Program and dates of Management and Analysis of Physics Datasets

13 March (Wed)	HEP computing, Simone Campana 12:30-14-30 (LECTURE) + 14.30-16.30 (LAB)
27 March (We)	Distributed computing and big data, Donatella Lucchesi 12:30- 14:30 Lecture
28 March (Thu)	Distributed computing and big data, Donatella Lucchesi 8:30- 10:30 Lecture
3 April (Wed)	Exercise on distributed cluster, Stefano Campese 12:30-16:30 LAB
17 April (Wed)	Storage, Simone Campana 12:30-14-30 (LECTURE) + 14.30-16.30 (LAB)
18 April (Thu)	Storage, Simone Campana 08:30-10:30 (LECTURE) + 14.30-16.30 (LAB)
2 May (Thu))	Exercises on usage of cluster Stefano Campese + Donatella Lucchesi 14:00 -18:00 LAB
3 May (Fri)	Exercises on data movement and analysis on cluster Stefano Campese +Donatella Lucchesi 14:00- 18:00 Lecture
10 May (Fri)	Machine learning exercise Stefano Campese + Donatella Lucchesi 12:30-16:30 LAB
16 May (Thu)	Exercise using spark exercise Stefano Campese 9:30 -13:30 LAB
22 May (Wed)	Data Management, Simone Campana 12:30-14-30 (LECTURE) + 14.30-16.30 (LAB)
23 May (Thu)	Data Management, Simone Campana 08:30-10:30 (LECTURE) + 14.30-16.30 (LAB)
6 June (Thu)	Data Management analytics Simone Campana 08:30-10:30 (LECTURE) + 14.30-16.30 (LAB)

Distributed Computing and Big Data
or…
Distribute Computing because of Big Data

Distributed is a very large and complex topic.
There books and course on distributed computing, here we want summarize the main facts and give you the main concepts of distributed computing.

These slides material comes from:

- ✓ My experience, I'm one of the author of the Grid computing middleware of CDF, one of the Tevatron collider at Fermilab, Chicago
- ✓ References to articles:
  - https://arxiv.org/ftp/arxiv/papers/0901/0901.0131.pdf
  - https://cloud4scieng.org/
  - WLCG http://wlcg-public.web.cern.ch/

Distributed Computed: Introduction

The idea of distributed computing goes back in 1961, when the computing pioneer John McCarthy predicted that "computation may someday be organized as a public utility"

In the mid 1990s, the term Grid was coined to describe technologies that would allow consumers to obtain computing power on demand. Ian Foster and others posited that by standardizing the protocols used to request computing power, we could spur the creation of a Computing Grid, analogous in form and utility to the electric power grid.

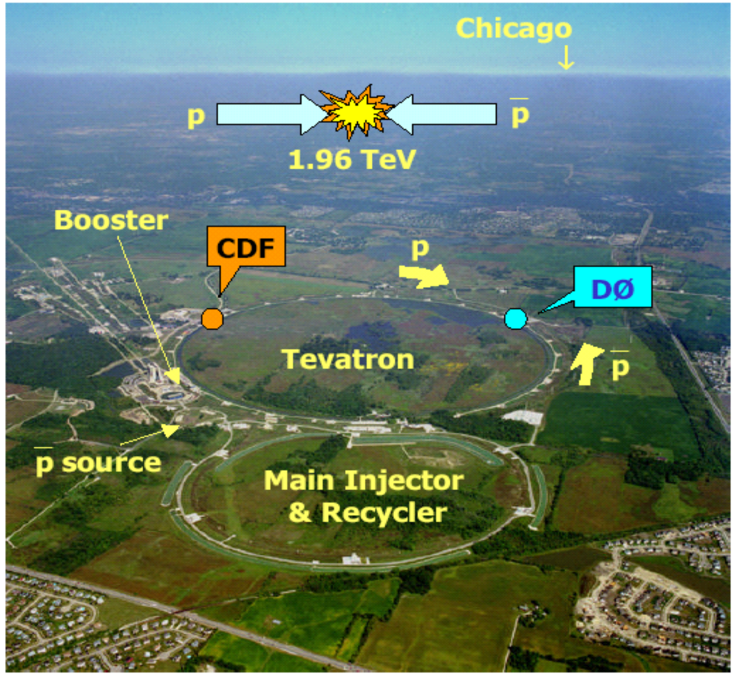
Grid and Cloud computing want to address the problems of managing large facilities which include computational resources and large data sets and give access to them to people around the world.

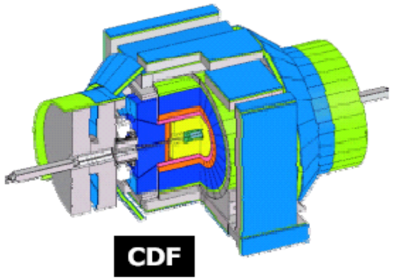
Examples:
Worldwide LHC Computing Grid (WLCG)
European Grid Infrastructure (EGI)

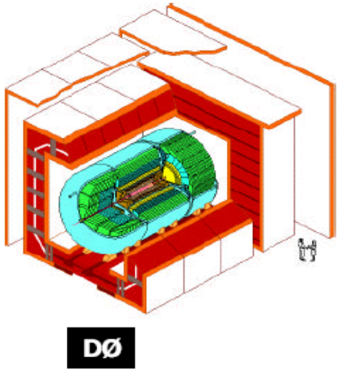
Distributed Computed: Introduction

[Exploration on the Big Data Frontier (TedxCERN)]

How we arrived to the Grid, the Tevatron Experiments example







Grid at the Tevatron Experiments

Several dedicated farms supported by the experiments

Use Grid resources

Use five access points:

NAMCAF	American resources
LcgCAF	European resources
PacCAF	Asian Grid

CDF way to Grid

The Grid

- Grids started off in the mid-90s to address large-scale computation problems using a network of resource-sharing commodity machines that deliver the computation power affordable only by supercomputers and large dedicated clusters at that time.

- The major motivation was that these high performance computing resources were expensive and hard to get access to, so the starting point was to use federated resources that could comprise compute, storage and network resources from multiple geographically distributed institutions, and such resources are generally heterogeneous and dynamic.

- Grids focused on integrating existing resources with their hardware, operating systems, local resource management, and security infrastructure.

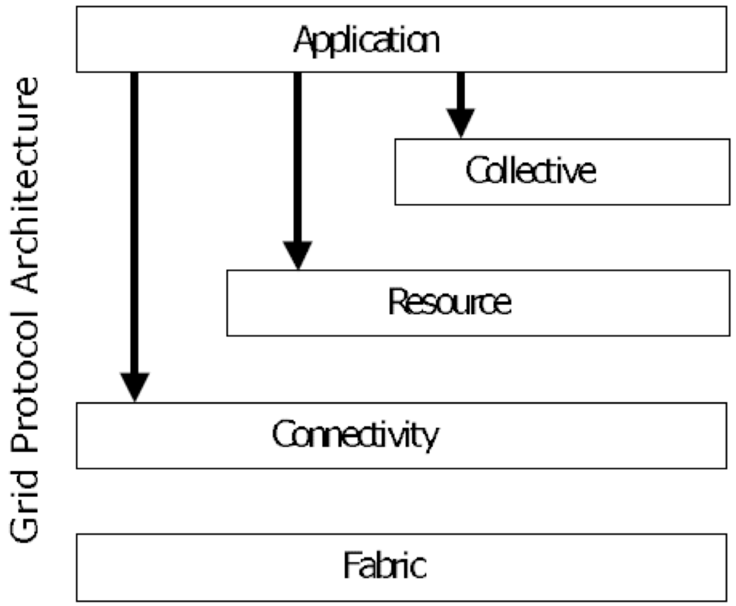
- Grid is built around the Virtual Organization (VO), a logical entity that refers to a dynamic set of individuals or institutions, for which Grids define and provide a set of standard protocols, middleware, toolkits, and services built on top of these protocols. Examples of VO:
    - Each LHC experiment
    - Cosmology theoreticians
    - You name it! Make examples of VO

The Grid

First of all, the system is in continuous evolution → only the basic elements will be presented.

Interoperability and security are the primary concerns for the Grid infrastructure as resources may come from different administrative domains, which have both global and local resource usage policies, different hardware and software configurations and platforms, and vary in availability and capacity.

Grids provide protocols and services at five different layers as identified in the Grid protocol architecture:
Fabric
Connectivity
Resource
Collective
Application



The Grid cont’d

Fabric Layer

Different types of resources: compute, storage and network resource, code repository, etc.
Grids usually rely on existing fabric components, for example local resource managers (Condor, PBS, etc), disk and tape mangers.

Connectivity layer

defines core communication and authentication protocols for easy and secure network transactions. The GSI (Grid Security Infrastructure) protocol underlies every Grid transaction.

Resource layer

defines protocols for the publication, discovery, negotiation, monitoring, accounting and payment of sharing operations on individual resources.

Collective layer

captures interactions across collections of resources, directory services such as monitoring and discovery service allows for the monitoring and discovery of VO resources. Typical services include co-allocating, scheduling and brokering

The Grid cont'd

Application layer

comprises whatever user applications built on top of the above protocols and APIs and operate in VO environments. In the years several application have been developed.

Examples? Name it

Components of Grid Environment

- The Grid components are related to atomic tasks for building a Grid software, referred as middleware: Resources Management, Data model, Security and Monitoring.
- Access to the Grid is given via VO.

Computing Resources Management

The scheduler is responsible for resource discovery, trading, assignment and selection. Since the fundamental of Grid is the coordination and sharing of resources located at various places in decentralized ownership, the architecture may be classified in a way these resources being contributed, consumed and shared. The Grid architecture can be classified in two ways: push vs. pull model.

Push Model

Pools of resources geographically distributed and heterogeneous are known by the scheduler. A computational task, job is submitted i.e. *pushed* to the computational resources, available from the pools. It does not scale above few 1000 of jobs

Pull Model

A task is retrieved, *pulled* from generally centrally located task pool, jobs VO-dependent are kept by the experiment application. Pilots jobs run at the sites and pull jobs when site is ready.

Components of Grid environment cont'd

- Each pool of resources has a head node responsible for scheduling and dispatching of jobs to the locally connected nodes in the network
- At the end of the task the output is pushed/pulled depending on the model or stored locally.

Data Model

- The amount of data maintains an ever-increasing trend in data intensive applications such as High Energy Physics, Bioinformatics etc.
- In Grid data management leverages data locality in order to minimize the amount of data movement and improve the applications performance and scalability.
- Computational tasks are scheduled close to the data, job goes where data is.
- Data-aware schedulers are critical in achieving good scalability and performance.
- Data management architectures are important to ensure that data management implementations scale to the required dataset sizes in the number of files, objects, and dataset disk space usage while at the same time, ensuring that data element information can be retrieved fast and efficiently.
- The structure of the data, data model, is task dependent and evolves with time. Each LHC experiment has a similar data model. Biophysics, astro-particle physics, etc. have their own data model.

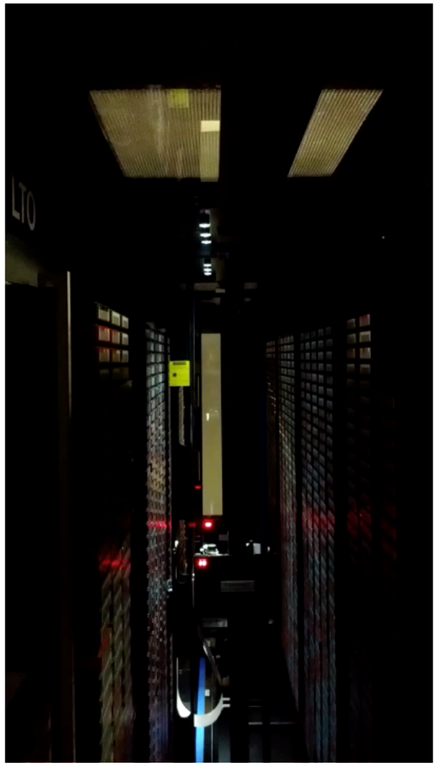
Components of Grid environment cont'd

Data Archival from hardware point of view

Data is stored on two supports depending on the data model: hard disk and tape.

<u>Tape</u>

- Magnetic tape is mainly used for offline data storage.
- A tape driver is needed to read and write a tape
- Sequential access



Components of Grid environment cont'd

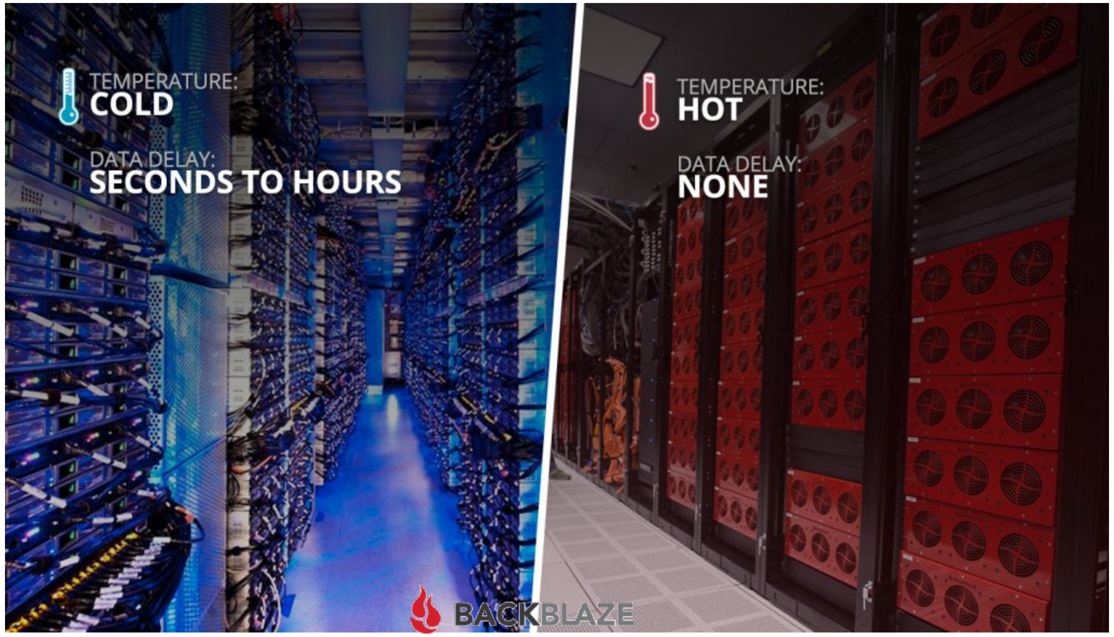
Data Archival

Disk

- Data is recorded by electronic, magnetic, optical, or mechanical changes to a surface of rotating disks. A disk drive is a device implementing such a storage mechanism.
- Disk are faster to access both in writing and reading than tape, not sequential access
- Disk are more expensive than tape.
- Easy to support from the human point of view

Disk activity

Hot storage is data that needs to be accessed right away.
Cold data is data that is accessed less frequently and also doesn't require the fast access of warmer data.



Components of Grid environment cont'd

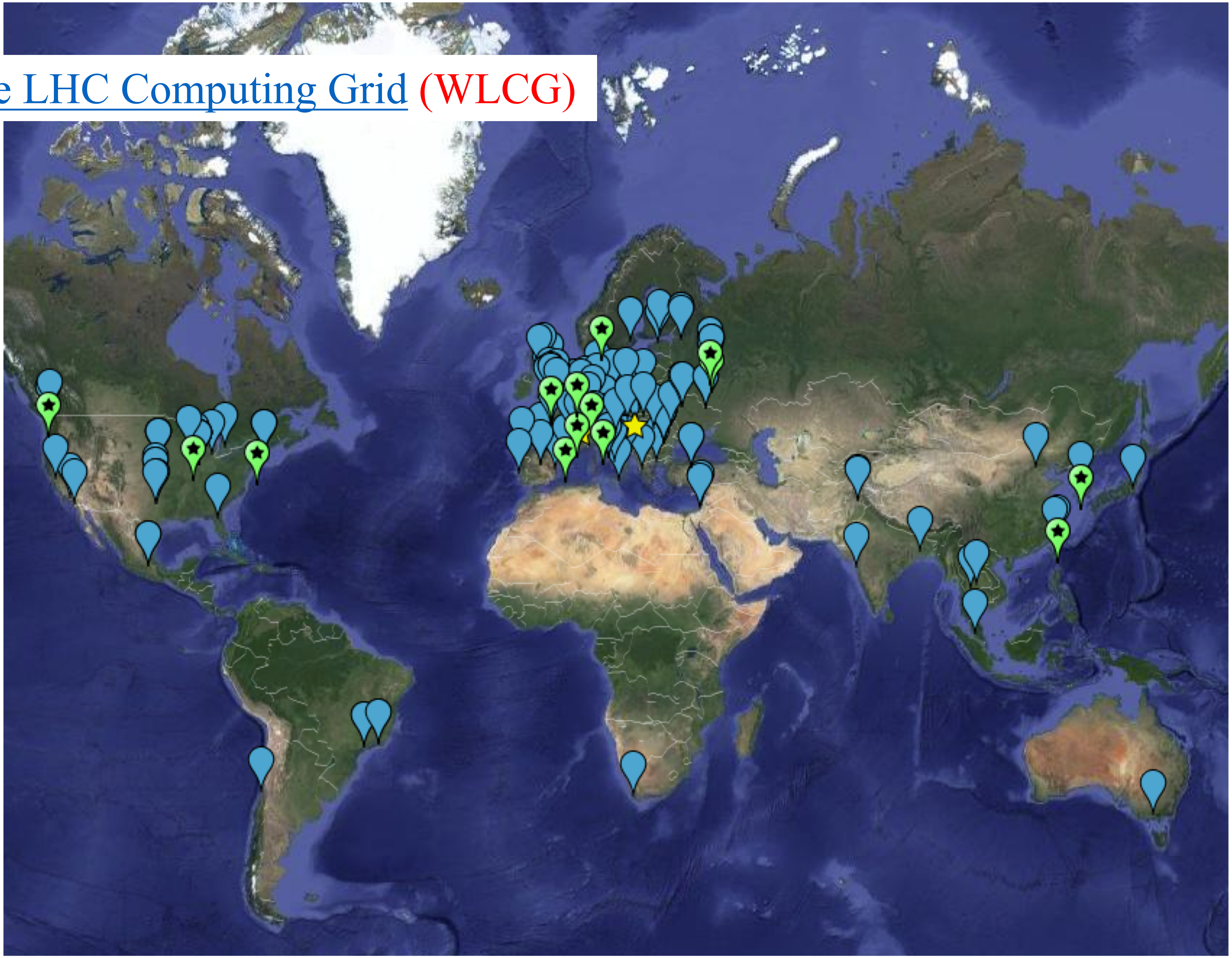
Security

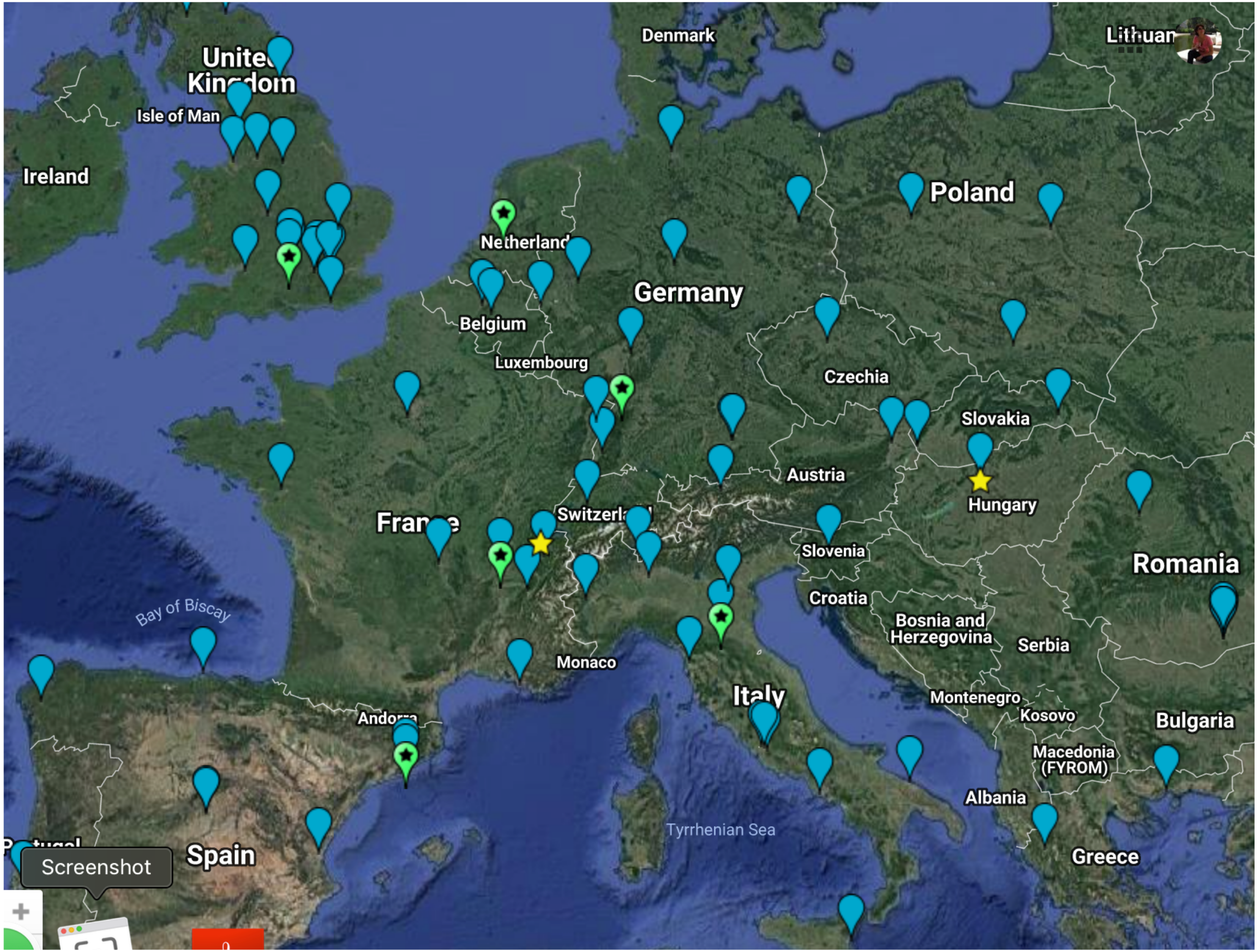
One of the most challenging task on Grid was to to develop a *secure* system → only properly *authenticated* and *authorized* users and resources participate and collaborate in the Grid environment.

Authentication is a process that uniquely distinguishes any person or resource in the Grid environment.
It is performed via digital certificates issued by Certification Authorities (CA)
Digital certificates are composed of a public key and a private key
In a Grid environment with many users and many organizations need single sign-on and identity certificates →CA

Authorization determines what each single person is allowed to do
Within the Virtual Organization (VO) each members has specified roles and rights.

Worldwide LHC Computing Grid (WLCG)



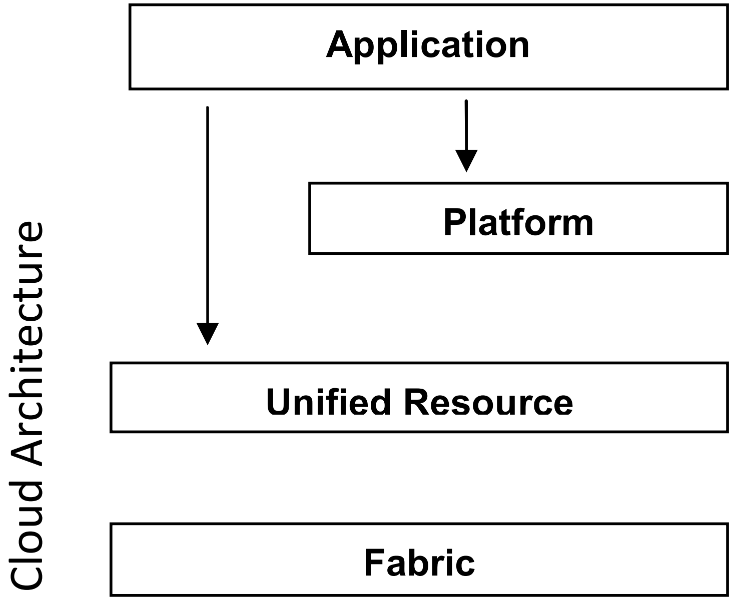

Denmark
Lithuan
United Kingdom
Isle of Man
Ireland
Poland
Netherland
Germany
Belgium
Luxembourg
Czechia
Slovakia
Austria
Hungary
France
Switzerl
Slovenia
Romania
Croatia
Bosnia and Herzegovina
Serbia
Bay of Biscay
Monaco
Italy
Montenegro
Kosovo
Bulgaria
Andorra
Macedonia (FYROM)
Albania
Tyrrhenian Sea
Greece
Spain
Screenshot


3/22/2019 8:02:30 pm
7:52 pm
8:02 pm
Running jobs: 316815
Active CPU cores: 555140
Transfer rate: 61.77 GiB/sec
WLCG
ALICE
ATLAS
Screenshot
US Dept of State Geographer
Image Landsat / Copernicus
© 2018 Google
Data SIO, NOAA, U.S. Navy, NGA, GEBCO
Google Earth

The Cloud

Clouds are developed to address Internet-scale computing problems where some assumptions are different from those of the Grids. Clouds are usually referred to as a large pool of computing and/or storage resources, which can be accessed via standard protocols via an abstract interface.

There are also multiple versions of definition for Cloud architecture, we define a four-layer architecture for Cloud Computing in comparison to the Grid architecture: fabric, unified resource, platform and application layers.



Fabric layer

contains the raw hardware level resources, such as compute resources, storage resources, and network resources.

Unified resource layer

contains resources that have been abstracted/encapsulated, usually by virtualization, so that they can be exposed to upper layer and end users as integrated resources, for instance, a virtual computer/cluster, a logical file system, a database system, etc.

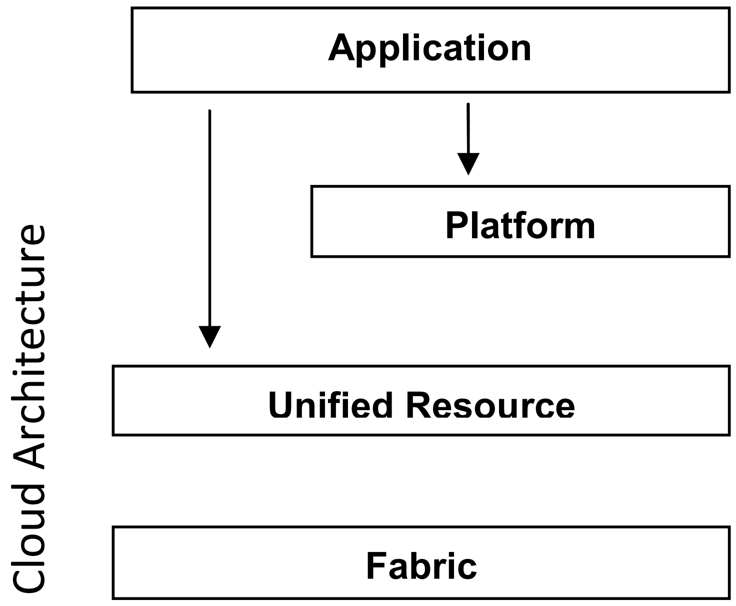
The Cloud cont'd

Platform layer

adds on a collection of specialized tools, middleware and services on top of the unified resources to provide a development and/or deployment platform. For instance, a Web hosting environment, a scheduling service, etc.

Application layer

contains the applications that would run in the Clouds.



The Cloud Services

Clouds in general provide services at three different levels: *IaaS, PaaS*, and *SaaS*

Infrastructure as a Service (IaaS)
provisions hardware, software, and equipments (mostly at the unified resource layer, but can also include part of the fabric layer) to deliver software application environments. Infrastructure can scale up and down dynamically based on application resource needs. Typical examples are Amazon EC2 (Elastic Cloud Computing) Service and S3 (Simple Storage Service)

Platform as a Service (PaaS)
offers a high-level integrated environment to build, test, and deploy custom applications. Generally, developers will need to accept some restrictions on the type of software they can write in exchange for built-in application scalability. An example is Google's App Engine, which enables users to build Web applications on the same scalable systems that power Google applications.

Software as a Service (SaaS)
delivers special-purpose software that is remotely accessible by consumers through the Internet.

There are several cloud providers:
Azure (Microsoft)
Google Cloud
AWS, Amazon Web Service

Which other cloud do you know?

cloudveneto portal

Is the Grid moving to the Cloud?

The answer is yes…
But there are several considerations before looking at the model

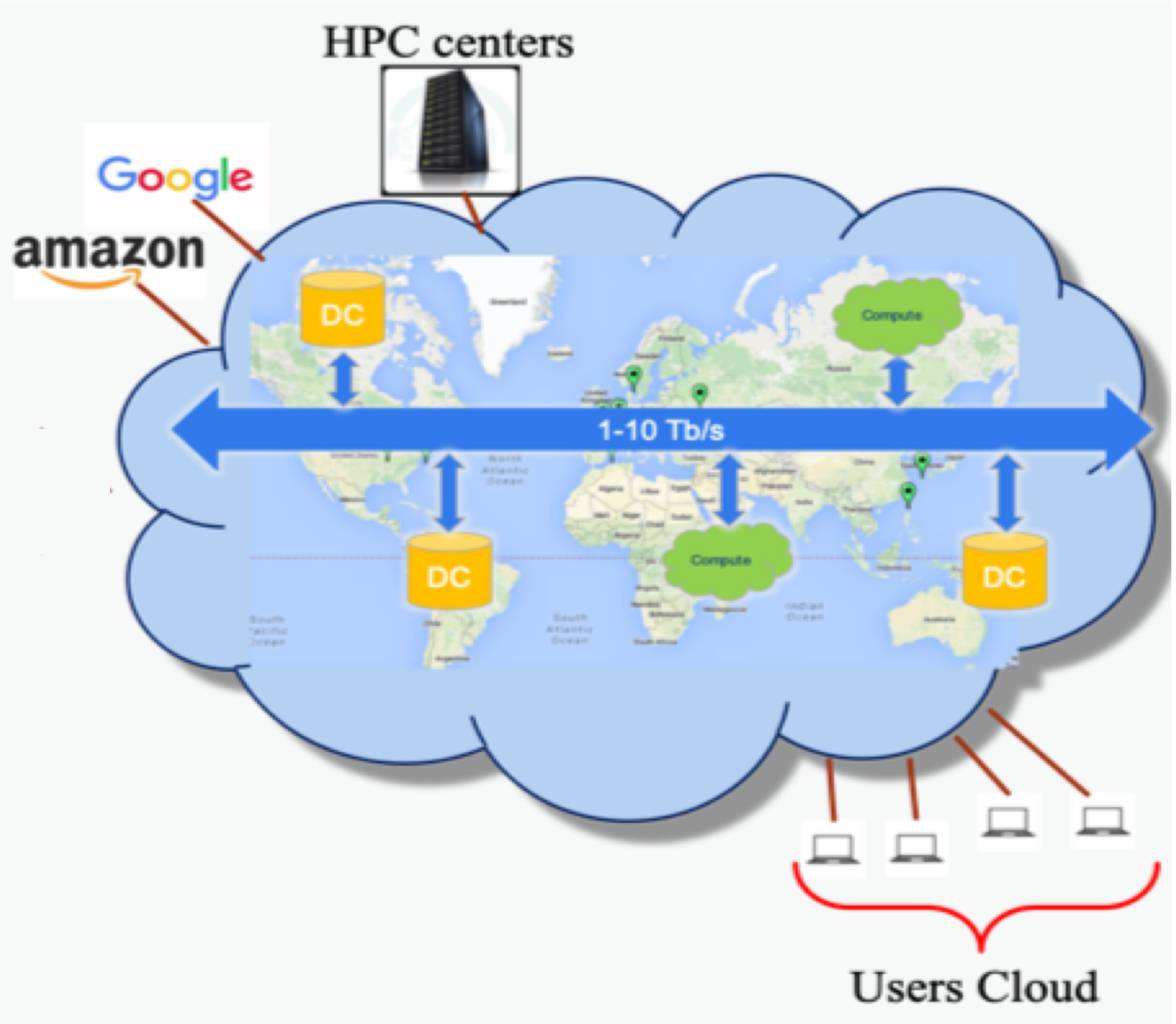
Recent Evolution

Which is the most advancement in technology in the recent years?

The network

- It is possible to transfer much more data in less time → it is possible to access data over the network
- The Grid paradigm, data locality, is not required anymore, it is possible to have a job running in Padova reading/writing data on a disk in Bari.
- Structured, semi-structured and unstructured data can be only in few large central repositories and retrieved on demand. This model is called the datalake,
- Commercial companies are already using it together with datalake analytics to manage data
- The Grid used by physicist is preparing the next generation of the distributed computing

Distribute Computing Evolution



- Very high speed network connect all the centers
- Large data centers host the data, following the datalake model
- Computing centers constitute the cloud for research
- Since the cloud is an elastic computing system is possible to include commercial companies and HPC centers.
- users can connect to the cloud via the users cloud

The role of HPC

HPC, High Performance Computing centers are not in Grid nor in Cloud
The mission of PRACE (Partnership for Advanced Computing in Europe) is to enable high-impact scientific discovery and engineering research and development across all disciplines to enhance European competitiveness for the benefit of society. PRACE seeks to realise this mission by offering world class computing and data management resources and services through a peer review process.

Data Centers

A data center can be really large

https://www.google.com/about/datacenters/

Google data center