White Paper

Extending Services Edge to Cloud – Requirements and Recommendations

Authors

Anna Scott
Chief Edge Architect, US Public Sector

Leland Brown
Technical Director
Advanced Communications-Military-Aerospace-Government, Intel Federal

Stan Mo
Edge Systems Software Architect, Federal Industrial Solutions, Network and Edge Solutions Group

Robert Watts
AI Architect, Federal Industrial Solutions, Network and Edge Solutions Group

Contributors

Milan Djukic
Senior Director, Intel Smart Edge Division

Matt LaTurner
Technical Director, Network and Edge Solutions

Craig Owen
Principal Engineer, Network and Edge Solutions Sales

Bala Parthas
Principal Engineer, Federal Industrial Solutions, Network and Edge Solutions Group

Karen Perry
Chief Solution Architect, Health and Life Sciences

Darren Pulsipher
Chief Solution Architect, US Public Sector

Dave Richard
Technical Director, US Public Sector

Gretchen Stewart
Chief Data Scientist, US Public Sector

Introduction: Realizing the vision of unified, end-to-end architecture

Edge, network, data center, and cloud are no longer distinct. The boundaries have blurred, enabling innovative applications and new workloads with optimized functionality.

Industry has had years to optimize architectures for the data center, cloud, and edge. While each of these environments has its own development and deployment parameters, they must be integrated seamlessly in today's edge-to-cloud workflows. The world of edge to cloud is increasingly complex, but open standards and versatile, flexible hardware and software technologies continue to support comprehensive solutions for difficult real-world problems. For example, recent releases of the 3GPP.org standards for 5G, advancements in artificial intelligence (AI) and machine learning (ML), and the growth of software-defined infrastructure and open platforms have redefined what is possible.

As these disparate environments and technologies converge, the concept of edge-to-cloud architecture has grown to encompass an increasing number of disciplines. This white paper addresses a broad set of areas—including cloud, 5G, edge, AI, and cybersecurity—that make up edge-to-cloud architectures.

Taxonomy

Currently, cloud and edge architectures can be defined by multiple terms. The following table defines the terms that are used in this paper.

Term	Location	Connectivity	Physical Security	Function	Real-World Example
Cloud	Stable, controlled environment	Reliable, high-bandwidth, low-latency connectivity	Controlled environment	"Unlimited" compute; able to support any application	Cloud service provider (CSP), data center
Data center	Stable, controlled environment	Reliable, high-bandwidth, low-latency connectivity	Controlled environment	"Unlimited" compute; able to support almost any application	Company/ organization data center
Converged edge	Limited environmental control; may need extended temperature and ruggedized compute	Unreliable or periodically unavailable connectivity; limited bandwidth; variable latency	Limited physical security (can be tampered with/ stolen); need for asynchronous operation disconnected from the cloud	Limited compute defined by application requirements; size, weight, power, and cost (SWaP-C) limited	Half rack on the factory floor; field office; tactical operations center
Dynamic edge	Field/tactical forward operating base (e.g., mobile or transitory operation); need extended temperature and ruggedized compute	Unreliable or periodically unavailable connectivity; limited bandwidth; low latency critical	No physical security (can be tampered with/stolen/ destroyed); need for asynchronous operation disconnected from the converged edge and cloud	SWaP-C limited; austere; compute limited; need for resilient, autonomous, clustered compute that can adapt to various applications and workloads	Drones surveying power lines; mobile workers; sensor devices for defense applications; any edge device

Table 1: Edge-to-cloud terminology.

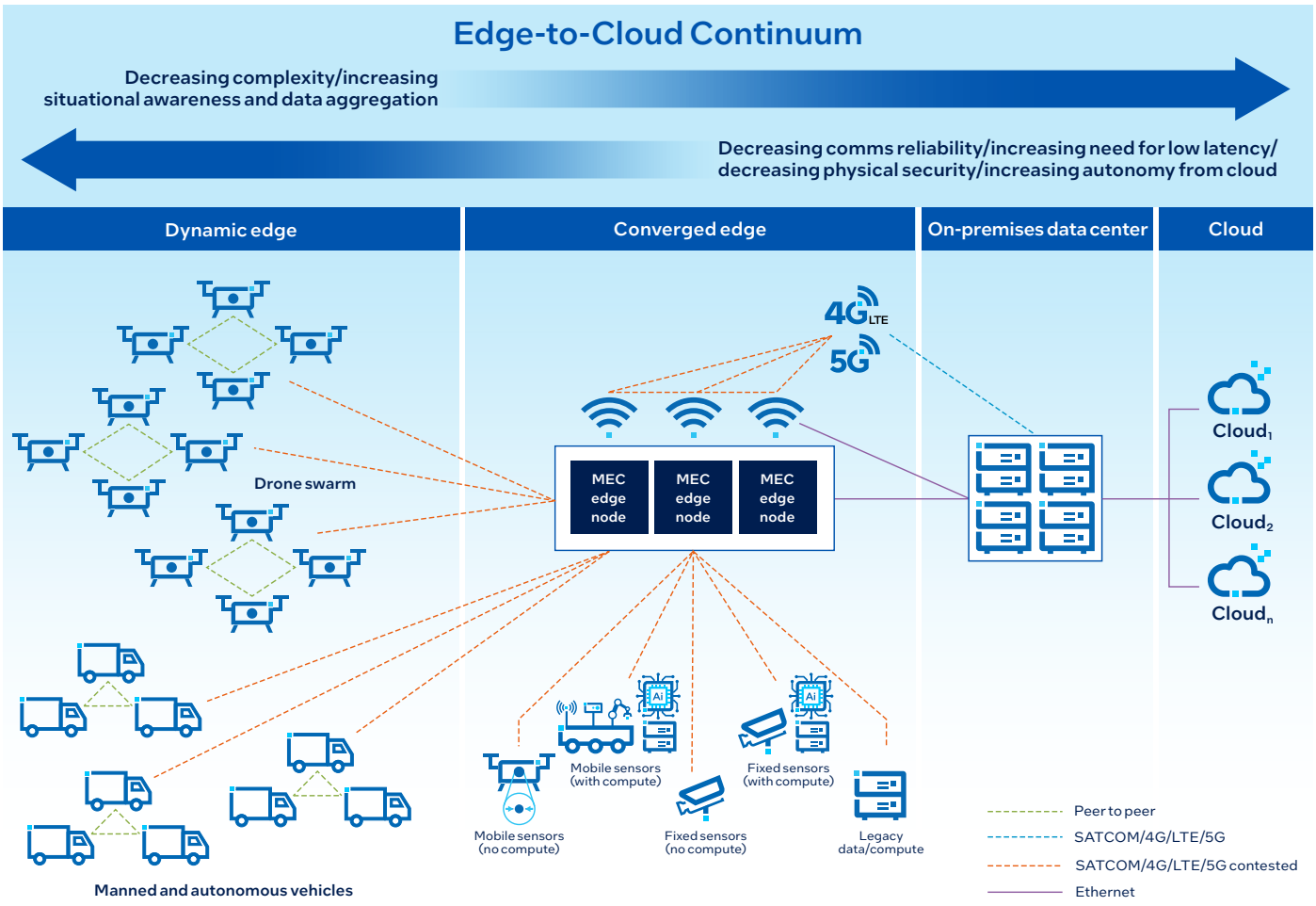

Figure 1: Edge-to-cloud architectures span multiple protocols, carriers, and devices.

Key requirements for edge-to-cloud architectures

This paper defines a set of requirements for edge-to-cloud architectures that apply broadly across multiple industries. These potential uses span from factory floors and healthcare to remote oil and gas installations and expeditionary, forward operating defense bases. This architecture requirement list is not comprehensive, and some of the capabilities are not technically achievable yet. However, they can help define areas for development and can guide the synergy of existing capabilities.

Edge-to-cloud architectures come in numerous forms, each designed for a specific problem in a unique operating environment. Despite this diversity, developing a core architecture that can be reused across applications is a valuable goal. Early evidence shows this is possible, especially in areas like computer vision.

At a systems level, edge-to-cloud architectures must be informed by and include the following:

- **Security:** Designed in for data, applications, networks, sensors, and hardware
- **Applications:** Must adapt to evolving needs and function at all locations across the edge-to-cloud environment; must be informed by latency requirements and network reliability
- **Data management, orchestration, and aggregation:** Designed to optimize data management and aggregation from the dynamic edge to the cloud
- **Communications:** Built for resilience, software defined, and, ideally, self-healing
- **Manageability:** Deploy at scale easily, without enlisting an army of support technicians to maintain hundreds to thousands of edge devices and nodes
- **Availability:** Support high availability of applications and services with autonomous edge functionality under intermittent connectivity conditions; support live failover capability and live migration of services
- **Cloud:** Integrate with multiple clouds and support hybrid cloud architecture
- **Cost:** Must be affordable to ensure development of deployable edge-to-cloud architectures

Securing edge-to-cloud architectures

Core edge-to-cloud architectures require security if they are going to be deployed and used at scale. The very nature of an edge-to-cloud architecture means that the attack surface will be expansive. Security must be designed in and extended to the hardware level and to application development. Consideration should also be given to the trustworthiness of the hardware components and how they are designed, manufactured, and packaged.[1]

Given the real-time nature of many edge applications, security must function in a way that does not increase latency and adversely impact time-critical functionality. Security measures must also be able to survive the physical loss of edge hardware without compromising the network/compute infrastructure.[2,3]

Current security trends focus on the development of zero-trust architectures per National Institute of Standards and Technology (NIST)[4] and Department of Defense (DOD) guidelines. Zero-trust architectures target exactly these types of edge-to-cloud architectures. Security solutions, both hardware and software, must secure edge devices and networks while nesting with security architectures that extend to the cloud. Finally, it is important to create security features that are easy to use since complex, time-consuming security features are often disabled or rarely used.

Application guidelines

Applications are defined by the end user needs, which can range from simple database queries to complex AI/ML algorithms. In practice, the application aspect of the edge-to-cloud architecture includes the applications/functions themselves—generally containerized but virtual machines are also used—and a system for orchestrating and managing the containers or native applications on the various networked nodes. Applications and how they are developed and managed is one of the primary areas for innovation in the definition of edge-to-cloud architecture. Below we cover five areas that we think are critical for applications that need to span edge to cloud.

Build on open, interoperable architectures

Intel's design philosophy combines the compatibility and scalability of Intel® hardware with industry-standard approaches to connectivity, memory, and storage. This type of open architecture is essential for edge to cloud. With an open architecture, it is possible to leverage the entire software ecosystem (both open source and proprietary software) to deliver applications. This flexibility allows systems to interact with other systems, evolve with time and technological advances, and adapt to changing conditions.It also prevents vendor lock-in.

Code once

Applications should be designed so they can work from edge to cloud without the need to recode. This means using standard container APIs so that applications can run on any available hardware,[6,7,8] plus designing output to work with low- and high-bandwidth connectivity.

Run anywhere

Applications should be designed for distribution across multiple nodes from edge to cloud and to optimize execution in the absence (or presence) of cloud connectivity. Applications (and their container orchestrators) also need to be lightweight enough to run on dynamic and converged edge nodes that are size, weight, and power (SWaP) limited.

Leverage AI/ML

AI/ML can dramatically impact edge-to-cloud architectures since the data needed for algorithm training is primarily at the edge, while the most-time-efficient training compute is in the cloud/data center. As capabilities evolve, we anticipate continuous model improvement will become possible, without the need to move the data to the cloud or data center for training and back to the edge for inference.[9,10]

Design for dynamic edge environments

Applications should be designed to take input on network quality of service and bandwidth to allow for optimization of data sharing based on dynamic connectivity. Security features should be designed into the applications so that additional checks can be created to compare actual activity against design. Finally, applications should allow users to assign prioritization of use and output so that the most-critical information can be shared first in network-constrained environments.

Data management and orchestration

Data centers and the cloud have well-developed systems for data management and orchestration that use containers and virtual machines (VMs) in an unchanging environment with high-speed interconnects. The edge needs similar management and orchestration, but the criteria for optimization are very different. Instead of always-on, homogenous compute, the edge has a large variety of compute types. Edge networks run on wired and wireless communications in environments that can change rapidly. Battery power runs out. Weather disrupts wireless communications. Compute can be damaged or stolen from the site. Design should follow an "edge-first" philosophy, which means the system can maintain continuous, optimal edge operation even when cloud connections and edge elements fail.

Managing the dynamic edge

The dynamic edge is the worst-case scenario in many ways. Think of a drone swarm operating in a remote location whose mission is to map downed power lines after a hurricane. Connectivity may not exist. The drones will need to be battery powered. The data collected can only be shared when the drones move back into a region that has both power and communications. Orchestrating the drones requires a new set of capabilities that are lightweight and highly coordinated.

Dynamic edge orchestration must match data, compute, storage, and networking to application latency requirements—including data-heavy workloads like ML and AI. For some dynamic edge applications, the situation is even more difficult because battery limitations and detectable spectrum can impact performance. The dynamic edge needs a lightweight, high-speed message bus[11] and a containerized application infrastructure capable of distributing functionality across multiple edge devices. The idea is to simplify integration of new and legacy applications with an efficient data pipeline between the two using a framework that makes it easy to upgrade and replace software.

Managing the converged edge

The converged edge is less complex. Converged edge networks typically have higher levels of compute in fixed locations that don't depend on battery power such as factory floors or tactical operations centers. This stability can support multi-access edge computing (MEC) nodes. MEC nodes bring much of the management and orchestration needed. MEC nodes can deliver applications at the edge while helping to ensure security, manageability, and ease of deployment. They can also provide failover capabilities and support operation while disconnected from the cloud and data centers.[12]

For both dynamic and converged edge environments, a control plane is a critical element for scalable, more secure, and manageable edge-to-cloud deployments. Control planes can be responsible for configuration and initialization, including reconfiguring, adding, or deprecating sensors and edge devices. Control planes provide telemetry- and policy-based tools for managing workloads, load balancing, and optimizing performance. MEC control planes also centralize security management, including permissions, patches, and device quarantines.

Communications, latency, and asynchronous operation

Wireless communications are central to enabling edge-to-cloud architectures. In theory, there could be a single edge-to-cloud architecture with only sensors and user devices at the edge. All other functionalities would live in the cloud, but that would require wireless comms with massive bandwidth over vast areas. Applications would need to withstand lost connections, and nothing could run in near-real time. It might not be the most economical, but it is technically feasible.

However, in many use cases, communications are not reliable and high bandwidth is not available. In addition, many applications must run at the edge because the physics of the data paths will always limit how quickly data can travel from the source to the cloud and back again. As a result, independent edge functionality and interplay with the cloud are central to a robust edge-to-cloud architecture. Maintaining edge-to-cloud architectures within these practical limitations requires:

- Multiple networks that can fail over seamlessly and are intelligently managed based on local conditions and situational awareness. Elements to manage may include varying radio gain to reduce visibility, selecting spectrum to match environment, and orchestrating connection topology to optimize power consumption and maximize critical communications paths.[13]
- Distributed, mesh compute architectures at the edge with failover capabilities so compute can continue functioning even if some network or compute elements are lost.
- Converged edge and dynamic edge compute that can operate asynchronously with no cloud connection for extended periods of time.
- Converged and dynamic edge compute tightly coupled to sensor data so that critical intelligence is developed and accessed locally then shared globally when connectivity is available.
- Mesh environments that use connectors, bridges, sensors, and relays to accommodate multiuse edge devices that may not conform to common protocols.

Data aggregation

One of the major advantages of wireless communications is that it makes aggregating mobile and fixed data sources technically and financially feasible—even when they are dispersed over large geographical areas. A successful architecture needs new ways to organize and aggregate data on a grand scale—ideally in real time. These methods must target the fidelity and timeliness of data at the dynamic edge so that it can support rapid assimilation and decision-making capabilities locally and in the cloud. In short, the system needs to maintain one version of the truth regardless of user location.

Intel advocates processing data at the edge, as close to the data source as possible. Processing on an edge device or node makes intelligence available locally for immediate use, and it creates a smaller postprocessing form that is easy to share with other edge nodes and the cloud.

Another critical concept for data aggregation is the spatial-temporal "scene graph" based on open standards such as glTF,[14] X3D,[15] and Open Geospatial Consortium (OGC) 3D Tiles.[16] This concept and technology are borrowed from the gaming world. A scene graph is a 2D or 3D digital twin of a physical environment mapped with photos, CAD drawings, lidar scan images, or 2D scale maps. A sensor overlay maps physical sensor locations and data onto the scene graph environment. This construct allows people, equipment, and objects to be mapped in real time—using radio frequency (RF), video inferencing, or audio data—alongside time series data like ambient temperature, barometric pressure, and air quality.

Scene graph data lives in the data management system and is easily accessible to multiple, containerized analytics applications—for example, operational analytics, heat maps, wayfinding, or logistics. These scene graphs are arranged as a hierarchy of scenes that are aggregated based on parent/child relationships. This enables accumulation of data and intelligence across the entire architecture so higher-level analytics can take advantage of all relevant data sources for intelligence and situational awareness. When combined with networked edge compute, dynamic scene graphs can power a fabric of interconnected nodes with full 4D (3D + time) situational awareness that can scale from small local scenes up to a global operating picture.

Manageability

There can be hundreds if not thousands of transient edge devices within an edge-to-cloud architecture connecting to multiple clouds and pulling data from a multitude of sensor types. These complex systems make manageability a major concern. If they can't be installed and maintained by a reasonable number of technicians, they won't be economically feasible. Table 2 shows a list of management tools and capabilities, many related to security and network, that can automate processes and help reduce IT overhead.[17]

Security	Edge	Network	Software	Cloud
Demarcated trust management	Edge node resiliency and failover management	Network control versatility	Alarming, telemetry, and health tools	Cloud application interoperability
Trusted and automated onboarding	Edge node hardware interoperability	Client network control (IP address range, DNS, DHCP, etc.)	Software distribution	Idempotent and asynchronous control
Zero-trust framework with role-based access control (RBAC) and "just-in-time" access (guest and native to all elements)			Consolidated application environment	Cloud-native platform
Transparent IPsec interception				
Embedded public key infrastructure (PKI) microservices				

Table 2: Management capabilities.

This is just a starting list, but hopefully it establishes the importance of designing systems so that they can be installed, operated, updated, and maintained by relatively few technicians over the system's lifetime.[18]

Availability

High availability of data and services is needed in edge-to-cloud architectures. Many edge applications require failover capabilities and redundancy so that critical services are not lost. Failover strategies must be designed in. The current norm is for critical systems to be hardwired. As we move critical functionality into the world of wireless communications, we need to architect these systems to support continuous functionality even when components fail.

Cloud

Cloud is obviously a critical element in edge to cloud.[19] The ideal instantiation within the edge-to-cloud architecture would allow use of multiple clouds, with data or application sharing across the clouds. This would enable optimization of cloud capabilities as defined by the problem set and location of the converged edge/dynamic edge data. The cloud will also be critical for providing the capacity needed for AI/ML model development and refresh.

Cost

Affordability is central to developing deployable edge-to-cloud architectures. Since the scale of end devices and applications can be significant, architectures must be designed so that they can be installed and maintained without a vast number of field technicians. Ideally, installations and updates will be remote and zero touch.

Conclusion: Explore edge-to-cloud possibilities

Today, society is in the initial stages of a major transformation in how compute enables functionality. Compute devices are no longer tethered with wired sensors, fixed data centers, and offices. Compute can live at the edge where the data is gathered, and the intelligence generated can be efficiently and more securely shared broadly via cloud architectures.

As these applications and deployments become more complex, developers are faced with nearly infinite choices when designing and building edge-to-cloud solutions. Few limitations exist beyond the self-imposed boundaries of cost, optimization, and mission.

However, emerging standards and cross-platform tools can help to simplify solution development, deployment, and management. With these supporting technologies and continued collaboration across a wide spectrum of technical capabilities, we can expect to see great advancements in edge-to-cloud deployments in the months and years to come.

Recommendations

Security should adopt zero-trust architectures and be designed to survive the loss of connectivity and the physical loss of devices.

Innovation should continue in application design to improve security, leverage network awareness, and achieve portability from the cloud to the dynamic edge without recoding.

New orchestrators that can manage data, optimize workloads, and overcome the specific challenges of both the dynamic edge and the converged edge should be incorporated into these architectures.

Interfaces of technology (AI, 5G, edge, and cloud) should be investigated to determine how to improve capabilities by treating different communication protocols as a continuum instead of as silos.

Data processing and aggregation should begin at the edge as close to the data source as possible. New techniques such as dynamic scene graphs should be explored to provide full 4D situational awareness.

Software-defined data, device, and security management platforms should be deployed within edge-to-cloud architectures to reduce complexity, automate key functions, and make system management financially feasible.

Edge-to-cloud systems should be evaluated holistically—network functions, compute capacity, application orchestration and performance, data management and aggregation—to maximize system availability and performance while minimizing ongoing support, maintenance, and operational costs.

Learn more about your emerging innovation opportunity

For more information on the technologies discussed in this white paper, please refer to the following resources, or contact IOTG-PublicSector@intel.com.

Related topics

Powering 5G Networks from Cloud to Edge

AI and Data Science

Cloud Performance, Flexibility, and Security

Intel® edge-to-cloud technologies

Hardware-Enabled Security Capabilities

Intel® Smart Edge

Intel® oneAPI Toolkits

Intel® Distribution of OpenVINO™ Toolkit

Transparent Supply Chain

References

1. "Transparent Supply Chain," Intel, accessed May 18, 2022, intel.com/content/www/us/en/products/docs/servers/transparent-supply-chain.html.
2. "Hardware-Enabled Security Technology," Intel, accessed May 18, 2022, intel.com/content/www/us/en/security/hardware/hardware-security-overview.html.
3. "Intel Smart Edge," Intel, accessed May 18, 2022, intel.com/content/www/us/en/design/technologies-and-topics/edge-cloud-computing/smart-edge-software.html.
4. "SP 800-207 Zero Trust Architecture," National Institute of Standards and Technology (NIST), August 2020, csrc.nist.gov/publications/detail/sp/800-207/final.
5. "Department of Defense Zero Trust Reference Architecture, Version 1.0," United States Joint Defense Information Systems Agency (DISA) and National Security Agency (NSA) Zero Trust Engineering Team, February 2021, dodcio.defense.gov/Portals/0/Documents/Library/(U)ZT_RA_v1.1(U)_Mar21.pdf.
6. "oneAPI: A New Era of Heterogeneous Computing," Intel, accessed May 18, 2022, software.intel.com/oneapi.
7. "Intel® DPC++ Compatibility Tool Transform Your CUDA Applications into Data Parallel C++ (DPC++) Code That's Based on SYCL," Intel, accessed May 18, 2022, intel.com/content/www/us/en/developer/tools/oneapi/dpc-compatibility-tool.html#gs.1dwx40.
8. "Intel® oneAPI Toolkits," Intel, accessed May 18, 2022, intel.com/content/www/us/en/developer/tools/oneapi/toolkits.html#gs.1dwzt8.
9. "AI and Data Science," Intel, accessed May 18, 2022, intel.com/content/www/us/en/artificial-intelligence/overview.html.
10. "Intel® Distribution of OpenVINO™ Toolkit," Intel, accessed May 18, 2022, software.intel.com/content/www/us/en/develop/tools/openvino-toolkit.html.
11. "ZeroMQ: An open-source universal messaging library," ZeroMQ, accessed May 18, 2022, zeromq.org/.
12. "Intel® Smart Edge," Intel, accessed May 18, 2022, intel.com/content/www/us/en/design/technologies-and-topics/edge-cloud-computing/smart-edge-software.html.
13. "Powering 5G Networks from Cloud to Edge," Intel, accessed May 18, 2022, intel.com/content/www/us/en/wireless-network/5g-network/overview.html.
14. "glTF Runtime 3D Asset Delivery," Khronos Group, accessed May 18, 2022, khronos.org/gltf.
15. "X3D," Wikipedia, accessed May 18, 2022, en.wikipedia.org/wiki/X3D.
16. "3D Tiles," Open Geospatial Consortium, accessed May 18, 2022, ogc.org/standards/3DTiles.
17. "Intel® Smart Edge Open: Streamline networking and application deployment at the edge," Intel, accessed May, 18, 2022, intel.com/content/www/us/en/developer/tools/smart-edge-open/overview.html.
18. "Intel® Smart Edge," Intel, accessed May 18, 2022, intel.com/content/www/us/en/design/technologies-and-topics/edge-cloud-computing/smart-edge-software.html.
19. "Cloud Performance, Flexibility, and Security," Intel, accessed May 18, 2022, intel.com/content/www/us/en/now/edge-to-cloud/cloud.html.

0722/DP/CMD/PDF