Infrastructures

Martina Castellucci

Preface: Glossary of Terms

- **Big Data:** Massive and complex data that requires advanced infrastructures for processing.
- **CPU (Central Processing Unit):** The main unit that executes instructions and operations.
- **GPU (Graphics Processing Unit):** Processor for parallel operations, useful in ML and simulations.
- **Cache:** Fast memory inside the CPU, divided into L1, L2, and L3 levels.
- **RAM (Random Access Memory):** Volatile, fast-access memory.
- **Storage (SSD/HDD):** Long-term storage devices, SSD is faster, HDD is larger but slower.
- **Latency:** Delay in accessing data.
- **Bandwidth:** Rate of data transfer.
- **Hyper-Threading:** Technology to execute multiple threads per core.
- **HTC (High-Throughput Computing):** Distributed computing model that maximizes job throughput.
- **HPC (High-Performance Computing):** Parallel computing model to minimize job execution time.
- **Grid Computing:** Distributed infrastructure based on cooperation among organizations.
- **Cloud Computing:** Centralized infrastructure managed by a provider.

- **WMS (Workload Management System):** System to manage job distribution across resources.
- **Pilot Job:** Process that verifies resource availability before job execution.
- **Amdahl's Law:** Formula to estimate the maximum theoretical speedup of parallel jobs.
- **Tiered Storage:** Hierarchical storage (fast to slow) for optimized performance and costs.
- **Batch Processing:** Processing large datasets in blocks.
- **Stream Processing:** Real-time processing of data streams.
- **Data Provenance:** Tracing the origins and lifecycle of data.
- **Parallel I/O:** Technique to improve data transfer rates by distributing file access.
- **ACL (Access Control List):** Security list that defines access rights.
- **RBAC (Role-Based Access Control):** Access management based on user roles.

1 Introduction

This document explores the concepts of Big Data, computing infrastructures, and hardware/software architectures used to handle complex and massive data. It analyzes CPU, GPU, memory, and storage technologies, CPU-driven and data-driven computing models, job distribution strategies, and the differences between HPC and HTC. It also expands on containerization, file systems, cloud computing models, and performance benchmarking.

2 Big Data: Definition and Characteristics

Big Data refers to data so massive and complex that traditional approaches cannot handle them efficiently. It includes structured, semi-structured, and unstructured data. The 5 V's of Big Data are: **Volume, Velocity, Variety, Variability, and Veracity**.

3 Digital Data Fundamentals

Data is stored in bits and bytes. A bit represents a 0 or 1, while a byte (8 bits) often represents a character. All digital information, from text to video, is stored as sequences of bits and bytes.

4 Hardware Components

The CPU executes instructions and consists of the Control Unit, ALU, and internal registers. The GPU accelerates parallel tasks and is key for ML and simulations. Modern CPUs are 32 or 64 bits. Cache (L1, L2, L3) speeds up data access compared to RAM. SSDs are faster than HDDs. The motherboard connects components, supporting multi-socket CPUs. Hyper-Threading allows multiple threads per core.

5 Latency and Bandwidth

Latency measures the time to access data; bandwidth measures the transfer rate.

6 Data Management Strategies

Policy-driven data management applies rules for data access, sharing, and protection. An embargo period restricts data during initial analysis for reproducibility. Tiered storage (SSD, HDD, tape) helps manage performance and costs. Quality of Service (QoS) ensures appropriate allocation of resources.

7 Data Movement and Copying

Data is moved using tools like scp, rsync, and FTP/SFTP. Distributed file systems (NFS, Lustre, GPFS) improve scalability and performance. Parallel I/O and striping enhance data throughput.

8 Batch vs. Stream Processing

Batch processing handles large data sets offline, while stream processing handles data in real-time.

9 Computing Models

CPU-driven models schedule jobs on available processors; data-driven models process jobs where the data resides. Hybrid models balance both strategies.

10 Job Scheduling and WMS

Jobs are distributed using push or pull models. Pilot jobs verify resource availability. Workload Management Systems (WMS) like SLURM and HTCondor assign jobs and manage queues.

11 Data Provenance and Security

Data provenance tracks the origin and transformations of data. Security mechanisms include ACLs and RBAC.

12 HTC and HPC

HTC maximizes throughput with independent tasks. HPC uses parallelization for single tasks requiring high performance. Performance is measured in FLOPS (GFLOPS, TFLOPS).

13 Cloud Computing Models

Cloud services include IaaS, PaaS, and SaaS. Each offers different levels of abstraction and control.

14 Distributed Infrastructures

Distributed infrastructures connect multiple data centers, requiring certificates and single sign-on for security.

15 Networking and Data Centers

Protocols like HTTP and Wi-Fi enable communication. Network devices include hubs, switches, and routers. Topologies like fat-tree and torus are used in data centers. Top-of-the-Rack switches reduce latency.

16 Benchmarking and Performance

Benchmarking tools like Linpack, IOR, and FIO test system performance. Amdahl's Law estimates maximum speedup:

$$\text{Speedup}_{\max} = \frac{1}{\alpha}$$

where α is the sequential portion. Efficiency compares speedup to processor count.

17 Containerization and Docker Ecosystem

Containerization improves portability and scalability. Docker uses Dockerfiles to build containers:

```
FROM ubuntu:20.04
RUN apt-get update && apt-get install -y python3
COPY myscript.py /app/
CMD ["python3", "/app/myscript.py"]
```

Docker Compose orchestrates multi-container apps using YAML. Volumes persist data. Best practices include using lightweight images, single processes per container, and tagging versions.

17.1 Docker Security

Avoid running containers as root. Use official images and vulnerability scanning.

17.2 Udocker and Singularity

Udocker runs containers without root, useful in HPC. Singularity supports MPI, GPU, and batch systems securely.

17.3 Containers vs Virtual Machines

Containers share the host kernel, unlike VMs that emulate entire OS stacks, making them faster and more efficient.

17.4 Networking and Image Transfer

Bind mounts and volumes enable data sharing. Docker registries store and distribute images.

18 Use Cases

BLAST and BWA (HTC) for bioinformatics. Simulations (HPC) for weather and molecular dynamics.

19 Conclusions

Integrating hardware, software, and data management strategies is essential to handle Big Data and modern computational challenges.

Pipeline Overview

Data Generation → Data Storage → Data Preprocessing → Job Scheduling → Job Execution → Result Aggregation → Visualization and Analysis

20 Comparison Tables

CPU vs GPU

Table 1: CPU vs GPU Comparison

Feature	**CPU**	**GPU**
Purpose	General-purpose tasks	Parallel computing and graphics
Cores	Few (4-16)	Thousands of lightweight cores
Clock Speed	Higher	Lower
Best For	Sequential tasks	Massively parallel tasks (e.g. ML)
Memory Access	Complex caching hierarchy	Shared global memory

Containers vs Virtual Machines

Table 2: Containers vs Virtual Machines Comparison

Feature	**Containers**	**Virtual Machines**
Kernel	Shared with host	Separate guest OS kernel
Performance	Lightweight	Heavyweight
Startup Time	Seconds	Minutes
Resource Usage	Low	High
Use Cases	Microservices, CI/CD	Legacy apps, full OS emulation

Batch Processing vs Stream Processing

Table 3: Batch vs Stream Processing Comparison

Feature	**Batch Processing**	**Stream Processing**
Data Ingestion	Processed in batches	Continuous data flow
Latency	Higher	Lower
Use Cases	ETL, Data warehousing	Real-time monitoring, analytics
Complexity	Simpler	More complex

HTC vs HPC

Table 4: HTC vs HPC Comparison

Feature	**HTC**	**HPC**
Job Type	Many independent tasks	Single large parallel task
Network	Commodity Ethernet	High-speed interconnects
Data Access	Distributed	Shared high-performance storage
Use Cases	Bioinformatics, parametric studies	Simulations, ML training

Docker vs Singularity vs Udocker

Grid vs Cloud Computing

Table 5: Docker vs Singularity vs Udocker Comparison

Feature	**Docker**	**Singularity**	**Udocker**
Root Access	Requires daemon	No root required	No root required
HPC Ready	Limited support	Native support	HPC-friendly
Security	Requires configuration	HPC integration	User-space execution
Image Format	Docker images	Singularity/SIF	Docker images

Table 6: Grid vs Cloud Computing Comparison

Feature	**Grid Computing**	**Cloud Computing**
Ownership	Federated institutions	Centralized provider
Control	Local administrators	Provider-managed
Scalability	Limited agreements	Elastic, on-demand
Billing	Shared or free	Pay-as-you-go
Use Cases	Research collaborations	Commercial workloads

Simulated Final Exam (31 Multiple Choice Questions)

1. Which of the following is NOT one of the 5 V's of Big Data?
   (a) Volume
   (b) Velocity
   (c) Variety
   (d) Visibility
2. A CPU typically has:
   (a) Thousands of lightweight cores
   (b) One or more heavy-weight cores
   (c) Shared global memory with GPUs
   (d) A separate OS kernel
3. What is the main advantage of using containerization technologies like Docker?

(a) Complete hardware emulation
(b) Lightweight and portable applications
(c) Faster disk speeds
(d) Dedicated GPU acceleration

4. The fastest memory in the CPU is:

(a) RAM
(b) HDD
(c) L1 Cache
(d) SSD

5. Docker Compose is used for:

(a) Writing Python scripts
(b) Managing container networks
(c) Orchestrating multi-container applications
(d) Managing database backups

6. Amdahl's Law calculates:

(a) The maximum theoretical speedup
(b) The size of a Docker image
(c) The storage tiering strategy
(d) The CPU cache latency

7. A workload management system (WMS) is responsible for:

(a) Managing data encryption
(b) Assigning jobs to resources
(c) Backing up user files
(d) Training machine learning models

8. Which file system is commonly used in distributed computing?

(a) NTFS
(b) GPFS
(c) exFAT

(d) FAT32

9. Hyper-Threading allows:

   (a) Multiple processes to run on one core
   (b) One thread per core
   (c) Shared GPU memory usage
   (d) Containerized execution

10. Which is an example of batch processing?

    (a) Real-time analytics
    (b) Data streaming
    (c) Daily sales report generation
    (d) Continuous user input

11. Singularity is especially suited for:

    (a) Mobile applications
    (b) HPC environments
    (c) Data visualization
    (d) Cloud storage

12. Which tool copies files securely between servers?

    (a) FTP
    (b) rsync
    (c) scp
    (d) Both b and c

13. Which of the following represents volatile memory?

    (a) SSD
    (b) HDD
    (c) RAM
    (d) Tape

14. Which technology is used to create container images?

    (a) Dockerfile

(b) YAML

(c) Makefile

(d) JSON

15. Which best describes tiered storage?

(a) Using RAID disks

(b) Hierarchical storage (e.g. SSD, HDD, Tape)

(c) Replicating data across containers

(d) Encrypting data during transit

16. The 3 main cloud service models are:

(a) SaaS, IaaS, PaaS

(b) XML, JSON, YAML

(c) NAS, SAN, DAS

(d) CPU, GPU, TPU

17. The push model in WMS means:

(a) Jobs wait for resources to pull them

(b) Jobs are actively sent to resources

(c) Jobs run only on GPUs

(d) Jobs can only be executed manually

18. Which system helps in distributing containers without root access?

(a) Docker

(b) Kubernetes

(c) Udocker

(d) Windows Subsystem for Linux

19. Which of the following is a high-speed interconnect technology?

(a) USB

(b) Ethernet

(c) InfiniBand

(d) Wi-Fi

20. Which model processes data in real time?

    (a) Batch processing
    (b) Stream processing
    (c) RAID 5
    (d) Grid computing

21. The most appropriate tool for HPC containerization is:

    (a) Docker
    (b) Kubernetes
    (c) Singularity
    (d) Ansible

22. Which of the following is NOT a benefit of containerization?

    (a) Portability
    (b) Lightweight
    (c) Full hardware emulation
    (d) Scalability

23. Data provenance helps with:

    (a) Visualizing GPU performance
    (b) Tracing data origins
    (c) Encrypting data files
    (d) Deleting large datasets

24. Which is an example of role-based access control?

    (a) NTFS permissions
    (b) ACLs
    (c) RBAC
    (d) RAID configurations

25. What is the main difference between a container and a VM?

    (a) VMs are faster
    (b) Containers share the host kernel

(c) VMs require Kubernetes

(d) Containers cannot run on Linux

26. Which HPC architecture supports GPU acceleration?

(a) Single-core CPUs

(b) NUMA only

(c) Hybrid CPU-GPU architectures

(d) Tape storage arrays

27. In distributed infrastructures, single sign-on ensures:

(a) Data replication

(b) Easier user authentication

(c) Faster disk access

(d) Container orchestration

28. Which file system is commonly used for parallel I/O?

(a) NFS

(b) GPFS

(c) NTFS

(d) FAT32

29. Data encryption during transit ensures:

(a) Faster computation

(b) Data privacy and integrity

(c) GPU optimization

(d) Parallel I/O

30. Which concept calculates the maximum speedup in parallel computing?

(a) Moore's Law

(b) Amdahl's Law

(c) Murphy's Law

(d) Docker Compose

31. Which cloud model provides the highest control over resources?

(a) SaaS

(b) IaaS

(c) PaaS

(d) FaaS