Websites:
http://www.sciencepub.net/newyork
http://www.sciencepub.net

Emails:
newyorksci@gmail.com
editor@sciencepub.net

New York Science Journal

MARSLAND PRESS
Multidisciplinary Academic Journal Publisher

MODERN COMPUTING: VISION AND CHALLENGES

Dr. Vivek Kumar

Assistant Professor In Computer Science, Hari College of Management, Hindon Bridge, Dehradun Road, Gagalheri-247669, Saharanpur (Uttar Pradesh)
viveks865@gmail.com

Abstract: Over the past six decades, the computing systems field has experienced significant transformations, profoundly impacting society with transformational developments, such as the Internet and the commodification of computing. Underpinned by technological advancements, computer systems, far from being static, have been continuously evolving and adapting to cover multifaceted societal niches. This has led to new paradigms such as cloud, fog, edge computing, and the Internet of Things (IoT), which offer fresh economic and creative opportunities. Nevertheless, this rapid change poses complex research challenges, especially in maximizing potential and enhancing functionality. As such, to maintain an economical level of performance that meets ever-tighter requirements, one must understand the drivers of new model emergence and expansion, and how contemporary challenges differ from past ones. To that end, this article investigates and assesses the factors influencing the evolution of computing systems, covering established systems and architectures as well as newer developments, such as serverless computing, quantum computing, and on-device AI on edge devices. Trends emerge when one traces technological trajectory, which includes the rapid obsolescence of frameworks due to business and technical constraints, a move towards specialized systems and models, and varying approaches to centralized and decentralized control. This comprehensive review of modern computing systems looks ahead to the future of research in the field, highlighting key challenges and emerging trends, and underscoring their importance in cost-effectively driving technological progress.
[Kumar, V. **MODERN COMPUTING: VISION AND CHALLENGES.** *N Y Sci J* 2024;17(5):32-35]. ISSN 1554-0200 (print); ISSN 2375-723X (online). http://www.sciencepub.net/newyork. 05. doi:10.7537/marsnys170524.05.

Keywords: Modern Computer, Vision, Types, Challenges

Introduction:

Introduction The Internet, the expansive computational backbone of interactive machines, is largely responsible for the 21st century's social, financial, and technological growth [1]. The growing reliance on the computing resources it encapsulates has pushed the complexity and scope of such platforms, leading to the development of innovative computing systems. These systems have genuinely improved the capabilities and expectations of computing equipment driven by rapid technical and user-driven evolution [2]. For instance, vintage mainframes combined centralized data processing and storage with transmission interfaces for user input. Due to advancements in clusters and packet-switching technologies, microchip gadgets, and graphical user interfaces, technology originally shifted from big, centrally-run mainframe computers to Personal Computers (PCs). The globalization of network standards made it possible for interconnected networks worldwide to communicate and share data [3]. Businesses slowly combined sensor and actuator goals with built-in network connectivity by creating architectures and standards that submit tasks to remote pools of computing resources, such as memory, storage, and data processing [4]. As a result, newer models like the Internet of Things (IoT) and edge computing are now beginning to expand the reach of technology outside the confines of traditional network nodes [5]. Over the past six decades, computing models have fundamentally shifted to address the problems posed by the ever-evolving nature of our civilization and its associated computer system architectures [6]. The evolution of computing from mainframes to workstations to the cloud to autonomous and decentralized architectures, such as edge computing and IoT technologies, however, maintains identical core parts and traits that characterize their function [7]. Research in computing underpins all of them! Advancements in areas like security, computer hardware acceleration, edge computing, and energy efficiency typically serve as catalysts for innovation and entrepreneurship that span across various business domains [8]. While computing systems and other forms of system integration create new problems/opportunities, software frameworks have been developed to address them. Thus, middleware, network protocols, and safe segregation techniques must be continually developed and refined to support novel computing systems—and their innovative use cases.

Early Computing to Modern Computing:

A Vision Over the last six decades, advancements in computing systems have optimized the efficiency of the available hardware [12]. Over this time period, novel computing models and innovations have been developed and replaced the previous state-of-the-art, all of which incrementally contribute to the current technology status [2]. Fig. 1 shows the transition from early computing to contemporary computing. Originally, a single system could only carry out a single task; hence, a user needed various systems working in tandem to achieve their desired tasks. However, to safely share information between computers—in order to overcome the problem of executing only one task at a time—a reliable communication mechanism is essential [13]. To that end, our investigation unfolds across three key sections: Section 3 delves into the evolution of computing paradigms, emphasizing technological drivers. Section 4 offers a comprehensive classification of computing systems. The discussion in Section 5 revolves around the impact and performance criteria of modern computing. Section 6 introduces the hype cycle for modern computing systems, spotlighting emerging trends.

Evolution of Computing Paradigms:

Technological Drivers Figure 1 illustrates the progression of computing technology starting from the year 1960.

Client Server

In the year 1960, a centralized platform (a.k.a, distribution integration) was developed to share workloads (a.k.a., jobs) between the resource providers (i.e., server instances) and service consumers (i.e., customers) [12]. Supporting it, a networking system was utilized for communications between client devices and servers, and servers exchange resources for customers to perform their tasks using a load balancing mechanism [14]. Illustrative examples of the clientserver model's application include the Email and the World Wide Web (WWW). However, users in this configuration were unable to freely interact with one another.

Supercomputer

A supercomputer is a powerful computer with extraordinary processing capability, such that it can handle complex calculations in several areas of science, including climate study, quantum physics, and molecular simulation [15]. Energy utilization and heat control in supercomputers endured as a key research problem throughout their growth in the 1960s [16]. Supercomputers, such as Multivac, HAL-9000, and Machine Stops, have been instrumental in underpinning/enabling dramatic technological advancements [14].

Proprietary Mainframe

To handle massive amounts of data (including dealing with transactions, customer data analysis, and censuses), a high-speed machine with large computing power is required [17]. Virtualization on mainframes allows for increased efficiency, protection, and dependability. In the year 2017, IBM announced the newest version of its mainframe, the IBM z14 [13]. Being built to support massive economic activity and despite their high price tag, mainframe computers deliver outstanding efficiency [14].

Cluster Computing

Cluster computing is a method of increasing the efficiency of a computing system by utilizing several nodes to complete a single operation [18]. In order to coordinate various computing nodes, this type of technology requires a rapid Local Area Network (LAN) for exchanging information among them [19].

Home PCs

The early days of the Internet coincided with the flourishing of Personal Computers (PC) kept at one's home [3]. The Internet was evolving into a foundational network, connecting local networks to the larger Internet using self-adaptive network protocols, such as Transmission Control Protocol/Internet Protocol (TCP/IP)—in contrast to the original Network Control Protocol (NCP)-based Advanced Research Projects Agency Network (ARPANET) mechanisms [2]. As a result, there was a sharp increase in the number of hosts on the Internet, which quickly overwhelmed centralized naming technologies like HOSTS.TXT. In the year 1985, the earliest publicly available version of a Domain Name System (DNS) was released for the Unix BIND system [20]. This system translates hostnames into IP addresses. Pioneer Windows, Icons, Menus, and Pointers (WIMP)-based Graphical User Interfaces (GUIs) on computers, such as the Xerox Star and the Apple LISA, proved that customers could successfully use machines in their homes for tasks like playing video games and surfing the Internet [21].

Open MPP/SMP

Massive Parallel Processing (MPP) and Symmetric MultiProcessing (SMP) systems are the two most common forms of parallel computing platforms [16]. In an SMP setup, multiple processors run the same Operating System (OS) concurrently while sharing the rest of the hardware's capacity (e.g., disc space and RAM). Naturally, resource pooling influences the computational speed of completing a given assignment. In an MPP scenario, the file system can be shared, while no other resources are pooled for use during task processing [14].

Grid Computing

This technology enables a group to work together towards the same objective by executing non-interactive, and largely IO-intensive tasks [19]. Each application running on only one grid is a top priority [12]. In addition to allocating and managing resources, grid computing also offers a reliable architecture, as well as tracking and exploration support.

WWW

The primary web browsers, websites, and web servers all came into existence in the later stages of the 1980s and early 1990s, underpinned by the development of Hyper Text Transport Protocol (HTTP) and Hyper Text Markup Language (HTML) [2]. The platform for the interconnected system of networks that makes up the WWW was made possible by the standardizing technology of TCP/IP network protocols. This allowed for a dramatic increase in the total number of servers linked to the Web and introduced Information Technology (IT) to the general public. Software applications were thus able to communicate with one another beyond address spaces and networking, e.g., via novel technologies like Remote Procedure Calls (RPCs) [22].

Commodity Clusters

Commodity cluster computing employs several computers simultaneously, which can inexpensively execute user tasks [19]. In an effort to standardize their processes, several companies use open standards while building commodity computers [14]. This allowed immediate computing business needs to be met using ready-made processors.

Peer to Peer (P2P)

P2P is a distributed framework to share workloads or jobs amongst multiple peers; alternatively, computers and peers may interact with one another openly at the application layer [23]. With no mediator in the center, users of a peerto-peer system can share resources like memory, CPU speed, and storage space. Peer-to-peer communication utilizes the TCP/IP protocol suite [24]. Interactive media, sharing file infrastructure, and content distribution are some of the most common use cases for P2P technology.

Web Services

The technology supporting web services enables the exchange of data between various Internet-connected devices in machine-understandable data formats, such as JavaScript Object Notation (JSON) and Extensible Markup Language (XML), over the WWW [25]. Commonly, web-based services operate as a connection between end users and database servers.

Service-Oriented Architecture (SOA)

The SOA paradigm enables software elements to be reused and made compatible through advertised service designs/APIs [26]. It is normally easier to include services in new apps: the apps can be architected to adhere to standardized protocols and leverage consistent design patterns. This frees the software engineer from the burden of recreating or duplicating current features or figuring out how to link to and interoperate with current systems—e.g., via using Software Development Kits (SDKs) that implement common functionalities, such as networking, retries, marshalling of data and error handling [27]. Each SOA API exposes the logic and data necessary to carry out a single, self-contained business operation (such as vetting the creditworthiness of a client, determining the loan's due date, or handling an insurance application) [28]. The loose integration provided by the service's design allows for the service to be invoked with limited knowledge of the underlying service implementation.

Virtualized Clusters

Virtualization enables a guest computer system to be implemented on top of a host computer system, which abstracts away the problem of physically supporting and maintaining multiple types/architectures of physical machines [19]. With a virtualized cluster, several Virtual Machines (VMs) may pool their resources to complete a single job. VM hypervisors, which execute the guest system on the host system, allow software-based virtualization to run either on top of an OS or directly (bare-metal) on hardware [14]. Costs and complexity are reduced, and a greater number of tasks may be completed with identical hardware by adopting a VMbased system.

High Performance Computing (HPC) System

HPC is the computing method of choice when dealing with computationally intensive issues, which tend to arise in the domains of commerce, technology, and research [14, 19]. A scheduler in an

HPC system manages accessibility to the various computing resources available for use in solving various issues [29]. HPC systems utilize a pooled set of resources, allowing them to perform workloads or tasks via the allocation of concurrent resources and online utilization of various resources.

Autonomic Computing

One of the first global initiatives to build computer systems with minimum human involvement to achieve preset goals was IBM's autonomic computing program in 2006 [30]. It was mostly based on research on nerves, thinking, and coordination. Autonomic computing research examines how software-intensive systems may make choices and behave autonomously to achieve user-specified goals [4]. Control for closed- and open-loop systems has shaped autonomic computing [31]. Complex systems can have several separate control networks.

Mobile Computing

The term "mobile computing" is used to describe a wide range of IT components that give consumers mobility in their usage of computation, information, and associated equipment and capabilities [32]. An especially popular definition of "mobile" is accessing information while moving, when an individual is not confined to a fixed place. Accessibility at a fixed spot may also be thought of as mobile, especially if it is provided by hardware that consumers can move as needed but that remains in one place while functioning [33]. Mobile computing devices are becoming essential across industries, boosting efficiency and creativity in fields such as healthcare, retail, manufacturing, and the arts.

Cloud Computing

Software as a service (SaaS), platform as a service (PaaS), and infrastructure as a service (IaaS) are all examples of Internet-accessible web services [1]. Google Mail is an excellent instance of a SaaS product since it provides a wide range of useful features without the burden of installation and ongoing upkeep costs. PaaS providers like Microsoft provide a scalable environment where users can install their applications [34]. Amazon is a prime instance of an IaaS provider since it provides users with access to servers, networks, storage, and other hardware components necessary to run applications and other workloads efficiently and effectively. Using distant facilities for performing user operations (processing, administration, and storage of data) over the Internet is known as "cloud computing," abbreviated as "XaaS," where X = "I," "P," "S," etc. Cloud computing enables the pooling of resources to reduce execution costs and enhance service accessibility [35]. There are four major types of cloud computing systems: public, private, hybrid, and communal. Dependability, safety, and cost-effectiveness are just a few examples of Quality of service (QoS) characteristics that should be considered while developing a successful cloud service.

References

[1] R. Buyya et al., "A manifesto for future generation cloud computing: Research directions for the next decade," ACM Computing Surveys (CSUR), vol. 51, no. 5, pp. 1–38, 2018.
[2] D. Lindsay et al., "The evolution of distributed computing systems: from fundamental to new frontiers," Computing, vol. 103, no. 8, pp. 1859–1878, 2021.
[3] R. Yamashita, "History of personal computers in japan," International Journal of Parallel, Emergent and Distributed Systems, vol. 35, no. 2, pp. 143–169, 2020.
[4] S. S. Gill et al., "Ai for next generation computing: Emerging trends and future directions," Internet of Things, vol. 19, p. 100514, 2022.
[5] J. Gubbi et al., "Internet of things (iot): A vision, architectural elements, and future directions," Future Generation Computer Systems, vol. 29, no. 7, pp. 1645–1660, 2013.
[6] R. Muralidhar et al., "Energy efficient computing systems: Architectures, abstractions and modeling to techniques and standards," ACM Computing Surveys (CSUR), vol. 54, no. 11s, pp. 1–37, 2022.
[7] A. Chakraborty et al., "Journey from cloud of things to fog of things: Survey, new trends, and research directions," Software: Practice and Experience, vol. 53, no. 2, pp. 496–551, 2023.
[8] A. Beloglazov et al., "Energy-aware resource allocation heuristics for efficient management of data centers for cloud computing," Future Generation Computer Systems, vol. 28, no. 5, pp. 755–768, 2012.
[9] V. Casamayor Pujol et al., "Fundamental research challenges for distributed computing continuum systems," Information, vol. 14, no. 3, p. 198, 2023.
[10] J. Shalf, "The future of computing beyond moore's law," Philosophical Transactions of the Royal Society A, vol. 378, no. 2166, p. 20190061, 2020.

4/25/2024