Deep Science Publishing
https://doi.org/10.70593/978-93-49910-08-9_1

Chapter 1: Building foundations for intelligent cloud infrastructure with a focus on scalability and security

1.1. Introduction

The Internet has become a service platform for almost every organization and an integral part of daily life for most people. Therefore, service users demand more from online service providers today than they did in the past. For example, a bank's online service must be available 24 hours a day, 7 days a week, perform transactions at lightning speed, and be secure from data leaks and cyberattacks. Banks are not alone in serving demanding customers. People demand a similar experience from hotels, airline systems, and almost any other service that has an online component. Service providers recognize that there is a large market and significant profit to be gained by providing a better online experience than their competitors (Armbrust et al., 2010; Amodei et al., 2016; Banerjee & Nayak, 2022).

Large online service providers themselves demand better infrastructure in order to meet the needs of the users who are placing ever-greater demands on them. They regularly report outages, which undermine consumer confidence in their ability to provide a reliable service. When major cloud services go down for hours or days, virtually every online service is impacted because they rely on one or more of these service providers. Creating an infrastructure that is massively scalable in the face of user demand and massively redundant, both geographically and in terms of mitigat-able failure points, is not easy. Indeed, these issues are compounded by the fact that the infrastructure required to scale reliable service is generally, in the early stages at least, prohibitively expensive. It is not economically feasible for all online service providers to support their own infrastructure, nor is it even likely to be necessary. A result of these competing needs of the user community and the service providers is the emergence of cloud services.

Although many enterprises are purchasing on-demand service from the cloud to reduce cost while meeting variability in workload, I believe that in order to keep the revenue or profit margins in balance with this increasing reliance on external service, enterprise growth will shift towards service enablement and increasing use of cloud service internally. Last year, we were spending two to three times stream membership services for communications and security, running middleware, database, and storage for file and database services, and hooks to connect for application services and content delivery. Such operations cannot be too difficult or expensive to support with adequately provisioned in-house infrastructure (Burns et al., 2016; Breck et al., 2017).

Fig 1.1: Building Foundations for Intelligent Cloud Infrastructure

1.1.1. Background and Significance

In name and function, cloud computing offers the illusion of a vast utilitarian resource that can be easily accessed from the desktop for virtually any and all needs. In practice, cloud infrastructures, like all computer systems, are the end products of complex and narrowly-defined resource and service interconnections. Moreover, the business models

for the enterprise clouds and service clouds available today offer continued revenue from hardware and software for the foreseeable future. The abundance of hosts, services, and data in the cloud offers attractive opportunities for users to bootstrap themselves to new revenue streams using cloud resources for mobile and compute and data-intensive applications. While the hype surrounding cloud computing disguises the tenacity of the development and growth for the infrastructure, virtualization is the secret sauce that enables seamless resource pooling and sharing at all levels - racks, data centers, clusters, and servers. It allows seamless trading of computation resources, transparent fault tolerant execution, load balancing to minimize environmental impact, and optimized usage for mobile workers.

1.2. Understanding Cloud Infrastructure

Definition and Components The cloud infrastructure concept refers to a number of physical and virtual components that must be present for clouds to work and provide the applicable cloud services. Cloud infrastructure represents the set of physical enablers such as data centers and the servers, networking components, storage, and virtualization software for the private clouds, the shared data centers, and the enablers for the public clouds, as well as the virtual components, such as cloud management platform components, the control software for managing private and hybrid clouds, and the cloud user and service catalog interfaces, which are provided for the public clouds. Cloud infrastructure provides the storage, compute performance, networking bandwidth, security, and other capabilities that users expect from clouds. Cloud infrastructure needs to be carefully constructed and managed properly to provide the expected performance and capacity without security breaches. The cloud infrastructure concept includes both a hardware and software solution stack organized as several layers, with the hardware stack containing the physical building blocks at the bottom of the stack, and the cloud services at the top of the stack. Among the different layers that form the cloud infrastructure concept, the cloud service layer is obviously the one that interfaces with the cloud users via the components, which in turn communicate with the other software components to be executed across the other layers of the cloud infrastructure stack. The lower layers of the cloud infrastructure stack are the virtualization layer, the host computing and storage virtualization layer, and layer 1, which is embedded computing based on blade servers with local storage. The middle and upper layers of the cloud infrastructure concept are important for the private clouds but not necessarily for the public clouds, which can use the cloud user interfaces to provide the public cloud services, even though this lack of additional middle layer infrastructure may limit the public cloud service scope.

1.2.1. Definition and Components

Providing services and solutions via cloud infrastructure has become common practice in the corporate world. Rather than buying and hosting services in their offices, big and small businesses nowadays prefer to pay a cloud provider for the services they need, whenever they need them. The Internet provides access to both the technical features and the data of the business, from anywhere, making remote work possible. It also provides computing power and services support to the professionals in charge of implementing and servicing the different software tools that need technical and management skills to be used by the business. In times of economic crisis, reducing the expenses and workload of the IT team is advisable, so with a small additional cost, the implementation of business solutions on cloud infrastructure is increasingly common.

Cloud infrastructure is composed of the hardware components that allow data to be stored and software solutions to be provisioned and accessed via the Internet, as a service. These data and solutions are provided from servers and data centers owned and maintained by third companies, which are called cloud service providers. Cloud infrastructure is made of both physical infrastructure and logical infrastructure. On one side, there are the physical components that make the cloud service accessible to the business – physical servers and data centers, routers, firewalls, and connectivity, among others. On the other side, there are the logical components that make the provisioning of multiple logical resources using the same physical resource possible, such as hypervisors, storage solutions, and resource management software tools, among others.

1.2.2. Types of Cloud Services

Cloud computing operates within a computing service-sharing ecosystem through which various vendors offer a multitude of service types to their customers. Cloud services can be broadly classified into three types: Software as a Service (SaaS), Platform as a Service (PaaS), and Infrastructure as a Service (IaaS).

SaaS enables cloud providers to deliver and manage fully functional and complete software solutions to their customers. The customers only need to use the software, and do not have to worry about hardware and operating system platforms or middleware components. Because the SaaS solutions are built and delivered by the cloud providers, cloud customers have little control over the offering. They can only customize this solution to a very small degree through configuration options offered by the provider. Cloud providers can therefore deal with large volumes of customers, which enables customers to use these offered software solutions at less cost.

The next type of cloud service, PaaS, enables customers to build their own applications on the platform provided by the cloud vendors. The cloud customer is responsible for

developing and deploying their applications, after which the cloud provider will manage the middleware and other layers of the cloud stack for the customer. The PaaS solution gives customers more flexibility and control than SaaS, but not to the level of IaaS.

1.3. Importance of Scalability

To derive the greatest value from the cloud (or at least the least guilt), cloud environments must support all three characteristics of cloud computing. Of these three characteristics, scalability can most influence how systems are constructed for the cloud and what the primary goals of these systems are. The noted cloud characteristics: elasticity, broad network access, rapid resource provisioning, measured service, on-demand provisioning, resource pooling, and data storage, can all be influenced by how systems scale within a cloud environment, and yet the question still remains: "What is scalability and how do we implement it?" This section aims to help the reader understand more about scalability in this context. Supporting systems with the ability to easily and effectively scale is necessary in order for the systems themselves to efficiently use the greater cloud infrastructure as well as to provide fault tolerance and reliability without adding additional management complexity. The ultimate goal is to allow the cloud consumer to focus on their application and how it will deliver business value without having to worry about the cloud and its resources.

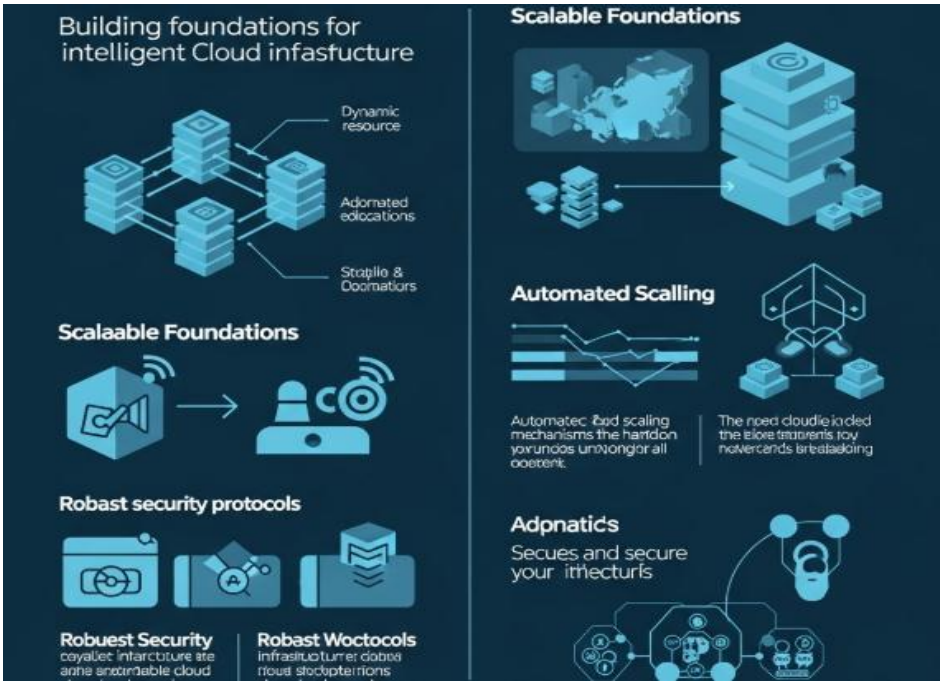

Fig 1.2: Scalability of Building Foundations

Fortunately, as with other principles for designing and using cloud infrastructure, scalability can be defined in certain terms. When defining scalability, a modular approach is encouraged. That is, defining the scalability of a piece of computer or a system only touches upon one part of the entire solution. Typically, scalability is defined as a system's ability to handle a growing amount of work by adding resources to the system. In cloud computing, the growing work is usually an increasing number of transactions, which scales the storage, compute, and transaction layers of an implemented solution. Systems need to be designed such that, when load requirements dictate, the additional resources are inexpensive to provision and can be effectively utilized without additional automated management processes. However, just as a system can scale by using the principle of additional resources, a system can also scale down.

1.3.1. Defining Scalability

The concept of cloud computing has become significantly popular with the ability to provide scalable services that allow consumers to pay according to their usage. Scaling up or down with minimum change in service or with little involved time and effort is very easy with cloud computing. With many advantages over traditional data centers, consumers are rapidly moving their applications and services to the cloud. Cloud is therefore providing business consumers to focus on their core competencies without worrying about the underlying infrastructure. It is also evident that future applications of cloud architecture will require addressing advanced issues such as security, availability, reliability, support of real-time services, multi-tenancy, and horizontal scalability. Scalability allows dynamic scalability of the data and service on demand according to consumer needs. Scalability controls the possible timescale and adjusts the treatment strategy, e.g., increasing physical resources in a limited period of time according to requirement, and reducing resources during a quiet period of the consumer's life cycle. This will make service needs more efficient and economical.

To understand how meaningful the concept of scalability is, we must consider with some care what meaning we have attached to the term. Scalability may refer to some simple procedure defined on a system, for example, that solution to a problem can be obtained by adding together the solutions of some subproblems that partition the original problem; or that solutions to problems of size n may be computed by a finite number of time-bounded sequential or parallel operations on an idealized machine. In the case of a parallel operation, it is usually a requirement that the number of operations be independent of n; otherwise, it merely reflects the notion of finite speedup. Our general understanding is that, when a system does have the ability to scale according to a proposed definition, in some form we can alter, augment, or construct a more efficient algorithm for the larger size system, without anticipating drastic changes, both in the

techniques that we will employ and the time required. Since many such algorithmic techniques are based on ever finer or coarser subdivisions of some key feature of the problem, such methods must be more efficient as the size of the divisions tends toward an appropriate limit. The notion of scalability would appear to extend to the versatile systems composed of the components with typically available speedup.

1.3.2. Types of Scalability

Scalability has different types. It can be classified as vertical and horizontal scalability. Mathematically, a function f is said to be vertically scalable if for some value x the function value f(x) is said to be vertically scalable if f(y) is very large compared to f(x) if y is much larger than x. On the other hand, a function f is said to be horizontally scalable if it is possible to make it a horizontal scale, it is for some value x if several values of y exist between zero and a(x) such that for each i between 1 and n, f(y(i)) is very large compared to f(x) if y(i) is much larger than x.

Vertical scalability designates a situation where a support is made ready to be more quickly able to accept requests coming from a client. The physical structure of such a facility or services becomes more important than whatever action that services might perform. In order to have more to accept to offer more quickly, the service upgrade becomes mandatory. Upgrading a service presents a disadvantage. It needs to take the functioning service offline or, if a service has to be online all time but offer a little accessibility, redundancy must be built. One strategy consists in leaving the original service running while slowly upgrading each of the service components. With horizontal scalability, the opposite situation is being considered. Response time is very important and thus it is more performant to have a concentration of components acting at the similar same on the service. Avoiding the upgrade of already existing services might also be a reason for horizontal scalability. Cold upgrades are being concerned. They do not need to be done each time we have an upgrade of service.

1.4. Security Considerations in Cloud Infrastructure

Building cloud infrastructure needs to keep several security best practices in mind. Compute and storage configuration, network management, and intelligent services need to have security principles integrated into their design, to avoid technology implementation with high risk of security breaches. Security capabilities and settings need to be deployed to protect against unintended exposure of cloud resources and susceptibility to threats. When building cloud infrastructure, the entire architecture needs to ensure protection against cyber attacks. Because technology risks are always evolving, there is a need for ongoing periodic evaluations to minimize inherent threats and

vulnerabilities based on the changing cyber landscape. Automated tools are available for threat and vulnerability discovery, automation, and optimization. Applying monitoring and cyber hygiene practices can help accelerate response and recovery from an instance of digital interference.

Enterprise workloads are often moved to the cloud because of the potential for faster deployment, consumption-based pricing, and reduced operational overhead. But often, organizations are hesitant or unwilling to move their sensitive workloads to the cloud due to concerns regarding governance, risk, and compliance.

1.4.1. Common Threats and Vulnerabilities

With the rapid development of cloud technology, security in cloud computing infrastructure has become one of the current hot research topics. The specific security requirements, architectures, and models for cloud security are still being developed and will become clearer as more field studies are performed. With more work in the area and more experience being implemented, particularly with specialist tools such as cloud security gateways and risk and compliance services, this area's requirements and standards will converge. Nonetheless, the general requirements over security in cloud computing architectures covered in this chapter have been identified as being the key ones. Like any technological advance, the cloud model offers a number of benefits, as well as a set of new risks and challenges. While there are undoubtedly many exciting business opportunities set to emerge from early implementation of the cloud model, there are many risks and challenges that will hinder the model's development until practical solutions are developed. This chapter seeks to identify the risks and challenges as discussion points for further development and standardization.

While new and innovative technologies and solutions may emerge in order to mitigate and address some of these issues over the next few years, quite a number of the risks and challenges identified will likely still remain and will require careful and structured strategic management. Users of cloud implementations will be faced with these risks and challenges in all likelihood as soon as they sign off on having their business placed in the hands of cloud providers. These risks and challenges will also exist long before business services on top of the cloud model are developed, tested, and implemented, and will remain for the foreseeable future. As mentioned, some will not be eliminated entirely through the use of cloud solutions. Thus, it is impossible to overstate the importance of determining how applicable security best practices and key management functions are to activities and services in the cloud.

1.4.2. Security Best Practices

Cloud computing enables on-demand, scalable virtualization technologies, which in turn facilitate the renting of computing resources such as CPU, memory, storage, network connectivity, etc. Transparency, flexibility, scalability, and elasticity offer cloud users many advantages, including not only the reduction of upfront capital expenses for enterprises but also an astonishing growth in the economy around the cloud. However, these features, together with resource multiplexing, also create many challenges for securing data when users from different tenants or domains share the same physical resources. This chapter primarily focuses on the security specifically in the cloud virtualization layers of the mainstream multi-tenant cloud models, because it is at these layers where the best security solutions must exist in order to better influence the overall productivity of the entire cloud economy. The solutions that we describe in this chapter lay the critical foundations upon which the upper SaaS levels can build upon to create concrete security technology.

Designing cloud infrastructure that emphasizes security is essential for privacy, compliance, regulatory, and legal requirements. Security in the cloud is a shared responsibility between the customer and the cloud vendor. However, cloud vendors have developed cloud infrastructures that provide users with hyper-secure configurations by default. It is therefore important to understand the best practices nearby vendors and experts recommend, as well as how to implement them on cloud machines. A major enabler of security in the cloud is a multi-project architecture. Multiple clinical projects should never share a single cloud project.

1.5. Designing for Scalability

As engineers, we strive to build a system that supports the current requirements adequately now and is also capable of absorbing the estimated growth in the future. Scalability is one of the quality attributes that are key to the success of a system. In fact, users often judge the quality of the service they get based on performance metrics like latency and throughput. Therefore, we will start exploring the various architectural patterns and design aspects that we can leverage to build a scalable infrastructure for providing services.

There are many architectural patterns that we can use to design a solution. The most common and less complex architectural pattern for small load is a Monolithic design. As the load increases and request volume exceeds a certain level, scaling a Monolithic application would require scaling up using larger servers, which may be limited by resource capability. At this point, it is best to help refactor the monolithic based design into a Service-oriented or Microservices or FAAS based architecture. Augmenting a

Monolithic architecture with caching, Partitioning, Queuing, Asynchronous Communication, Data Replication techniques, and Load Balancing algorithms will also create a robust architecture. These are the architectural patterns leveraged by organizations to design scalable solutions.

Load Balancing is an important aspect that has to be considered, both at the network level and at the application level. The key to scalability and consequent performance assessment for any service is load balancing. An efficient load balancer will be able to map the incoming requests to the active server in an efficient manner, allocating load to each instance evenly without letting it be a bottleneck. This can be achieved by using different load balancing algorithms efficiently. Load balancing helps in Failover Management, Test and Maintenance of servers, Load Priority Management and SSL Offloading, which assists in delivering a secure service.

1.5.1. Architectural Patterns

Scalable architectures are designed with business-specific and technology-specific factors in mind. While some principles for scalable architectures are common, there is no single architectural design that will meet every need. Experts recommend use of some of the following patterns when designing systems for scale.

Using caches If parts of your application require processing of complex data or computation, or make calls to external services, place caches to reduce costs and speed things up. These caches can be held in memory or implemented as distributed caches in a cloud.

Using asynchronous processes Some application processing can occur asynchronously after the user transaction has been completed. Possible examples include cancelling a transaction, notifying a user, or scraping and caching third-party data. Many business processes use task queue solutions to allow the user to keep working while handling post-transaction processing outside the critical customer transaction path.

Microservices If your application has frequent but small feature updates, is experiencing scale in specific areas while not in others, and is not already an extremely modular monolithic or SOA architecture, consider splitting into microservices. This is a key pattern that many companies use, resulting in each individual microservice being inexpensive to deploy and maintain while allowing scaling for just that microservice when needed. For an entire application with numerous microservices, tools such as containers or serverless functions can help achieve the desired cost balance as well.

Mobile offloading While desktop users expect instant response times, mobile users are more tolerant of short delays. Offload mobile processing or transactions to a backend

system that can handle batch processing, thus allowing the mobile app to quickly end the transaction.

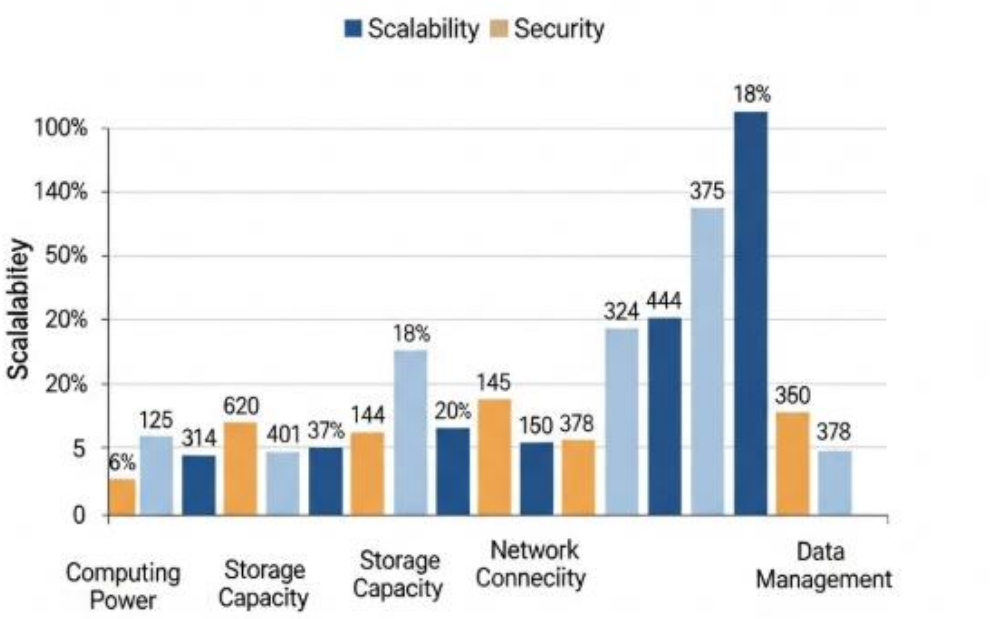

Fig : Cloud Infrastructure with a Focus on Scalability and Security

1.5.2. Load Balancing Techniques

Load balancing is an important part of the design process for scalable solutions. It considers distributing workloads across multiple resources or providing money-saving redundancy within a single system. Financially incentivized load balancing can also be found in applications that distribute work to consumer-grade personal computers. Load balancing can be accomplished in at least three ways: a central load balancer between clients and endpoints, clients working together to choose combinations of endpoints, or endpoints collaborating to bring clients together in ad hoc sessions. Load balancing can appropriately use DNS for small-scale systems, but there are also commercial solutions available that use algorithms to constantly adjust connections across systems.

To provide service, endpoint systems take requests from clients and respond. In many designs, requests generate work that is then scheduled, then executed, and then results sent back to the original client. Server and endpoint system implementations scale out the execution platform, normally into a grid, to allow the requested work of many clients to be processed rapidly. By further decomposing a piece of work into subtasks, larger problems can be split across many grid elements. When the execution is complete, results are sent back to other systems for return to client systems.

Many services are designed to have result transfer handled automatically by the clients. Clients listen for results, thus asynchronously recovering from transient faults. By automatically restarting in a timely way, clients can deal with failed systems or simple overload cases. Unlike in many old designs, results do not execute on APIs on the original client systems but just receive. In this way, both clients and service endpoints implement an appropriate load balancing on a micro and a macro scale. A wide variety of task types may be distributed using this type of approach, from video rendering to massive simulations.

1.6. Conclusion

Even if cloud computing is now well established, there are some aspects in the management of cloud resources and services that still need to be improved. Some drawbacks of the current cloud scenarios include limited scalability and difficulty in coping with high scalability demands, high prices for providers and end-users, difficult resource management and load balancing, resource failures and services overloading, and security and privacy issues. Some innovative solutions are already being studied or developed to address the aforementioned issues. Smart-specialty and geo-distributed clouds, content and service delivery marketplaces, cloud brokers and hybrid clouds, cloud federation, and cloud architecture and infrastructures are essential to allow for handling and for supporting future scalability trends. The demand for cloud services will sharply rise in coming years. The inevitable emergence of future trends, with particular reference to the increasing number of devices and users that require any type of service, will require upcoming cloud providers to heavily invest in porting into the cloud any efficient solution for security and privacy issues, resource failure handling, and resource management, monitoring, and proactive prediction. Service and resource delivery will also need to be fine-tuned, and many possible cloud scenarios must be prepared inside and outside cloud users companies and power and maintenance issues must be highlighted also when requesting such service. On the other hand, for cloud users, either small or medium enterprises, or students and researchers, cloud services are attractive primarily for the economical advantages they offer. The price of cloud services, and in general of IT resources and services, must balance the mass adoption by users with the further growth of service and cloud provider companies. There need to be healthy ecosystems to allow for managed and maintained future trends.

1.6.1. Future Trends

Although cloud computing is still a relatively young industry compared to other technology sectors, its evolution over the past two decades has already resulted in a

dizzying array of services and capabilities. As we look toward the horizon, we can expect to see advances in artificial intelligence, distributed architectures, operational simplicity, edge computing, and related capabilities that will continue to augment and advance the core tenets of the cloud vision, which is to provide each organization with the IT infrastructure and tools to enable them to focus on their mission. Key trends will include further integration of workloads and tools to operate them across on-premises, public cloud, and edge deployments; an evolving technology stack, with a wider spectrum of choices for customers across domains, such as distributed file systems, databases, operational orchestration, etc.; heavy enterprise focus on security and application resilience, driven both by regulations and a wider threat landscape; increasing automation and simplification of how all infrastructure is operated, including infrastructure as code technologies of all sorts, inference-driven recommendations about capacity and resource allocation; integration of hyperscale infrastructure to key edge locales, enabling scale and performance for the edge while minimizing latency and egress; and finally, evolving technologies that tie together what has traditionally been seen as infrastructure management versus application and workload management, as application workloads.

References

Amodei, D., Hernandez, D., Sastry, G., et al. (2016). Concrete problems in AI safety. arXiv preprint arXiv:1606.06565.

Armbrust, M., Fox, A., Griffith, R., et al. (2010). A view of cloud computing. Communications of the ACM, 53(4), 50–58.

Banerjee, S., & Nayak, R. (2022). Intelligent Cloud Computing for Smart Innovation. Springer.

Breck, E., Cai, S., Nielsen, E., Salib, M., & Sculley, D. (2017). The ML test score: A rubric for ML production readiness and technical debt reduction. Proceedings of NIPS 2017 Workshop on ML Systems.

Burns, B., Grant, B., Oppenheimer, D., Brewer, E., & Wilkes, J. (2016). Borg, Omega, and Kubernetes. Communications of the ACM, 59(5), 50–57.