Data Integration and Large Scale Analysis

07- Cloud Computing Fundamentals

Dr. Lucas Iacono - 2025

HCC
Institute of
HUMAN-CENTRED COMPUTING

TU Graz

Part B

Large-Scale Data Management & Analysis

- Cloud Computing
  - Fundamentals [Nov 21]
  - Resource Management and Scheduling [Nov 28]
  - Distributed Data Storage[Dec 05]

Institute of HUMAN-CENTRED COMPUTING

Part B

Large-Scale Data Management & Analysis

- Large-Scale Data Analysis
  - Distributed, Data-Parallel Computation [Dec 12]
  - Distributed Stream Processing [Dec 19]
  - Distributed Machine Learning Systems [Jan 16]

HCC
Institute of HUMAN-CENTRED COMPUTING

TU Graz

Agenda

- Motivation and Terminology
- Cloud Computing Service Models
- Cloud, Fog, and Edge Computing

Motivation and Terminology

Motivation and Terminology

- How new it is?

Motivation and Terminology

60s

John McCarthy conceptualize the idea of computing as a **public utility** at a lecture at MIT, envisioning "computing on demand."

ARPANET is developed, enabling computational resources sharing over a network.

70s

IBM launches CP/CMS (Control Program/Cambridge Monitor System), an early virtualization system that allows multiple users to run applications on a single mainframe.

Distributed computing concept emerges, focusing on resource sharing across multiple machines **(cluster computing)**.

80s

CompuServe starts offering small-scale cloud-like storage

Tim Berners-Lee invents the World Wide Web

90s

Salesforce launches its CRM SaaS platform

AWS presents EC2 and S3

Motivation and Terminology

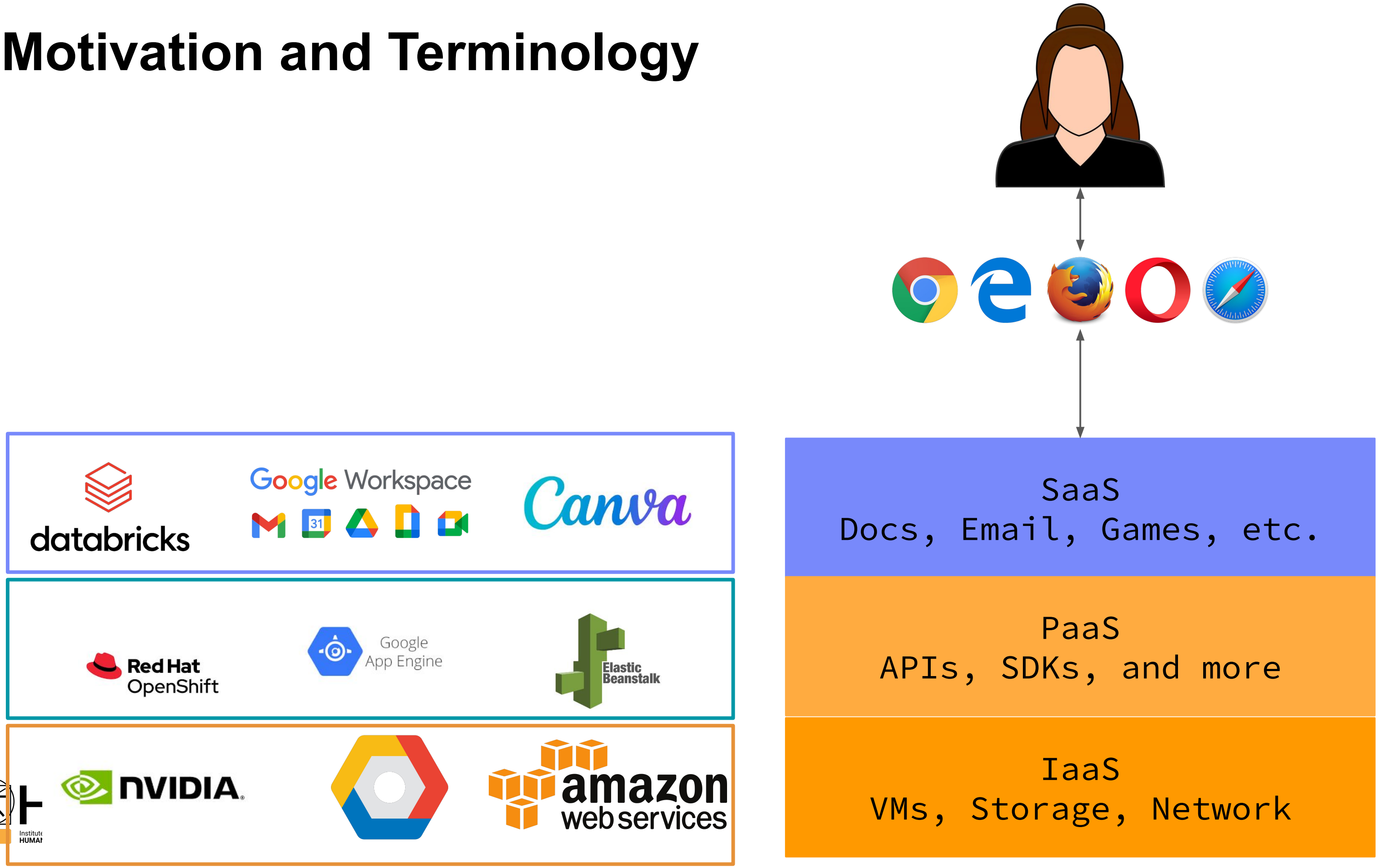
- ## Definition
  - “A Cloud is a type of **parallel** and **distributed system** consisting of a collection of interconnected and **virtualized computers** that are dynamically provisioned and **presented as one or more unified computing resource**(s) based on service-level **agreements** established through negotiation between the **service provider and consumers**” [*].


Future Generation Computer Systems

Cloud computing and emerging IT platforms: Vision, hype, and reality for delivering computing as the 5th utility

Buyya, R., Yeo, C. S., Venugopal, S., Broberg, J., & Brandic, I. (2009). Cloud computing and emerging IT platforms: Vision, hype, and reality for delivering computing as the 5th utility. Future Generation computer systems, 25(6), 599-616.

Motivation and Terminology



Motivation and Terminology

- **Transforming IT Industry/Landscape**
  - **Since ~2010** increasing move from on-prem to cloud resources
  - System **software licenses** become increasingly **irrelevant**
  - Few cloud providers dominate IaaS/PaaS/SaaS markets
  - **2023** revenue:
    - Microsoft ($ 111.6B)
    - Amazon AWS ($ 88B)
    - Oracle Cloud ($ 35.3B)
    - IBM Cloud ($ 20.8B)
    - Google Cloud (8.41B)
    - Alibaba Cloud ($ 3.789M)

Motivation and Terminology

- **Argument #1: Pay as you go:**
  - **No upfront cost** for infrastructure
  - **Variable** utilization ➔ over-provisioning
  - **Pay per use** or acquired resources
- **Argument #2: Economies of Scale**
  - **Lower cost** for purchasing and managing IT infrastructure at scale ➔ lower cost (applies to both HW resources and IT infrastructure/system experts)
  - Focus on **scale-out on commodity HW** over scale-up ➔ lower cost

Motivation and Terminology

- **Argument #3: Elasticity**
  - System can scale up according to demand
  - **Virtually unlimited resources** allows to reduce time as necessary ((Task time = Time x Resources))
- **Argument #4: Availability**
  - Resources are available 24x7

Characteristics and Deployment Models

- **On-demand self service.** Users can access and manage computing resources themselves without requiring human assistance.
- **Broad network access.** Services are accessible from anywhere with an internet connection, using various devices.
- **Resource pooling.** Computing resources are shared among multiple users through virtualization, securely and efficiently

Characteristics and Deployment Models

- **Rapid elasticity.** Resources can be scaled up or down quickly based on demand, in real time.
- **Measured service.** Resource usage is monitored and recorded to optimize consumption and bill only for what you use.

Cloud Computing Service Models

(computing as a utility)

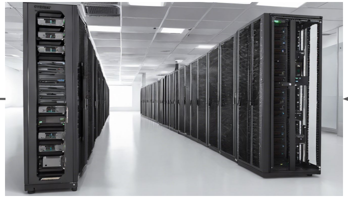
HCC
Institute of
HUMAN-CENTRED COMPUTING

Anatomy of a Data Center

CPUs

Servers (Sockets, RAM, Disks)

Rack
(16-64 servers + rack switch)



Cluster
(++Racks + Cluster switch)

Data center (Google Netherlands)

Fault Tolerance

- **Yearly Data Center Failures**
  - ~0.5 **overheating** (power down most machines in <5 mins, ~1-2 days)
  - ~1 **PDU failure** (~500-1000 machines suddenly disappear, ~6 hrs)
  - ~1 **rack-move** (plenty of warning, ~500-1000 machines powered down, ~6 hrs)
  - ~1 **network rewiring** (rolling ~5% of machines down over 2-day span)
  - ~20 **rack failures** (40-80 machines instantly disappear, 1-6 hrs)
  - ~5 **racks go wonky** (40-80 machines see 50% packet loss)
  - ~8 **network maintenances** (~30-minute random connectivity losses)
  - ~12 **router reloads** (takes out DNS and external vIPs for a couple minutes)
  - ~3 **router failures** (traffic rerouting, ~ 1 hour to stabilize.)
  - ~dozens of minor 30-second DNS issues
  - ~1000 **individual machine failures** (2-4% failure rate, at **least twice**)
  - ~thousands of hard drive failures (**1-5%** of all disks will **die**)

Fault Tolerance

- **Other Common Issues**
  - **Configuration issues**, partial SW updates, SW bugs.
  - **Transient errors**: no space left on device, memory corruption, stragglers.
- **Error Rates at Scale**
  - Cost-effective commodity hardware.
  - **Error rate increases** with increasing **scale.**
  - **Fault Tolerance** for distributed/cloud storage and data analysis.

Failures are inevitable. large-scale systems are designed with **fault tolerance** to ensure they can **detect**, **recover** from, and **mitigate** the impact of errors, ensuring **reliability** and **minimal downtime**.

Fault Tolerance

Cost-effective Fault Tolerance (BASE)

- **BA**sically available: system mostly operational even during failures, providing partial functionality if necessary.
- **Soft state**: state of the system may change over time, even without new input, due to replication or recovery processes.
- **Eventual consistency**: data may not be instantly consistent across all nodes but will eventually synchronize to the correct state

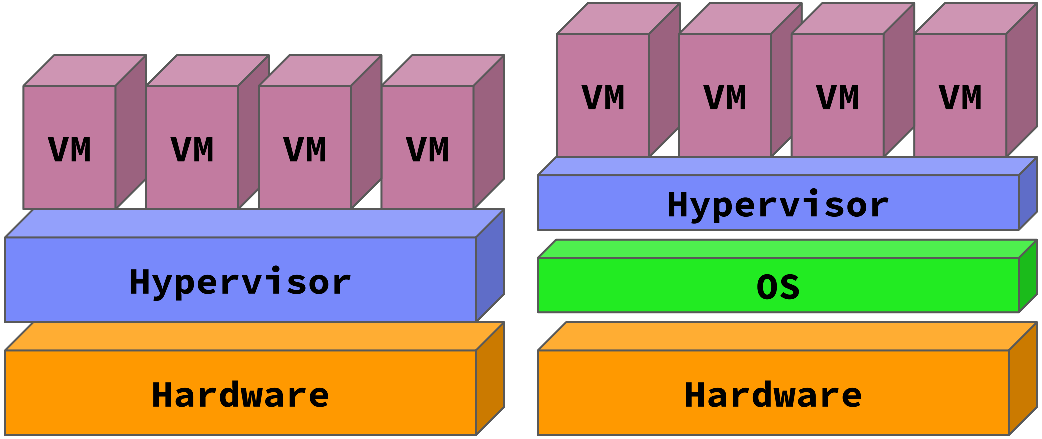
Fault Tolerance

Cost-effective Fault Tolerance (BASE)

- **Data corruption prevention**
  - ECC (error correction codes)
  - CRC (cyclic redundancy check)
- **Resilient storage**
  - Replication (multiple data copies stored across nodes)
  - Checkpointing (save application state)
  - Lineage (data origin + dependencies)
  - Resilient compute: task re-execution / speculative execution

Virtualization I

- **Native Virtualization**
  - Simulates most of the HW interface
  - Unmodified guest OS to run as if it were on real HW
  - Examples: VMWare, Virtualbox, AMI (HVM)
- **Advantages**
  - Guess OS as it is!
  - High compatibility
  - High security



Virtualization II

- **Para-virtualization**
  - No HW interface simulation, but special API (hypercalls) replacing hardware instructions.
  - Requires modified guest OS to use hyper calls, trapped by hypervisor
  - Examples: Xen, KVM, Hyper-V, AMI (PV)
- **Advantages**
  - Faster and more efficient than native virtualization (skips the overhead of hardware simulation).
  - Ideal for environments where performance is critical, and modifying the OS is acceptable.

Virtualization III

- **OS-level Virtualization**
  - Allows multiple isolated environments (virtual servers or containers) to run on the OS.
  - **Guest OS** appears isolated but **same as host OS**
  - Examples: **Solaris/Linux containers, Docker**

- **Application-level Virtualization**
  - App executed within a virtualized runtime environment that abstracts away dependencies on the host system.
  - Examples: Java VM (JVM), Ethereum VM (EVM), Python virtualenv

Virtualization: summary

Topic	Native Virtualization	Para Virtualization	OS-level Virtualization	Application-level Virtualization
Definition	Simulates hardware fully	Uses hypercalls, no HW emulation	Containers sharing host OS	Virtualizes specific applications
Guest OS	Unmodified	Modified	Same as host OS	NA
Performance	Moderate (HW overhead)	High	Very high	High
Examples	VMware, VBox	Xen, KVM	Docker, Linux Containers	JVM, Python virtualenv
Use Case	Mixed OS environments	High-performance VMs	Cloud-native apps	App portability across platforms

Containerization

- **Docker Containers**
  - Shipping container analogy
  - Arbitrary, self-contained goods, standardized units
  - Containers reduced loading times ➔ efficient international trade
  - Self-contained package of necessary SW and data (read-only image)
  - Lightweight virtualization w/ shared OS and resource isolation via control groups

Containerization

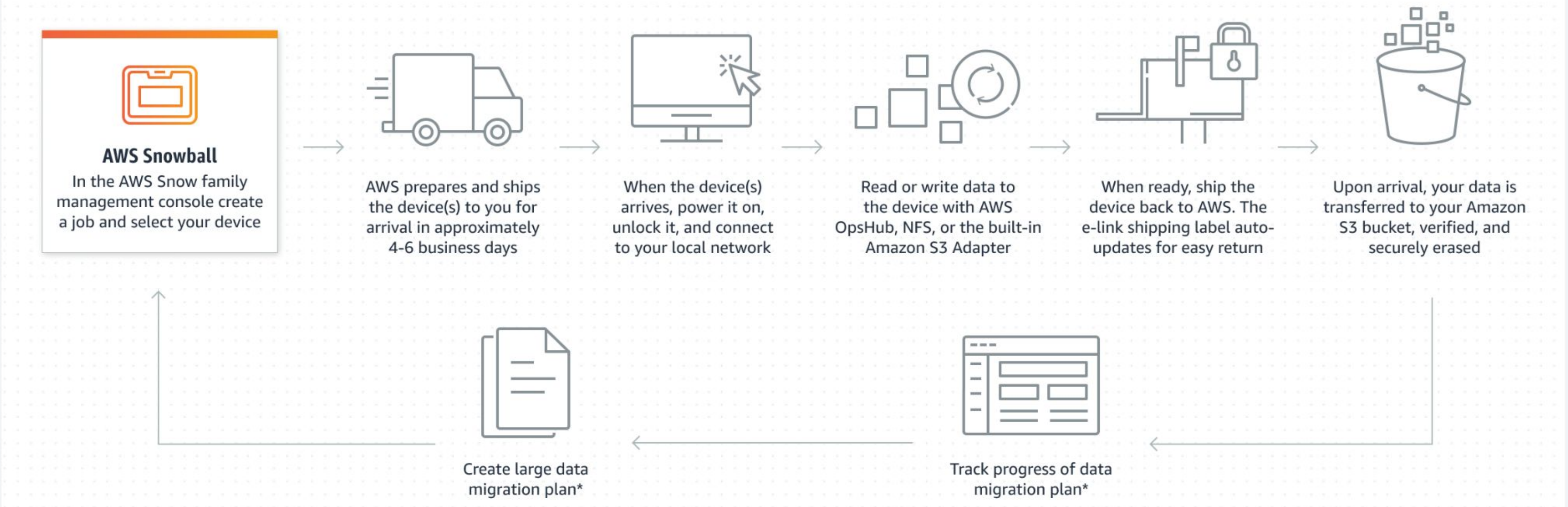
Cluster Schedulers

- Container orchestration: scheduling, deployment, and management
- Resource negotiation with clients
- Typical resource bundles (CPU, memory, device)
- Examples: Kubernetes, Mesos, (YARN), Amazon ECS, Microsoft ACS, Docker Swarm

Cloud: How far can we go?

Snowmobile Service:

- Data transfer on-premise → cloud via 100PB trucks w/ 1Gb link



How far can we go?

Microsoft Underwater Datacenter

How far can we go?

Data centers in the space: why and how?

Space Data Centers – Future Development and Application Perspectives

Tassa Daniel
Cesi Engineering School
Rue Isabelle Autisser
Lagord, France
Email: daniel.tassa@viacesi.fr

Lilie-Leopold-Kateya
Africa Space Innovation Centre
Cape Peninsula Univ of Technology
Cape Town, South Africa
e-mail: leopoldl@cput.ac.za

Oluwaseyi P. Babalola
Africa Space Innovation Centre
Cape Peninsula Univ of Technology
Cape Town, South Africa
e-mail: babalolao@cput.ac.za

Gunjan Gupta
Africa Space Innovation Centre
Cape Peninsula Univ of Technology
Cape Town, South Africa
email: guptag@cput.ac.za

Ayodele A. Periola
Africa Space Innovation Centre
Cape Peninsula University of Technology
email: periolaa@cput.ac.za

Innocent E. Davidson
Africa Space Innovation Centre
Cape Peninsula University of Technology
email: davidsoni@cput.ac.za

Abstract— **The exponential growth of digital data presents escalating challenges for terrestrial data centers, including energy consumption, ecological impact, and cybersecurity risks. Deploying data centers in space emerges as a compelling alternative, leveraging microgravity, extreme temperatures, and isolation to enhance computational efficiency and environmental sustainability. This study explores the technological foundations and feasibility of emerging Space Data Center concepts. It also highlights the environmental burden imposed by terrestrial data centers, particularly their high water and land usage for hosting web services. In response, a space-based architecture is proposed to support low-cost, eco-friendly web development and content delivery. The research underscores space data centers as a forward-looking strategy for sustainable digital infrastructure.**

Keywords – Data centers; satellite; space Integrated; computing networks; photonics.

I. INTRODUCTION

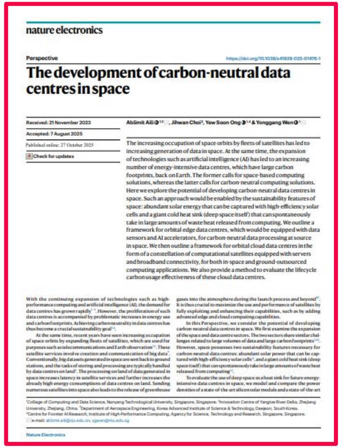
Data centers are vital for hosting digital content accessed over the Internet, supporting key applications such as social media and data storage. Terrestrial data centers consist of multiple servers requiring substantial power and cooling, often leading to high water and land footprints. Water-intensive cooling systems, such as chillers and indirect evaporative cooling, contribute significantly to environmental strain [1], [2], while power demands escalate with data center scale [3], [4]. These constraints limit deployment in water-scarce, landlocked, or densely populated regions, resulting in content latency and power strain, especially in developing areas. Solutions to mitigate these challenges have been proposed in [5], [6]. The discussion in [5], and [6] also shows that future trends in the continued use of terrestrial data centers aim to address challenges in ensuring that data centers deploy their own power sources and systems. However, it is recognized that these approaches do not eliminate the competition for natural resources between terrestrial data center operators and other entities seeking to deploy power systems for other applications.

Terrestrial centers process space-based data [7], but the growing volume from small satellites and the latency in downlinking such data hinder timely decision-making. To address these limitations, there is a need for alternative infrastructure with lower environmental impact and reduced latency, the Space Data Center (SDC). SDCs eliminate dependence on Earth's land and water resources and are positioned to process satellite data in orbit, enabling rapid access and decision-making for latency-sensitive applications. They also host caches to support low-latency content delivery in landlocked regions, offering advantages over terrestrial centers for satellite-based communications.

The concept of SDCs is gaining global attention, with initiatives such as the European Union's ASCEND (Advanced Space Cloud for European Net Zero Emission and Data Sovereignty) project [8], [9] exploring large-scale orbital data centers powered by solar energy to reduce the carbon footprint of information technology. This paper explores the feasibility, technological underpinnings, and environmental implications of deploying data centers in space. It also investigates the role of SDCs in supporting web development tasks, where the reliance on cloud-based tools typically hosted on terrestrial infrastructure incurs significant environmental costs. By shifting such workloads to SDCs, the potential exists to achieve more sustainable computing.

The research being presented focuses on the design and application of SDCs in the area of web development. The motivation for the consideration of the web development application is that web development utilizes a significant proportion of existing computing resources. It is proposed that web development be migrated to SDCs. This has the benefit of reducing the environmental toll due to the use of existing terrestrial data centers for executing web development. The research presents a network architecture i.e., entity identification and describing entity relations. This is done to achieve the goal of executing web development aboard SDCs. In addition, the research recognizes the SDC

 ISBN: 978-1-68558-288-3



nature electronics

Perspective https://doi.org/10.1038/s41928-025-01476-1

The development of carbon-neutral data centres in space

Received: 21 November 2023

Accepted: 7 August 2025

Published online: 27 October 2025

Check for updates

Ablimit Aili, Jihwan Choi, Yew Soon Ong & Yonggang Wen

The increasing occupation of space orbits by fleets of satellites has led to increasing generation of data in space. At the same time, the expansion of technologies such as artificial intelligence (AI) has led to an increasing number of energy-intensive data centres, which have large carbon footprints, back on Earth. The former calls for space-based computing solutions, whereas the latter calls for carbon-neutral computing solutions. Here we explore the potential of developing carbon-neutral data centres in space. Such an approach would be enabled by the sustainability features of space: abundant solar energy that can be captured with high-efficiency solar cells and a giant cold heat sink (deep space itself) that can spontaneously take in large amounts of waste heat released from computing. We outline a framework for orbital edge data centres, which would be equipped with data sensors and AI accelerators, for carbon-neutral data processing at source in space. We then outline a framework for orbital cloud data centres in the form of a constellation of computational satellites equipped with servers and broadband connectivity, for both in-space and ground-outsourced computing applications. We also provide a method to evaluate the lifecycle carbon usage effectiveness of these cloud data centres.

With the continuing expansion of technologies such as high-performance computing and artificial intelligence (AI), the demand for data centres has grown rapidly. However, the proliferation of such data centres is accompanied by problematic increases in energy use and carbon footprints. Achieving carbon neutrality in data centres has thus become a crucial sustainability goal.

At the same time, recent years have seen increasing occupation of space orbits by expanding fleets of satellites, which are used for purposes such as telecommunications and Earth observation. These satellite services involve creation and communication of big data. Conventionally, big datasets generated in space are sent back to ground stations, and the tasks of storing and processing are typically handled by data centres on land. The processing on land of data generated in space increases latency in satellite services and further increases the already high energy consumptions of data centres on land. Sending numerous satellites into space also leads to the release of greenhouse gases into the atmosphere during the launch process and beyond. It is thus crucial to maximize the use and performance of satellites by fully exploiting and enhancing their capabilities, such as by adding advanced edge and cloud computing capabilities.

In this Perspective, we consider the potential of developing carbon-neutral data centres in space. We first examine the expansion of the space and data centre sectors. The two sectors share similar challenges related to large volumes of data and large carbon footprints. However, space possesses two sustainability features necessary for carbon-neutral data centres: abundant solar power that can be captured with high-efficiency solar cells, and a giant cold heat sink (deep space itself) that can spontaneously take in large amounts of waste heat released from computing.

To evaluate the use of deep space as a heat sink for future energy-intensive data centres in space, we model and compare the power densities of a state-of-the-art silicon solar module and a state-of-the-art

[1]College of Computing and Data Science, Nanyang Technological University, Singapore, Singapore. [2]Innovation Centre of Yangtze River Delta, Zhejiang University, Zhejiang, China. [3]Department of Aerospace Engineering, Korea Advanced Institute of Science & Technology, Daejeon, South Korea. [4]Centre for Frontier AI Research, Institute of High-Performance Computing, Agency for Science, Technology and Research, Singapore, Singapore. e-mail: ablimit.aili@zju.edu.cn; ygwen@ntu.edu.sg

HCC
Institute of
HUMAN-CENTRED COMPUTING

Cloud vs other HPC Technologies

Computing Techniques	Features
Cloud Computing	Cost efficient, almost unlimited storage, backup and recovery, easy deployment
Grid Computing	Efficient use of idle resources, parallelism can be achieved, handles complexity
Cluster Computing	Reduced cost, processing power, improved network technology, scalability, availability

Manvi, S. S., & Shyam, G. K. (2014). Resource management for Infrastructure as a Service (IaaS) in cloud computing: A survey. Journal of network and computer applications, 41, 424-440.

Cloud Computing Service Models

SaaS
Docs, Email, Games, etc.

PaaS
APIs, SDKs, and more

IaaS
VMs, Storage, Network

IaaS

IaaS
VMs, Storage, Network

Combination of hosting, hardware provisioning, basic services, and other services needed to run a cloud.

Uses of IaaS:

- Provides access to shared resources on need basis, without revealing details like location and hardware to clients.
- Provides details like server images on demand, storage.
- Offers full control of server infrastructure, not limited specifically to applications, instances and containers.

IaaS

Major issues associated with IaaS:

- Resource management.
- Internet access.
- Virtualization.
- Data management.
- APIs.
- Interoperability.

IaaS
VMs, Storage, Network

PaaS

Provision of a computing platform and the provision and deployment of the associated set of software applications (called a solution stack).

Uses of PaaS:

- Provides tools, frameworks, and runtime environments to build and deploy applications efficiently.
- Removes the need to worry about hardware, operating systems, or server management.
- While IaaS provides control over server infrastructure, PaaS limits control to what's essential for applications, offering an environment tuned specifically for application development and management.

PaaS
APIs, SDKs, and more

Red Hat OpenShift

Google App Engine

Elastic Beanstalk

HCC
Institute of HUMAN-CENTRED COMPUTING

PaaS

Major issues associated with PaaS:

- Provider reliability
- Resource management
- Internet dependency
- Performance
- Scalability costs

PaaS
APIs, SDKs, and more

SaaS

Software as a Service is a software distribution model in which applications are hosted by a vendor or service provider and made available to customers over a network.

Uses of SaaS:

- On-demand software access.
- Scalability
- Cost-effective solution
- Cross device availability
- Collaboration and real-time updates

SaaS
Docs, Email, Games, etc.

databricks
Google Workspace
Canva

SaaS

Major issues associated with SaaS:

- Internet dependency
- Performance
- Long-term cost
- Data Security and Privacy

SaaS
Docs, Email, Games, etc.

databricks

Google Workspace

Canva

Serverless Computing

A cloud computing paradigm where developers can build and run applications without managing server infrastructure. Operational responsibilities like fault tolerance, scaling, and resource allocation are handled by cloud providers.

Uses of SaaS:

- **Function-as-a-Service (FaaS)**: Runs stateless, event-driven functions in response to triggers.
- **Serverless Databases:** Auto-scale capacity as needed and hibernate during periods of inactivity, reducing costs while maintaining availability.

Kounev, S., Herbst, N., Abad, C. L., Iosup, A., Foster, I., Shenoy, P., ... & Chien, A. A. (2023). Serverless computing: What it is, and what it is not?. Communications of the ACM, 66(9), 80-92.


research

Serverless Computing: What It Is, and What It Is Not?

Other “layers”: Serverless

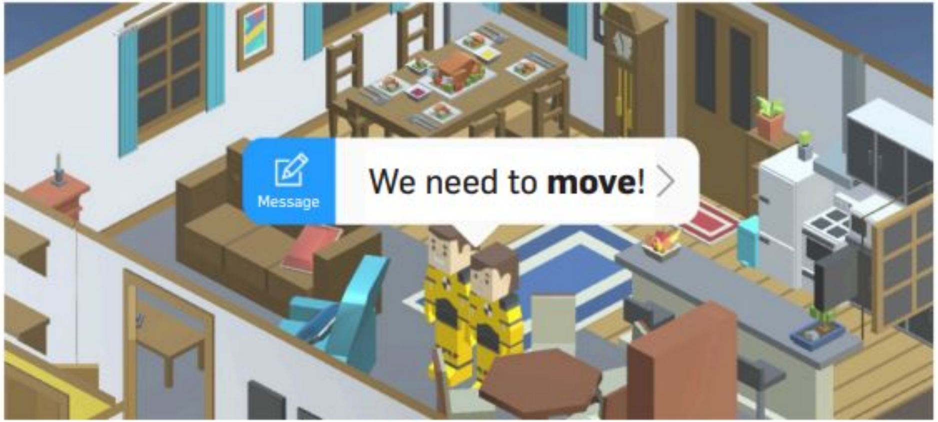
Serverless

IaaS/PaaS/SaaS

Self-hosting


research

Serverless Computing: What It Is, and What It Is Not?



	Packaging	Delivery	Operations	Legal	Financial	Personnel
Modern movers	All objects	Any route	All decisions	All covered	Fine-grained Utilization-based	Small team
Traditional movers	Limited support	Major roads	Basic	Basic	Coarse-grained	Large team
Moving it yourself (with family and friends)	Yourself	Yourself	Yourself	Yourself	Yourself	Yourself

Institute of HUMAN-CENTRED COMPUTING

GPU, Fog, and Edge Computing

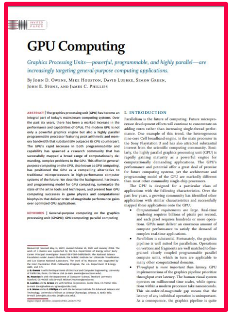
HCC
Institute of
HUMAN-CENTRED COMPUTING

GPU Computing

GPU processor used as a **highly parallel programmable processor** to handle computationally intensive tasks.

Uses:

- **Machine Learning and AI:**
  - **Training and inference** for deep learning models.
  - Accelerating **neural networks** computations.
- **Scientific Applications:**
  - Protein folding simulations.
  - Computational biophysics and molecular dynamics.
  - Fluid dynamics and heat transfer.


INVITED PAPER

GPU Computing

Graphics Processing Units—powerful, programmable, and highly parallel—are increasingly targeting general-purpose computing applications.

By John D. Owens, Mike Houston, David Luebke, Simon Green, John E. Stone, and James C. Phillips

Owens, J. D., Houston, M., Luebke, D., Green, S., Stone, J. E., & Phillips, J. C. (2008). GPU computing. Proceedings of the IEEE, 96(5), 879-899.

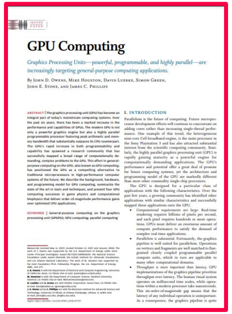
GPU Computing

Uses:

- **Real-Time Graphics:**
  - Game physics and rendering.
  - Simulation of physical environments in games.

- **Data Processing:**
  - Sorting and searching large datasets.

- **General Numerical Computations:**
  - Differential equation solvers.


GPU Computing

Graphics Processing Units—powerful, programmable, and highly parallel—are increasingly targeting general-purpose computing applications.

By John D. Owens, Mike Houston, David Luebke, Simon Green, John E. Stone, and James C. Phillips

Owens, J. D., Houston, M., Luebke, D., Green, S., Stone, J. E., & Phillips, J. C. (2008). GPU computing. Proceedings of the IEEE, 96(5), 879-899.

Providers (AWS, NVIDIA, Microsoft, Google, IBM, Oracle, Huawei, Tencent)

AWS:

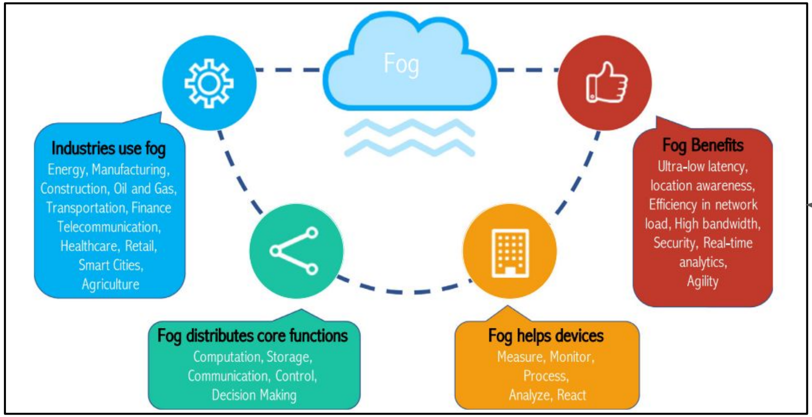
- EC2 with GPU Instances (NVIDIA Tesla V100, T4, K80)
- Deep Learning AMIs (Amazon Machine Images).
- AWS Lambda for GPU-powered serverless workflows.

NVIDIA Cloud

- NVIDIA GPU Cloud (Focused on NVIDIA GPUs, including A100 and V100)
- Optimized software stacks for AI, HPC, and data analytics.
- Pre-built containers for machine learning frameworks.

Fog Computing

- Enables computing, storage, networking, and data management on network nodes **within the close vicinity** of IoT devices.
- Computation, storage, networking, decision making, and data management not only occur in the cloud → also **along the IoT-to-Cloud path** (preferably close to the IoT devices).

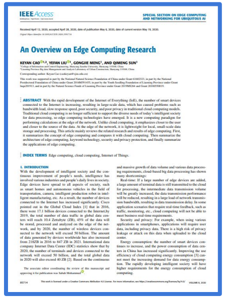


Yousefpour, A., Fung, C., Nguyen, T., Kadiyala, K., Jalali, F., Niakanlahiji, A., ... & Jue, J. P. (2019). All one needs to know about fog computing and related edge computing paradigms: A complete survey. Journal of Systems Architecture, 98, 289-330.

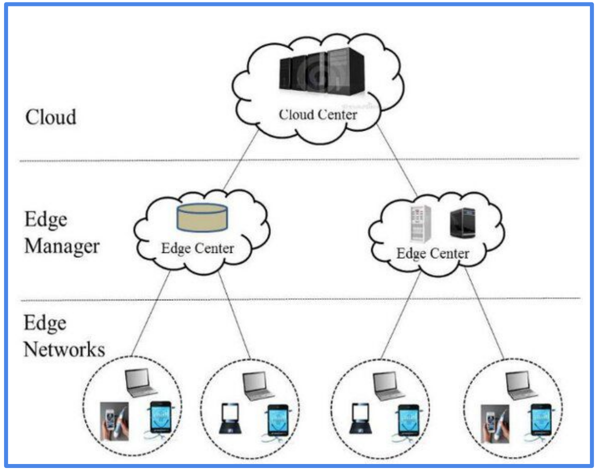
Edge Computing

- Enhances →management, storage, and processing for data generated by connected devices.
- Edge computing operates at the edge of the network, positioned near IoT devices (one hop away.)
- The **edge is not directly on the IoT devices!**

Cao, K., Liu, Y., Meng, G., & Sun, Q. (2020). An overview on edge computing research. *IEEE access*, *8*, 85714-85728.


IEEE Access

An Overview on Edge Computing Research



Ren, Y., Zhu, F., Qi, J., Wang, J., & Sangaiah, A. K. (2019). Identity management and access control based on blockchain under edge computing for the industrial internet of things. Applied Sciences, 9(10), 2058.

HCC Institute of HUMAN-CENTRED COMPUTING

A short view about a public cloud…

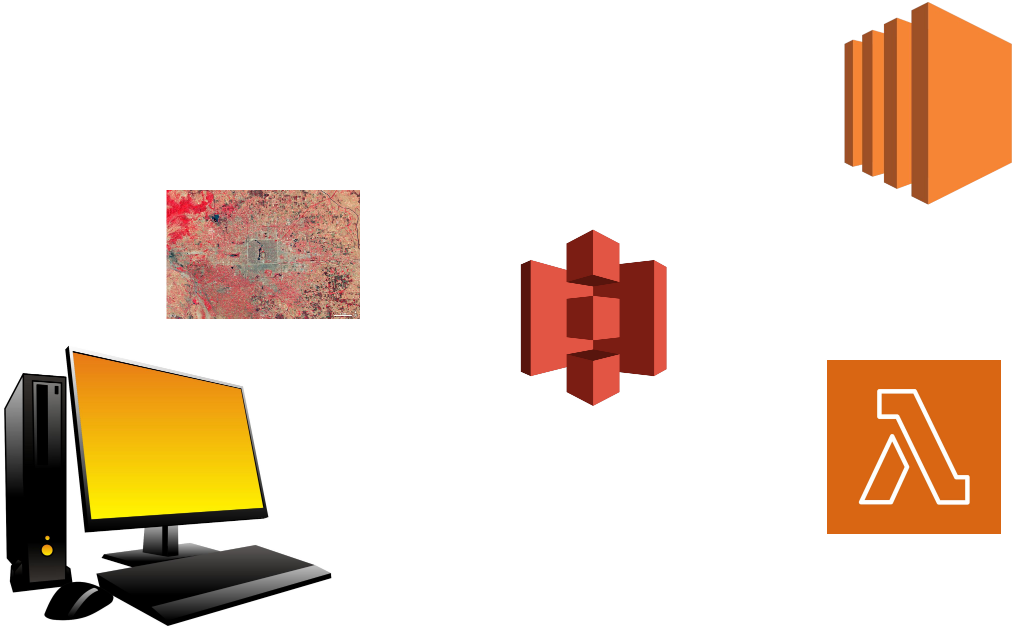
HCC
Institute of
HUMAN-CENTRED COMPUTING

Connecting…

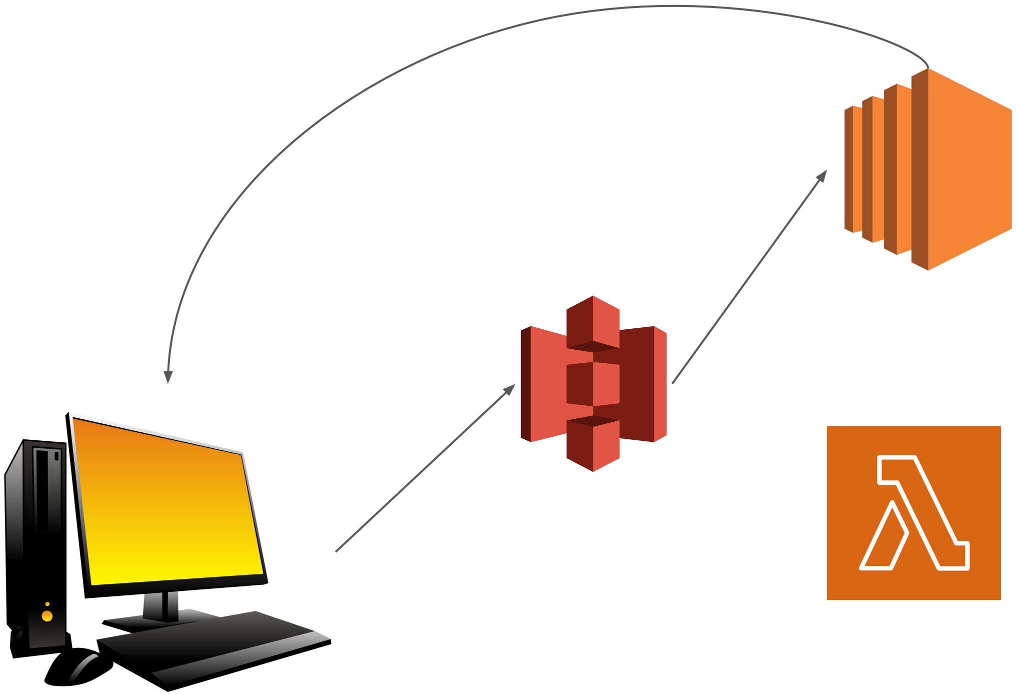
```
https://us-west-2.console.aws.amazon.com/

ssh -i <my-key> ubuntu@<my-instance-ip-public-address>
```

Cloud pipeline

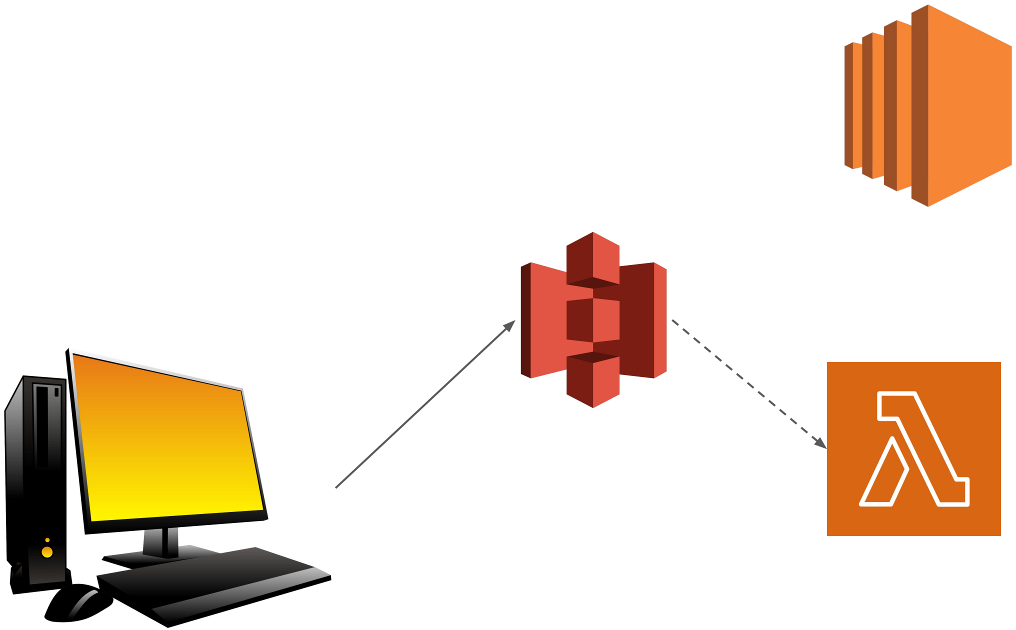


Cloud pipeline

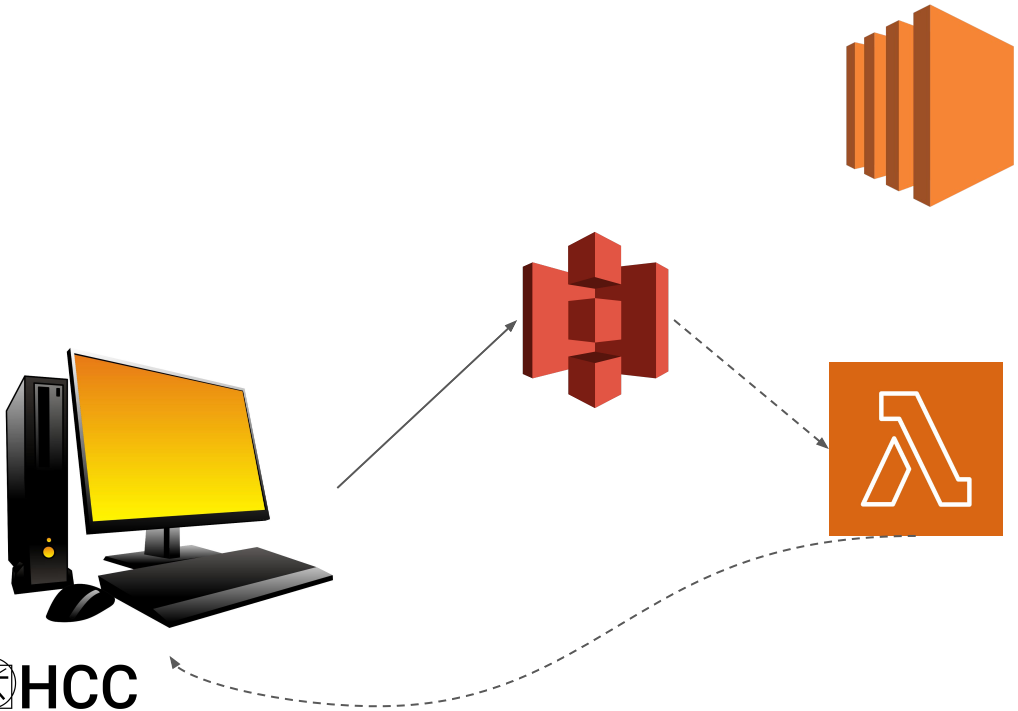


HCC
Institute of
HUMAN-CENTRED COMPUTING

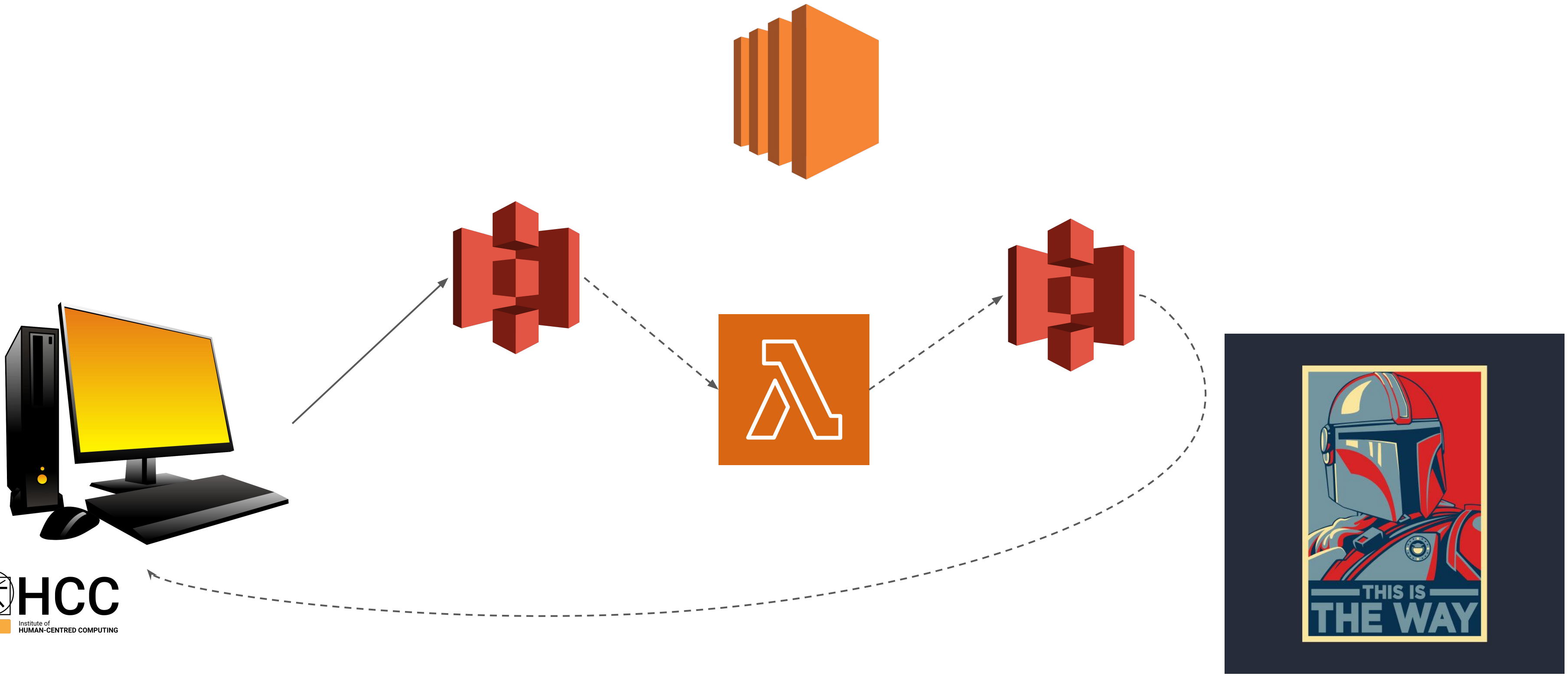
Cloud pipeline



Cloud pipeline



Cloud pipeline



HCC
Institute of
HUMAN-CENTRED COMPUTING

Summary and Q&A

Summary and Q&A

- **Summary and Q&A**
  - Cloud Computing Motivation and Terminology
  - Cloud Computing Service Models
  - Cloud, Fog, and Edge Computing
- **Next Lectures**
  - 08 Cloud Resource Management and Scheduling [Nov 28]