 in today's currency) versus only US$2,500 per month for several accounting machines. By contrast, a typical 1401 system rented for about US$6,500 per month or could be purchased outright for US$500,000—equivalent to rental price of US$45,000 per month and a purchase price of US$3.4 million in today's terms.[43]

Business and workforce impacts of computation in the workplace

Many voiced concerns over the impacts to employees, their humanity, and their ability to adjust to the introduction of automation enabled by computers into the workplace. A 1960 article in the Harvard Business Review by a renowned sociologist based on a study of 19 San Francisco Bay Area-based organizations across industry types and sizes found the following:[44]

- Electronic data processing reversed the trend toward a decentralized company and office. Prior growth was associated with "a certain amount of dispersion of functionality and authority. Now that data could be processed quickly, records could be kept centrally, reducing the need for branch-level paperwork, and resulting in workers being transferred, downgraded, or dismissed."[45]
- This consolidation of power with the electronic data processing teams at the central hub disrupted the prior organizational exhibits and helped lead to "empire-building" as those closest to the information began making independent decisions and undercutting other departments. The report noted that "vice-presidents in charge of...find their official functions atrophied as there is little for them to be in charge of."[46]
- Middle-management jobs that had been used as the training ground to groom the next generation of leaders were transformed. "Eager professionals found themselves checking data for errors before it was processed, rather than taking initiative or proving shrewd judgment."[47]

Yet, for many organizations, these concerns were minimized by the promise of "increased productivity, greater efficiency, speed and accuracy." The development of business-friendly computer applications accelerated.

Document management systems

In addition to data processing improvements, document management systems were also introduced in this period, leading many to speculate about a future "paperless society," but instead leading to an influx of new data-entry clerk jobs as paper-based documents needed to be "computerized."[48]

The word processor and printer were important computer "peripherals" or connected devices that emerged to facilitate document management. In 1961, IBM introduced the Selectric typewriter that replaced the standard movable carriage and individual typestrikers with a revolving ball, allowing it to print faster than a traditional typewriter. In 1964, IBM brought out the MT/ST (magnetic tape/Selectric typewriter) which combined the features of the Selectric typewriter with a magnetic tape, which was the first reusable storage medium for typed information. IBM used the term "word processing" in its marketing materials for this product, touting it as "an electronic way of handling a standard set of office activities—composing, revising, printing, and filing written documents."[49]

By the early 1970s, IBM had developed floppy disks, which could be used to hold programs. This allowed word processing to shift from hard-wired instructions built into the

"Having the data on floppy disks also meant that the information became portable. Many of the universities that offered computer access at the time had one or two computing centers that housed the large equipment which required a constant air-conditioned facility. This facility would have terminals and printers. Researchers carrying data on floppy disks from building to building wanted better options to share computing while still being able to work from their offices or labs."

machinery to software on disks. This made the idea of "upgrading" software much easier to facilitate. When the programs were built into the machines themselves, it was difficult to change and expensive to upgrade these programs. With floppy disks, programs could be updated more economically. The emergence of the C programming language in 1972 helped in this regard by moving software development to a more accessible language.

Having the data on floppy disks also meant that the information became portable. Many of the universities that offered computer access at the time had one or two computing centers that housed the large equipment which required a constant air-conditioned facility. This facility would have terminals and printers. Researchers carrying data on floppy disks from building to building wanted better options to share computing while still being able to work from their offices or labs.

This led to a trend toward miniaturization of mainframe computing. The first commercially successful minicomputer that was small enough to sit on a desktop was developed by Digital Electronics Corporation in 1965. The PDP-8 sold for only US$18,000, only one-fifth of the cost of the low-end IBM/360 mainframe in use at the time. The combination of speed, size and cost enabled the establishment of the minicomputer in thousands of manufacturing plants, offices and scientific laboratories.[50] An iteration on this product, the PDP-8E, appeared in 1970. This general-purpose minicomputer allowed many types of peripheral devices to be connected to it such as teletypewriters and line printers.[51]

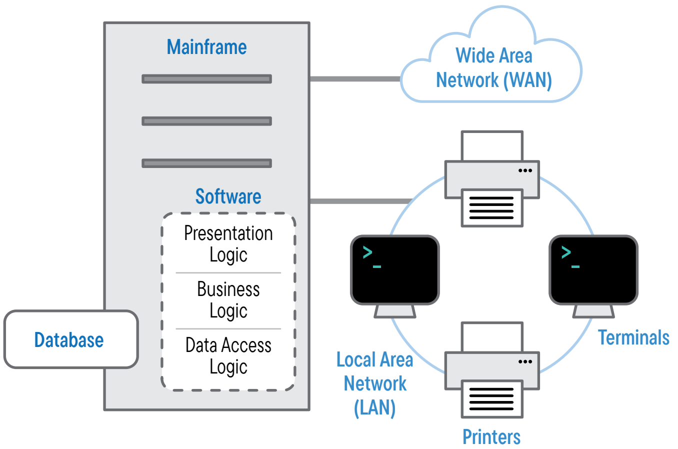
These minicomputers included typewriter-like input devices and cathode-ray tube monitors. To share computing information and resources, universities and some early businesses that purchased such hardware infrastructure networked them—linking the growing set of office or lab computers via network cable (ethernets) to create the first local area networks (LANs) and combining these LANs together with a new type of wide area network (WAN) technology to manage the digital-data traffic.[52]

Data processing and document management benefits enabled by the new technologies were resulting in a sharp growth in the amount of proprietary data and intellectual property that was being created. The need to store and access this information efficiently and benefit from the growing amounts of digital intelligence that enterprises were beginning to accumulate resulted in the first iteration of organizations having an IT group.

Monolith IT architectures

In the 1960s and 1970s, many large enterprises with the financial means and high customer volumes, such as banks, insurance, government entities and telecommunication companies, created IT groups to manage their technology infrastructure. At the time, this consisted of the organization's computer mainframe, corporate databases, peripheral devices, LANs and WAN. Each of these resources were connected directly to the other point-to-point to operate like a single system. They were custom-configured to meet the organization's specific needs. This is shown in Exhibit 3.

Exhibit 3: Monolithic IT Architecture of the 1960s and 1970s



Source: Franklin Templeton Industry Advisory Services. For illustrative purposes only.

In most instances, this entire set of computer technologies was purchased from a single company. IBM was the leading supplier of the time, but the firm had several competitors—in the large mainframe space from firms like Texas Instruments and Xerox[53] and in the mid-sized mainframe space from a group known as the "BUNCH." This was an acronym for Burroughs, Sperry's old Univac division, NCR Corporation, Control Data Corporation, and Honeywell.[54] These firms offered either large or mid-sized mainframes and had also created the accompanying databases and peripherals that allowed them to offer an all-in-one set of technologies.

Internal IT teams in major enterprises would have service contracts with their main supplier. Rather than the IT team maintaining the equipment, they were used to establish and troubleshoot the networks that were being created and administer the database, ensuring the logical data architecture and storage was understood and maintained, and helping users extract information as required to run and produce reports. Hence the name, information technology, or IT, group.

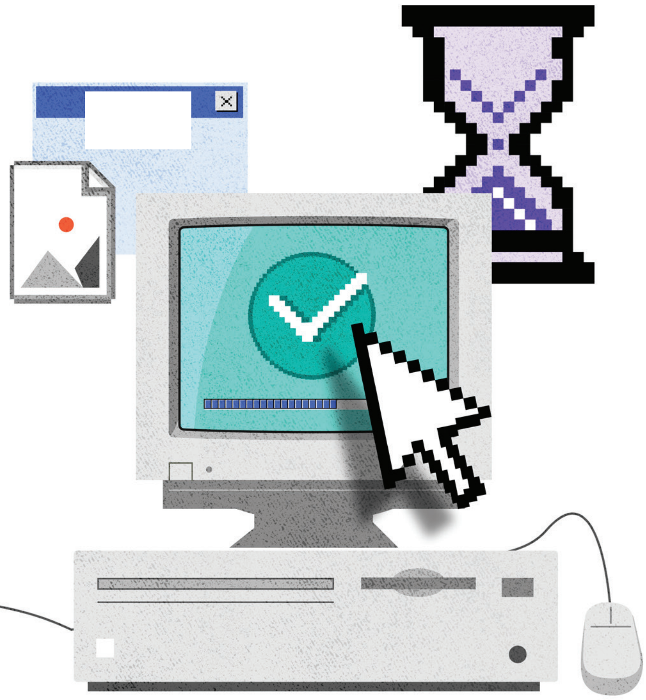
The growing use of computers was allowing for a "knowledge explosion." An article from 1971 noted that "the accumulation of scientific data, which a century ago was doubling every 100 years is now doubling every 6 years." The article also noted that "Science is a body of knowledge which has been estimated to be the equivalent of some 10,000,000 books on science and its applications. More importantly, this body of knowledge is increasing at a rate estimated to be about 1,000,000 book equivalents a year, or on the order of 100 books an hour."[55]

The desire to better harness and apply this knowledge was growing. In 1969, Xerox Corporation bought Scientific Data Systems, a mainframe computer manufacturer. Shortly thereafter, they started the Xerox Palo Alto Research Center (PARC) in California. The center opened in 1970. "By the mid-1970s, close to half of the top 100 computer scientists in the world were working at Xerox PARC, and the laboratory boasted similar strength in other fields, including solid-state physics and optics."[56]

One of Xerox PARC's key initiatives was to define the "office of the future." This was where the shift from the first cycle of commercial technology innovation gave way to the next cycle. The creation of the PC kicked off this next cycle of innovation, followed later by the introduction of the internet.

Section III

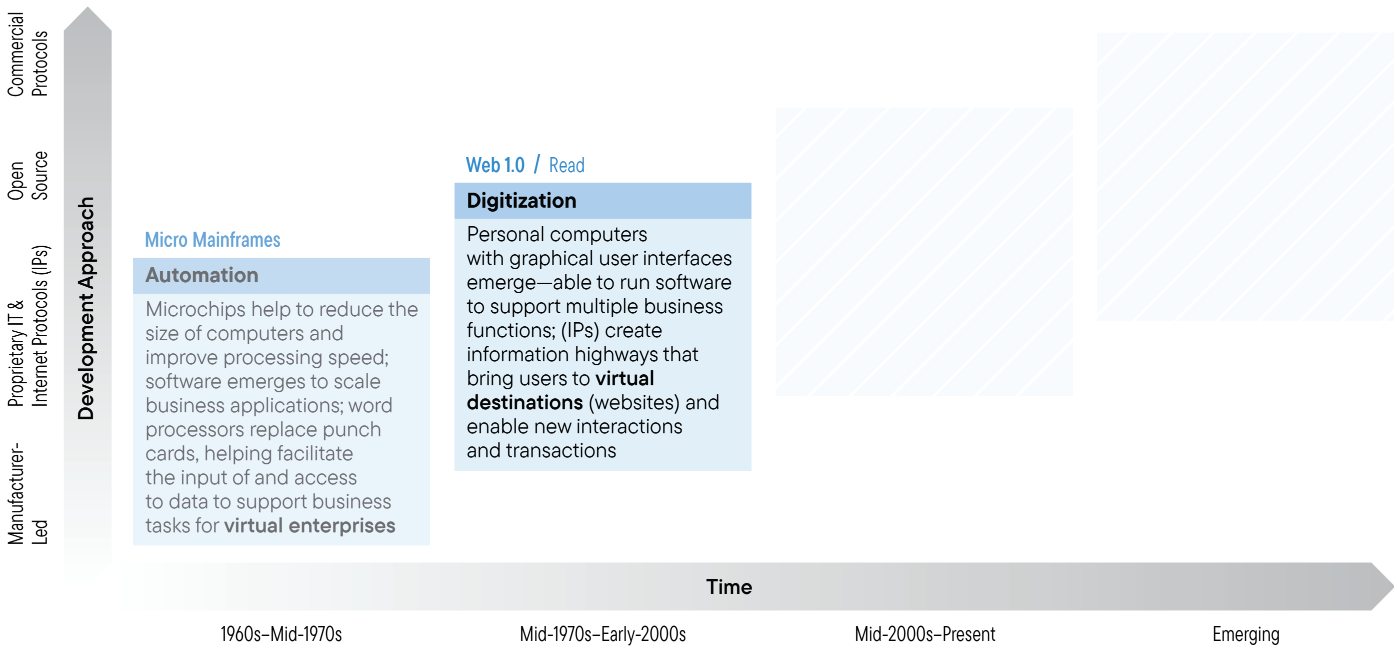
Second cycle of commercial technology innovation—digitization



Whereas the first cycle of commercial technology innovation transformed the way in which enterprises could run their business, the second cycle extended the opportunities of technology to individuals, both within and outside of commercial enterprises. People began to be able to do more on their own within their workplace and the resulting explosion in creativity and intellectual property changed the way in which business was architected and delivered. Individuals could also engage in business activities independently, using new abilities and access points to reshape the way they pursued their livings and engaged with the world and each other.

Exhibit 4 shows the progression in technology approach and illustrates that the digitization cycle that began in the mid-1970s extended all the way until the early 2000s.

Exhibit 4: Modern Commercial Technology: Cycles 1 & 2



Source: Franklin Templeton Industry Advisory Services. For illustrative purposes only.

Personal computers

The defining event that prompted the shift between the first and second cycles was the introduction and adoption of the PC. Up until this event, computing capabilities had been tied exclusively to the enterprise's mainframes and the peripherals that could connect users to that hub. PCs began to disperse the ability to create and run programs, perform tasks and produce intellectual property.

The introduction of the microprocessor in 1971 was a seminal event in the development of personal computing. Invented by an engineer at Intel, the first microprocessor was a $\frac{1}{16} \times \frac{1}{8}$-inch chip called the 4004 that had the same computing power as the massive ENIAC computer.[57] This microprocessor allowed minicomputers to shrink further down to microcomputers which soon became known as PCs.

In 1974, a company called Micro Instrumentation and Telemetry (MITS) introduced a mail-order build-it-yourself PC kit called the Altair. Thousands of hobbyists bought the US$400 kit, but the functionality of this early offering was limited. It had no keyboard and no screen, and its output was just a bank of flashing lights. Users input data by flipping toggle switches.[58]

In 1975, MITS hired a pair of Harvard students named Paul G. Allen and Bill Gates to adapt the BASIC programming language for the Altair. The software made the computer easier to use and it proved to be a commercial success. In April 1975, the two programmers took the money they made from "Altair BASIC" and formed a company of their own, Microsoft. The following year, in 1976, two engineers in the Homebrew Computer Club in California's Silicon Valley, named Steve Jobs and Stephen Wozniak, built a homemade computer called the Apple I, which was also a hobbyist PC kit. The Apple I was more sophisticated than the Altair. It had more memory, a cheaper microprocessor and a monitor with a screen. Users could store their data on an external cassette tape, which Apple soon swapped for floppy disks. By April 1977, the Apple II was released—a pre-built computer for which the company encouraged programmers to create "applications."[59] Many consider the Apple II to be the first PC useful for business tasks.

"This is where the story of the "office of the future" initiative being run at Xerox PARC comes back into play. Researchers there had been developing their own workstation that they codenamed Alto. The Alto performed word processing, electronic messaging, printing, filing and document distribution. It was also being used to deliver interactive graphics experiments such as statistical tables and exhibits."

One of these applications built for the Apple II was VisiCalc (visible calculator)—the first spreadsheet computer program for PCs. VisiCalc allowed data sorting and storage in tabular rows and columns. Changing a single value modified the entire spreadsheet because changes made to one cell were automatically applied to all connected cells. The implications for small business were far-reaching. A 20-hour-per-week bookkeeping task could be reduced to a few minutes of data entry.[60] This was considered the first "killer app"—a piece of software so crucial that people bought the computer just to access it. VisiCalc sold 700,000 copies in six years and up to 1,000,000 copies during its existence.[61]

This is where the story of the "office of the future" initiative being run at Xerox PARC comes back into play. Researchers there had been developing their own workstation that they codenamed Alto. The Alto performed word processing, electronic messaging, printing, filing and document distribution. It was also being used to deliver interactive graphics experiments such as statistical tables and exhibits. The Alto combined a keyboard for data entry and a mouse for commands. The mouse offered two-dimensional motions to move an arrow on a screen that allowed users to interact with a GUI filled with "icons" that could be used to launch programs. This approach allowed infrequent users to initiate actions without having to know the accompanying program commands.[62]

Despite the revolutionary potential of this design, Xerox managers were uninterested in commercializing the Alto as they were focused instead on copier and printer innovations. In the end, Xerox funded the production of only 2,000 machines.[63] Many of the PARC researchers began to leave, spreading out to other technology-oriented firms in the area, including Microsoft and Apple.

After several of his employees had gone to see a demo of Alto, they convinced Steve Jobs to join them in late 1979 for a visit to Xerox PARC. Jobs described the visit as follows: "I was so blinded by the first thing they showed me, which was the GUI. I thought it was the best thing I'd ever seen in my life…And within, you know, 10 minutes, it was obvious to me that all computers would work like this someday."[64] Jobs arranged for his entire programming team to be given full technical demos. In exchange, he sold 100,000 shares of Apple to Xerox, and the Xerox management was none the wiser.[65]

Meanwhile, Apple had hired Microsoft as its first third-party software developer to work on its new offering, the Macintosh. Because Jobs knew that Microsoft was aware of the Alto and its innovations, he made Microsoft sign an agreement as part of their deal in 1981, stating that "Microsoft could not release mouse-based software until a year after the Mac, which the contract stated would happen in the fall of 1983."[66] Unfortunately for Apple, the Mac's release date got pushed back, but the contract date stayed the same. While the Mac would not debut until 1984, Microsoft's Bill Gates made a surprise unveiling of a GUI environment he called Windows in November 1983 at the computer industry's premier trade show.[67]

While battles over the usability of emerging PCs continued between Microsoft and Apple, another established participant saw a different potential to turn its business users into computer users.

In 1981, IBM entered the PC market. This was a bold move for the company. At the time, an entry-level computer at IBM was a US$90,000 IBM System/38 minicomputer or the 50-pound IBM portable computer selling at US$9,000. The typical margins on these machines were 20% to 60%, and IBM additionally captured the software and services that these computers required. Building the PC mandated a major change in approach.[68]

Up until this time, over its 70 years of its history, IBM had designed and made nearly everything it sold. To get a working computer that could be sold at its target price of US$1,500, it needed to use "off-the-shelf" parts. IBM went to Microsoft for the operating system, Intel for the processor and Epson for a dot-matrix printer. Even more surprisingly, the development team opted to make the IBM PC an "open architecture" product and published a technical reference on the system's circuit design and software source codes. This enabled other companies to develop software and build peripheral components. In another major departure, IBM sold the PC through retail stores.[69]

"By 1984, the computer industry passed a remarkable milestone. For the first time, the value of desktop PCs sold in the United States overtook the sales of mainframes. A New York Times article at the time noted that "mainframes are still used by thousands of big corporations…but the biggest segment of the industry—mid-sized machines that represent about 4 out of every 5 mainframe sales—have stopped growing and might soon contract."

The impacts of IBM entering the market in this way were significant. It moved the PC out of the realm of hobbyists and "tore down the wall between professional and personal computing." According to the director of communications for the IBM PC following its launch, "it legitimized computing at the individual, personal level. It also created an ecosystem for technology introductions, how we do open systems, applications, and add-on hardware development, and how we approach distribution channels."[70] Indeed, companies such as Compaq, Dell and HP, among others, reverse-engineered the IBM PC and came out with their own lines of "IBM-compatible" PCs and peripherals, creating a multibillion-dollar industry. In 1983, Time magazine put the PC on its cover as the "Machine of the Year".[71]

By 1984, the computer industry passed a remarkable milestone. For the first time, the value of desktop PCs sold in the United States overtook the sales of mainframes. A New York Times article at the time noted that "mainframes are still used by thousands of big corporations…but the biggest segment of the industry—mid-sized machines that represent about 4 out of every 5 mainframe sales—have stopped growing and might soon contract."[72]

Feeding this trend was a set of enhancements tied to the expanded use of PCs and advances in communication technology within the office environment that were changing the way in which white-collar workers operated.

Another of the innovations coming out of Xerox PARC and the Alto project was the concept of "office automation networks." Within the Xerox PARC environment, there were over 100

> **“The 1980s are when more technology competitors and commercial off-the-shelf (COTS) systems started to move technology away from the dominance of a handful of firms such as IBM out to a more dispersed set of providers. New competitors to IBM, such as Oracle and other Unix-based operating systems were competing in the database space. New monolith systems providers were also emerging to offer solutions focused on functional specialties.”**

Altos networked together able to share messages, documents and more, including an ability to send mail electronically between users on the network.[73] This marked a significant evolution from the office automation of the 1960s. At that time, office automation was seen as the “application of computers to well-structured, high volume office tasks such as payroll processing, accounts payable, purchasing, etc.,” whereas the office automation network of the late 1970s referred to “the application of computer and communications technology to less structured office functions that improve the productivity of the white-collar labor force.”[74]

Analysis done by the Massachusetts Institute of Technology (MIT) showed that in the late 1970s, organizations had made fundamentally different decisions about the capital/labor decisions in the factory versus the office environment. The average capitalization per factory worker was cited at about US$25,000, whereas the average capitalization per office worker was seen as between US$2,000 and US$6,000.[75] The report notes that in the factory, opportunities for automation have led the organization to trade labor costs for capital investment, with the result being impressive productivity increases. Such capital investments had not been done in the office environment, despite office costs increasing from 20%–30% of total company costs to 40%–50% over the past 20 years.[76]

Interest in purchasing technology solutions to enhance white-collar productivity in office environments soared by the early 1980s. Desktop computer sales jumped to US$11.6 billion in 1984, with most of those sales going to corporations rather than to home users. Mainframe sales in that same year slipped down to US$11.4 billion.[77]

For the office user, desktop PCs that each contained their own disk drives, processors and memory were linked to a broader set of services enabled by office automation networks—configurations of networked computer hardware and software linked to a LAN that connected the computers with a series of printers and a mainframe computer or a file server with even greater processing power and memory storage.[78] The types of functions integrated by an office automation system included electronic publishing, electronic communications, electronic collaboration, image processing and office management.[79]

Introduction of enterprise architecture

By the mid-to-late 1980s, more enterprises, including medium-sized organizations, had begun to implement office automation solutions, and were creating IT groups to manage the LANs that these solutions required. New commercial off-the-shelf software was making it easier for companies to provide business applications to their employees to expand the benefits of automation and facilitate their work activities, and specialized commercial off-the-shelf hardware was making it easier to create flexibility and take control over an organization’s tech capabilities.

Demands to better access, incorporate and share data began to rise and interest in automating company-specific processes grew. IT teams began to build out proprietary “infrastructures” for their organization—connecting their office automation networks to new types of business systems and bringing in new types of relational databases to manage the resulting deluge of information and data being created by the new offerings.

The 1980s are when more technology competitors and commercial off-the-shelf (COTS) systems started to move technology away from the dominance of a handful of firms such as IBM out to a more dispersed set of providers. New competitors to IBM, such as Oracle and other Unix-based operating systems were competing in the database space. New monolith systems providers were also emerging to offer solutions focused on functional specialties. These included enterprise relationship management systems, customer relationship management systems and accounting systems where interfaces, business logic and databases were delivered as one component. Commercial software providers were being broadly adopted via offerings such as Microsoft Word and Lotus 1-2-3.

Companies quickly began to purchase various types of these offerings as they reduced the development time needed to introduce new capabilities, allowed for faster insertion of new technology, and lowered the lifecycle cost of the offering by being able to defray development costs across a broad commercial user base.

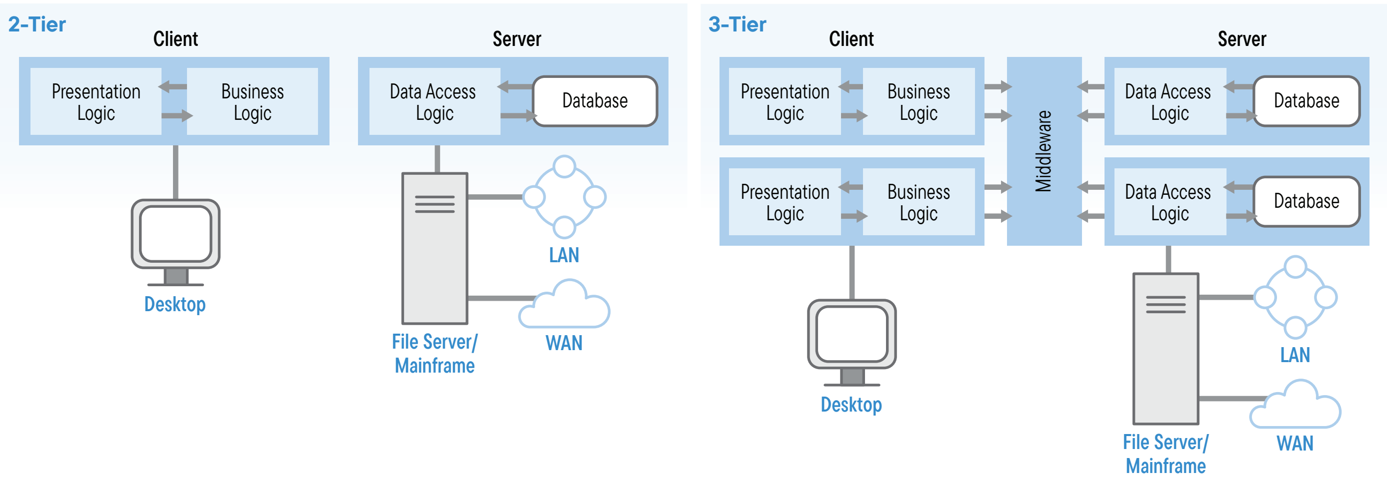
Relational databases were a key enabler of improved business functionality. In the 1960s, databases were either network-driven or hierarchical—both of which stored data in tree-like parent-child structures. Data was implicitly joined together in this approach. A good example is a folder system for storing files. In the 1970s, a new approach was introduced in a paper by E. F. Codd that changed the way that people thought about databases. Instead of using tree-like structures, Codd speculated that the data schema and the data storage approach could be uncoupled. Data could be stored in tables that had rows and columns, and the tables could be mapped to the schema, enabling the computer to search for required information without having to navigate the entire tree. This made it much easier and faster for people to extract specific bits of information and join them together—a must have for the white-collar workforce looking to create reports and analytics.[80]

As the complexity of a company's internal technology grew, the need to have a plan to guide the development of their platforms and their strategic decision-making processes about their technology approach and how it tied to the business's needs became more important, leading to the emergence of enterprise architecture. This approach "combines people, data, and technology to show a comprehensive view of the inter-relationships within an information technology organization. The process is driven by a comprehensive picture of an entire enterprise from the perspectives of the owner, designer, and builder."[81]

Having an enterprise architecture approach allows for more open collaboration between the business and IT units; gives the business the ability to prioritize investments; makes it easier to evaluate existing architecture against long-term goals; establishes processes to evaluate and procure technology; and provides a comprehensive view of the IT architecture to all business units outside of IT.[82]

Early enterprise architectures were typically two- or three-tiers. These were often called client/server architectures. The client-server model describes a process in which an application is divided into two parts that work together to provide a service to the end user. The "client" side of the process typically resides on a workstation or computer. Requests from the client are routed as requests to the "server" that calls up and delivers the requested program. The server usually resides on a larger machine such as a mainframe and receives these requests. Because the server sits on a resource that has significantly more processing power, it is able to fulfill requests from multiple clients concurrently.[83] In essence, the networked computers and the file server that routed their requests were distributed systems that worked as if they were one system. This is illustrated in Exhibit 5.

Exhibit 5: Client/Server Architectures



Source: Franklin Templeton Industry Advisory Services. For illustrative purposes only.

As more business applications were added to the enterprise via both COTS systems and through native development to meet bespoke business needs, more internetworking technologies such as routers were also brought into the organization to facilitate the growing complexity of the architecture. An intermediate tier was added between the client and the server, called the application server or middleware. This middleware helped the more complex distributed system still appear as a single system to the user.[84]

The integration of technology into the business environment was institutionalized with the introduction of enterprise architecture. The business and the IT teams were now partners in delivering on the business's requirements and goals. This alignment was tested by the emergence of the internet and the need to offer websites as this created new challenges for both the business and technology organizations with regards to how they delivered information and services to their employees and customers.

Rise of the internet

The origins of the internet can be traced back to a series of research projects that the US Department of Defense commissioned as part of its Cold War efforts to combat the Soviet Union. In the immediate aftermath of World War II, there was a significant concern that the Soviet Union might launch a nuclear attack on the US homeland. To ensure that the US government had sufficient warning to respond to such an attack, the Department of Defense commissioned the construction of a computer network to gather and process data from multiple radar stations to monitor for potential air-based threats.

"It was the first networking initiative involving computers. In the end, the system was made up of 23 concrete-hardened bunkers scattered across the United States, each equipped with a SAGE computer and connected via dedicated network lines. Air surveillance data from many radar stations was processed to provide a uniform view of the airspace. This information was processed at SAGE facilities and information required to observe, exhibit, instruct and designate targets was provided to the Air Defense Command."

None of the large mainframes in existence at the time were capable of interpreting real-time data; they were not able to share information across multiple locations and there were no graphical interfaces. Together, MIT and IBM pursued more than a decade-long project to create the SAGE (Semi-Automatic Ground Environment) computer network.

It was the first networking initiative involving computers. In the end, the system was made up of 23 concrete-hardened bunkers scattered across the United States, each equipped with a SAGE computer and connected via dedicated network lines. Air surveillance data from many radar stations was processed to provide a uniform view of the airspace. This information was processed at SAGE facilities and information required to observe, exhibit, instruct and designate targets was provided to the Air Defense Command.[85] The SAGE project helped IBM design core memory, and it established a template for later work in GUIs. The SAGE system also became the foundation for other aero-based networks such as the airline reservation system SABRE and later, air traffic control systems."[86]

Work on SAGE was still ongoing when the Soviet Union launched the Sputnik satellite in 1957. This event touched off a frenzied race in the United States to catch up to and surpass the Soviets' scientific as well as defense capabilities. US President Dwight Eisenhower created the Defense Advanced Research Projects Agency (DARPA) to coordinate the US efforts around research and development, hoping to bring together the best scientific minds in the country to help American military technology stay ahead of US enemies.[87]

DARPA contracted with multiple leading research universities to help in this regard. To accelerate its work, one of the first projects it launched was an effort to connect various university research computers to share information more quickly, just as the SAGE work was meant to share defense information. Up until this time, scientists still had to send data in the form of printouts, which slowed down their ability to collaborate.[88]

In October 1962, J.C.R. Licklider of MIT was working as the head of DARPA's computer research program. Earlier, in August 1962, he had written a series of memos laying out an "Intergalactic Computer Network" concept that featured a “globally interconnected set of computers through which everyone could quickly access data and programs from any site.” Licklider convinced his successors at DARPA of the importance of this networking concept.[89]

After conducting some experimentation to ensure the viability of a decentralized network, a proposal to form the Advanced Research Projects Agency Network (ARPANET) was approved in the mid-1960s. One of the key enhancements that had to be invented was an ability to send packets of data. Up until this time, telecommunications relied on circuit switching, which meant that a dedicated circuit had to be opened between two points for the duration of the communication. In a packet-switching approach, data are divided into many small packets and any open channel can be used to send the packets. Each packet can take a different route because there is no end-to-end connection. The packets are put back together once they are received. This approach allowed computers to share channels and lessened the potential for a complete network failure because there are many possible routes available.[90]

New types of protocols had to be designed to make packet switching work. An interface messaging processor—equivalent to today’s routers—was required to ensure the receipt and full transmission of the packet. Another type of protocol was used to lay out the template for holding the message and addressing the packet so that it arrives at the right location. Finally, a third type of protocol was required to establish the connection between the two host computers.[91]

In 1969, the first successful use of the ARPANET occurred as computers at the University of California in Los Angeles, the University of California in Santa Barbara, the Stanford Research Institute and the University of Utah were networked and the first fully readable message—login—was delivered.[92] Over time, other applications were developed for the ARPANET that paved the way for the modern internet. In 1972, the file transfer protocol (FTP) was developed to allow the exchange of files between two host computers. In addition, a mail program was created that allowed users to send and receive text messages electronically. Finally, the “@” sign was established to distinguish the name of the user from the name of the server. Later, the ethernet, another network technology to link computers together, was also developed.[93]

After the creation of the ARPANET, other organizations began creating their own networks of computers that were incompatible with ARPANET and each other. Work on a universal language to pass packets of data between any two networked computers regardless of their hardware and software began.

“In 1983, the domain name system (DNS) was established, providing the .com, .edu, .gov, .net, .org system for naming websites. By 1987, there were more than 20,000 host computers on the fledgling internet.”

In 1974, the transmission control protocol (TCP) was developed. It put the data packets into a digital envelope where the address could be read by any computer, but only the final host machine could open the envelope and read the message inside. Later, a second protocol was developed called the Internet Protocol or IP. When combined with TCP, this second protocol helps internet traffic find its destination. Every device connected to the internet is given a unique IP number or address. In 1982, after the introduction of TCP/IP, the ARPANET grew to become a global interconnected network of networks, internetworking these networks or more simply referred to as the “internet.”[94]

In 1983, the domain name system (DNS) was established, providing the .com, .edu, .gov, .net, .org system for naming websites. By 1987, there were more than 20,000 host computers on the fledgling internet.[95] The real event that brought the potential of this ecosystem into the mainstream, however, was in 1990 when Tim Berners-Lee, a scientist at CERN, the European Organization for Nuclear Research, developed the three fundamental technologies that turned the internet into the web.

The first of the three fundamental technologies is the hypertext mark-up language (HTML). This is the formatting markup language of the web and the blueprint for a website. It tells the browser what is on the page in terms of text, links, and where to find images. The second technology is the uniform resource identifier (URL). This is a unique address used to identify each resource on the web. Finally, Berners-Lee invented the hypertext transfer protocol (HTTP), which allows for the retrieval of linked resources from across the web. After these inventions, Berners-Lee also wrote the first web page editor/browser that he called the WorldWideWeb.app. By the end of 1990, the first web page was served on the open internet, and in 1991, people outside of CERN were invited to join this new web community. In 1993, CERN agreed to make all the underlying codes available on a royalty-free basis. This unleashed a global cycle of creativity.[96]

The world wide web (WWW) has enabled people to be connected to each other instantly, 24/7, across the globe. The growth of websites that Berners-Lee's inventions enabled has expanded exponentially as shown in Exhibit 6.

When CERN opened its network to the world in 1993, there were 130 websites in existence. By 1995, that figure had climbed to 23,500, and by 2001—10 years after the first website was launched, there were over 29 million websites.[97] By 2021, there were over 1.7 billion websites across the world and 4.5 billion people contributed with online interactions each day.[98]

The impacts of the internet and the web upended business operations, changing behaviors inside and outside of organizations, and, even more profoundly, changing the demands on their IT infrastructure.

New digital companies were being formed to exclusively use the web to conduct business. In 1994, Mosaic Netscape—the dominant web browser of the time—introduced a security protocol called secure socket layers (SSL), which protected sensitive information transferred over the web.[99] This opened the way for online transactions.

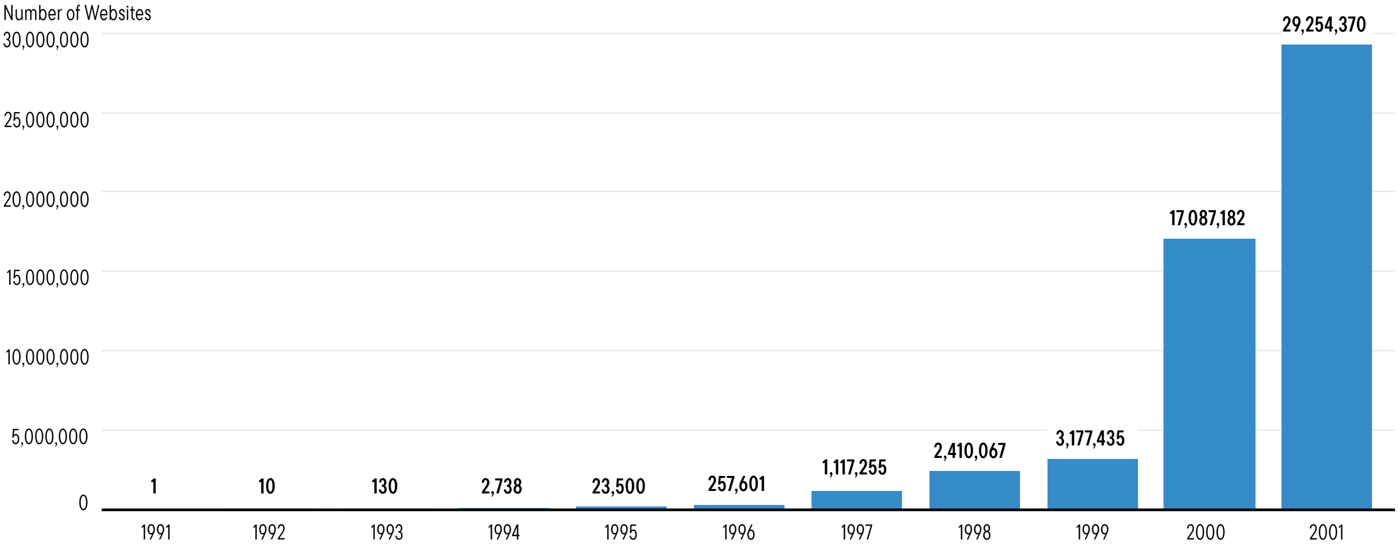
One of the first e-commerce sites was Amazon.com, which started in 1995 as an online bookstore. Traditional brick-and-mortar bookstores were limited to about 200,000 titles. Amazon launched, as without physical limitations, was able to offer exponentially more products to the shopper. It was one of the first online retailers to offer and add user reviews and a rating scale for its products.[100] This concept of letting customers publicly opine and *write* a review on Amazon's website regarding their shopping experiences was completely novel at the time.

Another 1995 launch was "AuctionWeb," a "site dedicated to bringing together buyers and sellers in an honest and open marketplace."[101] The site sold US$7.2 million worth of merchandise in 1996. By the following year, its sales soared with just one product—Beanie Babies, a Ty Warner line of cuddly stuffed animals—bringing in over US$500 million or 6% of the site's total volume. AuctionWeb also introduced its own seller feedback forum in 1997 that allowed members to rate their transactions and create a virtual community. Later that year, AuctionWeb was renamed eBay, and by September 1998, the company had gone public. Expected to make its IPO (initial public offering) at about US$18 per share, the company stock reached US$53.50 per share on opening day.[102]

The excessive IPO-day gains of eBay confirmed a trend that had begun earlier with another early internet star. In 1994, two Stanford University students, Jerry Yang and David Filo, created "Jerry's and David's Guide to the World Wide Web" as a web directory to help people find content. Links were manually curated by individuals that worked for the site called "surfers."[103] Later that year, the company changed its name to Yahoo!. It went public in April 1996, and although the IPO priced at US$13 per share, it opened its first day of trading at US$24.50 per share and closed the day at

Exhibit 6: Growth of Websites in the First Decade of the Internet

1991–2001



Source: Austin, Ben. "The growth of the Internet: from 1990 to 2019." Absolute web site. May 13, 2019.

US$33 per share, representing a market cap of US$848 million. Yahoo! at the time was the third-largest IPO in history. The stock peaked at US$118.75 per share in 2000, putting the company's market capitalization at US$125 billion.[104] One Yahoo! executive noted, "Our company was 5 years old, and we were worth more than Ford, Chrysler and GM combined. Hell, we were worth more than Disney, Viacom, and News Corp combined. Each of those great companies could have been swallowed up by us."[105]

This quote captures a sentiment that was pervasive in the second half of the 1990s. New internet startups were disrupting business models and attracting investor attention, causing many traditional companies to question their own value proposition. A rush by brick-and-mortar companies to establish a web presence ensued.

A new breed of internet consultancies emerged to facilitate this migration. *The Inside Consulting* newsletter, which tracked the growing set of providers, wrote that "they capitalized on a company's sense of inadequacy in the face of a major technology dislocation. Their selling point, basically, was YouSuck.com, and they had presentations that would scare the bejesus out of clients. They would say, 'It doesn't matter what industry you're in or where you're located, we can take you online.'"[106]

Challenges to early enterprise architecture approach

Getting legacy firms online from a technological perspective added more complexity to an IT infrastructure that was already growing complicated and unwieldy.

As early client-server architectures expanded, shortcomings became apparent. Upgrading or switching out applications became exceedingly difficult. Applications were either directly coupled point-to-point or they were integrated via the middleware. To make all the various components work together to deliver business workflows, IT groups had created many customizations and designed their own user interfaces that sat on top of their system architecture.[107]

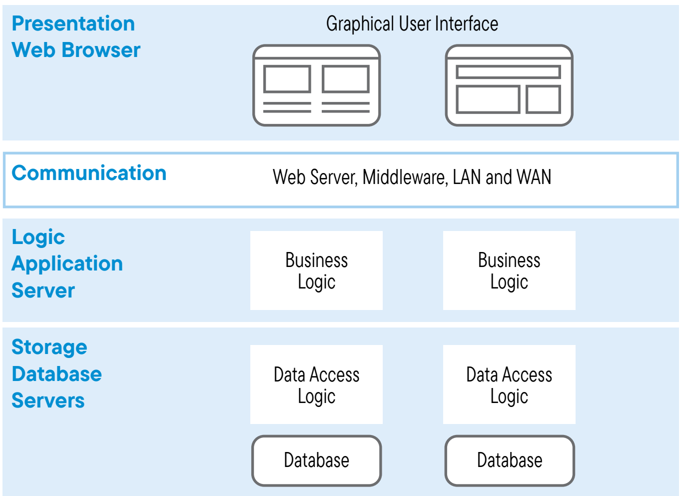
Because workflows were being created by linking many pieces together and each underlying application had its own internal data, a lot of data issues began to emerge. For example, the listing of a customer name may appear somewhat differently from system to system, and to make custom workflows effective, all these differences would have to be manually mapped and associated.[108] Adding web capabilities on top of this ecosystem created additional challenges.

New sets of business requirements emerged. Firms had to create new content to feed templated web pages and deliver that content through to the website on a timely basis, keeping materials fresh and updated. Organizations looking to offer online transactions needed to design the workflows and determine how to link new vendors offering online catalog templates, shopping carts and payment plug-ins to their internal inventory management, payment systems, and billing systems. Those launching intranets had to work out new single-sign-on options as they tried to link together multiple applications through a single web access point. New types of analytics looking at web metrics, such as click-throughs, time spent on each page, and other behavioral activities had to be designed and delivered into the website team to help inform updates and user-interface design decisions.

From an IT perspective, the enterprise architecture became more complex, moving from a three-tier system to a four-tier or *n*-tier system. This is illustrated in Exhibit 7.

The main changes in the new approach were that the business logic was now separated out from the client and housed in a new application server layer, and the middleware, WAN and LAN connections were combined into a new communications tier which also housed the web server. The client layer also became more complex, transforming into a GUI as the web browser—which is in effect a very thin client—was added. To make this more intuitive, the architectural layers became tagged with logical, not just physical, interpretations. The database server layer became the

Exhibit 7: 4- or N-Tier Client/Server Architecture



Source: Franklin Templeton Industry Advisory Services. For illustrative purposes only.

storage layer; the application server became the logic layer; the web server became the communication layer and web browser/client became the presentation layer.

Just adding the web server and web browser into the existing infrastructure was insufficient to solve the growing set of business challenges. "Digitalization meant that your 200 internal users are now joined by 200,000 external users who dare to expect to use your systems 24 hours a day and worse yet, they expect super response times," a leading provider of enterprise architecture consulting lamented.[109]

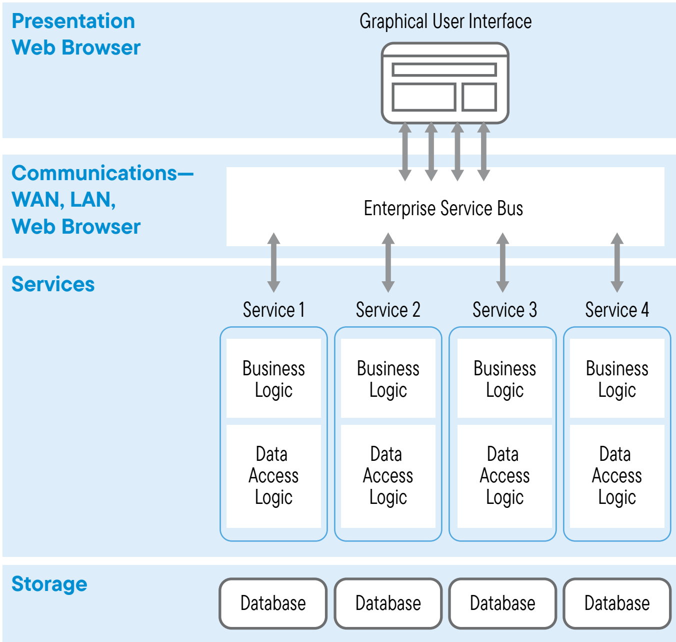
Foundational changes in the approach to IT enterprise architecture resulted. "Having data distributed and stitched together by a fundamentally asynchronous integration layer doesn't look so clever," the consultant concluded.[110] Companies started rebuilding their platforms around a concept called service-oriented architectures (SOAs).

This approach sought to redevelop applications as a collection of "services"—smaller transactions to enable better performance and containerized to deliver at the scale required. An IBM literature explains, "each service in an SOA embodies the code and data required to execute a complete, discrete business function (e.g., checking a customer's credit, calculating a monthly loan payment). Services use common interface standards and an architectural pattern that can be rapidly incorporated into new applications. This removes tasks from the application developer who previously had redeveloped or duplicated existing functionality or had to know how to connect to provide interoperability with existing functions."[111] The services can be written in a variety of programming languages and utilize web protocols for delivery, and they are published in a registry that enables developers to find and reuse them to assemble new applications or business processes.[112] This SOA approach is illustrated in Exhibit 8.

The ultimate benefit of the SOA approach was that "services can be built from scratch but are often created by exposing functions from legacy systems of record as service interfaces."[113] This enabled organizations to reimagine rather than replace embedded architectures.

By the early 2000s, these new architectures were beginning to be deployed, but another set of technology innovations were emerging that would force yet another shift in thinking and approach.

Exhibit 8: Service-Oriented Architecture



Source: Franklin Templeton Industry Advisory Services. For illustrative purposes only.

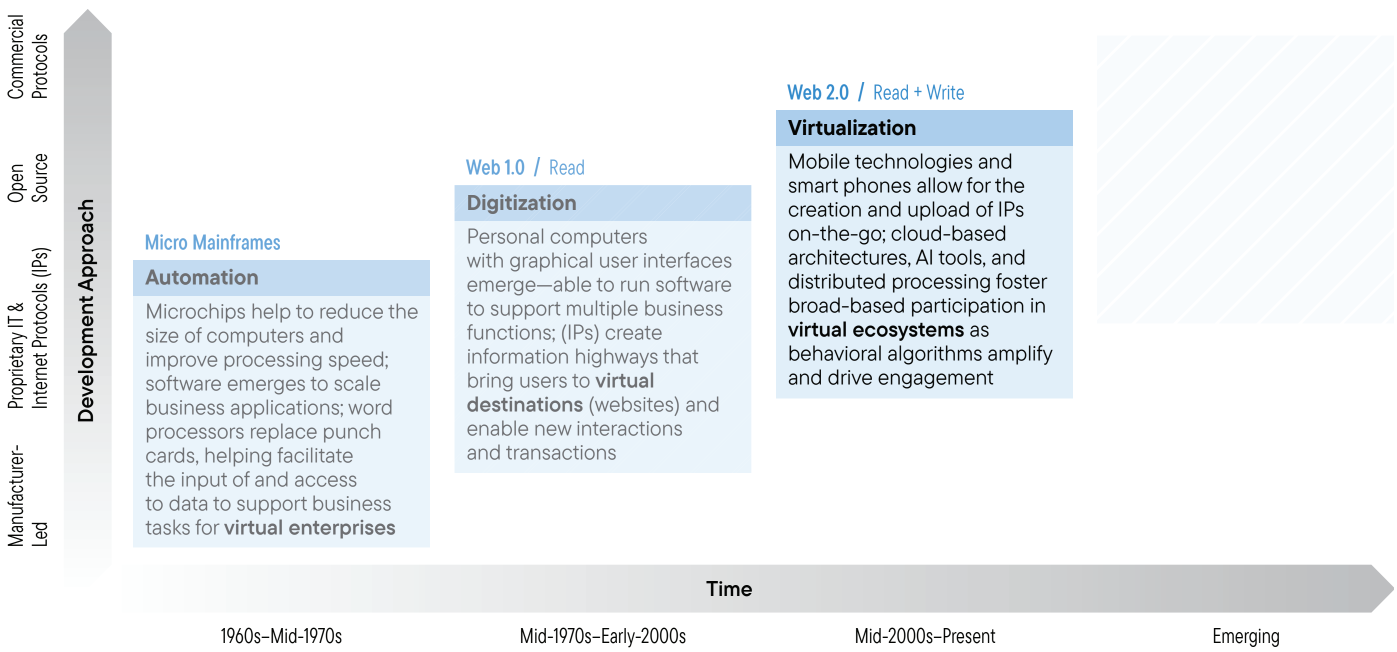
Section IV

Third cycle of commercial technology innovation—virtualization

Whereas the second cycle of commercial technology innovation extended the benefits of computing and digital engagement to individuals as well as to businesses, the third cycle of innovation made individual participation a lynchpin of the ecosystem. Work and digital engagement were freed from the physical need to be at a computer. Anytime/anywhere access to the internet became commonplace, and new devices allowed individuals to capture and integrate their daily experiences into their personal and business interactions. Novel business models emerged that relied on this ability to connect with others via tech-driven platforms and the underlying technologies used to enable this ecosystem underwent a foundational change.

Exhibit 9 shows the progression in technology approach and illustrates that the virtualization cycle, begun in the mid-2000s, is ongoing to the present day.

Exhibit 9: Modern Commercial Technology: Cycle 1, Cycle 2 and Cycle 3



Source: Franklin Templeton Industry Advisory Services. For illustrative purposes only.

To understand how profound this next cycle of virtualization has been for the technologies that we use to deliver industry and commerce, it is important to go back to the very beginnings of the commercial uses of computing.

Cloud computing and web services

The large mainframes of the 1950s were slow and unwieldy, making it difficult for more than one user and more than one program to run at a time. Each request had to be programmed uniquely and each program was run as a single batch. There were often long periods of time when a computer sat idle as the programming instructions were being input. Theories began to emerge that it would be more economical to have a group of users working at the same time on inputting programs so that the pauses from one user could be filled by running the program from another user. This was known as multi-programming. If done correctly, processing capacity could be optimized and there would be little to no downtime. The same concept would hold true for better utilizing storage and other computer resources.

By the 1960s, it was possible to test this concept as the "state" of each user and each program could be monitored in machines with newly deployed core memory and the order in which to run the program could be switched based on processing availability. A test run of this concept was completed as part of the compatible time-sharing system (CTSS) experiment at the MIT Computational Laboratory in 1961 using a modified version of one of the IBM 700 series computers to let two users "share" the same mainframe.[114] One of the researchers supporting that experiment postulated upon its completion that computing might one day be sold as a utility.[115]

One outcome of this experiment was that DARPA funded a new project with MIT called the project on mathematics and computation to explore a time-sharing approach that would make computing a utility available to anyone, anywhere—similar to telephone or electric utilities. MIT brought in GE and Bell Laboratories to partner on this venture. The result of this work was a system called Multics (multiplexed information and computing service), which was the most advanced time-sharing computer of its time but was plagued with errors. By 1969, Bell Labs had pulled out of the project and a group of Multics programmers went on to create its own project—Unix.[116] To its founders, the Unix project was equally about the community and collaborative culture as it was about the technology. As a result, the group opted to make its source code an open architecture—meaning any developer could access it and enhance it. This was the origin of the "open source" community.

Meanwhile, as Multics was still being explored, another pathway to sharing computer resources emerged. In the late 1960s, IBM began working on ways to create "virtual" machines that would allow users to share the mainframe. To accomplish this, IBM created a new type of two-part operating system where one part sat on the mainframe and created virtual machines and the other part acted as an operating system for each unique virtual machine, allowing it to interact with the mainframe.[117] It allowed multiple distinct computing environments to reside in one physical environment.

These activities—resource sharing and virtualization—were happening concurrently with ARPANET's first successful attempt to create an internetwork between host computers using protocols as opposed to fixed lines. Together, the concepts they introduced—(1) offering computing or storage as a utility, (2) allowing multiple people to share the same computer resources, and (3) accessing services via networking—are at the heart of cloud computing, the current driver of change that is transforming how we live, work and operate in the world.[118]

"These activities—resource sharing and virtualization—were happening concurrently with ARPANET's first successful attempt to create an internetwork between host computers using protocols as opposed to fixed lines. Together, the concepts they introduced—(1) offering computing or storage as a utility, (2) allowing multiple people to share the same computer resources, and (3) accessing services via networking—are at the heart of cloud computing, the current driver of change that is transforming how we live, work and operate in the world."

These ideas had been percolating in the computing world, but they required foundational technologies developed in the 1990s to make the vision a reality. The growth of the internet and the world wide web, the emergence and growing popularity of e-commerce, and the challenge to existing business models enabled by dot.com start-ups helped deliver the building blocks for the cloud. These innovations helped to establish a new, distributed version of the client-server model computing.[119]

Just as computer users could utilize their terminal to call up an application from a server back in the 1980s, new users could call up a remote application using their web browser via the internet by the mid-2000s as shown in Exhibit 10.

A key facilitator that allowed this to happen was the US government's Telecommunications Act of 1996. Up until this time, most users of the internet were connected by narrowband analog telephone lines that required a modem to translate digital computer data into analog data and a dial-up process to use the telephone system. New options enabled by the Act resulted in cable television companies upgrading fibers for two-way transmission, with the resulting broadband capabilities becoming available in 1998. In the telephone industry, digital subscriber lines (DSLs) were put on the market and internet service providers (ISPs) entered contracts with local phone companies to utilize these new and faster network lines. The result was that millions of personal and business computers began to be connected to these "always-on" lines.[120]

This technology shift allowed Marc Benioff—a sales executive at Oracle, which was one of the largest software companies in the world at the time—to enable his vision for Salesforce.com in 1999. The company was founded on the premise that software should be made available to the masses, on a 24/7 basis, over a global cloud computing infrastructure. This meant that no longer would companies have to bring in expensive software computer companies to install standalone computer platforms inside company walls. The company

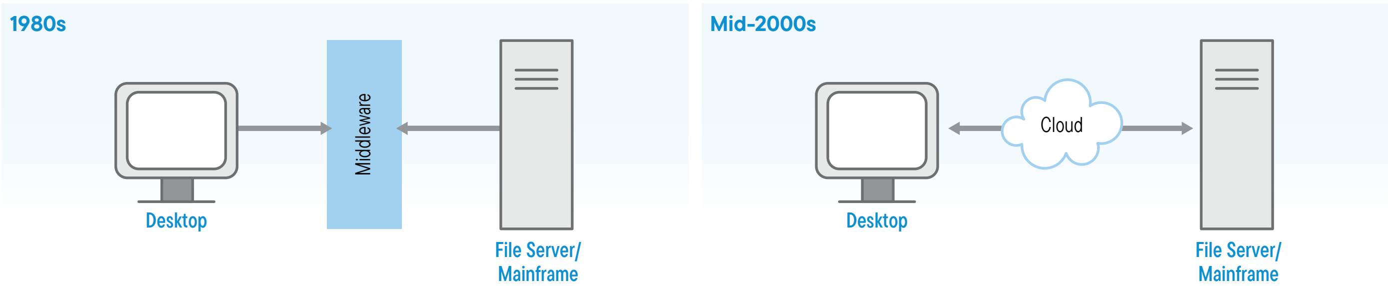
> **"Google used a unique approach to conduct internet searches—a proprietary algorithm program called PageRank that determined a website's relevance by considering the number of pages, along with the importance of pages, that linked back to the original site."**

built a customer relationship management tool and made the product available for companies to access virtually—or on the cloud. This launched the software-as-a-service (SaaS) model.[121]

Another innovator that emerged at the time that helped lay the foundation for distributed cloud services was Google. The company was launched in 1999 by Larry Page and Sergey Brin, two computer scientists from Stanford University. Google used a unique approach to conduct internet searches—a proprietary algorithm program called PageRank that determined a website's relevance by considering the number of pages, along with the importance of pages, that linked back to the original site. This contrasted with other web search engines of the time that ranked results based on how often a search term appeared on a page.[122]

To obtain the processing power required to run such extensive on-demand searches, Google runs on a distributed network of thousands of low-cost computer servers that use the idle capacity of always-on phone and cable lines. This allowed Google to "harness the power of multiple computational units...to collect data from widely dispersed locations... and perform enormous computations that simply cannot be done by a single CPU."[123] Google's model showed the power that could be drawn upon to facilitate on-demand computing rather than having dedicated processing abilities.

Exhibit 10: Return of Client/Server Architectures



Source: Franklin Templeton Industry Advisory Services. For illustrative purposes only.

“This is where Amazon steps into the story. In 2000, Amazon was a successful e-commerce platform looking for ways to scale. It wanted to build a business-to-business e-commerce platform called merchant.com to help third-party merchants like Target or Marks & Spencer build online shopping sites on top of Amazon’s e-commerce engine. Yet, like many startups, Amazon had not built an enterprise architecture meant to scale. It had started small and kept adding and expanding.”

While Salesforce.com proved the model of SaaS and Google proved the effectiveness of tapping into distributed processing to meet on-demand needs, the development ecosystem required to facilitate wholesale adoption of the new innovations had not yet been created.

This is where Amazon steps into the story. In 2000, Amazon was a successful e-commerce platform looking for ways to scale. It wanted to build a business-to-business e-commerce platform called merchant.com to help third-party merchants like Target or Marks & Spencer build online shopping sites on top of Amazon’s e-commerce engine. Yet, like many startups, Amazon had not built an enterprise architecture meant to scale. It had started small and kept adding and expanding. Instead of an organized development environment, AWS CEO Andy Jassy noted that they “had unknowingly created a jumbled mess.”[124]

The company began an effort to “untangle that mess into a set of well-documented APIs.”[125] An API is an application programming interface. It provides a way for computer programs to communicate with each other, acting essentially like a software interface offering a service to another piece of software. APIs are “essentially the pipes that connect servers, applications, and databases from different companies and allow them to talk to each other. Developers can use APIs to bring data into and out of their own platform, expand functionality without having to manually build every single feature, create software libraries for future use, and remotely control different protocols and technologies.”[126]

Upon creating its APIs, Amazon told its developers to start building internal applications using this decoupled, API-access approach, believing that it would allow it to consume services from its peer internal development teams and avoid reworking. Yet, after a time, the company realized that the expected increase in productivity was not showing up, even as the company was hiring more and more software engineers. The executive team dug into the problem and found that each project team was still building its own resources for each individual project—the database, the computing, and the storage components—with no thought to scale or reuse. Jassy noted that Amazon required a set of “common infrastructure services everyone could access without reinventing the wheel every time, and that’s what they set out to build.”[127]

In 2002, Amazon launched its first version of Amazon Web Services (AWS)—a developer toolkit that offered free SOAP (simple object access protocol) and XML (extensible markup language) interfaces to the Amazon product catalog. This offering had a limited set of services, basically enabling third-party sites to search and display products from the Amazon.com website and enable visitors to the site to add items to their Amazon.com shopping carts.[128]

In 2003, during a retreat held at Jeff Bezos’s house, Amazon’s executives performed an exercise designed to identify the company’s core competencies. Among these, they listed their ability to run infrastructure services like compute, storage and database, and in addition, they had become adept at running reliable, scalable, cost-effective data centers to serve their e-commerce platform. Jassy finishes the anecdote by noting that the team “began to wonder if they had an additional business providing infrastructure services to developers.”[129]

As the concept evolved, the Amazon team began to think of this set of services as “an operating system of sorts for the internet.” The team believed that companies would build applications from scratch on top of the infrastructure services if the right selection of services existed. Jassy noted that “we realized we could contribute all of those components of that internet operating system, and with that we went to pursue a much broader mission, which is AWS today, which is really to allow any organization or company or any developer to run their technology applications on top of our technology infrastructure platform.”[130]

The company launched AWS Elastic Compute Cloud (EC2) in August 2006. By the second quarter of 2022, AWS generated $$19.7 billion in revenue for Amazon and accounted for 17.4% of Amazon's total revenues.[131]

AWS, and similar development environments that launched later, such as Microsoft's Azure and Google's Cloud, allowed for a new model of enterprise architecture to emerge—cloud-based architectures that allow for a microservices approach and where developers and organizations pay only for the resources they use.

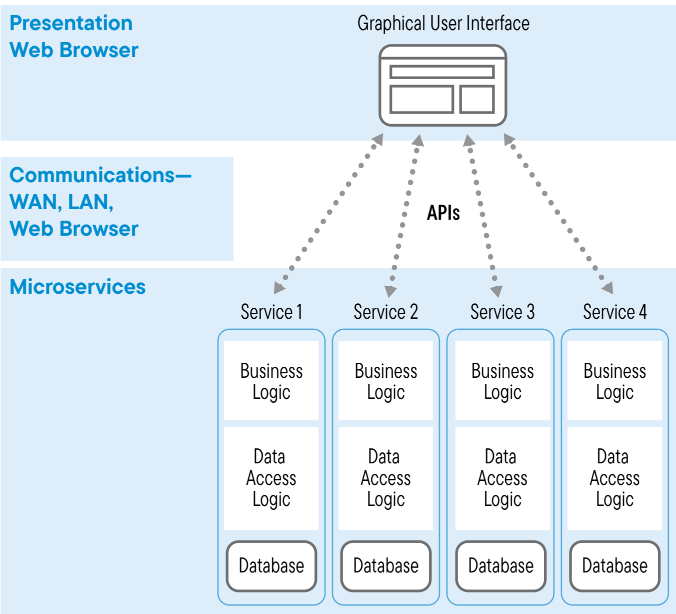
Microservices architecture

In many ways, microservice architectures and the SOA discussed earlier are very similar. Both are made up of loosely coupled, reusable and specialized components that work independently of each other. Both combine the code and the data required to execute a complete business function. The main distinction between the two comes down to the scope. SOAs have an enterprise scope, whereas microservices architecture works at the application level.[132] This is shown in Exhibit 11.

Because of this difference in scope, there are differences in how the two architectures are applied.

- **Reuse:** In SOA, the goal is to increase the enterprise's scalability and efficiency through the reuse of services. In a microservices architecture, reuse creates dependencies that reduce agility and resilience. Microservices components generally prefer to copy code and accept data duplication to improve decoupling.[133]
- **Data duplication:** The aim in providing services in an SOA is for all applications to obtain and make updates to data directly from the primary source, which reduces the need to update all the various systems that may utilize the data. The goal is to keep the data synchronized. In a microservices architecture, each service should have local access to all the data it needs to ensure its independence from other services and other applications.[134]
- **Communication**: To make services work together in an SOA, there is typically a piece of communication technology called an enterprise service bus that manages and coordinates service delivery. In a microservices architecture, each service is developed independently and communicates via APIs. Thus, microservices are a cloud-native architectural approach.[135]

Exhibit 11: Microservices Architecture



Source: Franklin Templeton Industry Advisory Services. For illustrative purposes only.

In essence, microservice architectures are what allow organizations to stop building and running as many proprietary, in-house applications, and purchasing as much computing infrastructure and data storage. Instead, organizations can use services like AWS, Azure or Google Cloud to develop and host their applications by leveraging cloud platforms. The main benefit is that the organization can move to on-demand delivery of IT resources and adopt a pay-as-you-go model. The elements of these cloud platforms include the compute, storage, database and networking functions required for software development and the hardware required to host it via infrastructure-as-a-service models.

Together, the microservice architectural approach and on-demand cloud platforms deliver on the vision of the MIT researcher who postulated back in 1961 that computing may one day be sold as a utility. Moreover, having this powerful processing backbone has been instrumental in enabling the growth of social media and the shift from Web1 to Web2.

The shift from Web1 to Web2

Web2 is the second stage of internet development that involves the evolution from basic, static web pages to increasingly dynamic pages with user-generated content. Web2 can be viewed from a strictly technological perspective as a new set of innovations that emerged to enhance the quality of web design and delivery and enable users to not just read from the web, but to also write to the web.

The simplicity of that formulation underplays the importance of the shift, however. Web2 is about more than the technology and the structural changes to web design—the societal impacts are equally, if not more, important to consider. For the first time, humans were not just using the internet as a tool, they were transposing human societies into this network of computers. It changed the nature of the web from what we did to how we lived. Exhibit 12 illustrates the power of this statement.

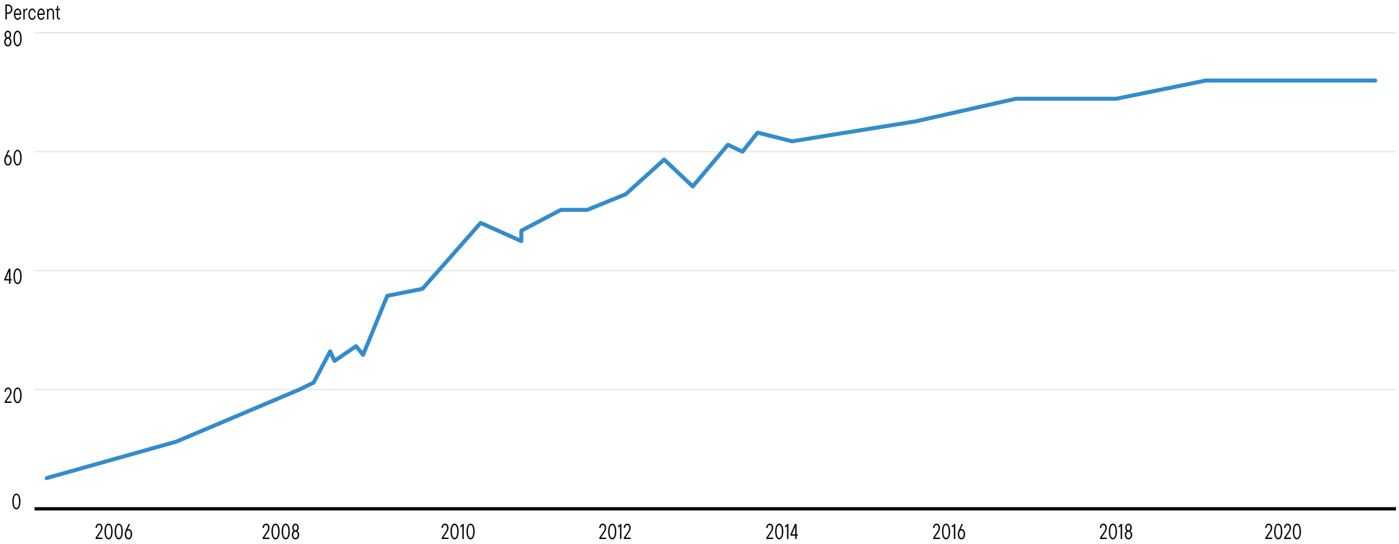
According to a 2021 Pew Research Center survey, in 2005, only 5% of Americans used at least one social media site, and by 2021, that figure was up to 72%. This shift in behavior has been transformational, as will be discussed in Part II of this series, but in sticking with our exploration of how technology has enabled this change, it is important to understand what new capabilities have created the social media phenomenon.

The most visible differences to the user between Web1 and Web2 are the presentation of the website and how users interact with the offering.

- **Site structure:** Most websites were constructed with a hierarchical structure—a front page leading to various sub-pages, augmented by cross-links and search functions. Web2 platforms tend to be user-centric with more customizable and personalized views such that the presentation of the site will appear differently from user to user. Users have their own display page based on their log-in. Sub-pages on the site are created by standard navigational links, user-defined links and system-suggested links, meaning that the website experience is no longer standardized, but user-centric.[136]
- **Nature of a "page":** In Web1 sites, content is centrally updated at somewhat predictable intervals and individual authorized users edit the site at differing frequencies. These sites are typically a single-writer user medium. Web2 offers dynamically generated pages from multiple sources of information. It is in shared and communal spaces where changes to content can be initiated by many contributors. It is "live" in the sense that it can be updated while a user is examining it. Web2 pages are also a broader mixture of audio, video, text and images.[137]
- **Access:** Web1 sites offered links to external sites that users could easily follow. Most sites tended to cover a single topic and did not require users to log on to access them. Web2 sites promote intrasite activities often requiring users to log on and build links to others on the site. Users are encouraged to create an account to engage with the site more fully. Navigation links are often directed solely within the site and external links may be difficult or impossible to add.[138]
- **Organization:** Web1 content was provided by a server's file system, where the tagging of the content to the page was templated. Web2 uses a relational database management system and a "folksonomy" approach to organization—a user-generated system of classifying and organizing online content using metadata (tags).[139]

Exhibit 12: % of US Adults Who Say They Use At Least One Social Media Site

2006–2020



Note: Respondents who did not give an answer are not shown.
Source: Social Media Fact Sheet. Pew Research Center. April 2021.

The shift from Web1 to Web2 has been supported by a significant improvement in the scripting and presentation technologies used to render the website and enable user interaction.

Foremost is the use of asynchronous JavaScript and XML (AJAX). AJAX is "a mixture of several technologies that integrate web page presentation, interactive data exchanges between the client and server, client-side scripts, and asynchronous updates of the server response. AJAX can be used for dynamic layout and reformatting of a web page, reduce the amount of reloading needed by sending a request for just a small portion of a page, and interact on-demand with the server."[140]

Supporting AJAX in making the web-page interactive is the document object model (DOM). This is an "interface that represents how the HTML and XML documents within the web page are read by the browser. It allows the JavaScript to manipulate, structure and style the website. DOM allows programmers to create applications that update the data on the page without needing a refresh; create applications that are customizable by the user; change the layout ofthe page without a refresh; and drag, move or delete page elements."[141]

In laypersons' terms, these technologies allow the website to communicate with users' web browsers behind the scenes without human interaction, which means that users do not need to click something for the website to respond.

Another set of technologies allow for multimedia—videos, audio files—to be embedded into web pages and to stream—play without requiring special plug-ins. In the early days of Web2, this was accomplished via Adobe Flash, video and music players, but over time, these applications gave way to HTML5.[142] The use of APIs to "mash up" content and display data in visually powerful ways or create a new service by combining capabilities from multiple services is also an important characteristic of Web2. Two examples are the ability of businesses to embed their Google map location in their advertisements and the ability of Uber users to play their Spotify playlists during their rides.

The presentation opportunities and technology innovations that define Web2, together with the processing and power of cloud computing, have created the platform to showcase user-driven content, but the most important technology that has enabled social media to develop and thrive has been the rise of mobile devices and smartphones.

"The launch of the iPhone by Apple in 2007 was a pivotal moment in this journey. The iPhone was revolutionary in its move away from a keyboard-based to a touchscreen approach and it set a new design standard with its superior graphical display and sleek casing, but many of the innovations that came together in the iPhone had been pioneered well before 2007."

Mobile devices and smartphones

Equally as transformative as the internet has been the rise of mobile computing and devices. Mobile has freed users from having to be at a computer terminal and allowed information and services to be accessible anytime, anywhere. Moreover, the utility of mobile devices has expanded to incorporate creative elements such as cameras and videorecording and to enable the download of applications such as word processing, video and picture editing to capture the users' thoughts and adjust their images. As a result, the mobile device has become a means through which users can both interact with and chronicle the world around them. This has been the fuel that has driven the rise of Web2. The following statistic helps to underscore just how transformational the emergence of mobile has been—in 1990, there were 12 million mobile users globally.[143] By 2020, that figure had risen to 5.22 billion.[144]

The launch of the iPhone by Apple in 2007 was a pivotal moment in this journey. The iPhone was revolutionary in its move away from a keyboard-based to a touchscreen approach and it set a new design standard with its superior graphical display and sleek casing, but many of the innovations that came together in the iPhone had been pioneered well before 2007. Exhibit 13 on the next page provides a timeline of the early history and key milestones in the development of cellphones up until Apple's iPhone launch.

The first portable cellphone was invented by Motorola in 1973. The prototype weighed 2.4 pounds (1.1 kilograms) and measured 9.1 x 5.1 x 1.8 inches (23 x 13 x 4.5 cm). It offered a talk time of only 30 minutes and required 10 hours to recharge. The phone was priced at US$3,995 or the equivalent of US$10,000 in today's terms.[145] This device and the following generation of cellular phones were collectively

Exhibit 13: Milestones in the Development of Cellphones Up to the Release of the iPhone

1970s

In 1979 Nippon Telegraph and Telephone introduces 1G to citizens of Tokyo

1973
First portable brick style cellphone invented by Motorola—prototype weighed 2.4 lbs. (1.1kg) and measured 9.1 x 5.1 x 1.8 inches (23 x 13 x 4.5 cm) and had 30 minutes of talk time, required 10 hours to recharge and was priced at US$3,995 (equivalent to US$10,000 today

1980s

Ameritech introduces 1G in 1983 to the United States & Canada

1989
Motorola introduces first flip phone, MicroTAC, allowing phones to be more mobile, than brick style phones, because of their bulk, were usually kept in cars

1990s–2000s

Europe agrees to global standards (IMT2000) and 2G launches on the Global System for Mobile Communications (GSM) in Finland in 1991—the first commercially operated digital cellular system

NTT DoCoMo deploys 3G in 2001 for the public in Japan and later that year, Manx Telecom released a trial 3G service for Europe

1992
- Nokia produces first mass-produced GSM (2G) phone, Nokia 1011
- Vodafone developer sends first text message

1994
IBM releases Simon, considered by many as the first smartphone, with apps and a touchscreen

1996
- Nokia releases Communicator 9000 with a QWERTY keyboard and access to email, web browsing, fax, word processing and spreadsheets
- Motorola released StarTAC, the first clamshell phone and the first to introduce vibration

1999
- Motorola Timeport is the first tri-band GSM phone able to work globally
- Samsung Uproar combines MP3 into phone to play music
- First camera phone released by Kyocera
- First GPS (global positioning system)- enabled phone from Benefon
- Emojis created

2001
- Ericsson T39 is first cell phone with Bluetooth
- Nokia 8310 is first phone with calendar feature

2002
- BlackBerry releases its first smart phone targeted at the business market
- Sanyo SCP-5300 introduces the first phone camera with flash that allows photos to be seen on screen rather than needing them to be downloaded to a computer

2007
First iPhone, iPhone 2G is launched, removing physical hardware buttons and replacing them with a touch-based interface

Source: Franklin Templeton Industry Advisory Services analysis based on Križanović, Ivana, "Cell phone history: From the first phone to today's smartphone wonders," versus web site, December 2, 2021; and Galazzo, Richard, "Timeline from 1G to 5G: A Brief History on Cell Phones," CENGN web site, January 24, 2022. For illustrative purposes only.

called brick phones because of their bulky appearance. Most were used in cars, where they could be kept plugged into the car's electrical system to charge.

Use of these phones was limited. After being introduced in Japan in 1979 and growing over the following years, the 1G cellular network was available by the mid-1980s across much of the developed world, but the "coverage was shoddy, with large amounts of static noise and background crackling. No roaming support was provided. Security didn't exist… there was no encryption, meaning anybody with a radio scanner could drop in on a call. Download speed…was also incredibly slow and only 2.4 kilobits per second (Kbps)."[146]

The launch of the 2G network in 1991 significantly improved mobile, allowing for encrypted calls, improved sound quality and significantly faster download speeds at 0.2 megabits per second (Mbps). 2G networks also allowed for the transfer of data from one phone to another, enabling access to media content such as ringtones and allowing for the introduction of short message service (SMS) texting and multimedia messages (MMS) as new forms of communications. Using the same control channels as talk, SMS and MMS messages are sent in packets of data from a transmitting cell phone to a tower and then to a receiving phone.[147]

The launch of 2G occurred in Finland and was the culmination of a long period of European planning to create a global standard for mobile communications. The 1G network had rolled out in a haphazard manner starting in Japan and then spreading to other countries over the course of many years. The goal of the Europeans was to create a mobile system able to work interoperator—across the various telecommunication companies—and internationally in terms of roaming. Three work initiatives came together to enable this effort.[148]

- A working group got mobile services added as a new band (900 megahertz, or MHz) in the International Table of Frequency Allocations and dedicated a total capacity of 1,000 channels for new civil mobile use, thus paving the way for the frequency spectrum for 2G.
- A working group was formed to harmonize the technical and operational characteristics of a public mobile communications system in the 900-MHz band.
- A team was established to define the standards and technical specifications for a global system of mobile (GSM) communications technology.

GSM became the predominant 2G technology that swept through most parts of the world and went on to serve 80% of the mobile market in decades to come.[149]

With a better network to utilize, innovations in cellphone design accelerated. Motorola released the first flip-phone in 1989, moving away from the brick design and making the phone truly mobile. In 1992, Nokia released the Nokia 1011, which was the first mass-produced phone for the 2G network and a Vodafone developer sent the first text message—"Merry Christmas!"—to a company director at the office's Christmas party.[150]

In 1994, IBM released the Simon, its version of a personal digital assistant (PDA) that is also acknowledged as the first smarthphone. In addition to being able to make and receive cellular phone calls, Simon could also send and receive faxes, e-mails and documents. It offered a series of business applications including a scheduler, calculator, world time clock, electronic notepad and a stylus that could be used with an input screen keyboard.[151] It only sold 50,000 units, and it did not offer internet browsing.[152]

Nokia introduced the next PDA generation in 1996. This offering featured a QWERTY keyboard and offered email and fax capabilities as well as web browsing, word processing and spreadsheets.

Meanwhile, the cellphone itself went through a series of design enhancements. Motorola introduced the StarTac—the first clam-shell design phone that also introduced vibration in 1996. Samsung combined an MP3 player to offer music in its Uproar cellphone offering in 1999, while Kyocera released the first phone with a camera that year, and the first global positioning system (GPS)-enabled phone was released.[153] Emojis were invented in 1999 by Shigetaka Kurita in conjunction with the launch of the mobile internet in Japan.[154]

Soon thereafter, the 3G network launched, setting the stage for the rise of social media. 3G had four times the data transfer capabilities of the 2G network, allowing transfers of up to initially two Mbps and later up to six Mbps, which in turn made video streaming, video conferences and live video chat possible at scale. 3G allowed users to listen to music, call, text and search the internet easily and effectively from their mobile devices. It also allowed for the inclusion of Bluetooth—a short-range wireless technology—into cellphones.[155]

Greater amounts of data bandwidth allowed for the first widely adopted smartphone targeted at the business market.

> **"All of these innovations came together and were improved with Apple's iPhone. Apple sold more than 100 million units in the iPhone's first four years. Described as having the functionality of a mobile phone, game console, an iPod and a handheld computer all in a single device, it found widespread appeal. The novelty of the physical design of the iPhone was of particular importance at the time."**

The BlackBerry RIM series was released in 2002 and went on to become a ubiquitous accessory for business executives throughout most of the 2000s.

In addition to better data processing abilities, the rise of 3G was paralleled by a rapid improvement in the camera capabilities offered in cellphones. Sanyo launched the SCP-5300 in 2002, the first camera phone with a flash that allowed photos to be seen on screen rather than needing to be downloaded to a computer.[156]

All of these innovations came together and were improved with Apple's iPhone. Apple sold more than 100 million units in the iPhone's first four years.[157] Described as having the functionality of a mobile phone, game console, an iPod and a handheld computer all in a single device, it found widespread appeal. The novelty of the physical design of the iPhone was of particular importance at the time. It had only one single physical button that displayed the device's main menu with the rest of the device consisting of a large, touch-sensitive screen. Though this design template is now commonplace, it was quite revolutionary at the time.

The tools of creation were now in users' hands. Consumer mobility allowed for new, simpler methods of interacting. This allowed social media and its resulting explosion in digital content to surge.

The rise of social media

Though many associate the rise of social media to companies that emerged in the early 2000s, the template for such platforms was really established in the 1980s and 1990s through online communication services such as CompuServe,

“CompuServe originated the model. It was a subscriber-based online service marketed through Radio Shack electronics stores under the name MicroNET. In 1981, the cost of access via the dial-up phone lines was US$5.00 per hour during non-peak hours (6:00 p.m. to 5:00 a.m. on weekdays, weekends and holidays) and US$22.50 per hour on weekdays from 5:00 a.m. to 6:00 p.m. It offered news aggregated from newspapers (with CompuServe publishing online editions), as well as electronic mail, bulletin boards, educational programs, financial programs, securities information and games.”

America Online (AOL) and Prodigy. Each of these firms began to build communities of users by offering multifunctional digital gathering places.

CompuServe originated the model. It was a subscriber-based online service marketed through Radio Shack electronics stores under the name MicroNET. In 1981, the cost of access via the dial-up phone lines was US$5.00 per hour during non-peak hours (6:00 p.m. to 5:00 a.m. on weekdays, weekends and holidays) and US$22.50 per hour on weekdays from 5:00 a.m. to 6:00 p.m. It offered news aggregated from newspapers (with CompuServe publishing online editions), as well as electronic mail, bulletin boards, educational programs, financial programs, securities information and games. The real-time chat program, CB Simulator, had 40 channels, including many special interest groups and topical forums. Prior to the internet, CompuServe users could only email other CompuServe users. In early 1981, CompuServe had 10,000 subscribers. By 1987 that figure had jumped to 380,000, and by the mid-1990s, the service had millions of subscribers.[158]

Another early entrant was Prodigy. In the early 1980s, an experiment in shopping and on-demand news delivery using television set-top boxes led three corporations to launch a new online service called Trintex in 1984. This joint venture between IBM, Sears and CBS looked like a consumer product (as opposed to CompuServe's ASCII, text-based interface). It had colorful graphical interfaces and was designed to provide a simple customer experience. Within 4 years, CBS had dropped out and the service was renamed Prodigy.[159]

Prodigy aggregated a wide array of services, including news, weather, syndicated columnists, ESPN sports, games, and shopping services ranging from groceries to airline reservations. It had a flat-rate pricing model offering tiered services starting at as little as US$9.95/month. Prodigy grew from 100,000 to 500,000 subscribers in the first year, and then doubled to almost 1 million by 1991. It suffered major attrition, however, as it began to increase its monthly subscription rates and when it began to charge for previously free services such as chat and email. In 1993, it transitioned its member access from a closed, proprietary computer software portal and expanded to include internet access with a web browser. Prodigy ultimately morphed into an ISP before ultimately being wound down by its new owners in 2002.[160]

AOL began in the mid-1980s as a gaming site for Commodore computer users under the name Quantum Link. In this incarnation, the company provided dial-up access to gaming servers and pioneered the massive multiplayer online game *World of Warcraft*. In 1985, it introduced graphical chat environments and online interactive serial fiction. These features attracted the interest of Apple and the electronics retailer Tandy. Joint agreements with these companies resulted in branded services using Mac and PC client software—AppleLink and PC Link, both of which launched in 1988. When Apple pulled out of their joint venture, the Quantum Link rebranded its service as America Online, which then got abbreviated down to AOL.[161]

AOL was the first company to mass distribute free software, and allow users to create their own chat room spaces rather than joining into pre-existing communities. In 1993, AOL opened access to its newsgroups and followed up with the ability to send email from AOL addresses to the internet at large—an innovation since all earlier providers only allowed email to other members within their community.[162]

AOL initially tried to remain a “walled garden” in the early 1990s, developing its own content and allowing only limited access to the internet through its software, but by 1995 it launched both a web browser and an aggressive campaign to sign on users—bulk-mailing trial offers on floppy disks and then compact disks to the general populace and paying to have them stacked as free giveaways near the checkout lines in supermarkets. It even ran AOL keywords to encourage users to join its forums on popular daytime shows such as Oprah Winfrey's *Oprah!*[163]

At its peak in 2002, AOL had more than 26 million subscribers. It had become much more of a dial-up ISP by this time rather than an active community, however. Broadband ate into AOL's dial-up business and its subscriber base fell away. Purchased by Time Warner, AOL was de-emphasized after the dot.com bubble burst, and it gradually faded away. Its subscriber base was down to only six million users worldwide by the beginning of 2009.[164]

These early online community ventures offered some important lessons for companies that followed. A professor specializing in sustainable social communities noted that social networking sites needed to reward their members at a fundamental level. "People need to believe that they will obtain some return on their investment of time and energy. Remuneration does not need to be financial; it needs to satisfy some basic, psychological or emotional need."[165]

Other key lessons were that the sites allowed users to "own their own words." What users published and posted was seen as their own self-expression and, hence, their own intellectual property. Tensions around how much to moderate forums were also evident from the outset. As early as 1996, there was an "AOL Sucks" campaign looking to lambast the company for revoking users' credentials for publishing certain words in public chat rooms. The final lesson was that customers want to know upfront the costs associated with the platform and that higher charges may inspire members to look elsewhere.[166]

"In 2003, Myspace launched. By 2006, it was the most visited website on the planet with more than 100 million users, spurred by the ability for users to share new music directly on their profile pages. The site was sold in 2005, however, to News Corp. The company did little to enhance and improve the technology platform and instead seemed to focus primarily on maximizing advertising revenues. By 2008, another social media platform, Facebook, began to overtake Myspace first in the global market and then in the US market."

While CompuServe, Prodigy and AOL all allowed users to participate in online communities, they did not make the site to be about their actual users. In 1997, a website called Six Degrees launched a profile-uploading service, and though it had amassed 3.5 million users by 1999, it was sold and shut down a year later.[167] This was followed in 2001 by Friendster. These rudimentary platforms attracted millions of users and enabled email address registration and basic online networking. LinkedIn, founded in 2002 as a networking site for career-minded professionals, followed this same simple profile and networking template. By 2020, LinkedIn had grown to more than 675 million users.[168]

In 2003, Myspace launched. By 2006, it was the most visited website on the planet with more than 100 million users, spurred by the ability for users to share new music directly on their profile pages.[169] The site was sold in 2005, however, to News Corp. The company did little to enhance and improve the technology platform and instead seemed to focus primarily on maximizing advertising revenues. By 2008, another social media platform, Facebook, began to overtake Myspace first in the global market and then in the US market.[170]

Mark Zuckerberg, the founder of Facebook, began programming when he was 11 years old. By the time he was in high school at Phillips Exeter Academy, he had created a piece of software called Synapse that learned a user's music taste through AI and listening habits. This caused both AOL and Microsoft to approach him about potential jobs. Instead, he went to Harvard University, beginning as a freshman in 2002. By October 2003, he had created a campus-wide controversy by launching a website called Facemash that displayed side-by-side photos of two Harvard students and let others judge their attractiveness, tracking the various voting rounds to create a schoolwide ranking. Within a few days, the site was pulled down, since no one had given permission to use their photos, but Facemash had still pulled in over 22,000 views in that period. After publicly apologizing, Zuckerberg began work on his next project—Thefacebook.[171]

In February 2004, this project launched. It had a profile where a user could upload a photo, share interests and connect with other people. It also included novel features that quickly differentiated it and allowed it to satisfy that "emotional need"

> **“It (Facebook) opened its platform to high school students and to the employees of Microsoft and Apple. By December, international schools were invited to join. It ended 2005 with 2,500 colleges and 25,000 high schools on the platform. In September 2006, the site was fully global and had opened to anyone over 13 with a valid email address. By December 2006, it had 12 million users.”**

that was critical to social networking sites. Users could alert each other to parties they were having, show their relationship status, see a visual display of their friends' networks, let viewers know if they were connected or away, and choose to keep some aspects of their profile private. Zuckerberg and his co-founders kept the membership exclusive to add to the site's appeal. At first, it was only open to people with a Harvard email address. Within the first month, 50% of the college's students had signed up. By the end of 2004, membership was open to nearly all universities in the United States and Canada, and there was a growing surge of people wanting to sign up.[172]

By 2005, the company dropped the “The” from the name, becoming solely Facebook. It had also moved its operations to Palo Alto, California, and gotten investment capital from leading venture capitalists and successful entrepreneurs. It opened its platform to high school students and to the employees of Microsoft and Apple. By December, international schools were invited to join. It ended 2005 with 2,500 colleges and 25,000 high schools on the platform. In September 2006, the site was fully global and had opened to anyone over 13 with a valid email address. By December 2006, it had 12 million users.[173]

In 2007–2008, Facebook took several steps that transformed the company from a social networking site to a social media ecosystem. It opened its Marketplace, which let users post classified ads to sell products and services; launched the Facebook Application Developer platform, enabling developers to create their own applications and games that integrated with Facebook; and enabled pathways for businesses to use the site, opening Pages for Businesses.

By the end of 2007, more than 100,000 companies had signed up. It also continued to add functionalities, introducing Facebook chat, People You May Know, Facebook Wall and Facebook Connect.

By the end of 2009, Facebook had 350 million registered users and 132 million unique monthly users, officially overtaking Myspace and becoming the most popular social media platform in the world. By November 2010, Facebook was valued at US$41 billion and had become the third largest web company in the United States, just behind Google and Amazon.[174]

While Facebook was building its web-based social media empire, another start-up venture opted to use the SMS text messaging platform to build out a social communications powerhouse—an unusual choice since mobile carriers at the time imposed a 140-character limit with the SMS protocol.

In 2006, Jack Dorsey was working at a podcasting platform, Odeo. On the side, he was working on an idea for a communications platform where groups of friends could keep tabs on what each other were doing based on their status updates. He proposed this idea to the Odeo co-founders, who gave him the go-ahead to develop the project further.

In March 2006, Dorsey sent his first message to internal Odeo employees, “just setting up my twttr.” The service was launched publicly in July 2006 and the name soon morphed into Twitter. They dubbed their business model “microblogging,” and it was such a novel concept that co-founder Biz Stone needed to create a funny YouTube video to explain it.[175]

In October 2006, Dorsey, Odeo founders Biz Stone and Evan Williams, and several other employees formed Obvious Corporation and bought Odeo out from the investors and other shareholders, paving the way to incorporate as Twitter.[176]

In 2007, at the South by Southwest Interactive conference (SXSW), Twitter as a company took off. More than 60,000 tweets were sent per day during the conference. Twitter had a huge presence at the event, displaying Twitter messages on massive plasma screens at the festival to build excitement for the service and targeting the journalists, media and bloggers that attended.[177] The growth from there was exponential. In November 2008, Twitter reached one billion tweets; by October 2009, that figure was up to five billion. By early 2010, more than 50 million Tweets were being sent each day. By early 2011, more than one billion tweets were being sent each week.[178]

A somewhat unusual feature of the platform was that it allowed users to devise their own enhancements that later became company-adopted features rolled out to all users. In the early days of the platform, users had no way of replying to one another on Twitter. Some users began including an "@" symbol before a user's name to identify another user within a tweet. This became so prevalent that Twitter added the functionality to the platform. Similarly, users wanted a way to repost a message from a Twitter user while including credit to the user who originally tweeted it. Users started to add RT—retweet—before sending the message, another enhancement adopted by the company.[179]

Other social media powerhouses then began to emerge, all built around users abilities to post and share photos and videos that they had created.

- Instagram was launched in October 2010, offering a photo-sharing service with an ability to organize photos with location information and hashtags to classify them, apply filters, follow other creators, and comment on and "like" other people's photos. The app attracted 25,000 users in its first day and had been downloaded 100,000 times by the end of its first week, reaching one million users within its first three months. By 2012, the app's user base had grown to 27 million, and in April, Facebook purchased the app for US$1.0 billion in cash and stock, allowing it to continue to operate as an independent company.[180]
- Pinterest's origins derived from an earlier app that two of its founders had created—Tote—an app where users could browse apparel and other goods from 30 retailers, save items and receive push notifications when those items were reduced in price. E-commerce had not yet become mainstream, and the app did not take off as hoped, but when examining the user data, the founders noted some specific behavior patterns. Users were browsing for products, saving them in the favorites function and then sending images of the products to themselves. The founders also noted that users had not looked for brand-specific items but for categories of items—shoes, shirts, dresses. This collecting behavior spurred the idea of offering people "buckets" to organize their collection of images. Pinterest came up with a grid concept for displaying collections of things that people liked. It launched in March 2011 and was featured later that year in TIME Magazine's "50 Best Websites" round-up, and by December, it was among the 10 largest social media services with more than 11 million unique visitors per week.[181]

"These sites were joined in later years by innovators such as TikTok, but the success of that platform and its 'for you' offering, as well as the ongoing success of the above-mentioned social media platforms, became tied increasingly to an ability to analyze and leverage the massive amounts of data being generated by users' interactions with these platforms and to apply algorithms to increase user engagement."

- Snapchat was launched in July 2011, originally under the name Picaboo as a nod to the founders' concept of having a photo show up and then soon disappear. After receiving a cease-and-desist notice from another company that had trademarked that name, the platform relaunched in September 2011 as Snapchat. The service became a big hit with young users who did not want their social media activities and history to be accessible to adults. By November 2012, 20 million snaps were being shared daily. The company soon thereafter offered an ability to post 10-second videos and create "stories"—serialized short video and photo posts that would remain accessible for only 24 hours.[182]

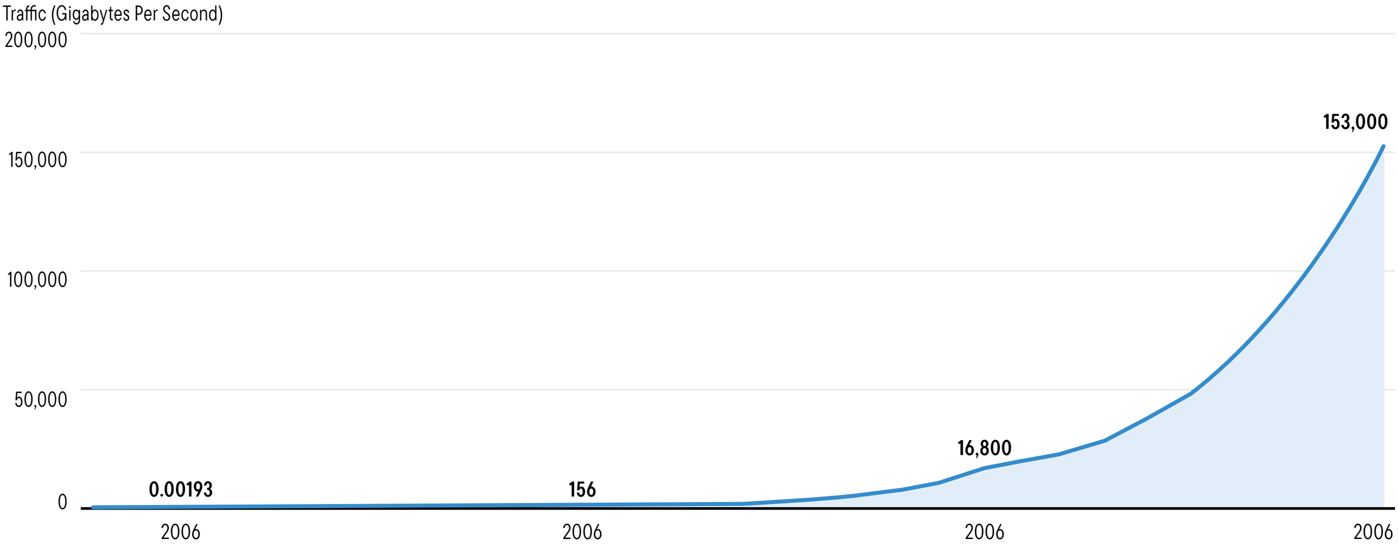
These sites were joined in later years by innovators such as TikTok, but the success of that platform and its "for you" offering, as well as the ongoing success of the above-mentioned social media platforms, became tied increasingly to an ability to analyze and leverage the massive amounts of data being generated by users' interactions with these platforms and to apply algorithms to increase user engagement. New approaches to data analysis were required to accomplish this aim, and the tools and techniques utilized became grouped under the term "Big Data."

Big data

As noted earlier, concerns about the amount of data being created by the computing-enabled automation of commerce were noted as early as 1971 as commenters lamented the challenges wrought by the "knowledge explosion." Efforts to better utilize the rising amounts of data resulted in the shift in database approach from hierarchical to relational database

Exhibit 14: Growth of Global Internet Traffic in the Past 30 Years

2017–2022



Source: WDR 2021 team calculations and Cisco Visual Networking Index: Forecast and Trends. 2017–2022.

management systems by the mid-to-late 1970s. Relational database management and data warehousing became core toolkits allowing for queries, analytical processing and standard reporting tools.

Compared to earlier eras, however, the amount of data being generated since the growth of internet technologies has been astronomical. Exhibit 14 shows the growth in global internet traffic over the past 30 years.

In 1992, global internet traffic was 100 gigabytes (GBs) per *day*. By 2002, internet traffic had increased to 156 GBs per *second*. By 2022, it was expected to reach 153,000 GBs per second—a 1,000 times increase in the past 20 years. This works out to be more than three zettabytes—3,000,000,000,000 gigabytes—per day. Termed an unimaginably big and abstract number, the World Bank attempted to quantify it as follows: the equivalent of 325 million households watching Netflix simultaneously for 24 hours a day versus only 10 households binge watching Netflix for 10 hours in 1992.[183]

The growth in devices connected to the internet has been a major driver of this rise in traffic. According to Cisco's internet traffic report, there will be 29.3 billion networked devices in 2023, up from 18.4 billion devices as recently as 2018.[184] Globally, devices connected to the internet are growing faster than the world's population and faster than the actual number of users of the internet. The average number of devices and connections per household and per capita is increasing, fed to a large extent by growth in machine-to-machine (M2M) connections that fall under the category of IoT.

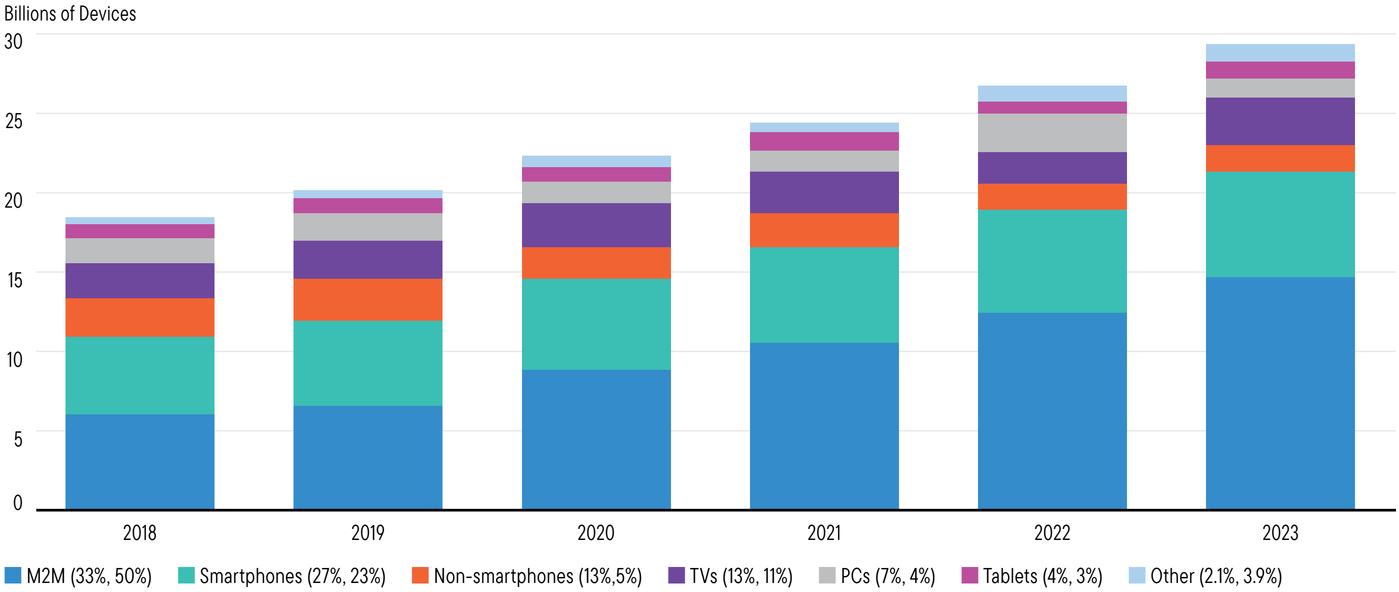
M2M connections include a broad range of devices from smart meters to video surveillance equipment to health care monitoring devices to package or asset tracking. By 2023, M2M devices are expected to account for 50% of all networked devices compared to only 33% in 2018. For comparison, smartphones are expected to represent only 23% of devices in 2023.[185] Growth in devices by channel is highlighted in Exhibit 15 on the next page.

The result of this growth in internet traffic and devices has been an unprecedented increase in the amount of data created, captured, copied and consumed globally. In 2010, the estimated volume of data was 2 zettabytes. By 2015, this figure had jumped to 15.5 zettabytes and then again surged to an estimated 64.2 zettabytes in 2020—a larger-than-projected increase caused by higher demand due to the COVID-19 pandemic, as more people worked and learned from home and used home entertainment options more often. Exponential growth is expected to continue, with estimates for 2025 currently set at 181 zettabytes.[186] This growth in data is shown in Exhibit 16 on the next page.

Only a small percentage of this newly created data is kept, however. In 2020, only 2% of the data produced and consumed was saved and retained into 2021. In part, this is because the installed base of storage capacity has not been growing as quickly as the data being produced. In 2020, storage capacity was estimated at 6.7 zettabytes, only 10% of

Exhibit 15: Growth in Connected Devices

2018–2023

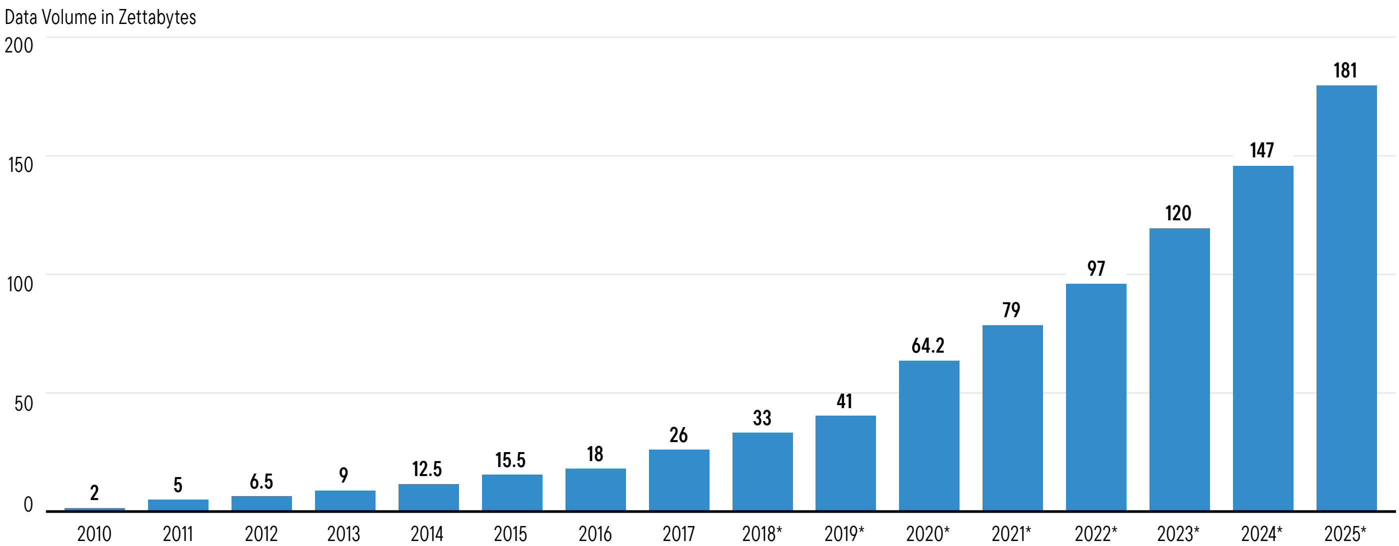


*Figures (n) refer to 2018, 2023 device share

Source: Cisco Annual Internet Report (2018–2023) White Paper. March 9, 2020. There is no assurance that any forecast, estimate or projection will be realized.

Exhibit 16: Volume of Data/Information Created, Captured, Copied, and Consumed Worldwide (in Zettabytes)

2010–2025



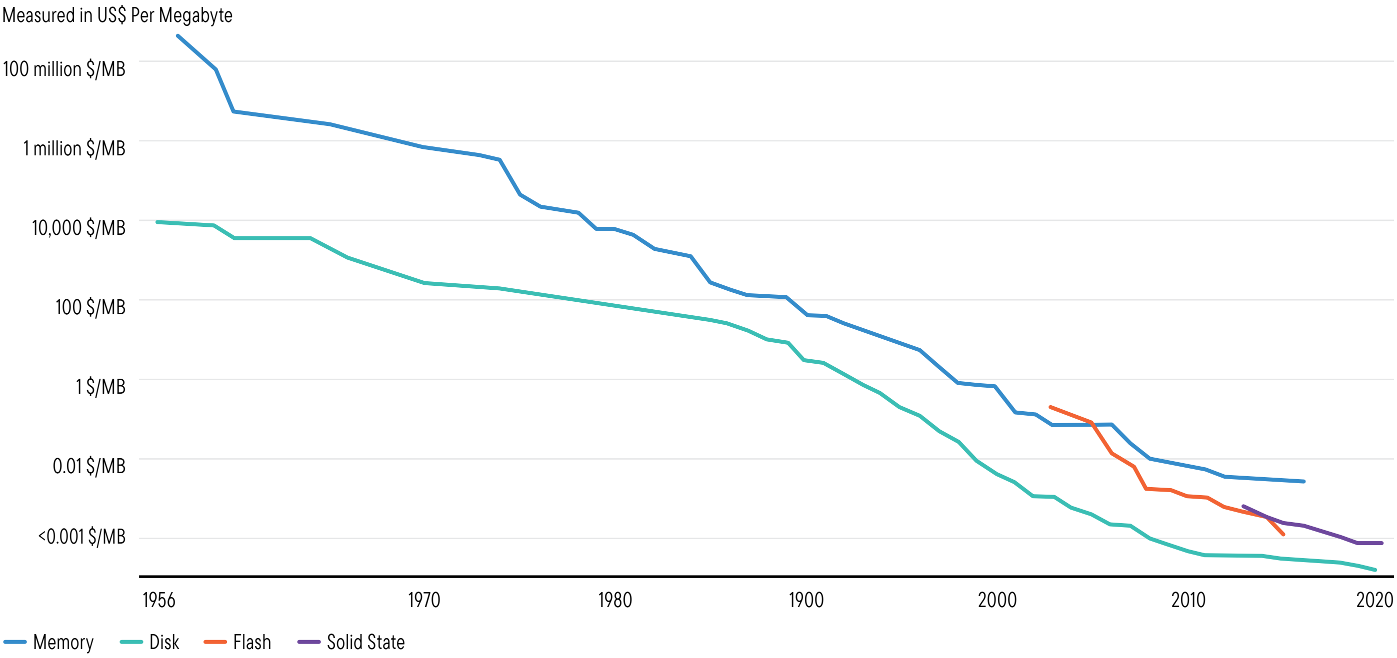
Sources: IDC, Seagate, Statista estimates, June 2021. *Estimate. There is no assurance that any forecast, estimate or projection will be realized.

the data produced that year. Although projected to grow at a five-year CAGR (compound annual growth rate) of 19.2% between 2020 and 2025, data storage capacity is only anticipated to reach 16 zettabytes by 2025—only 8.8% of projected data creation.[187]

Creating adequate storage capacity to keep up with the speed of data creation is the issue—not the costs of storing and utilizing this data. As shown in Exhibit 17 on the next page, the costs of computer memory and storage have been falling consistently since the emergence of commercial computing at the outset of the 1960s. In 1970, the cost of computer memory was approximately US$1.0 million per megabyte of data and the cost of storage was a few hundred US dollars per megabyte. By 2020, both figures were down to less than one US cent per megabyte.[188]

Exhibit 17: Historical Cost of Computer Memory and Storage

1956–2020



Source: John C. McCallum, 2022. For each year, the time series shows the cheapest historical price recorded until that year.

This trend of becoming increasingly less expensive to process and store data is both a proof point on Moore's Law—proposed way back at the outset of the invention of the integrated circuit chip—and an enabler to new data-driven business models. Indeed, data have gone from the exhaust to the utility that drives much of today's economy, particularly in the social media realm.

Within each social media ecosystem, "users don't just log in and browse, they tell the platforms their name, where they live, what they like and who they know, painting the most vivid picture currently possible for marketers looking to target specific consumers."[189] This provided marketing professionals unprecedented access to valuable, actionable data about consumers' demographics, buying habits and more. In 2018, businesses spent US$107.5 billion for online advertising, compared to US$124.2 billion collectively across television, radio, newspaper and magazine advertising.[190]

The leading provider of online advertising is Google. More than 80% of Alphabet's (Google's parent company) revenue in 2020 (US$183 billion) came from Google Ads (US$147 billion). In 2021, Google commanded nearly a 29% share of digital ad spending globally, followed by Facebook at just under 24%. Alibaba was a distant third with less than 9%.[191]

Google pioneered the pay-per-click (PPC) model that dominates online advertising today. In this approach, advertisers put in an auction bid designating the maximum amount that they are willing to pay to have their organization highlighted in response to certain keywords. The keywords themselves and the demographics around who is most likely to respond to the keywords are determinants of the cost-per-click, and ad sites provide analytics to help advertisers model various keyword bids. These platforms also help advertisers determine the ad "quality" or how well their ad performs and make suggestions on how to improve keyword selection to enhance quality. Facebook lets advertisers set a goal for their advertising and track their stated objectives against their actual advertising results as part of a campaign.

In 2017, 65% of small- to mid-sized business owners surveyed in the United States indicated that they would be carrying out a PPC campaign.[192] PPC advertising allows businesses to promote their company on search results, on websites, and on social media platforms. Ads that appear in search results earn more than 45% of page clicks, and people who choose to click on those ads are twice as likely to buy a product or service than an organic visitor.[193]

Google also allows advertisers to target a location, language, or audience for ad placement, and as of 2019, allowed advertising on Google Maps—a product that has more than one billion active monthly users and is updated tens of thousands of times per day. Google also allows YouTube creators who want to make money from their channel and

“Google’s use of its vast repositories of information to manage these data-driven advertising campaigns does not require much change in approach from the historic way in which relational database management systems perform data management. The scale of data being considered is much larger, the data sets are unique, and the analysis is examining new types of patterns, but most of the information driving ad-targeting is structured, tagged and curated.”

are eligible, to turn on ads for video. Eligible channels must have logged 4,000 public watch hours within the past 12 months and have at least 1,000 subscribers.[194]

Google’s use of its vast repositories of information to manage these data-driven advertising campaigns does not require much change in approach from the historic way in which relational database management systems perform data management. The scale of data being considered is much larger, the data sets are unique, and the analysis is examining new types of patterns, but most of the information driving ad-targeting is structured, tagged and curated. There is, however, a significantly larger pool of data being generated by the myriad of devices being connected to the internet, from satellites and from the internet traffic itself, which is at best semi-structured and in many instances completely unstructured.

By the early 2010s, it was increasingly clear that enterprises needed tools, technologies and analytic techniques that were able to extract meaningful information out of unstructured data. This marked a fundamental shift in thinking. Up until this time, “enterprise database assumptions had focused on ACID (atomicity, consistency, isolation and durability).” Thinking about unstructured data led to a “transformation of data use cases as companies realized that data previously thrown away or kept in static archives could provide value to understanding customer behavior, propensity to take action, risk factors, and complex organizational, environmental, and business behaviors.”[195]

The blueprint on how to attack this issue of utilizing and extracting meaning from unstructured data was inspired by two Google technical papers published in 2003–2004 that outlined how the company set up and delivered its Google File System (GFS), a scalable distributed file system, and how it processed that data (through MapReduce). Filesystems are an integral component of most operating systems that direct inquiries to where the required data are stored. When one saves a file to a magnetic hard disk, it gets recorded as a logical series of 1s and 0s. Every time a sufficiently large-enough collection of these 1s and 0s get generated, they are formed into a block. The file system is the part of the operating system that makes sense of the blocks. Google’s file system was designed to be highly distributed and able to support extremely large files (since Google was indexing all the web’s information). This was a departure from an earlier file system design in both its design and its operations.[196]

Google also explained how it processed the data—this was called MapReduce. In essence, most applications up until that time pulled data from the database into the application’s logic to process the data and perform the business function. MapReduce reversed this paradigm. It was a framework that allowed applications to process huge amounts of data, in parallel, on large clusters of commodity hardware in a reliable manner. It “sent the computer to where the data resides.”[197] The benefit of this approach was that the data just stayed where it was in whatever form it had been received and filed. It did not require “clean” data that had been tagged, stored and mapped to relational databases.

The open-source community embraced and ran with this concept. Open source is a term that originally referred to open-source software. This was software where the code was designed to be publicly accessible—people can see, modify and distribute the code as they see fit. Open-source software is developed in a decentralized and collaborative way, relying on peer review and community production. As noted earlier, the origins of the open-source community can be traced back to time-sharing computers and the teams that developed the Unix computer.

The Apache Software Foundation is the world’s largest open-source group of developers. In 2006, a team of developers came out with the Hadoop project, “a framework that allows for the distributed processing of large data sets across clusters of computers using simple programming models designed to scale up from single servers to thousands of machines, each offering local computation and storage.”[198]

Hadoop became the core offering that led to a whole series of new technologies aimed at processing “big data”—the massive data that cannot be handled with conventional database processing techniques and that is larger and more complete than what can be stored on a relational database management system.[199]

Hadoop clusters can store and process structured, semi-structured and unstructured data. One of the main features of Hadoop is its ability to accept and manage data in its raw form. It can also access historical data from legacy databases. A whole series of tools were then developed to work within the Hadoop framework, forming a Hadoop stack. Other technology vendors followed, building out the whole field of big data offerings and allowing many of the capabilities to become services that could be utilized on demand in cloud computing platforms.

These tools made it possible to query extremely large sets of data that might exist in a range of structured to unstructured formats, but the job of sifting through that data, parsing it to be machine readable, and analyzing it to improve business and public policy decision-making processes proved challenging. Increasingly, those looking to optimize big data opportunities began to incorporate an emerging set of AI tools.

The artificial intelligence toolkit

While the pursuit of AI has a thread that runs throughout the entirety of the computing age, early promises of how the technology could enable “smart machines” went unfulfilled for decades. A whole series of AI experimentation went on in the late 1950s to the mid-1970s, but efforts in this period suffered from a lack of computational power. Computers could not store enough information or process it quickly enough—as one leading researcher of the time put it, “computers were still millions of times too weak to exhibit intelligence.”[200]

Interest in AI revived somewhat in the 1980s with the expansion of the algorithmic toolkit. Two researchers popularized “deep learning” techniques that allowed computers to learn using experience and another researcher introduced expert systems that mimicked the decision-making process of a human expert. The Japanese government heavily funded expert systems and other AI-related endeavors as part of its Fifth Generation Computer Project, investing US$400 million between 1982 and 1990, but saw limited results and interest tail off.[201]

These efforts inspired the next generation of talented engineers and scientists, and during the 1990s, many of the landmark goals of AI began to be realized. In 1990, a new approach to machine learning emerged called boosting. Instead of using a single strong model, boosting generates many weak models and converts them into a strong model by combining their predictions. In 1995, random decision forests were introduced—an algorithm that creates and merges multiple AI decision trees to significantly improve accuracy and decision-making. By 1997, the results of these innovations were evident when IBM’s Deep Blue computer beat world champion and grand master Gary Kasparov at chess.[202]

By the early 2000s, big data advancements had repositioned the possibilities of AI, solving for both storage capacity and issue of processing. The concept of deep learning began to take hold. Deep learning is a type of machine learning based on artificial neural networks in which multiple layers of processing are used to extract progressively higher-level features from data. Basically, the processors keep finding results and then use those results to continue looking for more patterns in the data. The machines are “trained” by algorithms that ingest massive amounts of data to process and reprocess. In 2009, a massive visual database of labeled images called ImageNet was launched. The Economist described this database as an “an exceptional event for popularizing AI throughout the whole tech community.”[203]

“By the early 2000s, big data advancements had repositioned the possibilities of AI, solving for both storage capacity and issue of processing. The concept of deep learning began to take hold. Deep learning is a type of machine learning based on artificial neural networks in which multiple layers of processing are used to extract progressively higher-level features from data. Basically, the processors keep finding results and then use those results to continue looking for more patterns in the data.”

By 2014, a group of scientists had created generative adversarial network (GAN) frameworks that teach AI how to generate new data based on the training set. By 2014, a Facebook research team had developed DeepFace, a facial recognition system that had a nine-layer neural network trained on four million images of Facebook users. The AI was able to spot human faces in images with the same accuracy as humans. That same year, Google introduced Sibyl, a large-scale machine learning program that is used in Google's prediction models, specifically for ranking products and pages, measuring user behavior, and advertising.[204]

The culmination of these efforts came in 2016 when DeepMind's AlphaGo computer beat the reigning 18-time world champion Lee Sedol at the game Go. Many consider Sedol to be the best player in the game's history. The game itself was considered nearly impossible for AI to play at anything other than the level of an amateur. Standard AI methods, which test all possible moves and positions using a search tree, cannot handle the sheer number of possible Go moves or evaluate the strength of each possible board position. There are 10 to the power of 170 possible board configurations—more than the number of atoms in the known universe. AlphaGo combined decision-tree approaches with two different neural networks to evaluate its playing options. Its 4-1 victory over Sedol in March 2016 was watched by more than 200 million people worldwide.[205]

To prepare AlphaGo for this competition, the team had trained the computer having it first play against many amateurs and then, as it improved, against many professionals to understand how humans approached the game. The team then had it play against itself thousands of times to learn from its mistakes. This was called reinforcement learning. Following its victory, the team returned to the lab and created AlphaGo Zero. Instead of giving it any training, the team provided the computer just the rules of the game and set it to playing against itself. This is called adversarial learning. The program accumulated thousands of years of human knowledge during a period of just a few days. It was able to beat the original AlphaGo program shortly thereafter, and it developed unconventional strategies and creative new moves that had never been seen before.[206]

Adaptations of AI are now emerging in everyday life. Individuals can use facial recognition to open their phones, use Siri or Alexa to get directions or make calls when they are driving, have Netflix recommend what they should watch based on their personal viewing patterns, run their homes through smart devices, see items they thought about buying in ads that appear on their search and web pages, receive targeted content in their social media feeds, and in some cases let their cars drive themselves. AI is powering the sensor data that is collecting information on weather, shipping, carbon emissions, traffic patterns and countless other items.

"Many innovations are probably yet to come from the technology offerings of the virtualization era—cloud computing, big data and the AI toolkit—and from new cloud-based development opportunities that deliver infrastructure, data, software and analytics as a service."

The tools to enable AI are being offered to businesses, governments and universities to improve outcomes. Offerings such as natural language processing, machine learning, predictive analytics and interactive voice are being used increasingly in commerce. More applications are likely. Already, many new types of edge computing analytics are being developed as "no code" platforms. These offerings leverage AI tools to let users pose analytic requests in everyday language and allow them to create and run their own work routines and analytic reports.

Many innovations are probably yet to come from the technology offerings of the virtualization era—cloud computing, big data and the AI toolkit—and from new cloud-based development opportunities that deliver infrastructure, data, software and analytics as a service.

The story of how technology is changing the ways in which societies operate does not end here, however. There is already another cycle of technology innovation unfolding that is still in its proof-of-concept stage. This new set of offerings may prove to be even more disruptive and transformational than the move from simple Web1 sites to Web2 online communities and ecosystems and even more transformational than the introduction of service-based cloud development platforms. Generally grouped under the heading of Web3, these new technologies take a radically different approach on how commerce should be organized and executed.

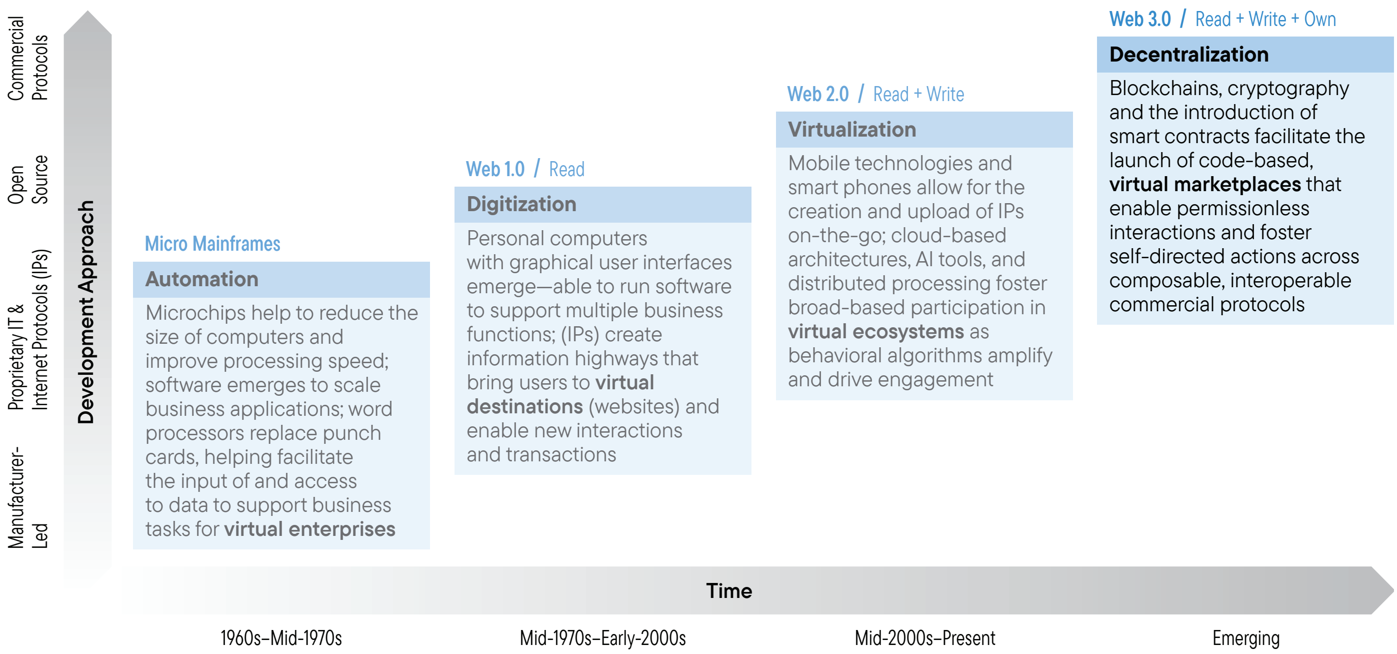
Section V

Fourth cycle of commercial technology—decentralization

In many aspects, the emerging commercial technologies are an extension of the current trend that is shifting technology away from a centralized to a distributed processing and development approach, but the coming cycle looks to move even further to a fully distributed *operating* environment. Opposition to the path that the current economy has taken is a driving force behind this goal—a desire to protect individual privacy, eliminate or greatly reduce the need for intermediaries, and remove the levers of commerce from only a limited set of providers. Instead, those driving new technologies are architecting a system with the specific intent of enabling a peer-to-peer economy where work, entertainment, engagement and commerce can occur directly between the individuals on either side of the interaction and where any economic benefit that accrues is directed to those that participate in the system, not those that own it.

Exhibit 18 shows the progression in technology approach and illustrates the decentralization cycle that is now emerging.

Exhibit 18: Modern Commercial Technology: Emergence of Cycle 4



Source: Franklin Templeton Industry Advisory Services. For illustrative purposes only.

Cryptography, the cypherpunks and the crypto wars

As noted back in Section I, one of the major accelerants in the evolution of computing and the shift from an electro-mechanical calculator to a computer was the outbreak of World War II. Not only did the military need computers to help with the tracking and deployment of air power and munitions, but strategists also needed to break the ciphers that were being used to encode enemy messages so that they could anticipate their plans and activities. The practice of deciphering these codes was termed cryptography. The basics of how codebreaking worked were kept secret for much of the period post World War II and into the Cold War years until two publications released in the 1970s brought an understanding of cryptographic techniques into the public domain.

The first was a government-sponsored work from the US Department of Commerce and the National Institute of Standards and Technology. The piece laid out the specifications for two cryptographic algorithms that "may be used by Federal organizations to protect sensitive data...during transmission or while in storage...to maintain the confidentiality and integrity of the information represented in the data." The two algorithms were the Data Encryption Standard and the Triple Data Encryption Algorithm. The paper laid out the steps to enable "the algorithms to define the mathematical steps required to transform data into a cryptographic cipher and also to transform the cipher back to the original form."[207]

The second work released in 1976 was a paper from two researchers at Stanford University entitled *New Directions in Cryptography*. They state in their introduction that "we stand today on the brink of a revolution in cryptography. The development of cheap digital hardware has freed it from the design limitations of mechanical computing and brought the cost of high-grade cryptographic devices down to where they can be used in such commercial applications as remote cash dispensers and computer terminals." They go on to note that privacy—"preventing unauthorized extraction of information from communications over an insecure channel"—was the major cryptographic concern and that the only way to assure privacy was "for the communicating parties to share a key which is known to no one else." They went on to propose two approaches to transmit key information over public, insecure channels.[208]

In 1985, a Ph.D. graduate from the University of California, Berkeley—Dr. David Chaum—released his paper, *Security without Identification: Transaction Systems to Make Big Brother Obsolete*. In it he wrote, "Computerization is robbing individuals of the ability to monitor and control the ways that information about them is used. Already public and private sector organizations acquire extensive personal information and exchange it amongst themselves. Individuals have no way of knowing if this information is inaccurate, outdated, or otherwise inappropriate. New and more serious dangers derive from computerized pattern recognition techniques: even a small group using these and tapping into data gathered in everyday consumer transactions could secretly conduct mass surveillance, inferring individuals' lifestyles, activities, and associations. The automation of payment and other consumer transactions is expanding these dangers to an unprecedented extent."[209]

"These three works laid the foundation for a practice known as asymmetric or public key cryptography. The public key encrypts the data being sent, a cipher is used to confirm that the entity receiving the data has the right secret key (digital signature) to decrypt it, and then the secret or private key is used to decrypt the data."

Chaum went on to propose pseudo-anonymous reputation systems—where a user's data is made unidentifiable to the public and they use a numerical digital signature to create a private key. He also espoused the creation of anonymous digital cash.[210]

These three works laid the foundation for a practice known as asymmetric or public key cryptography. The public key encrypts the data being sent, a cipher is used to confirm that the entity receiving the data has the right secret key (digital signature) to decrypt it, and then the secret or private key is used to decrypt the data.

In 1988, Timothy C. May, a political and technical writer as well as an engineer and senior scientist at Intel, published the *Crypto Anarchy Manifesto*, noting that "computer technology is on the verge of providing the ability for individuals and groups to communicate and interact with each other in a totally anonymous manner. Two persons may exchange messages, conduct business, and negotiate electronic contracts without ever knowing the true name or legal identity of the other...These developments will alter completely the nature of government regulation, the ability to tax and control economic interactions, the ability to keep information secret, and will even alter the nature of trust and reputation."[211] May based his vision on "public key encryption, zero-knowledge interactive proof systems, and various software protocols for interaction, authentication, and verification."[212] He drew on the term "Crypto" from the Greek meaning secret or hidden.

In 1992, Tim May, along with two other friends—Eric Hughes, a mathematician at the University of California, Berkeley, and John Gilmore, a computer scientist at Sun Microsystems—met together to discuss issues surrounding cryptography and privacy. They launched a series of projects that began to draw in like-minded individuals. Jude Milhon, a programmer, author and civil rights activist, was one member of their community. She coined the term "cypherpunks" to describe the community, playing on the word "cipher" and combining it with the sci-fi genre "cyberpunk."[213]

In 1993, Eric Hughes published *A Cypherpunk's Manifesto*. The manifesto states, "Privacy is necessary for an open society in the electronic age. Privacy is not secrecy. A private matter is something one doesn't want the whole world to know, but a secret matter is something one doesn't want anyone to know. Privacy is the power to selectively reveal oneself to the world." The manifesto additionally states that privacy in an open society requires anonymous transaction systems and cryptography since "to reveal one's identity with assurance when the default is anonymity requires the cryptographic signature." The manifesto declares that its signatories are "dedicated to building anonymous systems... defending our privacy with cryptography, with anonymous mail forwarding systems, with digital signatures, and with electronic money."[214]

One of the forces motivating the cypherpunk movement was the US National Security Agency's introduction of the "clipper chip" in 1993. The clipper chip was a state-of-the-art microchip developed by government engineers that could be inserted into consumer hardware telephones, providing the public with strong cryptographic tools, but still leaving law enforcement and intelligence agencies the ability to access unencrypted versions of those communications. The technology relied on a system called "key escrow," in which a copy of each chip's unique encryption key would be stored with the government. Although the White House mobilized political and technical allies in support of the proposal, it faced an immediate backlash from technical experts, privacy advocates and industry leaders. The ensuing dispute was dubbed the start of the "Crypto Wars." When a computer scientist affiliated with the cypherpunks, Matt Blaze, broke the encryption of the clipper chip in May 1994, the government withdrew its proposal.[215]

> **"Between 1994 and 1997, the US government continued to explore a way to access the keys to encrypted communications. For a time, there was an effort to pursue 'software key escrow' to preserve access to phone calls, emails, and other communications and storage applications."**

Between 1994 and 1997, the US government continued to explore a way to access the keys to encrypted communications. For a time, there was an effort to pursue "software key escrow" to preserve access to phone calls, emails, and other communications and storage applications. Under these plans, a government-certified third party would keep a "key" to every device. Opposition remained stiff as privacy, security and economic concerns won out.[216]

Concurrently, a related battle was being fought over the US export controls and encryption technology. Until 1996, cryptographic tools were classified as munitions with strict limits on the type of encryption that could be exported and a mandated maximum cryptographic key length of 40 bits—a figure at which the encryption "could be broken in a matter of days using a single personal computer."[217]

By the mid-1990s, experts were projecting that billions of US dollars in potential losses could result from this policy, particularly as non-US based encryption was readily available. In 1996, US President Bill Clinton issued an executive order that moved most commercial encryption tools from the US Munitions List to the Commerce Control List. Other steps were taken to gradually liberalize encryption export controls.

In May 1999, the US Ninth Circuit Court of Appeals ruled that the federal government's restrictions on encryption were unconstitutional, affirming a lower court's ruling that export control over "cryptographic software and related devices and technology are in violation of the First Amendment on the ground of prior restraint." A sponsor on the case noted that "government efforts to control encryption may well implicate not only the First Amendment rights of cryptographers intent on pushing the bounds of their science, but also the constitutional rights of each of us as potential recipients of encryption's bounty."[218]

In September 1999, the White House announced a sweeping policy change that removed virtually all restrictions on the export of retail encryption products, regardless of key length. The Crypto Wars were over, and the public had won. Much of the growth of the web and the digital economy that has taken place in the past 20 years can be directly traced back to these efforts to protect commercial encryption for secure digital communications. Such advancements include electronic banking, electronic medical records systems, online bill payment tools, home automation systems, e-filing systems for taxes, and more.[219]

Digital cash and cryptographic payment systems

While the Cypherpunks and the Crypto Wars focused primarily on asymmetric or key-based cryptography to ensure privacy, another stream of work also progressed during this period geared at exploring the possibilities of anonymous digital cash and payment systems.

In 1997, a British affiliate of the cypherpunk movement, Dr. Adam Black, created Hashcash, which was designed as an anti-spam mechanism that would essentially add a (time and computational) cost to sending email, thus making spam uneconomical. The project was inspired by a paper from two computer scientists—*Pricing via Processing, or Combatting Junk Mail*—that proposed forcing senders to pay every time an email message was sent, but rather than paying in money, the senders would pay in time by being forced to access their computer's memory to solve a computational puzzle, devised on the fly for that particular message.

Hashcash was a plugin software for mailers that would add a "hashcash stamp" to an email before it could be sent. The sender of the email would have to expend computer time to create the stamp that would be affixed to the email header (about 10 seconds). The recipient email system would also need to spend computing time to verify the stamp and open the email, but the amount of time would be much more negligible. Sending email in this way for the average user would add no nominal cost, but spammers that were sending tens of thousands of emails would be deterred.

> **"This process of spending computer time and processing power to create an encrypted hash function and then requiring another computer to spend some, but a more nominal amount trying to match the sequence of numbers to un-encrypt the data is called proof-of-work (PoW)."**

The process of creating the stamp relied on a hash function—a one-way algorithm that transforms data of any size down to a fixed string of numbers. The sender then verifies how many bits of this string of numbers need to be matched to unlock the email. Running the algorithm, creating the hashcash stamp, and designating the set of numbers that need to be verified each take up computing time and processing power; thus, they can be calculated as a cost. To verify the stamp and open the email, the receiving machine needs to start testing various strings of numbers until they get a match. Hashcash chose a very simple 16-bit string of numbers. According to the organization's frequently asked questions, it should take 2^160 tries to match the numbers. While this sounds excessive, it would take a PC of the time about one-third of a second to complete.[220]

This process of spending computer time and processing power to create an encrypted hash function and then requiring another computer to spend some, but a more nominal amount trying to match the sequence of numbers to un-encrypt the data is called proof-of-work (PoW).

In 1998, computer engineer Wei Dai—another cypherpunk community member—published a proposal that talked about b-money—"an anonymous, distributed electronic cash system." He notes that he had been reflecting on Tim May's *Crypto Anarchy Manifesto* and realized that a "community is defined by the cooperation of its participants, and efficient cooperation requires a medium of exchange (money) and a way to enforce contracts. Traditionally, these services have been provided by the government or government-sponsored institutions, and only to legal entities." Wei Dai went on to say that he would "describe a protocol by which these services can be provided to and by untraceable entities."[221]

“As noted, many of the innovations that came together for the launch of Bitcoin were pioneered years earlier by members of the cypherpunk movement—a group similar to the open-source community that believed in sharing its code and ideas. This is an important nuance, because to the present day, no one knows the identity of the supposed inventor of bitcoins, Satoshi Nakamoto.”

In 2004, yet another cypherpunk affiliate, developer Hal Finney, built upon Adam Back’s Hashcash and created the first reusable proof-of-work (RPoW) system, which would work with token money. Just as a gold coin’s value is linked to gold mining cost, the value of an RPoW token is guaranteed by the value of the real-world resources (computer time) required to “mint” a proof-of-work token. In his proposal, the RPoW token would be a piece of Hashcash. The innovation in his approach was that it would be possible to reuse the tokens without having to repeat the work that was required to generate them by tracking each token and exchanging it sequentially to avoid double-spending. The creation and exchange of tokens could be “monitored by users throughout the world to verify its correctness and integrity in real time.”[222]

These innovations—PoW, anonymous monetary transfer protocols and reusable tokens where transactions are publicly verified—were all important precedents that came together in the launch of bitcoin.

Bitcoin and blockchain

As noted, many of the innovations that came together for the launch of Bitcoin were pioneered years earlier by members of the cypherpunk movement—a group similar to the open-source community that believed in sharing its code and ideas. This is an important nuance, because to the present day, no one knows the identity of the supposed inventor of bitcoins, Satoshi Nakamoto. There are speculations that the original white paper may have been written by a group of people, a supposition supported by the broad range of concepts covered in the paper—computer science, cryptography, economics, accounting, programming and probability theory. The abstract of the paper also mentions “we propose a solution” as opposed to “I propose a solution,” lending credence to the idea that it was a group effort.[223]

Regardless of who authored it, on August 18, 2008, the domain name bitcoin.org was registered anonymously, and on October 31, 2008, a white paper was published to a cryptography mailing list titled, *Bitcoin: A Peer-to-Peer Electronic Cash System*. The paper detailed a “peer-to-peer system for electronic transactions without relying on trust.”[224] On January 3, 2009, Satoshi Nakamoto launched the Bitcoin network by mining “The Genesis Block” and received a reward of 10 bitcoins. The first transaction on the blockchain network was for 10 bitcoins sent to Hal Finney on January 12, 2009.

Many view the financial crisis and credit bubble that shook the global economy in the fall of 2008–winter of 2009 as the impetus for the launch of Bitcoin. Embedded in the coin base of Bitcoin’s Genesis Block was the text “The Times Jan/03/2009 Chancellor on brink of second bailout for banks.” In February 2009, Satoshi Nakamoto wrote a post on the message board for the P2P Foundation, an organization focused on peer-to-peer technology introducing Bitcoin. In it they wrote, “Banks must be trusted to hold our money and transfer it electronically, but they lend it out in cycles of credit bubbles with barely a fraction in reserve. We have to trust them with our privacy, trust them not to let identity thieves drain our accounts. Their massive overhead makes micropayments impossible.”[225]

The set-up of Bitcoin was very similar in many ways to the RPoW system described by Hal Finney in 2004, with one major exception. Finney’s system relied on a centralized network server to hold the account ledger that tracked the sequential reuses of the RPoW token, and his system sought transparency by having the source code transparent and replicable so that others could download the code and monitor the transactions to ensure their validity. Bitcoin took a very different approach, introducing a new technology to the world—blockchain.

Blockchain is a distributed database or ledger that is shared among the nodes of a computer network. It stores information electronically in a digital format. Unlike a typical database that stores data in tables or files, a blockchain collects information together in groups, known as blocks, that each have a set limit of storage capacity.

When the block is filled with data, it is closed and linked to previously filled blocks to form a chain of data. This approach to creating the ledger results in an irreversible timeline when implemented in a decentralized manner, since every node on the computer has visibility into the blocks on the chain and each copy of the ledger—held on every single one of the nodes—would have to be changed in tandem to implement a change to the database, otherwise a fraud could be easily identified. As such, blockchains are seen as immutable—none of the records of transactions can be altered, deleted or destroyed. In this way, blockchains guarantee the fidelity and security of a record of data and generate trust without the need for a trusted third party.[226]

Another attribute of the Bitcoin network is that transactions are accomplished pseudo-anonymously. Activity is linked to a user's digital wallet, which is identified by a long cryptographically protected address, not by any identifying characteristics of the individual owner of the wallet. Every transaction ever done involving a specific digital wallet address is archived in the blockchain metadata and can be reconstructed and tracked, but to uncover the identity of the underlying owner of the wallet requires external investigation and sophisticated cybersecurity techniques.

With the launch of Bitcoin, the vision for a transactional network based on trustless transactions, cryptographically protected digital signatures and electronic money laid out in the *Crypto Anarchist Manifesto* and the *Cypherpunk Manifesto* came to pass—just over 15 years after their publication. With Bitcoin, two people, anywhere in the world, can send bitcoins to each other without the involvement of a bank, government or another institution.

"The following examples should make this concept clear. If I text you a photo from my phone, you have a copy of the photo and I have a copy of the photo. If I send you a bit coin from my Bitcoin wallet, that bitcoin is digitally removed from my ownership and given to you. This is an advancement of concept of sequential reuse introduced by Hal Finney in 2004. This is also why bitcoins and other cryptocurrencies can be used as money. If I give you a US$50 bill, you now have that bill and I no longer have the money. Without digital scarcity, it would be impossible to trust that a digital token might truly represent a store of value."

Yet, the innovation introduced by the launch of Bitcoin did not stop there. There is one other innovation that bitcoin delivered which is instrumental to understanding why the entire cryptocurrency space that developed subsequently is so revolutionary. Bitcoin created the concept of digital scarcity. Before Bitcoin, any object in a digital network, like an email or a photo, could be copied an infinite number of times. Bitcoin is the first example of a digital good whose transfer stops it from being owned by the sender.

The following examples should make this concept clear. If I text you a photo from my phone, you have a copy of the photo and I have a copy of the photo. If I send you a bitcoin from my Bitcoin wallet, that bitcoin is digitally removed from my ownership and given to you. This is an advancement of concept of sequential reuse introduced by Hal Finney in 2004. This is also why bitcoins and other cryptocurrencies can be used as money. If I give you a US$50 bill, you now have that bill and I no longer have the money. Without digital scarcity, it would be impossible to trust that a digital token might truly represent a store of value.

Nakamoto introduced other mechanisms as well to help ensure the value of bitcoins. Bitcoin has systematically limited the total supply of its coins to 21 million. The level of difficulty in mining bitcoins also automatically changes when the collective computing power in the network goes up. In other words, it becomes more difficult to mine bitcoins as more bitcoins circulate. Bitcoin halving occurs every time 210,000 bitcoins are mined. Halving reduces the reward paid to the miners for their proof-of-work verification. Upon launch, validators received 50 bitcoins for mining a block. Today, that figure is down to 6.25 bitcoins. These mechanisms result in bitcoin mining becoming both more difficult and less rewarding as competition on the network for access to coins increases.[227]

The final innovation introduced with Bitcoin was the ability to support micropayments. Today, a US dollar can be reduced to 100 pennies, but it cannot be reduced beyond that amount. The Bitcoin network allows for the reduction of a single bitcoin into units called a Satoshi—named after the anonymous creator—which equates to one hundred millionth of a single bitcoin (0.0000001).

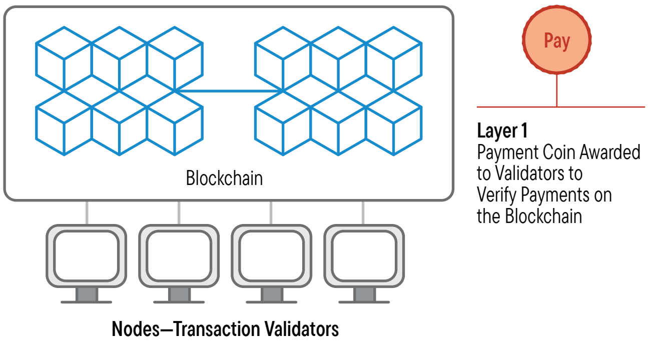
While Bitcoin was the first payment blockchain, it has not been the only one to emerge. Others, such as Dogecoin, Shiba Inu, Litecoin and others have subsequently emerged. Some, such as Dogecoin, take an opposite approach to Bitcoin and have abundant supplies of coins. Each platform has its own schedule for how their coin pool operates. Understanding each platform's approach is leading to a new form of economic value analysis called tokenomics.

Exhibit 19 provides a conceptual model of a crypto payment blockchain like the Bitcoin network.

Exhibit 19: Initial Cryptocurrency Offerings

2009 to Present

Payment Blockchains (e.g., Bitcoin)



Source: Franklin Templeton Industry Advisory Services. For illustrative purposes only.

As shown, the blockchain itself holds the blocks of transactions that are chained together. The blockchain is distributed, which means that a copy is held in the records of each node that acts as a transaction validator for the network. Validating the transaction—using PoW on the Bitcoin network—earns the validator a reward that is paid in the native payment coin.

In the Bitcoin network, new bitcoins are minted to reward the validators for their work. In other payment networks, a pool of payment coins may be created when the network launches and be systematically released as a reward. Coins are deposited into the validators' wallets that reside on the blockchain network. Validators and holders of coins can transfer these coins to other parties at will so long as the addresses of the wallets are verified and the sender is confirmed to possess enough coins to cover the transfer amount. Upon verification, the coins are moved immediately from the sender to the receiver.

While these decentralized peer-to-peer payment networks were revolutionary at the time, in terms of the development of the broader crypto domain, these offerings proved to be only the foundation for a more transformational cycle of innovation that followed.

Ethereum and digital ecosystems

In a 2014 message, Ethereum's founder, Vitalik Buterin, wrote, "Welcome to the new beginning." He goes on to note that the designer(s) of Bitcoin "desired to test two parameters—a trustless decentralized database enjoying security enforced by the austere relentlessness of cryptography and a robust transaction system capable of sending value across the world without intermediaries. Yet, the past 5 years have painfully demonstrated a third missing feature: a sufficiently powerful Turing-complete scripting language."[228]

He went on to announce his plans for the Ethereum network. "Ethereum," he explained, "is a modular, stateful, Turing complete contract scripting system married to a blockchain." The goal of the model was "to provide a platform for decentralized applications—an Android of the cryptocurrency world, where all efforts can share a common set of APIs, trustless interactions and no compromises."[229]

Put more simply, Ethereum set out to build a new open-source development platform where programmers could build a whole array of applications to operate in a decentralized manner. His model for this vision was the android platform that launched in 2007 by the Open Handset Alliance, a group of prominent companies that includes Google, HTC, Motorola, Texas Instruments and others. Android is the operating system that sits inside 2.5 billion active devices and is also an open development platform available to anyone—developers, designers and device makers.[230]

The big difference between android and Ethereum, however, was that the transactions that took place within the set of apps being built on Ethereum would be decentralized, trustless peer-to-peer transactions that would need to be verified in a decentralized manner and recorded on a blockchain. Android app transactions are done using fiat currency, and transactions are enabled over traditional payment and banking rails.

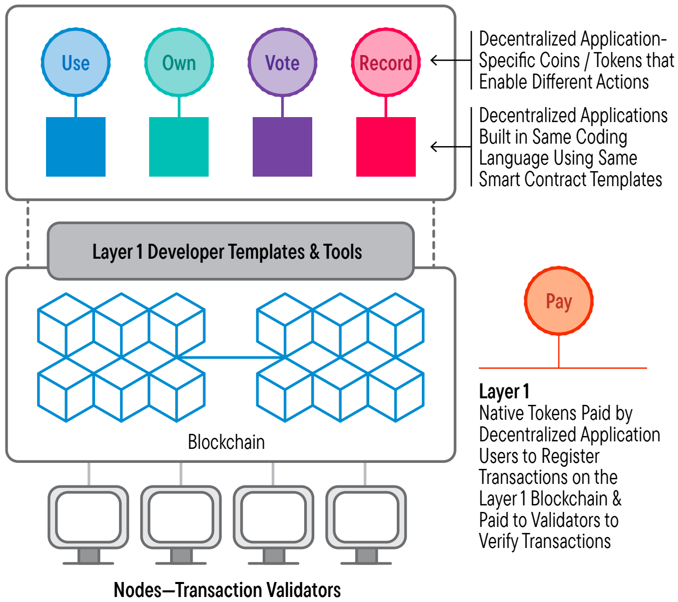
Ethereum was not building a payment blockchain, it was building a development platform and app ecosystem that ran on a virtual computer and sat on top of a payment blockchain. This is illustrated in Exhibit 20 on the next page.

The architecture of digital ecosystems like Ethereum is quite different than the design of payment blockchains. Most obviously, there are now two parts to the environment.

Exhibit 20: Next Generation Cryptocurrency Offerings

2015 to Present

Digital Ecosystems (e.g., Ethereum)



Source: Franklin Templeton Industry Advisory Services. For illustrative purposes only.

The first is where the blockchain and the digital wallets with the platform's native payment token transactions reside, like in the payment network. However, the L1 in Ethereum and other digital ecosystems has another tier that does more than store payment transactions. It is a virtual computer. It both holds the back-end logic for all the decentralized apps built on the platform as well as all the transactions done with those decentralized apps.

Ethereum defines these transactions as "cryptographically signed instructions from the accounts." There are two types of these transactions—one that creates a message and one that creates a contract. The contract is a collection of code that defines its function, rules and the data required to determine its state (executed or not executed). In essence, the contract is a program that runs on the network. When a message call is sent to the transaction account, the code in the contract is automatically executed, per its pre-coded instructions. As such, it is known as a smart contract.[231]

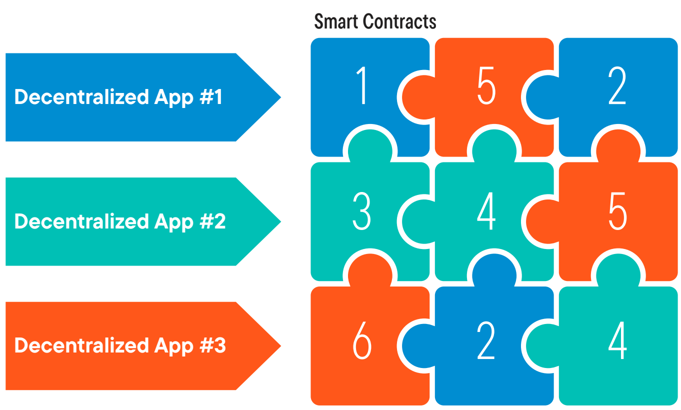
Smart contracts are not controlled by any user. They are deployed to the network and then run automatically as programmed. Like a regular contract, smart contracts can define rules, but instead of enforcing them through a centralized legal system, they enforce them via the code. Smart contracts cannot be deleted or amended, and interactions with them are irreversible. They are also built with open APIs so that they are reusable by others. For example, a smart contract may already be deployed on the network and has been programmed to initiate a payment and transfer money from one account to another account on the first of every month. A developer looking to build a new program to rent a property can simply insert that smart contract into the app rather than having to reprogram another smart contract to do the same thing. This ability to use other bits of preprogramed code and insert it into a new program is called composability and is illustrated in Exhibit 21.

Ethereum and other digital ecosystems also publish certain types of smart contracts as templates that all the developers on the platform use for similar types of activities. Such templates are called digital primitives because they can be repurposed to provide many different functions, but they are still recognizable and can interoperate with any other incarnation of that same form. For example, the ERC20 (Ethereum request for comment 20) contract on the Ethereum network allows decentralized app developers to create their own token. Any token created with an ERC20 contract can be recognized and can interact with any other token created on the same template. This allows apps to interact with each other and create new types of processes.

There are many types of ERC20 tokens within Ethereum and similar interoperable tokens on other digital ecosystems. Broadly speaking there are:

- currency tokens designed to work as a means of payment or exchange;
- security tokens designed to represent ownership shares in a project or decentralized app;
- utility tokens that entitle the holder to obtain the services of a specific decentralized app;

Exhibit 21: Smart Contract Approach to Application Development



Source: Franklin Templeton Industry Advisory Services. For illustrative purposes only.

- asset-backed tokens that grant rights to a specific real-world asset; and
- governance tokens that entitle holders to vote on matters affecting a specific project or decentralized app.

Each of these token types can be recognized and utilized by the decentralized apps within the same digital ecosystem. For example, a *currency token* such as ETH (the payment token on the Ethereum network) can be loaned to a decentralized lending app in return for a pre-negotiated interest rate that will be paid to the original holder of the ETH. In turn, the lending app can then issue the user that loaned it the ETH a new *security token* that represents the combined value of the ETH it tendered plus the interest it will be earning. Individuals can hold on to that security token and sell it back to the lending protocol when they want their ETH returned. Alternatively, users could exchange that security token to obtain their favorite decentralized gaming app and be given a *utility token* that entitles them to play their favorite game. Because the contracts are interoperable, the underlying virtual computer can tie these transactions together and retrace them all the way back to the original loan of the ETH.

With this set of capabilities, decentralized digital ecosystems have now enabled the final part of the vision of the *Crypto Anarchist Manifesto*—"two persons may exchange messages, conduct business, and negotiate electronic contracts without ever knowing the true name or legal identity of the other."[232]

> **"Nonetheless, the growth of the entire cryptocurrency ecosystem has been impressive. As of August 10, 2022, the market value of all cryptocurrencies was US$1.1 trillion. There were 11,869 decentralized applications live within the various digital ecosystems, which had 254,721 open smart contracts. In the preceding 24 hours, the volume of cryptocurrency traded equated to US$76 million and 1.63 million unique users logged into decentralized apps and spent US$21.7 million on transactions."**

In introducing this section, we noted that this set of technologies is emerging and are still in a proof-of-concept stage. Indeed, Bitcoin at best can be considered a teenager as it approaches its 14th birthday, and Ethereum has been around since only 2015. Competitors to Ethereum such as Polkadot, Avalanche, Solana, Cardano, Stellar, and Ethereum 2.0 have been around for an even shorter period of time.

Nonetheless, the growth of the entire cryptocurrency ecosystem has been impressive. As of August 10, 2022, the market value of all cryptocurrencies was US$1.1 trillion.[233] There were 11,869 decentralized applications live within the various digital ecosystems, which had 254,721 open smart contracts.[234] In the preceding 24 hours, the volume of cryptocurrency traded equated to US$76 million[235] and 1.63 million unique users logged into decentralized apps and spent US$21.7 million on transactions.[236]

Based on the developments that have already occurred, it is possible to state now that a completely new online paradigm is emerging. Just as there was a significant change in approach and ability that exemplified the demarcation of Web1 from Web2, there is a new set of offerings and behaviors emerging that are creating the foundation of Web3.

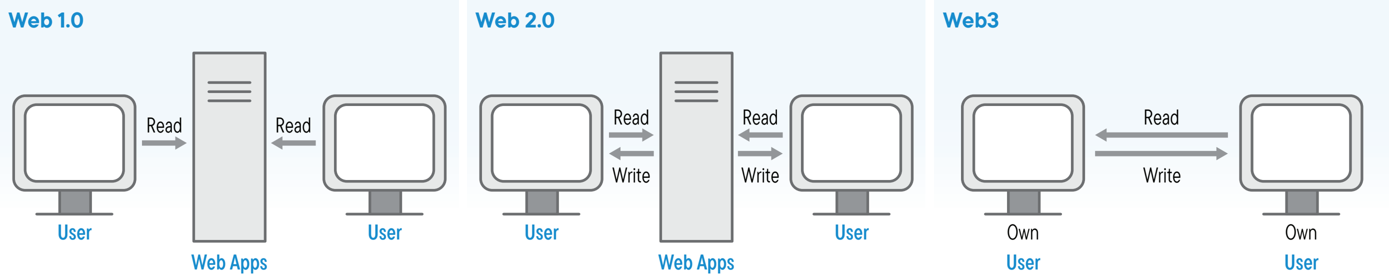
The shift from Web2 to Web3

Web3 marks an evolution in the ways that the users of the internet and mobile technologies can interact with each other. One venture capitalist described it as "the internet owned by the builders and users and orchestrated by tokens."[237]

Exhibit 22 on the next page provides a simple schematic to illustrate how this differs from both Web1 and Web2.

Web1 offered users an ability to come to a centrally hosted website and see the content listed there or engage with the functions offered (read). It was a one-way communication. In Web2, the communication became two-way with users also able to contribute content and interact with the content being created by others (write). This two-way interaction was still facilitated by a centrally hosted web-based platform. Web3 removes that central intermediary. Web3 enables users to read and write directly one-to-another via commercial protocols (smart contracts offered by decentralized apps on digital ecosystem platforms) and to purchase, earn and own stakes in those protocols.

Exhibit 22: Shifting Nature of User Engagement with Web Apps



Source: MacManus, Richard. "Web3 Architecture and How It Compares to Traditional Web Apps." The New Stack web site. October 4, 2021. For illustrative purposes only.

Several core principles define Web3:[238]

- **Decentralized**—Instead of large swathes of the internet being controlled and owned by centralized entities, ownership gets distributed across the builders and users of the ecosystem;
- **Permissionless**—Everyone has equal access rights or permissions to participate in Web3 and no one gets excluded;
- **Trustless**—Actors in the ecosystem operate using incentives and economic mechanisms instead of relying on trusted third parties; and
- **Native payments**—Cryptocurrency is used as money as opposed to fiat currencies, which eliminates the need for banks and payment processors.

There are many tangible changes that differentiate the way that Web3 is built and operates as compared to Web2.

Web3 infrastructure differs from Web2

- In Web2, websites and apps are run on multiple centralized servers that are either owned or rented as a service from central entities. Each app provider creates a proprietary business logic and chooses how much or how little of the code to share with others.
- In Web3, offerings are run on virtual computers that are attached to a blockchain network. Apps are built using smart contracts that are designed to be composable and interoperable with all other apps in the ecosystem.

Web3 transactions occur differently than in Web2:

- In Web2, transactions are processed and executed by each individual platform or app. Payments are facilitated by a set of third parties that tie directly into the rails of government-sponsored banking systems that rely on messaging protocols that flow between all the various counterparties to affirm, enact and confirm each transaction and its accompanying set of instructions. Payments are made in government-issued currencies, also known as fiat currencies.
- In Web3, transactions rely on peer-to-peer networks that verify transactions via consensus mechanisms—such as PoW—in exchange for rewards. Payments are made using digital currency that is created, managed and destroyed by computer protocols inside the ecosystem according to a proscribed set of rules, and payments are authorized by cryptographically encoded digital signatures that trigger an automatic and instantaneous movement of digital currencies or assets between the sender's and the receiver's accounts.

Web3 application development happens differently than in Web2:

- In Web2, developers build and deploy applications that run on a dedicated cloud-based or local server, deliver an integrated front-end, and collect and manage their own data, often hosting it with a contracted third party. They can choose from a broad set of programming languages, and it is up to the developers to decide if their code will be kept proprietary or be shared with a broader community. To use functionality from an app, programmers must develop an API to call their functionality.
- In Web3, the back-end programming for an app is done using smart contracts, with each having its own API. These contracts are written in the same or compatible programming languages. App developers program their envisioned functions using commonly agreed templates. The program and all transactions are stored on the network's virtual computer. For programs written on the specific types of smart contract templates, the source code is reusable and interoperable across all decentralized apps using those same templates. The front-end user interface for decentralized apps is coded separately, can

be written in any programming language, and can be hosted on decentralized storage solutions or within cloud-based or local servers.[239]

In addition to the above-mentioned differences, Web3 also introduces new considerations that are not enabled by Web2.

- **Ownership:** In the Web2 world, digital assets are contributed by and associated with specific creators, but the user cannot establish ownership rights over those assets. Web3 gives a user an ability to directly own digital assets. A digital asset can be embedded into a smart contract and become a tradable token with a built-in set of rules that govern what rights and obligations are allowed regarding the asset. The contract moves with the asset wherever the asset sits. If the collection of payments is allowed by the smart contract, such payments would be initiated and automatically deposited into the digital wallet of the asset owner. If certain financial penalties are required because objective conditions have not been satisfied, the smart contract would automatically assess or collect such penalties. The asset itself can be transferred by the owner at will from one to another of their own wallets. Alternatively, ownership of the token can be exchanged to another individual or group, in which case the ownership of the asset and all the associated rights and obligations would be automatically re-registered to the new owner(s).

- **Censorship resistance:** In the Web2 world, there is a power imbalance between the platform and its content creators. Digital assets are held on the platform's servers and the data related to those assets is housed in the platform's databases. If users are seen as violating the platform's rules, they can be barred from accessing the platform. The digital assets they created and the data that resulted from interactions with the asset are lost to the users. Moreover, the data that the users generate is often leveraged by the platform itself with little or no compensation given to the users. In Web3, establishing ownership over their digital assets by embedding them into tokens with smart contracts enables the holders to not only own, but access all the data related to their assets since it would sit in their digital wallet on the blockchain networks—a publicly accessible utility.

- **Utility:** In Web2, certain business models that facilitate services have utility such as securing a ride via a platform, but digital assets that are posted onto platforms offer no such utility. In Web3, smart contracts can also be used to convey certain rights or invoke certain guarantees to token owners beyond the exchange of monetary flows. Smart contracts associated with a token may automate and standardize a multitude of copyright-related transactions; for instance, those authorizing the use and exploitation of copyright-protected content and remuneration such as royalties.[240] Tokens can be used to encapsulate, enforce and transfer property rights; for instance, the registration of land and record of ownership title.[241] Tokens can be used to create, validate and administer digital identities; for instance, creating a repository of identity data that supports proof of a person's unique identity.[242] Tokens can be used to control and administer access to certain personal data that users wish to make accessible to third parties in exchange for monetary compensation. Tokens can grant special rights, such as an ability to gain access to an event, to receive special products or to secure special services.

- **Governance:** Web3 also has given rise to a new type of decentralized autonomous organization (DAO). These structures issue tokens that grant holders an ability to participate in the coordination and decision-making around commercial protocols. DAOs are peer-driven forums for managing and evolving decentralized businesses. The digital web2 world is dominated by powerful networks. Traditional networks typically follow a standard trajectory: acquire users for the network by adding value, reach critical network effects, start extracting value. At any network's terminal point, the corporate model of maximizing shareholder profit is fundamentally at tension with increasing user benefit. In a Web3 terminal point, a user/community-led governance structure can better align incentives between the network and the user.[243]

Envisioning the peer-to-peer Web3 world

Explaining what these new capabilities can do is not equivalent to knowing how they will change behavior and force a further re-architecting of the way that commerce is delivered. Simply telling people in 2006 that they could build friend networks, post content, comment on content, receive news and join communities does not capture the impact that social media has wrought. There are, however, a few business models that have emerged in recent years that give a hint as to how the new Web3 world may evolve.

In 2007, two housemates in San Francisco were struggling to pay their rent and came up with an inventive way to make some extra money. The two noticed that a local industrial design conference had booked up the surrounding hotels.

“Having the platform at the center to intermediate these interactions might not be needed if there were a way for the service provider and the service consumer to contract directly with each other. One could easily imagine a decentralized application facilitating such a transaction in a trustless manner where the smart contract takes care of the contract fulfillment and the payment.”

They put down three air mattresses and offered a bed and some breakfast for any designers that needed a place to stay for the conference. They advertised their availability through a website they built called AirBedandBreakfast.com. After doing another test case by renting rooms around the Democratic National Convention in Denver and refining its offering as part of the prestigious startup accelerator Y Combinator, the company simplified its name to Airbnb and re-launched in March 2009.[244] By 2021, Airbnb had six million active listings, earned their hosts US$34 billion in revenues, and generated worldwide corporate revenues of US$6.0 billion.[245]

In 2008, two friends that were both successful entrepreneurs who had sold businesses they had co-founded for significant sums of money were attending a tech conference in Paris. One winter evening, the two had attempted, but were unsuccessful, in trying to hail a cab. They ended up wishing that they could just request a ride from their phone. In March 2009, they began development of a smartphone app that let people tap a button and get a ride. On July 5, 2010, the first Uber rider requested a trip across San Francisco.[246] In the fourth quarter of 2021, Uber drivers and couriers earned an aggregate US$9.5 billion across a total of 1.77 billion trips, the platform had 118 million active users, and the company generated worldwide corporate revenues of US$5.8 billion.[247]

In 2012, a former Amazon employee that had been working in the company’s fulfillment shipping services saw the need for an app specifically made for grocery shopping. Working with some friends as part of the Y Combinator incubator, they came up with their idea for a hyper-local, on-demand grocery delivery service that links consumers with personal shoppers. Instacart was born. Customers can use the mobile app to choose their preferred grocery store, browse through grocery items, fill a shopping cart and confirm their order. Shoppers are self-employed or part-time Instacart employees who obtain the order, shop for the products requested by the customer and deliver them to the customer’s door.[248] In 2021, Instacart had more than 10 million users on the platform, had 500,000 shoppers, was connected to more than 45,000 stores, and had a valuation of US$39 billion.[249]

People often refer to these companies as part of the “sharing economy.” But who is doing the sharing? In each instance, the company provides no more than the platform to arrange the transaction. The services these companies provide—renting a room, securing a ride, getting groceries delivered—are performed peer-to-peer. The supplier of the service is an individual. Hosts rent their rooms in their own home. Drivers utilize their own cars to provide rides. Shoppers devote their own time to go to the store, shop, and deliver the groceries. In addition, the users of the service are individuals. Travelers use the rooms they rent on Airbnb. Riders secure transportation to their destination via Uber. Customers receive the groceries that they order and pay for them via Instacart.

Having the platform at the center to intermediate these interactions might not be needed if there were a way for the service provider and the service consumer to contract directly with each other. One could easily imagine a decentralized application facilitating such a transaction in a trustless manner where the smart contract takes care of the contract fulfillment and the payment. Moreover, both the service provider and the consumer could be a part of the communities that govern the decentralized apps that they use frequently to have a say about how the platforms operate, even going as far as choosing to distribute some of the profits generated by renting rooms, giving rides, or getting groceries to the individuals that make up the ecosystem—service providers and consumers.

Indeed, we may soon see commercial protocols—the way to access and utilize decentralized apps—begin to compete with commercial platforms. Already, there are decentralized messaging apps, decentralized gaming platforms, decentralized exchanges, decentralized virtual real estate communities, and more emerging in the Web3 environment. These early offerings give a glimpse of what is possible as Web3 moves out of its proof-of-concept stage and into a period of general adoption.

Conclusion and preview of part II

Technology innovation has been a powerful force reshaping the way that enterprises, governments and institutions operate over the past 50 years. One of the most important streams of this work has been the growing ability to use computational power and the networks they enable to automate business interactions, support the creation of intellectual property, access and analyze information, process transactions, facilitate communication and pursue entertainment.

These goals have remained similar for each of the four cycles of commercial technology innovation we discuss in the paper. What has changed has been the sophistication of the technologies that are being used to achieve those aims. Each cycle of technology innovation we discuss in the paper—automation, digitization, virtualization and decentralization—provide new offerings with enhanced abilities. Each time these technologies advance, those pursuing commerce have had to alter their approach to utilize the new functions, and those responsible for delivering the infrastructures that enable such commerce have been forced to re-architect the organization's technology to take advantage of new capabilities.

The speed of change over the past 50 years has been impressive as innovation has followed innovation. From early computers that took up entire rooms and required hours to run a single program that was hand-fed into the system by punch cards to today's cloud-computing platforms, where developers can use APIs to subscribe to services from a mobile phone, has already been an incredible progression. Early glimpses of what the emerging Web3 landscape might offer seem to indicate that another paradigm shift is at hand that will foundationally change the way that societies operate.

This paper has focused on the technologies themselves—how they started, developed, iterated and changed over the years. It has also laid out how the deployment of these technologies has shifted, moving from monolithic systems to tiered client/server offerings, to service-oriented and then microservice architectures. For the next cycle of innovation, we have tried to draw out how the design, delivery and operations of the Web3 world might require a growing focus on what having self-executing, composable and interoperable code that sits within self-contained ecosystems might mean for the delivery of services.

In looking across the four cycles of technology progression, we have also begun to lay out a story of how the growing set of capabilities enabled by technology have changed the offerings and behaviors of enterprises providing commercial engagement and individuals consuming those opportunities. Thus far, our exploration of those topics has been cursory. What Part II of this series will lay out is our hypothesis that there has been a discrete set of megatrends that began and has become amplified across each successive cycle of technology innovation already completed and is likely to reach its ultimate expression in the next cycle of decentralization.

The five megatrends have been driven by technology, but their influence goes far beyond the mechanisms by which they are enabled. Indeed, these megatrends have already altered the societies we live in today and are likely to shape the societies that emerge in coming years in even more profound and disruptive ways. A preview of the five megatrends is provided on the next page.

"In looking across the four cycles of technology progression, we have also begun to lay out a story of how the growing set of capabilities enabled by technology have changed the offerings and behaviors of enterprises providing commercial engagement and individuals consuming those opportunities. Thus far, our exploration of those topics has been cursory."

Democratization of access

Access to the rails of commerce by which enterprises offer their goods and services is shifting from proprietary to open architecture, allowing for a growing set of direct transaction opportunities that offer both individuals and enterprises unprecedented control over their own buying and selling channels.

Decomposition of business delivery

The attributes that define business value are shifting as organizations realize increasing opportunities from their intangible as well as tangible assets and determine more ways to reach customers by offering variants of their core competencies through a growing set of partnerships, service relationships and affiliations.

Expanding power of the crowd

The ability for individuals to connect one to another regardless of geographic location or socioeconomic status and coalesce around topics of common interest and concern is creating a group voice that is becoming amplified in ways that shape the supply and demandfor goods and services, leading to a new set of crowd factors that are helping to redefine value in the consumer economy.

Institutionalization of the individual

Opportunities for individuals to forge a social or personal brand identity, leverage a growing set of their personal assets, and obtain both the knowledge and access to optimize their personal finances are allowing them to operate in a more strategic manner that positions them to manage their own life as a cause that requires the marshalling of resources to achieve long-term goals.

Quantification of behavior

The combination of access to increasingly powerful computational processing and analytic tools together with the growing body of data being generated and collected through commercial and personal interactions has led to new types of analysis that look to extract insights from patterns of behavior and apply those findings to optimize the delivery of goods and services in an increasingly personalized and tailored manner.

Part II of this series will delve into each of these megatrends and show how they emerged and are becoming amplified across each of the technology cycles, already transforming society in profound ways and potentially providing clues as to how the next epoch of technological opportunities may reshape the ways that we live, work, engage, and invest.

Endnotes

1. Source: Brady, Will, and Elkner, Jeffrey. "History Of Information Technology." Introduction to Information and Communication Technology. February 8, 2019.
2. Ibid.
3. Source: "The Engines." Computer History Museum web site.
4. Source: "Ada Lovelace." Computer History Museum web site.
5. Source: "The Engines: Babbage Engine." Computer History Museum web site.
6. Source: Craig, William. "The History of Computers—A Timeline From 1800s to Now." Webfx.com. July 21, 2021.
7. Ibid.
8. Source: Arzoomanian, Rich. "A Complete History of Mainframe Computing." Tom's Hardware web site. June 26, 2009.
9. Source: "Electro Mechanical Computers." Kullabs web site.
10. Ibid.
11. Source: Craig, William. "The History of Computers in a Nutshell." Webfx.com. July 21, 2021.
12. Ibid.
13. Source: Brockmeier, Erica. "The World's First General Purpose Computer Turns 75 Today." Penn Today. February 22, 2021.
14. Source: Arzoomanian, Rich. "A Complete History of Mainframe Computing." Tom's Hardware web site. June 26, 2009.
15. Ibid.
16. Ibid.
17. Ibid.
18. Source: "LEO." Wikipedia.
19. Ibid.
20. Ibid.
21. Source: Fabry, Merrill. "The Story Behind America's First Commercial Computer." *Time*. March 2016.
22. Source: Craig, William. "The History of Computers in a Nutshell." Webfx.com. July 21, 2021.
23. Source: Grad, Burton. "First-Hand: The First Commercial Computer Application at General Electric." Engineering and Technology History Wiki. December 2006.
24. Ibid.
25. Source: IBM Archives. "IBM 701."
26. Source: Arzoomanian, Rich. "A Complete History of Mainframe Computing." Tom's Hardware web site. June 26, 2009.
27. Source: Craig, William. "The History of Computers in a Nutshell." Webfx.com. July 21, 2021.
28. Source: Arzoomanian, Rich. "A Complete History of Mainframe Computing." Tom's Hardware web site. June 26, 2009.
29. Ibid.
30. Source: Bellis, Mary. "The History of the Transistor." ThoughtCo. July 3, 2019.
31. Source: "Core Memory." Centre for Computing History web site.
32. Source: "What is Core Memory?" Technopedia.
33. Source: "RAMAC, The First Magnetic Hard Disk. IBM100.
34. Ibid.
35. Source: Yost, Micah. "A Brief History of Software Development." Medium.com web site.
36. Source: Rowe, Dusty. "How the 1960s Shaped the Modern Technology Landscape." Betty Blocks, blog. January 16, 2020.
37. Source: "Computer Science: Microchip Technology." Encyclopedia.com.
38. Ibid.
39. Source: Garner, Robert, and Spicer, Doug. "IBM 1401: The Legendary Data Processing System." Computer History Museum web site. 2009.
40. Ibid.
41. Ibid.
42. Ibid.
43. Ibid.
44. Gavett, Gretchen. "Today's Automation Anxiety Was Alive and Well." Harvard Business Review. February 8, 2016.
45. Ibid.
46. Ibid.
47. Ibid.
48. Ibid.
49. Source: Kunde, Brian. "A Brief History of Word Processing (Through 1986)." Stanford.edu.
50. Source: "Internet History of the 1960s." Computer History Museum web site.
51. Source: "DEC's Blockbuster: The PDP-8." Computerhistory.org.
52. Source: "Chapter 2—The Evolution of Distributed Systems." Umbc.edu.
53. Source: Shneiderman, Dee. "Computers of the '70s." Classrom.com.
54. Source: Sanger, David. "Bailing Out of the Mainframe Industry." New York Times. February 5, 1984.
55. Source: Grabbe, Orlin J. "The Knowledge Explosion." Tomorrow's World Magazine. March 1971.
56. Source: Perry, Tekla S, and Wallich, Paul. "Xerox Parc's Engineers on How They Invented the Future." IEEE Spectrum. October 1, 1985.
57. Source: "Invention of the PC." History.com. Last updated February 8, 2019.
58. Ibid.
59. Ibid.
60. Source: "VisiCalc of Dan Bricklin and Bob Frankston Guide: History, Origin, and More." History-computer.com. Last updated January 17, 2022.
61. Ibid.
62. Source: Crandall, Rick. "Invention of Graphical Executive Information Systems (EIS)." © 2009. Rickcrandall.net.
63. Source: Dyck, Jeremey. "The Xerox Thieves: Steve Jobs & Bill Gates." *BC Digest*. October 5, 2019.
64. Ibid.
65. Ibid.
66. Ibid.
67. Ibid.
68. Source: "The PC: Personal Computing Comes of Age." IBM100.
69. Ibid.
70. Ibid.
71. Ibid.
72. Source: Sanger, David. "Bailing Out of the Mainframe Industry." *New York Times*. February 5, 1984.
73. Source: Dyck, Jeremey. "The Xerox Thieves: Steve Jobs & Bill Gates." *BC Digest*. October 5, 2019.
74. Source: Zisman, Michael. "Office Automation: Revolution or Evolution?" Library of the Massachusetts Institute of Technology. April 1978.
75. Ibid.
76. Ibid.
77. Source: Sanger, David. "Bailing Out of the Mainframe Industry." *New York Times*. February 5, 1984.
78. Source: "Local Area Networks (LANS)" Local Area Networks (LANS)." Referenceforbusiness.com.
79. Source: "Office Automation Systems." Encyclopedia.com.
80. Source: "The History of Databases." ThinkAutomation web site.
81. Source: White, Sarah. "What Is Enterprise Architecture? A Framework for Transformation." *CIO*. October 6, 2018.
82. Ibid.
83. Source: "The Client/Server Model - IBM Documentation." IBM web site.
84. Source: Chapter 2—The Evolution of Distributed Systems." Umbc.edu.
85. Source: "SAGE Computer System Explained—Everything You Need To Know." History-Computer.com. Last updated February 14, 2022.
86. Source: "Internet History of 1960s."Computer History Museum web site.
87. Source: "From ARPANET to the Internet." Science Museum web site. November 2, 2018.
88. Source: "ARPANET: The Internet in its Infancy." IONOS. March 19, 2018.
89. Source: Leiner, Barry M; Cerf, Vinton G; Clark, David D; Kahn, Robert E; Kleinrock, Leonard; Lynch, Daniel C; Postel, Jon; Robers, Larry G; and Wolff, Stephen. "A Brief History of the Internet." Internet Society. 1997.
90. Source: "ARPANET: The internet in its infancy." IONOS. March 19, 2018.
91. Ibid.
92. Ibid.
93. Ibid.
94. Source: "From ARPANET to the Internet." Science Museum web site, November 2, 2018.
95. Source: Zimmermann, Kim Ann, and Emspak, Jesse. "Internet history timeline: ARPANET to the World Wide Web." Live Science web site. April 8, 2022.
96. Source: Berners-Lee, Sir Tim. "History of the Web." World Wide Web Foundation.
97. Source: Austin, Ben. "The Growth of the Internet: From 1990–2019." Absolute.com. May 13, 2019.
98. Source: "How Many Websites Are There?" Websitesetup.org. Last updated 2021.
99. Source: Hardy, James. "The History of E-Commerce." History Cooperative. July 20, 2014.
100. Source: "The History of Ecommerce: How Did It All Begin?" Miva.com blog.
101. Source: "Our History." eBay Inc. web site.
102. Ibid.

103. Source: "The History of Yahoo, and How It Went from Phenom to Has-been." *FastCompany.* March 21, 2018.
104. Source: Duggan, Wayne. "This Day In Market History, April 12: The Yahoo! IPO." Benzinga. April 12, 2022.
105. Ibid.
106. Source: Hammonds, Keith H. "Scient's Near-Death Experience." *FastCompany.* January 31, 2001.
107. Source: "A Brief History of Enterprise Architecture: Lessons We Can Learn and Apply." Digital Works group web site.
108. Ibid.
109. Ibid.
110. Ibid.
111. Source: "What is SOA (Service-Oriented Architecture)?" IBM web site. April 7, 2021.
112. Ibid.
113. Ibid.
114. Source: "Compatible Time-Sharing System." Wikipedia en-academic.com.
115. Source: Varghese, Blesson. "History of the Cloud." BCS web site. March 19, 2019.
116. Source: Ritchie, Dennis. "Development of the C Language." Bell Labs/Lucent Technologies. 2003.
117. Source: Brodkin, Jon. "With Long History of Virtualization Behind It, IBM Looks to the Future." *Network World.* April 30, 2019.
118. Ibid.
119. Ibid.
120. Source: "A Brief History of the Internet." Stanford.edu.
121. Source: O'Connell, Brian. "History of Salesforce: Timeline and Facts." TheStreet. February 27, 2020.
122. Source: Bellis, Mary. "The History of Google and How It Was Invented." Thoughtco.com. Updated on January 29, 2020.
123. Source: "Distributed Systems and Parallel Computing." Google Research.
124. Source: Miller, Ron. "How AWS Came to Be." TechCrunch. July 2, 2016.
125. Ibid.
126. Source: Israel, Spencer. "APIs Are the Secret Sauce Of Fintech." Benzinga. November 17, 2020.
127. Source: Miller, Ron. "How AWS Came to Be." TechCrunch. July 2, 2016.
128. Source: "Amazon.com Launches Web Services; Developers Can Now Incorporate Amazon.com Content and Features into Their Own Web Sites; Extends ''Welcome Mat'' for Developers." Amazon.com, Inc.—Press Room. July 16, 2022.
129. Source: Miller, Ron. "How AWS Came to Be." TechCrunch. July 2, 2016.
130. Ibid.
131. Source: "Amazon Q2 Results: AWS Shines Once Again." Seeking Alpha. July 29, 2022.
132. Source: "SOA vs. Microservices: What's the Difference?" IBM. May 14, 2021.
133. Ibid.
134. Ibid.
135. Ibid.
136. Source: Cormode, Graham, and Krishnamurty, Balachander. "View of Key differences between Web 1.0 and Web2." FirstMonday.org.
137. Ibid.
138. Ibid.
139. Source: "What is Web2?" Iplocation.net blog. June 24, 2016.
140. Source: : Cormode, Graham, and Krishnamurty, Balachander. "View of Key differences between Web 1.0 and Web2." First Monday.org.
141. Source: Maldonado, Leonardo. "What's the Document Object Model, and Why You Should Know How to Use It." Freecodecamp.org. November 7, 2018.
142. Source: "What is Web2?" Iplocation.net blog. June 24, 2016.
143. Source: "Mobile Subscriptions 1990." WorldMapper.
144. Source: "Number of Smartphone and Mobile Phone Users Worldwide in 2022/2023: Demographics, Statistics, Predictions." Financesonline.com.
145. Source: Krizanovic, Ivana. "Cell Phone History: From the First Phone to Today's Smartphone Wonders." Versus.com. August 4, 2020.
146. Source: Galazzo, Richard. "Timeline from 1G to 5G: A Brief History on Cell Phones." CENGN web site. September 21, 2020.
147. Ibid.
148. Source: "A Brief History of 2G Mobile Communication Technology." Xoxzo Official blog. August 1, 2018.
149. Ibid.
150. Source: Krizanovic, Ivana. "Cell Phone History: From the First Phone to Today's Smartphone Wonders." Versus.com. August 4, 2020.
151. Source: Simon user manuals.pdf. Microsoft.com.
152. Source: Krizanovic, Ivana. "Cell Phone History: From the First Phone to Today's Smartphone Wonders." Versus.com. August 4, 2020.
153. Ibid.
154. Source: Chan, Ai Sin, "A Brief History of 2G Mobile Communication Technology. Xoxzo blog.
155. Source: Galazo, Richard. "Timeline from 1G to 5G: A Brief History on Cell Phones." CENGN web site. September 21, 2020.
156. Ibid.
157. Source: Artman, Jason. "Why Is the iPhone So Popular?" Chron.com.
158. Source: "CompuServe: The First Major Online Service (1979)." Retro Planet blog.
159. Source: Gagne, Ken and Lake, Matt. "CompuServe, Prodigy et al.: What Web2 Can Learn from Online 1.0." *Computerworld.* July 15, 2019.
160. Ibid.
161. Ibid.
162. Ibid.
163. Ibid.
164. Ibid.
165. Ibid.
166. Ibid.
167. Source: Press, Gil. "Why Facebook Triumphed Over All Other Social Networks." Forbes. April 8, 2018.
168. Source: "The Evolution of Social Media: How Did It Begin and Where Could It Go Next?" Maryville University.
169. Ibid.
170. Source: Press, Gil. "Why Facebook Triumphed Over All Other Social Networks." Forbes. April 8, 2018.
171. Source: "The History of Facebook: From BASIC to Global Giant." Brandwatch. January 25, 2019.
172. Ibid.
173. Ibid.
174. Ibid.
175. Source: "Twitter Timeline—Milestones Over 10 Years." Hush Digital blog. March 9, 2016.
176. Source: "How did Twitter start? What is the history behind Twitter?" Profilerehab.com.
177. Ibid.
178. Source: "Twitter Timeline - Milestones Over 10 Years." Hush Digital blog. March 9, 2016.
179. Source: Kormos, Jessica. "The Real History of Twitter, in Brief." Lifewire.com. November 25, 2020.
180. Source: Blystone, Dan. "What Is Instagram? Its History and How it Works." Investopedia.com. July 11, 2022.
181. Source: "How Pinterest Became an $11 Billion Company by Organizing the World's Hobbies." Product Habits blog.
182. Source: Mehrotra, Shikhar. "When Was Snapchat Created?" ScreenRant. February 6, 2022.
183. Source: "World Development Report 2021: Crossing Borders." WorldBank. March 24, 2022.
184. Source: "Cisco Annual Internet Report (2018–2023)." Cisco web site. March 9, 2000.
185. Ibid.
186. Source: "Total Data Volume Worldwide 2010–2025." Statista.
187. Source: "Global Datasphere: Storage Capacity 2025."Statista.
188. Source: "Historical Cost of Computer Memory and Storage." Our World in Data.
189. Source: "The Evolution of Social Media: How Did It Begin and Where Could It Go Next?" Maryville University.
190. Ibid.
191. Source: Elias, Jennifer. "How Google's $150 Billion Advertising Business Works." *CNBC.* October 13, 2021.
192. Source: "Small Business SEO in 2017: A Survey." Clutch.co.
193. Ibid.
194. Source: Elias, Jennifer. "How Google's $150 Billion Advertising Business Works." October 13, 2021.
195. Source: Park, Hyoun. "The Death of Big Data and the Emergence of the Multi-Cloud Era." KDnuggets blog.
196. Source: "The Google File System." Paper Trail blog. October 1, 2008.
197. Source: "Hadoop—MapReduce." Tutorialspoint.com.
198. Source: Apache Hadoop.
199. Source: "What Is Big Data and Hadoop - Raise the Bar & Be a Star." TechVidvan web site.
200. Source: Anyoha, Rockwell. "The History of Artificial Intelligence—Science in the News." Harvard University. August 28, 2017.
201. Ibid.
202. Source: Gladchuk, Veronika. "The History of Machine Learning: How Did It All Start?" Labelyourdata.com. July 5, 2020.
203. Ibid.

204. Ibid.
205. Source: "AlphaGo." Deepmind.com.
206. Ibid.
207. Source: FIPS 46-3, Data Encryption Standard (DES) (withdrawn May 19, 2005.) Nist.gov.
208. Source: Diffie, Whitfield and Hellman, Martin. "New Directions in Cryptography." IEE Transactions on Information Theory, Vol. IT-22, No. 6. November 1976.
209. Source: Chaum, David. "Security without Identification." Communications of the ACM, vol. 28 no. 10, October 1985 pp. 1030–1044.
210. Source: Lopp, Jameson. "Bitcoin and the Rise of the Cypherpunks." CoinDesk. Updated April 27, 2022.
211. Source: "The Crypto Anarchist Manifesto." Bitcoin.com. December 7, 2019.
212. Ibid.
213. Source: Ong, Jie Yee. "Who Are the Cypherpunks And What Are They Advocating For?" Makeuseof.com. May 25, 2021.
214. Source: Hughes, Eric. "A Cypherpunk's Manifesto." Activism.net.
215. Source: Thompson, Andi Wilson; Kehl, Danielle; and Bankston, Kevin. "Doomed to Repeat History? Lessons from the Crypto Wars of the 1990s." New America. June 17, 2015.
216. Ibid.
217. Source: Lopp, James. "Bitcoin and the Rise of the Cypherpunks." CoinDesk. Updated April 27, 2022.
218. Source: "US Export Control Laws on Encryption Ruled Unconstitutional." Electronic Frontier Foundation press release.
219. Source: Thompson, Andi Wilson; Kehl, Danielle; and Bankston, Kevin. "Doomed to Repeat History? Lessons from the Crypto Wars of the 1990s." New America. June 17, 2015.
220. Source: Hashcash.org.
221. Source: Dai, Wei. "B-Money." Satoshi Nakamoto Institute. November 1998.
222. Source: Finney, Hal. "Reusable Proofs of Work." Femmetic.net.
223. Source: Saadat, Syed Yusuf. "Bitcoin: Emergence and Evolution (Part 2)."CPD blog. October 10, 2019.
224. Source: "The History of Bitcoin: A Complete Timeline of the Start of Web3." History Cooperative web site. Latest updates November 15, 2021.
225. Source: Williams, Chris. "Bitcoin's Birthday: Satoshi Nakamoto's Hidden Message Explained." Crypto Briefing. January 3, 2021.
226. Source: Hayes, Adam. "What Is a Blockchain? How Does It Work?" Investopedia.com. Updated June 24, 2022.
227. Source: "How Bitcoin Works." Investopedia. May 11, 2022.
228. Source: Cuofano, Gennaro. "Ethereum: The History of Ethereum. FourWeekMBA. March 7, 2022.
229. Ibid.
230. Source: "What is Android?" Android web site.
231. Source: "Ethereum Virtual Machine (EVM). Ethereum.org. Last edit August 23, 2022.
232. Source: "The Crypto Anarchist Manifesto." Bitcoin.com. December 7, 2019.
233. Source: Cryptocurrency Prices, Exhibits and Market Capitalizations: CoinMarketCap.com.
234. Source: "Dapps Industry Overview." DappRadar.
235. Source: Cryptocurrency Prices, Exhibits and Market Capitalizations: CoinMarketCap.com
236. Source: "Dapps Industry Overview." DappRadar.
237. Source: MacManus, Richard. "Web3 Architecture and How It Compares to Traditional Web Apps." The New Stack. October 4, 2021.
238. Source: "What is Web3 and Why Is It Important?" Ethereum.org
239. Source: "What You Need To Know About DApp Development." OpenGeeksLab.
240. Source: Bodo, Balazs; Gervais, Daniel; and Quintais, Joao Pedro. "Blockchain and Smart Contracts: The Missing Link in Copyright Licensing?" International Journal of Law and Information Technology. Oxford Academic. September 25, 2018.
241. Source: Hansen, J. Dax; Rosini, Laurie; and Reyes, Carla L. "More Legal Aspects of Smart Contract Applications." PerkinsCoie.
242. Ibid.
243. Source: "DAOs and Web3 Governance: The Promise, Implications and Challenges Ahead." Antler. January 3, 2022.
244. Source: "The History of Airbnb." Keycafe blog.
245. Source: Airbnb Fourth Quarter and Full-year 2021 Financial Results: Airbnb press release. February 15, 2022.
246. Source: "The History of Uber—Uber's Timeline." Uber Newsroom.
247. Source: "Uber Announces Results for Fourth Quarter and Full Year 2021: Uber press release. February 9, 2022.
248. Source: "Instacart Startup Story: Delivering Groceries Instantly." Startuptalky.com. May 16, 2021.
249. Source: Thomas, Marques. "23 Revealing Instacart Statistics, Trends & Facts (2022)." Querysprout.com.

Notes

FRANKLIN TEMPLETON