How the LHC experiments have interpreted the Grid computing model

Mattia Cinquilli

CERN

Inverted CERN School of Computing, 25-26 February 2013


Distributed
Computing
partitioning
vertical
grid
computer
algorithm
business
power
scale
model
database
share
speed
network
servers
passing
efficient
architecture
number
dependency
OS
parallel
computing
synchronization
transaction
CPU
task
vertical
model
CPU
algorithm
distributed
business
network
node
partitioning
communication
cloud
speed
program
grid
performance
resources
message
function
resources
load
process
computing
system
share
distributed
servers
communication
program
scale
passing
data
parallel
processor
efficient
computer
node
system
function
work
horizontal
processor
message
work
number
horizontal
load
performance
OS
future
future
cloud
transaction
process
architecture

CERN School of Computing

Outline

- **Introduction to Distributed Computing and Grid**
- **Worldwide LHC Computing Grid**
- **LHC's experiments computing models**
- **Evolutions**

Introduction

- **Start**
  - Single processor
- **Supercomputer**
- **Mainframes**
- **More resources needed, new use cases**
  - Cluster Computing

Distributed Computing

- **Distributed computation**
  - Set of processes cooperating for a common result
  - Peculiarities: processes do not share memory and communication through network
    - communication has delays (unpredictable)
  - Possibly large geographic areas
  - Asynchronous
- **Pros:**
  - Scalability
  - Redundancy
- **Cons:**
  - Complexity
- **Client-Server**
  - Various topologies:
    - 2-tiers
    - 3-tiers
    - N-tiers

Sharing of resources through time

ERA	Computer-Human relation	Sharing architecture	Supercomputing architecture
70's	1-many	Time sharing systems	Supercomputer
80's	1-1	PC & workstation	Supercomputer
90's	Many-1	Clustering	Cluster of Workstations
2000	Many-many	Grid	Grid
~Today	Many-many	Grid/Cloud	Grid/Cloud

CERN School of Computing

Grid Computing

- **A large-scale geographically distributed hardware and software infrastructure composed of heterogeneous networked resources owned and shared by multiple administrative organizations**
- **Main advantages:**
  - Usage of distributed resources
  - Responsibilities distributions
  - No central management: local
  - Resources are: loosely coupled, heterogeneous and geographically distributed

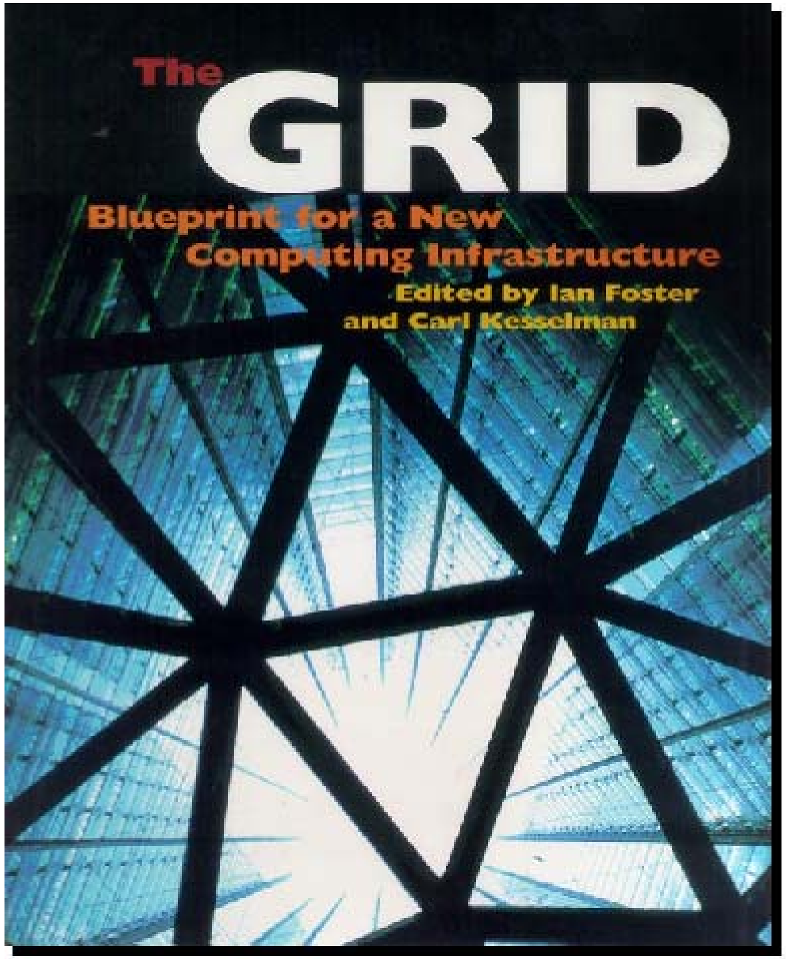
Grid Computing evolution and topology

- **Intra-grid**
  - Combination of clusters used by single organizations
  - “Department” or “Campus Grid”
- **Extra-grid**
  - Combination of intra-grids geographically distributed used by more organizations
  - VPN connections
- **Inter-grid, “Global Grid”**
  - Combination of intra-grids geographically distributed used by more organizations
  - Connections over the Internet

Definition of Grid Computing

- **"A computational grid is a hardware and software infrastructure that provides dependable, consistent, pervasive, and inexpensive access to high-end computational capabilities."**

- **I. Foster and C. Kesselman, "The Grid: Blueprint for a New Computing Infrastructure", Morgan Kaufman, USA, 1998**



CERN School of Computing

plus…

- **"...Grid concept is coordinated resource sharing and problem solving in a dynamic, multi-institutional virtual organizations."**

- **I. Foster, C. Kesselman and S. Tuecke "The anatomy of the Grid", International Journal of Supercomputer Applications, 15(3), 2001**

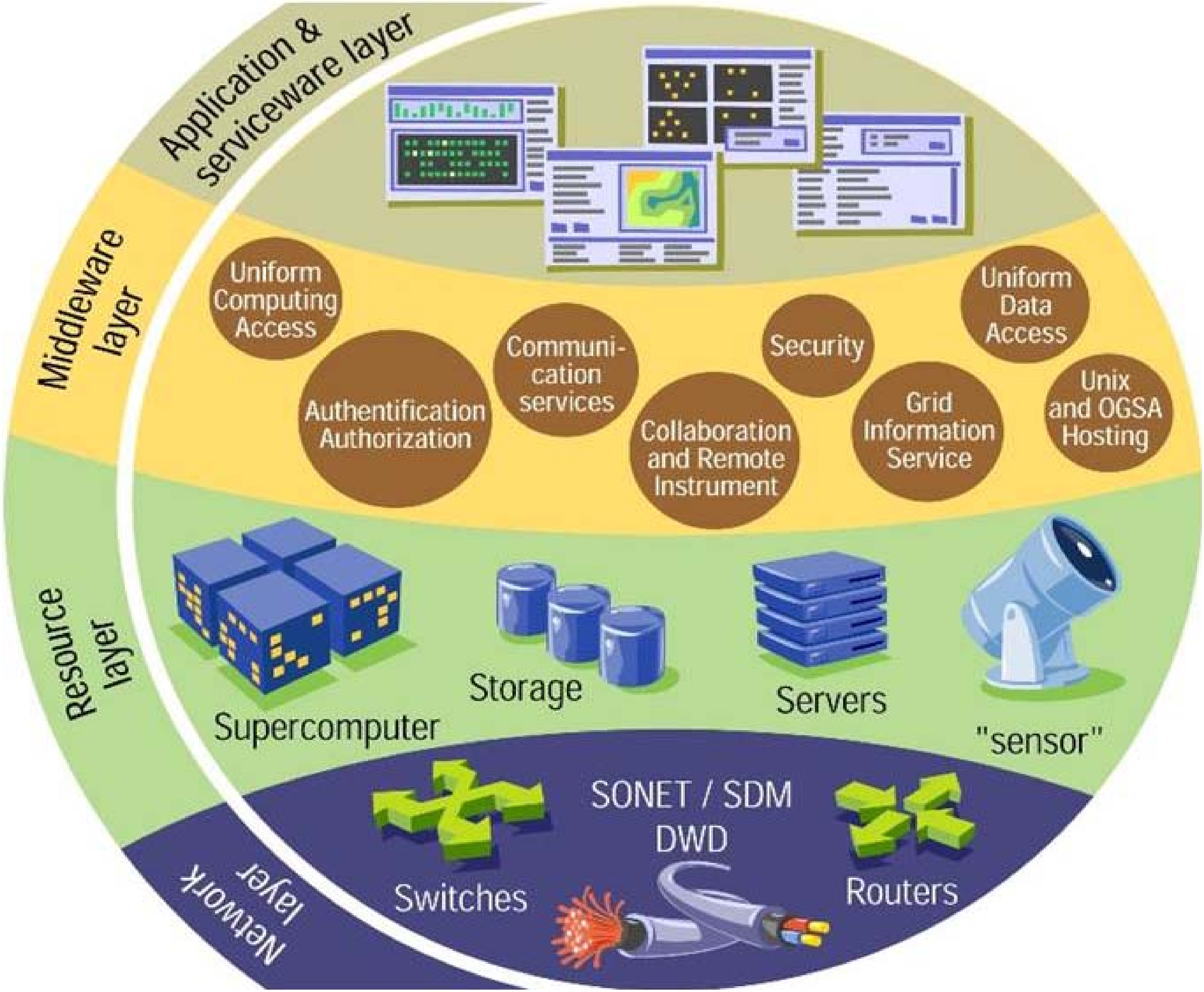
…again…

- **“A Grid is a system that:**
  - Coordinates resources that are not subject to centralized control
  - using standard, open, general-purpose protocols and interfaces
  - to deliver nontrivial qualities of service”

- **I. Foster “What is the Grid? A Three Point Checklist”, 2002**

CERN School of Computing

Architectural levels



gridcafe.org

LHC use case

- **LHC will run for 20 years**
  - Experiments are producing about 15 Million Gigabytes of data each year
  - LHC data analysis requires a computing power equivalent to ~200,000 of today's fastest PC processors
  - Requires many cooperating computer centres
    - CERN can provide ~20% of the capacity
- **A challenge**
  - Computing power
  - Volume of data

Main LHC computing requirements

- **Data volume**
  - Manage very large volume of data at very high data rates
    - Proton collision rate is 10^7-10^9 Hz
    - Estimated ~15PB/year of data
- **Resources**
  - Provide enough computational and storage capacity
- **Users**
  - Allow data access to thousands of dispersed users
    - More then 8000users, from ~35 countries
- **Data archival**
  - Long term data archival
    - LHC lifetime estimated at ~15 years

CERN School of Computing

Grid as a Solution

- **Placing all computing and storage power at CERN to satisfy requirements it's not possible**
- **CERN member states have always encouraged to have ~60% of computing needs located outside CERN**
  - Institutes participating in LHC have pre-existing resources
  - WAN will improve
- **Need a distributed system to coupe with the scale**
  - Technical and social advantages

The Worldwide LHC Computing Grid

CERN School of Computing

WLCG

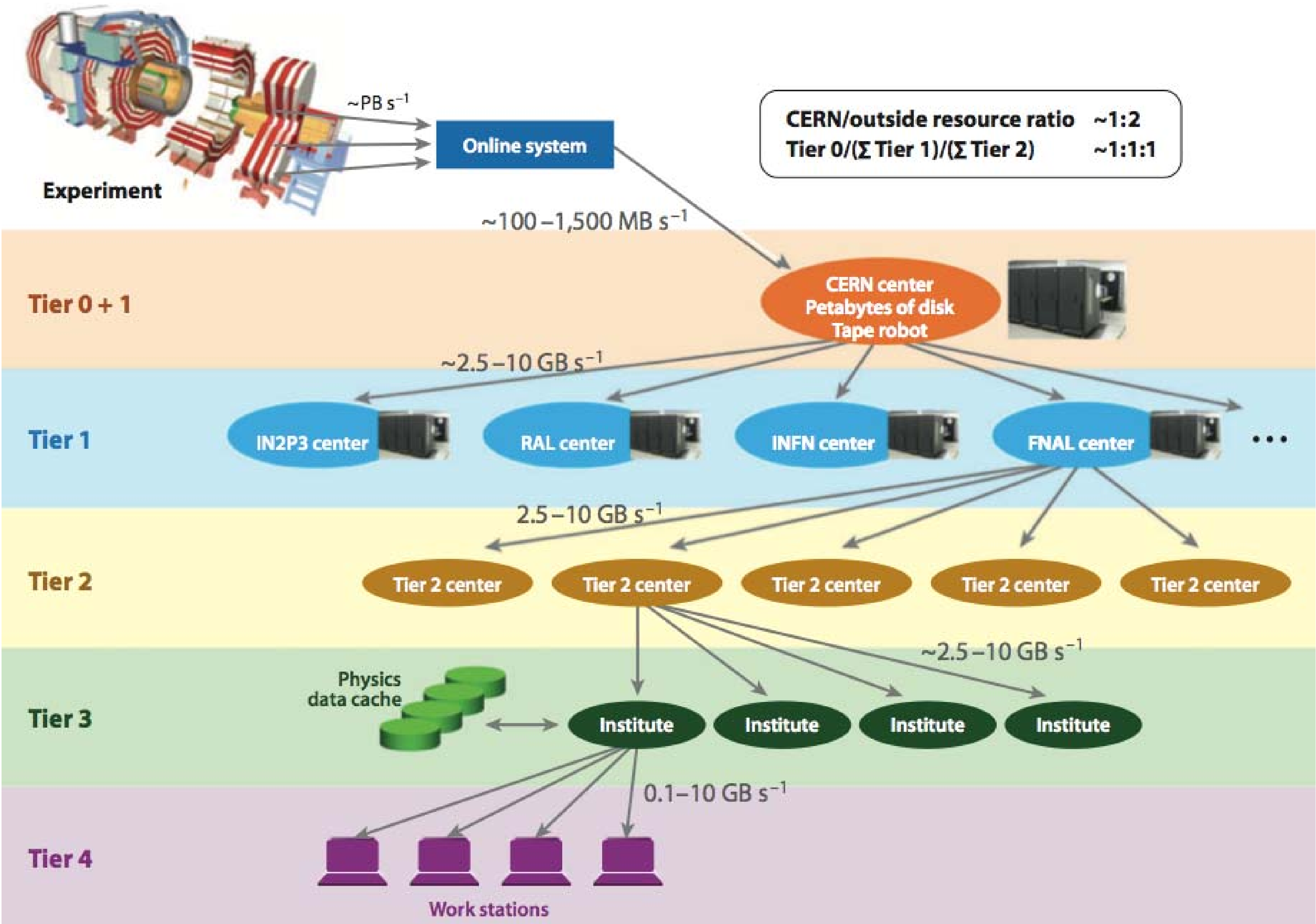
- **A distributed computing system to provide the computing infrastructure for the LHC experiments**
  - Managed and operated by a worldwide collaboration
  - Between the experiments and the participating computer centres
  - Memorandum of Understanding between CERN and founding agencies (includes Service Level Agreements)
- **The resources are distributed**
- **Resource usage available to LHC collaborations independently from their location**

Worldwide LHC Computing Grid

Distributed Computing Infrastructure for LHC experiments

- **Linking 3 distributed infrastructures**
  - OSG Open Science Grid in the US
  - EGI European Grid Infrastructure
    - Includes Asian and South American sites
  - NDGF Nordic Data Grid Facility
- **Linking more than 300 computer centers**
  - Providing more then 300000 cores
  - With more than 2000 (active) users
  - Moving ~10GB/s for each experiment
- **Archiving ~15PB per year**

MONARC model



Main building blocks

- **Virtual Organization (VO)**
  - Entity corresponding to a particular organization in the real world with specific privileges to their users. E.g.: a user belonging to the atlas VO will be able to exploit resources reserved for the ATLAS collaboration.
- **Computing Element (CE)**
  - Grid entry point to the site, which manages and knows the internal status of resources
- **Storage Element (SE)**
  - Atomic unit in the storage Grid infrastructure, supporting a variety of protocol families; it can correspond to various systems (disk server/arrays, tape disk)

CERN School of Computing

Other building blocks

- **User Interface (UI)**
  - User entry point to the Grid, located at between the application and middleware level
- **Information System (IS)**
  - Has the global overview on the resources and their status
- **Workload Management System (WMS)**
  - Redirect user jobs to the appropriate Grid sites (CE), performing the match-making
- **MyProxy**
  - Service providing a “single sign-on” like functionality for the authentication in the Grid services
- **File Transfer Service (FTS)**
  - Low level data movement service to schedule asynchronous file replication from a source SE to another (third party copy)

Grid Monitoring

Experiment usage of WLCG

- **Take most of advantages from the underlying system**
- **Application layer of the Grid architecture**
  - Insulates users from hardware, network, middleware, and all other complexities of the underlying system
- **Developed software layer to integrate the applications with the distributed computing environment**
  - Implementation based on different choices resulted in distinct models

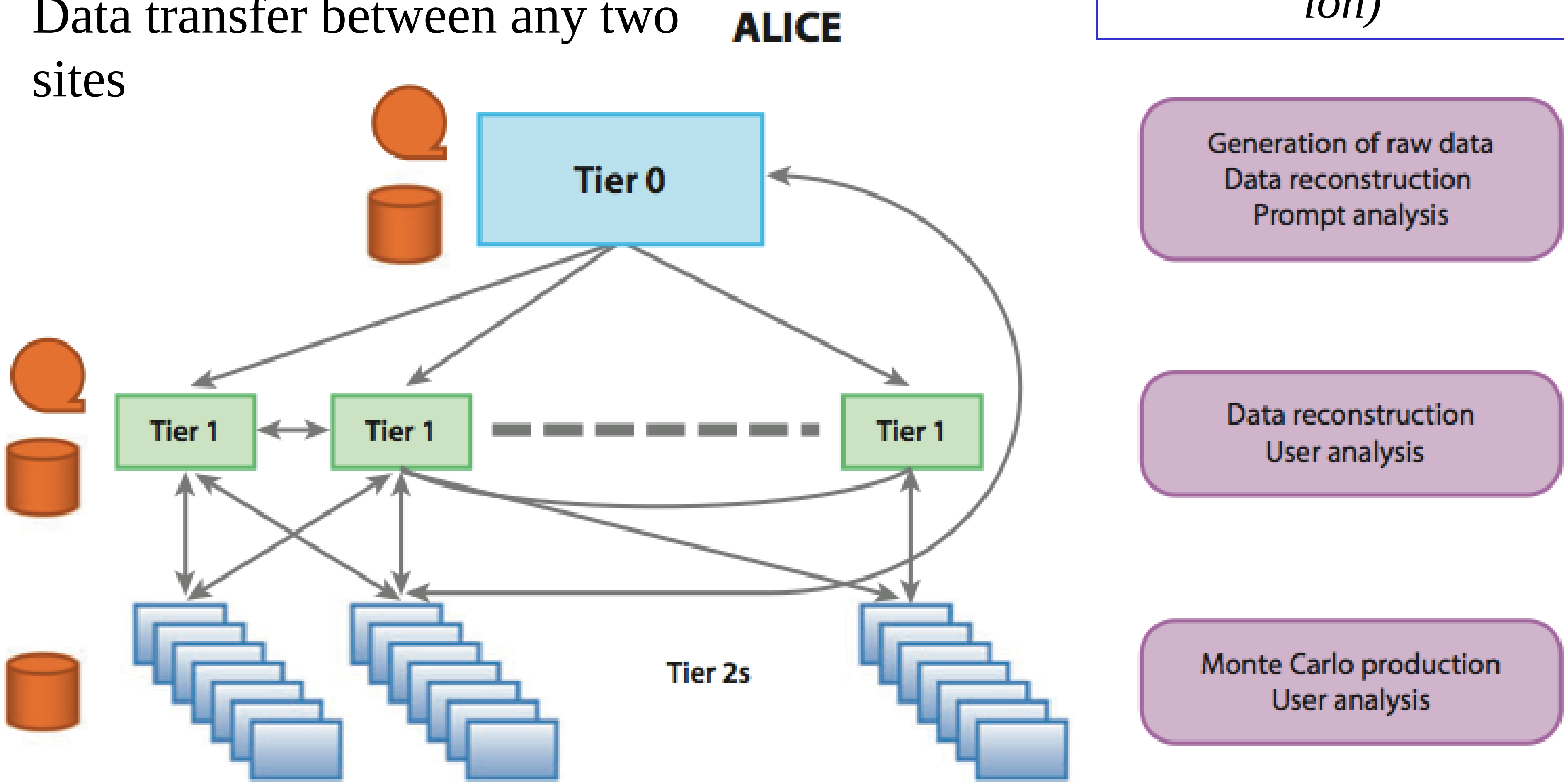
Grid: LHC's experiment interpretations

- **Why different interpretations**
  - Founding agencies
  - Schedules
  - Going into something still not well known, not already experienced in the same way in the world
- **Contra**
  - Lot of effort, not just developing but also maintaining/operating
- **Pros**
  - Possibility to have independent solutions
  - Possibility to optimize the infrastructure for specific activities
  - Easier to develop (not too generic)

ALICE Computing Model (simplified)

- Data archival the only difference between Tier-1/Tier-2
- Data transfer between any two sites

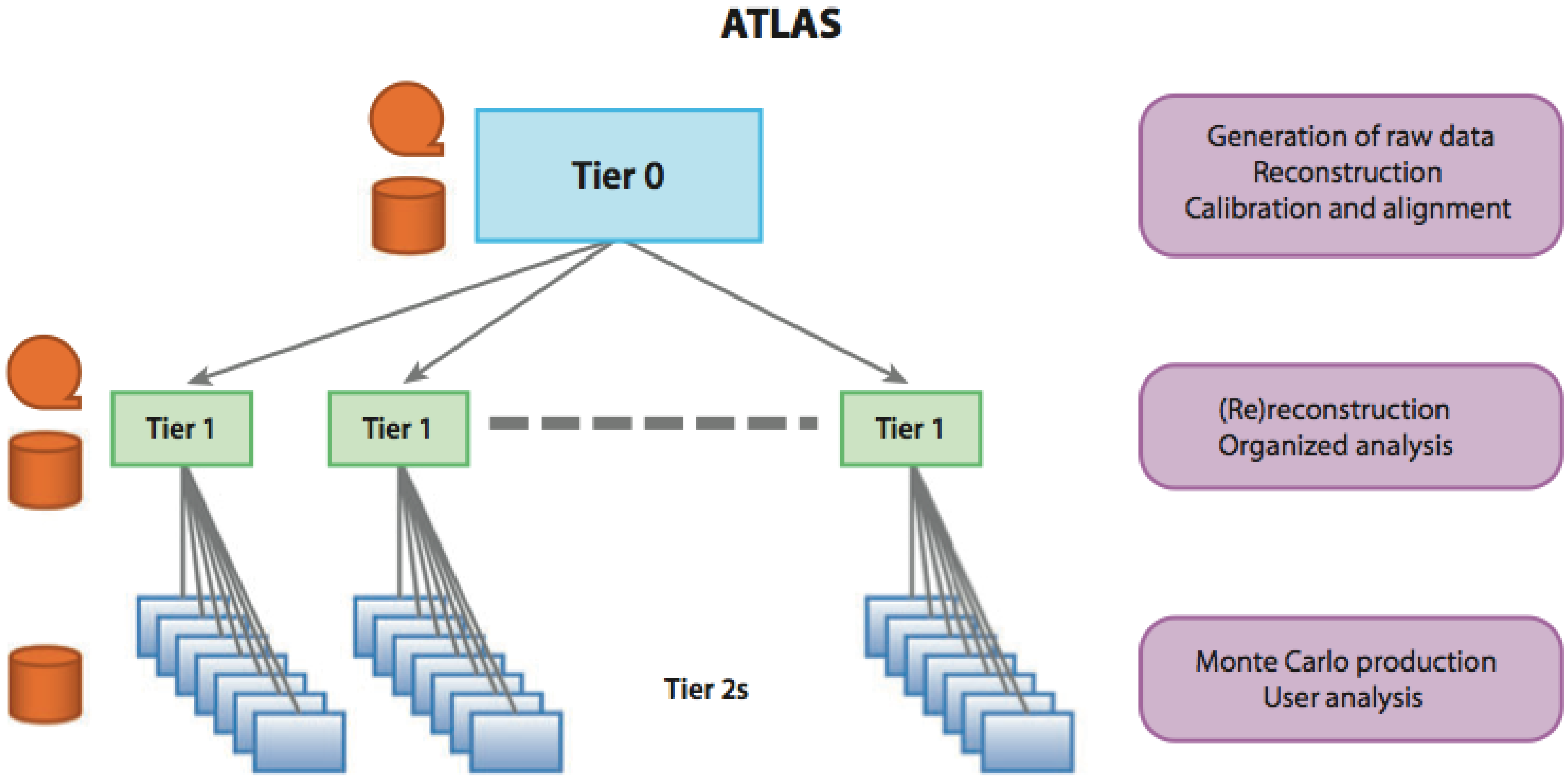
Data rate to Tier-0 1.5GB/s (during heavy-ion)



ATLAS Computing Model (simplified)

- The closest to the MONARC concept

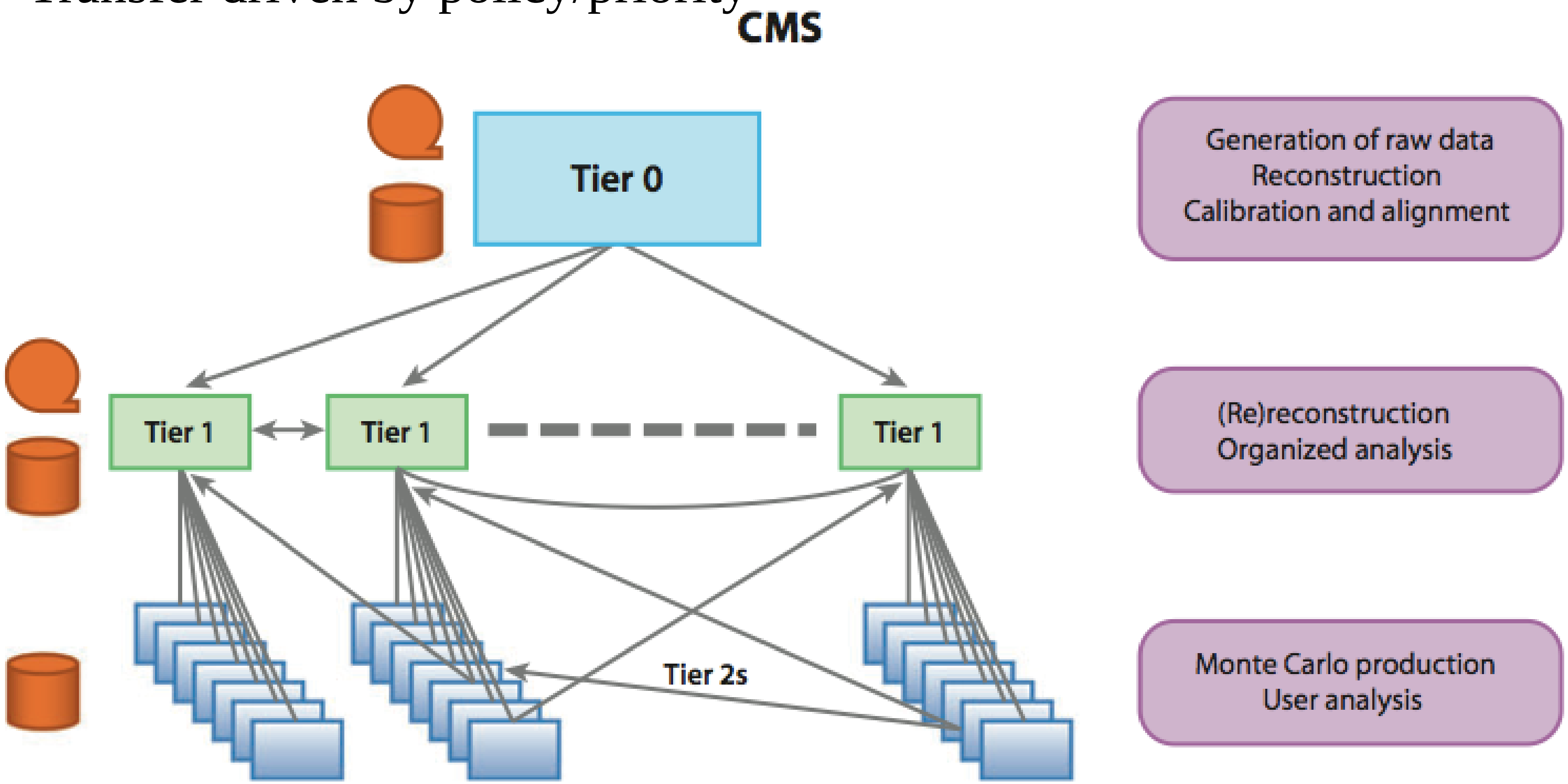
Data rate to Tier-0
~350MB/s



CMS Computing Model (simplified)

- Less hierarchical then MONARC
  - Data moved from any Tier1-Tier2 site
  - Transfer driven by policy/priority

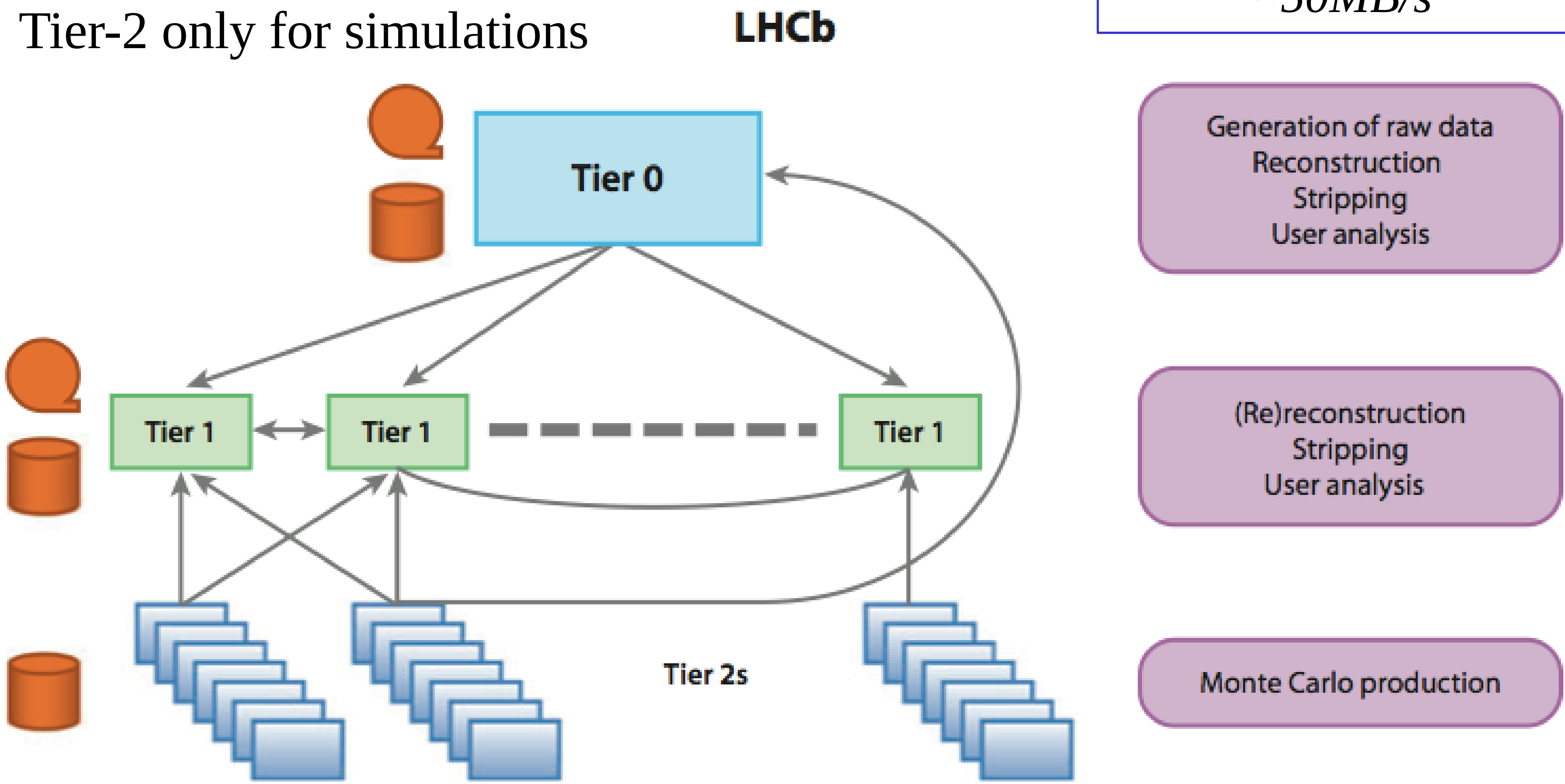
Data rate to Tier-0 ~250MB/s



LHCb Computing Model (simplified)

- Only Tier-1 used for analysis
- Tier-2 only for simulations

Data rate to Tier-0 ~50MB/s



CERN School of Computing

Data management, the basics

- **Common basic concepts**
  - Jobs run where data is
    - importance of data placement
  - High-level dataset replication systems using low-level transfer systems
  - All models designed around the fact that network capacity and reliability won't be enough
- **but also several differences, mainly:**
  - ATLAS uses a very hierarchical model of T0→T1→T2 transfers
    - very close to WLCG MONARC model
  - CMS enabled transfers between all T1-T2 sites
    - more connections, mode dynamic
  - LHCb uses Tier-1 for data analysis, Tier-2s only for simulation
  - ALICE allows data transfer between any 2 sites

CERN School of Computing

Data placement for analysis

- **Two fundamental ways of placing the data:**
  - Static, Pre-placement, Predictive
    - Data is statically allocated to the storage elements of the various Grid sites
  - Dynamic, Adaptive, Opportunistic
    - Data is dynamically allocated depending on the current request rate by user (popularity based) and site load

	Static	Dynamic
ALICE		✓
ATLAS	✓	✓
CMS	✓	
LHCb	✓	

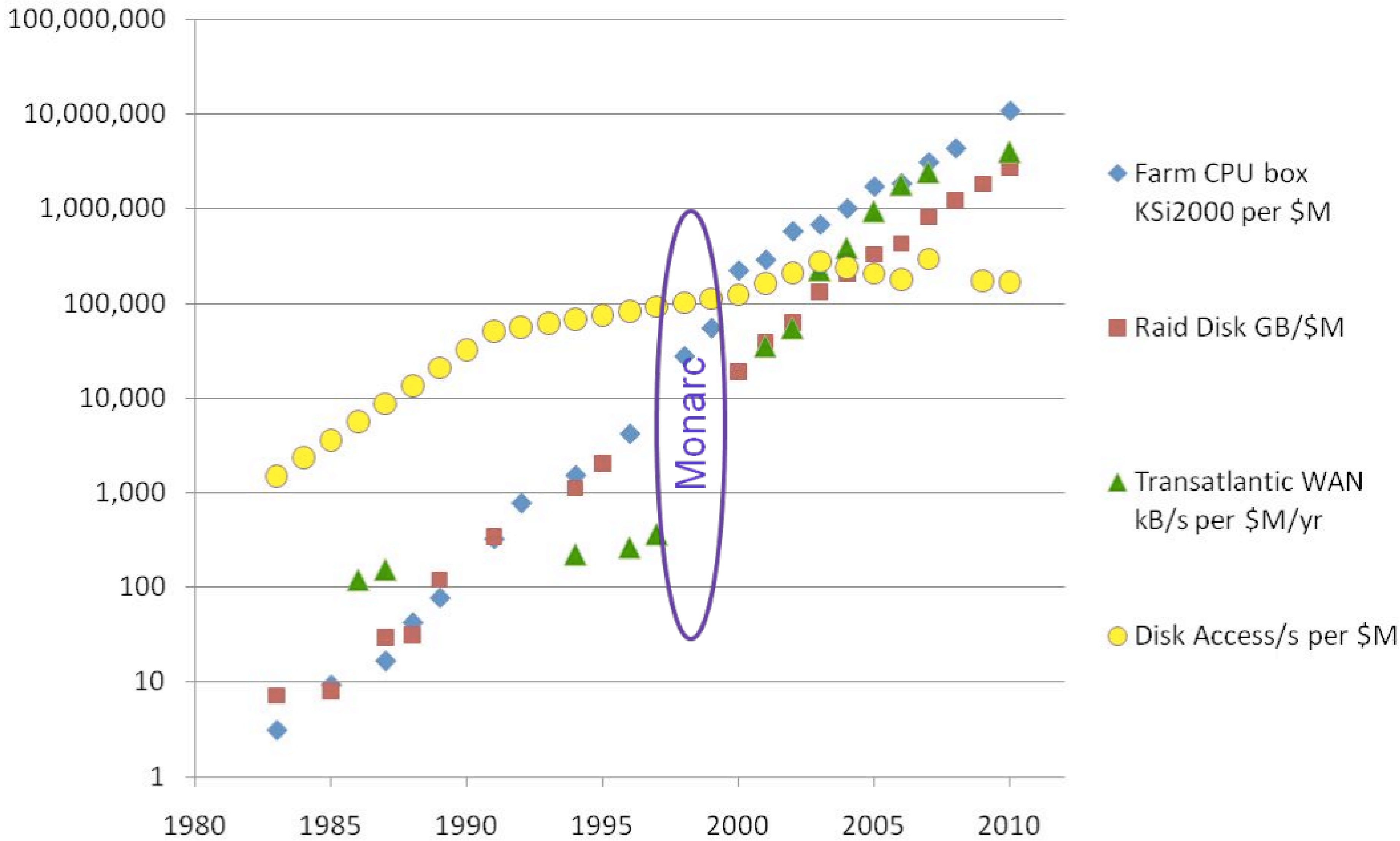
Data location principle: why?

- **Data driven scheduling**
  - Sending jobs close to where the data is stored
- **Why was this the chosen solution at the time?**
  - The network is a very limited resource, potentially a bottleneck
  - Need a *hierarchical mass storage*, cannot keep everything in a local disk space
    - Disk/Tape hierarchy
  - Job runs “close” to data, achieving efficient CPU utilization
  - Need a structured and predictable data utilization
- **Impact the whole model, including:**
  - Data management– disk/network resource usage
  - Job scheduling - CPU resource usage

What happened during Run-1 (2009-2012/13)

- **Data transfers between sites more reliable than predicted**
  - WAN network performances rapidly improving
  - Network infrastructure reliable
- **Geographically distributed job submission and resource usage are working well**
  - Relevant effort on workload management tools to help the final users
  - Grid usage transparent to the end user
- **Hierarchical mass storage system is complex to manage and requires effort**
  - Retrieving files directly from remote sites sometime easier than using a local hierarchical mass storage system

Resource evolution



Network as a resource

- **The WAN bandwidth is comparable with the backbone available at LAN level**
- **Some Tier-2 sites are larger then some Tier-1 sites**
  - Most (All?) of Tier-2's in US have 10Gbps capability
  - Large flows for some Tier-1 – Tier-2 (even 10Gbps)
  - Tier-2 - Tier-2 data flows are becoming significant.
- **Regional transfer of data is basically broken**
  - Data locality concept can be relaxed
  - Remote access of data stored in any regional site

CERN School of Computing

Data management evolution

- **Evolution from a hierarchical mass storage system to a full mesh/peer-to-peer**
  - Reduced latency in data transfers
  - Increased working efficiency
- **Possibility to remotely read data when needed without dramatic impact on CPU efficiency**
  - Hiding local data issues/failures to the users
- **New challenge coming in the next future related to the available IOPS on storage systems**
  - Need to optimize the IO at the application level: “disks will not increase the performance too much in the future”
    - Or would instead be ~like the network

CERN School of Computing

Data management evolution Goals

- **Reliability, Transparency, Usability**
- **Allows a user to open (almost) any file, regardless of their location or the file's source**
  - As the Grid definition says!
- **Diskless sites**
  - Grid Tier-3
  - Clouds (private/public/institutional)
  - Other opportunistic resources
- **Fallback solutions**
- **WAN data access**
  - Clouds of site, federated storages

gLite Workload Management System

- ***"…comprises a set of grid middleware components responsible for the distribution and management of tasks across grid resources, in such a way that applications are conveniently, efficiently and effectively executed"***
    - From the gLide middleware documentation

- **Push model**
  - Working as a super-batch system
  - Jobs submitted to the WMS which schedules the jobs to a Grid CE (computing center)
    - Matchmaking operation
  - Computing centers implement their internal batch queues to schedule jobs on the worker nodes.

Push mode

- **Pros**
  - Potentially a unique generic service for all the experiments
- **Cons**
  - Black-hole worker nodes
  - Difficult to manage all real time information from a central point
  - Complexity
  - Difficult to satisfy specific use cases
- **Experiments have then implemented their solutions**
  - Integration between application and middleware layer

CERN School of Computing

Job Scheduling

- **The four LHC experiments have independently either developed or integrated their WMS**
  - Frameworks born to manage high-level user workflows
  - Direct control on translation from workflow into grid jobs
  - All experiments are converging on pilot job management systems

- **Pull model**
  - Pilot jobs are asynchronously submitted jobs which are running on worker nodes
  - Users submit jobs to a centralized queue
  - Pilot jobs communicate with the WMS (pilot aggregator) pulling user jobs from the repository

Job scheduling: pull mode

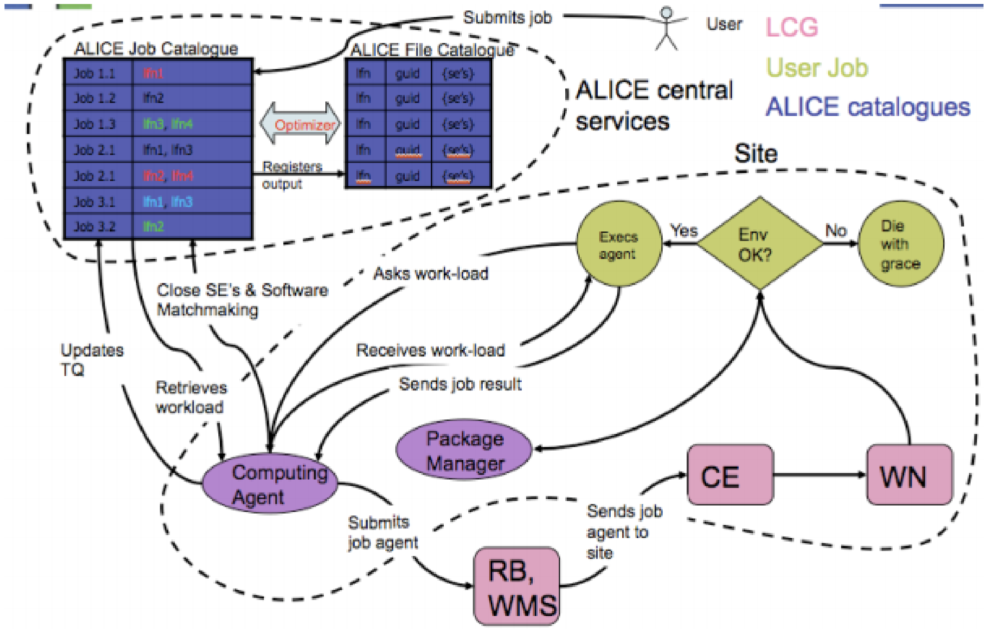
- **Pros**
  - Priority management
  - Resource usage optimization
  - Reduced overhead
  - Increased efficiency
    - Real jobs are sent only when resource is acquired
- **Cons**
  - Identity issues
    - Job that authenticates different from job that runs.
    - User credentials management at worker node level
  - Monitoring
  - Currently experiment specific solution

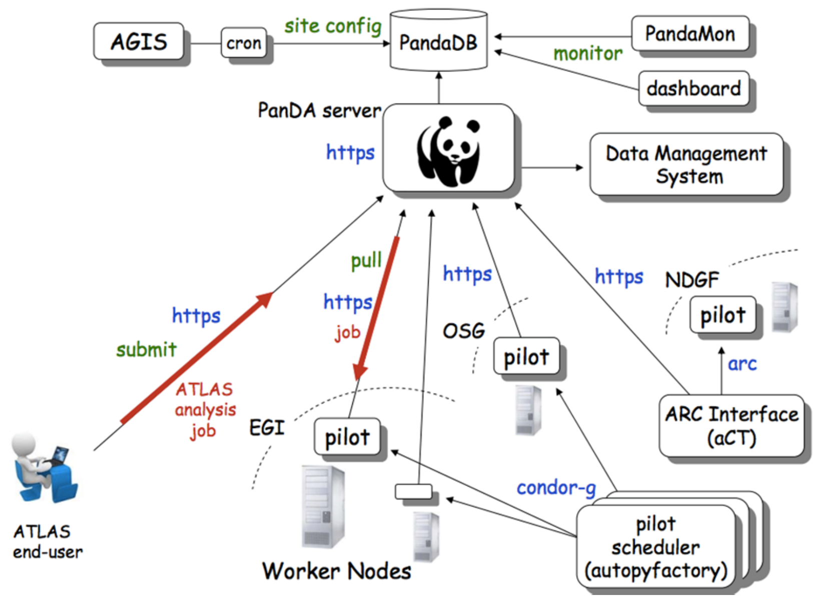
CERN School of Computing

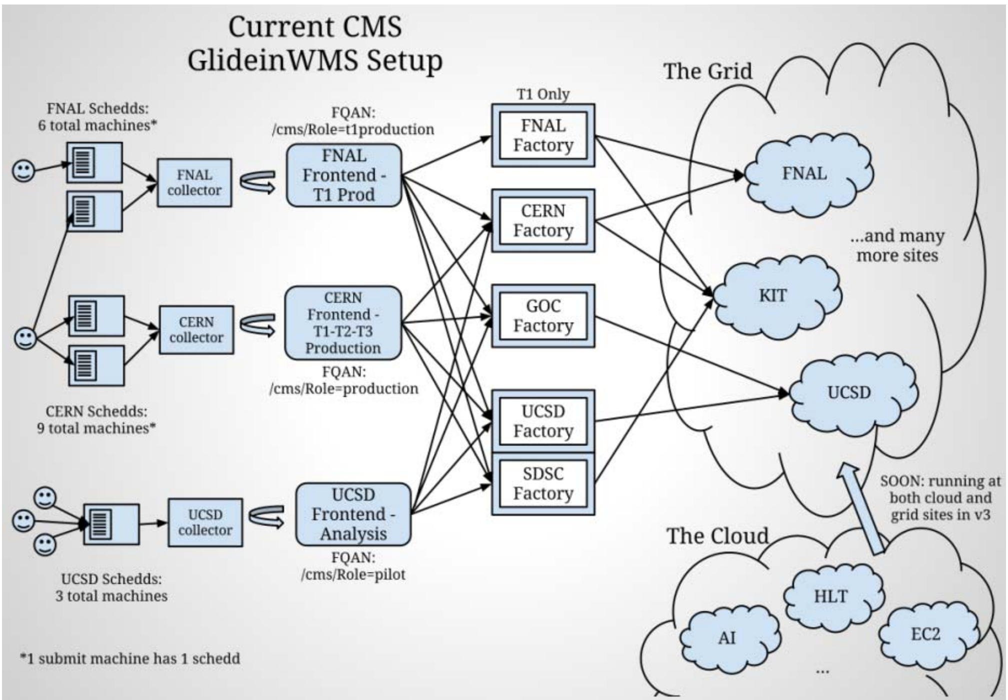
Job Management Systems

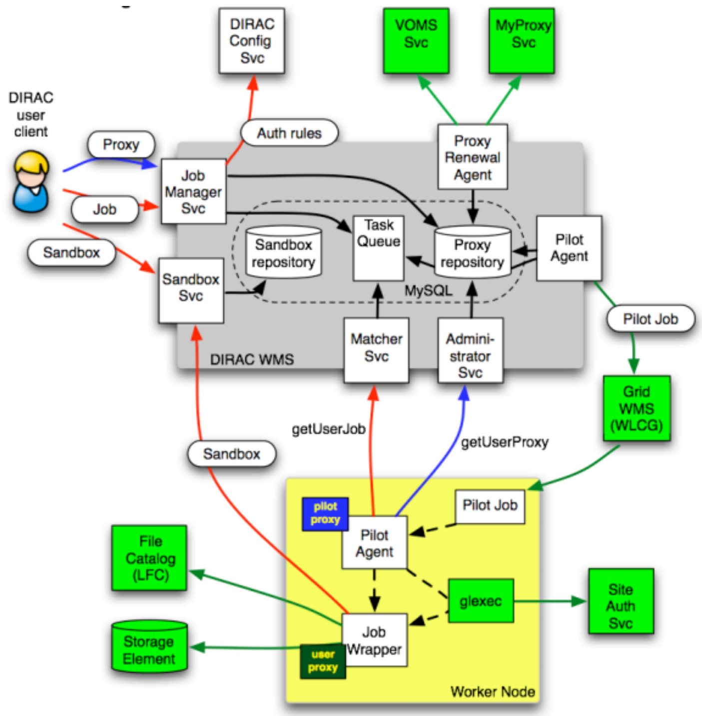
- **CMS**
  - *glideinWMS* a solution fully relying on Condor
- **ATLAS**
  - *PANDA* uses Condor mainly to abstract the Grid submission layer and CE implementations.
- **LHCb**
  - *DIRAC*: dedicated workload management based on pilot agents
- **ALICE**
  - *ALiEn*: dedicated Grid framework including job management system

Job management in the experiments







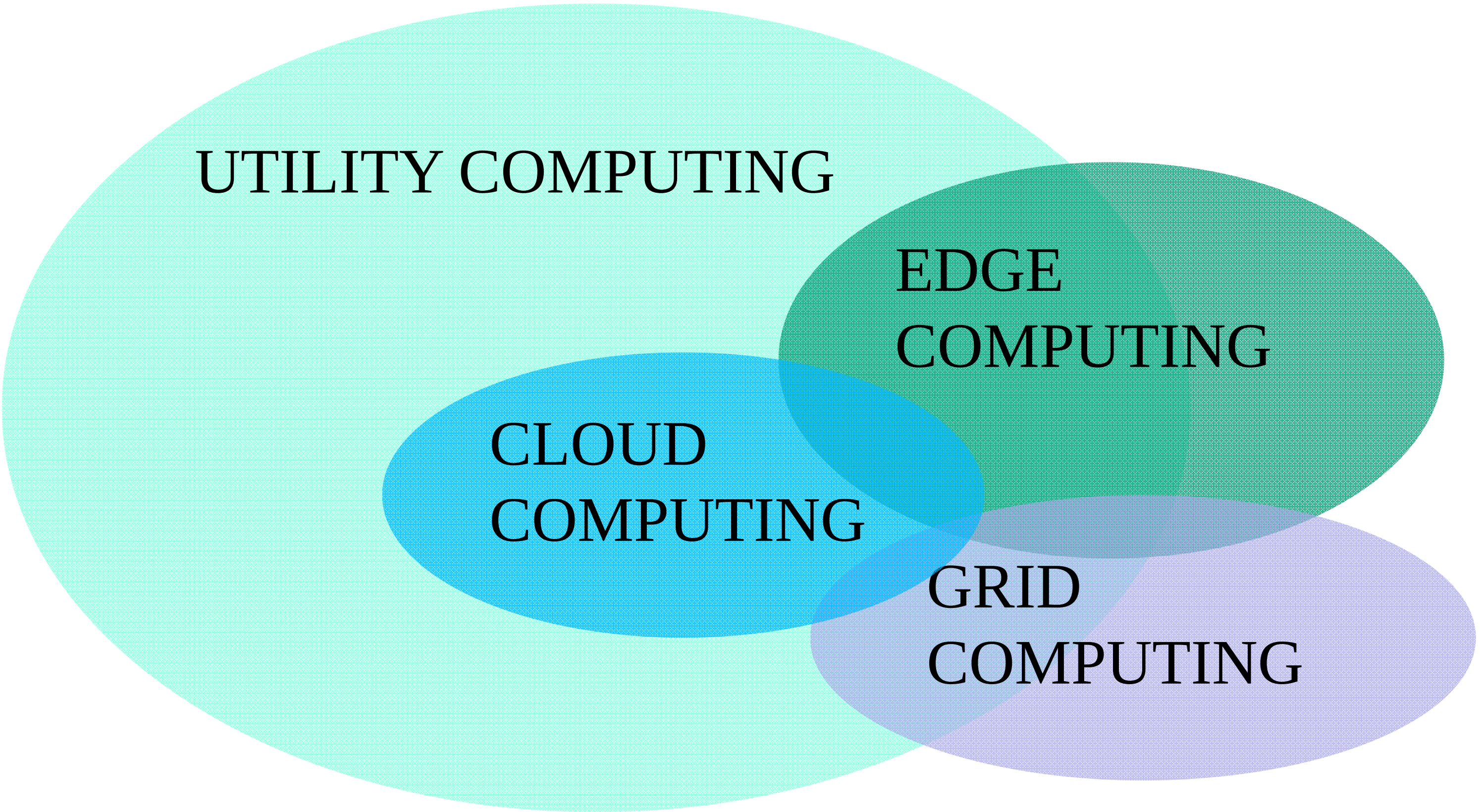


CERN School of Computing

Evolutions

- **Maintaining different dedicated solution to perform similar tasks is expensive**
  - Evolving towards common solutions as solution
- **Ad hoc solutions require dedicated experience and "no manuals"**
  - Evolving towards standards instead of ad hoc home made solution
- **Computing models and solutions will always evolve following trends (resources, markets, …)**
  - 20 years experiment!

Computing as a Service



CERN School of Computing

Grid and Clouds

- **Cloud Computing**
  - Remote data center with a well defined business model
    - Resources are centralized
    - Virtualization as key feature
- **Can it be adopted by LHC experiments/WLCG?**
  - Probably yes, but …(see next slide)
- **Why?**
  - Infrastructure virtualization
  - Sites are going to cloud
  - Dynamic provisioning of resources
  - Additional resources
  - Offload work to absorb computing peaks
    - Avoid to maintain huge amount of resources not always needed

iCSC CERN School of Computing

Clouds in the Grid?

- **Different kind of clouds available:**
  - Public/Commercial Clouds
    - Commercial resources
  - Private/Academic Clouds
    - A Grid site using a cloud infrastructure to manage its resources
- **Many aspect to be evaluated**
  - Cost of private Clouds
  - Which is the efficiency impact? (CPU, I/O, …)
  - How to integrate Cloud resources into the Grid middleware?
  - Standard technologies

CERN School of Computing

References

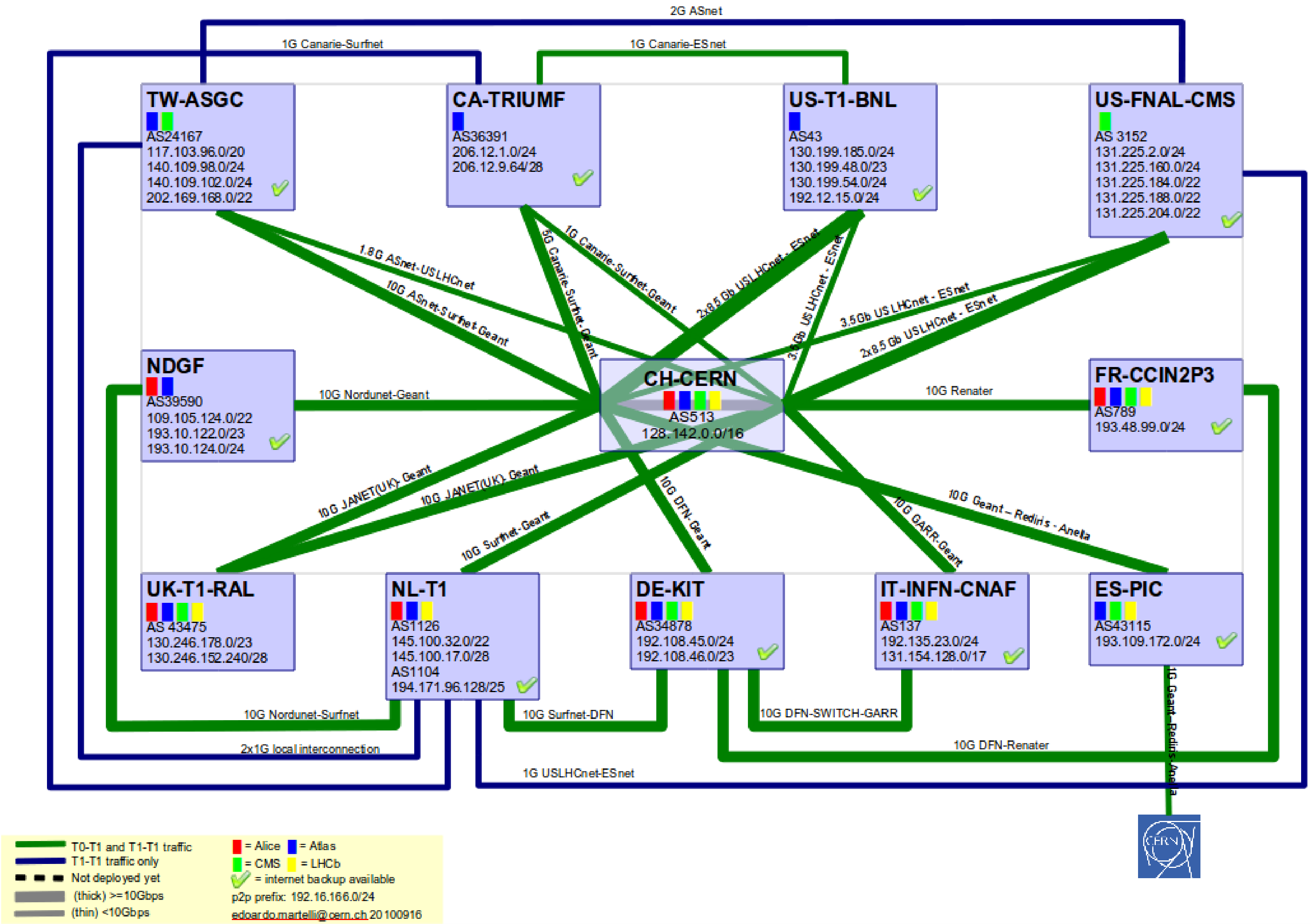
- I. Bird, Computing for the Large Hadron Collider, Annual Review of Nuclear and Particle Science, 2011
- F. Panzieri et all, Distributed Computing in the 21st Century: Some Aspects of Cloud Computing, Dependable and Historic Computing, 2011
- M. Cinquilli, Automated system development for the Grid distributed analysis workflow inside the CMS experiment, M.S. Thesis, 2010
- J.P. Baud et all, The LHCb Data Management System, CHEP2012
- G. Bagliesi et all, CMS resource utilization and limitations on the grid after the first two years of LHC collisions, CHEP2012
- T. Maeno et all, Overview of ATLAS PanDA Workload Management, CHEP2012
- C. Grigoras et all, A new communication framework for the ALICE Grid, CHEP2012
- F. Barreiro et all, Exploiting Virtualization and Cloud Computing in ATLAS, CHEP2012

Backup slides

CERN School of Computing

LHC Optical Private Network

LHCOPN



Parallelism

- **Task parallelism**
  - parallelization of computer code across multiple processors in parallel computing environments
  - distributes execution processes across different parallel nodes

- **Data parallelism**
  - computing across multiple processors
  - focuses on distributing the data across different parallel computing nodes