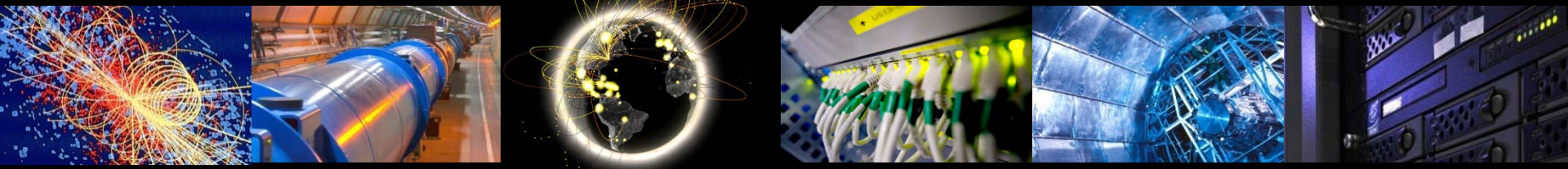
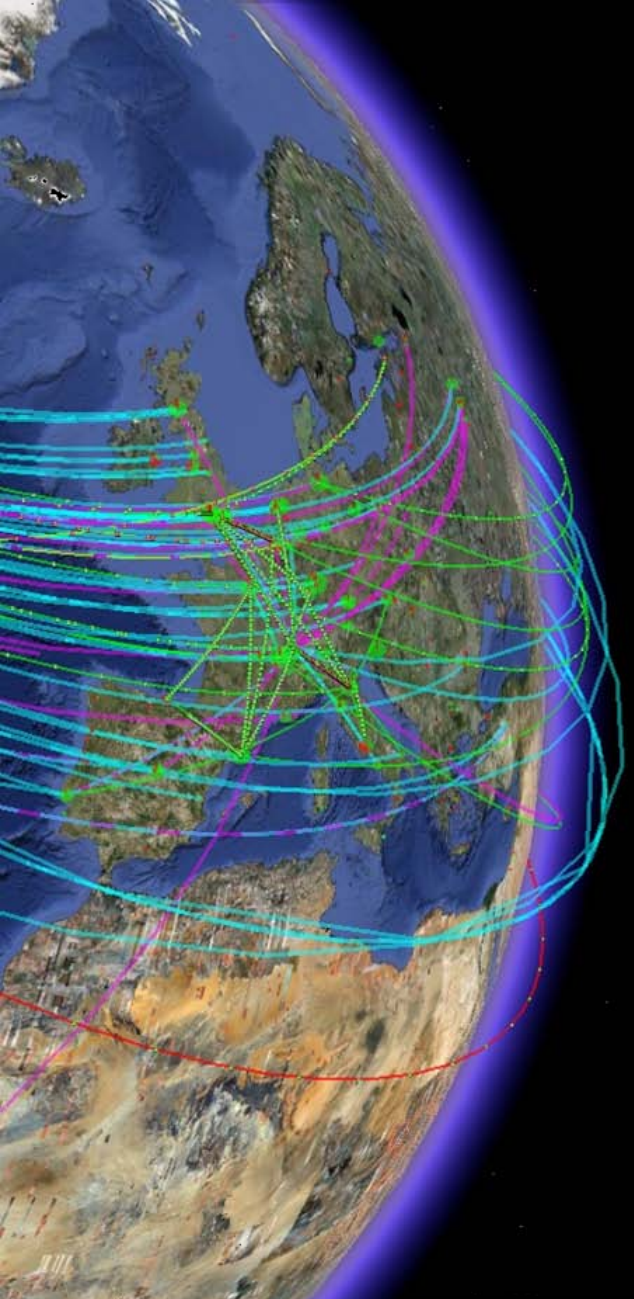
Grid Computing
ESI 2011

Markus Schulz

IT Grid Technology Group, CERN
WLCG
Markus.schulz@cern.ch

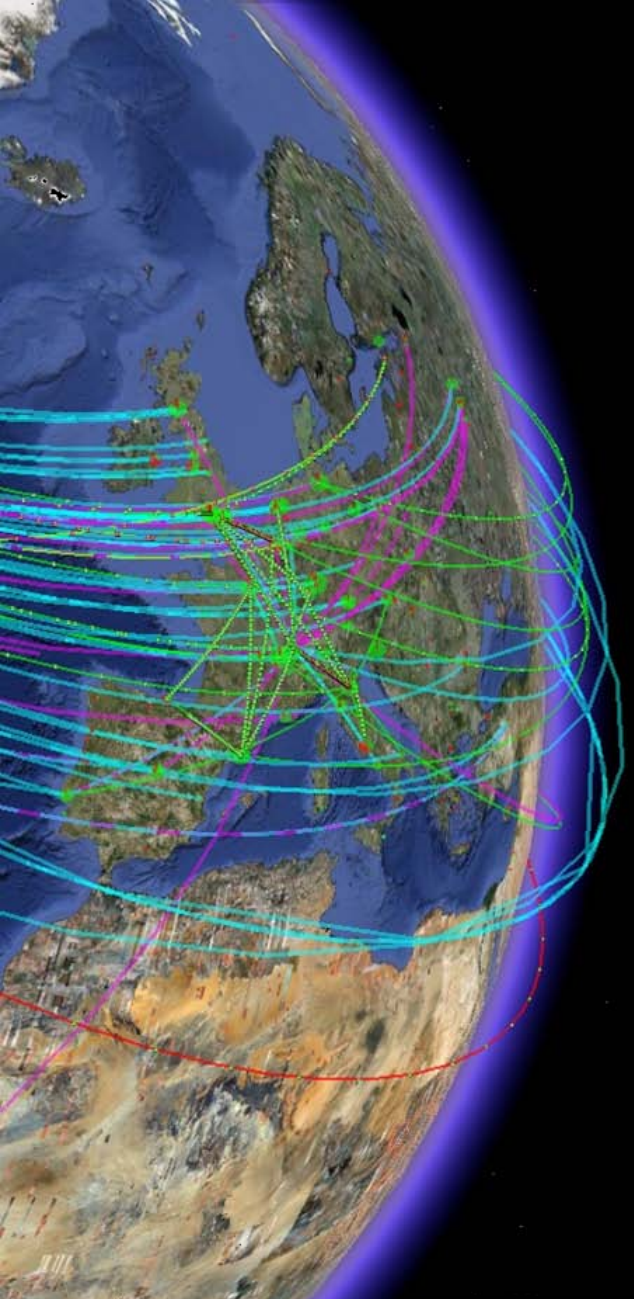


WLCG
Worldwide LHC Computing Grid



Overview

- Grid Computing
  - Definition, History, Fundamental Problems, Technology
  - EMI/gLite
- WLCG
  - Challenge
  - Infrastructure
  - Usage
- Grid Computing
  - Who?
  - How?
- What's Next?



Focus

- Understanding concepts
- Understanding the current usage
- Understanding whether you can profit from grid computing
- Not much about projects, history, details…

- If you want practical examples:
- https://edms.cern.ch/file/722398/1.4/gLite-3-UserGuide.pdf

What is a Computing Grid?

- There are many conflicting definitions
  - Has been used for several years for marketing…
    - Marketing moved recently to "Cloud-computing"
- Ian Foster and Karl Kesselman
  - "coordinated resource sharing and problem solving in dynamic, multi-institutional virtual organizations. "
  - These are the people who started globus, the first grid middleware project
- From the user's perspective:
  - I want to be able to use computing resources as I need
  - I don't care who owns resources, or where they are
  - Have to be secure
  - My programs have to run there
- The owners of computing resources (CPU cycles, storage, bandwidth)
  - My resources can be used by any authorized person (not for free)
  - Authorization is not tied to my administrative organization
- **– NO centralized control of resources or users**

Other Problems?

- The world is a fairly heterogeneous place
  - Computing services are extremely heterogeneous
- Examples:
  - Batch Systems (controlling the execution of your jobs)
    - LSF, PBS, TorQue, Condor, SUN-GridEngine, BQS, …..
    - Each comes with its own commands and status messages
  - Storage: Xroot, CASTOR, dCache, DPM, STORM,+++
  - Operating Systems:
    - Windows, Linux (5 popular flavors), Solaris, MacOS,….
    - All come in several versions
  - Site managers
    - Highly experienced professionals
    - Scientists forced to do it (or volunteering)
    - Summer students doing it for 3 months…….

What is a Virtual Organization?

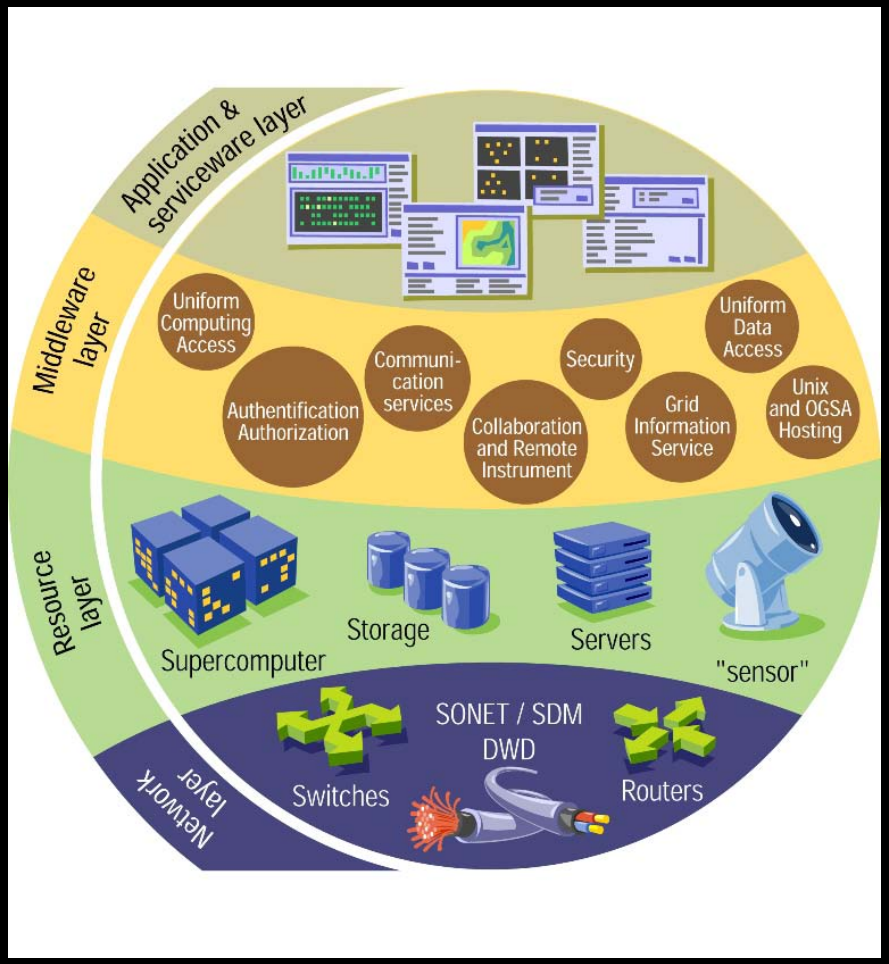
- A Virtual Organization is a group of people that agree to share resources for solving a common problem
  - The members often belong to different organizations
  - The organizations are often in different countries
  - High Energy Physics Collaborations are a good example

Fundamental Problems

- Provide security without central control
- Hide and manage heterogeneity
- Facilitate communication between users and providers

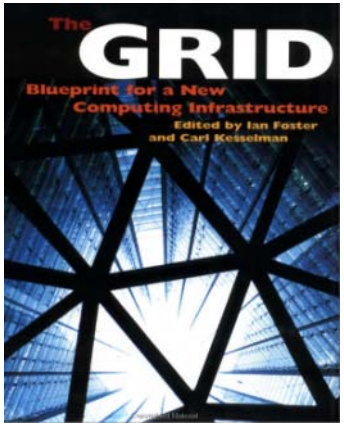
- Not only a technical problem!

- Grid Middleware is the software to address these problems



Short History

- 1998 The GRID by Ian Foster & Carl Kesselman
  - Made the idea popular
- 1998 Globus-1 first middleware widely available
  - Proof of concept
  - www.globus.org evolved to gt-5 (2010)
- Since 1998 several hundred middleware solutions
- OpenGridForum works towards standardization
  - Progress is slow…..
- LHC experiments use:
  - gLite, ARC, OSG (globus, VDT), Alien



Software Approach

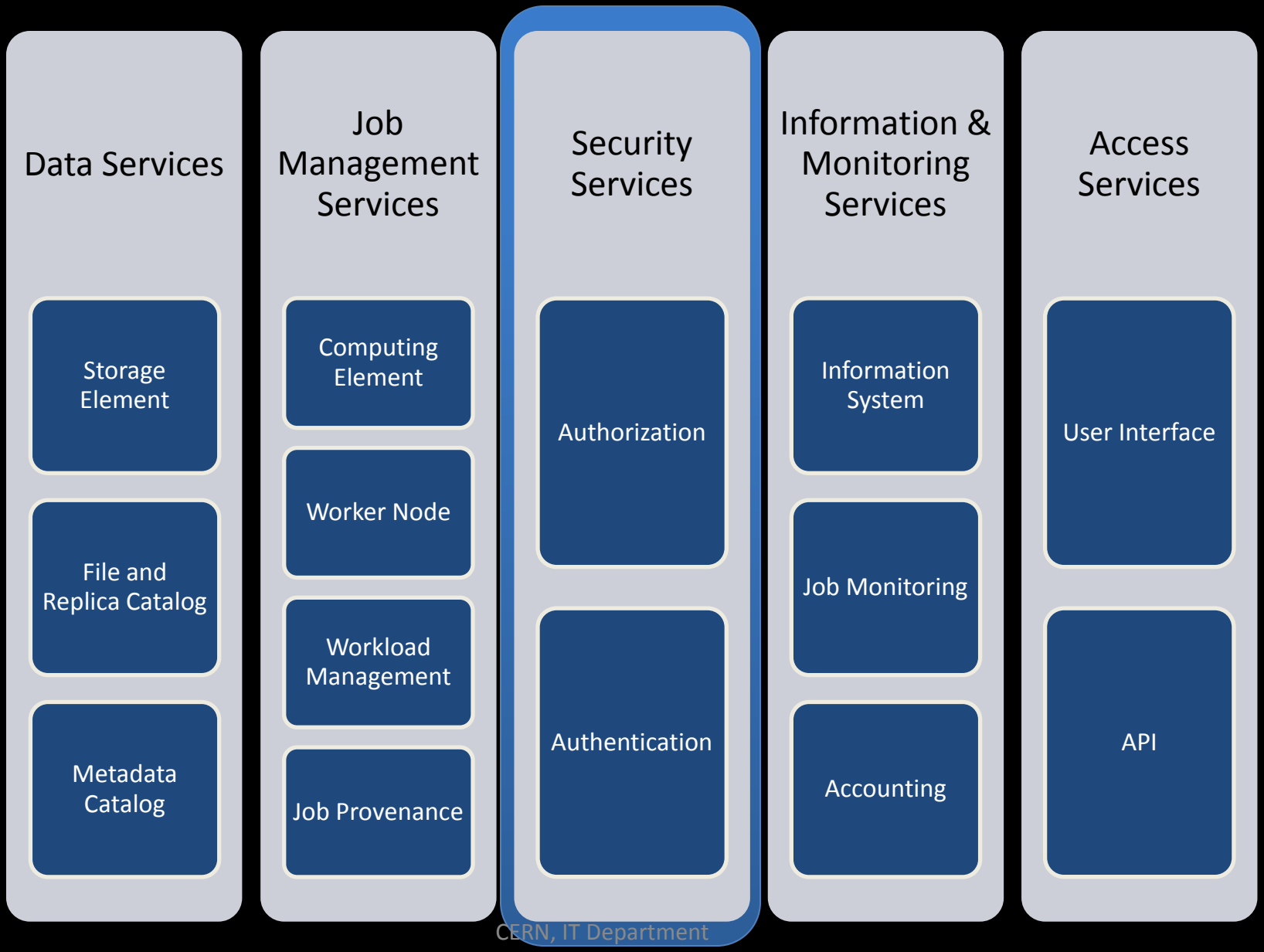
- Identify an AAA system that all can agree on
  - Authentication, Authorization, Auditing
  - That doesn't require local user registration
  - That delegates "details" to the users (Virtual Organizations)
- Define and implement abstraction layers for resources
  - Computing, Storage, etc.
- Define and implement a way to announce your resources (Information System)
- Build high level services to optimize the usage
- Interface your applications to the system

gLite as an example

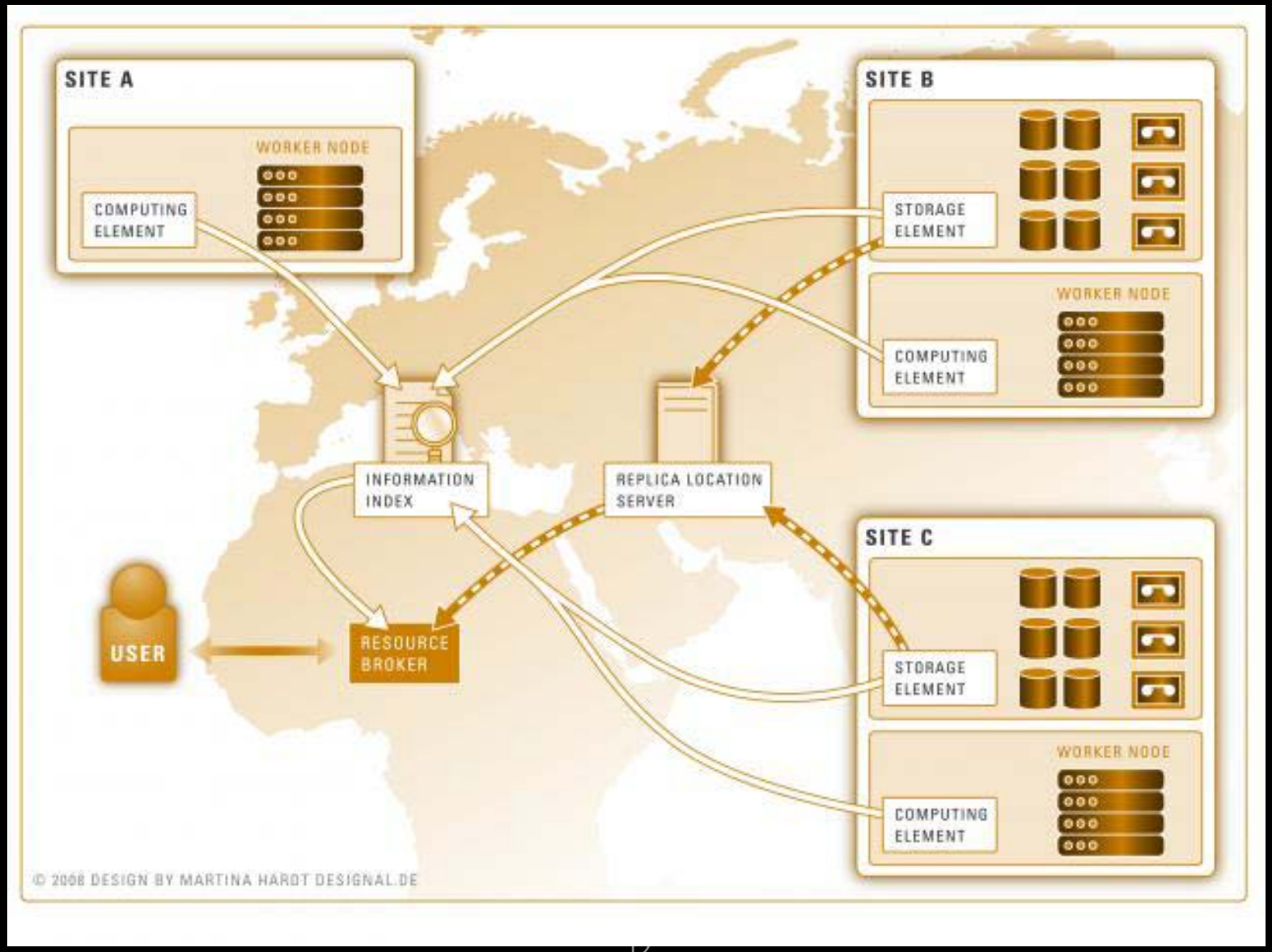
gLite

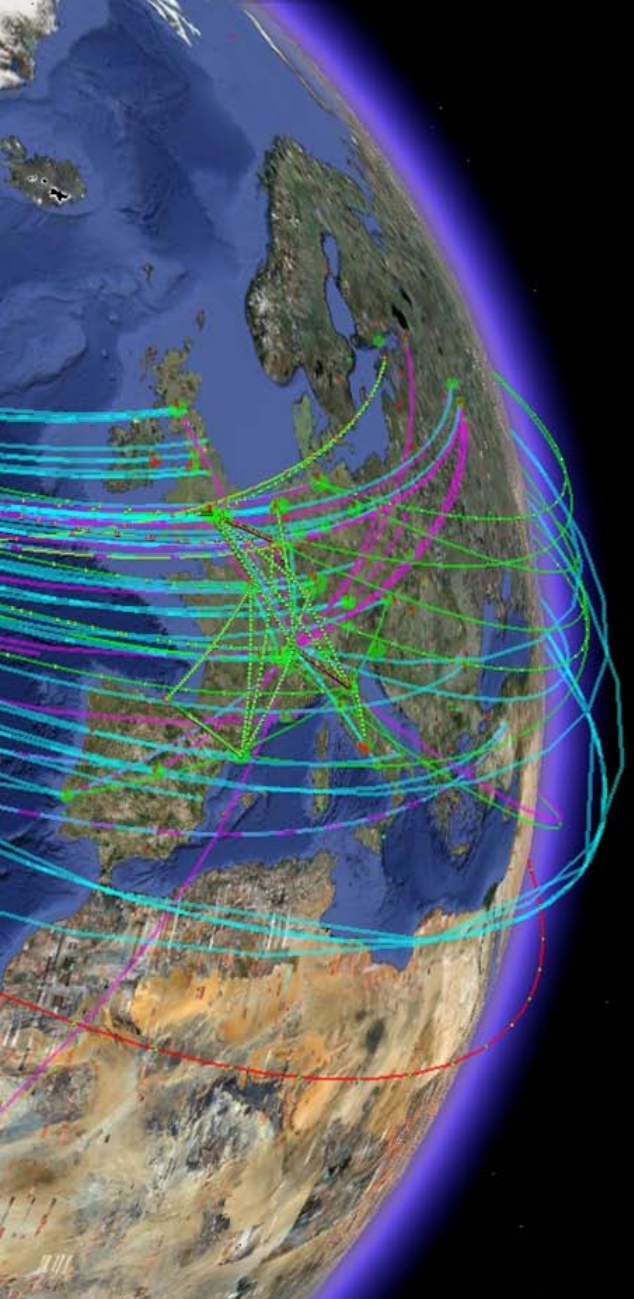
Lightweight Middleware for Grid Computing

gLite middleware



The Big Picture





Security

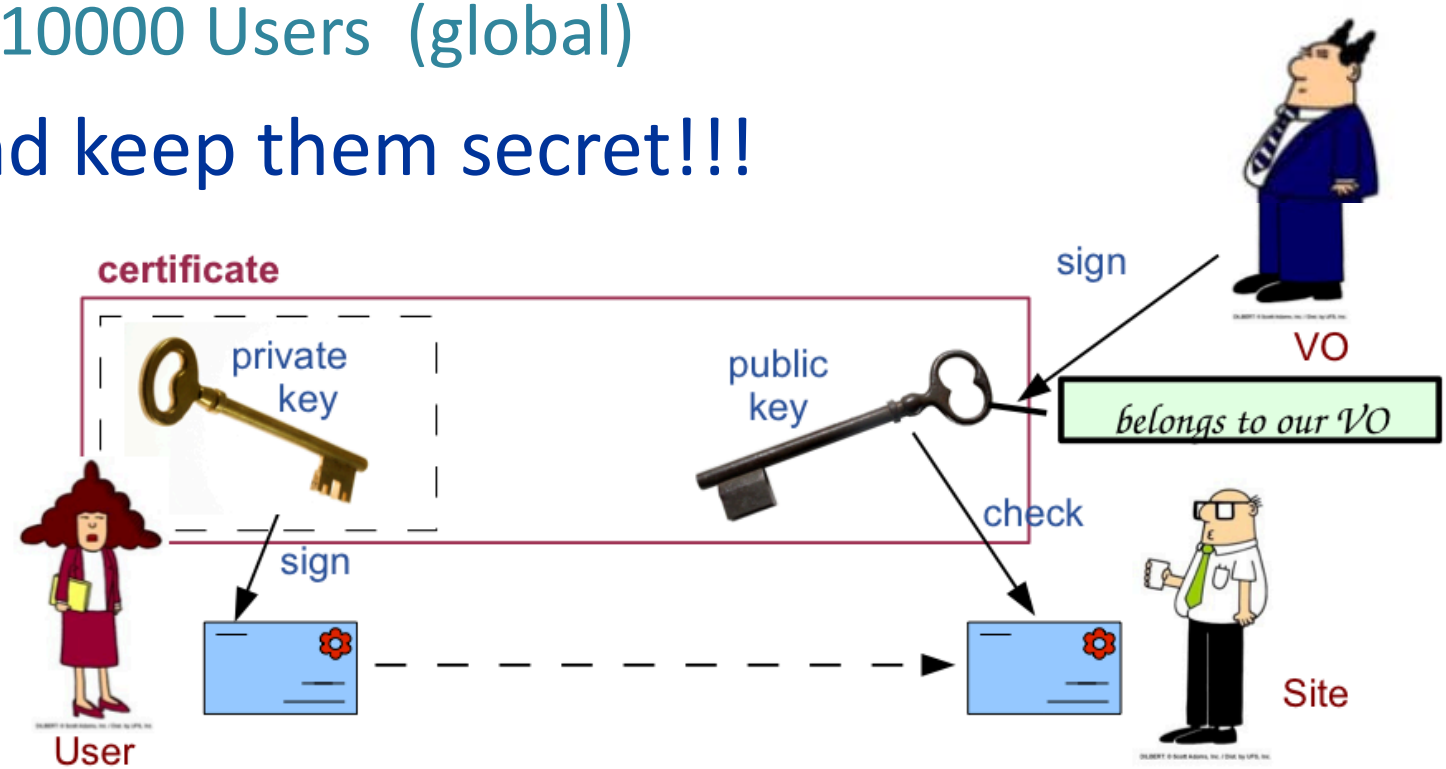
- Authentication
  - Who you are
- Authorization
  - What you can do
- Auditing/Accounting
  - What you have done
- How to establish trust?

Authentication

- Authentication is based on X.509 PKI infrastructure (Public Key)
  - Certificate Authorities (CA) issue (long lived) certificates identifying individuals (much like a passport)
    - Commonly used in web browsers to authenticate to sites
  - Trust between CAs and sites is established (offline)
  - In order to reduce vulnerability, on the Grid user identification is done by using (short lived) proxies of their certificates
- Short-Lived Credential Services (SLCS)
  - issue short lived certificates or proxies to its local users
    - e.g. from Kerberos or from Shibboleth credentials
- Proxies can
  - Be delegated to a service such that it can act on the user's behalf
  - Be stored in an external proxy store (MyProxy)
  - Be renewed (in case they are about to expire)
  - Include additional attributes -→ Authorization

Public Key Based Security

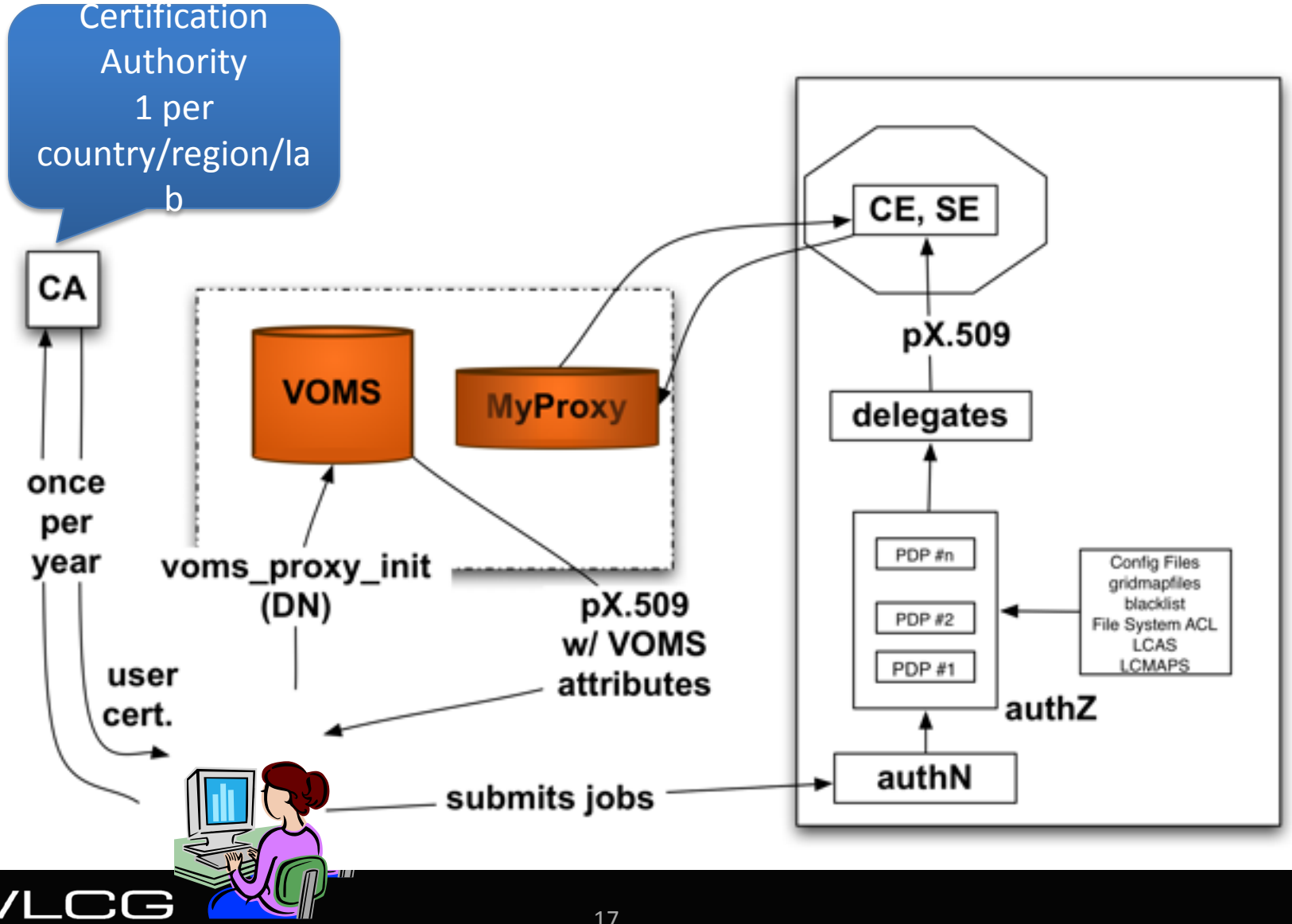
- How to exchange secret keys?
  - 340 Sites (global)
    - With hundreds of nodes each?
  - 200 User Communities (non local)
  - 10000 Users (global)
- And keep them secret!!!



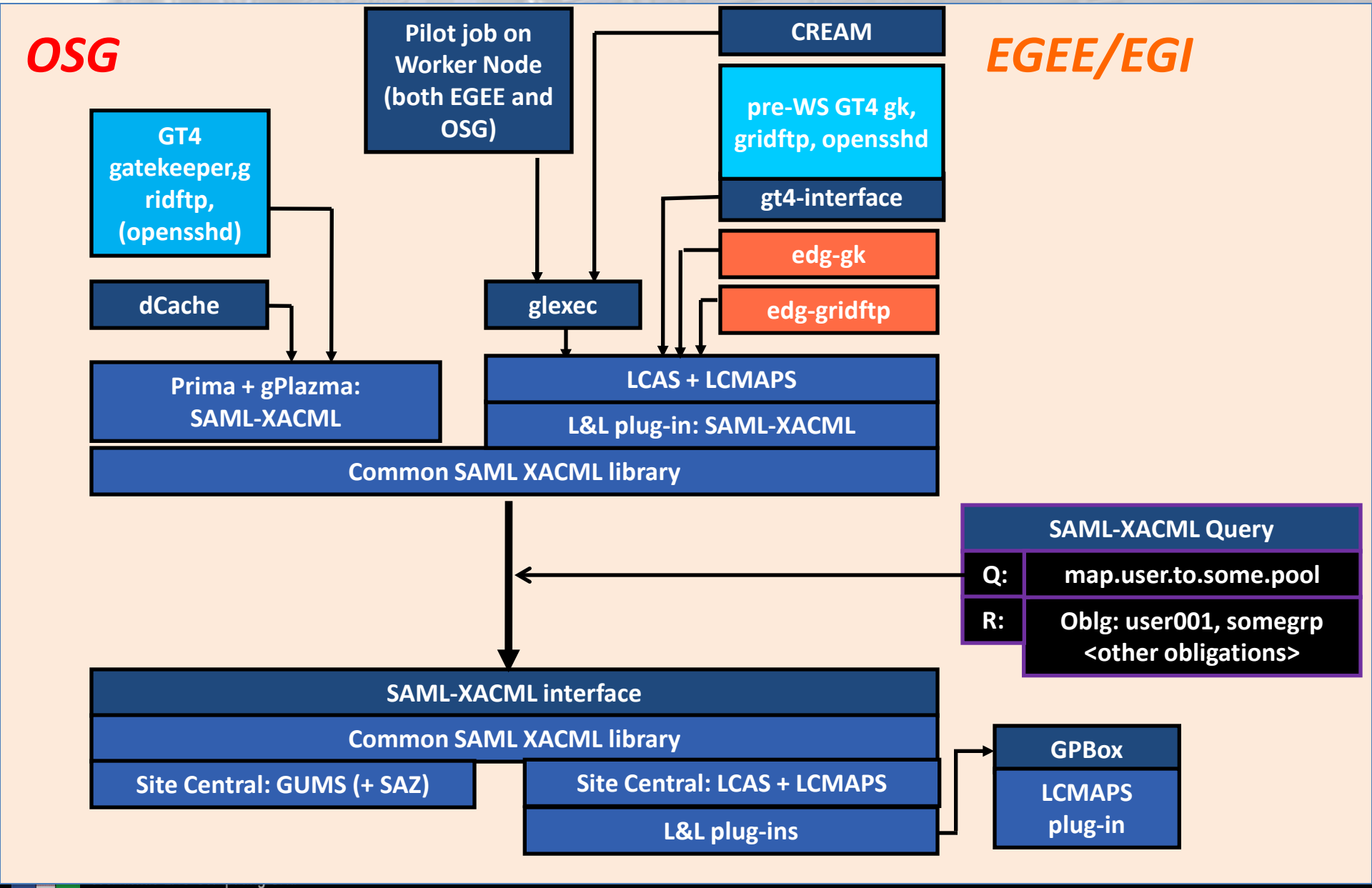
Authorization

- VOMS is now a de-facto standard
  - Attribute Certificates provide users with additional capabilities defined by the VO.
  - Allows group and role based authorization
  - Basis for the authorization process
- Authorization: currently via mapping to a local user on the resource (or ACLs)
  - glexec changes the local identity (based on suexec from Apache)
- Designing an authorization service with a common interface agreed with multiple partners
  - Uniform implementation of authorization in gLite services
  - Easier interoperability with other infrastructures
  - ARGUS

Security - overview



Common AuthZ interface

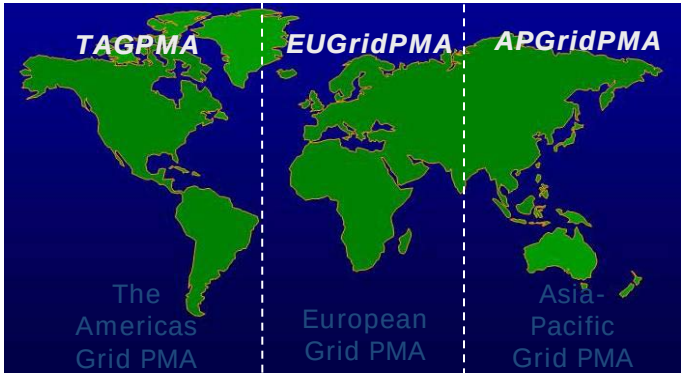


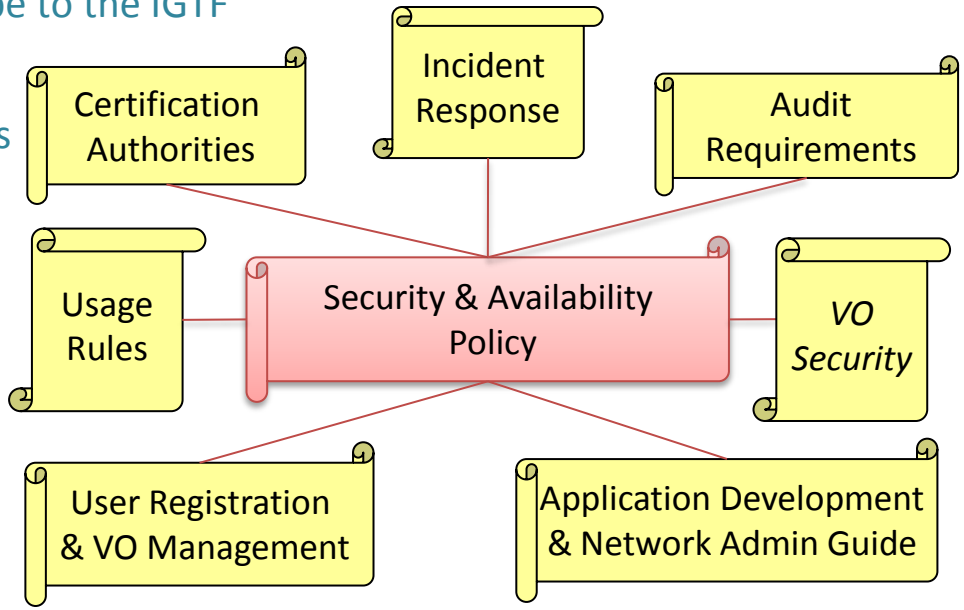
Trust

- You can setup a Certification Authority and several Registration Authorities
  - Using openssl
  - 1h work
- No one will trust your certificates
- Trust is based on:
  - Common policies
  - Common infrastructure to follow up on security problems

Security groups

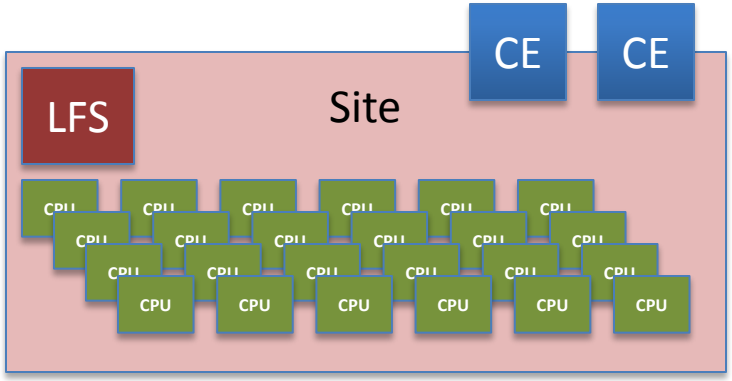
- Joint Security Policy Group:
  - Joint with WLCG, OSG, and others
  - Focus on policy issues
  - Strong input to e-IRG
- EUGridPMA
  - Pan-European trust federation of CAs
  - Included in IGTF (and was model for it)
  - Success: most grid projects now subscribe to the IGTF
- Grid Security Vulnerability Group
  - Looking at how to manage vulnerabilities
  - Risk analysis is fundamental
  - Balance between openness and security
- Operational Security Coordination Team
  - Main day-to-day operational work
  - Incident response and follow up
  - Members in all NGI and sites
  - Frequent tests (Security Challenges)



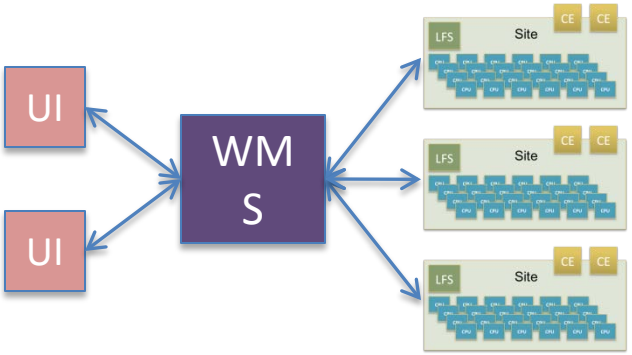


Computing Access

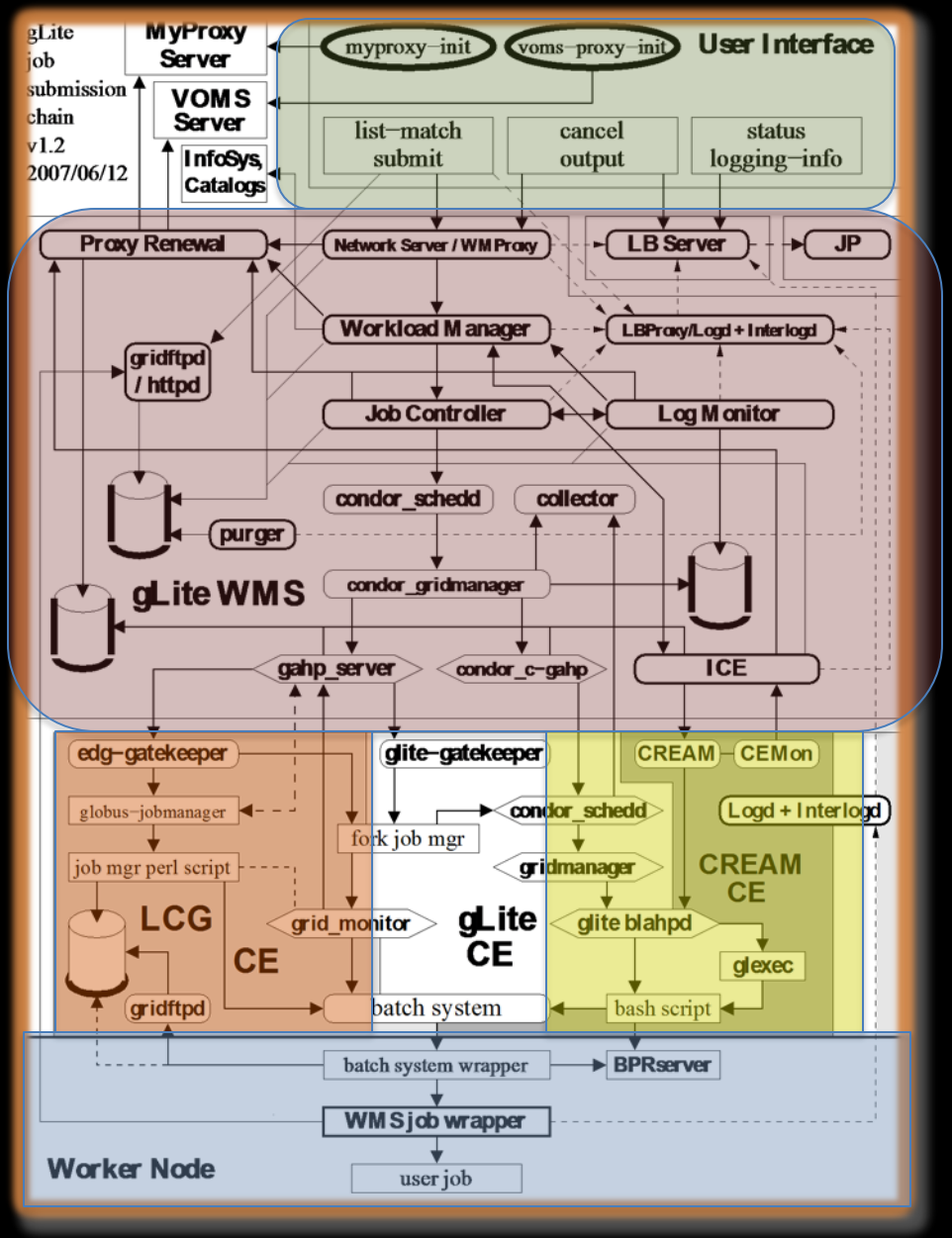
- Computing Elements (CE)
  - gateways to farms



- Workload Management
  - WMS/LB
  - Matches resources and requests
    - Including data location
  - Handles failures (resubmission)
  - Manages complex workflows
  - Tracks job status



Workload Management (compact)



Desktops

A few ~50 nodes

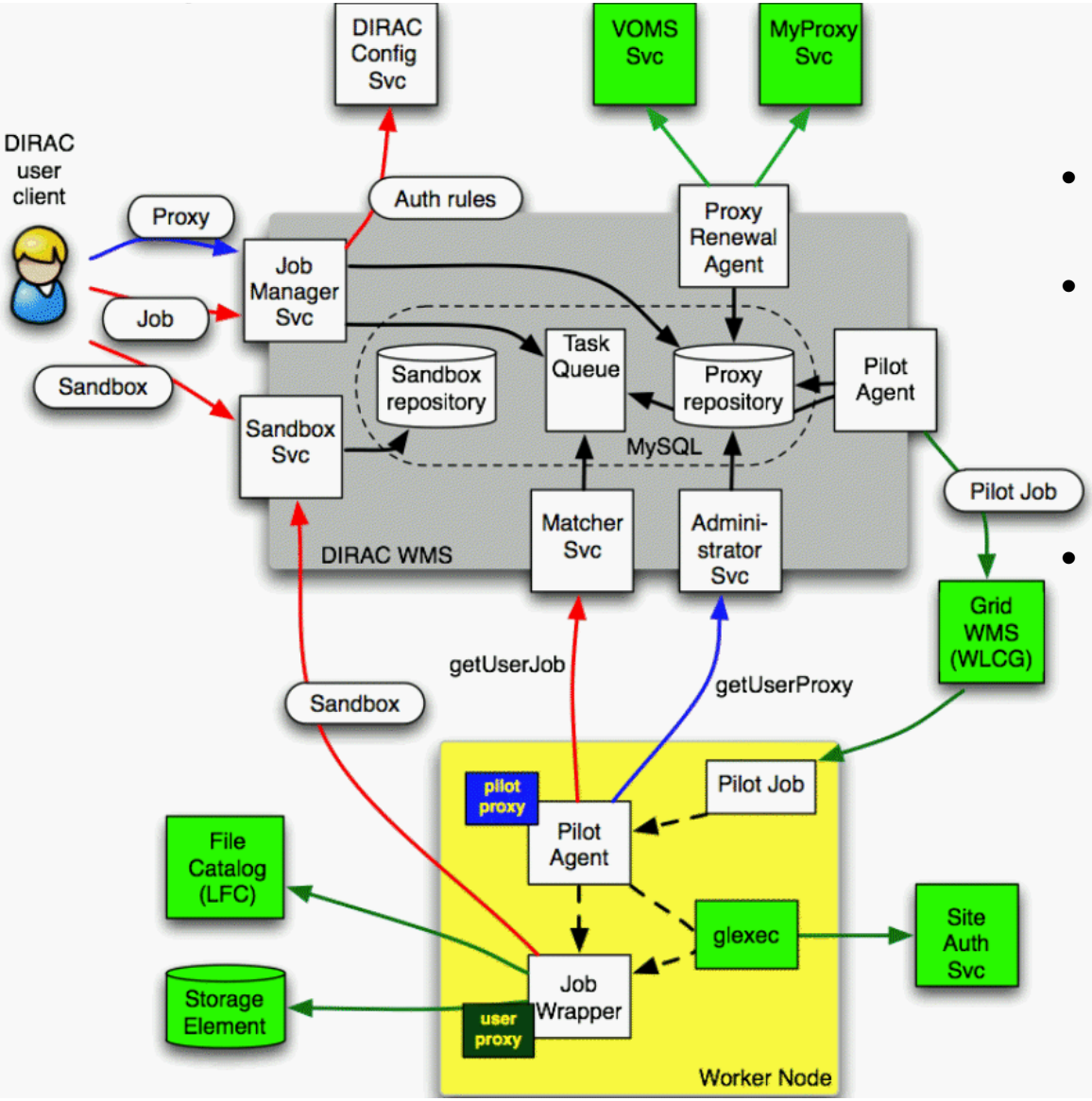
1-20 per site

1-24000 per site

Job Description Language

```
[
  Executable = “my_exe”;
  StdOutput = “out”;
  StdError = “err”;
  Arguments = “a b c”;
  InputSandbox = {“/home/giaco/my_exe”};
  OutputSandbox = {“out”, “err”};
  Requirements = Member(
    other.GlueHostApplicationSoftwareRunTimeEnvironment,
    "ALICE3.07.01“
  );
  Rank = -other.GlueCEStateEstimatedResponseTime;
  RetryCount = 3
]
```

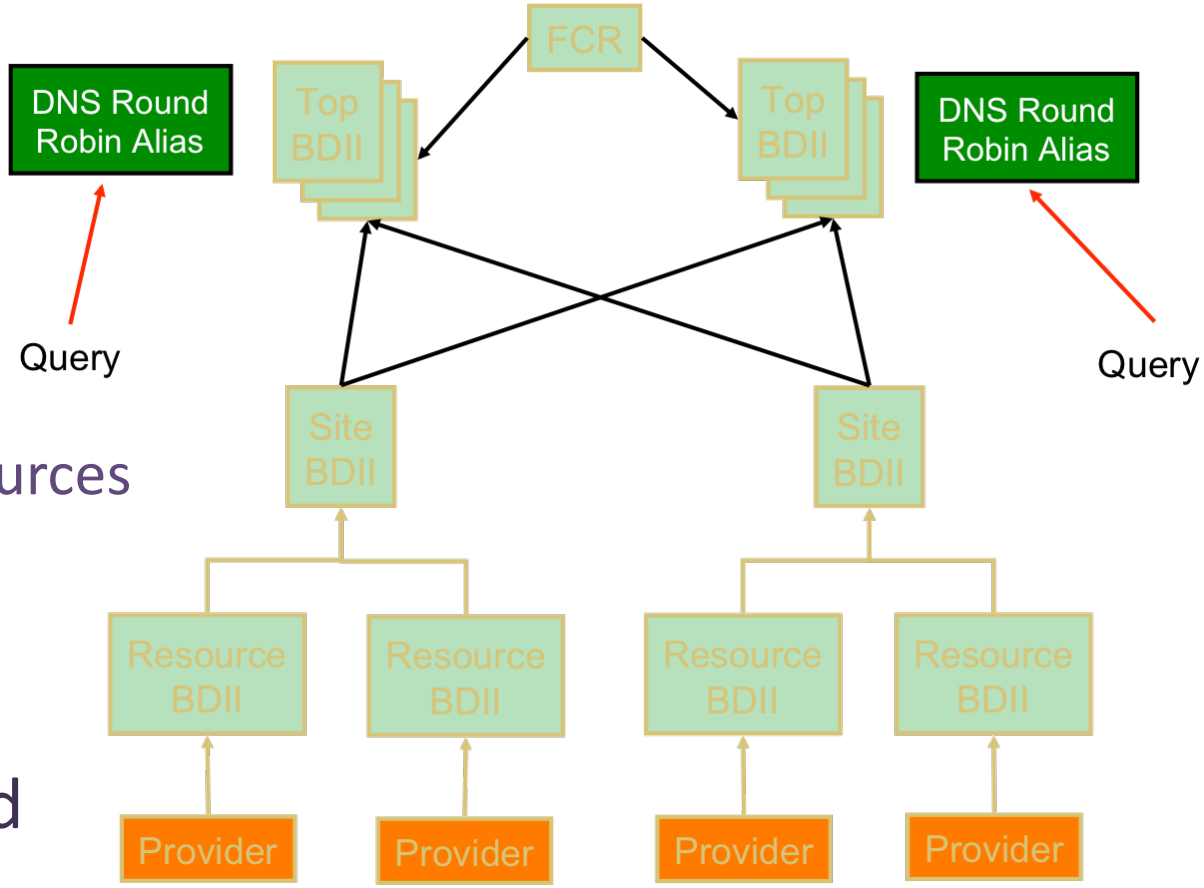
Pilot Jobs



- All WLCG experiments use a form of pilot jobs
- They have given a number of advantages
  - Responsive scheduling
  - Knowledge of environment
- They have also faced some resistance
  - Security
  - Traceability

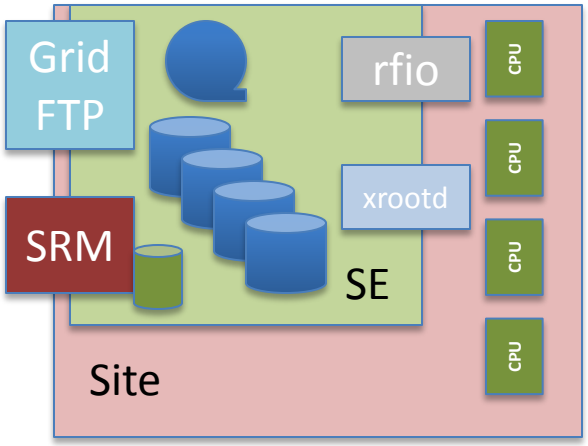
Information System

- BDII = Yellow Pages
  - realtime
- Light weight Database
- LDAP protocol
- GLUE 1.3 (2) Schema
  - Describes resources and their state
  - Approx 100MB
  - Update 2min
- Several hundred instances



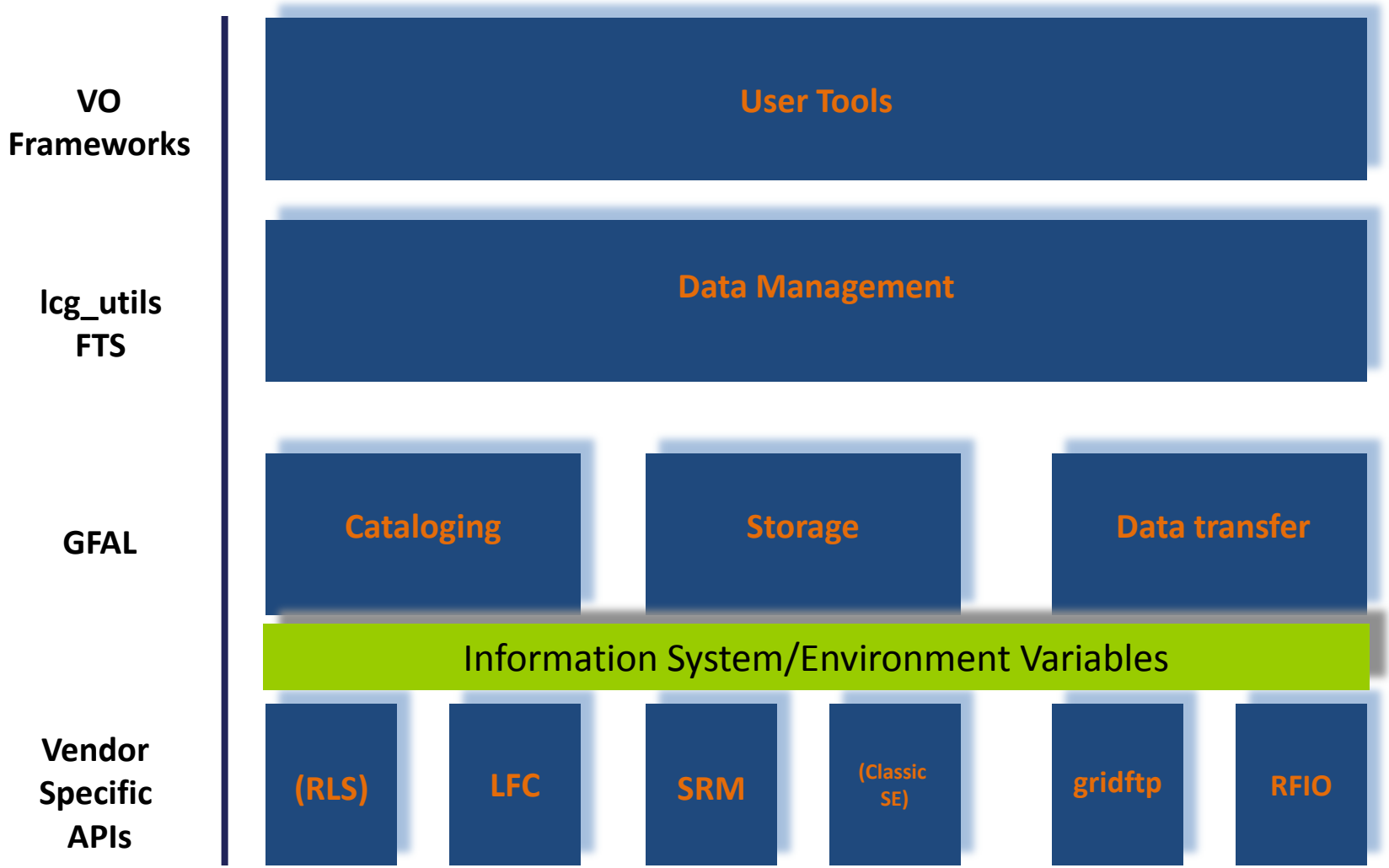
Data Management

- Storage Elements (SEs)
  - External interfaces based on SRM 2.2 and gridFTP
  - Many implementations:
    - CASTOR, Storm, DPM, dCache, BestMan….
  - Many local interfaces:
    - POSIX, dcap, secure rfio, rfio, xrootd



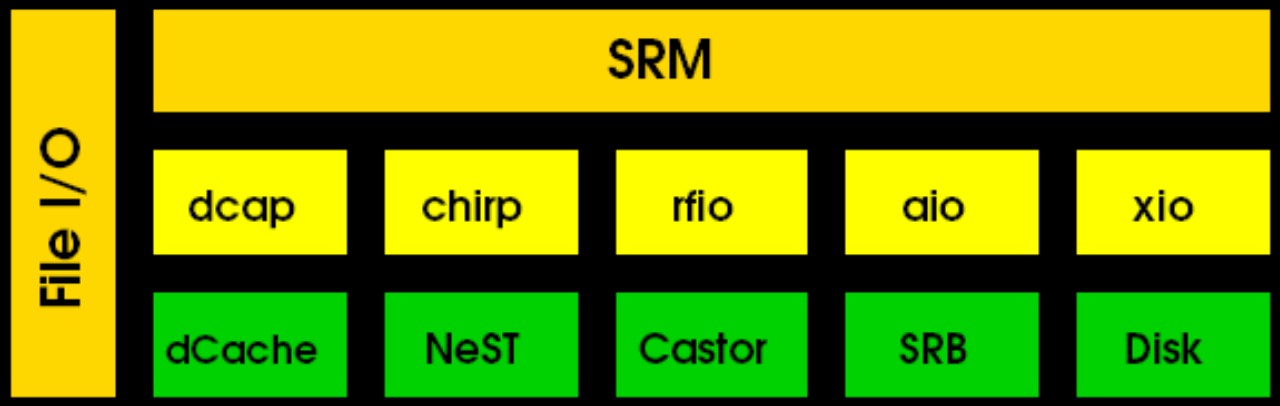
- Catalogue: LFC (local and global)
- File Transfer Service (FTS)
- Data management clients gfal/LCG-Utils

Data Management



General Storage Element

- Storage Resource Manager (SRM)
  - hides the storage system implementation (disk or tape)
  - handles authorization
  - translates SURLs (Storage URL) to TURLs (Transfer URLs)
  - disk-based: DPM, dCache,+; tape-based: Castor, dCache
  - Mostly asynchronous
- *File I/O*: *posix-like* access from local nodes or the grid
  - ➔ GFAL (Grid File Access Layer)



Approach to SRM

- An abstraction layer for storage and data access is necessary
  - Guiding principle:
  - Non-interference with local policies

- Providing all necessary user functionality and control
  - Data Management
  - Data Access
  - Storage management
  - Control:
    - Pinning files
    - Retention Policy
    - Space management and reservation
  - Data Transfers
- Grid enabled and based on current technology
  - Interface technology (gSOAP)
  - Security Model (gsi security)
  - To integrate with the grid infrastructure

SRM basic and use cases tests

Summary of S2 SRM v2.2 basic tests - Mozilla Firefox

File Edit View History Bookmarks Tools Help

Summary of S2 SRM v2.2 basic test - Thursday 30 August 2007 11:46pm CEST

SRM function	CERN C2	CNAF C2	CERN C2-1	BNL2 dCache	DESY dCache	UKED dCache	FZK dCache	IN2P3 dCache	NDGF dCache	SARA dCache	FNAL dCache	UCSD dCache	CERN DPM	UKED DPM	UKGL DPM	LAL DPM	LBNL BeStMan	CNAF StoRM	CNAF StoRM2	UKBR StoRM	IFIC StoRM
WLCG MoU SRM v2.2 methods																					
Ping	Out Log	Out Log	Out Log	Out Log	Out Log	Out Log	Out Log	Out Log	Out Log	Out Log	Out Log										
PtP	Out Log	Out Log	Out Log	Out Log	Out Log	Out Log	Out Log	Out Log	Out Log	Out Log	Out Log										
StOfPut	Out Log	Out Log	Out Log	Out Log	Out Log	Out Log	Out Log	Out Log	Out Log	Out Log	Out Log										
PutDone	Out Log	Out Log	Out Log	Out Log	Out Log	Out Log	Out Log	Out Log	Out Log	Out Log	Out Log										
PtG	Out Log	Out Log	Out Log	Out Log	Out Log	Out Log	Out Log	Out Log	Out Log	Out Log	Out Log										
StOfGet	Out Log	Out Log	Out Log	Out Log	Out Log	Out Log	Out Log	Out Log	Out Log	Out Log	Out Log										
BoL	Out Log	Out Log	Out Log	Out Log	Out Log	Out Log	Out Log	Out Log	Out Log	Out Log	Out Log										
StOfBoL	Out Log	Out Log	Out Log	Out Log	Out Log	Out Log	Out Log	Out Log	Out Log	Out Log	Out Log										
AbortR	Out Log	Out Log	Out Log	Out Log	Out Log	Out Log	Out Log	Out Log	Out Log	Out Log	Out Log										
AbortF	Out Log	Out Log	Out Log	Out Log	Out Log	Out Log	Out Log	Out Log	Out Log	Out Log	Out Log										
RelFiles	Out Log	Out Log	Out Log	Out Log	Out Log	Out Log	Out Log	Out Log	Out Log	Out Log	Out Log										
GetReqSum	Out Log	Out Log	Out Log	Out Log	Out Log	Out Log	Out Log	Out Log	Out Log	Out Log	Out Log										
GetReqToks	Out Log	Out Log	Out Log	Out Log	Out Log	Out Log	Out Log	Out Log	Out Log	Out Log	Out Log										
GetTrProts	Out Log	Out Log	Out Log	Out Log	Out Log	Out Log	Out Log	Out Log	Out Log	Out Log	Out Log										
Ls	Out	Out	Out	Out Log	Out Log	Out Log	Out Log	Out Log	Out Log	Out Log	Out Log										

Summary of S2 SRM v2.2 use-case tests - Mozilla Firefox

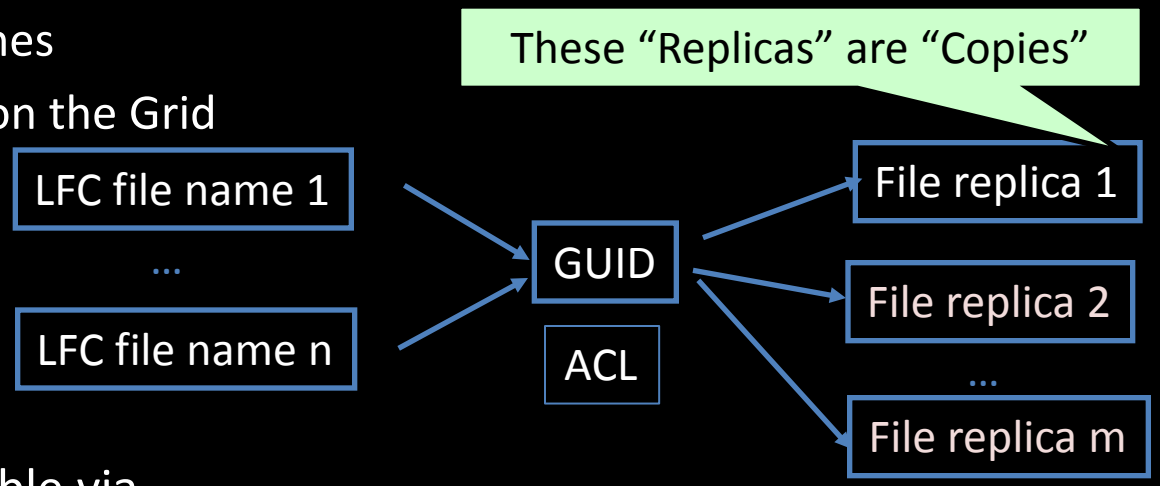
File Edit View History Bookmarks Tools Help

Summary of S2 SRM v2.2 use-case test - Thursday 30 August 2007 09:32pm CEST

SRM test	CERN C2	CNAF C2	CERN C2-1	BNL2 dCache	DESY dCache	UKED dCache	FZK dCache	IN2P3 dCache	NDGF dCache	SARA dCache	FNAL dCache	UCSD dCache	CERN DPM	UKED DPM	UKGL DPM	LAL DPM	LBNL BeStMan	CNAF StoRM	CNAF StoRM2	UKBR StoRM	IFIC StoRM
CheckGarbageSpaceCollector	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	Out
CheckNs	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer
ExtendFileLifeTime	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer
FileNames00	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer
FileNames01	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer
GetRemove01	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer
GetNewPartialFn	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer
GetNewPartialNo	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer
LsDnCommOffset	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer
LsDnDetail	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer
LsDnFull	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer
LsFullDetail	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer
LsNonExistent	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer
LsTopDn	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer
MMin00	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer
MMinExistent	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer
MvBeingPut	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer
MvDnBeingPutInfo1	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer
MvDnBeingPutInfo	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer
MvDn	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer
MvNewDn	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer
MvNewFile	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer
OverwritePn	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer
Pn01	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer
Pn02	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer
Pn03	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer
Pn07	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer
PrepareToPutFVFR	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer
PutGet01	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer
PutGet01File1	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer
PutGet01File	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer
PutNewLine	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer
PutNoOverwrite	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer
PutNoOverwriteIsNeg	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer
PutOverwrite	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer
PutOverwriteIsNeg	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer
PutRemove01	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer
PutRemove02	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer
PutRemove03	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer
PutNewPartialFn	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer	OnLer

LCG “File” Catalog

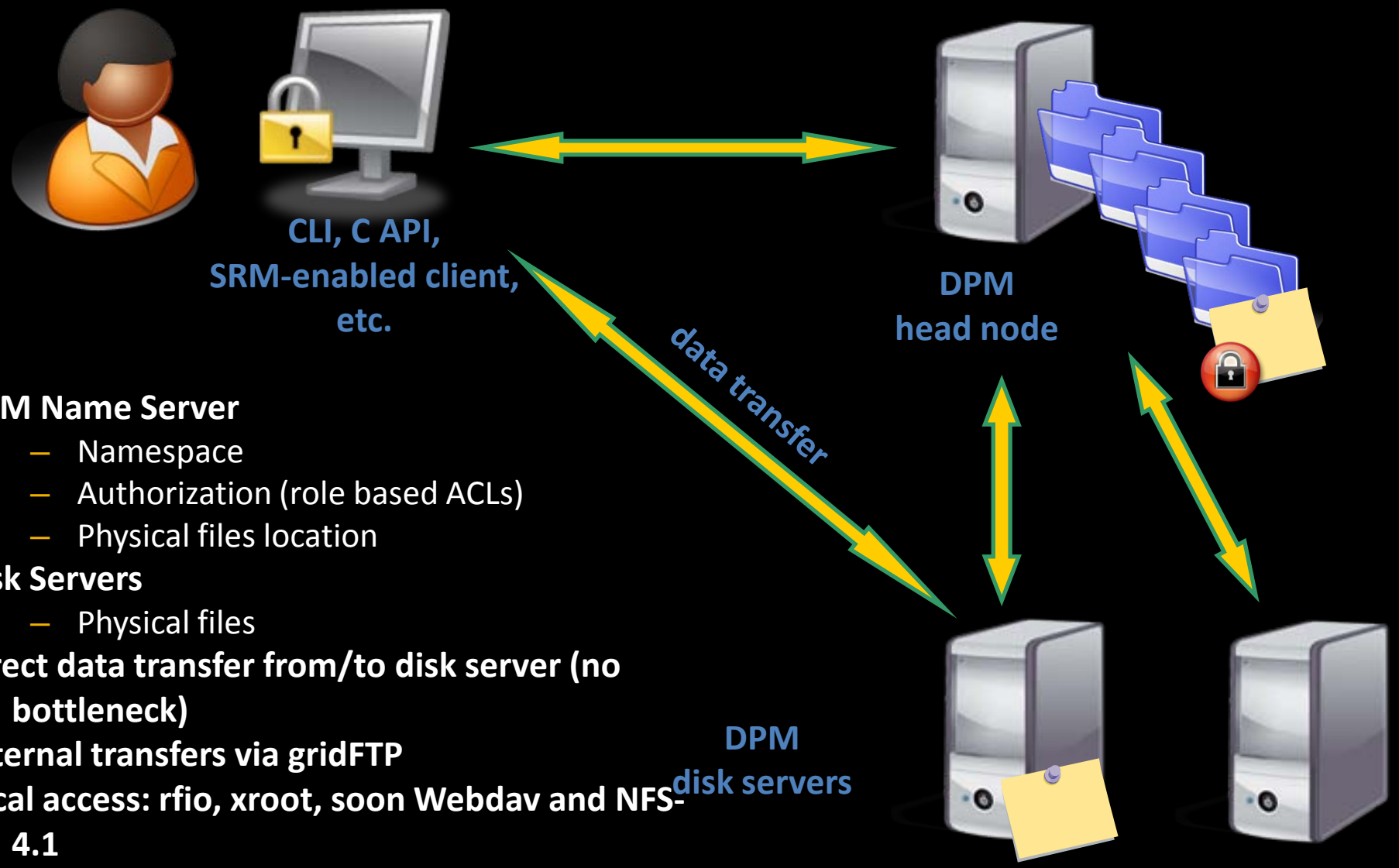
- The LFC stores mappings between
  - Users’ file names
  - File locations on the Grid



- The LFC is accessible via
  - CLI, C API, Python interface, Perl interface
    - Supports sessions and bulk operations
  - Data Location Interface (DLI)
    - Web Service used for match making:
      - given a GUID, returns physical file location
- ORACLE backend for high performance applications
  - Read-only replication support

All files are “Write Once Read Many”

DPM: user's point of view



DPM Name Server
- Namespace
- Authorization (role based ACLs)
- Physical files location

Disk Servers
- Physical files

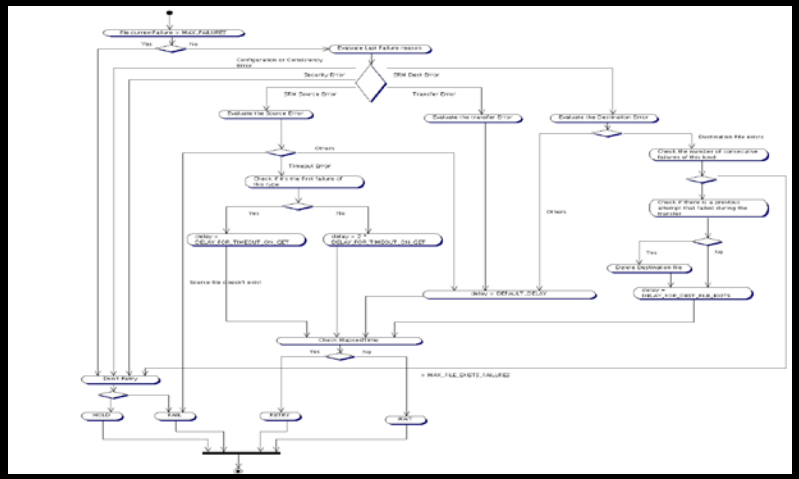
Direct data transfer from/to disk server (no bottleneck)

External transfers via gridFTP

Local access: rfio, xroot, soon Webdav and NFS-4.1

Easy to deploy, easy to operate

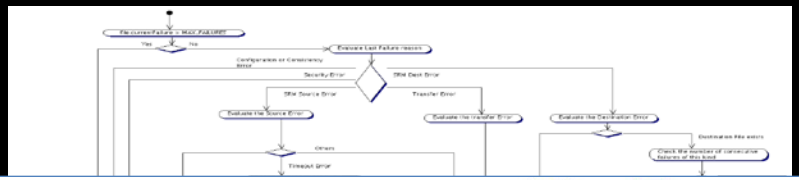
FTS: key points



- Scalable File Transfer service
- Reliability
  - It handles the retries in case of storage / network failures
    - VO customizable retry logic
  - Service designed for high-availability deployment
- Security
  - All data is transferred securely using delegated credentials with SRM / gridFTP
  - Service audits all user / admin operations
- Service and performance
  - Service stability: it is designed to efficiently use the available storage and network resources without overloading them
  - Service recovery: integration of monitoring to detect service-level degradation

FTS: key points

- Scalable File Transfer service
- Reliability
  - It handles the retries in cas storage / network failures
    - VO customizable retry logic
  - Service designed for high-availability deployme
- Security
  - All data is transferred secu gridFTP
  - Service audits all user / ad
- Service and performance
  - Service stability: it is desig and network resources wit
  - Service recovery: integratio degradation



FTS Report

Disclaimer
This page contains a report generated from information stored in the FTS Database and is intended for reporting purposes only. Since the format will probably change in the future, it's therefore recommended not to use parsing robots on it.

Statistics concerning all the transfers performed yesterday
Between 2006-10-12 08:00:00 +02:00 and 2006-10-13 08:00:00 +02:00

CERN-.* Filter Show VO details

Channel Name	VO Name	Total	% Failures	# Succ.	# Fail.	1st Failure Reason	% 1st Failure Reason	2nd Failure Reason	% 2nd Failure Reason	Avg. Size (GB)	Avg. Duration (sec)	Avg. Tx Rate (MB/sec)	Eff. Tx Bytes (GB)	Tx Bytes (GB)
CERN-PIC	[All]	12262	73.97	3192	9070	Dest SRM	56.22	Other	37.53	0.53	263.03	1.62	1700.41	1700.41
	atlas	8932	99.92	7	8925	Dest SRM	57.13	Other	38.08	0	220	0	0	0
	cms	208	0	208	0					2.7	767.55	3.64	561.26	561.26
	dteam	974	0.51	969	5	Other	80	Source SRM	20	0.95	356.31	2.88	923.83	923.83
	lhcb	2145	6.53	2005	140	Source SRM	99.29	Other	0.71	0.11	165.85	0.81	215.32	215.32
	ops	3	0	3	0					0	202.67	0	0	0
CERN-RAL	[All]	8699	59.26	3544	5155	Other	84.91	Source SRM	14.88	0.85	478.22	2.59	3026.81	3027.57
	alice	1155	82.6	201	954	Other	99.58	Dest SRM	0.31	1.86	1805.05	1.11	372.95	372.95
	atlas	4512	88.52	518	3994	Other	84.85	Source SRM	15.15	1.79	1428.94	1.49	926.26	926.57
	cms	227	3.08	220	7	Dest SRM	85.71	Source SRM	14.29	2.53	348.65	10.08	555.61	555.61
	dteam	1077	3.99	1034	43	Other	86.05	Source SRM	9.3	0.95	276.64	4.01	980.47	980.91
	lhcb	1725	9.1	1568	157	Source SRM	99.36	Other	0.64	0.12	146.03	1.16	191.52	191.52
	ops	3	0	3	0					0	27	0.01	0	0
CERN-SARA	[All]	8792	42.55	5051	3741	Dest SRM	83.77	Source SRM	12.22	1.34	108.02	15.4	6777.95	6784.92
	alice	3134	15.12	2660	474	Source SRM	57.17	Dest SRM	41.14	1.66	109.53	18.43	4426.44	4430.29
	atlas	2018	53.32	942	1076	Dest SRM	72.4	Source SRM	16.54	1.15	144.44	9.42	1085.07	1087.6
	dteam	3488	61.32	1349	2139	Dest SRM	98.74	Other	0.98	0.93	81.91	14.66	1260.74	1261.32
	lhcb	148	35.14	96	52	Dest SRM	92.31	Other	3.85	0.06	76.1	0.93	5.7	5.7
	ops	4	0	4	0					0	97.25	0.02	0	0
CERN-INFN	[All]	11492	42.31	6630	4862	Dest SRM	43.85	Other	37.7	1.13	395.77	3.21	7514.29	7614.84
CERN-CERN	[All]	1536	39.71	926	610	Source SRM	58.36	Dest SRM	15.9	0.07	287.71	0.38	67.89	69.08
CERN-ASCC	[All]	6851	23.54	5238	1613	Source SRM	50.84	Other	28.89	1.14	1098.6	1.08	5955.81	6080.58
CERN-GRIDKA	[All]	12755	21.38	10028	2727	Source SRM	64.36	Other	32.53	0.87	371.97	3.19	8762.02	8767.53
CERN-TRIUMF	[All]	2244	20.63	1781	463	Other	61.77	Source SRM	31.1	1.04	395.15	3.63	1847.25	1917.13
CERN-BNL	[All]	13975	19.42	11261	2714	Source SRM	69.97	Other	24.24	0.44	190.38	3.41	4951.59	4960.34
CERN-IN2P3	[All]	11697	13.76	10087	1610	Source SRM	48.57	Other	47.45	1.22	296.21	5.33	12329.63	12329.63
CERN-FNAL	[All]	917	4.58	875	42	Transfer	97.62	Other	2.38	0	379.88	0	0	0

Click on the Channel Name to show the VO details

Other Software in gLite

- Encrypted Data Storage
  - DICOM SE, HYDRA (distributed key store)
- Several grid enabled storage systems
- Meta Data Catalogues
  - AMGA
- Logging and Bookkeeping
  - Doing exactly this
- Accounting
  - APEL, DGAS
- ARGUS
  - Global/local authorization and policy system

gLite code base

Total Physical Source Lines of Code (SLOC)

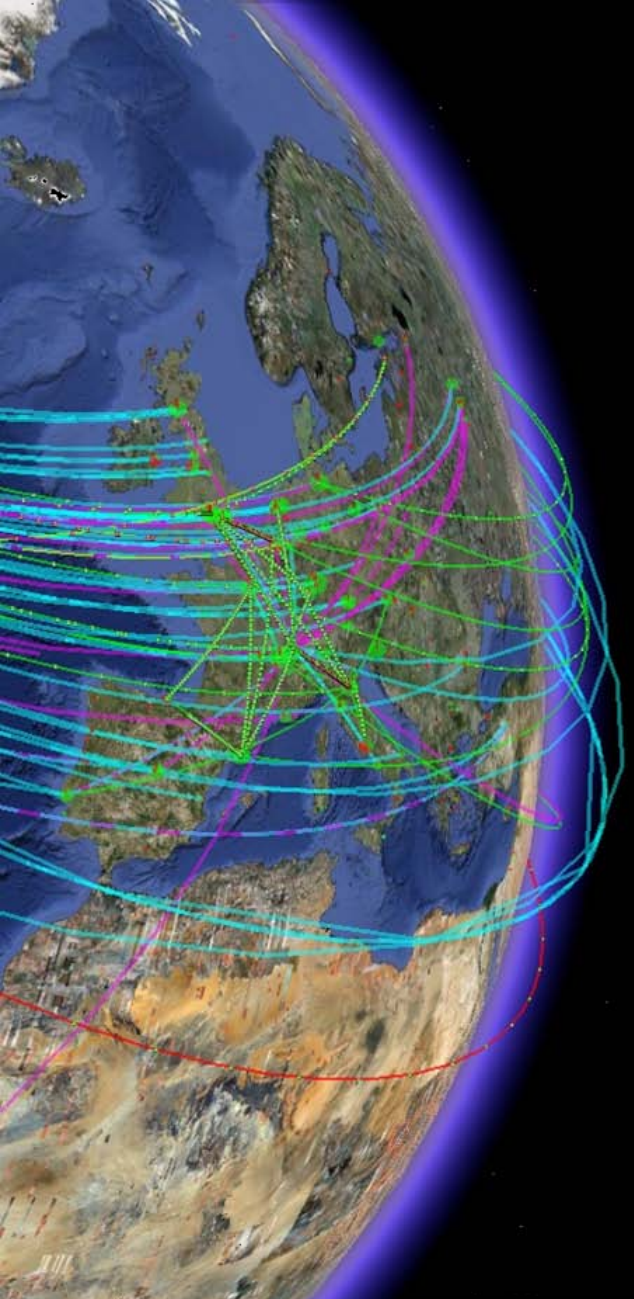
SLOC = 1622714

Total SLOC grouped by language (dominant language first)

Language	Total SLOC
ansic	578598 (35%)
cpp	491801 (30%)
java	251382 (15%)
sh	191798 (11%)
python	54510 (3%)
perl	39258 (2%)
yacc	7445 (0%)
jsp	4444 (0%)
lex	2274 (0%)
csh	701 (0%)
awk	307 (0%)
fortran	124 (0%)
sed	68 (0%)
asm	4 (0%)

- **Distributed under an open source license.**
- **Main platform is Scientifc Linux (recompiled RH EL).**
- **Many 3rd party dependencies**
  - tomcat, log4*,gSOAP , ldap etc.

- **~ 20 FTEs, 80 people, 12 institutes (mostly academic)**
- **Geographically distributed, independent**
  - Coding conventions, Documentation, Naming Conventions
  - Testing and quality, dependency management



Summary Middleware

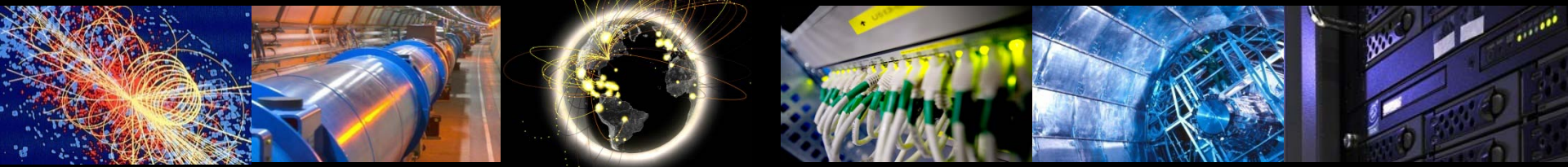
- Middleware allows to:
  - Find resources
  - Access Computing and Storage
  - Mange workflows
  - Move data
  - Locate data
- Provides,
  - without central control
  - Security and accounting

CERN-IT-GT

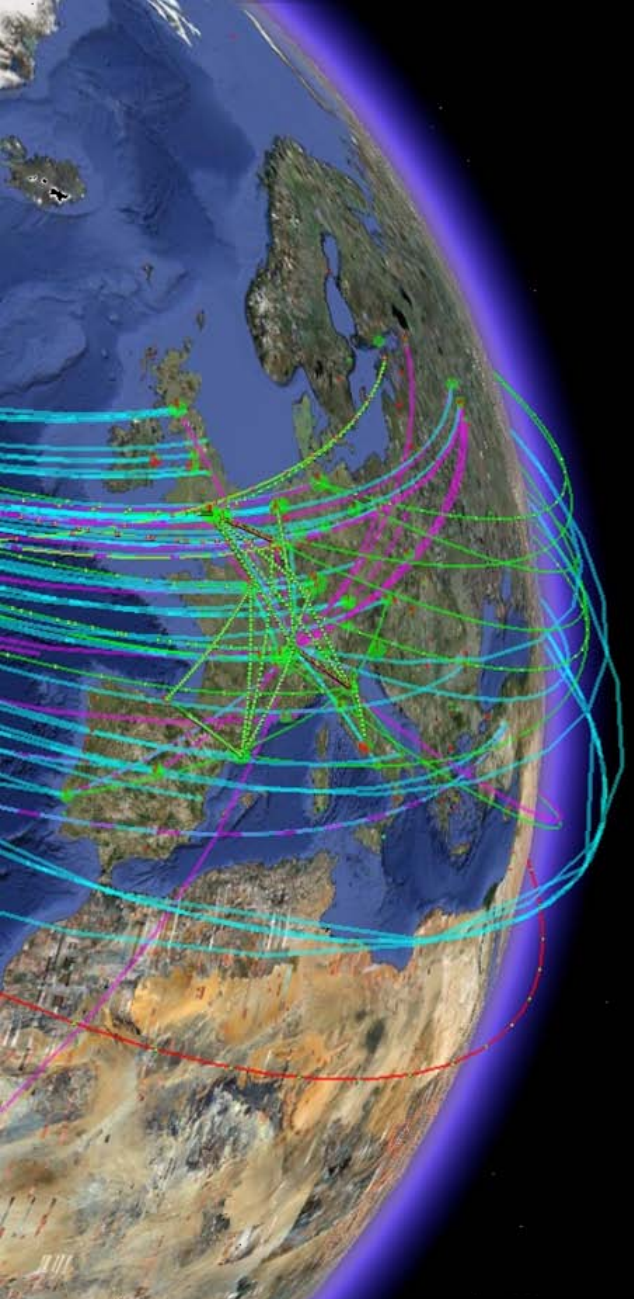
WLCG
Worldwide LHC Computing Grid

Markus Schulz

August 2010
Openlab Summer Students



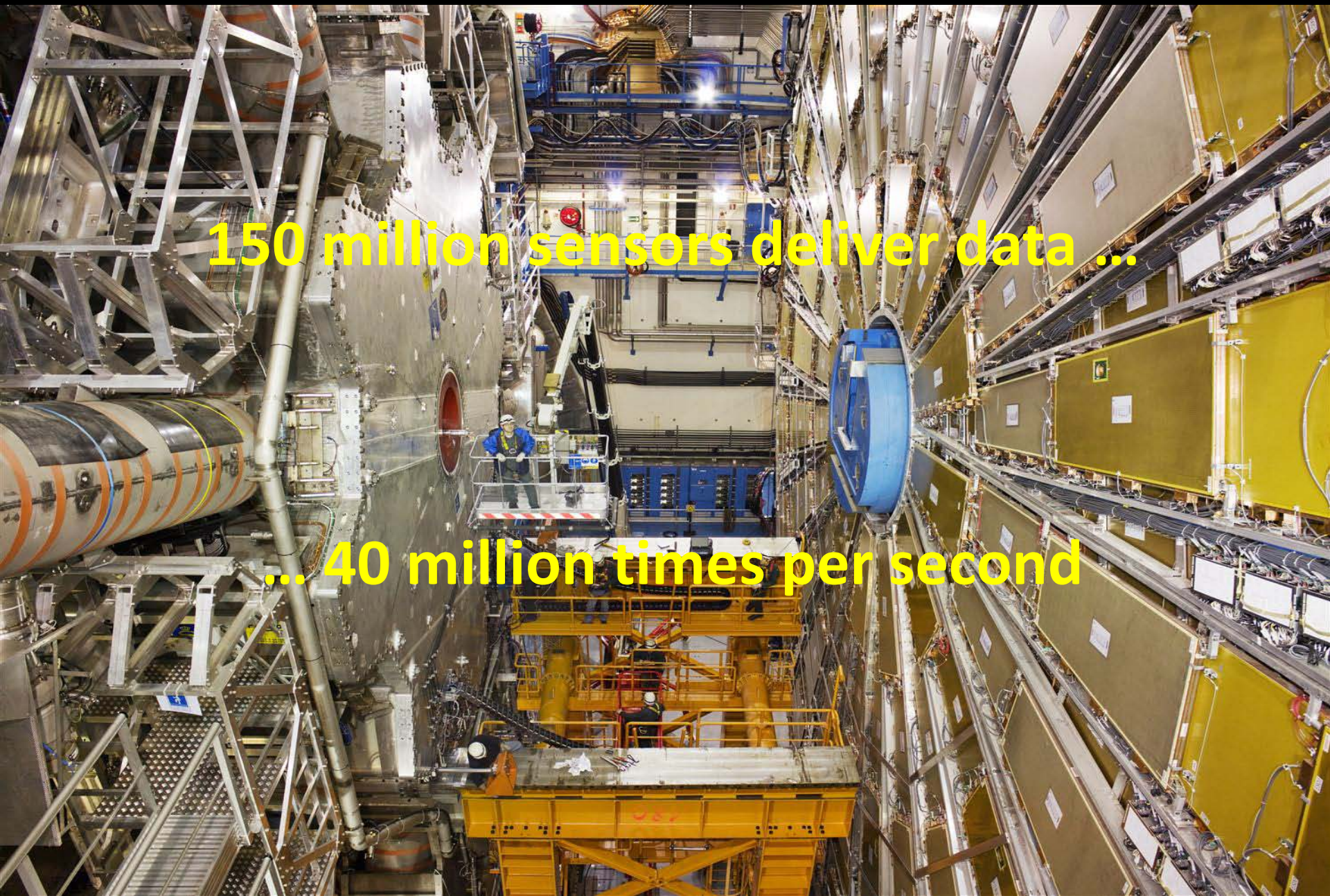
WLCG
Worldwide LHC Computing Grid



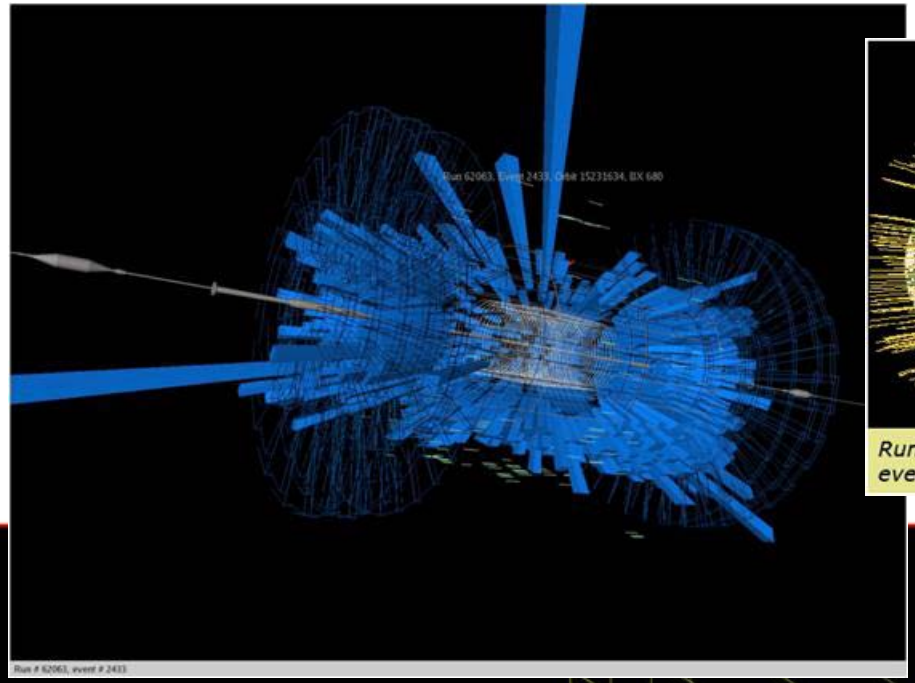
Overview

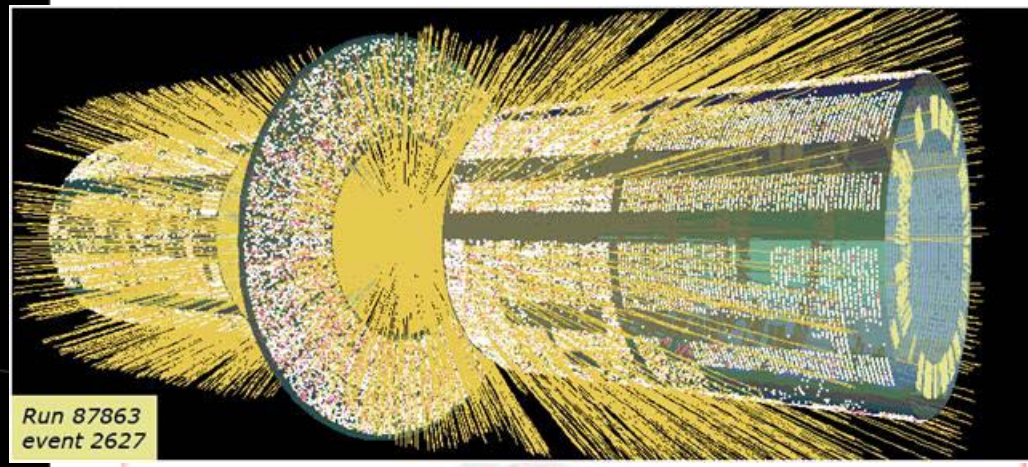
- An example for large scale scientific grid computing
- The LHC challenge
- Why grid computing?
- First full year with data

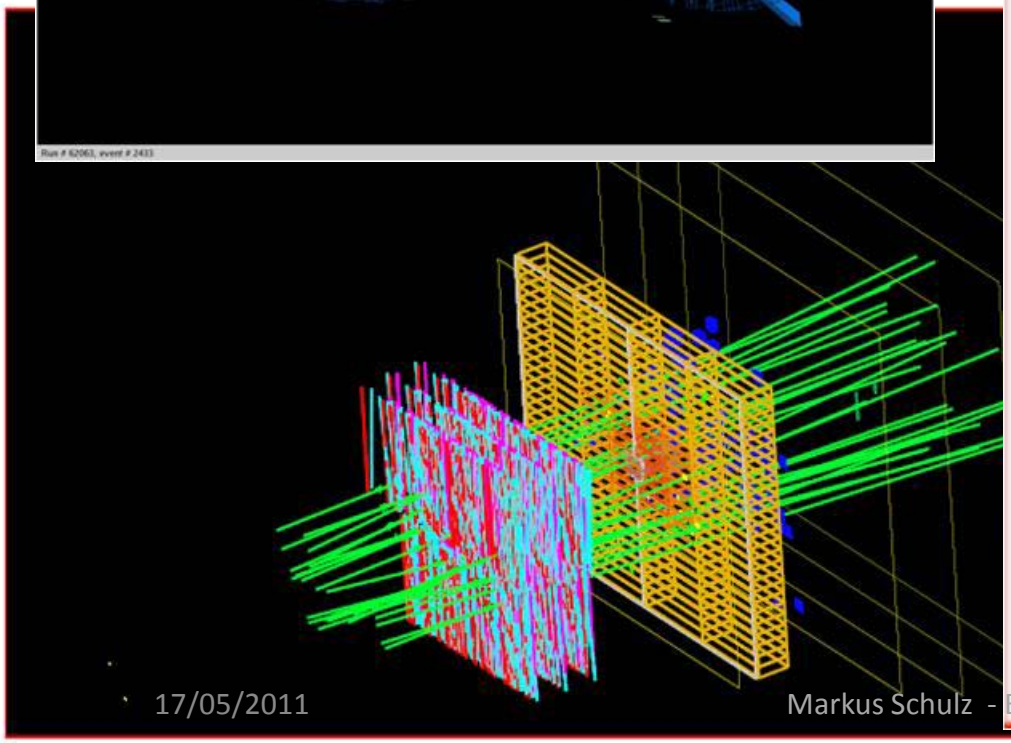
One of our data generators: ATLAS

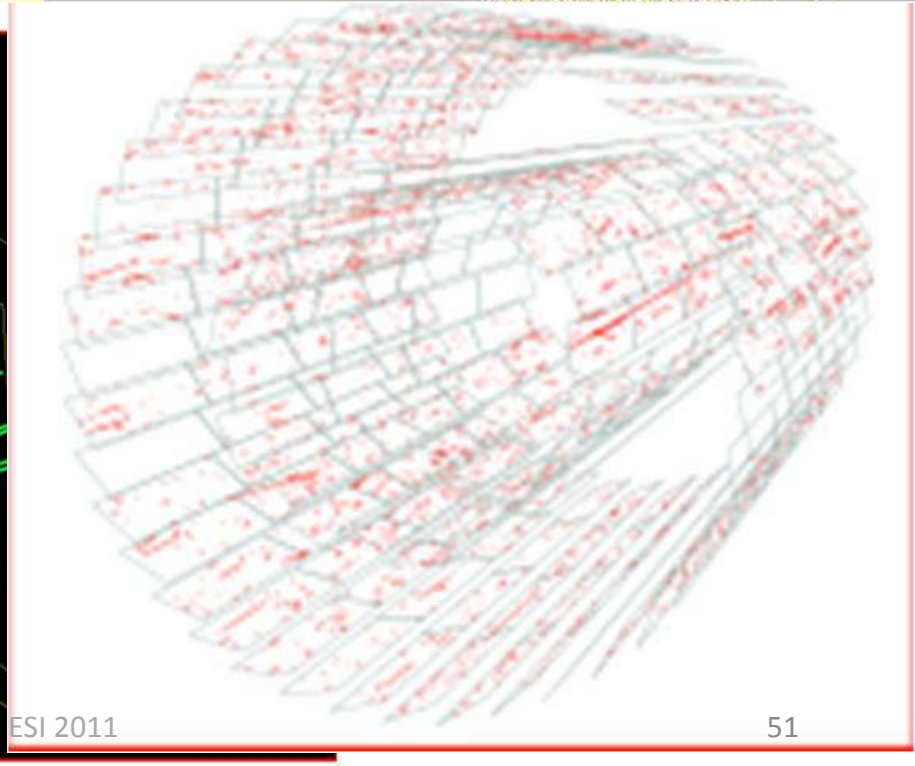


First events



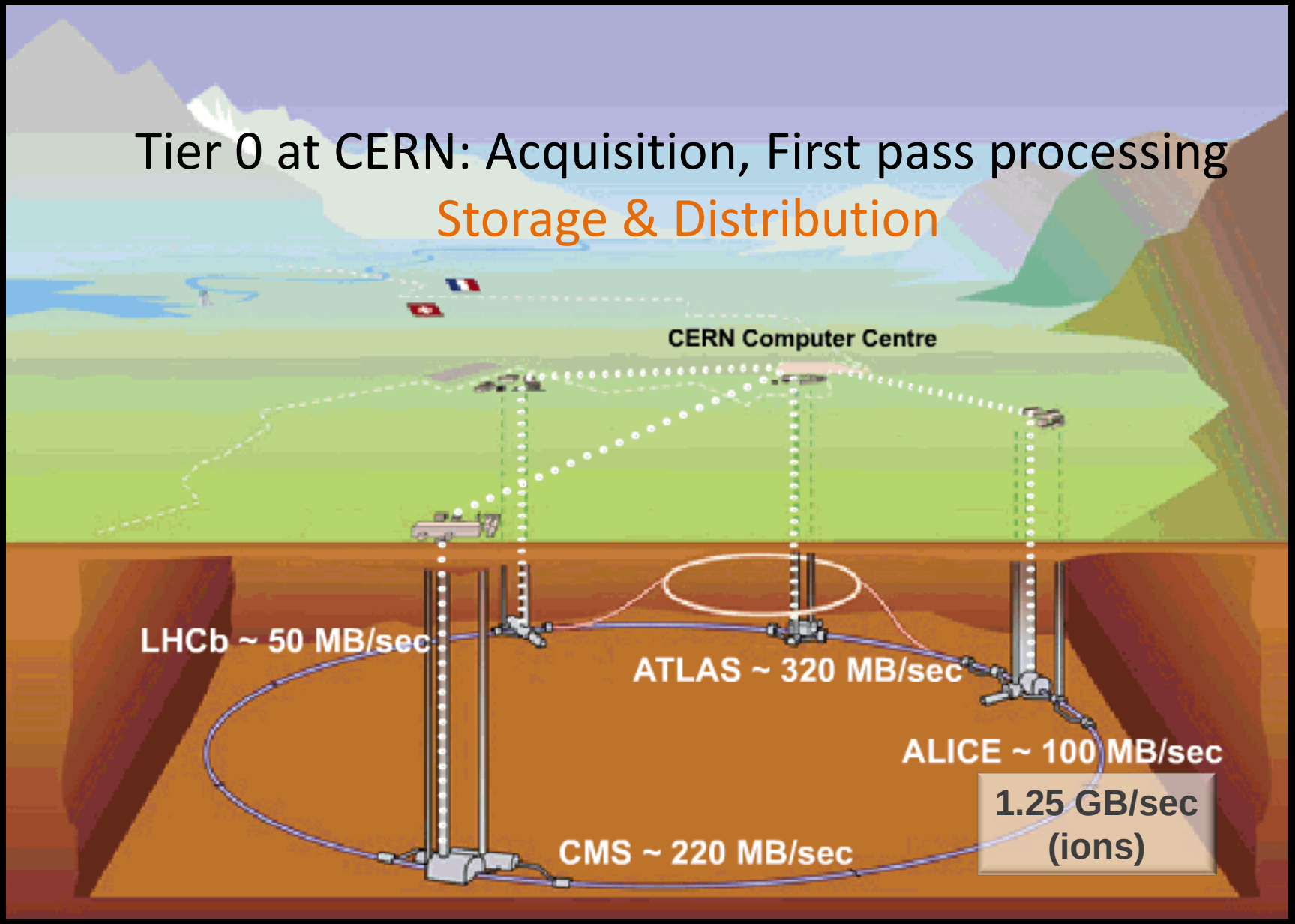




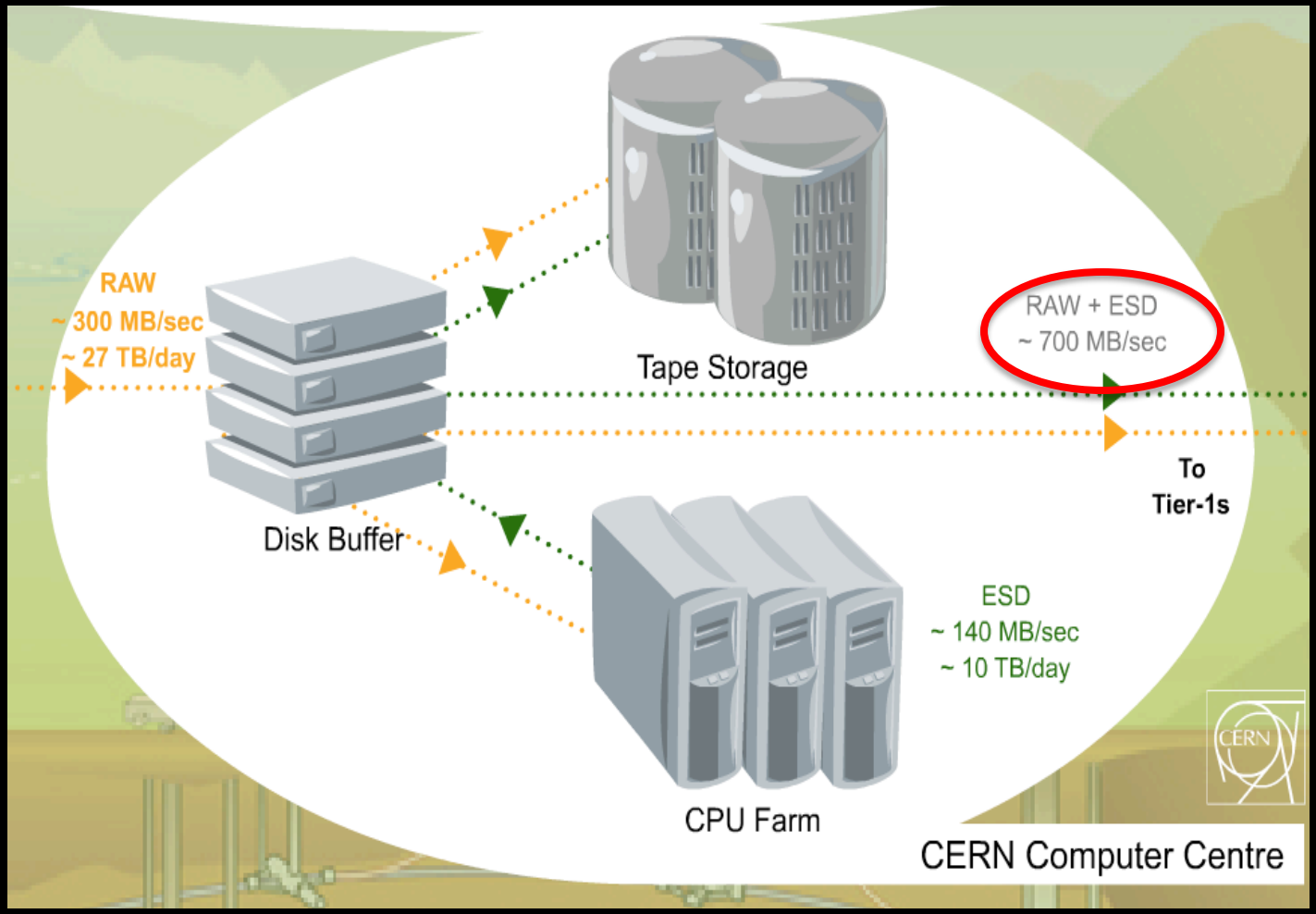


Tier 0 at CERN: Acquisition, First pass processing

Storage & Distribution

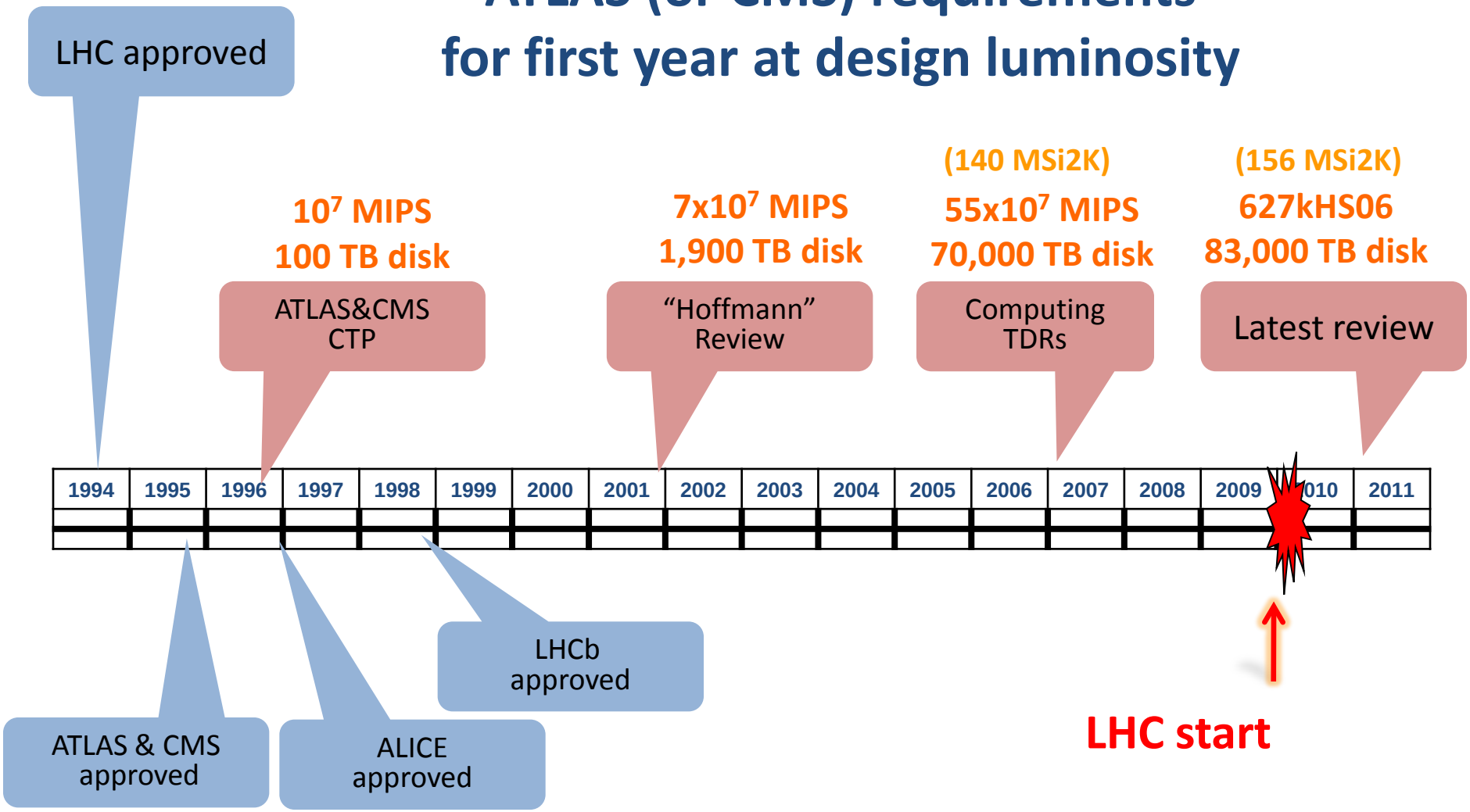


Flow in and out of the center



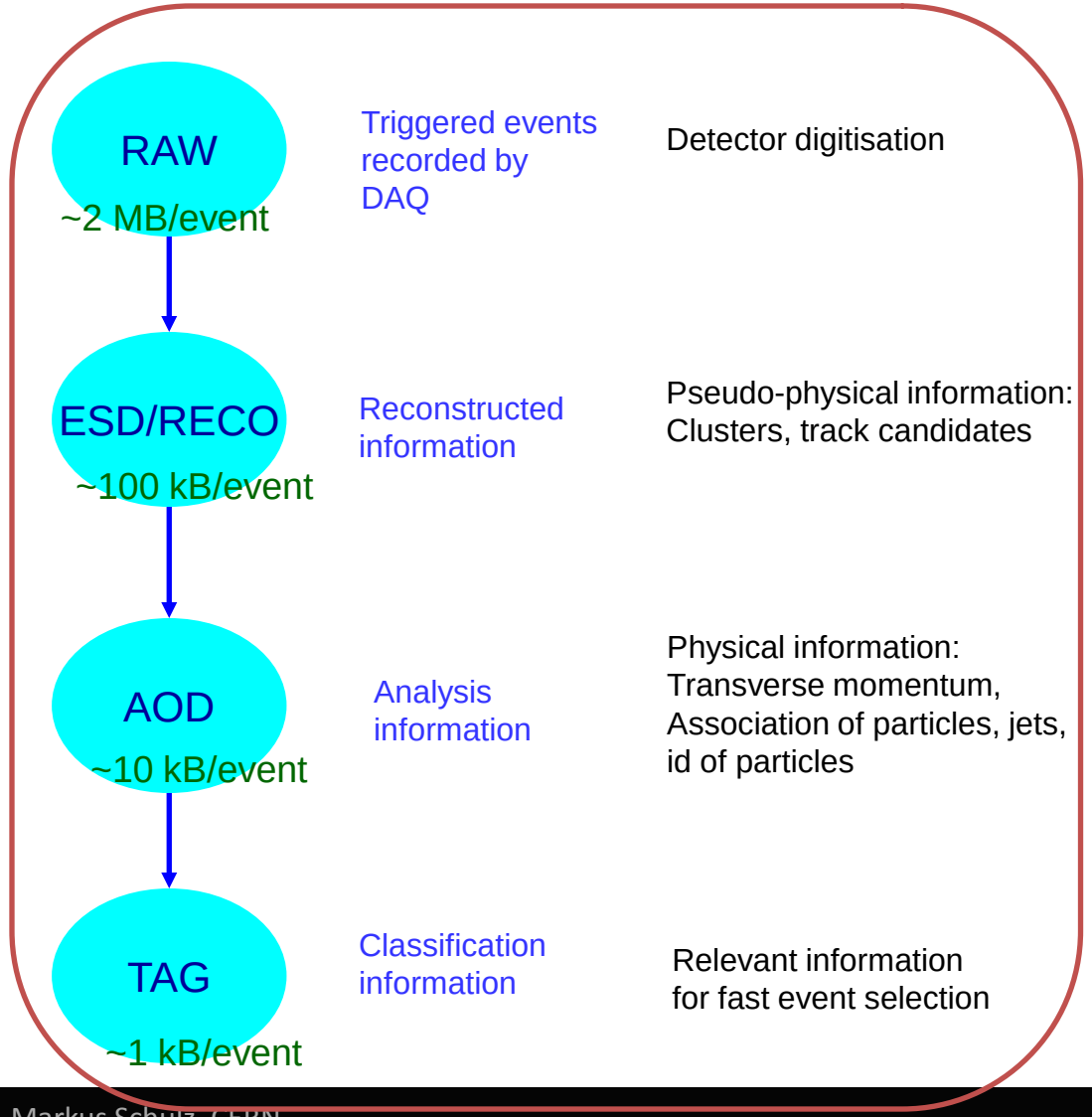
Offline Requirements

ATLAS (or CMS) requirements for first year at design luminosity



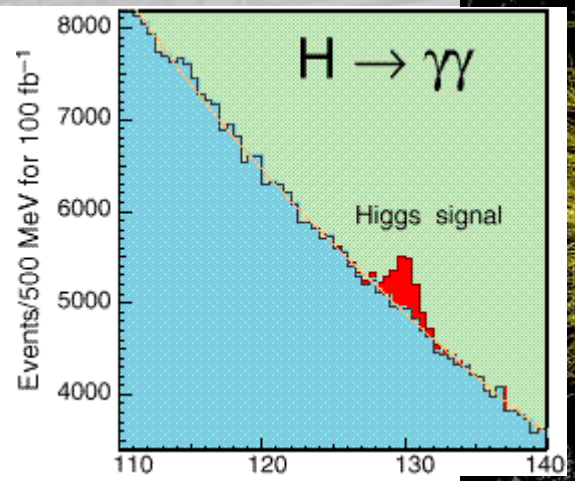
Data and Algorithms

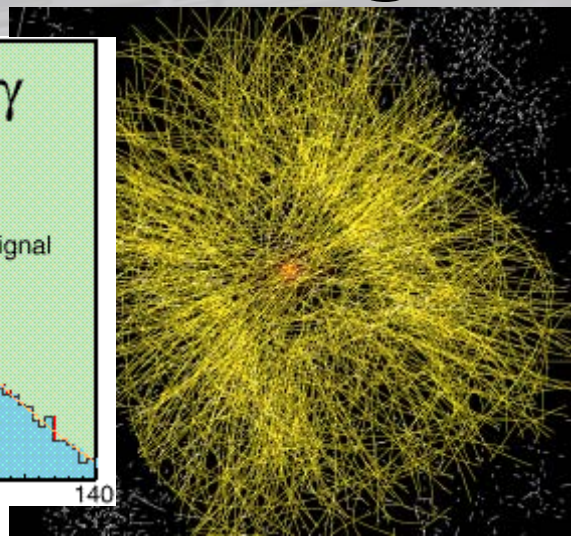
- HEP data are organized as *Events* (particle collisions)
- Simulation, Reconstruction and Analysis programs process "one Event at a time"
  - Events are fairly independent → Trivial parallel processing
- Event processing programs are composed of a number of Algorithms selecting and transforming "raw" Event data into "processed" (reconstructed) Event data and statistics

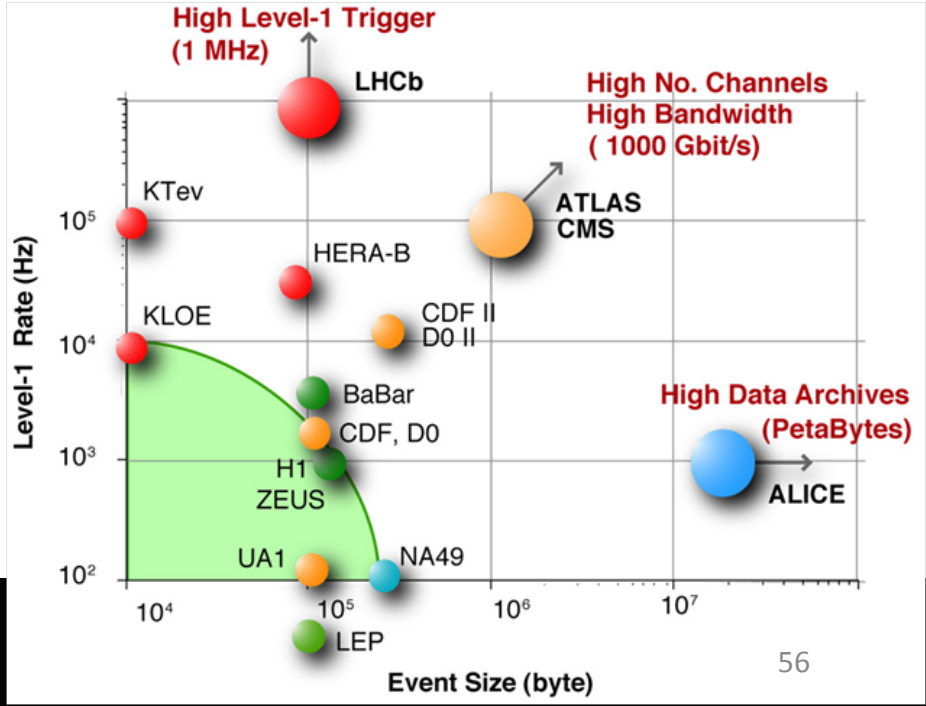


The LHC Computing Challenge

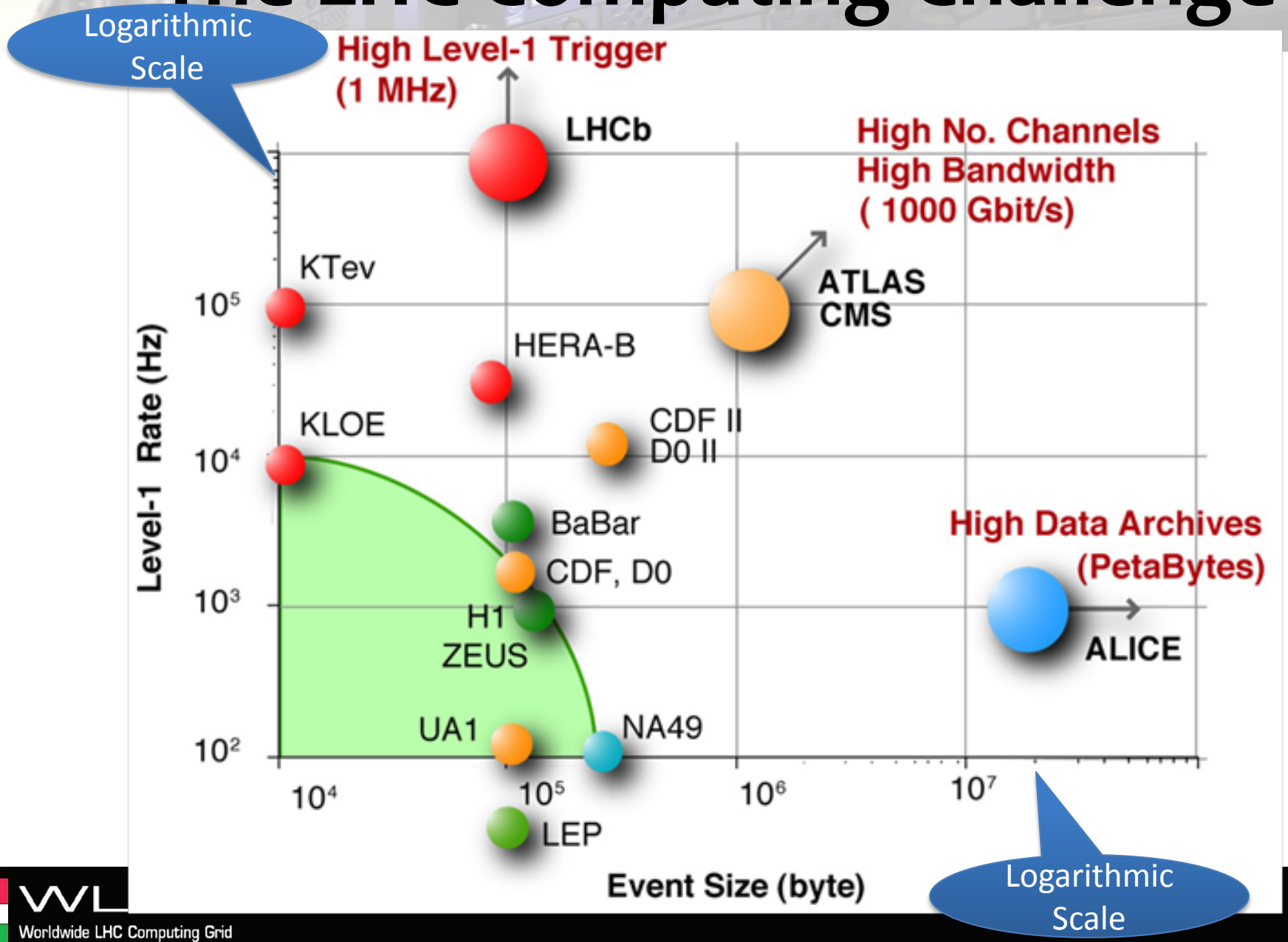
- Signal/Noise: 10^{-13} (10^{-9} offline)
- Data volume
  - High rate * large number of channels * 4 experiments
  - ➔ **15 Peta Bytes of new data each year**
- Compute power
  - Event complexity * Nb. events * thousands users
  - ➔ **280 k of (today's) fastest CPU cores**
  - ➔ **45 PB of disk storage**
- Worldwide analysis & funding
  - Computing funding locally in major regions & countries
  - Efficient analysis everywhere
  - ➔ **GRID technology**



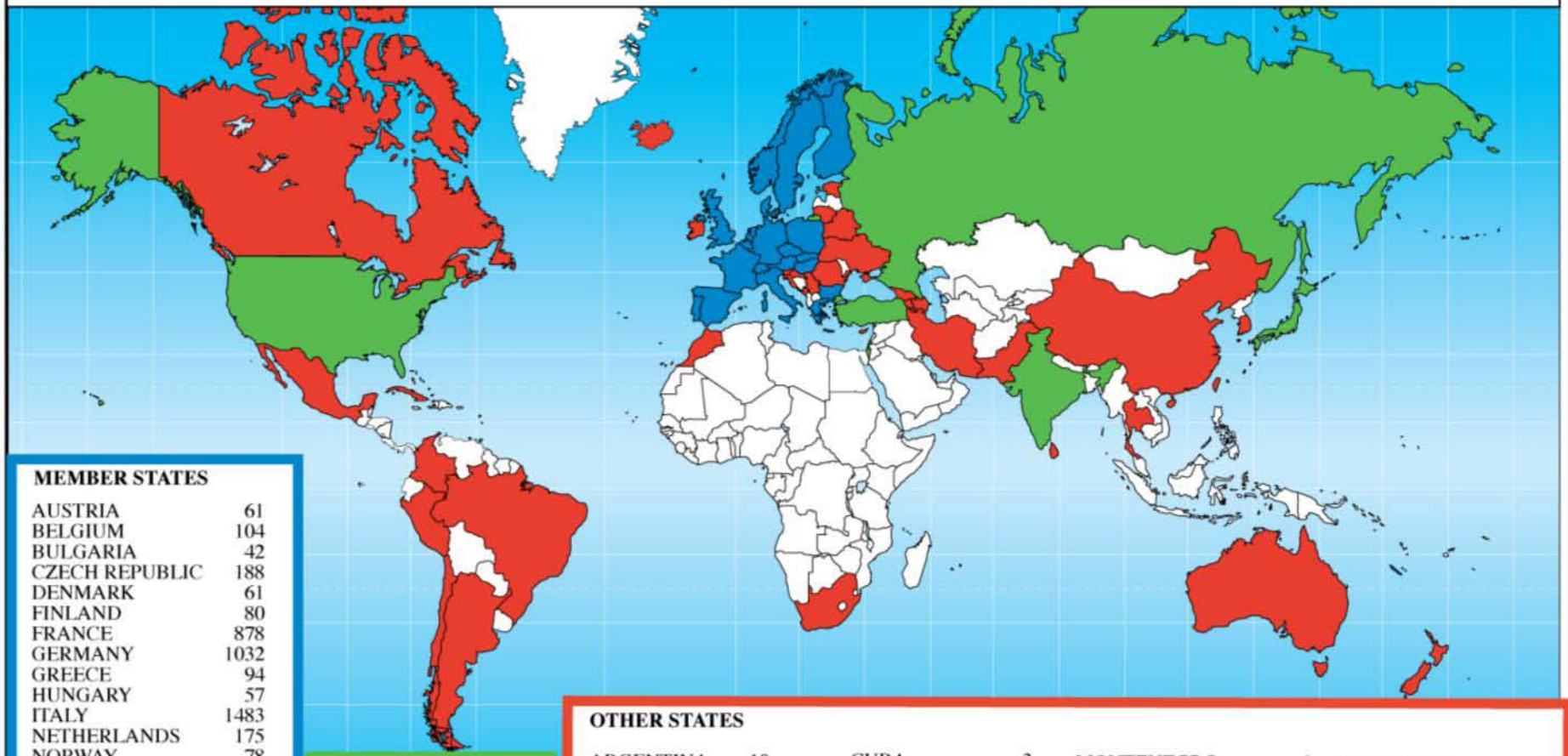




The LHC Computing Challenge



Distribution of All CERN Users by Nation of Institute on 6 January 2009



MEMBER STATES

AUSTRIA	61
BELGIUM	104
BULGARIA	42
CZECH REPUBLIC	188
DENMARK	61
FINLAND	80
FRANCE	878
GERMANY	1032
GREECE	94
HUNGARY	57
ITALY	1483
NETHERLANDS	175
NORWAY	78
POLAND	174
PORTUGAL	111
SLOVAKIA	49
SPAIN	286
SWEDEN	73
SWITZERLAND	330
UNITED KINGDOM	715

6071

OBSERVER STATES

INDIA	89
ISRAEL	59
JAPAN	200
RUSSIA	883
TURKEY	52
USA	1485

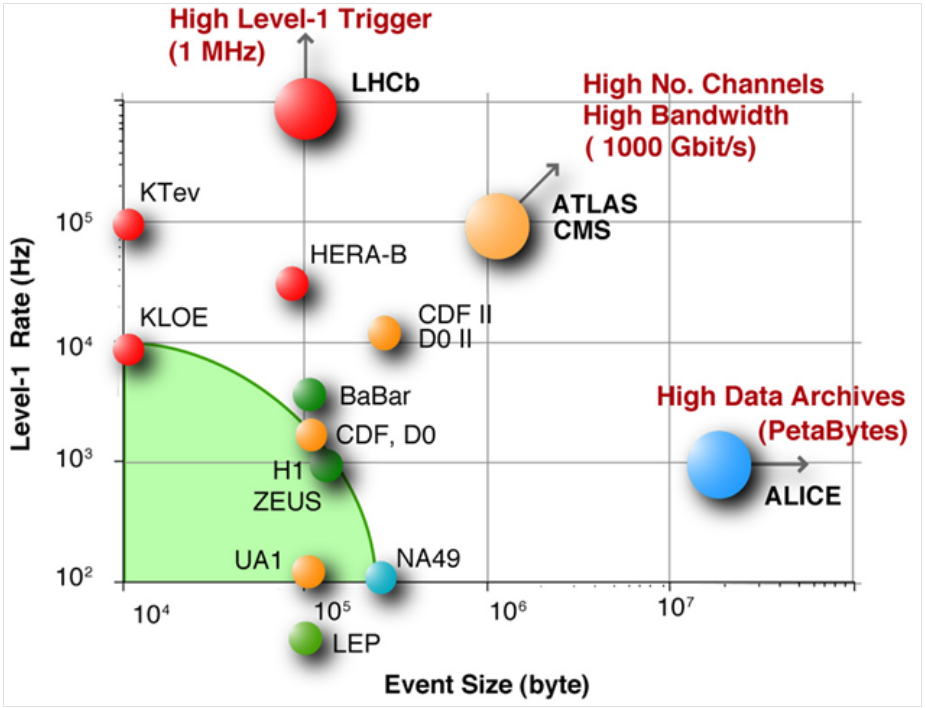
2768

OTHER STATES

ARGENTINA	10	CUBA	3	MONTENEGRO	1	SRI LANKA	1
ARMENIA	15	CYPRUS	6	MOROCCO	5	TAIWAN	42
AUSTRALIA	14	ESTONIA	11	NEW ZEALAND	6	THAILAND	1
AZERBAIJAN	1	GEORGIA	11	PAKISTAN	24	UKRAINE	18
BELARUS	19	ICELAND	1	PERU	1		
BRAZIL	73	IRAN	12	ROMANIA	49		
CANADA	136	IRELAND	12	SERBIA	17		
CHILE	4	KOREA	51	SLOVENIA	16		
CHINA	64	LITHUANIA	5	SOUTH AFRICA	8		
COLOMBIA	11	MEXICO	28				
CROATIA	20						

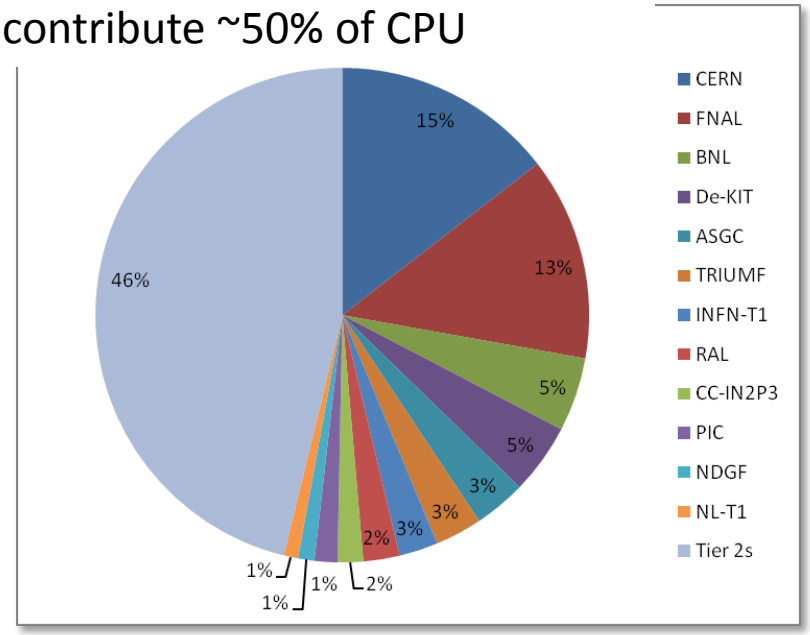
696

Why a grid?



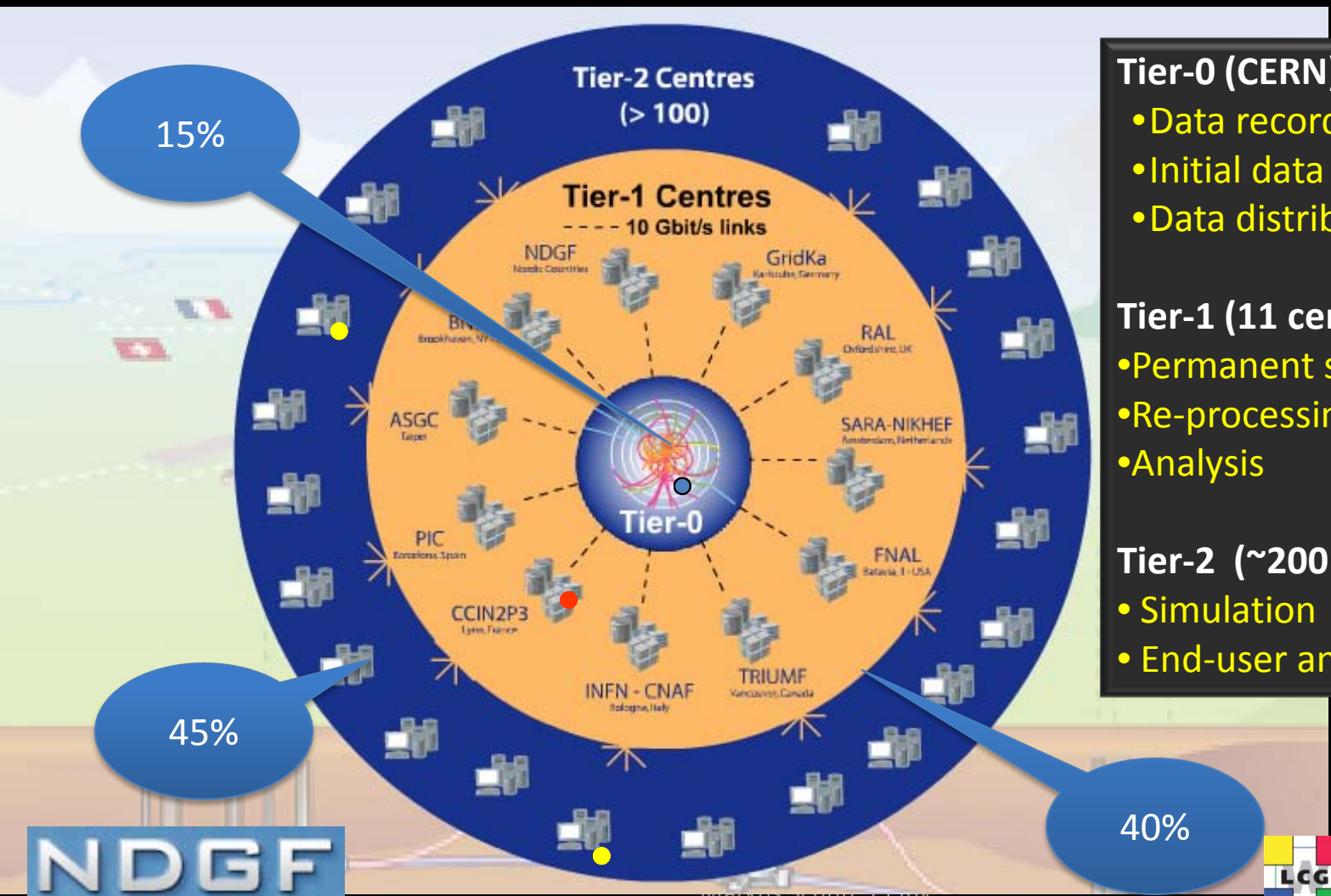
- LHC brings the community into unchartered territory in many domains, including computing

Smaller centres (T2s) contribute ~50% of CPU



- Both practical and political considerations point to a distributed computing solution
- The grid model is particularly well suited to collaboration

Architecture



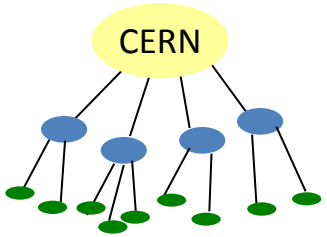
Tier-0 (CERN): (15%)
- Data recording
- Initial data reconstruction
- Data distribution

Tier-1 (11 centres): (40%)
- Permanent storage
- Re-processing
- Analysis

Tier-2 (~200 centres): (45%)
- Simulation
- End-user analysis

History

- 1999 - MONARC project
  - First LHC computing architecture – hierarchical distributed model
- 2000 – growing interest in grid technology
  - HEP community main driver in launching the DataGrid project
- 2001-2004 - EU DataGrid project
  - middleware & testbed for an operational grid
- 2002-2005 – LHC Computing Grid – LCG
  - deploying the results of DataGrid to provide a production facility for LHC experiments
- 2004-2006 – EU EGEE project phase 1
  - starts from the LCG grid
  - shared production infrastructure
  - expanding to other communities and sciences
- 2006-2008 – EU EGEE project phase 2
  - expanding to other communities and sciences
  - Scale and stability
  - Interoperations/Interoperability
- 2008-2010 – EU EGEE project phase 3
  - More communities
  - Efficient operations
  - Less central coordination
- 2010 – 201x EGI and EMI
  - Sustainable infrastructures based on National Grid Infrastructures
  - Decoupling of middleware development and infrastructure



WLCG Collaboration Status
Tier 0; 11 Tier 1s; 64 Tier 2 federations
(124 Tier 2 sites)

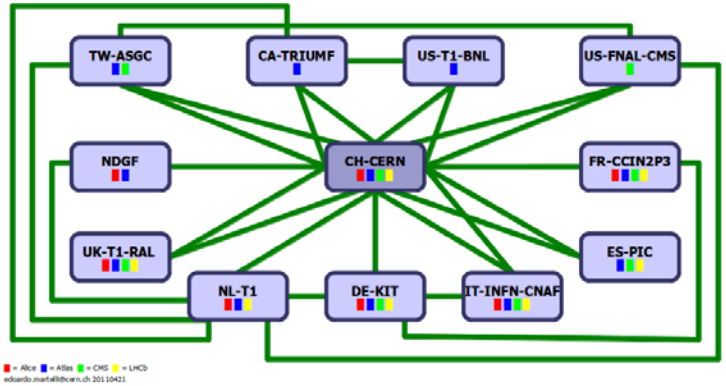
Today we have 49 MoU signatories, representing 34 countries:

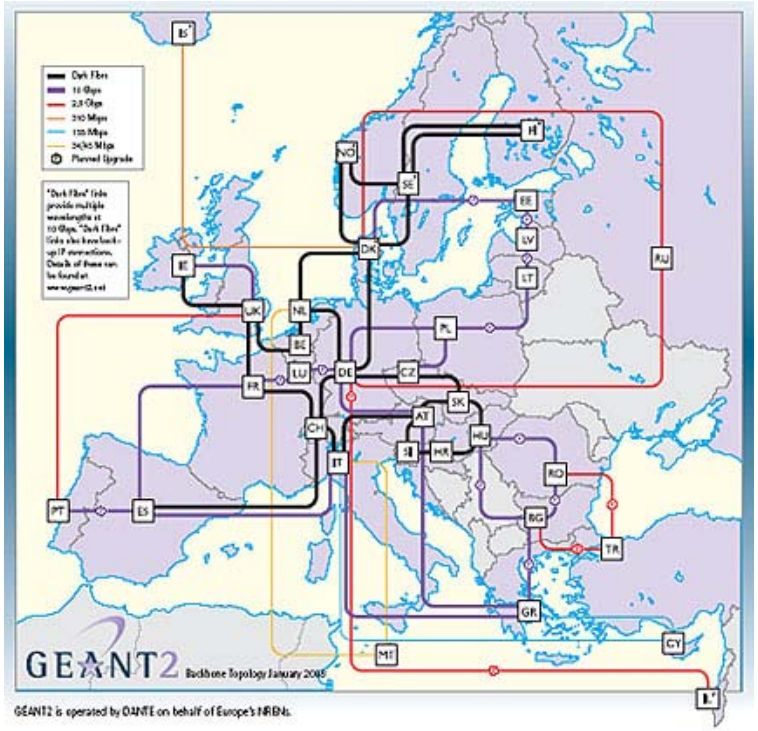
Australia, Austria, Belgium, Brazil, Canada, China, Czech Rep, Denmark, Estonia, Finland, France, Germany, Hungary, Italy, India, Israel, Japan, Rep. Korea, Netherlands, Norway, Pakistan, Poland, Portugal, Romania, Russia, Slovenia, Spain, Sweden, Switzerland, Taipei, Turkey, UK, Ukraine, USA.

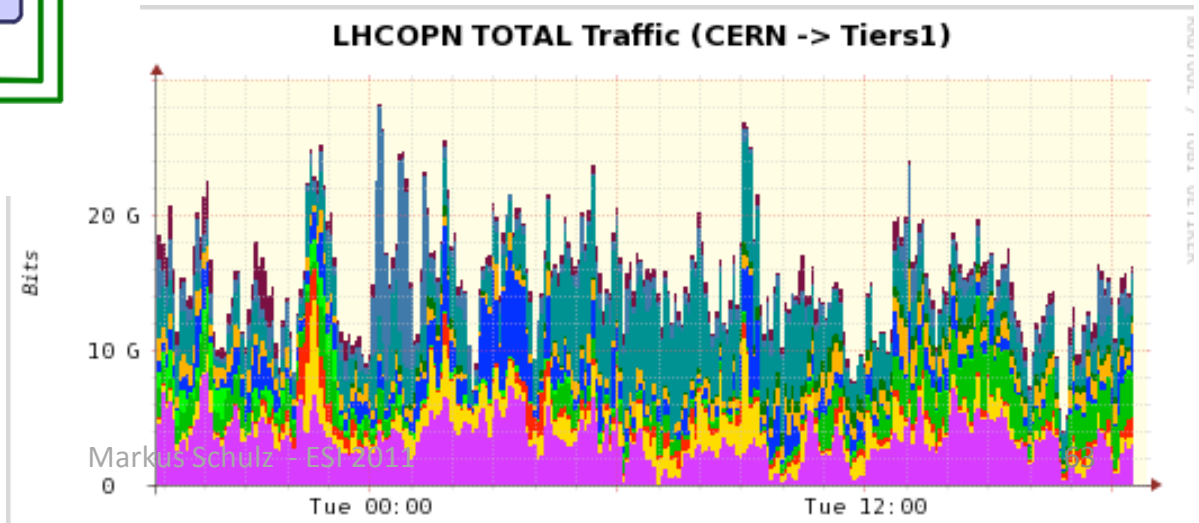
Network

- CERN runs a commercial Internet Exchange Point (IXP)
- Dedicated 10Gb/s optical links to tier 1s
- Rest on the research networks, GEANT2 in Europe

LHCOPN



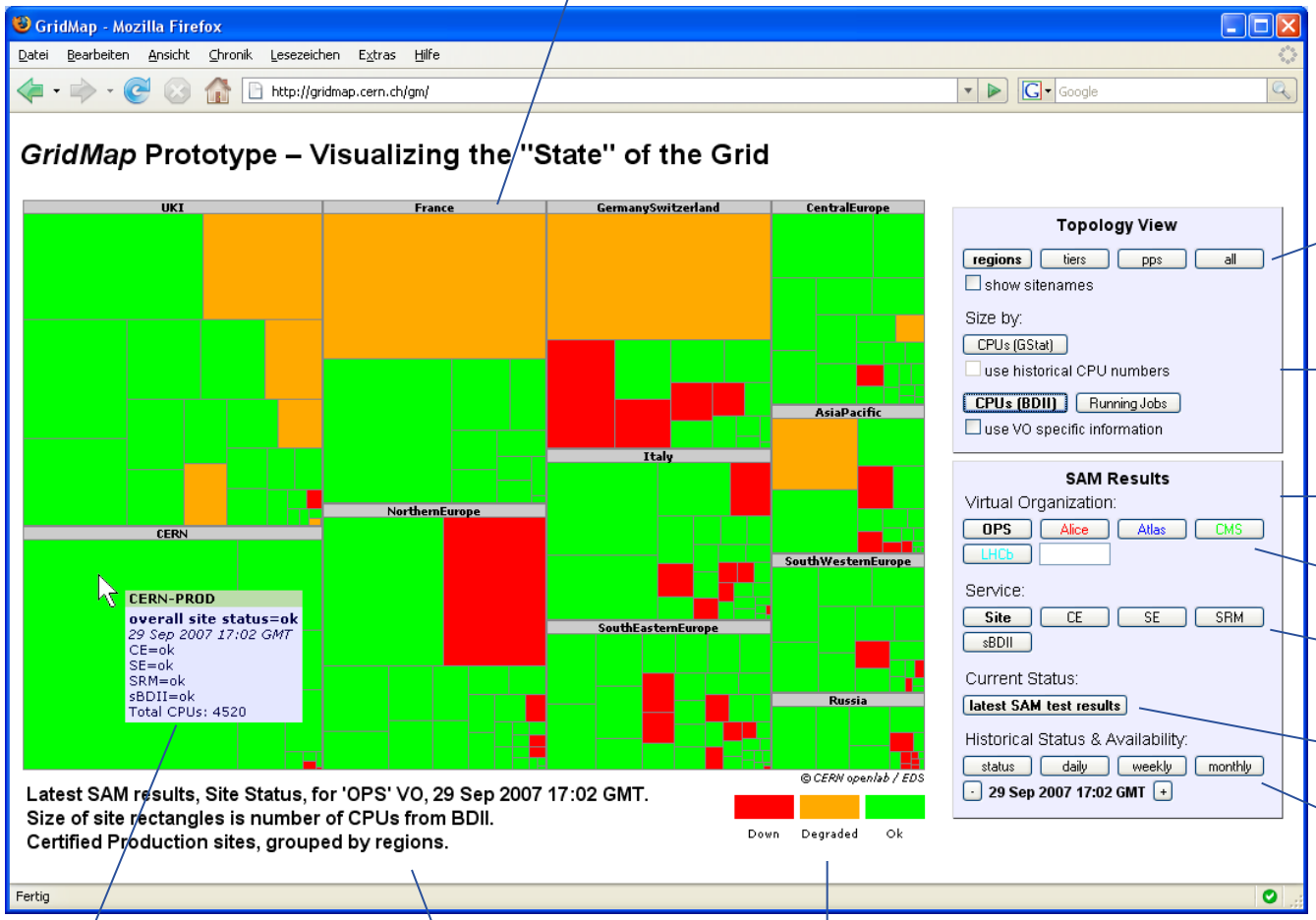




Monitoring

Link: http://gridmap.cern.ch

Drilldown into region by clicking on the title



Grid topology view (grouping)

Metric selection for **size** of rectangles

Metric selection for **colour** of rectangles

VO selection

Overall Site or *Site Service* selection

Show SAM status

Show GridView availability data

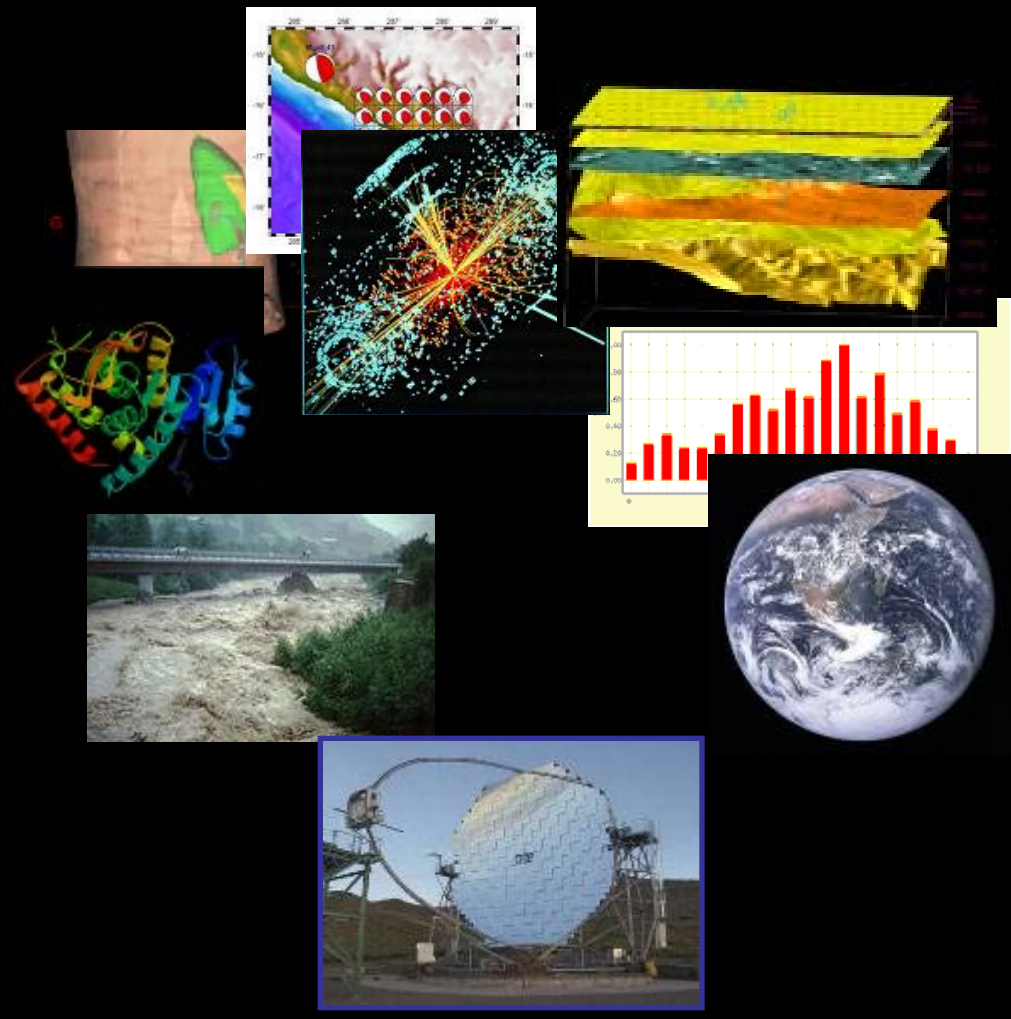
Context sensitive information

Description of current view

Colour Key

EGI Infrastructure

- >270 VOs from several scientific domains
  - Astronomy & Astrophysics
  - Civil Protection
  - Computational Chemistry
  - Comp. Fluid Dynamics
  - Computer Science/Tools
  - Condensed Matter Physics
  - Earth Sciences
  - Fusion
  - High Energy Physics
  - Life Sciences
- Further applications joining all the time
  - Recently fishery (I-Marine)

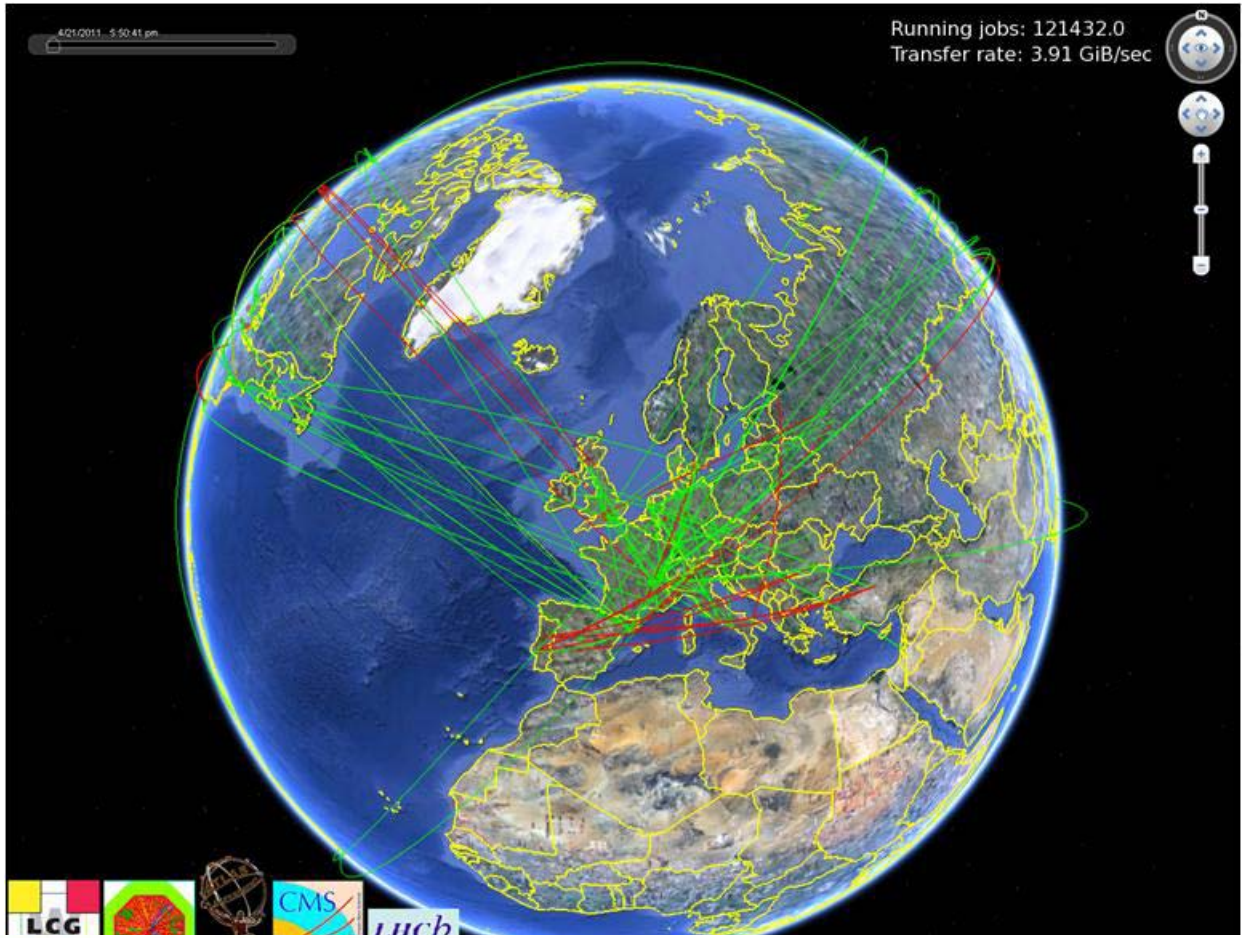


Applications have moved from testing to routine and daily usage

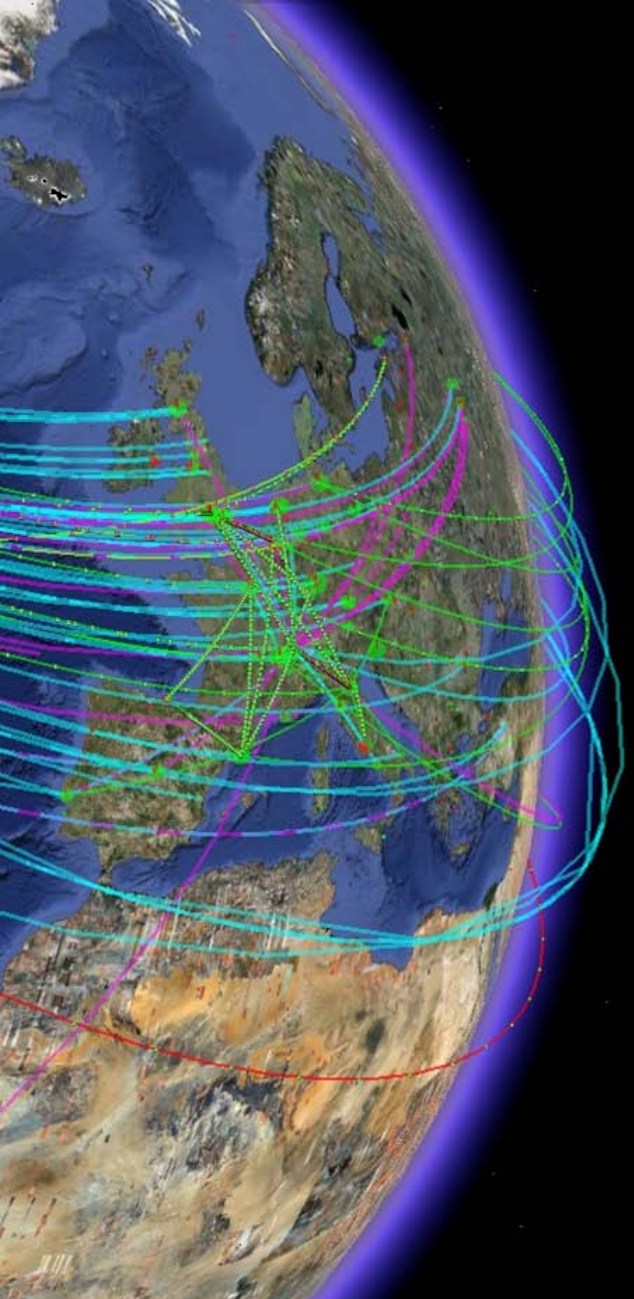
NGIs in Europe

www.eu-egi.eu

WLCG today



WLCG
Worldwide LHC Computing Grid



Usage

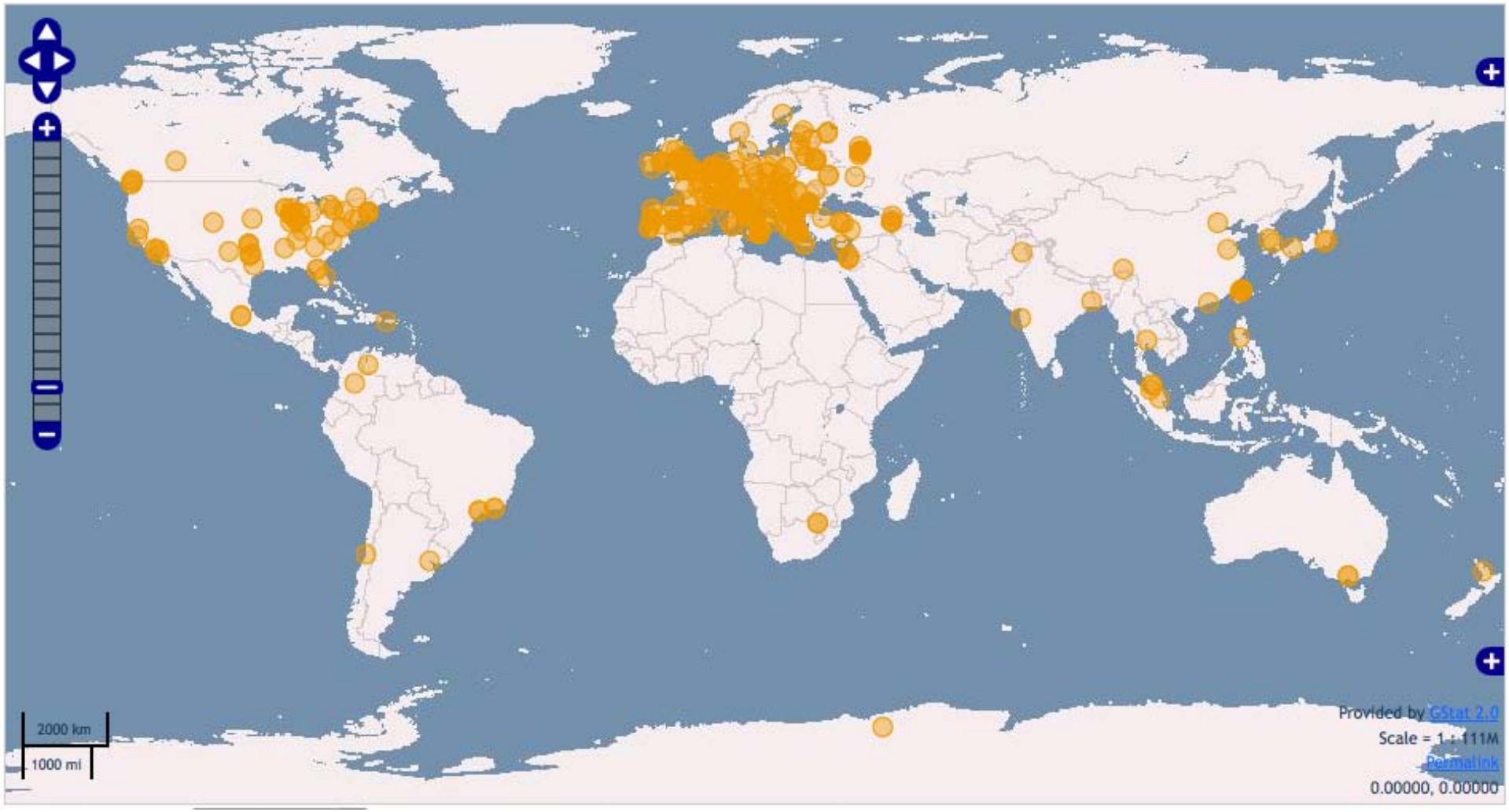
We have a working grid infrastructure
- **With (still) adequate resources**

Experiments use distributed models

Network traffic close to planned
- **Highly reliable**

Large numbers of Individual users
- **CMS ~800**
- **ATLAS ~1000**
- **LHCb/ALICE ~200**

Gstat / installed capacity



Data taken live from the LDAP based information system (BDII)

First year of running - resources

REBUS: Federation Capacities

Topology | Pledges | Capacities

Capacities > Federation Capacities

VO: ALL Year: 2011 Month: 4

Note: Sorting by multiple columns at the same time can be activated by 'shift' clicking on the column headers which they want to add to the sort. Hovering mouse over the column headers to get descriptions of table columns.

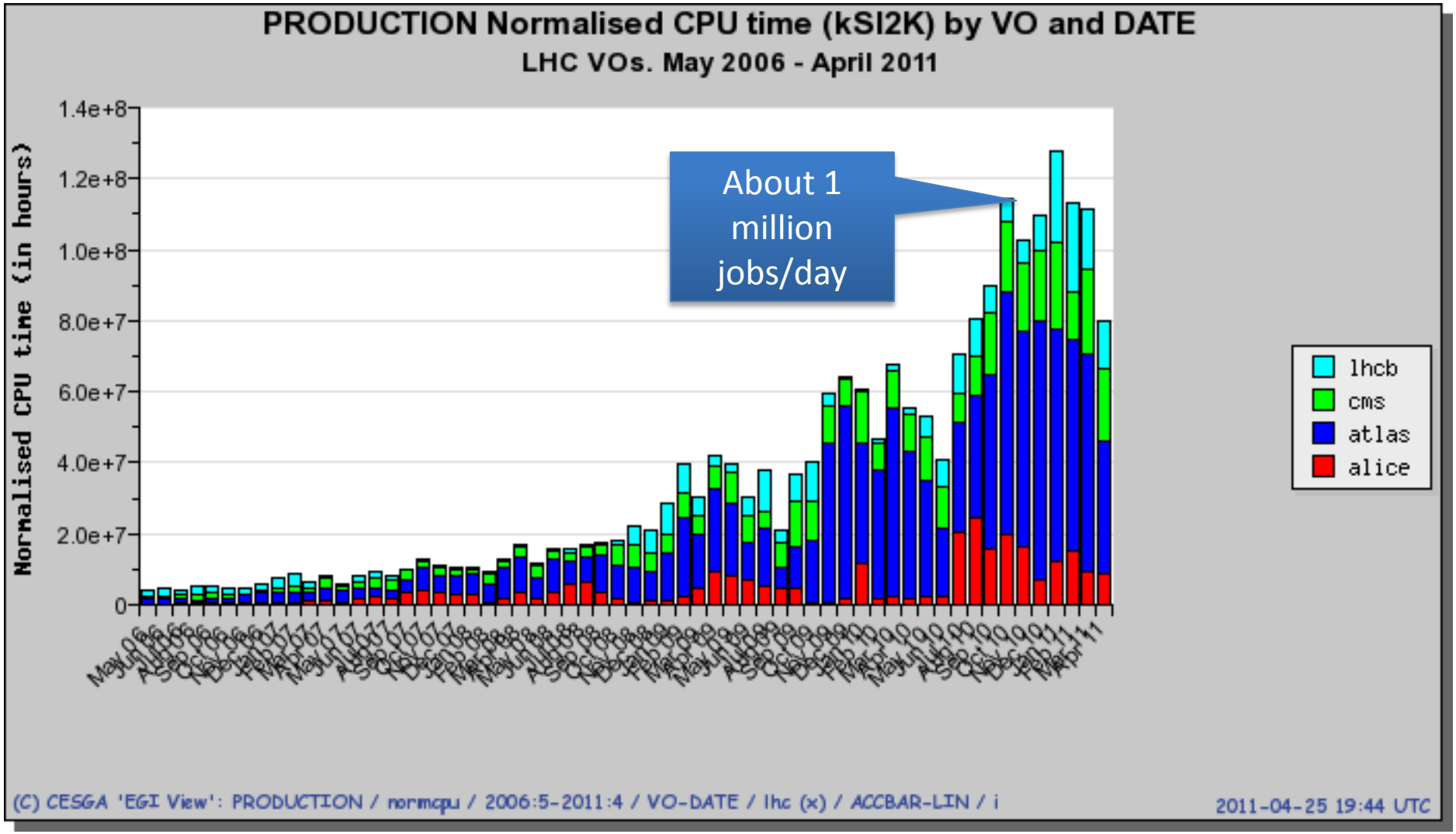
All Tiers | Tier 0 | Tier 1 | Tier 2

JSON CSV XLS Search:

Tier	Country	Federation	Physical CPU	Logical CPU	HEPSPEC06	Total Online Storage (GB)	Total Nearline Storage (GB)
Tier 0	Switzerland	CH-CERN	4,882	23,756	336,314	19,054,839	37,519,529
Tier 1	Canada	CA-TRIUMF	460	1,200	13,381	2,077,923	1,588,800
Tier 1	France	FR-CCIN2P3	1,614	9,068	77,985	6,288,801	22,534,710
Tier 1	Germany	DE-KIT	2,562	13,764	107,891	6,894,613	360,000
Tier 1	Italy	IT-INFN-CNAF	2,252	8,192	85,516	15,525,896	0
Tier 1	Netherlands	NL-T1	1,194	4,512	48,241	4,152,588	2,342,204
Tier 1	Nordic	NDGF	9,608	9,608	89,836	3,337,372	4,464,000
Tier 1	Spain	ES-PIC	744	2,976	32,312	4,373,458	3,618,942
Tier 1	Taiwan	TW-ASGC	1,203	4,812	46,223	2,400,078	3,666,000
Tier 1	UK	UK-T1-RAL	1,568	6,272	64,288	8,244,579	21,777,133
Tier 1	USA	US-FNAL-CMS	1,692	6,768	56,000	6,500,000	21,000,000
Tier 1	USA	US-T1-BNL	[illegible]	[illegible]	[illegible]	[illegible]	6,000,000
Tier 2	Australia	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	0

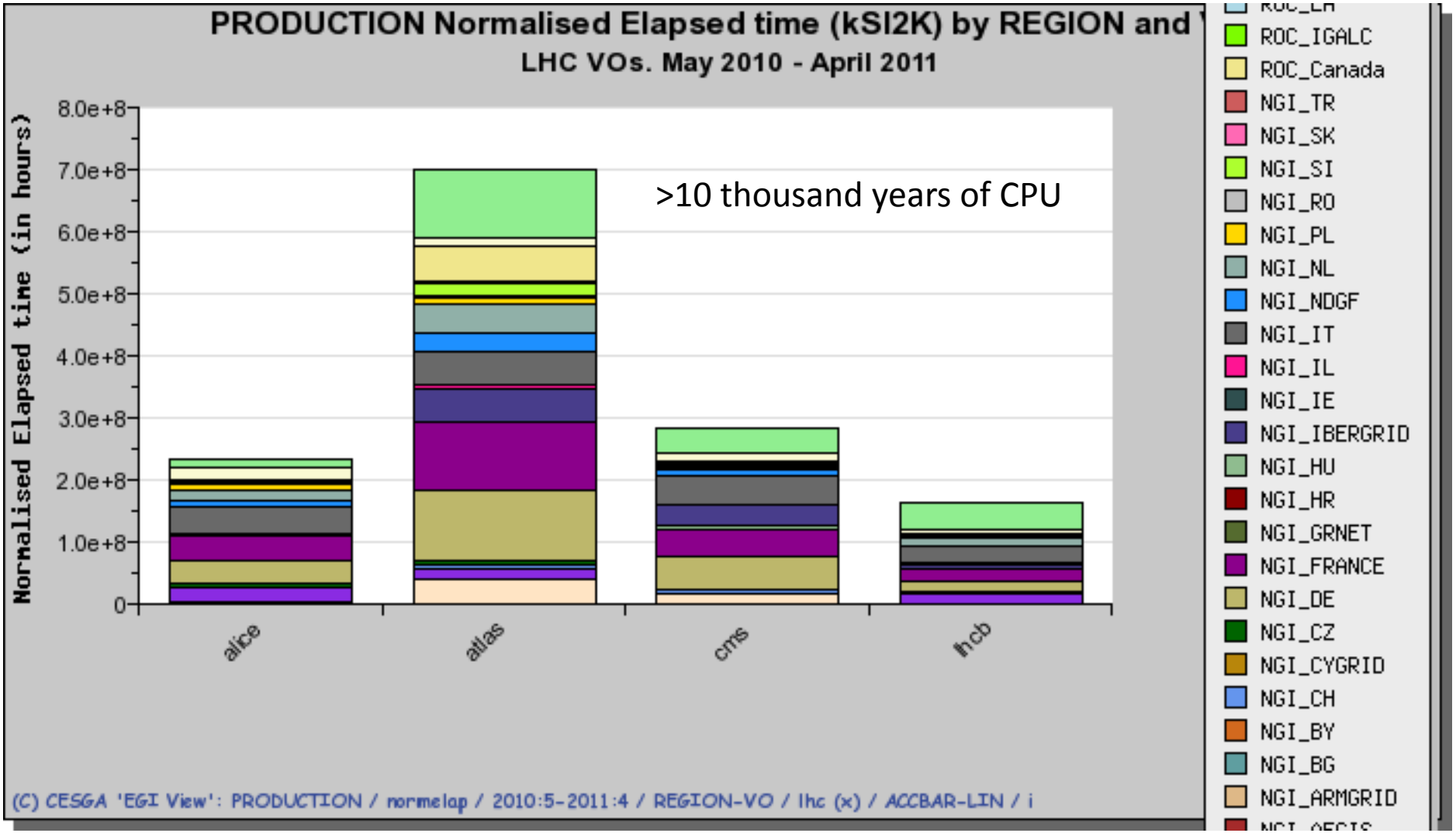
CPUs	264k cores
Disk	158PB
Tape	126PB

CPU usage

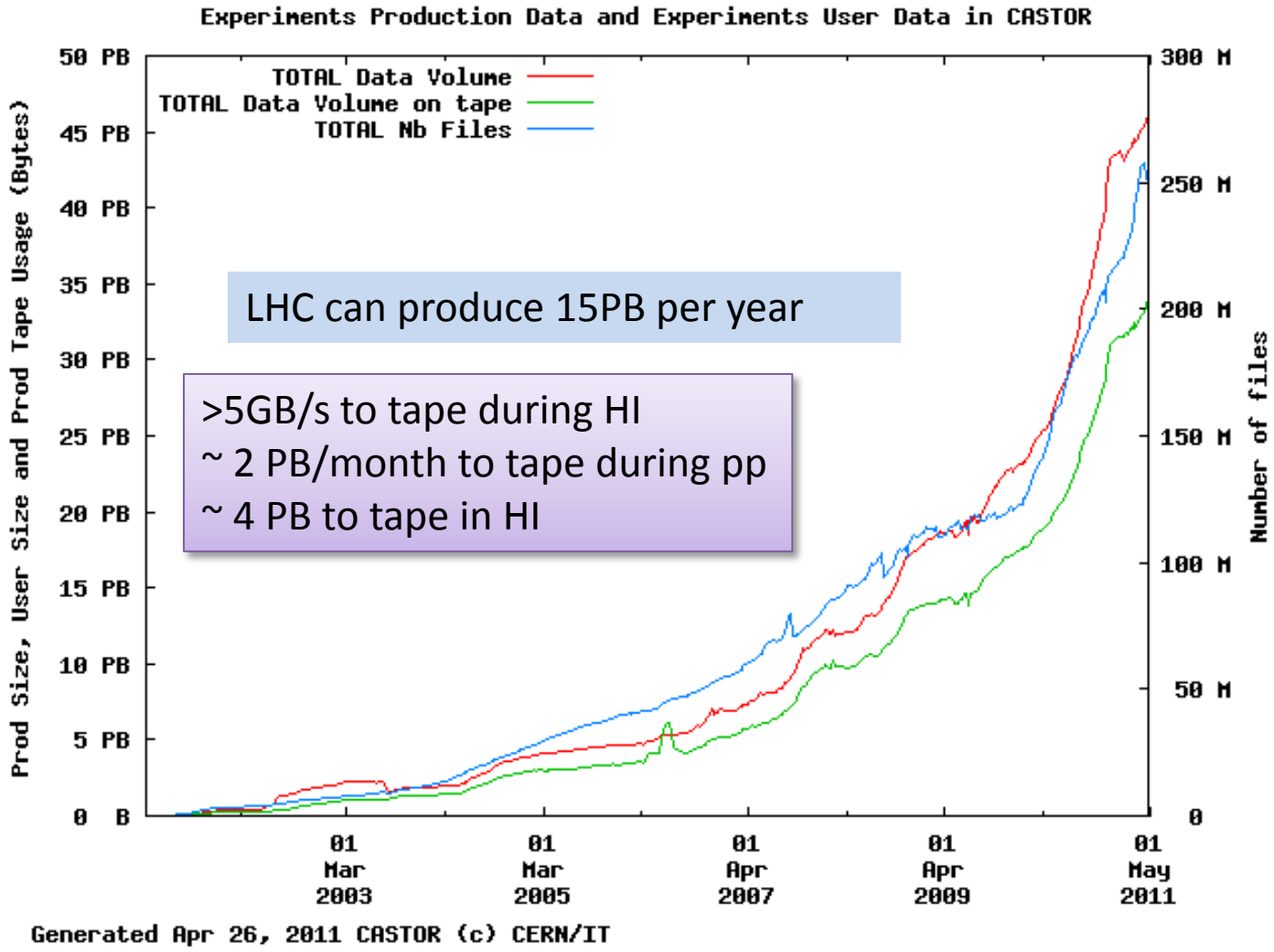


A lot of work continues even when there's no beam

Accumulated CPU usage



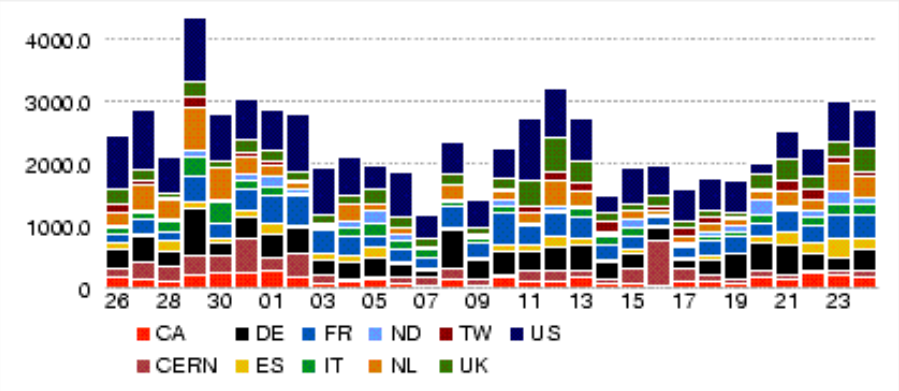
CASTOR – CERN tape storage

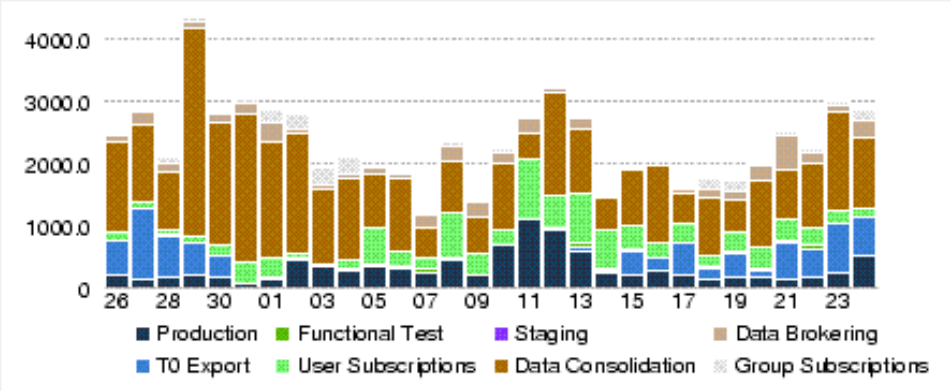


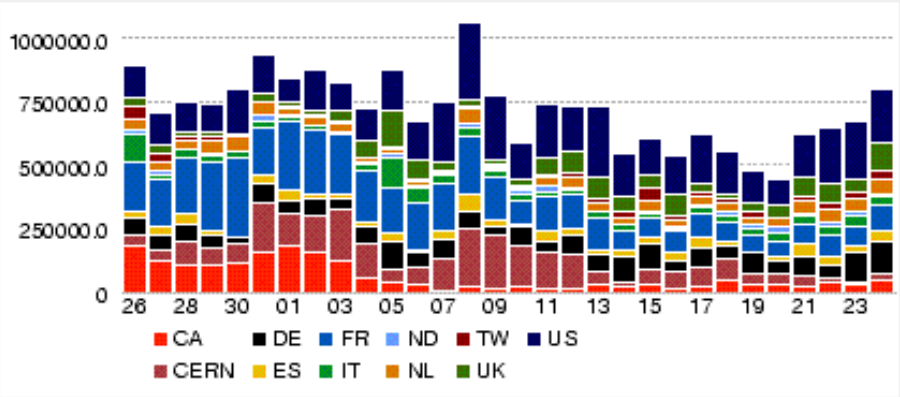
LHC can produce 15PB per year

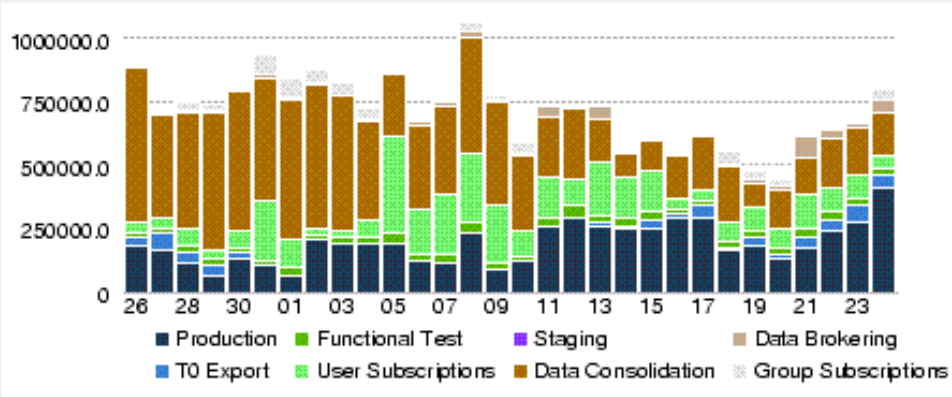
>5GB/s to tape during HI
~ 2 PB/month to tape during pp
~ 4 PB to tape in HI

Atlas data throughput





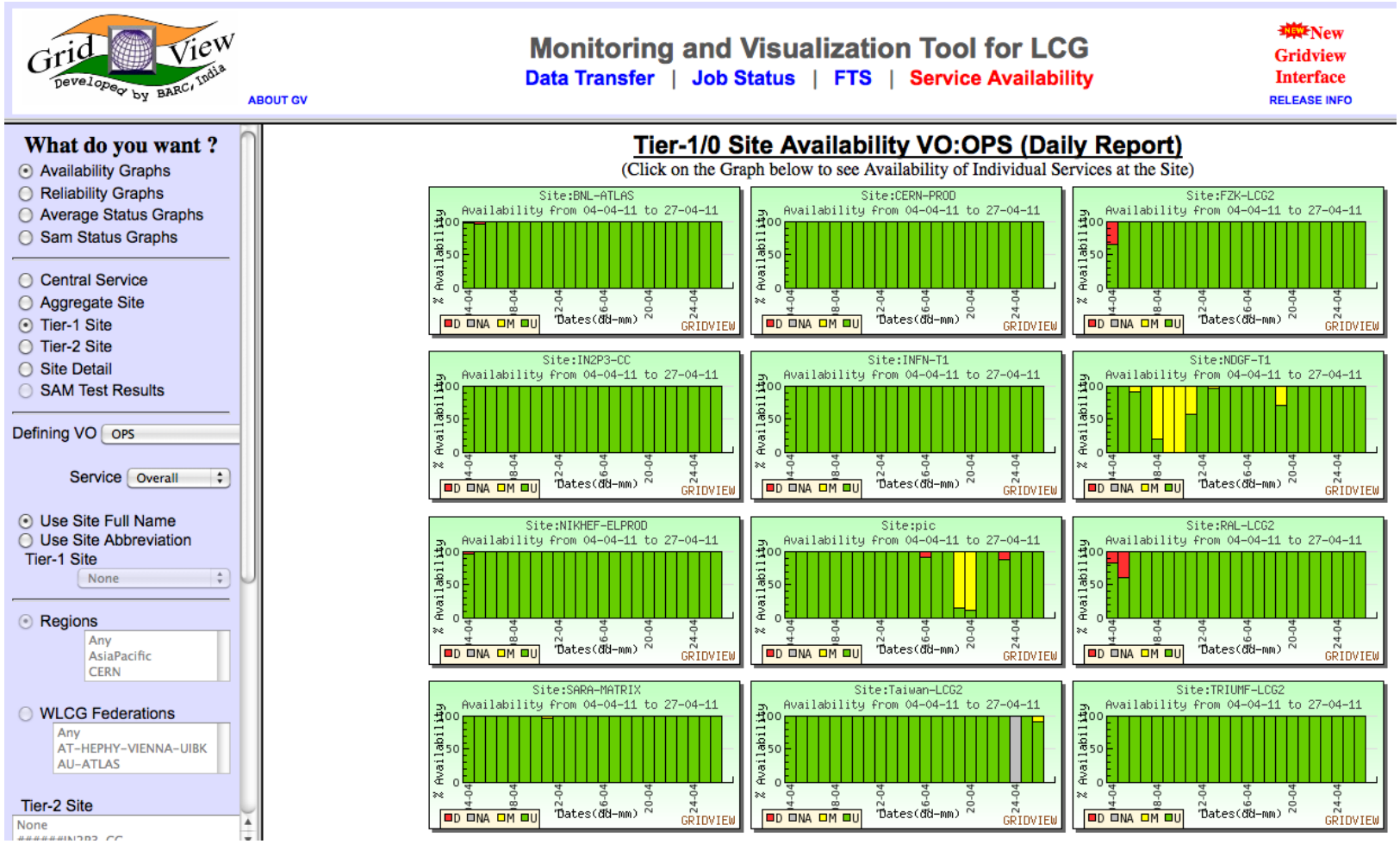




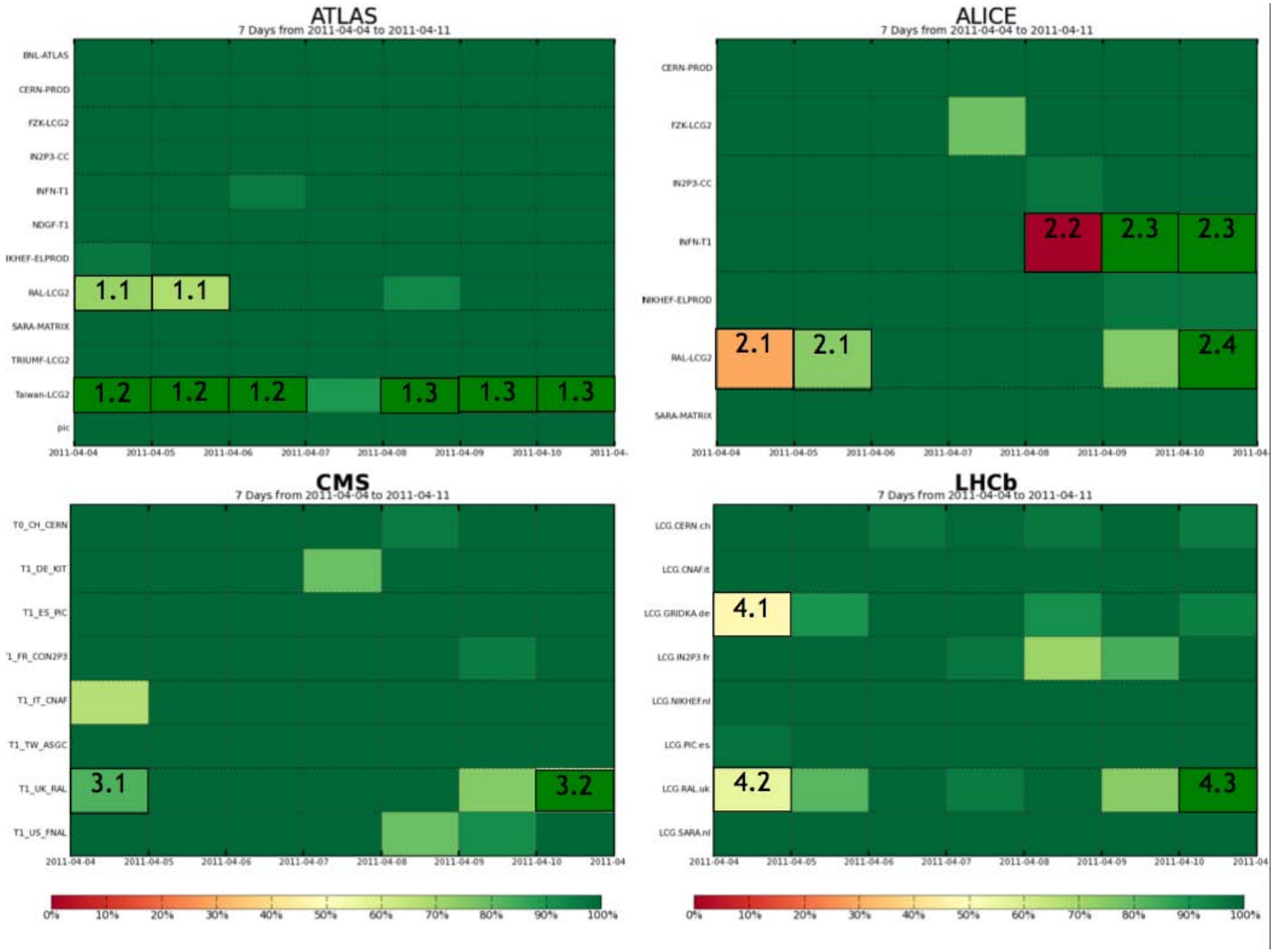
Transfers typically managed and scheduled with FTS

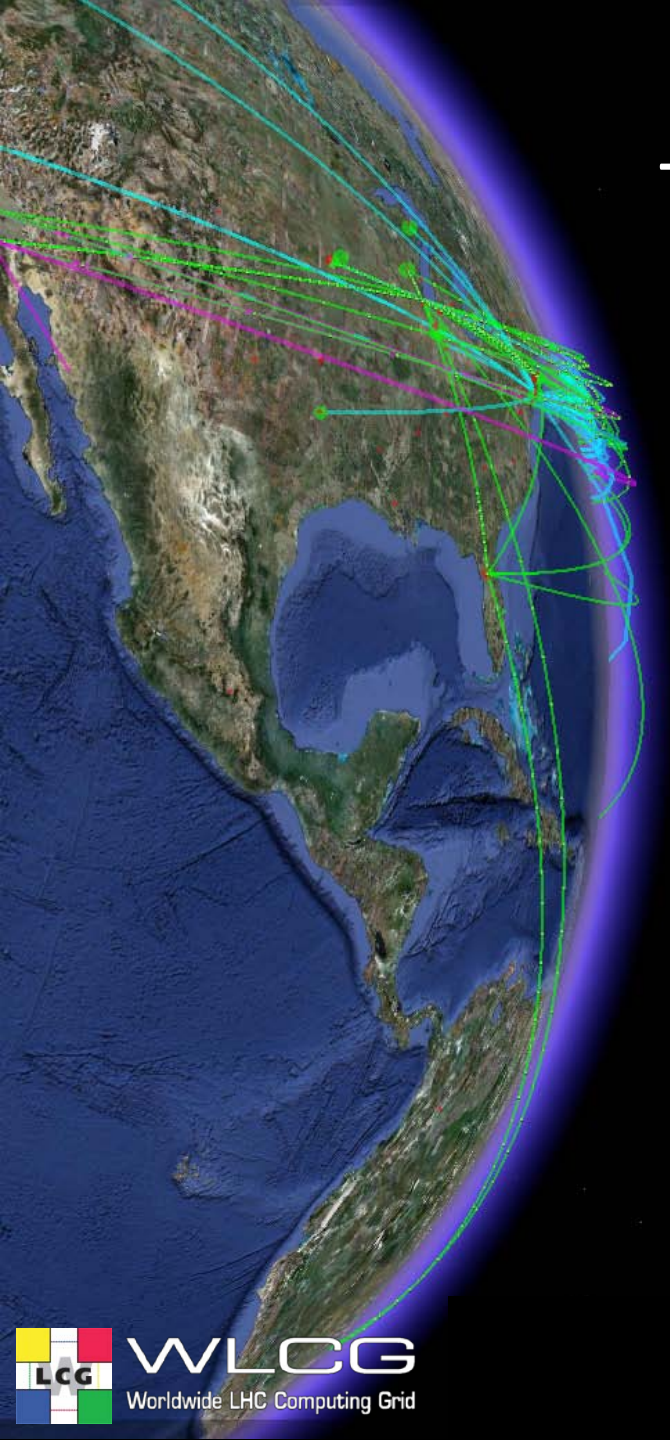
SAM and availability

- Grid community puts a great effort into operations
- Infrastructure is continually monitored with active followup of issues



At the WLCG Management Board





To Grid or not to Grid?

Grid

- Distributed community (VO)
  - Different organizations
  - Distributed resources
- Longer term project (> 2 years)
  - With massive computing requirements (>> 100 PC nodes)
- Computing requires modest parallelization
  - MPI is available on some sites, but not easy to use in a Grid

- Don't expose middleware directly to end users
  - Link from workflow management/portals
  - Shield users from failures/complexity
  - Distributed computing requires management of failures
- Join an existing infrastructure
  - EGI is in Europe a good choice
- Use workflow management software from other Vos
  - Dirac, Panda, gCube from D4Science …..
- Get sufficient expertise…..

Half Grid

- Distributed small community (< 100)
  - Closely linked (same region or organization)
  - Distributed resources
- Medium term project (< 2 years)

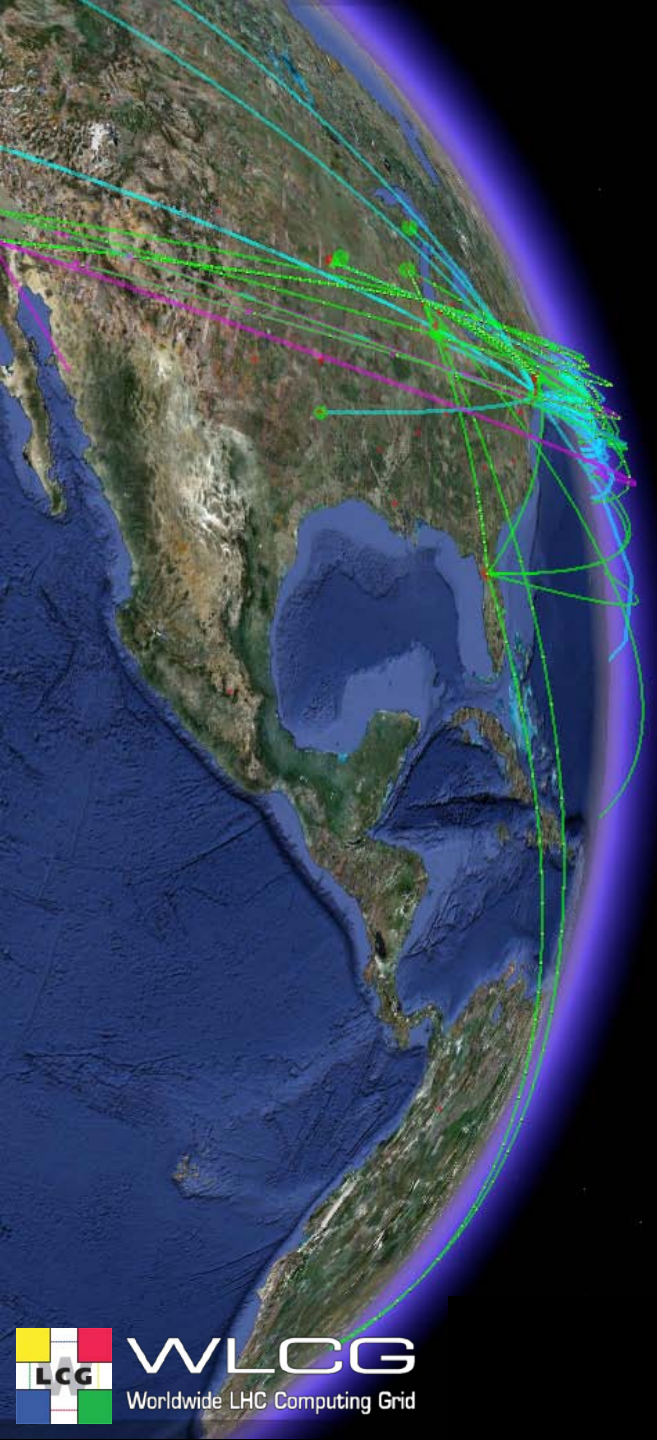
- Join an existing VO (use their experience)
- Or:
  - Link your resources via Condor
    - http://www.cs.wisc.edu/condor/
- Or:
  - Use cloud computing (OpenStack, O

- Or:
  - Use volunteer computing (BOINC (like Seti@home)
  - We interfaced gLite and BOINC… not much use by HEP
- You still need to invest, but you will see results faster

No Grid

- Local team
  - Closely linked (same region or organization)
  - Distributed resources
- Short or medium term project (< 2 years)
- Massive parallel processing needed or HPC needed

- If you choose using the gird nevertheless…
  - Understand the startup costs

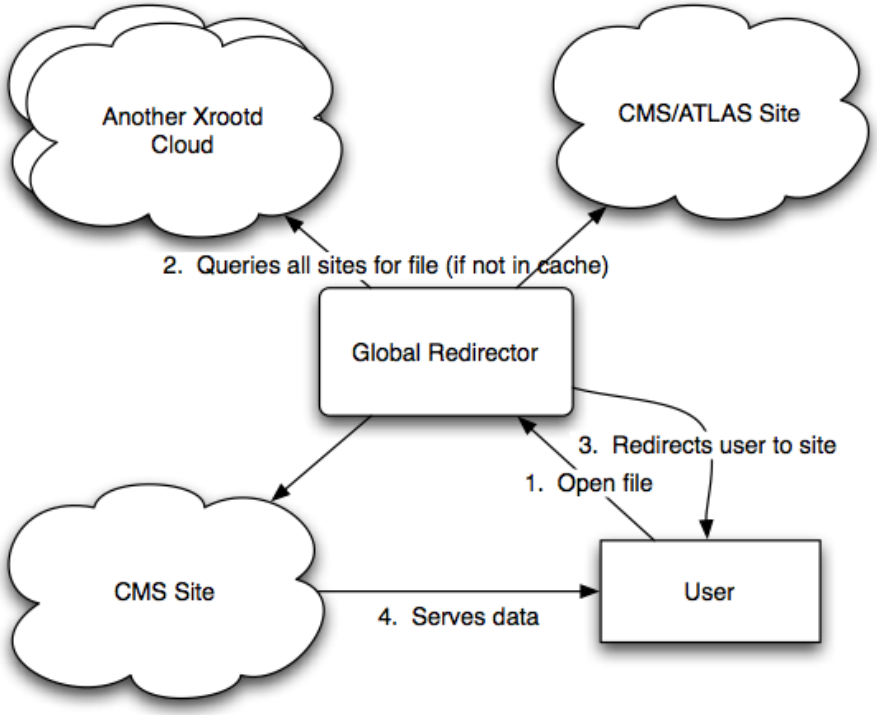


Future

- WANs are now very stable and provide excellent performance
  - Move to a less hierarchical model
- Virtualization and Cloud Computing
- Adapting standards
- Integrating new technology

Data access: a WAN solution

- Data access over the WAN is now a possibility
  - More efficient use of storage
  - Greater job reliability
  - Not necessarily more WAN traffic
    - Can be combined with various caching strategies
  - Can be quicker than pulling something locally from tape
- NFSv4.1 offers this possibility (WAN optimised operations, parallelism)
- A global xrootd federation is being demonstrated by CMS:



Virtualisation & cloud

Virtualisation is interesting in a number of domains

- Application Environment
- HEP applications are platform dependent
  - Sites & laptops are varied

CernVM
Software Appliance

OpenNebula.org
The Open Source Toolkit for Cloud Computing

- Infrastructure Management

openstack
CLOUD SOFTWARE

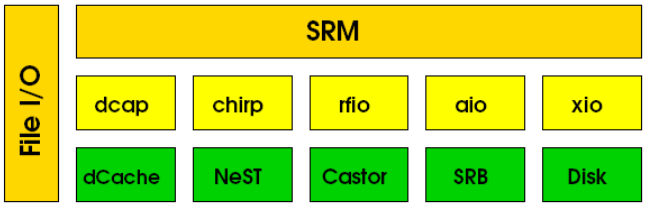
- Direct cloud use by LHC experiments
  - Simulation
  - Elasticity
  - Reprocessing & analysis
    - Data cost

amazon web services

Adoption of standards

EMI has embraced the adoption of standards, many applications see the benefits

- OGF, OASIS
- Storage Resource Manager (SRM)
  - hides the storage system implementation (disk or active tape)
  - handles authorisation
  - Many implementations: DPM, dCache, StoRM, BeSTman, Castor, dCache
- GLUE 2.0 Information Schema



- Non-grid standards
  - NFSv4.1
  - SRM/https
  - WebDAV
  - HTTP
  - SSL

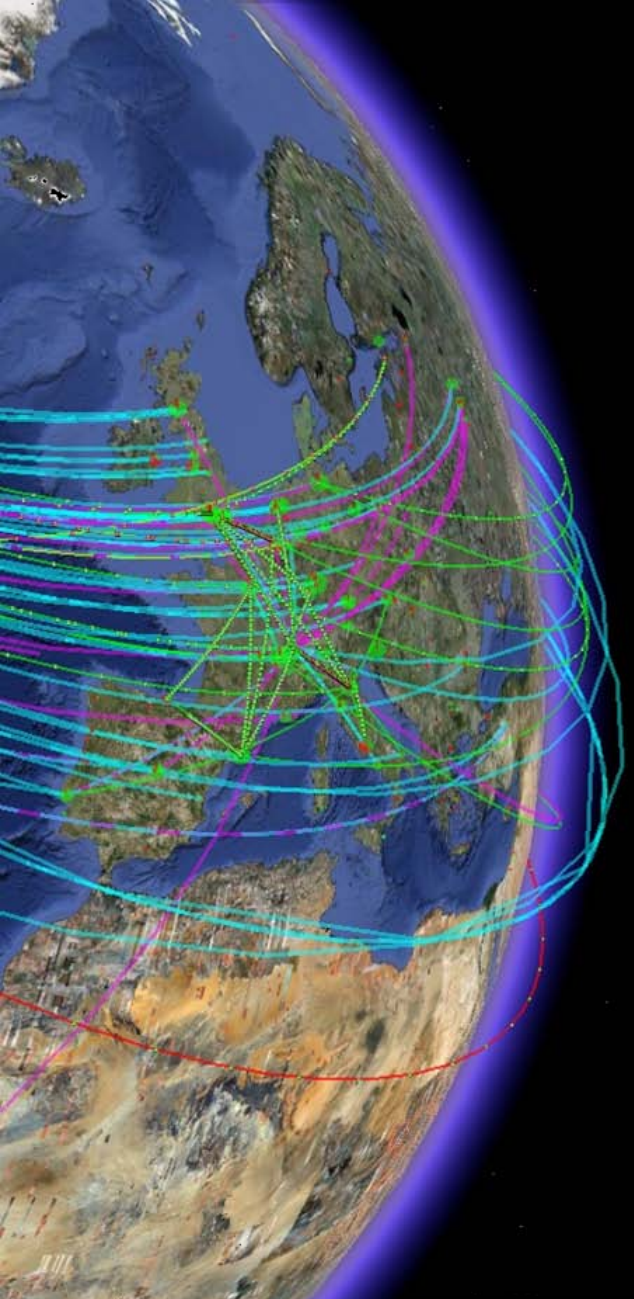
- Why NFSv4.1?
  - Simplicity
    - Regular mount-point and real POSIX I/O
  - Performance
    - pNFS : parallel NFS
    - Clever protocols

Other future developments…

- WLCG and experiment frameworks require long-term planning
- Many projects are taking advantage of emerging technology
- An incomplete selection:

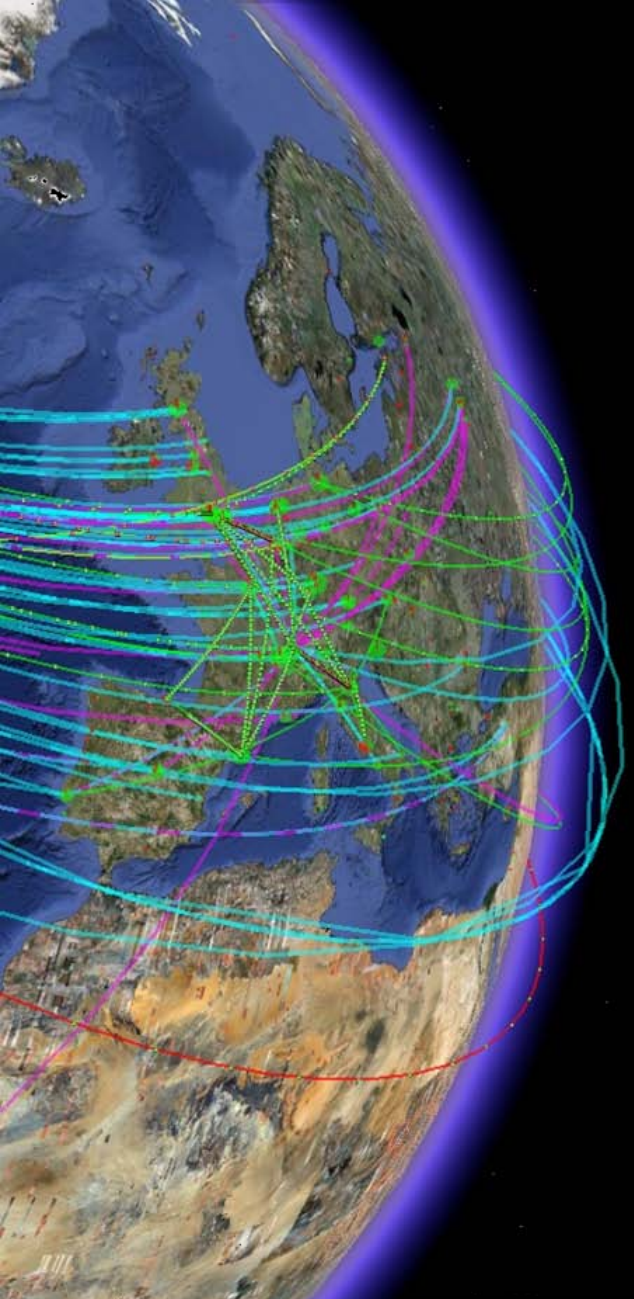
Multicore	Efficiency, esp in memory usage
LHCONE	Improved networking in T2
NoSQL	Performance improvements
CERNVMfs	Distribution of applications
Authentication	User friendly ways to authenticate
…etc	

------BEGIN CERTIFICATE------
MIIHmCCdaSDFpopiopjan242ASD2qrA2



Summary

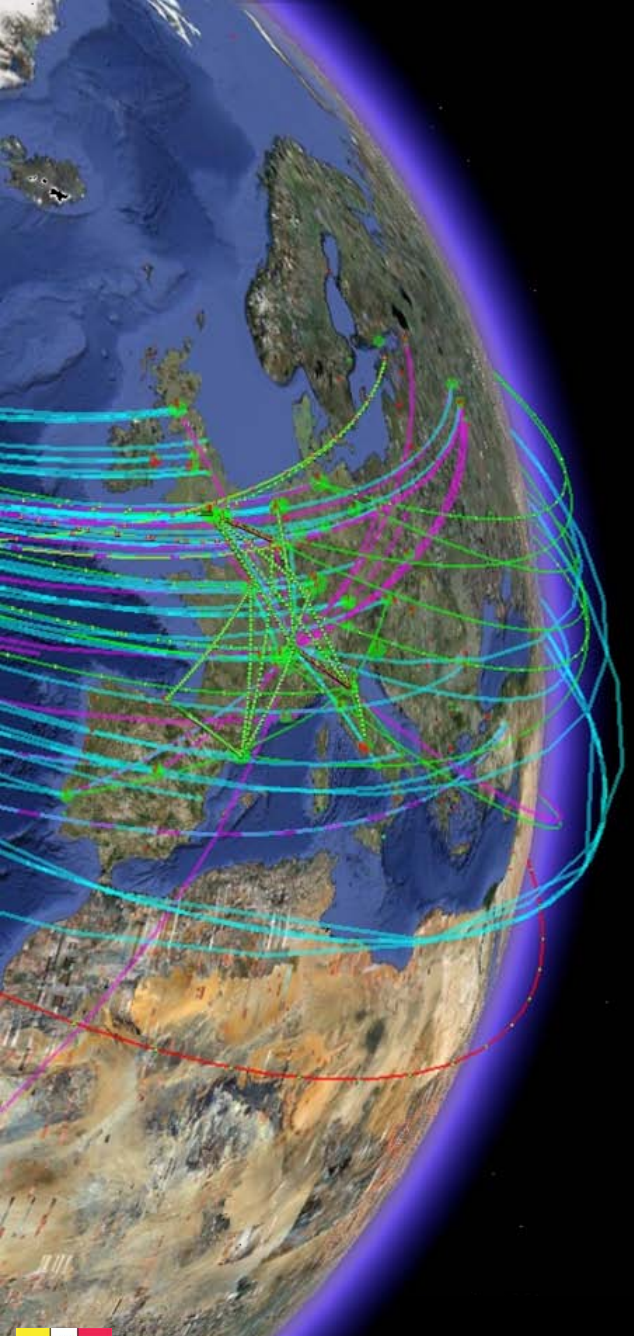
- Grid Computing and WLCG has proven itself during the first year of data-taking of LHC
- Grid computing works for our community and has a future



Overview

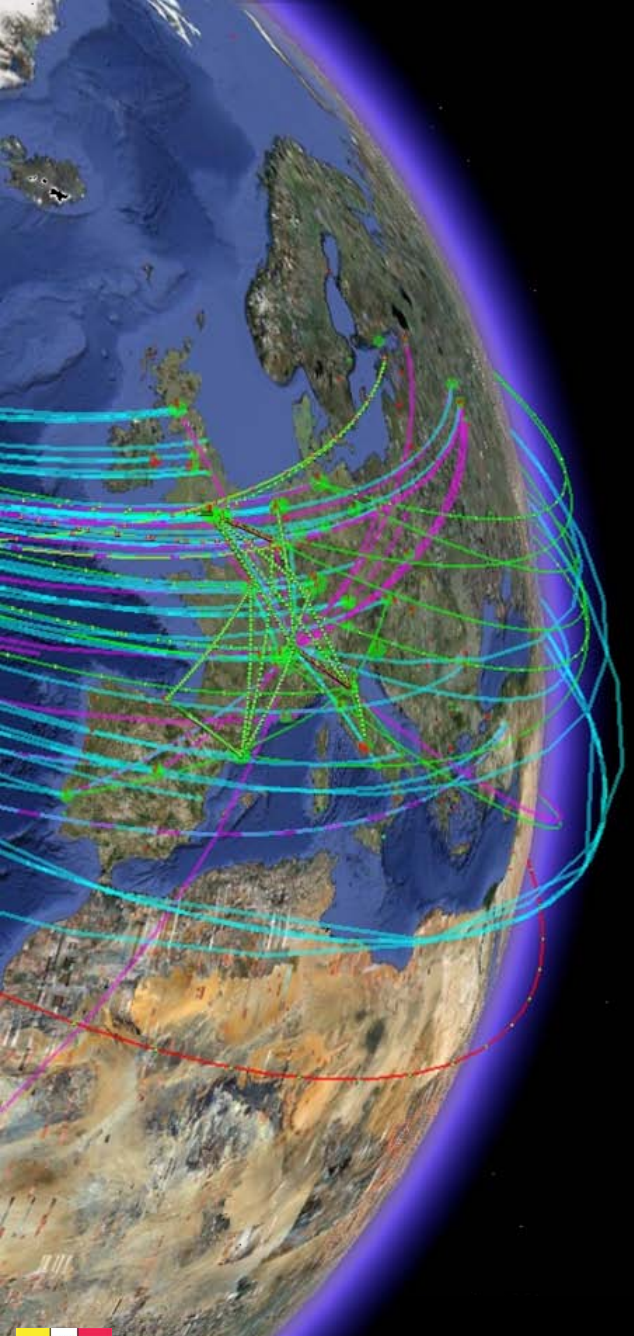
- If you want to use gLite read the user guide:
- https://edms.cern.ch/document/722398/

- There is NO way around it ☺
  - Unless you are in an LHC experiment



Thank you

WLCG
Worldwide LHC Computing Grid



Extra Slides

European Middleware Initiative (EMI)

Primary Objectives

Consolidate

Consolidate the existing middleware distribution simplifying services and components to make them more sustainable (including use of off-the-shelf and commercial components whenever possible)

Evolve

Evolve the middleware services/functionality following the requirement of infrastructure and communities, mainly focusing on operational, standardization and interoperability aspects

Support

Reactively and proactively maintain the middleware distribution to keep it in line with the growing infrastructure usage

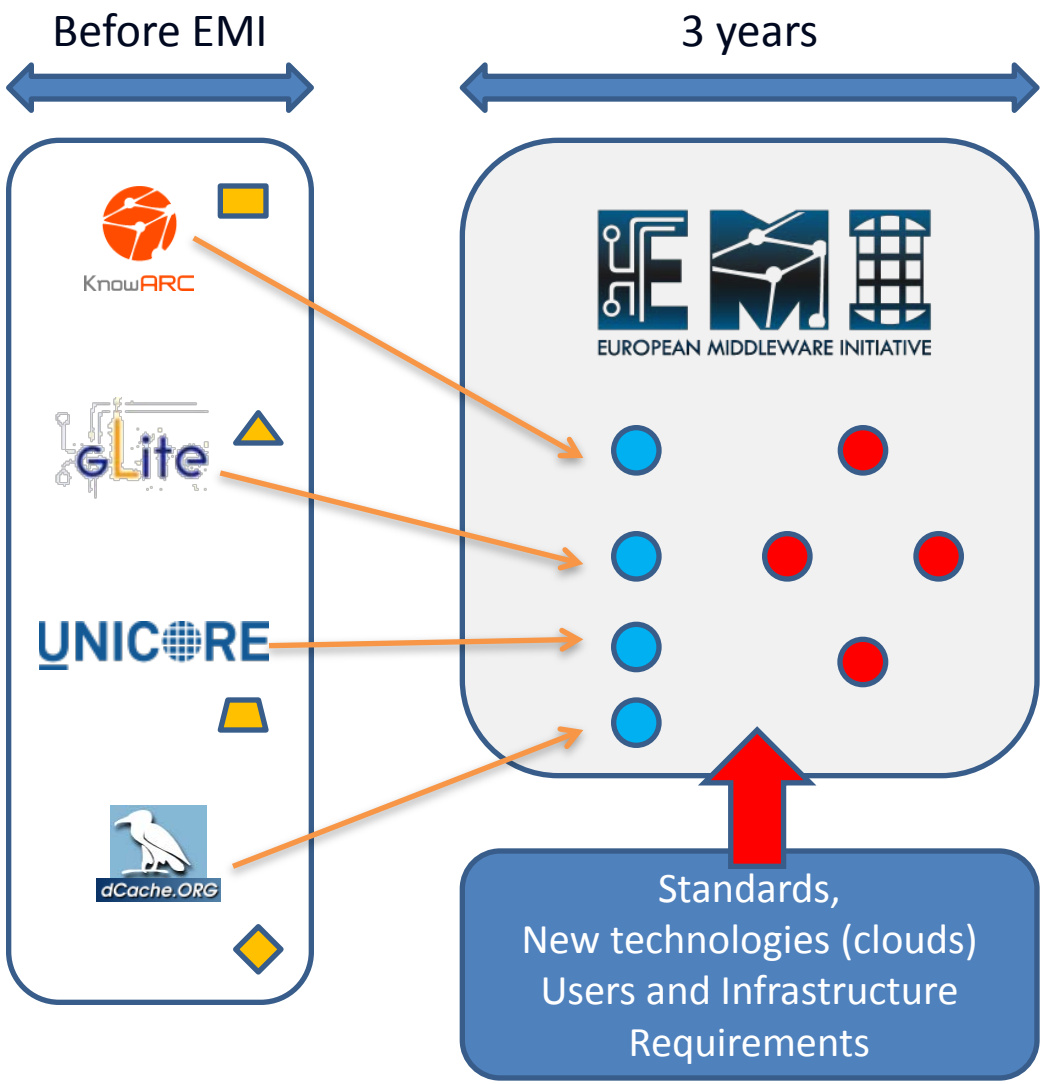
Partners (26)


Lund University
Uppsala Universitet
University of Oslo
Google
Manchester 1824
The University of Manchester
Helsinki Institute of Physics
Science & Technology Facilities Council
Jülich Forschungszentrum
icm
NIKHEF
Technische Universität Dresden
CERN
KISTI
www.kisti.re.kr
CESNET
CESGA
SWITCH
INFN
CINECA
grnet
CSIC
NO
SE
FI
RU
EE
LV
LT
GB
IE
DE
PL
BE
CZ
SK
UA
HU
PT
ES
HR
BA
RS
GE
AZ
AM
GR
IS

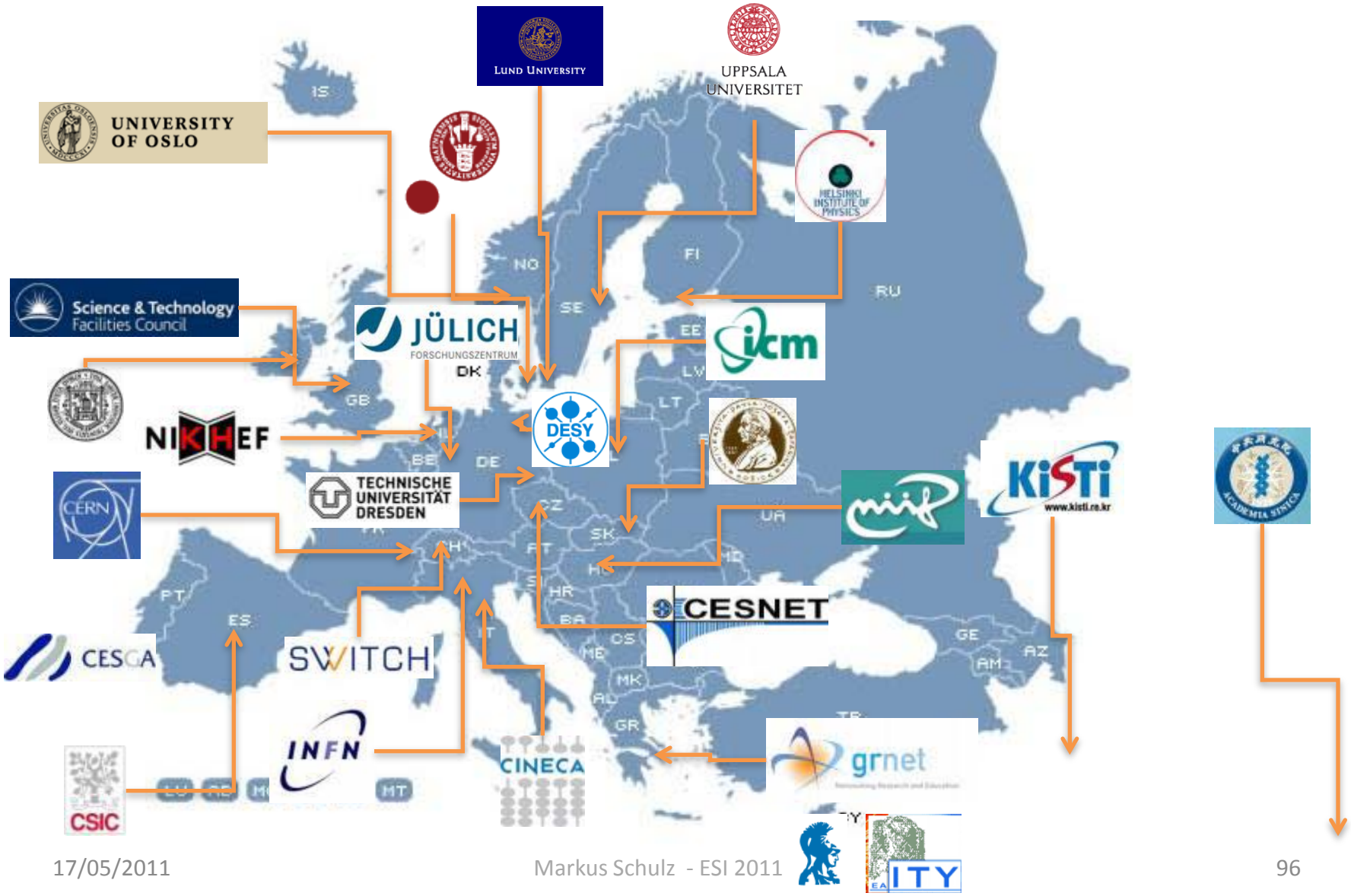
Technical Areas

Compute Services	A-REX, UAS-Compute, WMS, CREAM, MPI, etc
Data Services	dCache, StoRM, UAS-Data, DPM, LFC, FTS, Hydra, AMGA, etc
Security Services	UNICORE Gateway, UVOS/VOMS/VOMS-Admin, ARGUS, SLCS, glExec, Gridsite, Proxyrenewal, etc
Infrastructure Services	Logging and Bookkeeping, Messaging, accounting, monitoring, virtualization/clouds support, information systems and providers

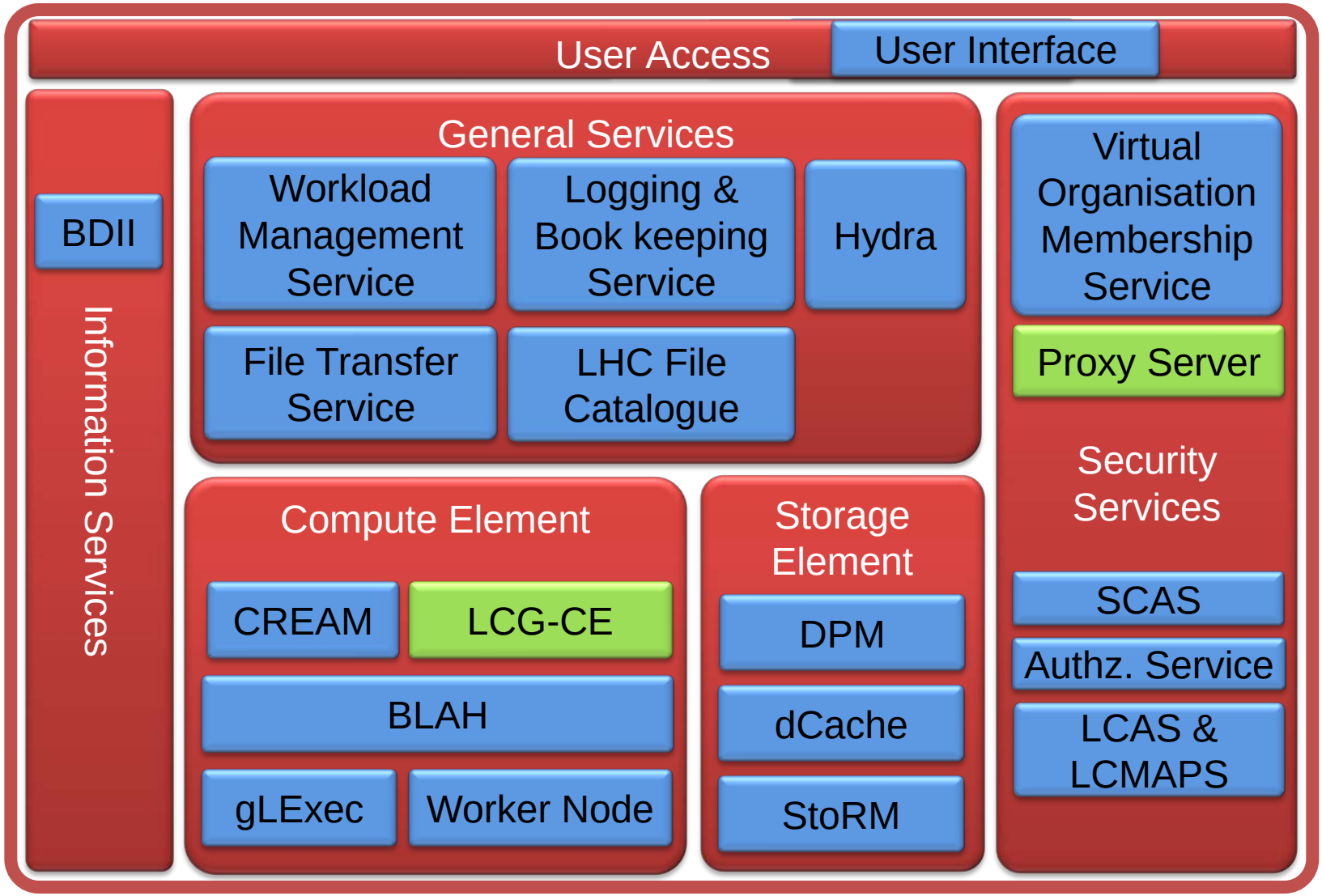
Middleware – the EMI project



EMI Middleware



EMI services



Not all EMI services are illustrated