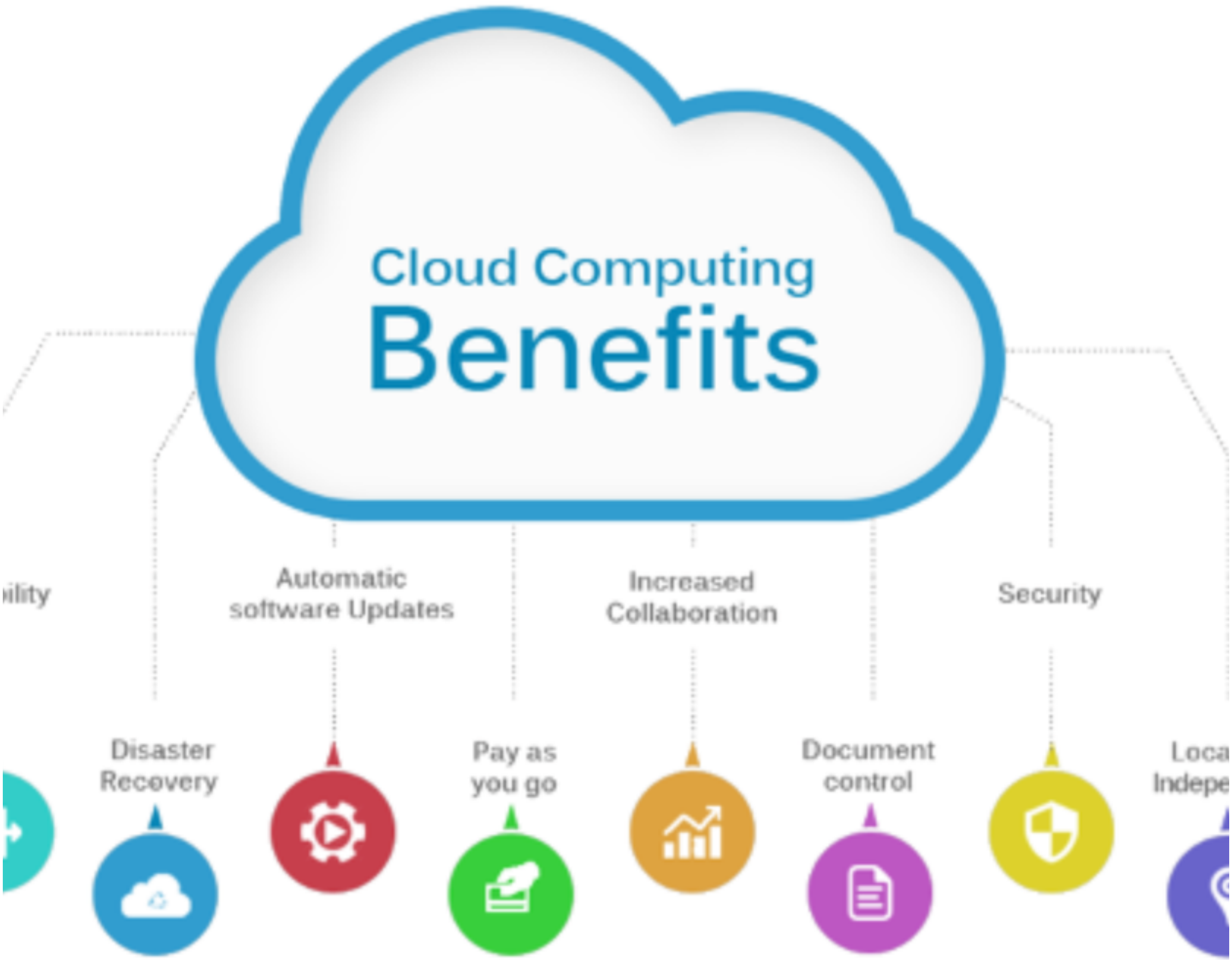
Cloud Computing Additional Information

Jennifer Morales (C)
Contingent Worker



What is Cloud Computing

Bad News: The Classic [X]-As-A-Service (-aaS) Model Is Dead

In the early days of cloud, the -aaS model served us well. However, a decade later, it provides a disservice to the market with its oversimplification. The reality is far more complex.

Good News: The New Model Isn't Overwhelming
Opportunity has driven innovation. We will share what is replacing the classic three-tiered -aaS model that maps to the same technology tiers familiar to I&O and the business.

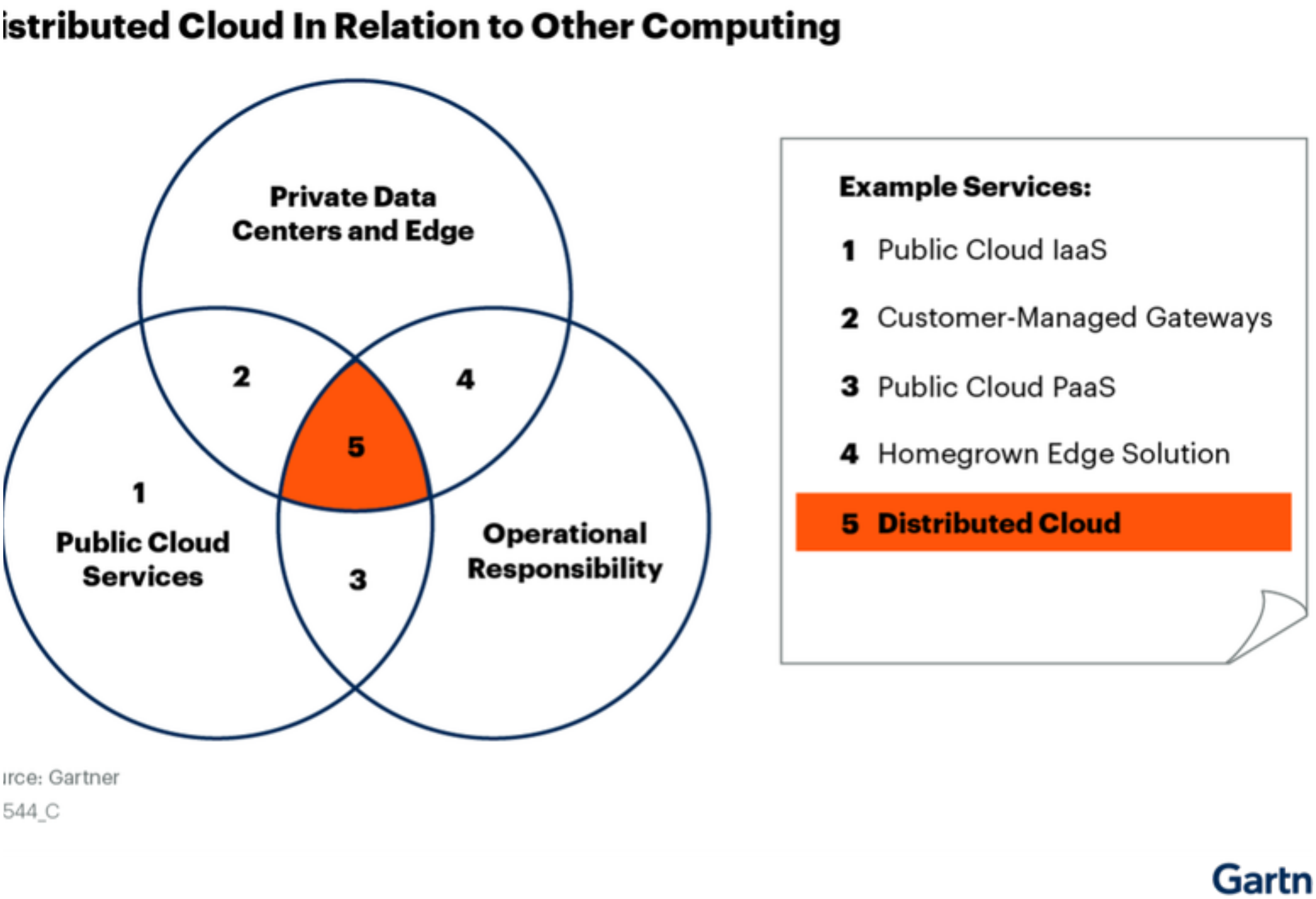
What is next in Cloud Computing?

The evolution of cloud computing is the distributed cloud. This article will define all the buzz terms and explain what a distributed cloud is and why we are interested in it.

What Is Distributed Cloud?

To understand what distributed cloud is, we must go back to the definition for both cloud computing and edge computing. Cloud computing is a style of computing in which scalable and elastic IT-enabled capabilities are delivered as a service (<A listing of AWS/Azure Services>) including using internet technologies. Edge computing is part of a distributed computing topology where information processing is located close to the edge, where things and people produce, store or consume that information.

Despite our colloquial usage of "public cloud," cloud computing has never included the location of the service as part of the definition. While we associate these services with a particular provider and/or location, it has always been a *style* of computing. Location matters, as many government entities can attest, not only for geopolitical reasons, but also for the availability of specific services in the right region.

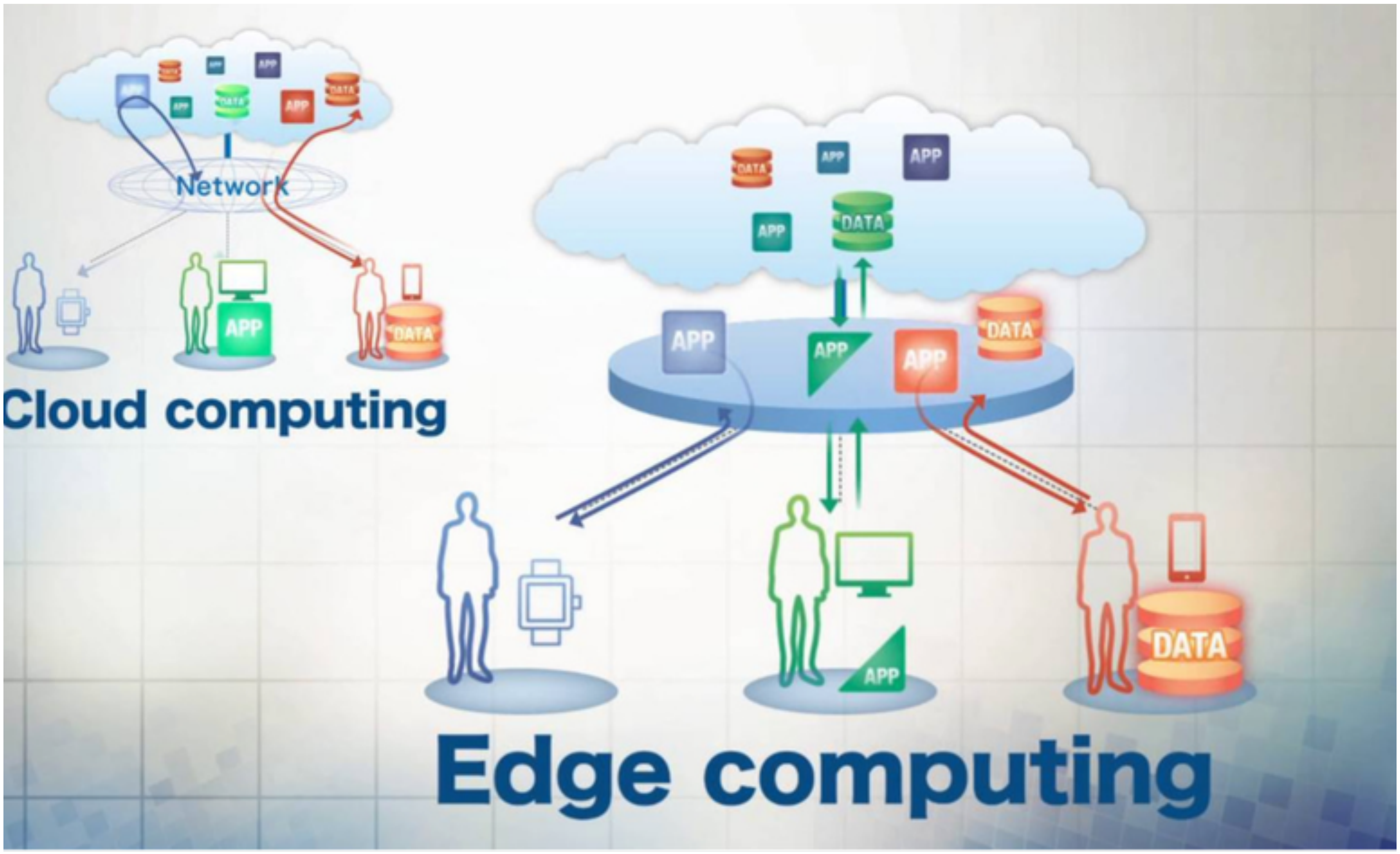


Distributed cloud is the distribution of public cloud services to different physical locations, including the edge, while ownership, operation, governance, updates, and evolution of the services remain the responsibility of the originating public cloud provider. A few examples of this are: Office365, Adobe, Netflix or YouTube offline.

Distributed cloud is at the intersection between edge computing and public cloud services. Organizations will have difficulty proceeding toward distributed cloud models without a well-defined strategy for both areas.

New technologies, programming languages, and tools are created regularly. Polyglot programming is not sticking to one programming language but using the best one for the particular problem at hand. The Distributed cloud allows you to take advantage of Polyglot programming. This is an evolution of micro-service-oriented architecture, which also allows for this concept.

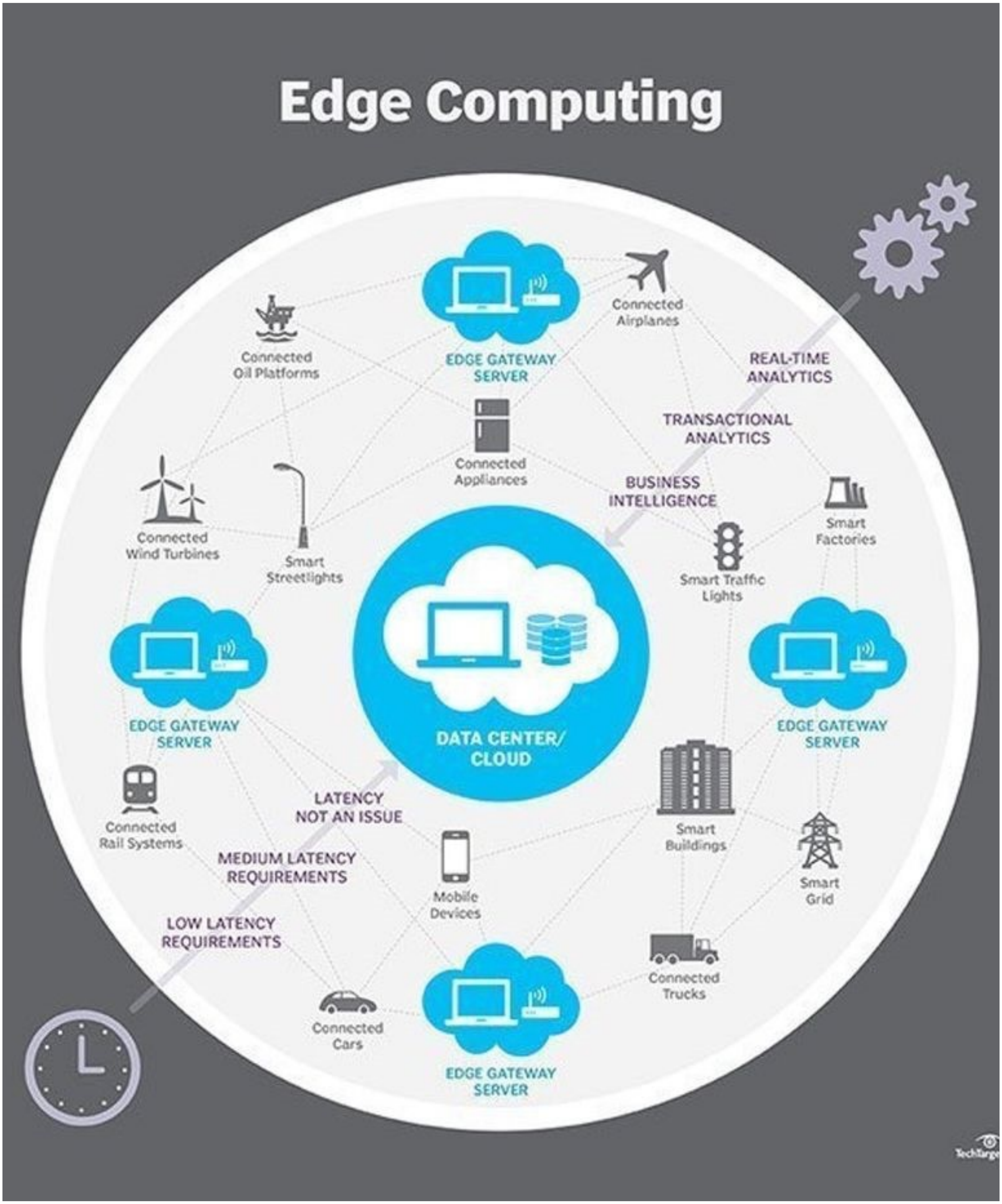
What is Edge Computing?



Edge computing is a distributed computing paradigm that brings computation and data storage closer to the user. This is expected to improve response times and save bandwidth. "A common misconception is that edge and IoT are synonymous. Simply stated, edge computing is a topology and location sensitive form of distributed computing, while IoT is a use case instantiation of edge computing." The term refers to an architecture rather than a specific technology. Even though most diagrams only show three levels for edge computing, in reality there may be many levels and the edge can be anywhere between the request and the origin.

The traditional computing paradigm built on a centralized data center and everyday internet isn't well suited to moving endlessly growing rivers of real-world data. Bandwidth limitations, latency issues and unpredictable network disruptions can all conspire to impair such efforts. Businesses are responding to these data challenges with distributed architectures, of which edge computing is a major component.

In simplest terms, edge computing moves some portion of storage and compute resources out of the central data center and closer to the source of the data itself. Rather than transmitting raw data to a central data center for processing and analysis, that work is instead performed where the data is actually generated - whether that's a retail store, a factory floor, a sprawling utility or across a smart city. Another way of saying that is the aim of edge computing is to move the computation away from data centers towards the edge of the network, exploiting smart objects, mobile phones, or network gateways to perform tasks and provide services on behalf of the cloud. Only the result of that computing work at the edge, such as real-time business insights, equipment maintenance predictions or other actionable answers, is sent back to the main data center for review and other human interactions.



By moving [services](#) to the edge, it is possible to provide content [caching](#), service delivery, persistent data storage resulting in better response times and transfer rates. At the same time, distributing the logic to different network nodes introduces new issues and challenges.

Privacy and security

The distributed nature of this paradigm introduces a shift in security schemes used in [cloud computing](). In edge computing, data may travel between different distributed nodes connected through the [Internet]() and thus requires special encryption mechanisms independent of the cloud. Edge nodes may also be resource-constrained devices, limiting the choice in terms of security methods. Moreover, a shift from centralized top-down infrastructure to a decentralized trust model is required. On the other hand, by keeping and processing data at the edge, it is possible to increase privacy by minimizing the transmission of sensitive information to the cloud.

Scalability

Scalability in a distributed network must face different issues. First, it must consider the heterogeneity of the devices, having different performance and energy constraints, the highly dynamic condition, and the reliability of the connections compared to more robust infrastructure of cloud data centers. Moreover, security requirements may introduce further latency in the communication between nodes, which may slow down the scaling process.

Reliability

Management of failovers is crucial to keep a service alive. If a single node goes down and is unreachable, users should still be able to access a service without interruptions. Moreover, edge computing systems must provide actions to recover from a failure and alerting the user about the incident. To this aim, each device must maintain the network topology of the entire distributed system, so that detection of errors and recovery become easily applicable. Other factors that may influence this aspect are the connection technologies in use, which may provide different levels of reliability, and the accuracy of the data produced at the edge that could be unreliable due to environment conditions. As an example, an edge computing device, such as a voice assistant may continue to provide service to local users even during cloud service or internet outages.

Speed

Edge computing brings analytical computational resources close to the end users and therefore can increase the responsiveness and throughput of applications. Some applications rely on short response times, making edge computing a significantly more feasible option than cloud computing.

Efficiency

Due to the nearness of the analytical resources to the end users, sophisticated analytical tools and Artificial Intelligence tools can run on the edge of the system. This placement at the edge helps to increase operational efficiency and is responsible for many advantages to the system.

Additionally, the usage of edge computing as an intermediate stage between client devices and the wider internet results in efficiency savings that can be demonstrated in the following example: In voice recognition, if the recognition is performed locally, it is possible to send the recognized text to the cloud rather than audio recordings, significantly reducing the amount of required bandwidth.

Applications

Edge application services reduce the volumes of data that must be moved, the consequent traffic, and the distance that data must travel. That provides lower latency and reduces transmission costs.

The evolving privacy and data security landscape

Virtually every country has enacted some sort of data privacy laws to regulate how information is collected, how data subjects are informed, and what control a data subject has over his information once it is transferred. Failure to follow applicable data privacy may lead to fines, lawsuits, and even prohibition of a site's use in certain jurisdictions. It is very important now to keep in mind what you must do, depending on where in the world you are doing business. We present some of the buzz terms and their definition here to help.

Digital sovereignty: the authority of an organization or ecosystem of organizations to govern its digital technology, services, and processes without interference of a body or state or underlying platform with more authority, power, and access.

Data sovereignty: laws that identify certain data as coming under the jurisdiction of a territory, generally where the data was collected (for example, the EU General Data Protection Regulation (GDPR) and India’s Personal Data Protection Bill).

Data residency: laws that place restrictions on where data can be physically located. This may be in absolute terms (for example, the International Traffic in Arms Regulations) or if some conditions are (not) met (such as Russia’s residency laws, China’s Cybersecurity Law and also the GDPR).

However, data sovereignty is often confused with data residency, where the latter refers to the physical location of the data within a country or legal jurisdiction. This is a large difference which is affected by legislation such as the GDPR and individual regulated industries. Where data residency ensures data is held in a legal jurisdiction, satisfying some legislation, data sovereignty ensures the legal responsibility to uphold data privacy and some forms of security classification. Merely hosting data in a particular geography, as has been seen by the multiple availability zone choices for European customers of the hyperscale cloud service providers, does not necessarily satisfy data sovereignty laws. This gap is especially relevant for government and highly regulated industries (such as financial services, defense, and healthcare), which bear a responsibility to ensure the strictest confidentiality of data.

Cloud and Development Terminology

Cloud computing is on-demand access, via the internet, to computing resources - applications, servers (physical servers and virtual servers), data storage, development tools, networking capabilities, and more - hosted at a remote data center managed by a cloud services provider (or CSP). The CSP makes these resources available for a monthly subscription fee or bills them according to usage.

Virtualization enables cloud providers to make maximum use of their data center resources. More and more corporations have adopted the cloud delivery model for their on-premises infrastructure so they can realize maximum utilization and cost savings vs. traditional IT infrastructure and offer the same self-service and agility to their end-users.

Types of Cloud Computing

As we stated at the begging of this article, the industry has started to evolve past the original ways of classifications of cloud computing. But the original terms and meanings are still being used today, so we have documented that information here.

Originally cloud computing would be deployed in different ways depending on what services a business needed. The first thing you would consider was the deployment model - public cloud, private cloud, hybrid cloud, and multi-cloud. Then you would pick the service category - Saas (Software as a Service), Paas (Platform as a Service), Iaas (Infrastructure as a service) and IPaaS (Integrated Platform as a Service). The technology and terminology are shifting (to Distributed Cloud and Edge Computing) but understanding the classic terminology is still important.

Deployment models

Public Cloud

Public cloud is a type of cloud computing in which cloud service providers makes computing resources - anything from SaaS applications, to individual virtual machines (VMs), to bare metal computing hardware, to complete enterprise-grade infrastructures and development platforms - available to users over the public internet. These resources might be accessible for free, or access might be sold according to subscription-based or pay-per-usage pricing models.

The public cloud provider owns, manages, and assumes all responsibility for the data centers, hardware, and infrastructure on which its customers' workloads run, and it typically provides high-bandwidth network connectivity to ensure high performance and rapid access to applications and data.

Public cloud is a multi-tenant environment - the cloud provider's data center infrastructure is shared by all public cloud customers. In the leading public clouds - Amazon Web Services (AWS), Google Cloud, IBM Cloud, Microsoft Azure, and Oracle Cloud - those customers can number in the millions.

Many enterprises are moving portions of their computing infrastructure to the public cloud because public cloud services are elastic and readily scalable, flexibly adjusting to meet changing workload demands. Others are attracted by the promise of greater efficiency and fewer wasted resources since customers pay only for what they use. Still others seek to reduce spending on hardware and on-premises infrastructures.

Private Cloud
This cloud model is great for organizations concerned about sharing resources on a public cloud. It is implemented on servers dedicated to the organization and accessed over the internet or through a private internal network.

A private cloud environment gives you complete control over data and security in order to meet specific regulatory and other compliance requirements (e.g., HIPAA for healthcare, GDPR, etc.).

Hybrid Cloud
Many organizations use a combination of several cloud environments. This is referred to as a hybrid cloud approach. Hybrid cloud often includes a combination of public cloud and private cloud, frequently in combination with some on-premise infrastructure. To create a true hybrid cloud architecture, you must set up communication or orchestration between the various deployments.

Hybrid cloud eliminates reliance on any single cloud provider and allows for additional levels of flexibility in terms of capabilities, security compliance, etc.

In the past, choosing a cloud provider meant picking public over private. This is no longer the case. To support regulatory, performance, and data gravity requirements, the cloud providers are now offering private cloud carveouts in public environments. VMware on AWS (VMC), Azure VMware Services (AVS), and Google's SAP, Oracle and Bare Metal solutions are good examples. Similarly, the cloud providers have been working on private cloud extensions. This blurring of public and private under a hybrid cloud umbrella is likely to accelerate in the future. Over time, we might no longer see a delineation between "public" and "private" but instead, between "dedicated" and "shared."

Multi-cloud
A multi-cloud approach is a particular case of hybrid cloud in which an organization uses services from multiple public cloud providers.

Service categories

Software as a Service (SaaS)

SaaS also known as cloud-based software or cloud applications - is application software that's hosted in the cloud and that you access and use via a web browser, a dedicated desktop client, or an API that integrates with your desktop or mobile operating system. In most cases, SaaS users pay a monthly or annual subscription fee; some offer 'pay-as-you-go' pricing based on your actual usage.

In addition to the cost savings, time-to-value, and scalability benefits of cloud, SaaS offers the following:

- **Automatic upgrades:** With SaaS, you take advantage of new features as soon as the provider adds them, without having to orchestrate an on-premises upgrade.
- **Protection from data loss:** Because your application data is in the cloud, with the application, you don't lose data if your device crashes or breaks.

SaaS is the primary delivery model for most commercial software today - there are hundreds of thousands of SaaS solutions available, from the most focused industry and departmental applications to powerful enterprise software database and AI (artificial intelligence) software. Popular SaaS products include Salesforce, Workday, or Microsoft Office 365.

It should be mentioned that Infrastructure cost savings with SaaS are often passed directly to the customer. This is one of the reasons SaaS competitors compete for cost effective infrastructure, because savings means more competitive pricing.

Platform as a Service (PaaS)

PaaS is considered the most complex of the three layers of cloud-based computing.
PaaS provides software developers with on-demand platform - hardware, complete software stack, infrastructure, and even development tools - for running, developing, and managing applications without the cost, complexity, and inflexibility of maintaining that platform on-premises.

With PaaS, the cloud provider hosts everything - servers, networks, storage, operating system software, middleware, databases - at their data center. Some PaaS solutions do offer on-premise or private cloud access to software, but his is often very expensive. Developers simply pick from a menu to 'spin up' servers and environments they need to run, build, test, deploy, maintain, update, and scale applications.

Today, PaaS is often built around *containers*, a virtualized compute model one step removed from virtual servers. Containers virtualize the operating system, enabling developers to package the application with only the operating system services it needs to run on any platform, without modification and without need for middleware.

Red Hat OpenShift is a popular PaaS built around Docker containers and Kubernetes, an open source container orchestration solution that automates deployment, scaling, load balancing, and more for container-based applications. Other examples of PaaS products include Google App Engine, web servers, and SQL servers.

Infrastructure as a Service (IaaS)
IaaS provides on-demand access to fundamental computing resources-physical and virtual servers, networking, and storage over the internet on a pay-as-you-go basis. IaaS enables end users to scale and shrink resources on an as-needed basis, reducing the need for high, up-front capital expenditures or unnecessary on-premises or 'owned' infrastructure and for overbuying resources to accommodate periodic spikes in usage. Popular examples of the IaaS system include Amazon Web Services (AWS) and Microsoft Azure.

In contrast to SaaS and PaaS (and even newer PaaS computing models such as containers and serverless), IaaS provides the users with the lowest-level control of computing resources in the cloud.

IaaS was the most popular cloud computing model when it emerged in the early 2010s. While it remains the cloud model for many types of workloads, use of SaaS and PaaS has grown at a much faster rate.

Integration Platform as a Service (iPaaS)

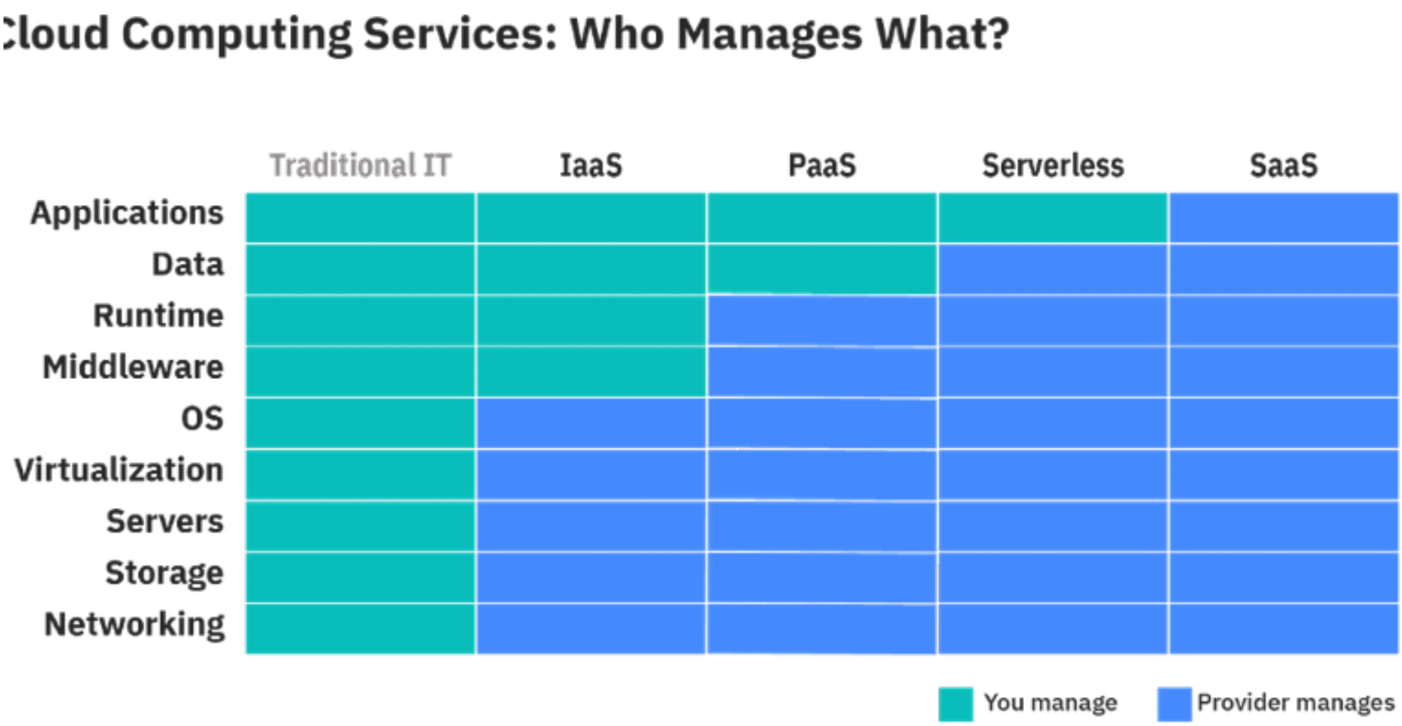
iPaaS is a suite of cloud services enabling customers to develop, execute and govern integration flows between disparate applications. Under the cloud-based iPaaS integration model, customers drive the development and deployment of integrations without installing or managing any hardware or middleware. The iPaaS allows businesses to achieve integration without big investment into skills or licensed middleware software. iPaaS used to be regarded as an integration tool for cloud-based software applications, used mainly by small to mid-sized business. In the meantime, however, a hybrid type of iPaaS, i.e. Hybrid-IT iPaaS, that connects cloud to on-premises, is becoming increasingly popular.

Serverless Computing

You cannot talk about cloud computing anymore without mentioning Serverless computing. Serverless computing (also called simply serverless) is a cloud computing model that offloads all the backend infrastructure management tasks–provisioning, scaling, scheduling, patching - to the cloud provider, freeing developers to focus all their time and effort on the code and business logic specific to their applications.

What's more, serverless runs application code on a per-request basis only and scales the supporting infrastructure up and down automatically in response to the number of requests. With serverless, customers pay only for the resources being used when the application is running—they never pay for idle capacity.

[FaaS, or Function-as-a-Service](), is often confused with serverless computing when, in fact, it's a subset of serverless. FaaS allows developers to execute portions of application code (called functions) in response to specific events. Everything besides the code - physical hardware, virtual machine operating system, and web server software management - is provisioned automatically by the cloud service provider in real-time as the code executes and is spun back down once the execution completes. Billing starts when execution starts and stops when execution stops.



Programming Terminology

Polyglot programming is the practice of writing code in multiple languages to capture additional functionality and efficiency not available in a single language. The use of domain specific languages (DSLs) has become a standard practice for enterprise application development. For example, a mobile development team might employ Java, JavaScript and employ Java, JavaScript and HTML5 to create a fully functional application. Other DSLs such as SQL (for data queries), XML (embedded configuration) and CSS (document formatting) are often built into enterprise applications as well. One developer may be proficient in multiple languages, or a team with varying language skills may work together to perform polyglot programming.

Cloud security

Traditionally, security concerns have been the primary obstacle for organizations considering cloud services, particularly public cloud services. In response to demand, however, the security offered by cloud service providers is steadily outstripping on-premises security solutions.

Nevertheless, maintaining cloud security demands different procedures and employee skillsets than in legacy IT environments. Some cloud security best practices include the following:

- **Shared responsibility for security:** Generally, the cloud provider is responsible for securing cloud infrastructure and the customer is responsible for protecting its data within the cloud - but it's also important to clearly define data ownership between private and public third parties.
- **Data encryption:** Data should be encrypted while at rest, in transit, and in use. Customers need to maintain full control over security keys and hardware security module.
- **User identity and access management:** Customer and IT teams need full understanding of and visibility into network, device, application, and data access.
- **Collaborative management:** Proper communication and clear, understandable processes between IT, operations, and security teams will ensure seamless cloud integrations that are secure and sustainable.
- **Security and compliance monitoring:** This begins with understanding all regulatory compliance standards applicable to your industry and setting up active monitoring of all connected systems and cloud-based services to maintain visibility of all data exchanges between public, private, and hybrid cloud environments.

Source articles for this page:

- What is Cloud Computing?
- What is Cloud Computing? Pros and Cons of Different Types of Services
- What Is Cloud Computing? A Beginner’s Guide | Microsoft Azure
- What is Cloud Computing & Why is it Important?
- What Is Edge Computing? Everything You Need to Know
- Edge computing
- What is Cloud Computing?