



THE FUTURE OF SOFTWARE WILL BE HYPERPLEXED

By Pranay Ahlawat, Vikas Taneja, and Bernd Schlotter

IN AN INCREASINGLY DIGITAL world, companies understand that enterprise software architectures yield tangible competitive advantage. Enterprise software architectures are the underpinnings of every technology-driven organization; not just hardware and software companies but also any organization that is building digital capabilities. Software architectures drive companies' innovation capabilities, determine their economics of research and development, and influence their operating models and ability to grow. No wonder the choice of enterprise software architectures has become a strategic imperative in business today.

Over the last five decades, four major changes in enterprise software architectures have taken place. They've evolved from mainframe architectures to monolithic, tiered, and, most recently, cloud-native software architectures. (See the sidebar "A Timeline of Enterprise Software Architectures.") With the growing use of new technologies—such as artificial intelligence (AI), robotics, augmented reality (AR), and virtu-

al reality (VR), powered by the Internet of Things (IoT) and 5G networks—there's a pressing need for a new generation of software architectures.

Few companies realize that to be successful, they have to completely reevaluate the manner in which they build such software. The challenge is that many software applications nowadays are born distributed, having to work across multiple clouds, at the edge, and in application-specific devices. To build and scale such applications quickly, companies have to ensure their enterprise software architectures support new kinds of programming, deployment, and operational paradigms. That's why we believe that the future of software architectures will be hyperplexed, by which we mean that they will have to support many widely-distributed software applications, several different types of devices, and new user experiences and interfaces. (See the sidebar "Defining Hyperplexed Architectures.")

Hyperplexed architectures, which we will describe at length in this article, will allow

A TIMELINE OF ENTERPRISE SOFTWARE ARCHITECTURES

Enterprise software architectures have evolved four times over the last five decades because of new technological paradigms. (See the exhibit.)

Mainframe Architectures (before 1970). At first, engineers interacted with mainframe computers by writing programs on punch cards, paper tapes, and magnetic tapes, which led to the emergence of high-level languages such as COBOL (circa 1959) and BASIC (1964). Fortran was the first compiler that freed developers from working with assembly languages—low-level programming languages with a correspondence between the instructions in the language and the architecture’s machine code instructions—that demanded enormous effort to solve relatively simple computational and logic problems.

Monolithic Architectures (1970–1990). With continued advancement in semiconductors and as problems and information became more complex, minicomputers, UNIX, and object-orientated programming languages such as C++, which organized

software design around data rather than functions and logic, emerged. This resulted in architectures becoming monolithic. The user interface and the data access code were physically combined and logically linked in a single platform.

Tiered Architectures (1990–2010). The Internet’s growth led to the demand for scalable architectures and allowed different parts of the software to sit on different machines. The result was tiered software architectures, whose functionality could be decomposed into three layers: the presentation layer, the applications layer, and the data-access layer. The emergence of middleware allowed software developers to focus on applications while virtualization helped with better hardware utilization, increased automation, and, ultimately, the ability to offer PaaS (platform as a service) and IaaS (infrastructure as a service).

Cloud Native Architectures (2010–present). In the last decade, cloud-based applications, which are more agile, cost less to develop, and provide access to greater

Tomorrow’s Enterprise Software Architectures Will Be Hyperplexed

	Pre-1970	1970-1990	1990-2010	2010-present	Near future
	Mainframe	Monolithic	Tiered	Cloud Native	Hyperplexed
Key technology paradigms	<ul style="list-style-type: none"> • COBOL • Mainframe 	<ul style="list-style-type: none"> • Minicomputers • UNIX • Object-oriented • Relational databases 	<ul style="list-style-type: none"> • Web • Integration middleware • Web services • Virtualization 	<ul style="list-style-type: none"> • Cloud • Mobile • Containerization • Big data and analytics 	<ul style="list-style-type: none"> • Edge and IoT • Digital twins • AR and VR • Distributed AI
	Yesterday’s architectures were ...		Tomorrow’s architectures will be ...		
Modality	Highly centralized		Massively distributed		
Data and compute intensity	Low data and low compute		Big data and large compute		
Security surface area	Controlled security		Large security surface area		
Speed of change	Low velocity of change		High velocity of change		
Operations	Manual		Highly automated and data-driven		

Source: BCG analysis.

Notes: IoT=Internet of Things; AR= augmented reality; VR=virtual reality; AI=artificial intelligence.

A TIMELINE OF ENTERPRISE SOFTWARE ARCHITECTURES (continued)

computing resources, have proliferated. They're designed to handle big data and analytics, and have been optimized for rapid releases and DevSecOps. These applications are increasingly being decomposed into microservices that can be stitched together by service fabrics and meshes. Newer virtualization

technologies, such as networking overlays and containerization, provide the ability to automate these microservices and deploy them at scale.

What's next? We see hyperplexed architectures as the natural evolution of enterprise software architecture.

a step change in the development, deployment, operation, and maintenance of distributed software. They will spur innovative software applications and create new markets for distributed software, platforms, and software tools. Building software on hyperplexed architectures will also speed up the time to market and lower development costs. Finally, hyperplexed architectures will provide novel, and more personalized, real-time customer experiences.

Brace yourself for the new world of hyperplexed software architectures.

The Three Drivers of Change

Today, three forces are driving change in enterprise software architectures—two that originated outside the software development process and one from within it.

New Technology Paradigms. The wave of technological change we've just described has made it possible to develop many new business-centered and consumer-focused applications. The impact of those changes on software design, however, are twofold: one positive and one negative. On the one hand, the new technologies have developed platforms and tools that will help solve difficult problems cheaper and faster than ever before. On the other hand, they have greatly increased the complexity of software architectures, forcing companies to rethink how they develop software.

Changes in Customer Needs and Innovative Applications. Buyers' expectations are

changing rapidly. As the consumerization of the business enterprise continues, enterprise software applications are increasingly becoming AI- and data-driven. They're providing new and personalized B2B customer experiences and are unlocking disruptive applications such as self-guided robotic surgery and AI-driven logistics. Next-generation applications will need to support several interfaces such as voice, wearables, touch, AR, and VR in addition to web and mobile. Consumer applications, too, are changing, driven by real-time connectivity and new hardware. For instance, sales of mobile gaming, AR, and VR devices are expected to increase at a compound annual growth rate of 65% between 2020 and 2024 and the use of wearable devices, such as watches and eyewear, will rise sharply over the next five years.

Rise of Open Innovation. Never before has open innovation been as much a part of software development as it is today. Open source software has become central to IT strategies, influencing every layer of the software stack, from operating systems and programming languages to middleware and development tools. Meanwhile, commercial open source companies like Red Hat and Docker, along with cloud vendors, are turning open source software into viable commercial products. Together, these forces have commoditized software development, shifting the speed, scale, and economics of innovation by offering powerful infrastructure and platform capabilities without companies having to

DEFINING HYPERPLEXED ARCHITECTURES

No word captures the future of enterprise software architectures better than hyperplexed. An amalgam of the terms, hyper (meaning, extremely) and plex (which stands for the number of parts of one unit), hyperplexed symbolizes the fact that tomorrow's software architectures will have to support many widely-distributed software applications. Hyperplexed architectures will simplify software development and shorten the

time to market by providing native platform and run-time support for multicloud, edge, IoT, and application-specific hardware devices. They will also ease distributed operations through integrated telemetry, AIOps (artificial intelligence operations), visibility, and security. And they will support the next generation of multimodal and real-time user experiences such as AR, VR, voice, and touch.

sink capital up front. In addition to leveling the playing field with startups, open innovation has forced digital giants to embrace open source software and standards.

These three forces, which are coalescing on an unprecedented scale, reinforce each other. New technologies, accompanied by novel platforms, will catalyze fresh applications and digital innovations—which, in turn, will drive the need for newer technologies, and so on. This virtuous cycle reinforces the need for hyperplexed software architectures. (See Exhibit 1.)

Characteristics of the New Architectures

Unsurprisingly, new software designs, platforms, and frameworks have emerged to address the next generation of highly distributed applications. Next-generation cloud-based IoT platforms and mobile backend as a service (MBaaS) offerings, such as Firebase, provide capabilities such as offline data sync and push notifications; next-gen data platforms, such as IBM Cloud Pak for Data, help with data integration, virtualization, and distributed AI use cases; and the current generation of data middleware vendors, such as Snowflake and MongoDB Atlas, provide support for application workloads that span multiple clouds.

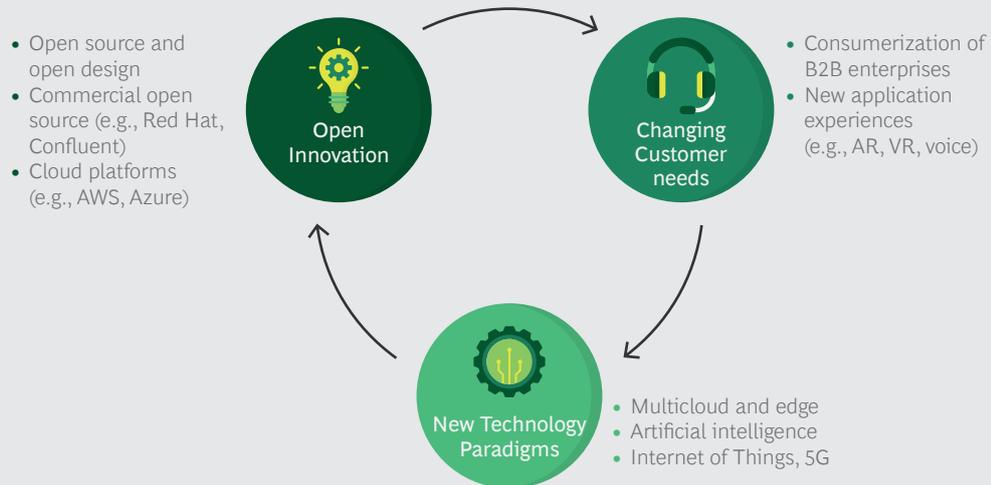
Despite these developments, however, our research and in-depth conversations sug-

gest that even the latest platforms, cloud-native solutions, edge, and IoT platforms have numerous shortcomings. They tackle only bits and pieces of the problem that companies face, leaving large gaps in development tooling, security, and monitoring, among other things. Companies are investing a great deal of time and money to integrate solutions, but they're unable to scale innovations quickly.

Our studies show that hyperplexed architectures will help overcome those deficiencies by offering six integrated capabilities. (See Exhibit 2.)

- 1. They will provide distributed platforms and control fabrics.** Hyperplexed architectures will usher in new platform capabilities, supercharging the efficiency of software developers and shortening applications' time to market. They will go beyond offering microservices and IoT frameworks to become the foundation on which companies deploy applications and services across heterogeneous infrastructures and the cloud. They'll also provide next-generation run times and frameworks to help with advanced use cases such as distributed AI and real-time communications. The new architectures will provide transparent error handling, monitoring, and continuous integration/continuous deployment (CI/CD) pipelines for software development. Together, the next-generation platform capabilities will enable

EXHIBIT 1 | The Three Drivers of Hyperplexed Architectures



Source: BCG analysis.

Note: AR=augmented reality; VR=virtual reality.

developers to build scalable software by allowing them to focus on business tasks, rather than worry about writing, piecing together, and maintaining fragmented frameworks and building blocks.

2. **They will work transparently from the cloud to the edge and things.** Today, companies run hybrid applications across multiple cloud and on-premise data centers, but soon they will have to go beyond the current boundaries. As edge computing grows, it will entail running fewer processes in the cloud and moving more of them to edge data centers, users' computers, IoT devices, and edge servers, which will increase software architectural complexity.

Hyperplexed architectures will allow software developers to leverage distributed infrastructure footprints transparently and natively, akin to what hyper-local cloud zones and modern content delivery networks are doing for latency-sensitive applications and gaming. Modern software applications are increasingly running on specialized hardware platforms such as wearables and VR headsets, application-specific integrated circuits and accelerators for AI and

inferencing workloads, as well as field-programmable gate arrays for specialized use cases such as software-defined networking devices. Hyperplexed architectures will abstract much of those applications' complexity and equip purpose-specific hardware devices seamlessly with software. In the future, every software stack will treat edge devices and hardware platforms as first-class citizens.

3. **They will support multimodal experiences.** Emerging software applications increasingly support novel user experiences such as voice, AR, VR, wearables, and touch. Gaming and front-end frameworks, such as Unity 3D and React 360, are already changing to support these experiences. In addition, driven by 5G networks, tomorrow's applications will see the growth of low-latency use cases such as real-time predictive analytics and connected gaming. Hyperplexed frameworks will support these experiences and handle the user-experience challenges of real time accessibility, push notifications, localization, and multimodality.
4. **They will manage complex security challenges.** Hyperplexed software

EXHIBIT 2 | Hyperplexed Architectures Will...

...Support multimodal experiences

AR and VR



Voice



Mobile 2.0 and Touch



Wearables



...Catalyze new ways of working

Systems engineering



Data-driven product management



Agile 2.0



Open innovation



...Provide distributed platforms and control fabrics



- Next-gen run times and frameworks
- Transparent error handling and monitoring
- CI/CD pipelines

...Use AI- and data-driven operations



- Integrated telemetry and monitoring
- Policy-driven compliance and governance
- AIOps and automation

...Manage complex security challenges



- Zero-trust security design
- Robust identity and access management
- Transparent encryption and certification management

...Work transparently from the cloud to the edge and things

Cloud and edge



On-premise and hybrid



Things



Specialized hardware



Source: BCG analysis.

Notes: AR=augmented reality; VR=virtual reality; CI/CD=continuous integration/continuous deployment.

architectures will have much larger attack surface areas. The highly distributed nature of the software and the use of open source software will make applications harder to lock down, and security will have to be an integral part of the software stack. The new architectures will embrace zero-trust security design, provide robust identity and access management, and make encryption transparent. For example, they will deploy well-articulated and integrated security models to reduce risk, building in features such as security operations

and AI-driven anomaly detection into software platforms.

5. They will use AI and data-driven ops.

The number of connected devices, the scale of telemetry data collection, and the scope of computation will demand that operations are kept cost-effective. Doing that will require specialized tools and AI-driven operations. Although these are early days yet, digital giants and a number of small startups are innovating rapidly, and a slew of AIOps (artificial intelligence operations)

offerings are beginning to arrive in the market.

Hyperplexed software will provide integrated monitoring and telemetry and allow operations and IT teams to steer policy-driven compliance and governance. They will preemptively flag risks and identify the causes of problems based on databased insights. As the algorithms learn, companies will use AI to automatically optimize the configuration of applications, services, and IT environments when conditions change.

- 6. They will catalyze new ways of working.** Hyperplexed applications will unlock multiple new use cases and cause a paradigm shift in how software applications are built, tested, and operated. Companies therefore will have to invest in building new skills as well as transforming traditional teams and their ways of working. The demand for new engineering disciplines, such as systems engineering and combined hardware-software engineering, will increase as will the use of data-driven innovation. Tomorrow's platforms will accelerate the trend of more social and collaborative development. That will enable software firms to create products rapidly, be leaner, and employ ultralight software-development methods.

The Emerging Enterprise Software Options

Three trends are shaping the development of the next generation of software architectures.

One, industry consortiums—such as the Storage Networking Industry Association and the Internet of Things Consortium—will help drive the development of new standards and blueprints. In the past, these bodies have been instrumental in bringing industry players together in order to establish standards such as Bluetooth, Java Enterprise Edition, and the Common Information Model.

Two, open source innovation and community-driven efforts will continue to play a

major role in the evolution of enterprise software, just as they did in the development of mobile, cloud-native, and AI applications. Open source foundations, such as the Linux Foundation, the Cloud Native Computing Foundation, and the Apache Foundation—in partnership with cloud giants such as AWS, Google, Alibaba, IBM/Red Hat, and Microsoft and hardware giants such as NVIDIA/Arm, Tesla, and Apple—will drive the development of the next generation of platforms, run times, and application frameworks.

Three, commercial innovation will drive the specialization, verticalization, and improvement of open source solutions. The latter will appear in the form of proprietary software, such as Oracle's and IBM's development of early relational database management systems; and commercial open source software, such as the contribution of Red Hat and Databricks to systems software and analytics platforms, such as Linux and Spark.

Meanwhile, in 2014 Google led a consortium of a group of companies including Red Hat, IBM, and Microsoft in open sourcing Kubernetes, an extensible container orchestration platform. With multiple companies such as Amazon, VMware, and Huawei actively backing and contributing to it, Kubernetes is becoming the de facto foundation of the future, and it offers several capabilities that a hyperplexed platform must possess. For instance, Kubernetes provides both portability and robust DevOps through frameworks such as Helm, Spinnaker, and Tekton. Kubernetes also supports a wide mix of workloads, such as serverless and serverless-style functions through extensions like KNative and Kubeless, and traditional virtual machines through Kubevirt. The system offers infrastructure, networking, and communications abstraction, as well as self-healing AI operations and distributed monitoring. Kubernetes-based platforms are also being used to construct applications that span multiple clouds, and specialized distributions are emerging for embedded use cases, IoT devices, and appliances in unattended locations.

Kubernetes is increasingly being used in innovative ways, but it solves only a part of the

problem; software developers still have to worry about operational details such as data-driven operations, multicluster management, and security. And Kubernetes isn't optimized for power-constrained hardware or other specialized use cases such as real-time, low-latency communications. Although the use of Kubernetes may be popular, it's clear that its ecosystem needs to evolve and be surrounded by next-generation run times, frameworks, and tools in order to become the hyperplexed platform of the future.

What Hyperplexed Architectures Portend

When it comes to software development today, it's important to remember that entry barriers are lower and innovation speeds are faster than ever. That is why customer-centric innovation and speed-to-market have become critical for success. To be competitive, keep pace with technological change, and capitalize on business opportunities, technology-driven organizations—hardware and software companies as well as enterprises building multicloud, hybrid, and connected applications—must reexamine their software architectures. (See the sidebar “BMW Group's New Architecture for Developing Autonomous Driving Systems.”) Three steps can help companies get started:

- **Reboot strategies.** Companies must revisit their approaches to technology and assess the role that hyperplexed architectures will play in their digital transformations and product roadmaps. CTOs and CIOs should continue to move quickly and make thoughtful investments in hybrid cloud platforms today to move up the experience curve and build the right technology and operating model for the future. They should also anticipate the shifts in the economics of development as well as the extent of competitive pressure—just as they did when exploring mobile applications and AI. They should keep on the lookout for emerging new consortiums and standards; budding open source and commercial innovations; and potential mergers and acquisitions targets. Smart companies will move early,

experiment, and learn quickly to gain first-mover advantage.

- **Seize new opportunities in infrastructure software.** Hyperplexed architectures will create new markets for platforms, cybersecurity, and development tools, which will be new avenues of growth for infrastructure software vendors as well as IT service firms. Companies must place their bets right away and start developing hyperplexed software portfolios and value propositions.
- **Revisit operating models and manage change.** Every organization should figure out the impact that new applications, architectures, and ways of working will have on its operating model and plan for the changes that will be necessary. Moving to new software architectures isn't just about technology; companies have to up-skill their engineering, design, and product development, and their program-management talent. In addition, they must update their product-planning, development, operations, and governance processes, building new capabilities such as systems engineering, AI operations, and Agile 2.0. Finally, companies must learn to manage organizational change, especially the ways of working triggered by the COVID-19 era, as they switch to new software architectures, designs, and operating models.

S SOFTWARE DESIGNS AND architectures are, clearly, on the cusp of change. Developing a new generation of software architectures will allow companies to innovate faster, create new customer experiences, and shift the economics of software development. Companies that rely on legacy technologies and outdated approaches, however, will find themselves at a disadvantage.

Using hyperplexed enterprise software architectures is essential to reap the benefits of the technological changes that are under way. Being reactive isn't an option because of the speed at which technology is changing, the intensity of competitive pressure, and the flood of private capital into soft-

BMW GROUP'S NEW ARCHITECTURE FOR DEVELOPING AUTONOMOUS DRIVING SYSTEMS

As the BMW Group stepped up its efforts over the last decade to develop advanced driver-assistance and autonomous driving systems, it realized that its traditional approach to software development wouldn't work. It was building a fleet of vehicles with prototype self-driving systems, each of which would generate several terabytes of LIDAR, radar, and camera data every hour. The data would have to be processed in real time, but inexpensively.

In addition to managing a data lake, BMW Group's new platform would have to run analytics, develop simulations, generate driving algorithms, and constantly update applications. Moreover, it needed to create an ecosystem of key business partners, which would see around 2,300 active users on the platform at any given time. BMW Group had no choice but to revisit both its hardware foundation and software architecture.

Initially, the BMW Group evaluated the possibility of using a standard private cloud platform but soon learned that scalability and cost-effectiveness would prove to be severe limitations. It decided to shift to a platform as a service, based on Red Hat OpenShift, and developed a Kubernetes-based platform.

BMW Group had to overcome many challenges to make that happen. It needed to move up the learning curve quickly, so it invested heavily in training. It had to build several capabilities, such

as infrastructure services in cloud-native storage, to scale effectively. The new platform provides an optimal mix of scalability and agility, and has turbocharged the group's innovation efforts.

BMW Group has been able to design, prototype, and deploy several applications based on its new capabilities in just three months. The platform has been designed to scale to 230 petabytes of storage, and has the computing power to absorb 240 million kilometers of test data. Because the Kubernetes-based platform provides an environment in which the group can partner with key vendors as well as other partners, it has helped slash development times. The platform has also helped improve productivity, generating more accurate simulations and faster analytics.

BMW Group's experience teaches three lessons. One, developing new technological capabilities demands new ways of working, so change management is critical and must be executed in parallel with the adoption of new technologies. Two, cloud-native software development still has several gaps such as the lack of infrastructure-management and operations capabilities, which need to be tackled internally. Three, moving quickly and making early investments is critical. Doing so has helped the BMW Group attract talent, build an effective operating-model, and shift to a stronger software foundation—quickly.

ware development. Hyperplexed architectures will become mainstream in the next three to five years, and unless companies stay ahead of that megatrend, they'll never

be able to catch up with rivals. As Marc Andreessen famously said a decade ago, "software is eating the world," so you can only ignore it at your own peril.

About the Authors

Pranay Ahlawat is an associate director in the San Francisco office of Boston Consulting Group. You may contact him by email at ahlawat.pranay@bcg.com.

Vikas Taneja is a managing director and senior partner in the Boston office of BCG. You may contact him by email at taneja.vikas@bcg.com.

Bernd Schlotter is a managing director and senior partner in the Silicon Valley-Bay Area office of BCG. You may contact him by email at schlotter.bernd@bcg.com.

Acknowledgments

The authors would like to thank Roger Premo, general manager, corporate strategy at IBM; Ashesh Badani, senior vice president, cloud platforms at Red Hat; and Brian Gracely, senior director of product strategy at Red Hat for their contributions to this article.

Boston Consulting Group partners with leaders in business and society to tackle their most important challenges and capture their greatest opportunities. BCG was the pioneer in business strategy when it was founded in 1963. Today, we help clients with total transformation—inspiring complex change, enabling organizations to grow, building competitive advantage, and driving bottom-line impact.

To succeed, organizations must blend digital and human capabilities. Our diverse, global teams bring deep industry and functional expertise and a range of perspectives to spark change. BCG delivers solutions through leading-edge management consulting along with technology and design, corporate and digital ventures—and business purpose. We work in a uniquely collaborative model across the firm and throughout all levels of the client organization, generating results that allow our clients to thrive..

© Boston Consulting Group 2021. All rights reserved. 2/21

For information or permission to reprint, please contact BCG at permissions@bcg.com. To find the latest BCG content and register to receive e-alerts on this topic or others, please visit bcg.com. Follow Boston Consulting Group on Facebook and Twitter.