# Focus on computing techniques

sara.vallero@to.infn.it

Smart workflows

Case studies

What about distributed computing?

Infrastructure explained

# What about distributed computing?

*I have a difficult problem that I cannot compute on my laptop…*

- Maybe the problem is too complex
- Or maybe there is too much data to be processed

**Strong scaling**

**Weak scaling**

**High performance or high throughput?**

**How to distribute data?**

**How to distribute the calculation?**

# How to distribute the calculation?

Split the task in many smaller pieces

**Embarrassingly parallel**

Sub-tasks are independent

**Challenging**

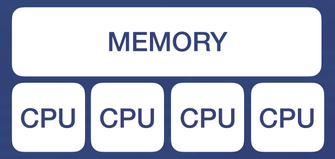
Sub-tasks need to communicate and synchronize

**Choose the right software stack**

**Choose the right infrastructure**

# Choose the right infrastructure

## MULTICORE




**Shared memory:**

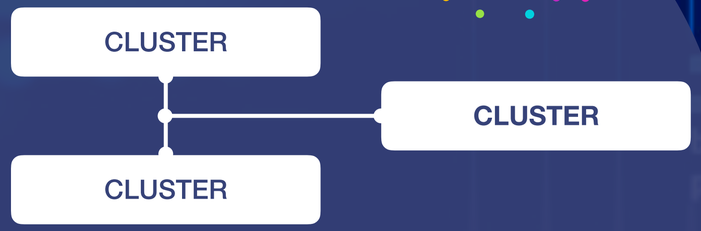
- Single address space
- All processes have access to the pool of shared memory

## CLUSTER


**Distributed memory:**

- Each processor has its own local memory
- Message passing (mainly) is used to exchange data between processors

## GRID




Several clusters spanning different Administrative Domains

## HPC cluster

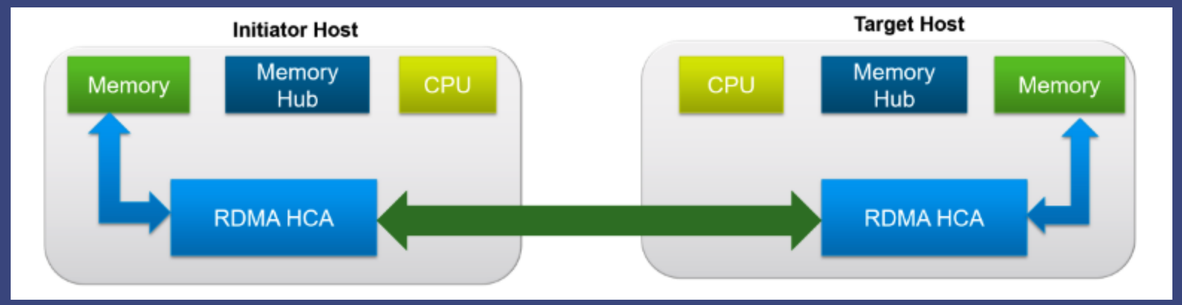
# HPC cluster

# HPC cluster

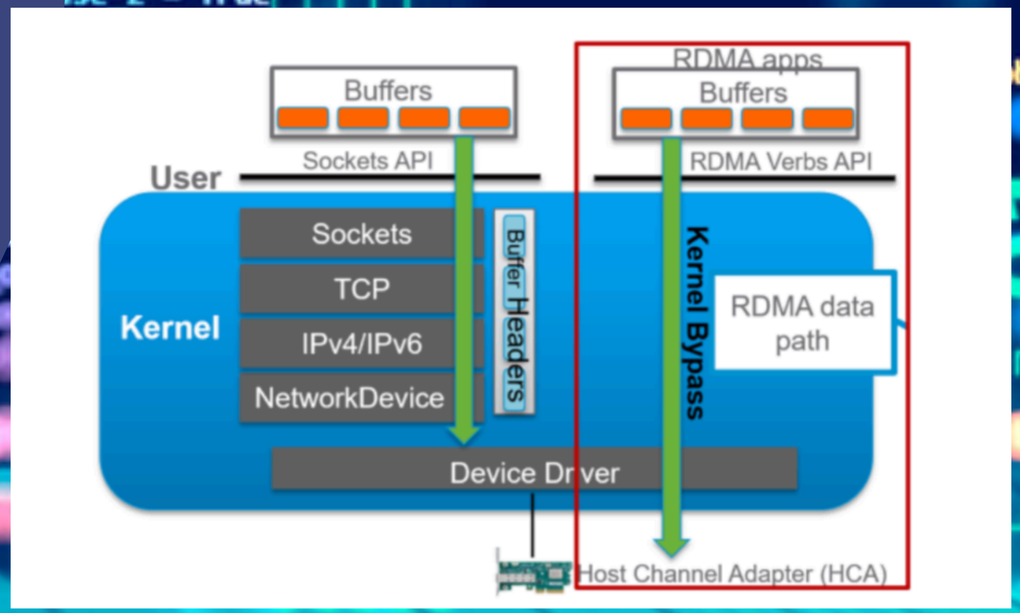
**Low latency interconnect:**
i.e. Mellanox's *InfiniBand* or Intel's *OmniPath*

Offer an RDMA implementation

**Remote Direct Memory Access (RDMA)**




- Transfer of memory between different computers
- Direct transfer minimizing CPU/Kernel involvement
- Bypassing the Kernel allows high I/O bandwidth and low latency
- Host Channel Adapter (HCA) needed on both source and destination



*I have a difficult problem that I cannot compute on my laptop…*

- Maybe the problem is too complex
- Or maybe there is too much data to be processed

**Strong scaling**

**Weak scaling**

**High performance or high throughput?**

**How to distribute the calculation?**

# High performance or high throughput?

## High-throughput Computing (HTC)

- Efficient execution
- Long execution times
- Large amount of tasks
- Tasks are loosely coupled (sequential)

**Example:**

- a relatively simple code to analyze a huge amount of experimental data
- each task processes a chunk of the data sample independently

## High-performance Computing (HPC)

- Large amount of computing power
- Short execution times
- Tasks are tightly coupled (parallel)
- Low latency interconnect

**Example:**

- simulate the atomic motion of a protein in water
- each atom interacts with the others in the system
- communicate data back and forth each sub-task

*I think we did not mention the Cloud…*

# The Cloud

## Application

Defined by:

- runtime environment
- resource requirements
- execution model

## Virtualization

- Virtual Machines
- Linux containers

**Cloud computing:** a style of computing in which scalable and elastic IT-enabled capabilities are delivered **as a service** using **Internet** technologies.

## Computing model

**BATCH**

multi-node workloads, eventually using MPI and inter-node communication

**PIPELINES**

multi-step data analysis, possibly requiring high-memory and many cores

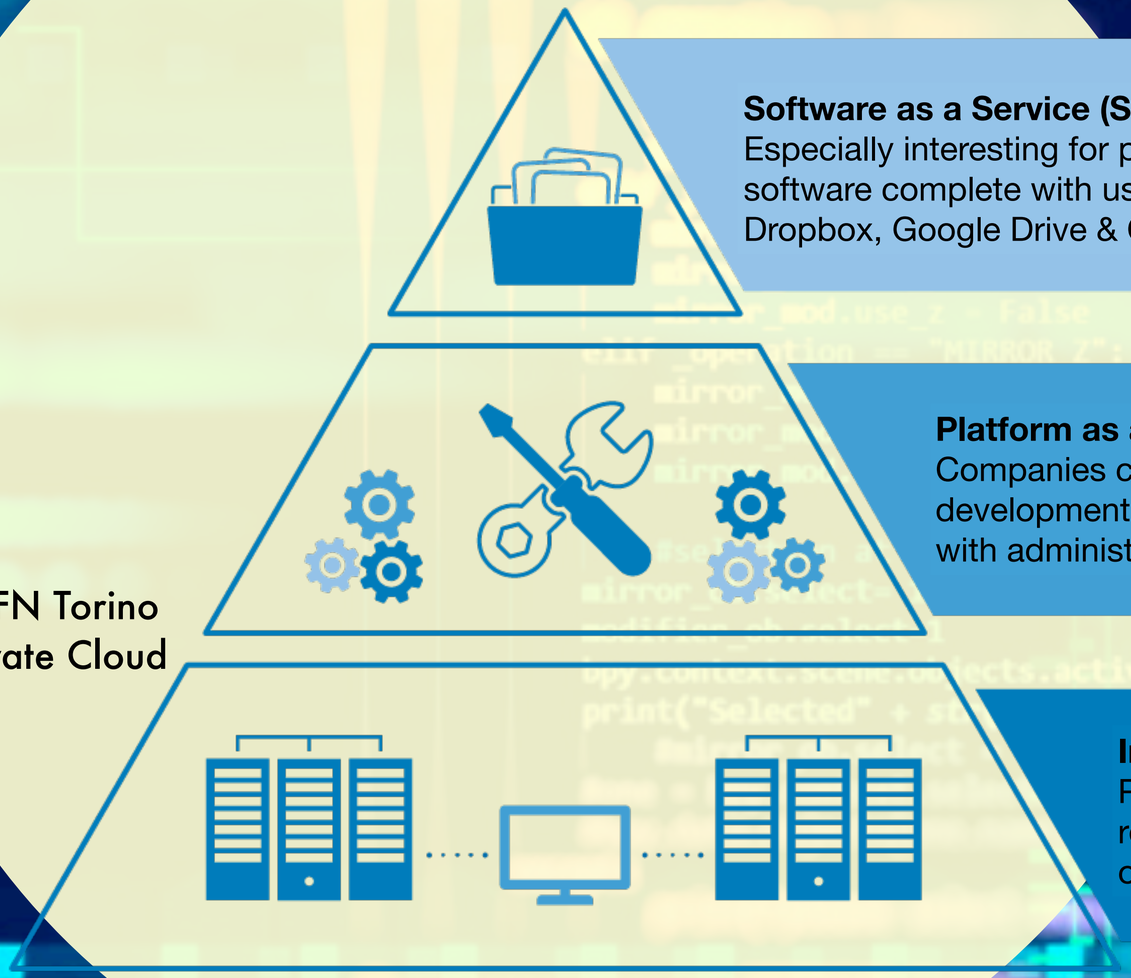
**VIRTUAL WORKSTATION:**

code execution in a single multicore node, possibly with GPU acceleration

## On demand

- Autoscaling

# The Cloud Pyramid



INFN Torino
Private Cloud

**Software as a Service (SaaS)**
Especially interesting for private users is cloud based application software complete with user interface, such as Microsoft Office 365, Dropbox, Google Drive & Co.

**Platform as a Service (PaaS)**
Companies can rent predefined platforms for software development, e.g. Microsoft Azure. The provider deals with administration of the underlying servers.

**Infrastructure as a Service (IaaS)**
Providers like Amazon Web Services (AWS) rent out storage and computing capacities on their servers.

# Lessons to take-home

- Understand your problem's features: is it embarrassingly parallel or not?
- Choose the right type of infrastructure to execute your task:
  - Multi-core
  - HTC cluster
  - Grid
  - HPC cluster
  - Any of those over a Cloud (maybe not HPC)