vl-e virtual laboratory for e-science

BiG Grid the dutch e-science grid

Grid Computing and Site Infrastructures

David Groep, NIKHEF

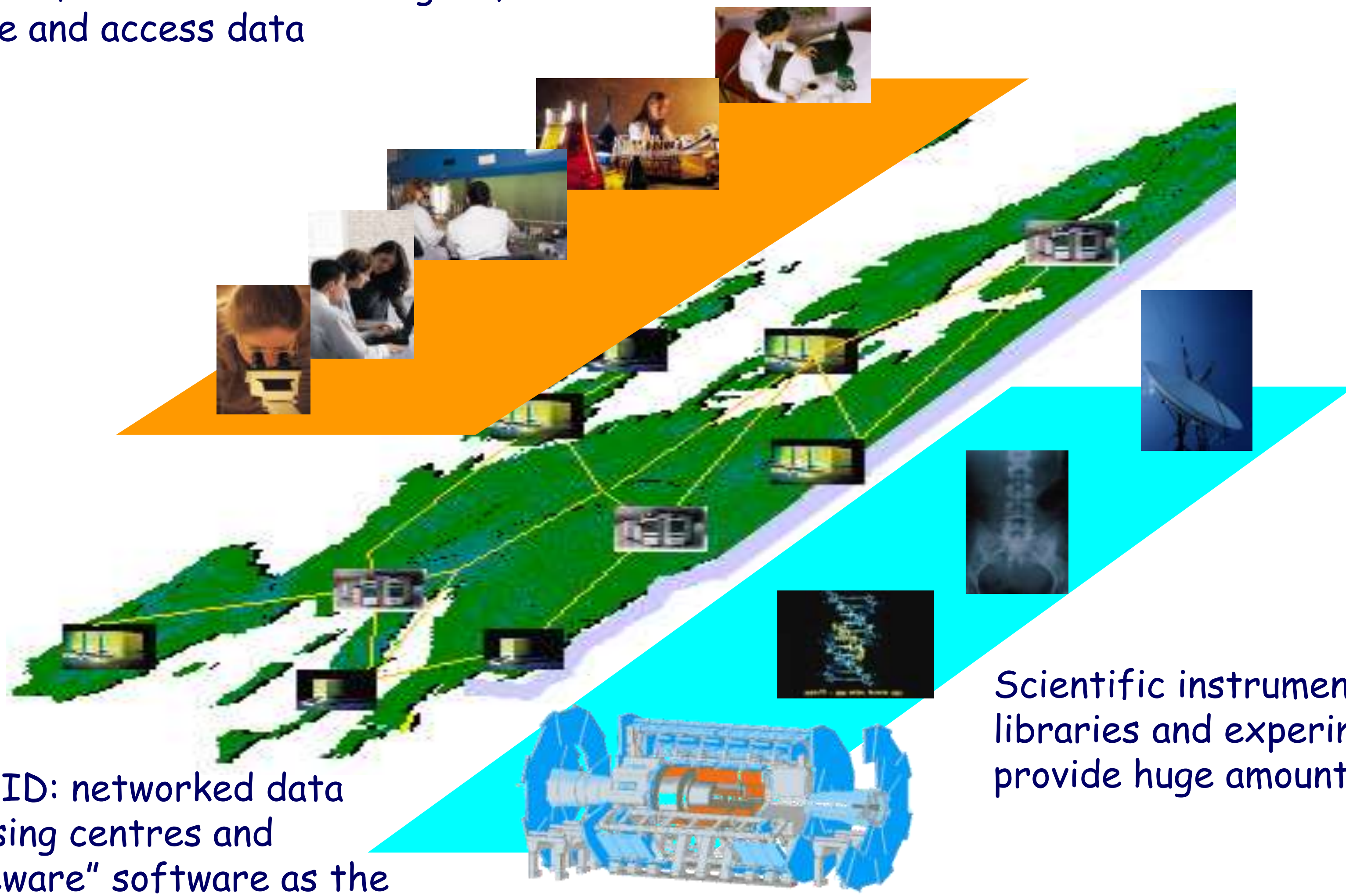
UvA SNE 2008

NIKHEF pdp

Graphics: Real Time Monitor, Gidon Moont, Imperial College London, see http://gridportal.hep.ph.ic.ac.uk/rtm/

Grid from 10 000 feet

Work regardless of geographical location, interact with colleagues, share and access data



The GRID: networked data processing centres and "middleware" software as the "glue" of resources.

Scientific instruments, libraries and experiments provide huge amounts of data

based on: Federico.Carminati@cern.ch

Why would we need it?

e-Science

Collected data in science and industry grows exponentially:

The Bible	**5**	**MByte**
X-ray image	5	MByte/image
Functional MRI	1	GByte/day
Bio-informatics databases	500	GByte each
Refereed journal papers	1	TByte/yr
Satellite world imagery	5	TByte/yr
US LoC contents	20	TByte
Internet Archive 1996-2002	100	TByte
Particle Physics today	5	PByte/yr
LHC era physics	**20**	**PByte/yr**

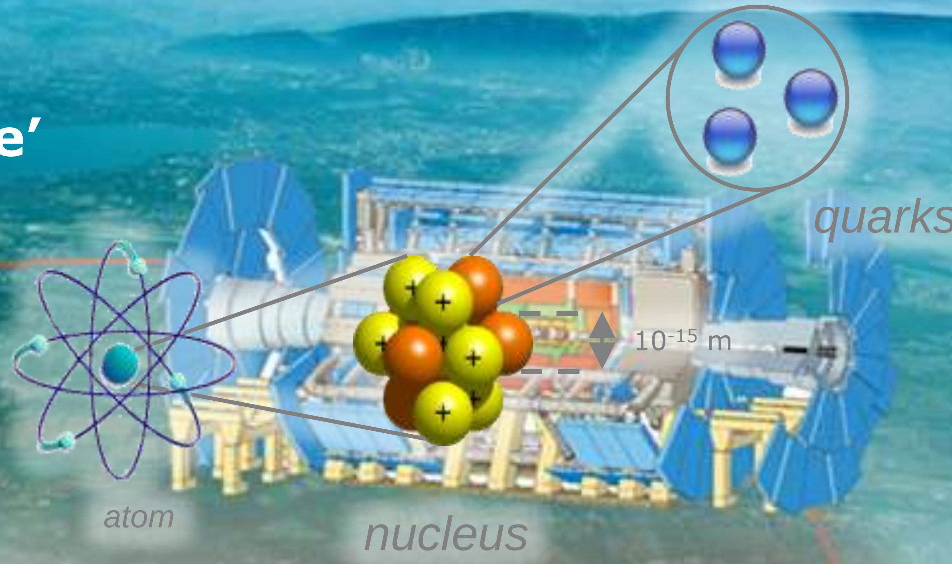
Enterprise

- Transaction processing
- Finance (what-if analyses)
- Pharma (in-silico drug design)
- Aerospace (fluid dynamics)

Some use cases: LHC Computing

Large Hadron Collider

- 'the worlds largest microscope'
- 'looking at the fundamental forces of nature'
- 27 km circumference
- Located at CERN, Geneva, CH



~ 20 PByte of data per year, ~ 40 000 modern PC style computers

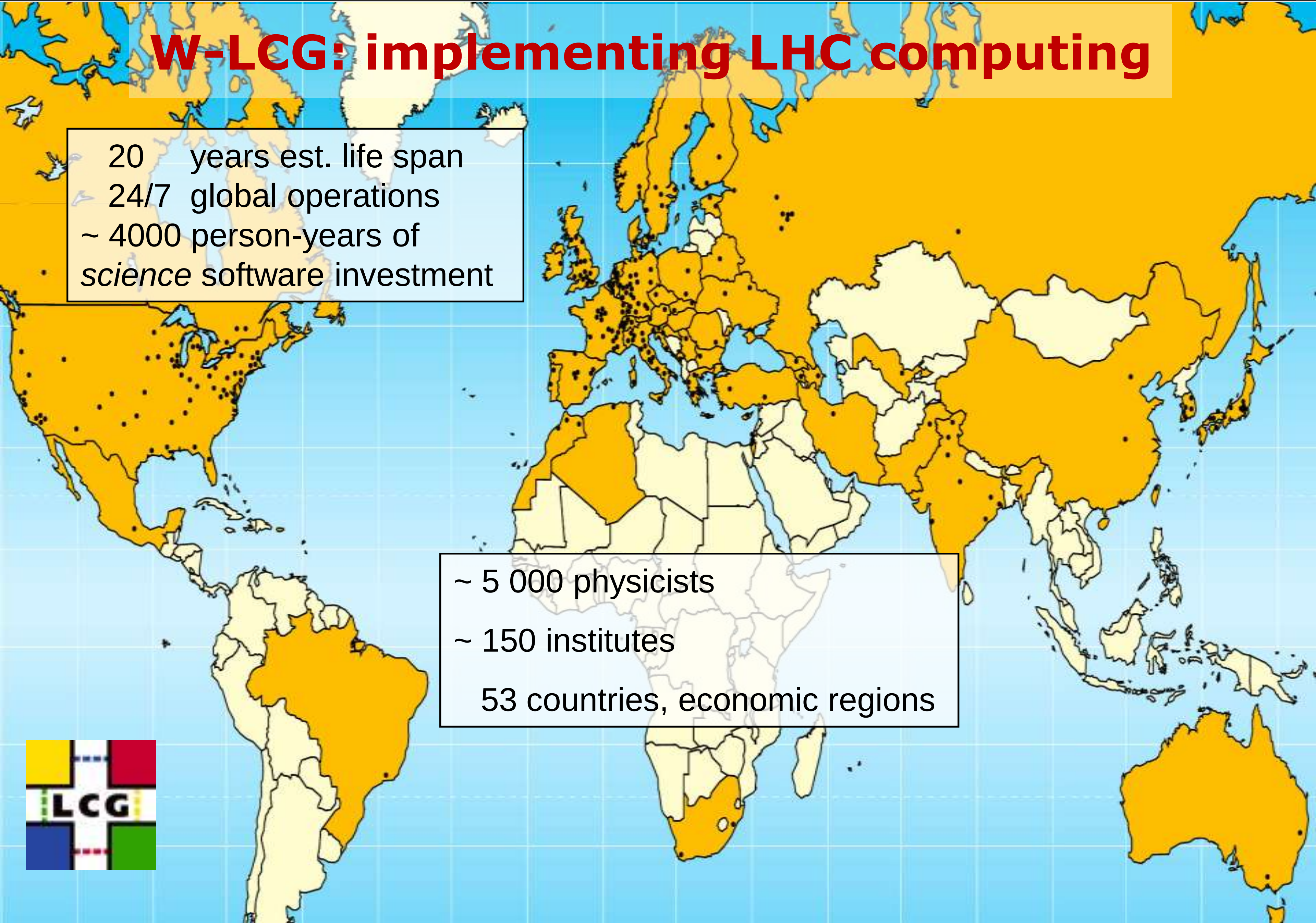
W-LCG: implementing LHC computing

20 years est. life span
24/7 global operations
~ 4000 person-years of *science* software investment

~ 5 000 physicists

~ 150 institutes

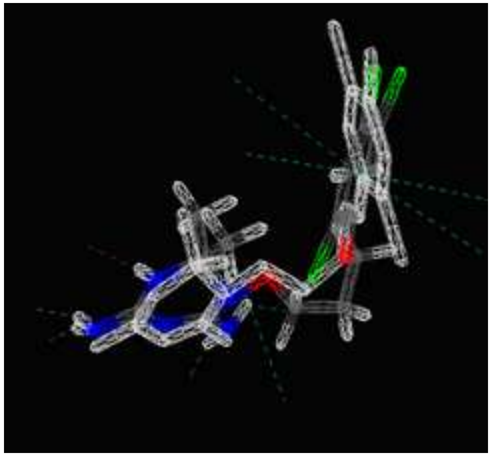
53 countries, economic regions



LCG

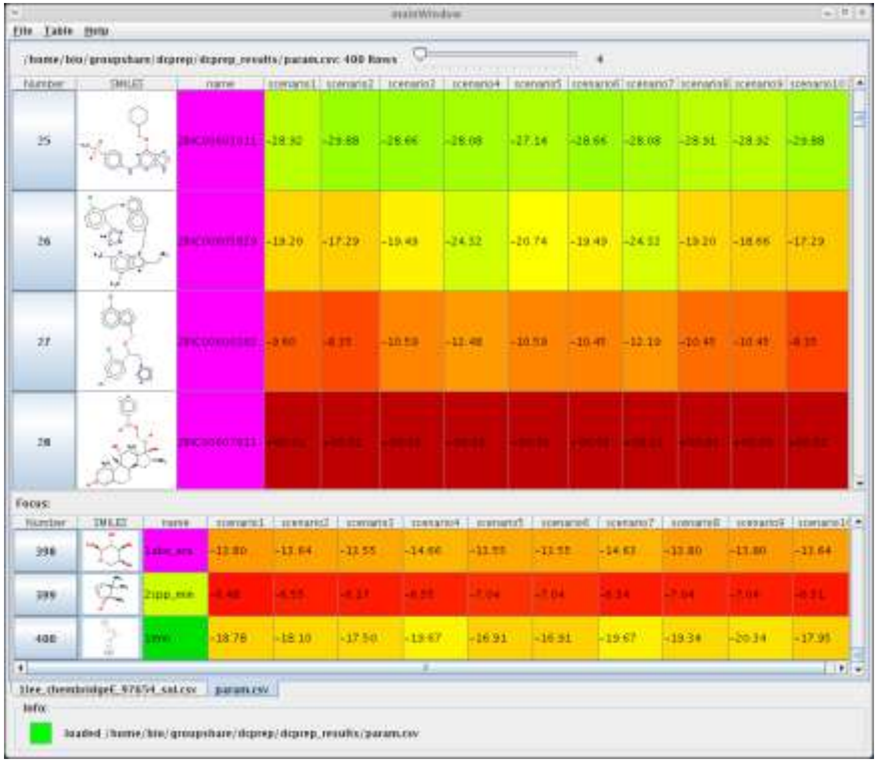
WISDOM: drug discovery

Wide-area In-Silico Docking On Malaria



over 46 million ligands virtually docked on malaria and H5N1 avian flu viruses in less than a month

used 100 *years* of CPU power
speedup ~ 100 times!



- **47 sites**
- **15 countries**

- 3000 CPUs
- 12 TByte disk

vl·e

egee Enabling Grids for E-sciencE

vl·e virtual laboratory for e·science

BiG Grid the dutch e·science grid

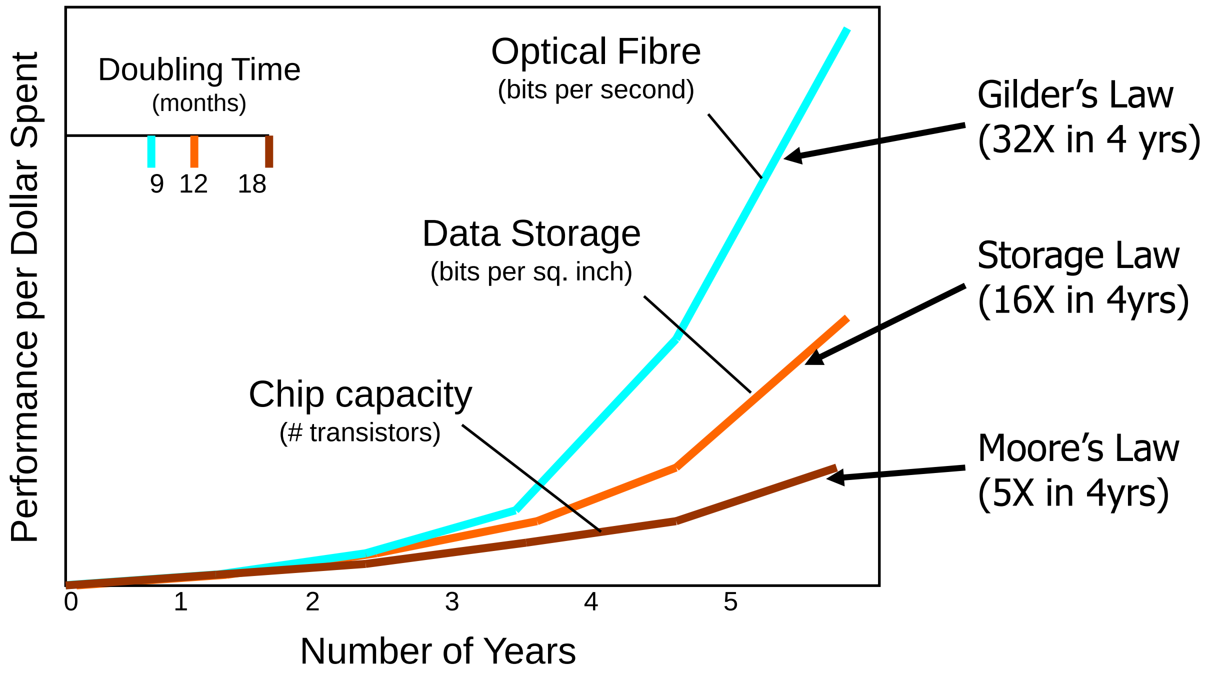
NIKHEF pdp

Building the Grid ...

Graphics: Real Time Monitor, Gidon Moont, Imperial College London, see http://gridportal.hep.ph.ic.ac.uk/rtm/

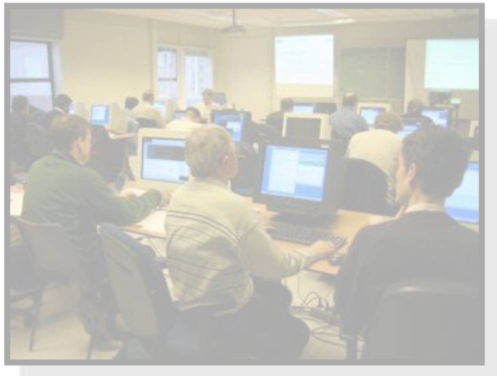
Why Grid computing – today?

- New applications need larger amounts of data or computation
- Larger, and growing, distributed user community
- Network grows faster than compute power/storage



Graphic: "The Triumph of Light", Scientific American, January 2001

What is Grid?

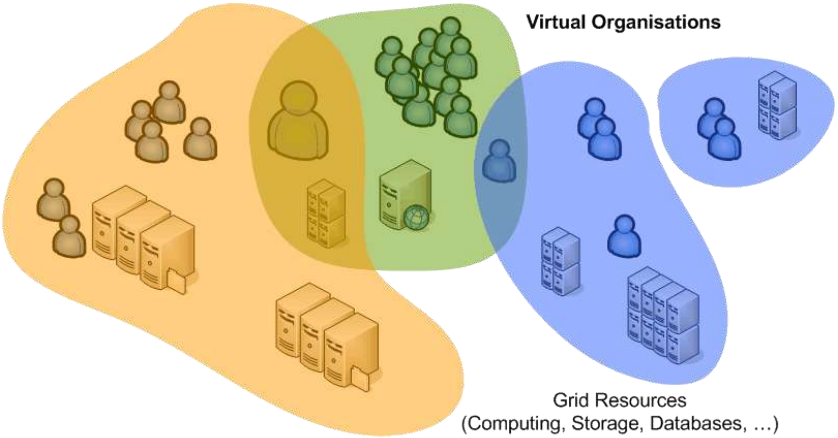


Cycle scavenging
- harvest idle compute power
- improve RoI on desktops

Cluster computing and storage
- What-if scenarios
- Physics event analysis
- Improve Data Centre Utilization

Cross-domain resource sharing
- more than one organisation
- more than one application
- more than one …

- open protocols
- collective service



Community Building
- authentication
- authorization
- virtual organizations

Scheduling and clustering
- resource management
- prioritization and fair-share

Hardware Infrastructures
- compute clusters
- disk and tape storage
- database services

Operational Security Policy
- distributed incident response
- policies

Managing Complexity
- systems management
- scaling
- multi-national infrastructures

Grid Structures

Definition of inter-organizational grids

Virtual Organizations

Security model

COMMUNITY BUILDING

Three essential ingredients for Grid

'inter-organizational resource sharing'

A grid combines resources that

- Are not managed by a single organization
- Use a common, open protocol … that is general purpose
- Provide additional qualities of service, *i.e.*, are usable as a collective and transparent resource

GRID today

DAILY NEWS AND INFORMATION FOR THE GLOBAL GRID COMMUNITY / JULY 22, 2002: VOL. 1 NO. 6

WHAT IS THE GRID? A THREE POINT CHECKLIST
By Ian Foster Argonne National Lab & University of Chicago

The recent explosion of commercial and scientific interest in the Grid makes it timely to revisit the question: What is the Grid, anyway? I propose here a three-point checklist for determining whether a system is a Grid. I also discuss the critical role that standards must play in defining the Grid.

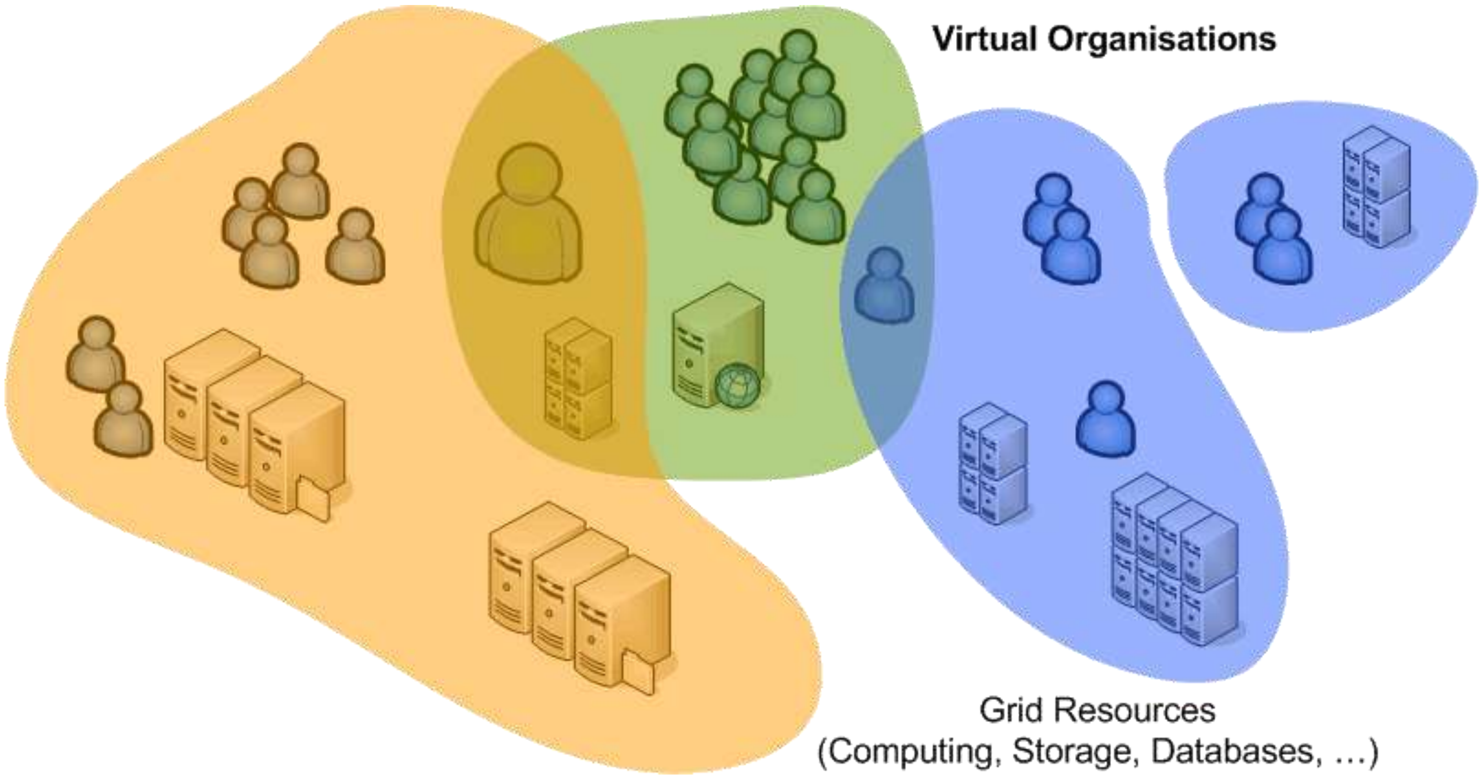
The Need for a Clear Definition Grids have moved from the obscurely academic to the highly popular. We read about Compute Grids, Data Grids, Science Grids, Access Grids, Knowledge Grids, Bio Grids, Sensor Grids, Cluster Grids, Campus Grids, Tera Grids, and Commodity Grids. The skeptic can be forgiven for wondering if

Source: Ian Foster in *Grid Today*, July 22, 2002; Vol. 1 No. 6, see http://www-fp.mcs.anl.gov/~foster/Articles/WhatIstheGrid.pdf

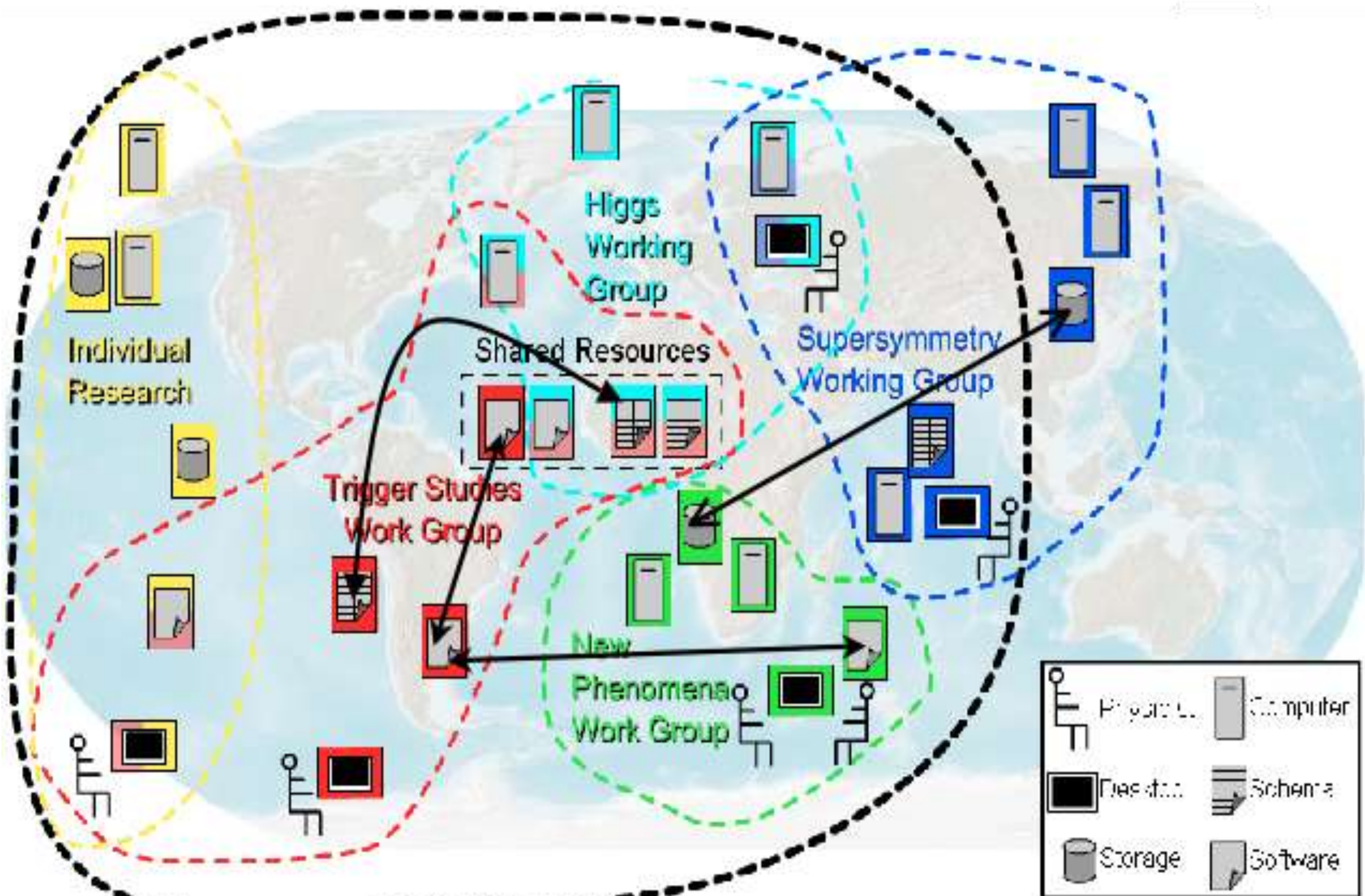
Virtual Organisations

The communities that make up the grid:

- **not under single hierarchical control**,
- (temporarily) **joining forces** to solve a particular problem at hand,
- bringing to the collaboration a subset of their resources,
- sharing those **at their discretion** and each **under their own conditions**.



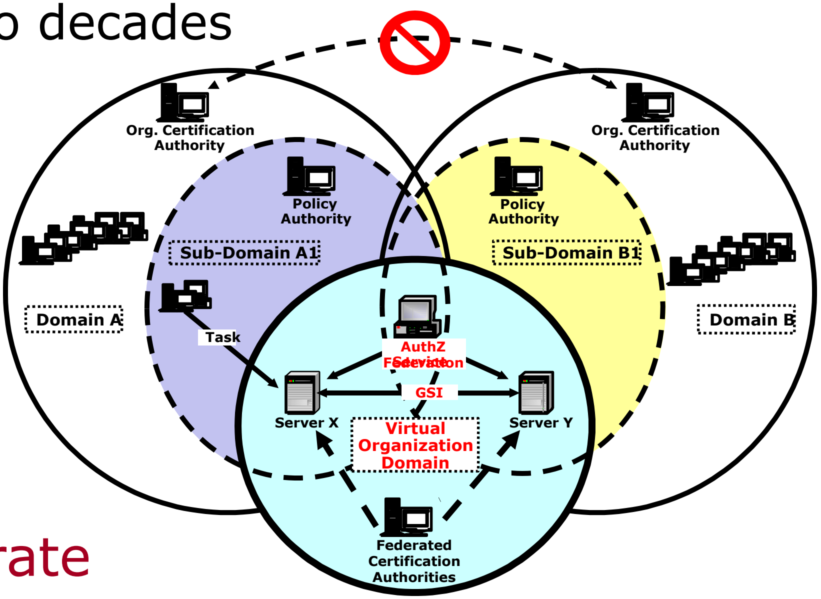
Although nothing is ever quite that neat …



graphic: Open Grid Services Architecture, © Global Grid Forum 2005, GFD.30

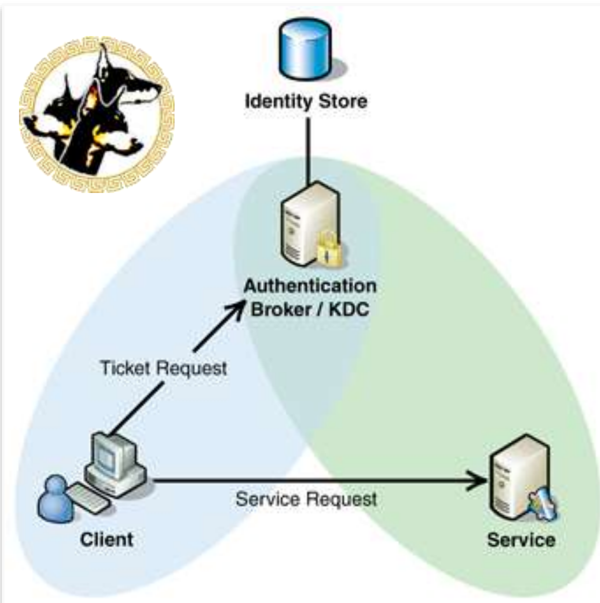
Federation in Grid Security

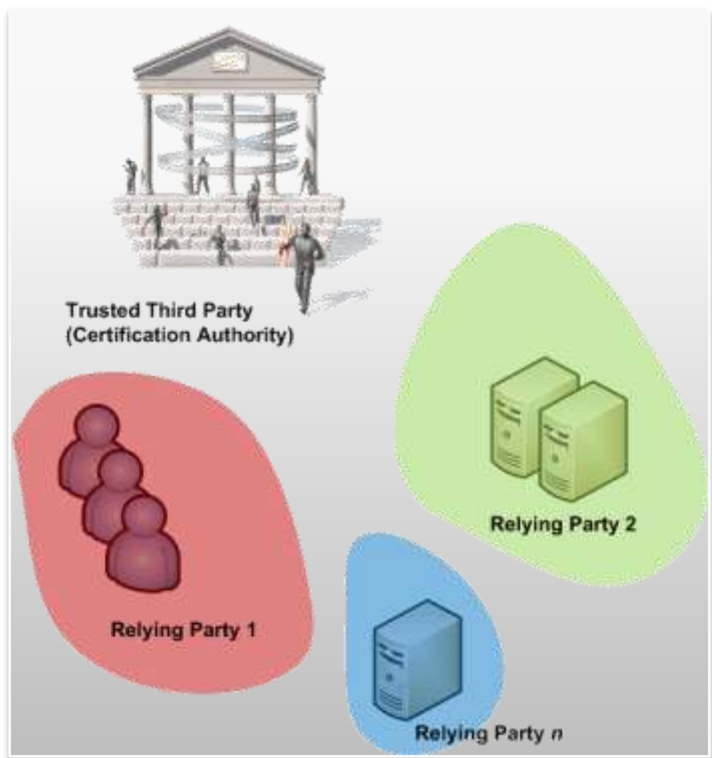
- There is no *a priori* trust relationship between members or member organisations!
  - VO lifetime can vary from hours to decades
  - VO not necessarily persistent (both long- and short-lived)
  - people and resources are members of many VOs



- but who to trust and how to federate
  - at the organisation level? eduroam™, inCommon, SWITCHaai, UK Access Mngt Federation
  - at the user and VO level? user AuthN and VO-centric AuthZ authorities 'orthogonal' to the org structure

Security Trust Mechanisms





Intra-organizational security vs. global grids

...

Direct (username-password) authN

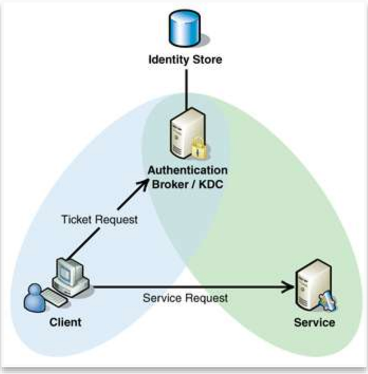
- Dedicated to each site where you want access
- Usually strongly linked to authorization
  - different accounts for different roles
- In a multi-organizational problem is

$$\mathcal{O}(n_{sites}) * \mathcal{O}(n_{users})$$

Federation technologies (see later) help in some respects

Kerberos

- Common trust domain around a KDC
- Based on service tickets, derived from a TGT
  - Encrypted with the service key from the target service
  - Whether you talk to the 'right' server is implicit in it's ability to decode your service ticket
- Cross-domain trust by recognizing KDC tickets
  - interesting in presence of symmetric crypto
  - but usually, alignment mismatch between organizations is the limiting factor
  - For multi-domain gets to be $\mathcal{O}(n^2)$ for *n* sites

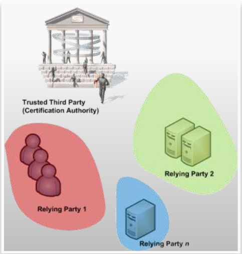


PKI

- Relying parties (sites and users) all recognise a trusted third party (CA)
- Problem is now $\mathcal{O}(n_{CA})$

  and n_{CA} is hopefully $<< n_{sites}$

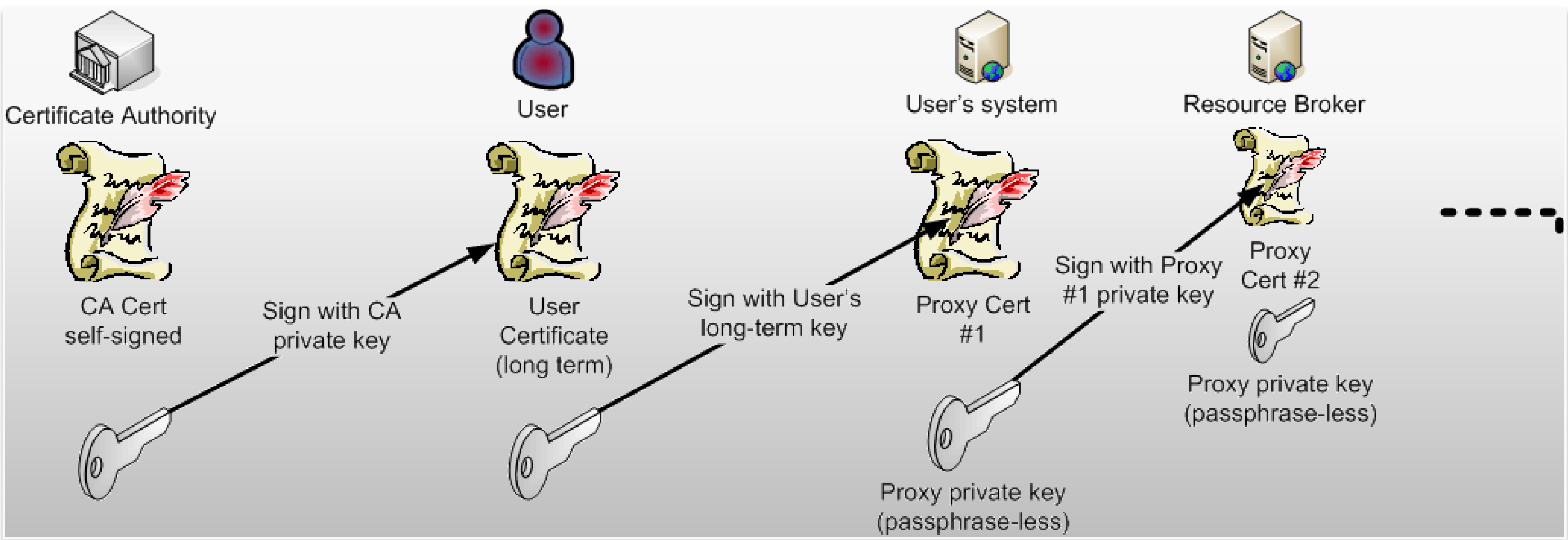
- But there will be more than one CA as well …



Delegation – carrying identity and rights forward

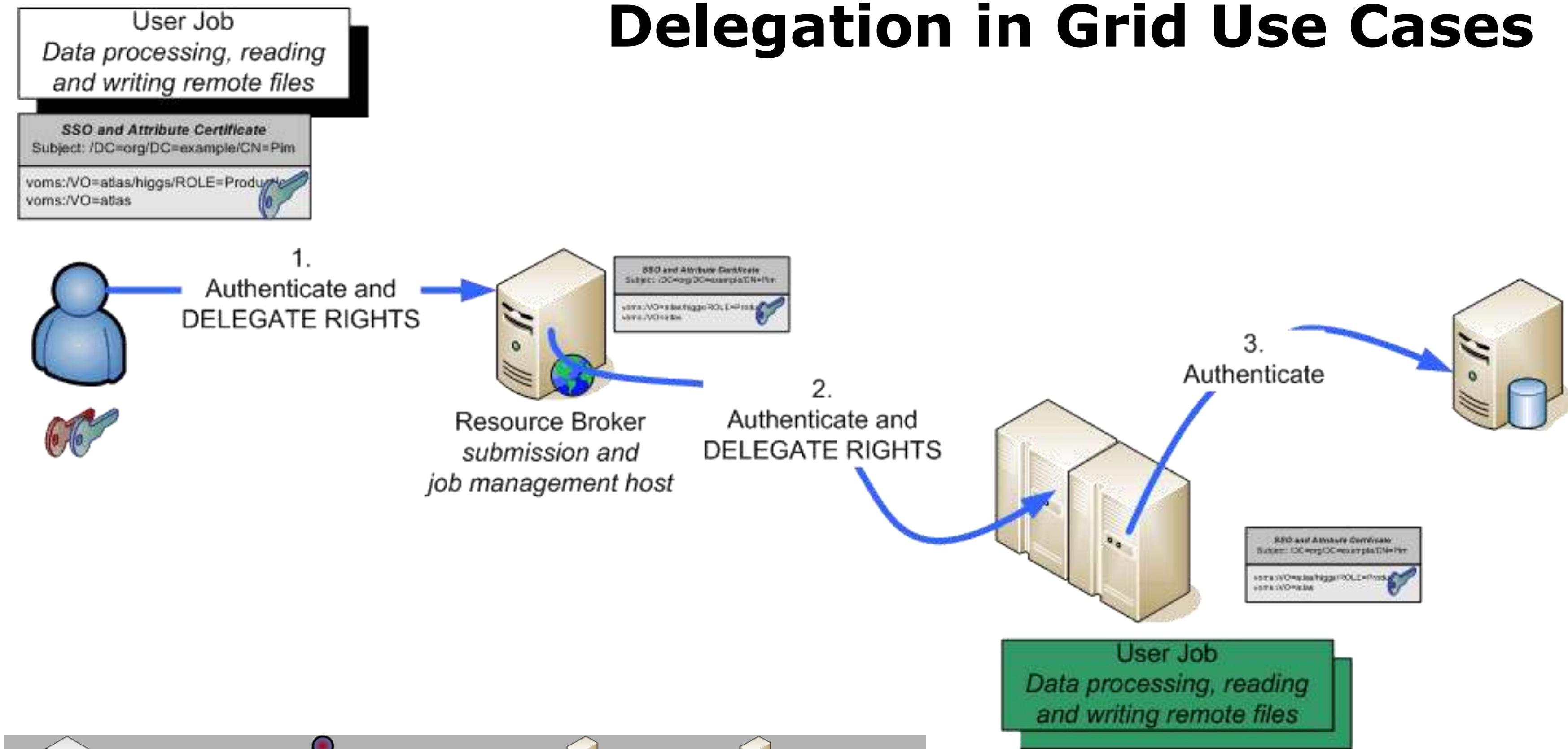
Single sign-on in a end-user PKI environment

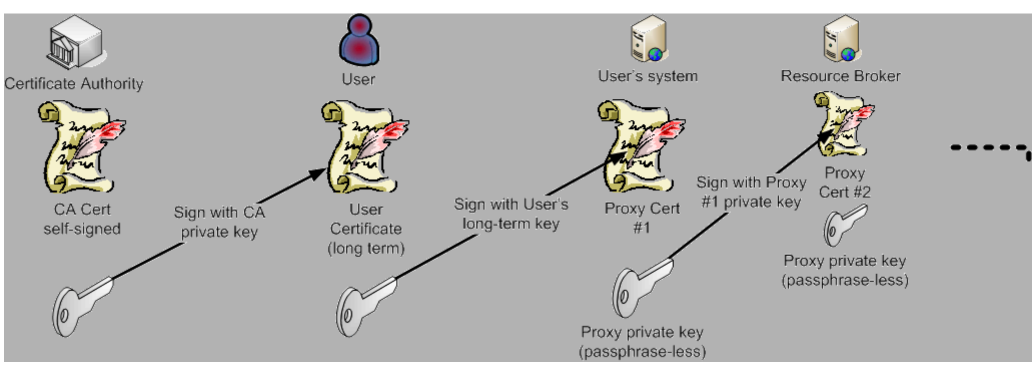
and a carrier for VO membership information



RFC 3820 "Proxy Certificates"

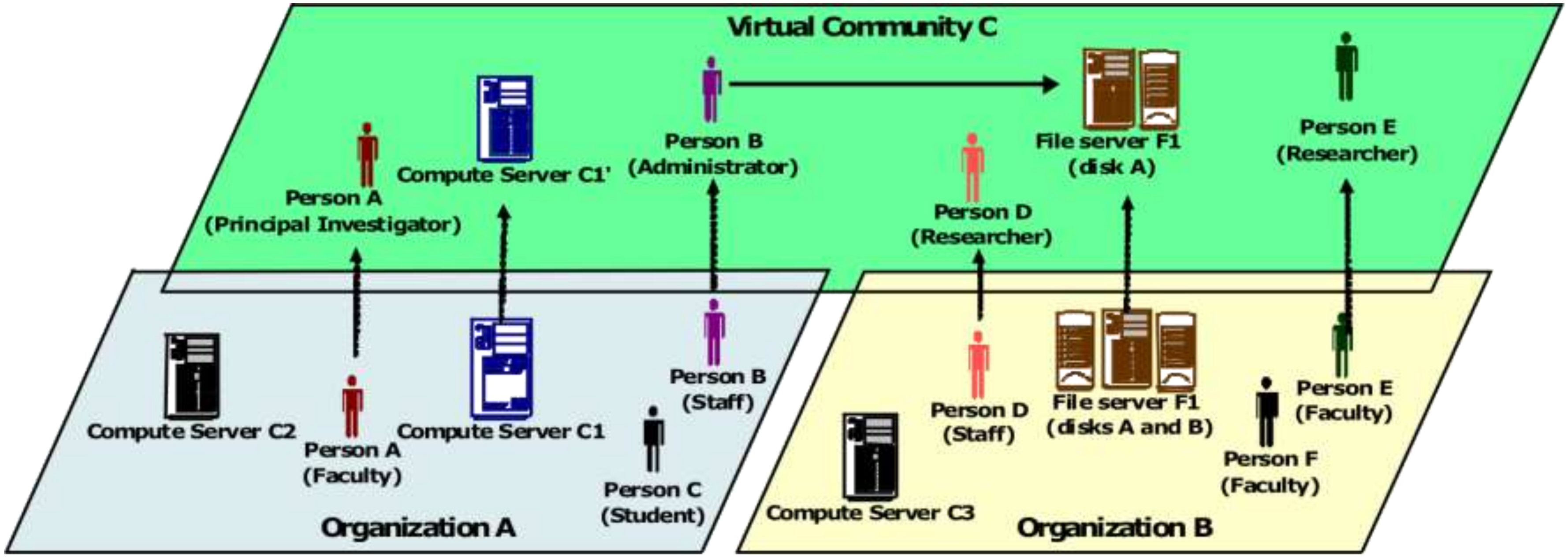
Delegation in Grid Use Cases





Organizing people

'Identity is not enough'



'virtual' organization roles are independent of home organization roles and **authority for the VO roles rests with the VO**

Graphic: Open Grid Forum OGSA Working Group, GFD.30

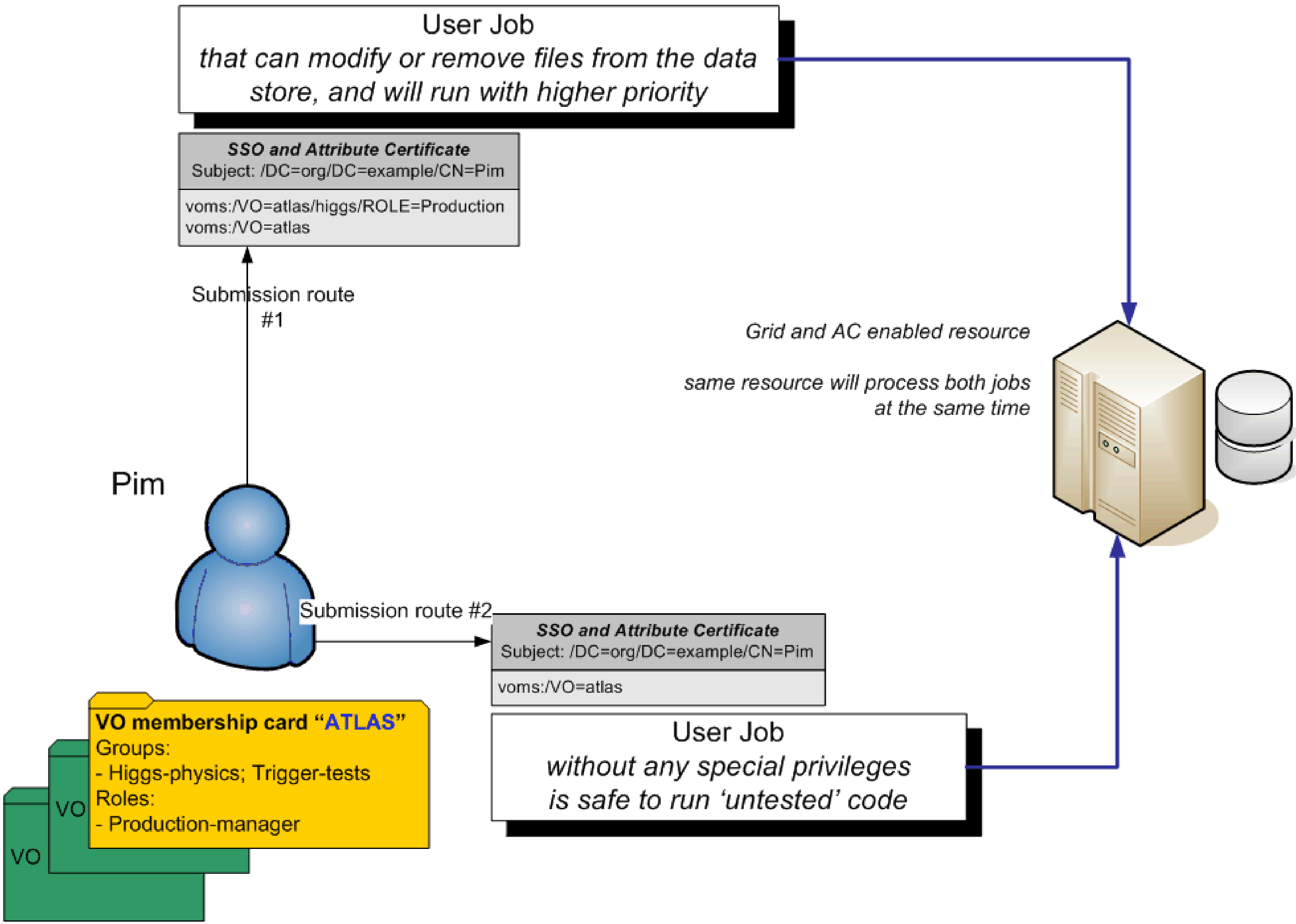
Authentication vs. *Authorization*

For user-centric delegation and VO-based grids

- **Single** Authentication token ("passport")
  - issued by a party trusted by all,
  - recognised by many resource providers, users, and VOs
  - satisfy traceability and persistency requirement
  - in itself does not grant any access, but provides a unique binding between an identifier and the subject

- Per-VO (per 'UHO') Authorisations ("visa") attributes
  - granted to a person/service via a virtual organisation
  - based on the 'passport' name
  - embedded in the single-sign-on token (proxy)
  - acknowledged by the resource owners
  - providers can obtain lists of authorised users per VO, but can still ban individual users

Role-based access control

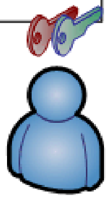


VOMS: Assertions in X.509 AC or SAML

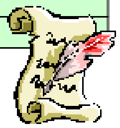
Virtual Organisation Management System (VOMS)

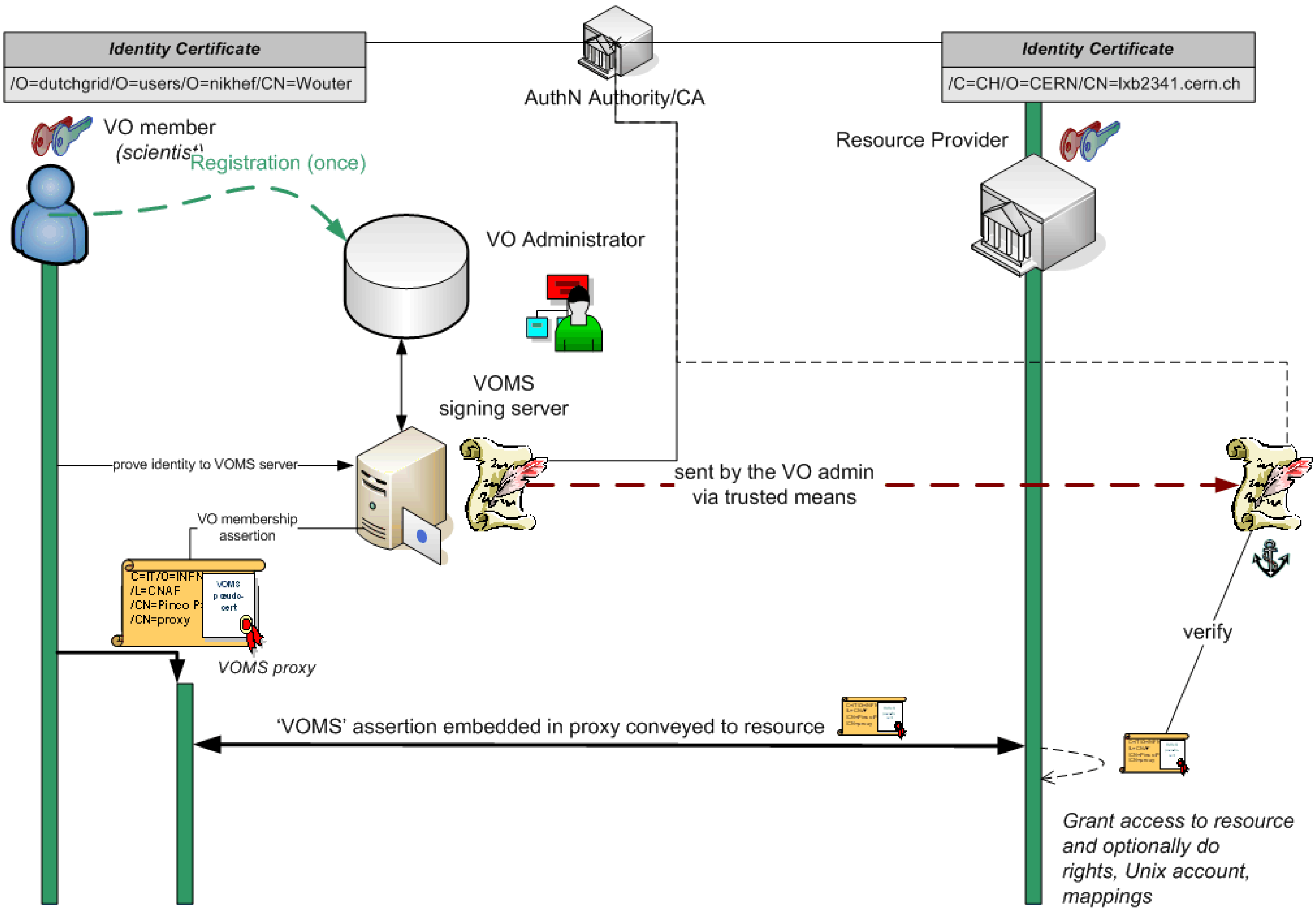
- push-model for signed VO membership tokens
  - using the traditional X.509 'proxy' certificate for shipping

VOMS proxy with embedded VO assertion
Serial Number: 26423 (0x6737)
Issuer: O=dutchgrid, O=users, O=nikhef, CN=David Groep
Not Before: Oct 16 12:46:28 2006 GMT
Not After : Oct 17 00:51:28 2006 GMT
Subject: O=dutchgrid, O=users, O=nikhef, CN=David Groep, CN=proxy
Subject Public Key Info:
Public Key Algorithm: rsaEncryption
RSA Public Key: (512 bit)
X509v3 extensions:
1.3.6.1.4.1.8005.100.100.5:
0...0...0...0......0W.U0O.M0K1.0...U./dteam/ne/ROLE=null/0...0...0...0
X509v3 Key Usage:
Digital Signature, Key Encipherment, Data Encipherment
Signature Algorithm: md5WithRSAEncryption



Attribute Certificate	
INTEGER	1
SUBJECT	/O=dutchgrid/O=users/O=nikhef/CN=David Groep
SERIAL	0396
ISSUER	/C=CH/O=CERN/CN=lcg-voms.cern.ch
OCTET STRING	/dteam/Role=NULL/Capability=NULL
OCTET STRING	/dteam/ne/Role=NULL/Capability=NULL
OBJECT	No revocation available
AuthorityKeyIdentifier	0....H....0.....<3...#..
SignatureAlgorithm	md5WithRSAEncryption




Identity Certificate
/O=dutchgrid/O=users/O=nikhef/CN=Wouter
AuthN Authority/CA
Identity Certificate
/C=CH/O=CERN/CN=lxb2341.cern.ch
VO member
(scientis
Registration (once)
Resource Provider
VO Administrator
VOMS
signing server
prove identity to VOMS server
sent by the VO admin
via trusted means
VO membership
assertion
C=IT/O=INFN
/L=CNAF
/CN=Pinco P
/CN=proxy
VOMS
pseudo-
cert
VOMS proxy
verify
'VOMS' assertion embedded in proxy conveyed to resource
Grant access to resource
and optionally do
rights, Unix account,
mappings

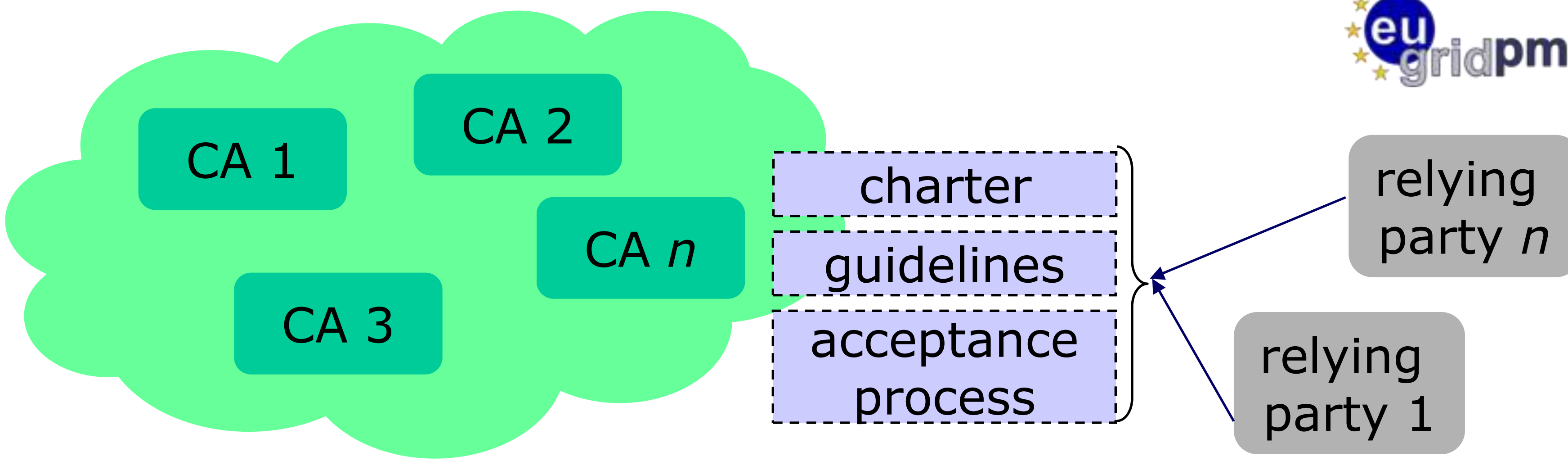
Federation: the IGTF

Home-organization based

User centric

FEDERATION AND CONFEDERATION

Federated PKI for authentication

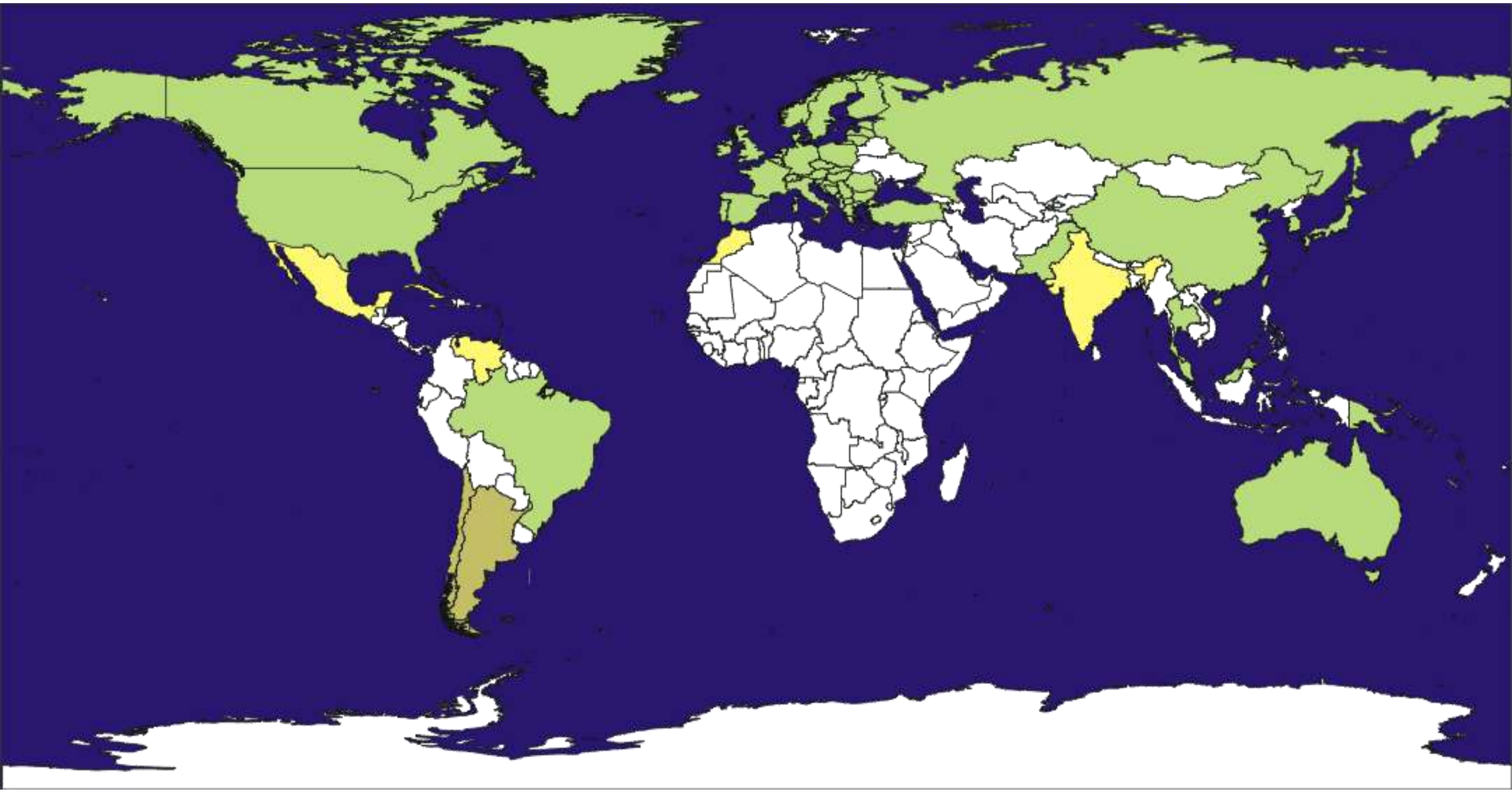


- A Federation of many independent CAs (a ‘policy bridge’)
  - common minimum requirements
  - trust domain as required by users and relying parties
  - well-defined and peer-reviewed acceptance process
- User has a single identity
  - from a local CA close by
  - works across VOs, with single sign-on via impersonation ‘proxies’ (RFC3820)
  - certificate itself also usable outside the grid

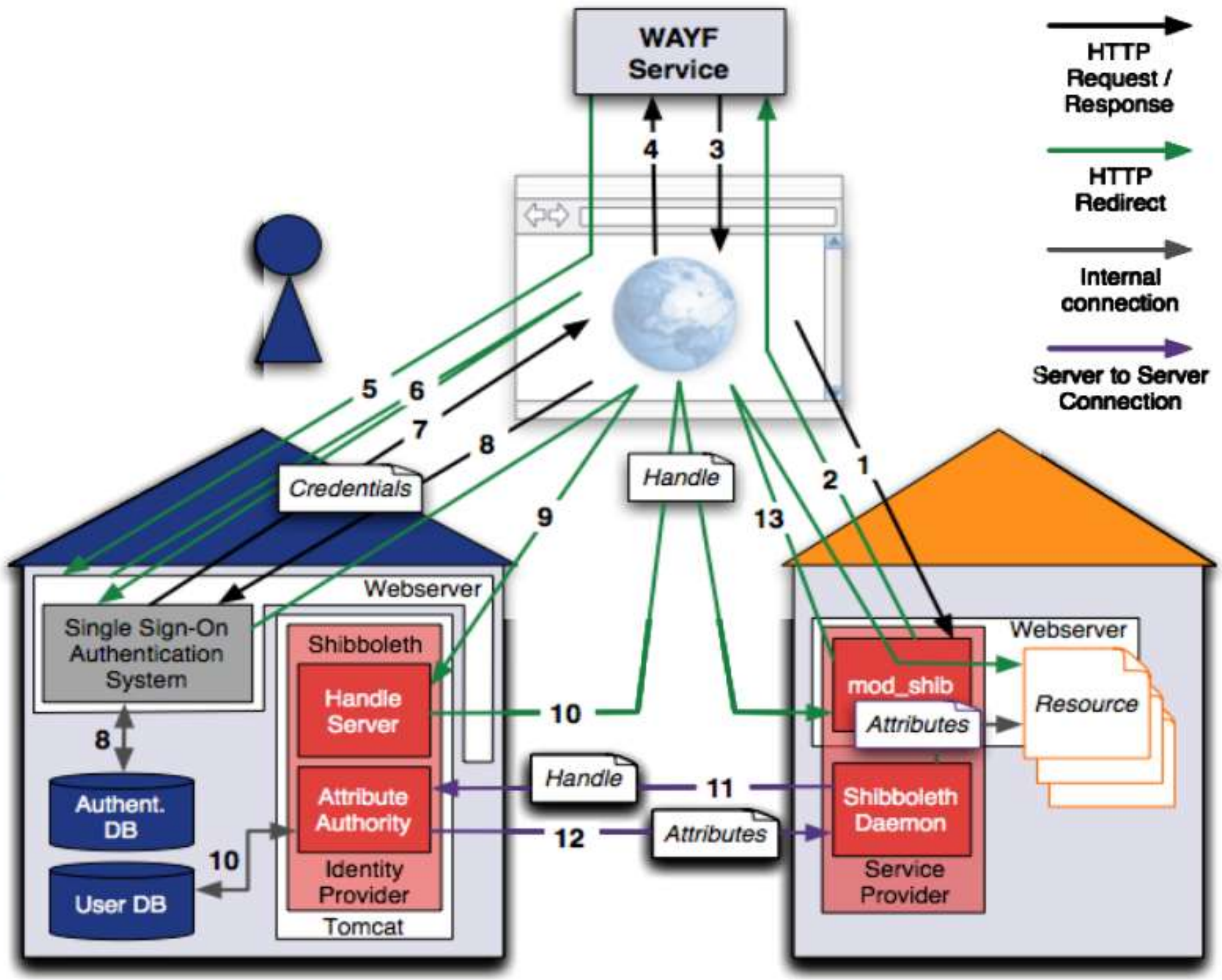
International Grid Trust Federation and EUGridPMA, see http://www.gridpma.org/

IGTF
International Grid Trust Federation
AP | EU | TAG

Federation of 3 Regional "PMAs", that define common guidelines and accredit credential-issuing authorities



Federation alternatives
hiding PKI from users



Federated Authentication

- Users authenticate to their home organization
- There they have a set of attributes
  - With a release policy
  - Home organisation authoritative for them
- Service Providers make access decision based on the attributes related to an abstract handle
  - User’s name (eduPersonPrinciple Name) is also an attribute
- But the home organisation cannot make assertions about VO membership
  - We need to move to a multi-authority world

Federation techniques getting popular

OpenID

A-SELECT

based around web services security protocols and SAML assertions

User Centric Identity?

- CardSpace,
  project Higgins,
- ...

see, e.g., Burton Group's blog
http://identityblog.burtongroup.com/

- Based on Web Services and 'SAML' assertions
  - Self-assertions
  - Assertions 'filled in' by trusted third parties, such as Visa, MC, etc.
- Required assurance depends on the target system

- *Interop testing just starting, see, e.g.* *http://identityblog.burtongroup.com/bgidps/2007/08/recapping-the-c.html*
- *Kim Cameron's Identity blog*

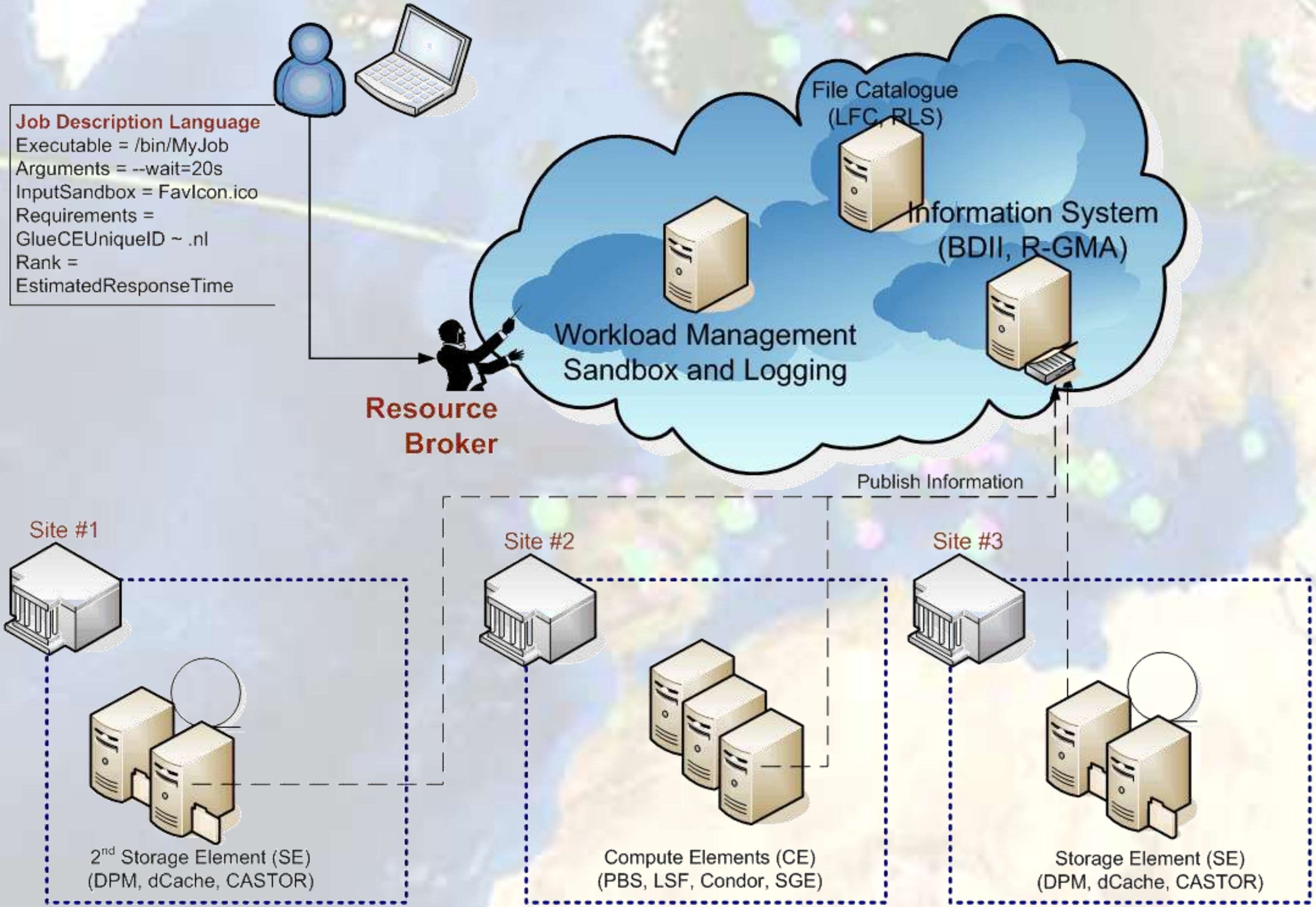
Accessing resources
Resource Brokering
Data access
Resource Information Systems

USING THE GRID

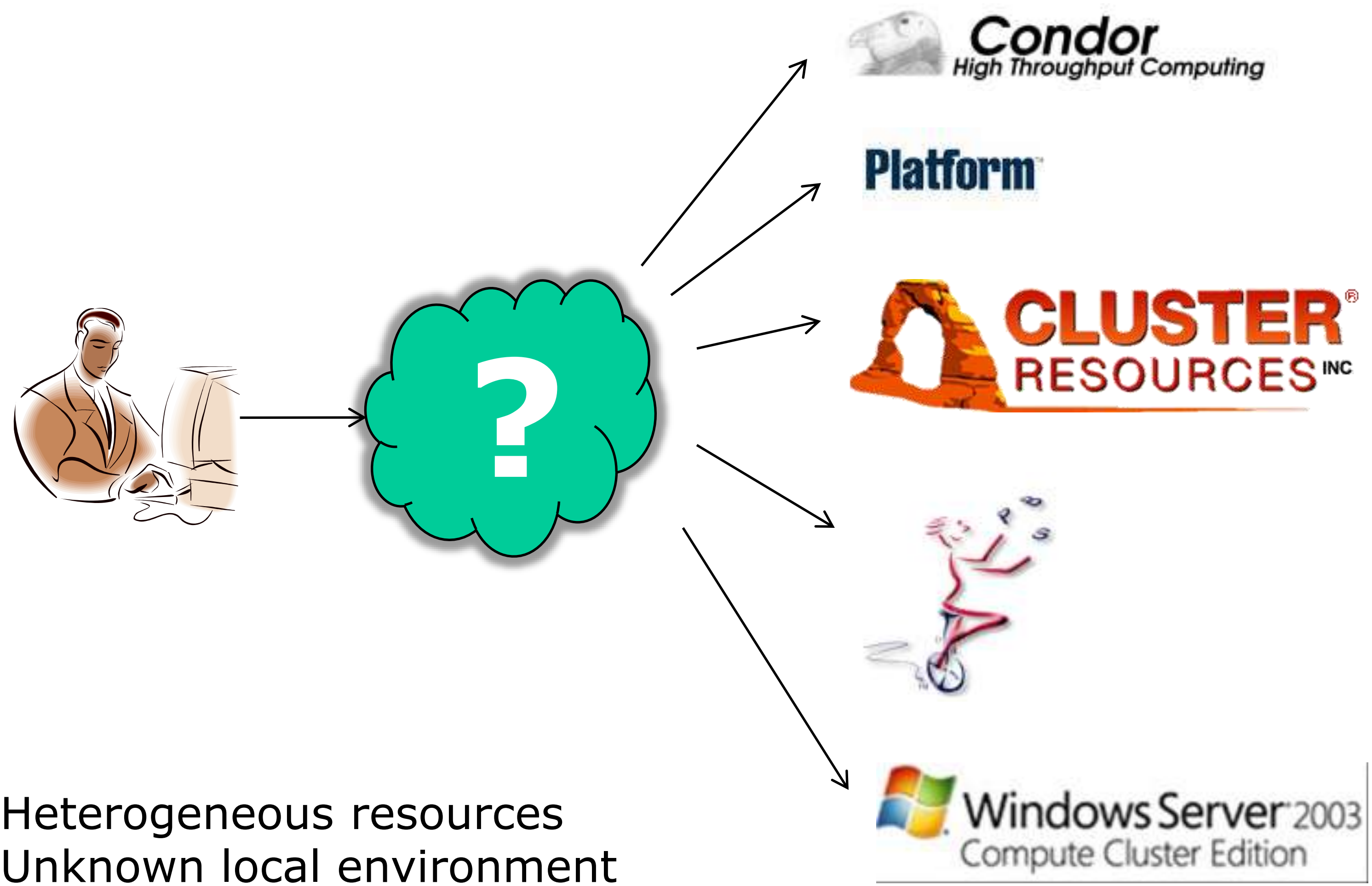
Services in a Grid

- Computing Element "front-end *service* for (set of) computers"
  - Cluster computing: *typically Linux with IP interconnect* with a 'head node' that batches or forwards requests to the cluster
  - Capability computing: *typically shared-memory supercomputers*
- Storage "front-end *service* for disk or tape"
  - Both disk and tape based
    - Varying retention time, QoS, uniformity of performance
  - Expressing ACLs in grid terms in challenging: *mapping of grid authorization to e.g. POSIX ACLs*
- File Catalogues … naming (data) objects in the Grid
  - for the really courageous people: represent computing, storage and data all as 'named objects' in a single 'grid name space'
- Information System … finding out resources on the Grid
  - Directory-based for static information
  - Monitoring and bookkeeping for real-time information
- Resource Broker …
  - Matching user job requirements to offers in the information system
  - WMS allows disconnected operation of the user interface

Typical grid topology for computational jobs



But you are not there yet …



- Heterogeneous resources
- Unknown local environment
- Unknown access policies

Computing: user expectations?

- Different user scenarios are possible and valid
  - paratrooper mode: come in, take all your equipment (files, executable &c) with you, do your thing and go away
    - you're supposed to clean up, but the system will likely do that for you if you forget. In all cases, garbage left behind is likely to be removed
  - two-stage 'prepare' and 'run'
    - extra services to pre-install environment and later request it
    - see later on such Community Software Area services
  - don't think but just do it
    - blindly assume the grid is like your local system
    - expect all software to be there
    - expect your results to be retained indefinitely
    - … realism of this scenario is unclear for 'production' grids
      - it does not scale to larger numbers of users
      - but large user communities hold 'power' over the resource providers (or the customers run away)

Submission

Basic operations

- Direct run/submit
  - useless unless you have an environment already set up
- Cancel
- Signal
- Suspend
- Resume
- List jobs/status
- Purge (remove garbage)
  - retrieve output first …

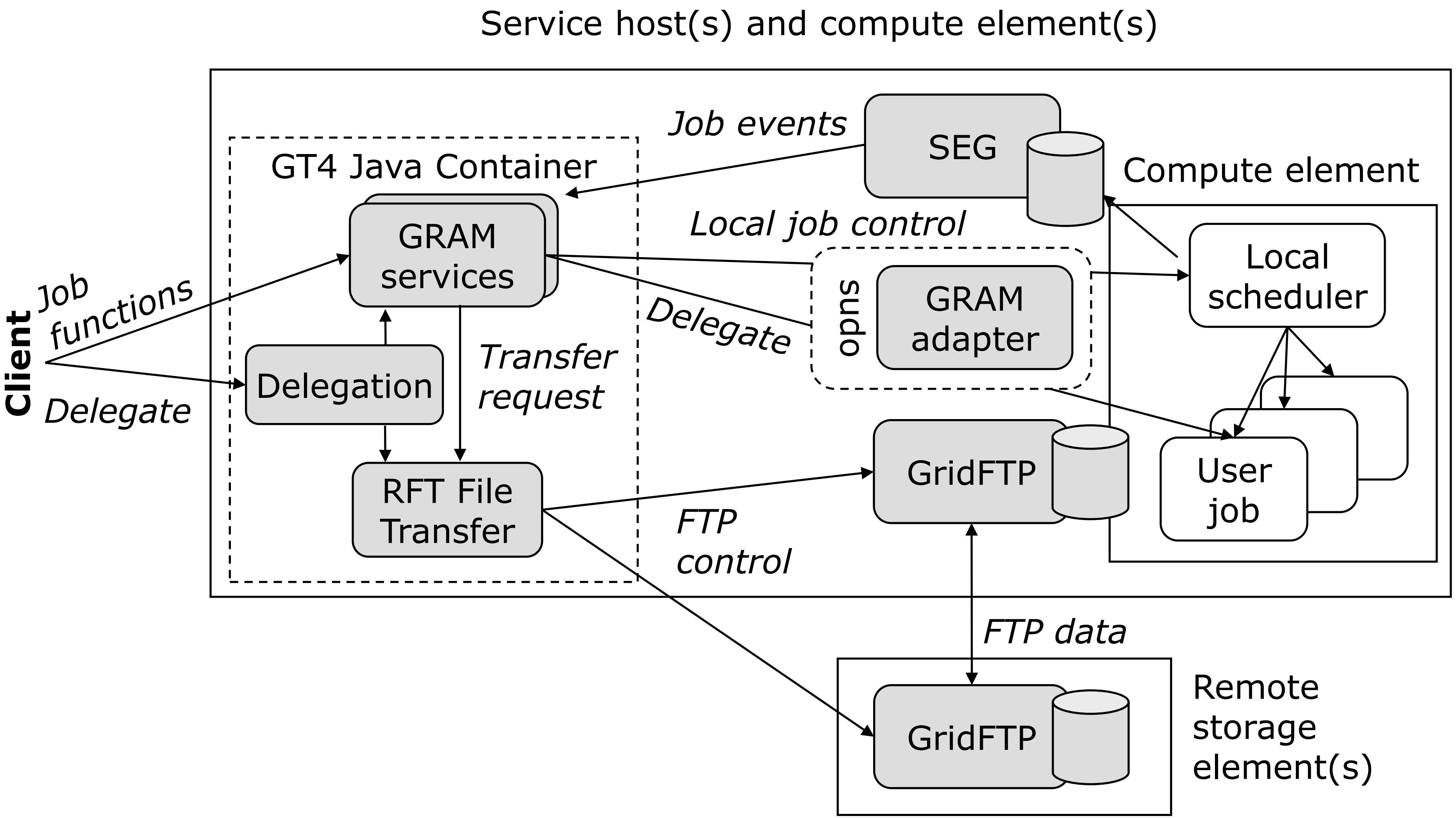
Other useful functions

- Assess submission (eligibility, estimated response time)
- Register & Start (needed if you have 'sandboxes')

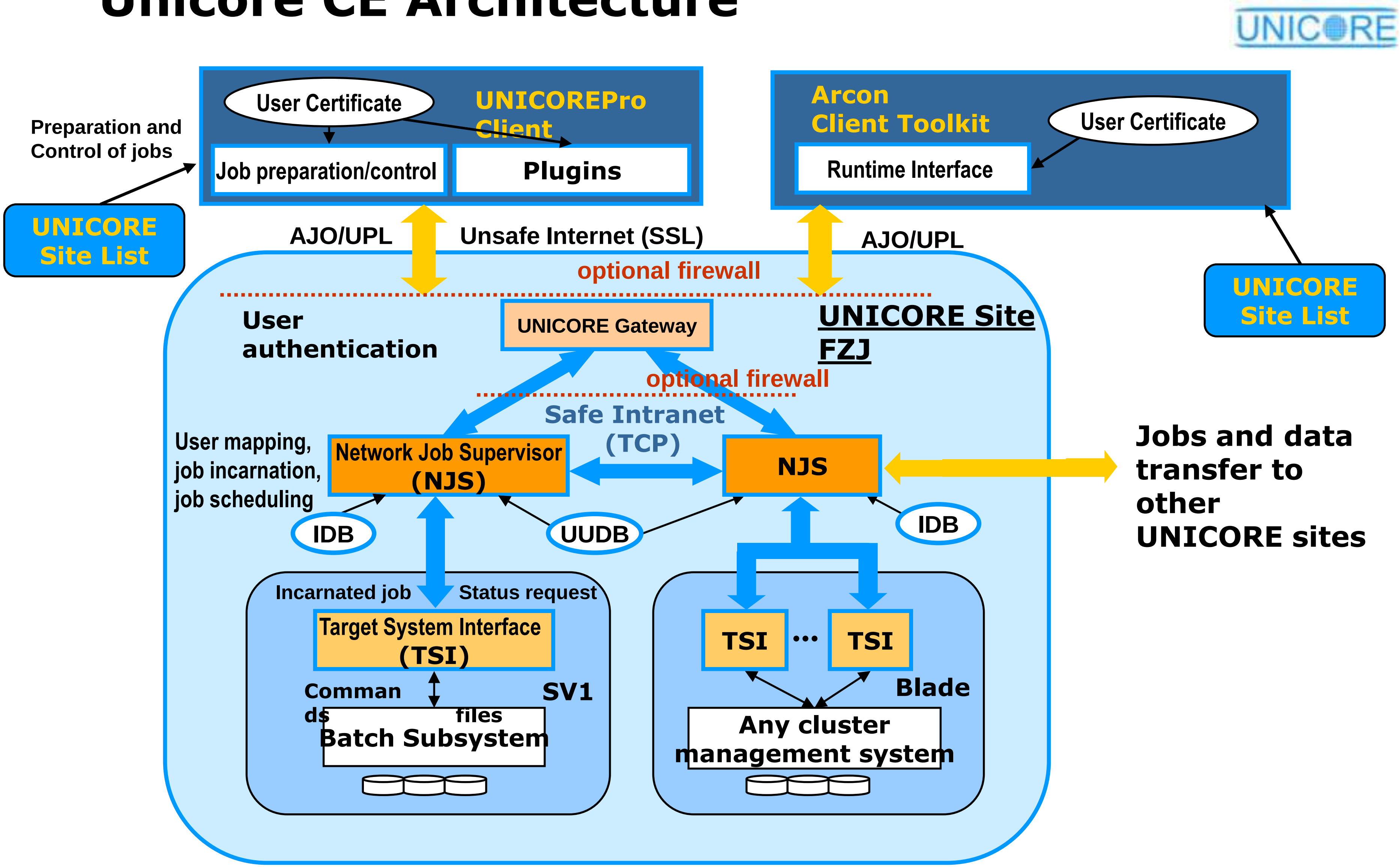
A seemingly simple task

- The CE seems conceptually simple
  - submit a job
  - wait for it to run
  - retrieve the results
  - or kill it prematurely …
  - *but:* there are a bazillion ways to implement it
    - with implicit or explicit data staging
    - hide the entire site structure and use forwarding nodes
    - or even allow automatic forwarding to another site
    - policy and prioritization
- the user does not want to know the difference
  - and an automatic resource broker needs a backend for every type
- back-end is usually just a simple old batch system

GT4 WS GRAM Architecture

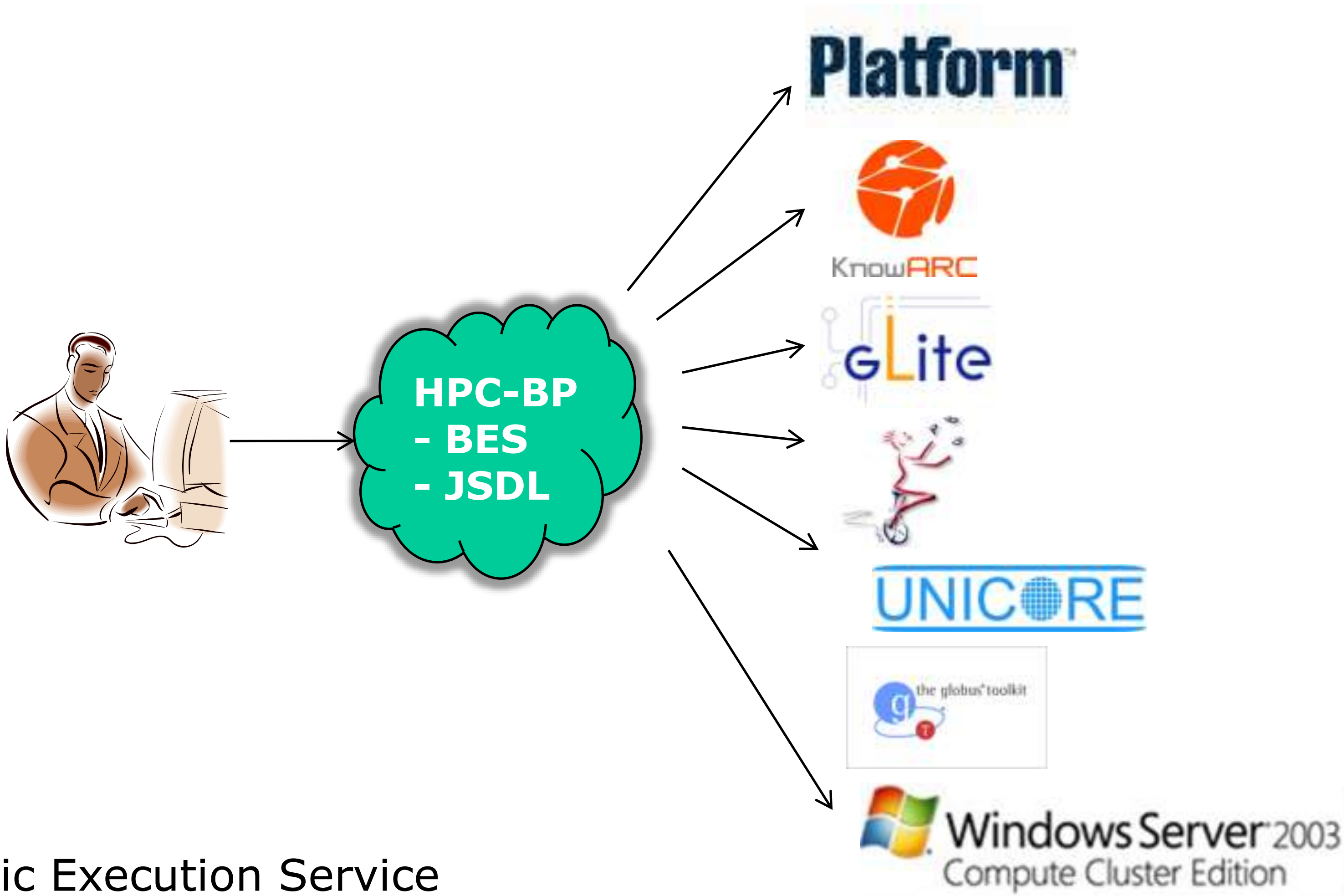


Unicore CE Architecture



Graphic from: Dave Snelling, Fujitsu Labs Europe, “Unicore Technology”, Grid School July 2003

Interoperability – but only basics at first



Basic Execution Service

- Job submission works, but
- security model, file staging, etc. still need to be resolved

Storage

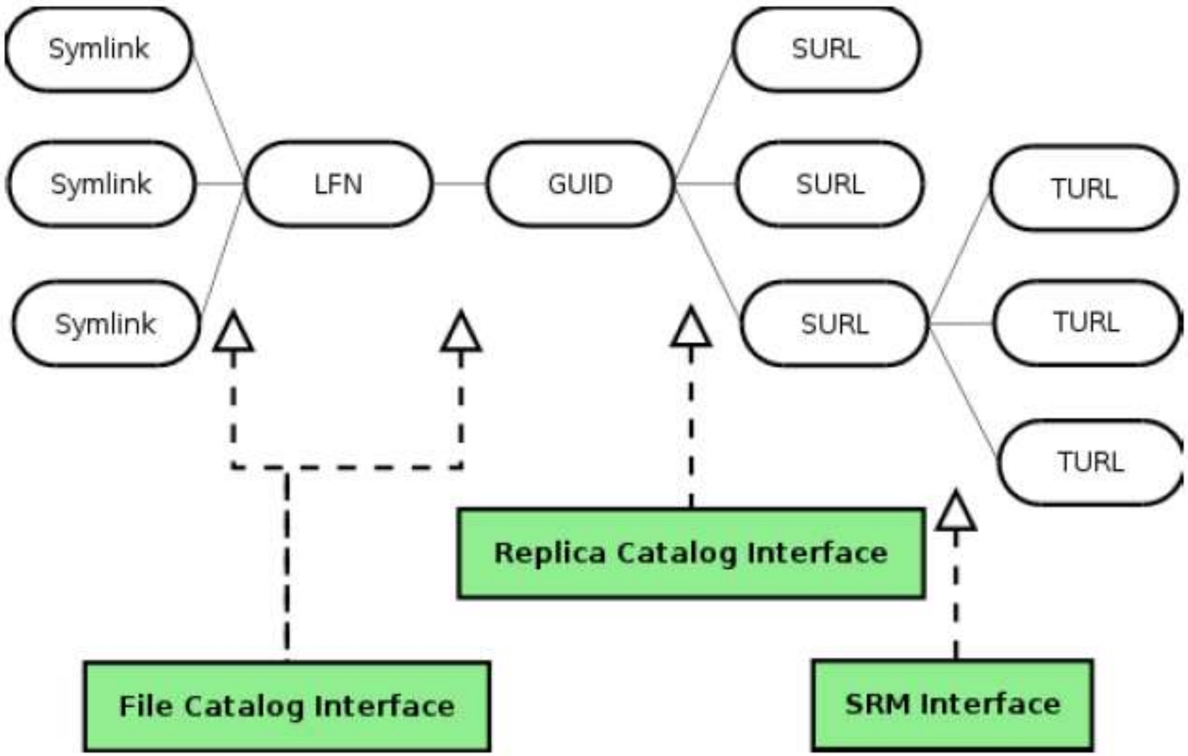
- Also here: different types and back ends
  - Simple disks: file system, GPFS, Lustre, …
  - MSS: DMF, HPSS, dCache/Enstore, CASTOR, …

- Separate functions of storage
  1. Presentation: file system view, logical naming
  2. Storage resource management:
     relocation, pinning, routing -- SRM
  3. Transfer protocols:
     GridFTP, Byte-IO, gsidcap, gsirfio
  4. Storage:
     file system, tape libraries

  - Today, the grid interfaces expose all of these levels …
    and, e.g., NFSv4 tried to combine all of that …

  - This will see a lot of change the coming time

Storage layering and interfaces



graphic: Peter Kunszt, EGEE DJRA1.4 gLite Architecture

How to you see the Grid?

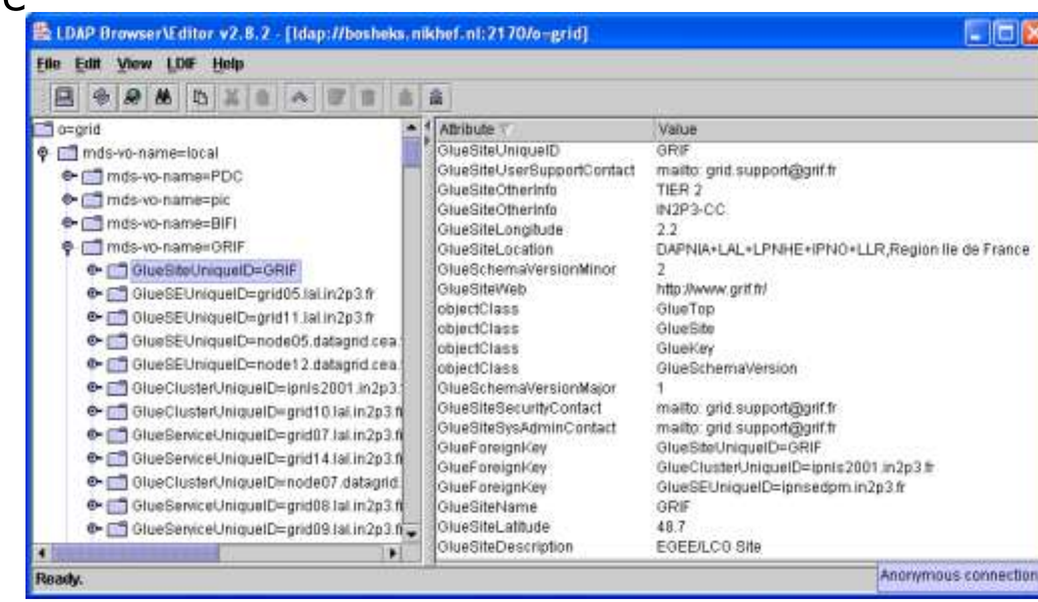
Broker matches the user's request with the site

- 'information supermarket' matchmaking (using Condor Matchmaking)
- uses the information published by the site

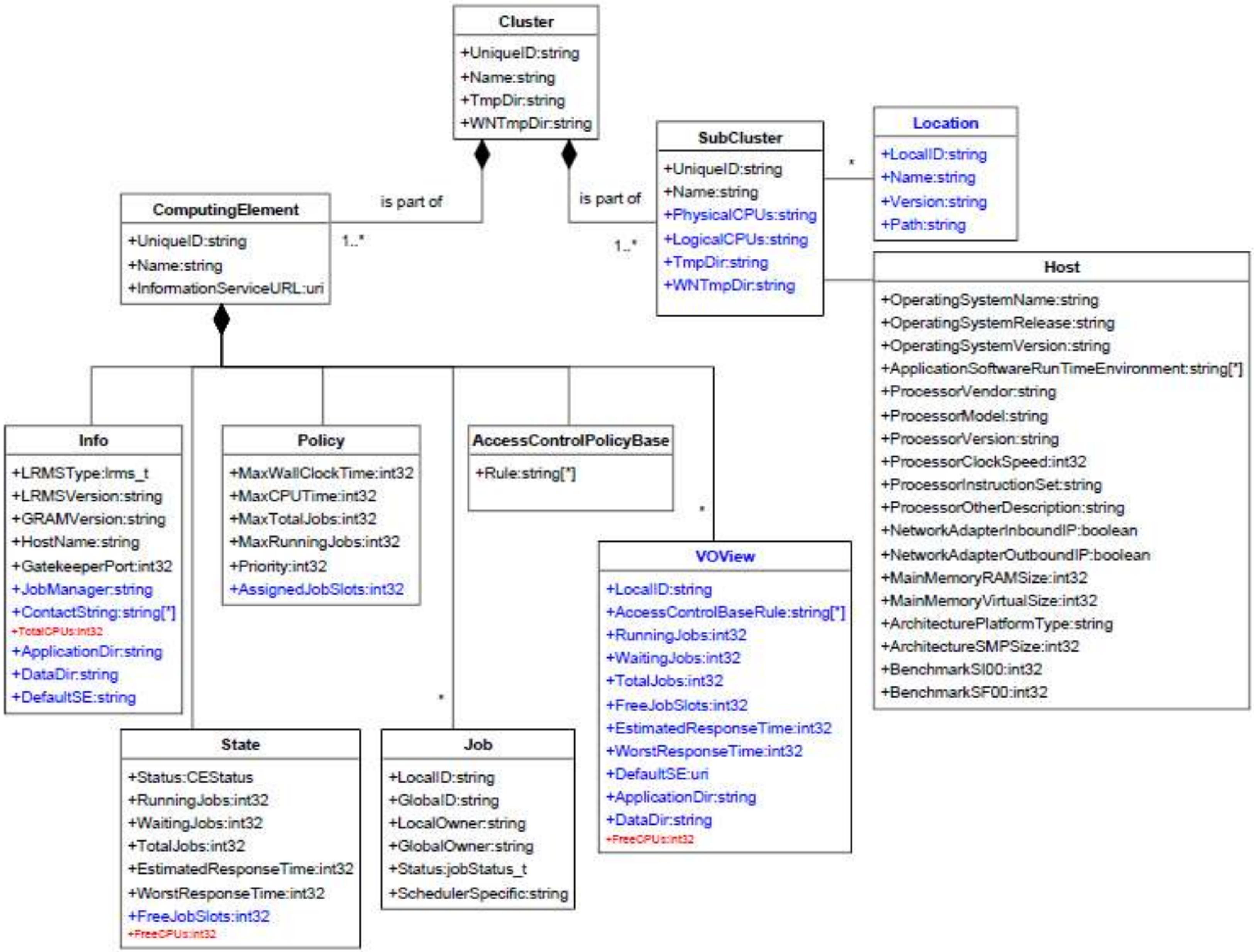
Grid Information system

'the only information a user ever gets about a site'

- So: should be 'reliable', consistent* and complete*
- Standard schema (GLUE) to describe sites, queues, storage *(complex schema semantics)*
- Usually presented as an LDAP directory



LDAP Browser Jarek Gawor: www.mcs.anl.gov/~gawor/ldap



From: the GLUE Information Model version 1.2, see document for details

Information system and brokering issues

- Size of information system scales with #sites and #details
  - already 12 MByte of LDIF
  - matching a job takes ~15 sec
  - Static and dynamic information is mixed ← this is ReallyBad™
- Scheduling policies are infinitely complex
  - no static schema can likely express this information
  - but negotiation processes take time at each request
    WS-Agreement is not really popular, at least not yet …
- Much information (still) needs to be set-up manually
  … *anything human will go wrong*

⇒ The info system is the single most important grid service

- Broker tries to make optimal decision based on this information
  … but a `reasonable' decision would have been better

Glue Attributes Set by the Site

- Site information
  - SiteSysAdminContact: mailto: grid-admin@example.org
  - SiteSecurityContact: mailto: security@example.org
- Cluster info

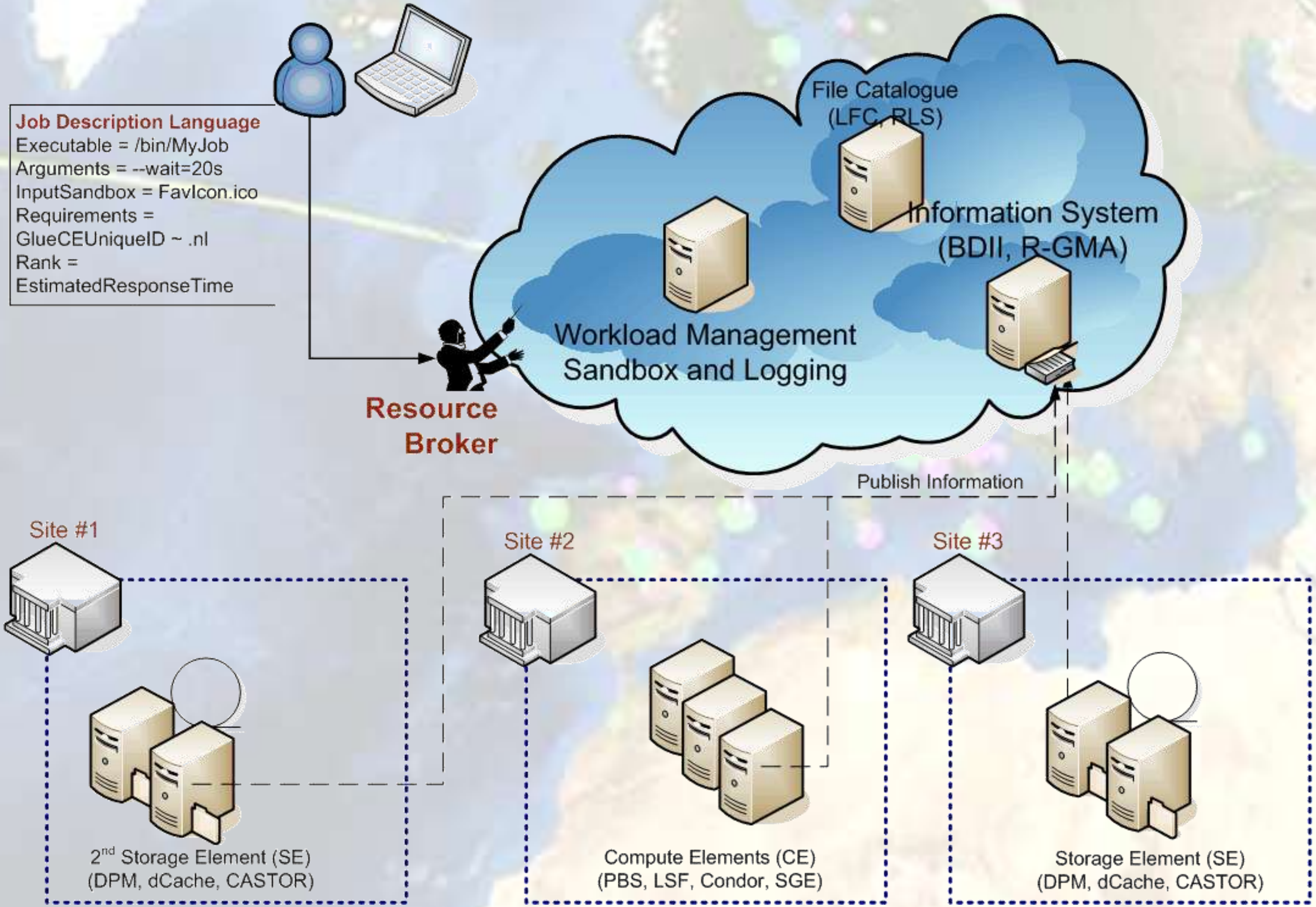
  GlueSubClusterUniqueID=gridgate.cs.tcd.ie

  HostApplicationSoftwareRunTimeEnvironment: LCG-2_6_0
  HostApplicationSoftwareRunTimeEnvironment: VO-atlas-release-10.0.4
  HostBenchmarkSI00: 1300
  GlueHostNetworkAdapterInboundIP: FALSE
  GlueHostNetworkAdapterOutboundIP: TRUE
  GlueHostOperatingSystemName: RHEL
  GlueHostOperatingSystemRelease: 3.5
  GlueHostOperatingSystemVersion: 3

  GlueCEStateEstimatedResponseTime: 519
  GlueCEStateRunningJobs: 175
  GlueCEStateTotalJobs: 248

- Storage: similar info (paths, max number of files, quota, retention, …)

Typical grid topology for computational jobs



Example: Compute Clusters

The Impact of Scale

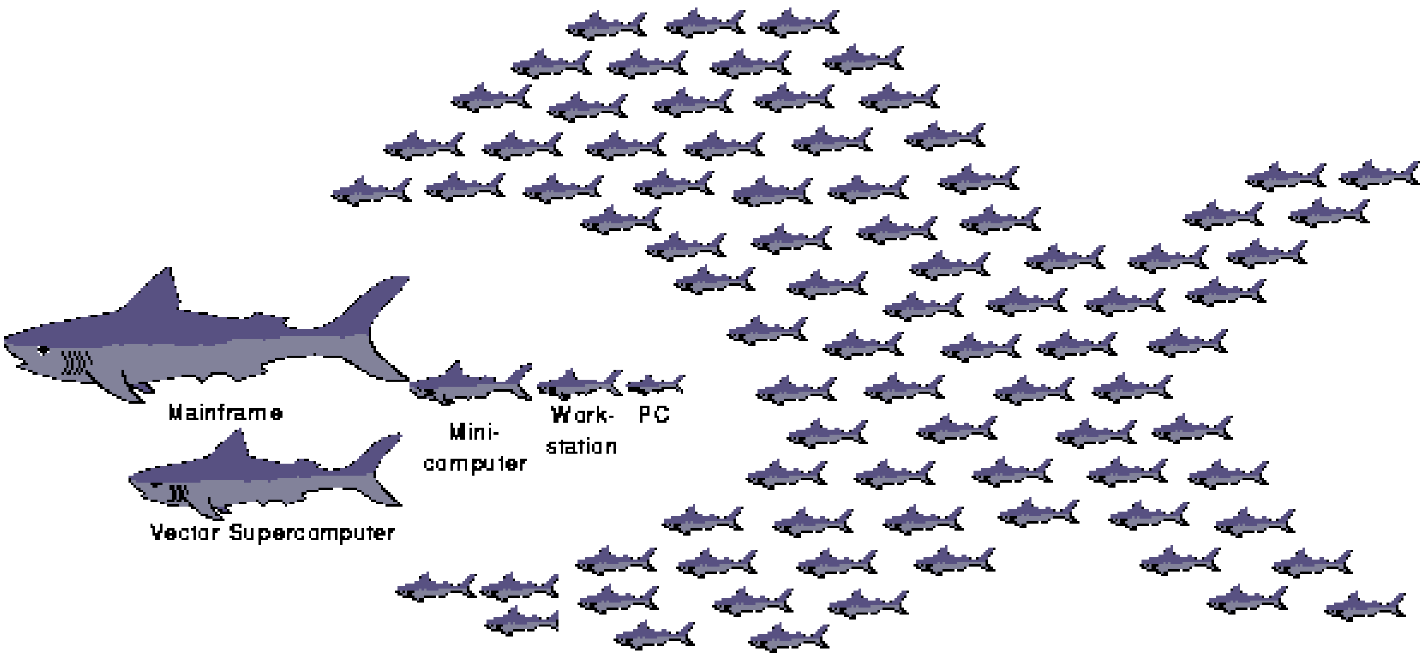
GRID SITE INFRASTRUCTURE

High Performance or High Throughput?

Key question: max. granularity of decomposition:

- Have you got one big problem or a bunch of little ones?
  - To what extent can the "problem" be decomposed into sort-of-independent parts ('grains') that can all be processed in parallel?
- Granularity
  - fine-grained parallelism – the independent bits are small, need to exchange information, synchronize often
  - coarse-grained – the problem can be decomposed into large chunks that can be processed independently
- Practical limits on the degree of parallelism –
  - how many grains can be processed in parallel?
  - degree of parallelism v. grain size
  - grain size limited by the efficiency of the system at synchronising grains

But the fact is:



NOW

'the food chain has been reversed', and supercomputer vendors are struggling to make a living.

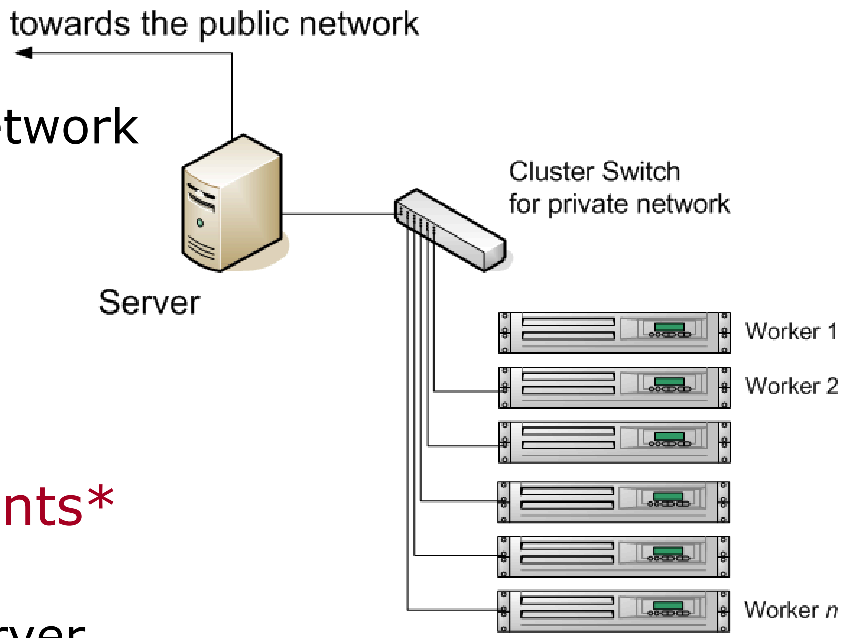
Graphic: Network of Workstations, Berkeley IEEE Micro, Feb, 1995, Thomas E. Anderson, David E. Culler, David A. Patterson

Using these systems

- As both clusters and capability systems are both ‘expensive’ (i.e. not on your desktop), they are resources that need to be *scheduled*
- And a wide multi-purpose grid will have **both** types of systems
- interface for scheduled access is a *batch queue*
  - job submit, cancel, status, suspend
  - sometimes: checkpoint-restart in OS, e.g. on SGI IRIX
  - allocate #processors
    (and amount of memory, these may be linked!)
    as part of the job request
- systems usually also have smaller interactive partition
  - more a ‘user interface’, not intended for running production jobs …

Cluster architectures

- ‘Beowulf’ virtual supercomputers
  - entire cluster managed by the server
  - users interact only with the server to start and manage jobs
  - parallelism supported by MPI, PVM, OpenMosix libraries
- classic network architecture
  - server connected to the public network
  - all WNs on a cluster-local LAN
  - usually using private IP space
  - no communication from the WNs to the outside world
- installs can even use diskless clients*
  - PXE boot refers to NFS root fs
  - all IO is done remotely on the server
  - But don’t try this for data-intensive computing!



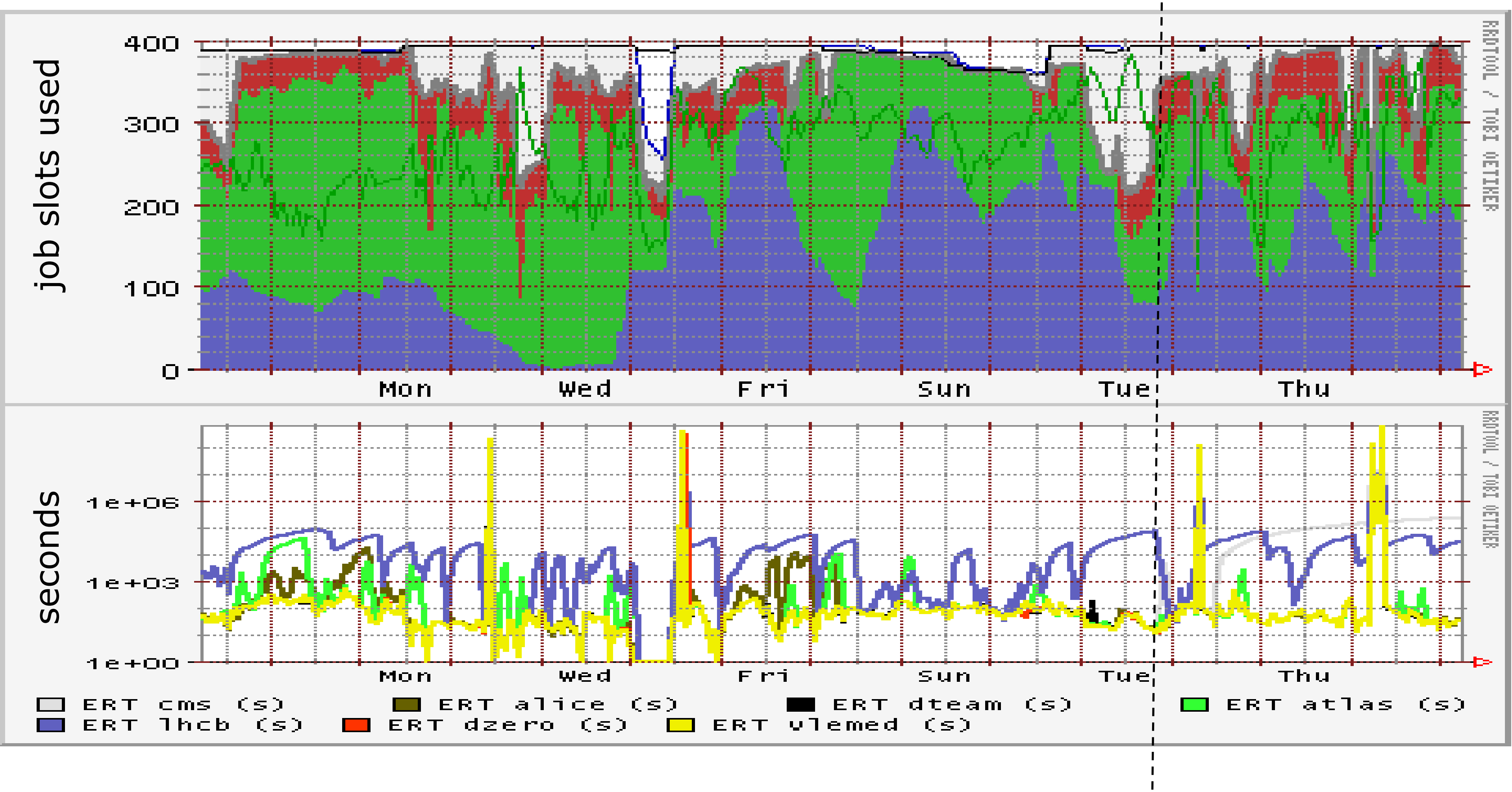
Example: scheduling policies - Maui

```
RMTYPE[0]                      PBS
RMHOST[0]                      tbn20.nikhef.nl
...
NODEACCESSPOLICY               SHARED
NODEAVAILABILITYPOLICY         DEDICATED:PROCS
NODELOADPOLICY                 ADJUSTPROCS

FEATUREPROCSPEEDHEADER         xps
BACKFILLPOLICY                 ON
BACKFILLTYPE                   FIRSTFIT
NODEALLOCATIONPOLICY           FASTEST

FSPOLICY              DEDICATEDPES
FSDEPTH               24
FSINTERVAL            24:00:00
FSDECAY               0.99

GROUPCFG[users]       FSTARGET=1     PRIORITY=10     MAXPROC=50
GROUPCFG[dteam]       FSTARGET=2     PRIORITY=5000   MAXPROC=32
GROUPCFG[alice]       FSTARGET=9     PRIORITY=100    MAXPROC=200   QDEF=lhcalice
GROUPCFG[alicesgm]    FSTARGET=1     PRIORITY=100    MAXPROC=200   QDEF=lhcalice
GROUPCFG[atlas]       FSTARGET=54    PRIORITY=100    MAXPROC=200   QDEF=lhcatlas

QOSCFG[lhccms]        FSTARGET=1-                    MAXPROC=10
```

MAUI is an open source product from ClusterResources, Inc. http://www.supercluster.org/

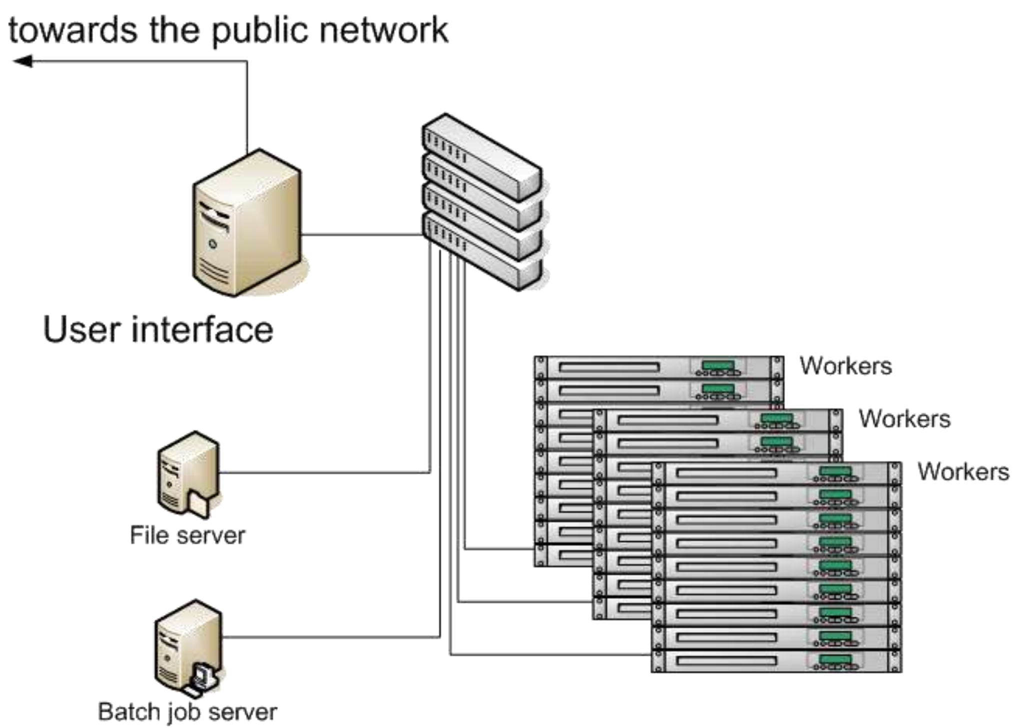
Fair shares and estimated response time

local 'fair shares', used to satisfy overall SLA requirements, need to be translated to an 'estimated response time' for the grid VOs and groups – an unsolved problem



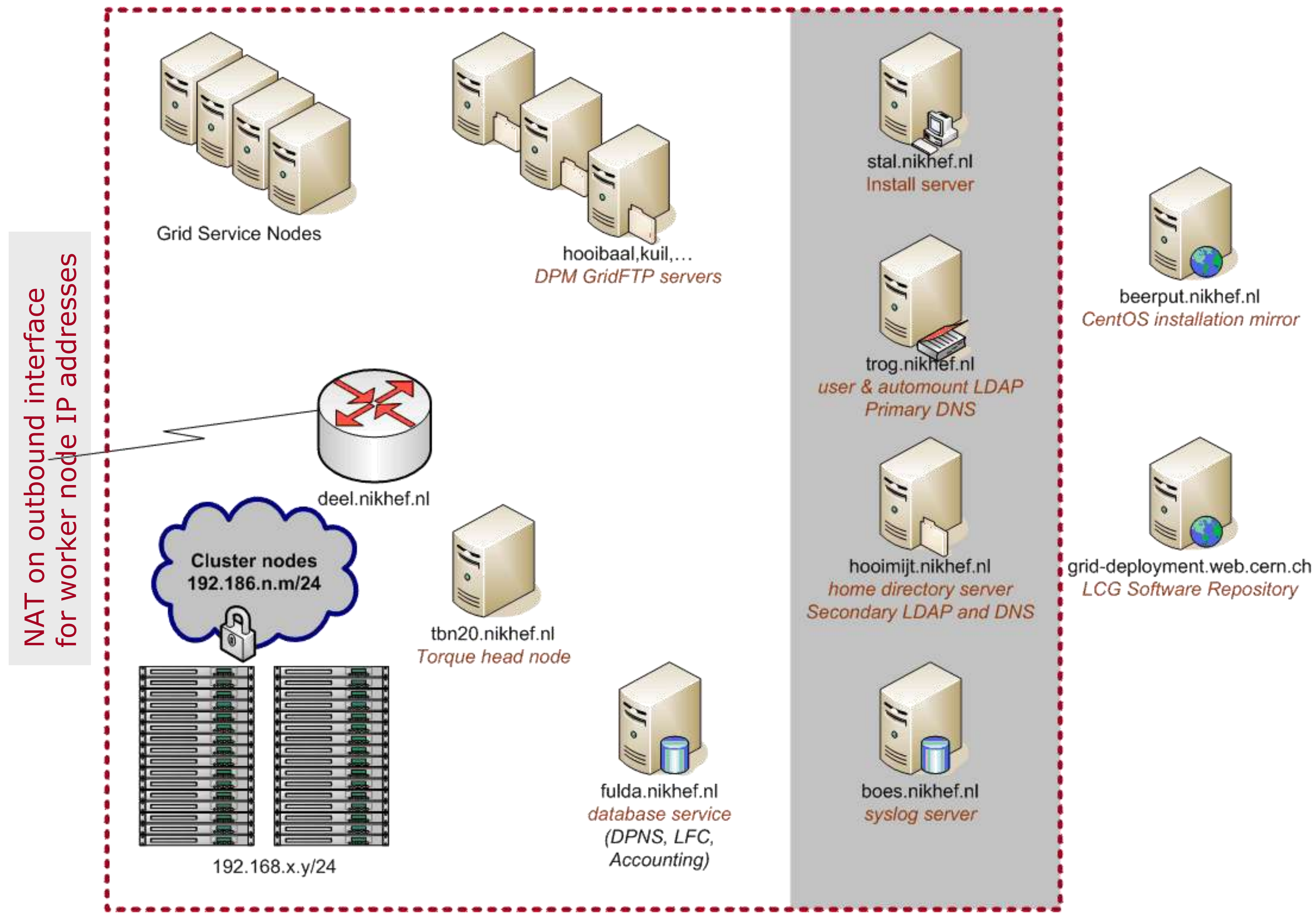
Growing your cluster

- Larger clusters accommodated by more switches, but
  - file I/O (headnode load) becomes bottleneck
    - system booting (PXE, NFS roots)
    - home directories
    - cluster job management
  - function separation (boot server, IO server) within the cluster helps only little



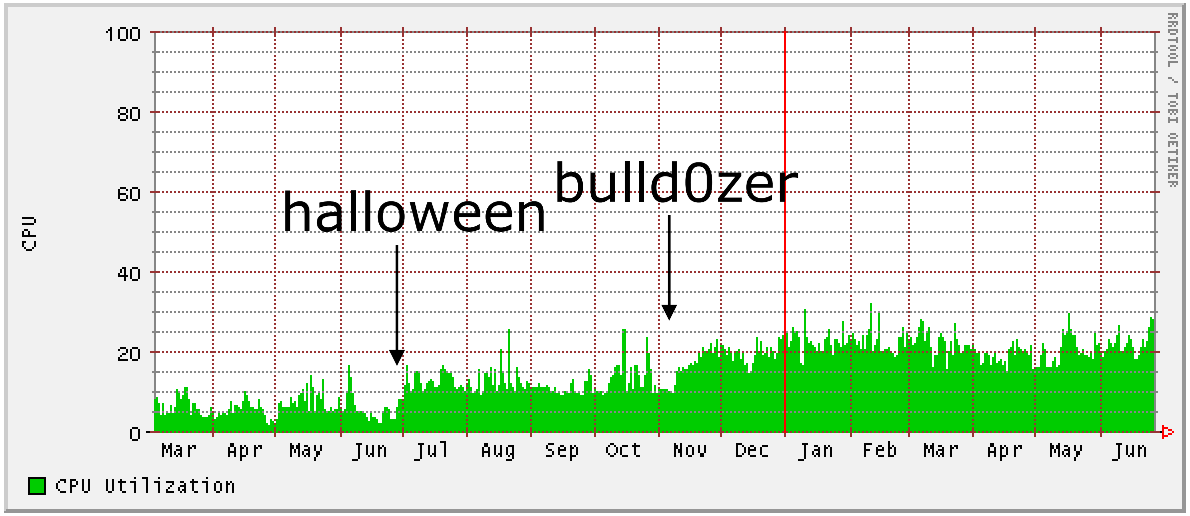
Extending your grid site

Storage, databases, information services



But NAT does not help

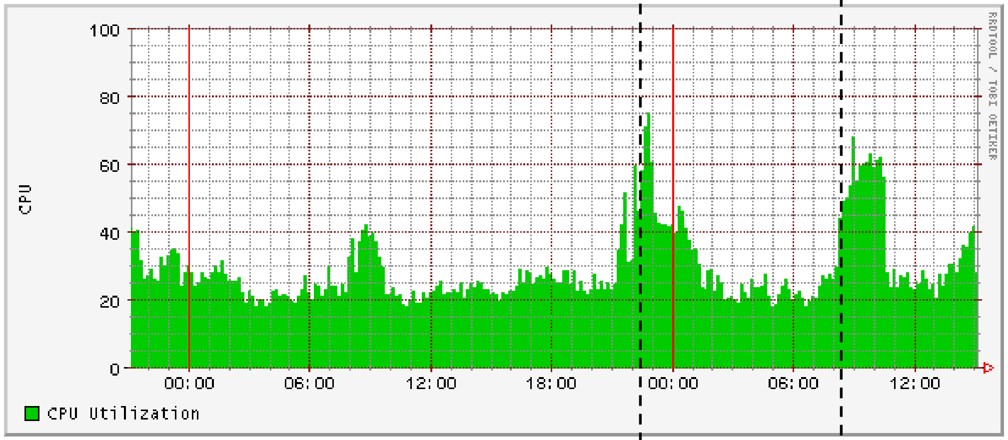
- The NAT kludge leads to several problems
  - with FTP-like protocols for data-transfer
  - with the load on the NAT box
- *and is certainly not the solution for protecting the WNs from attacks from the public internet, as commonly perceived*
  - *can do that easily with 'permit tcp established' followed by 'deny any any'*

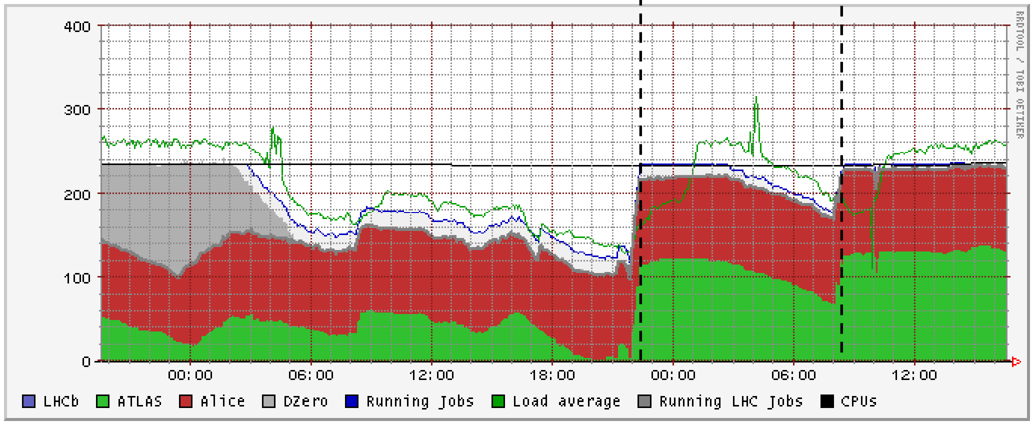


CPU load avearage 'deel.nikhef.nl'
(Foundry BigIron 15k with 2x BMGR8 Mngt-IV module)

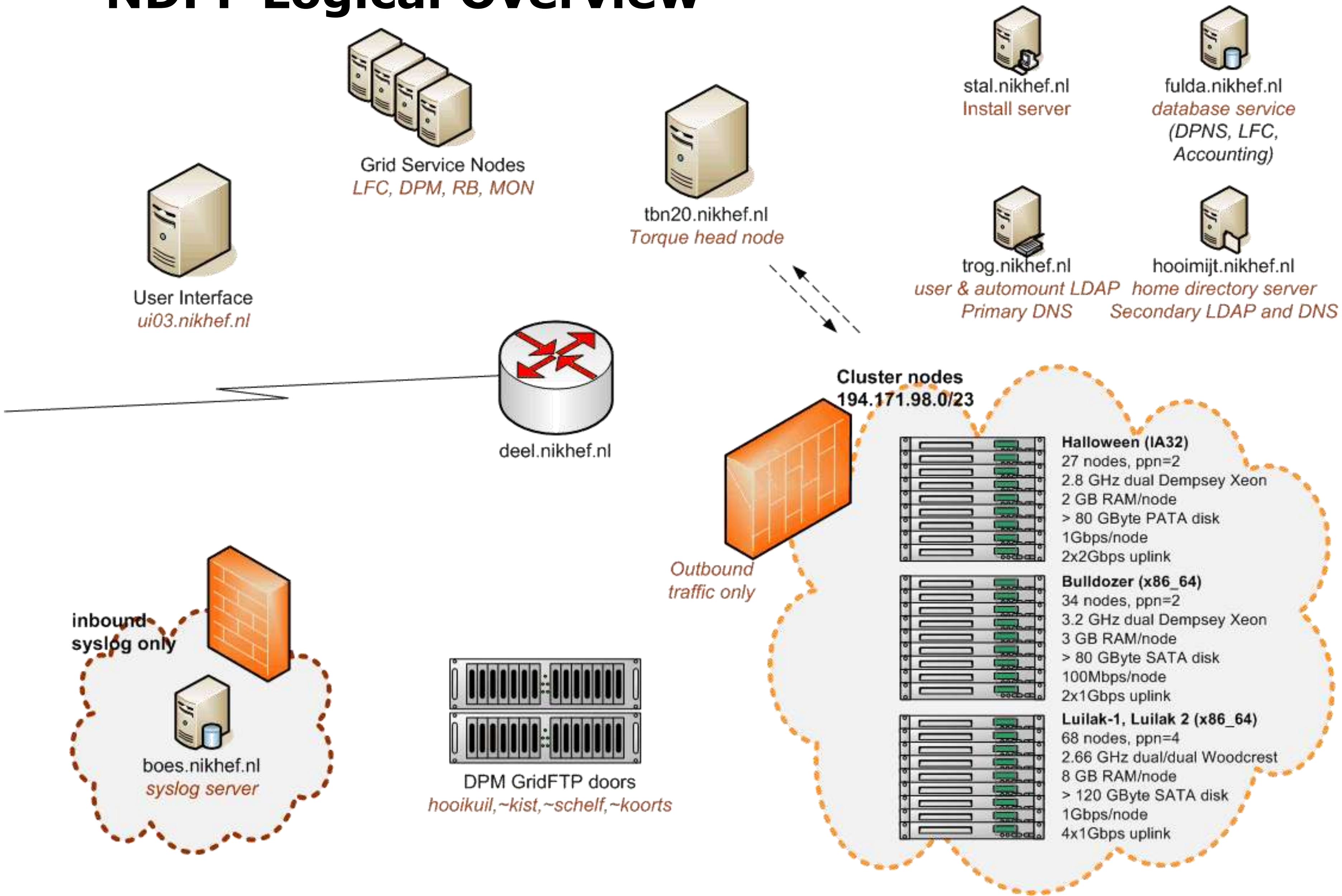
Data intensive jobs

ATLAS HEP jobs retrieving input data sets





NDPF Logical Overview

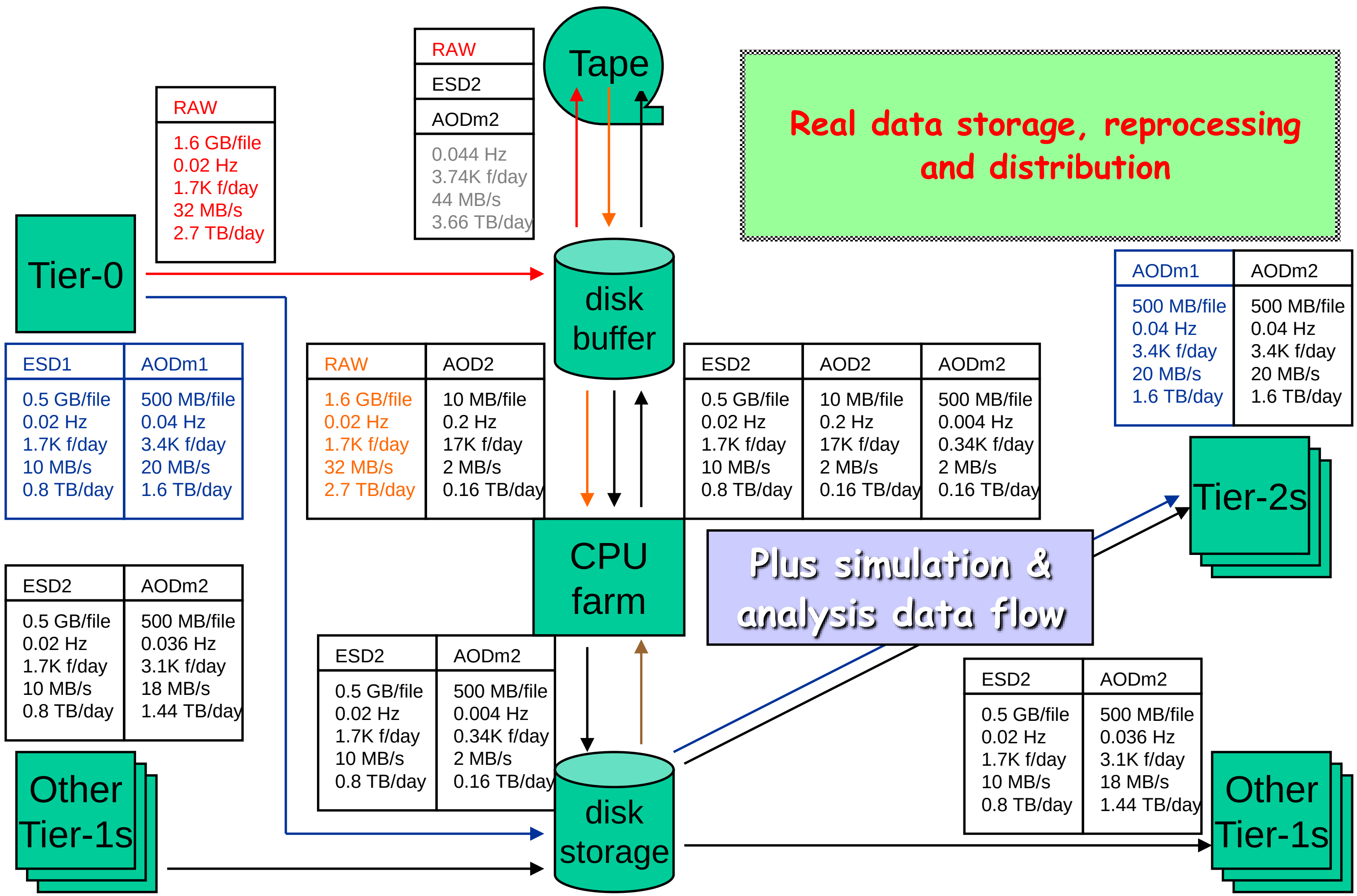


The LHC OPN

OPN Routing Creativity

NETWORK

Remembering the Atlas Tier-1 data flows



ATLAS data flows (draft). Source: Kors Bos, NIKHEF

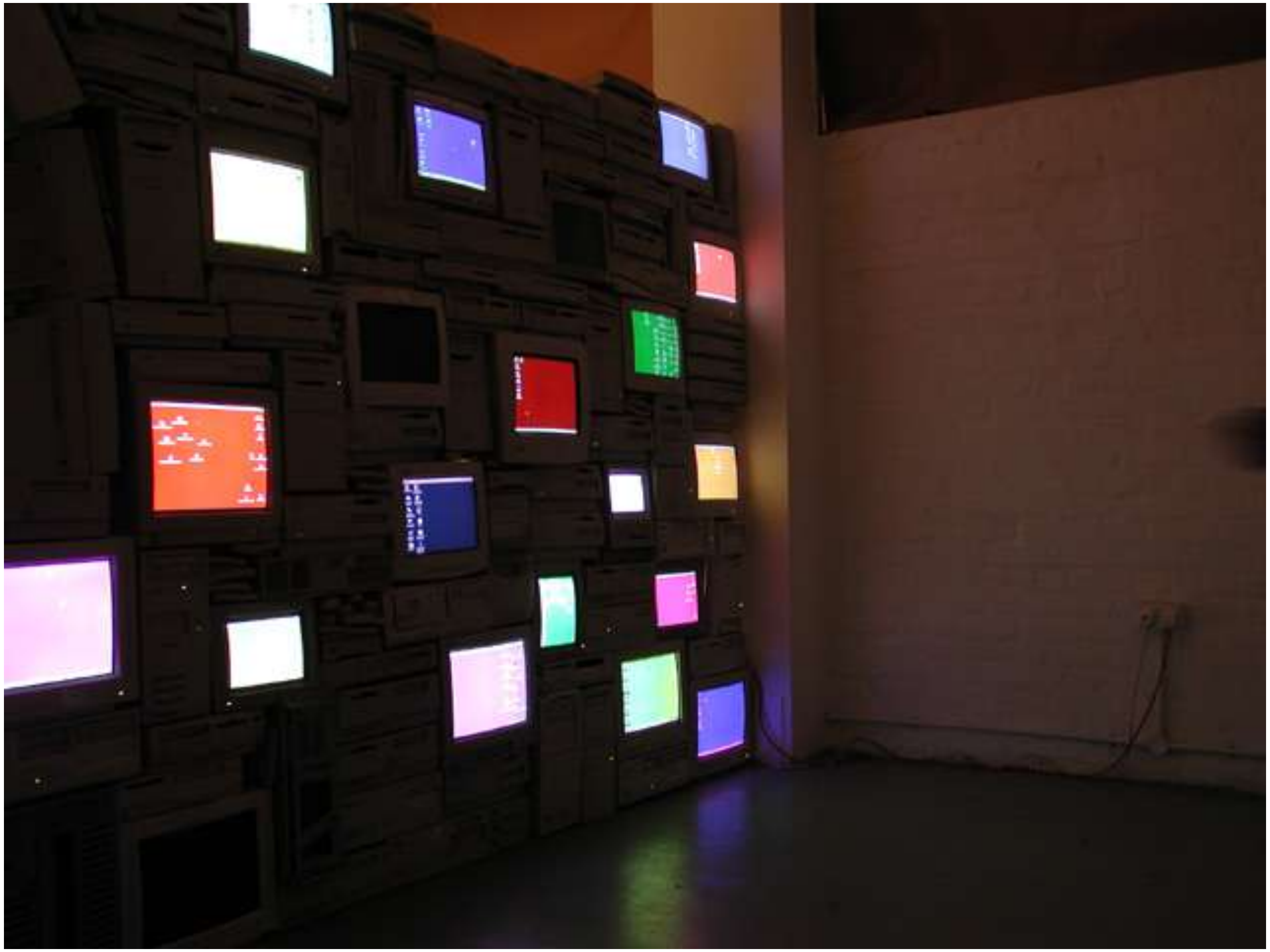
Streams and Firewalls

- Data transfer target:
  300 MByte/s out of CERN to **each** of the ~10 T1s
  - 24 GBit/s aggregate bandwidth
  - you cannot traverse firewalls at that speed

- OPN is really there to allow un-firewalled connections
  - internal routing only (BGP)
  - all participants sign a common policy
  - exclusively for data transfers
  - no direct connections to 'The Internet'

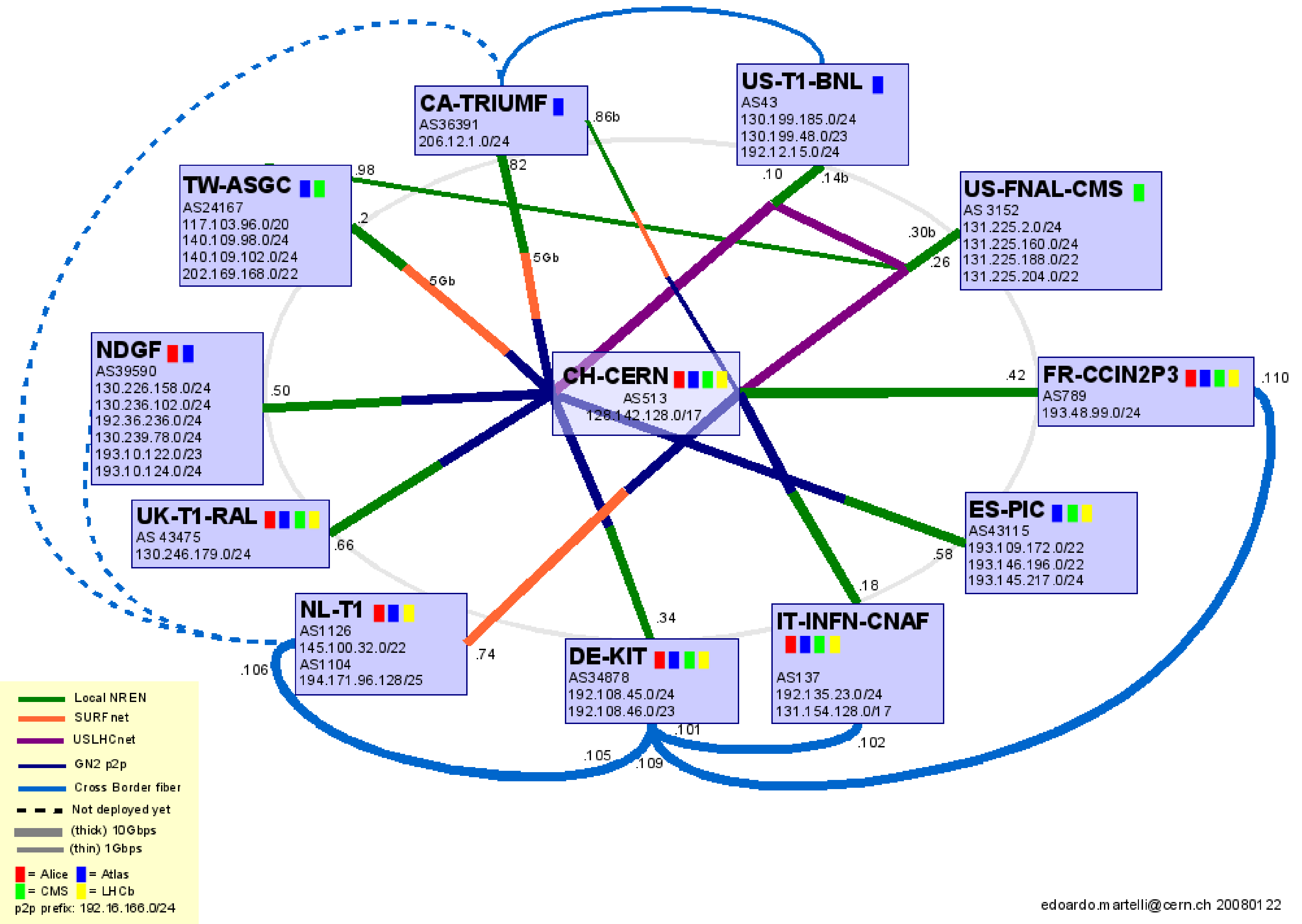
"*Firewall*" by Sandy Smith, www.computersforart.org

Firewall (2)



"*Firewall*" by Sandy Smith,
www.computersforart.org

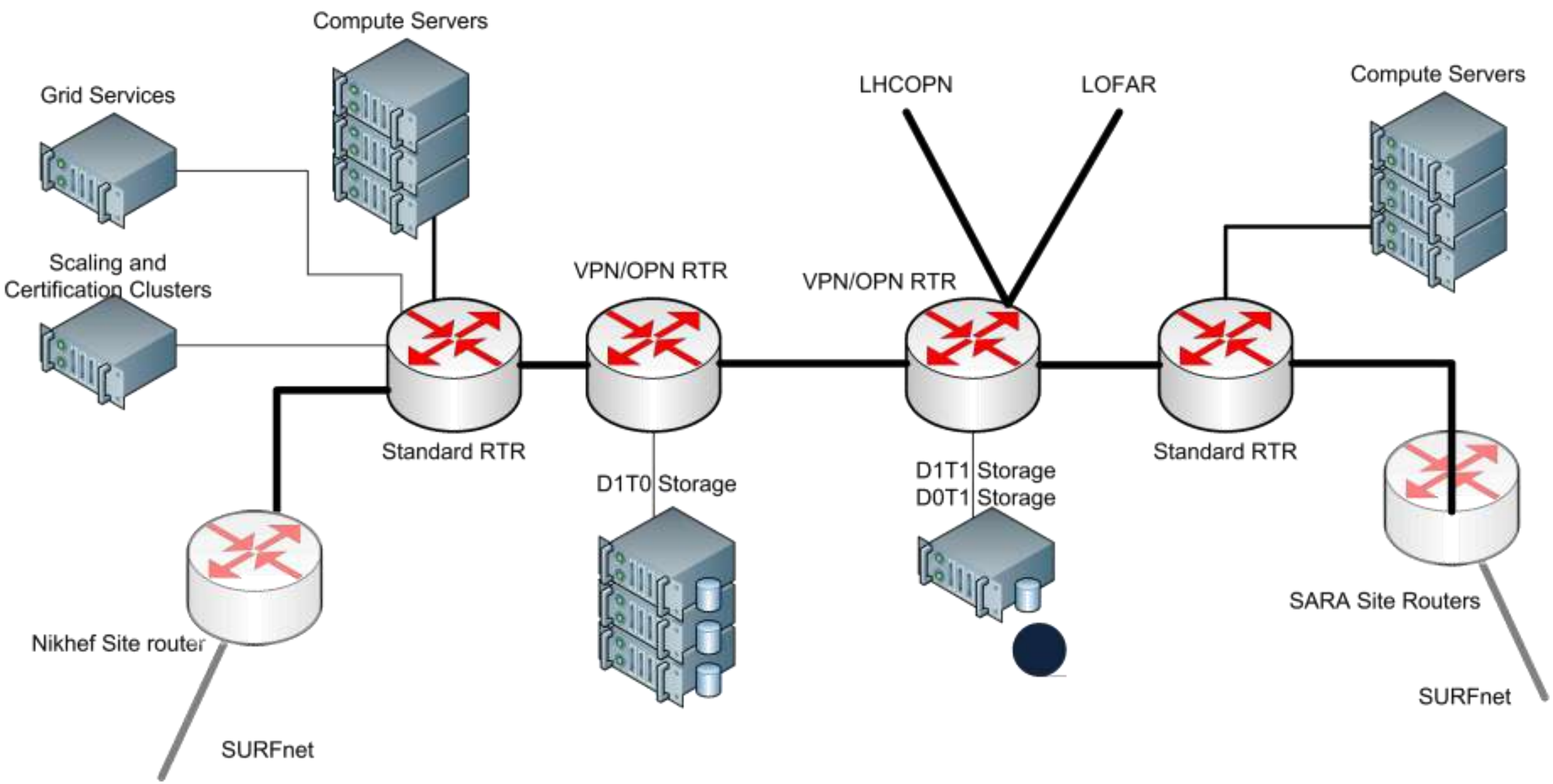
LHCOPN – current status



edoardo.martelli@cern.ch 20080122

Policy impact of the OPN

- Since only storage systems (not WNs) may use the OPN, router needs to distinguish between the two classes
  - If you have a single core router in your grid cluster where you want to terminate the OPN, you are almost forced to use source-based routing
  - but then you loose the features of BGP for fail-over &c
  - since a single router has a single routing policy, you need a *second* router to get the policy right …
  - With two independent OPNs, you need 3 routers
  - With three independent OPNs, you need 4 routers
  - ...
  - you actually need virtual routers in your box ☺



From: BiGGrid Network Plan 2008 BIGGRID-TC-2008-015 (draft)

Remember the Market

- How many OPNs are there, worldwide?
- Looking for OPN/VPN capable routers, you find their primary target market is corporate VPN use
  - Indeed, many independent end points
  - But each end-point uses ~ 54 kpbs – 2 Mbps
  - Then, try to push 10 Gbps through a CISCO VRF module ...
    ... which is kind enough to process each packet in software
  - Ouch
    Effective max speed ~ 900 Mbps per router
    (still pretty good, being targeted to the VPN dial-in market)

Managing many heterogeneous systems

OS level tricks

Procuring your systems: Help! I'm a publicly (co)funded body ...

SYSTEMS MANAGEMENT

Think BIG

Examples: CERN Computer Centre

- not only systems management
- but also asset management
- *and you are not even allowed to look inside Google's data centres!*

Installation

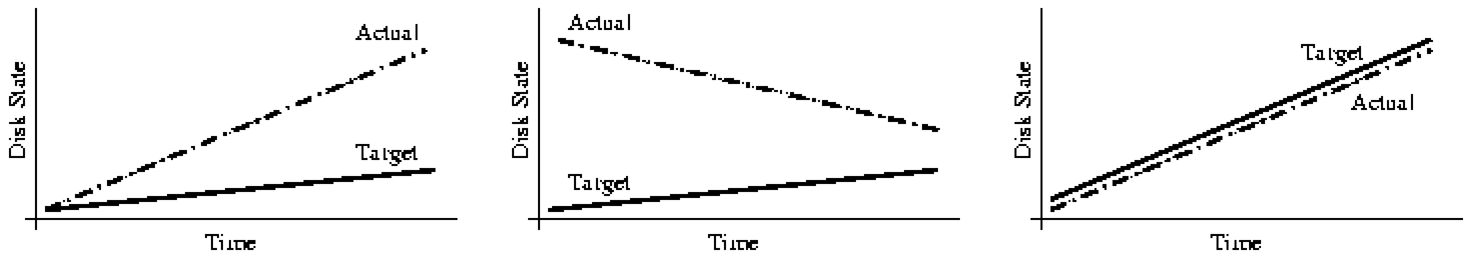
Installing and managing a large cluster requires a system that

- Scales to $\mathcal{O}$(10 000) nodes, with
  1. a wide variety in configuration ('service nodes')
  2. and also many instances of identical systems ('worker nodes')
- Is *predictable* and *consistent*
- Can rapidly recovery from node failures
  by commissioning a new box (i.e. in minutes)
- Preferably ties in with monitoring and recovery ('self-healing')

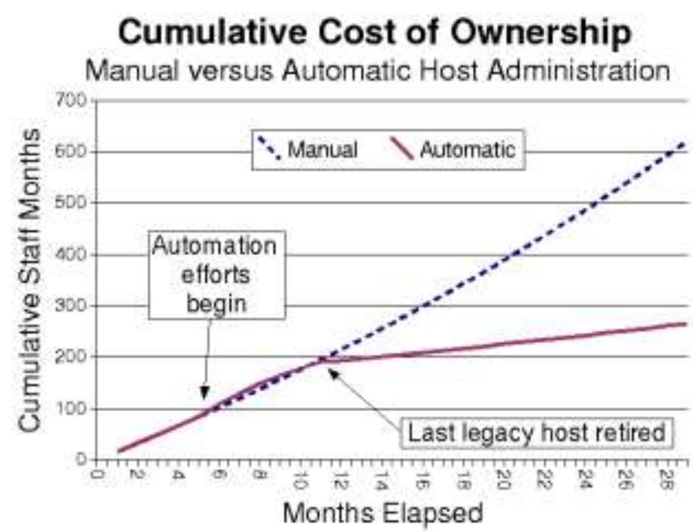
Popular systems include

- Quattor
- xCAT, NPACI Rocks
- SystemImager & cfEngine
- LCFGng

Divergent, Convergent, and Congruent Systems

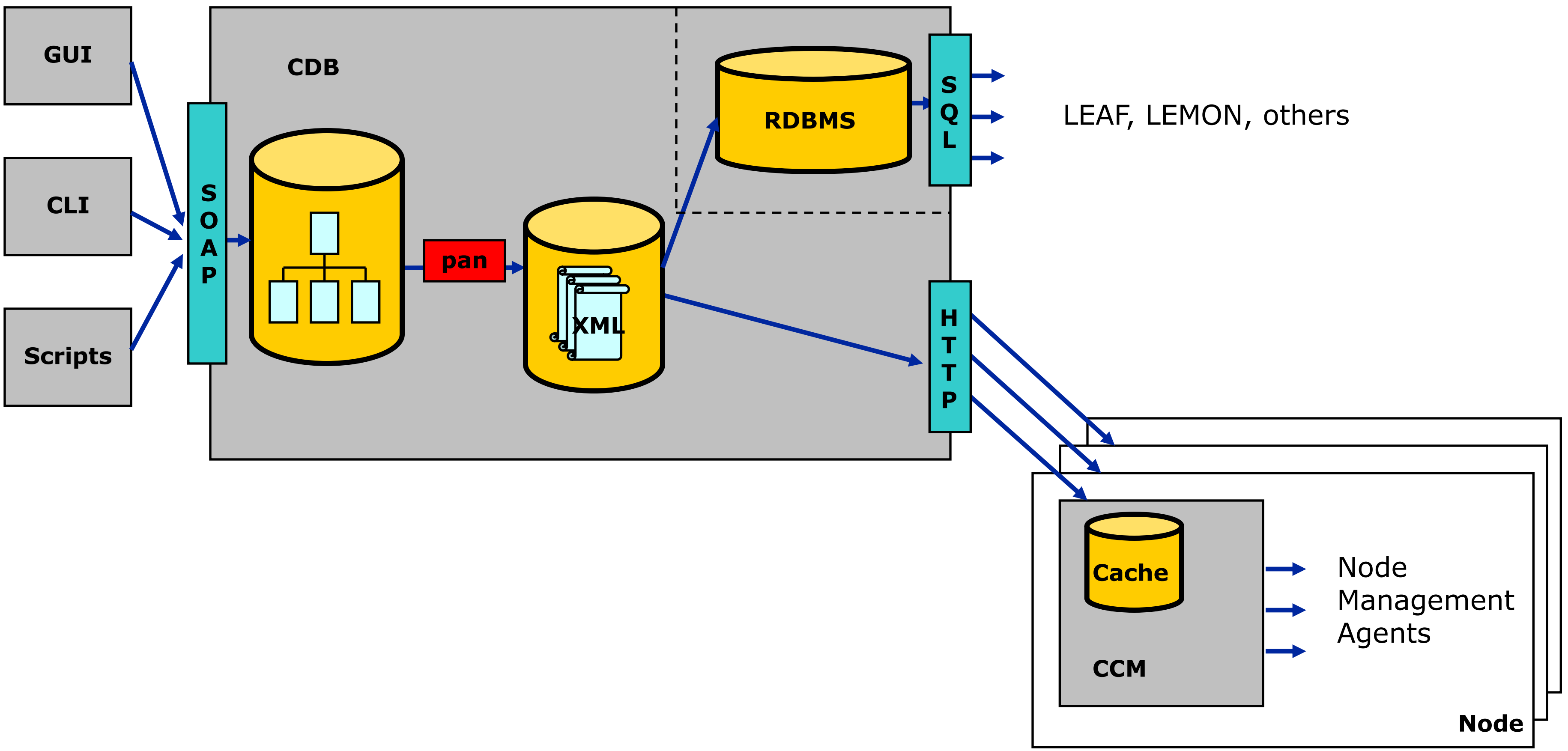


- Different characteristics
  - Incremental: cfengine, LCFGng
  - Deterministic by re-install: xCAT, Rocks
  - Transactional: Quattor
- Can a self-modifying system reach consistent (or even stable) state without repeatable deterministic ordering of changes on a host?



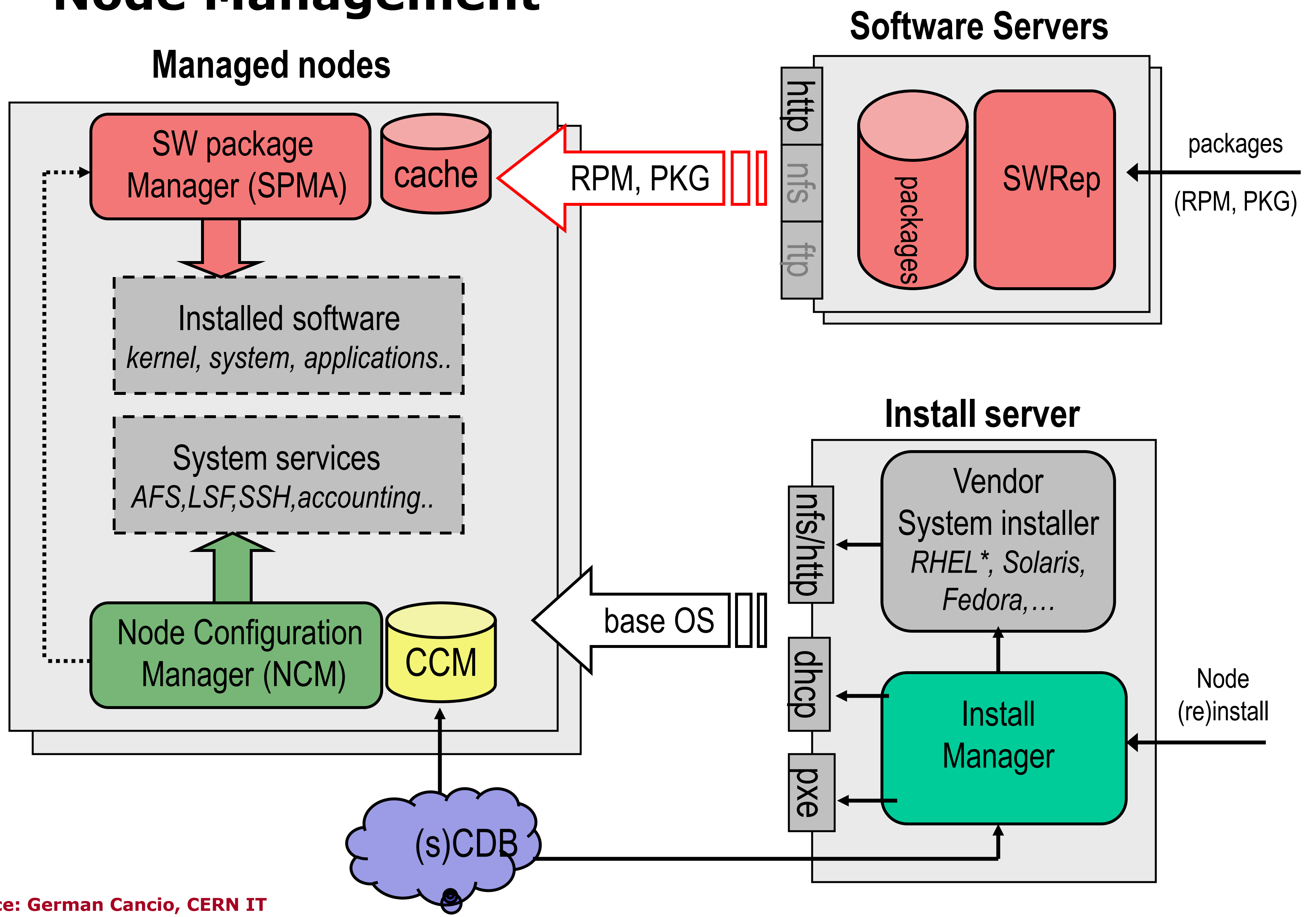
See also
http://www.infrastructures.org/papers/turing/turing.html
(figures are from paper referenced)

Complete Node Configuration Database



Source: German Cancio, CERN IT

Node Management



Source: German Cancio, CERN IT

Miscellaneous systems configuration hints

an arbitrary selection of deployed configurations

User and system directories and maps

Large number of alternatives exists (nsswitch.conf/pam.d)

- files-based (/etc/passwd, /etc/auto.home, ...)
- YP/NIS, NIS+
- Database (MySQL/Oracle)
- LDAP

We went with LDAP:

- information is in a central location (like NIS)
- scales by adding slave servers (like NIS)
- is secure by LDAP over TLS (unlike NIS)
- can be managed by external programs (also unlike NIS)
  (we can even do real-time grid credential mapping to and from uid's)

But you will need to run *nscd*, or a large number of slave servers

- with *nscd*, a single server can easily handle ~200 nodes/500 cores
- in rare cases, (statically linked) programs run into trouble

Logging and Auditing

Auditing and logging

- syslog (also for grid gatekeeper, gsiftp, credential mapping)
- process accounting (psacct)

For the paranoid – use tools included for CAPP/EAL3+: LAuS

- system call auditing
- highly detailed:
  useful both for debugging and incident response
- default auditing is critical: system will halt on audit errors ☺
  - and once in a while you hit a kernel bug that cannot be reproduced, as we did in RHEL3 ☹

If your worker nodes are on private IP space

- need to preserve a log of the NAT box as well

Grid and Cluster Logging

Grid statistics and accounting

- *rrdtool* views from the batch system load per VO
  - combine *qstat* and *pbsnodes* output via script, cron and RRD
- *cricket* network traffic grapher
- *ganglia* monitoring
- *Nagios* probe-based alarms and (*grid)* monitoring
- extract *pbs accounting data* in dedicated database
  - grid users have a 'generic' uid from a dynamic pool – need to link this in the database to the grid DN and VO
- from accounting db, upload (*anonymized)* records
  - grid accounting system for VOs and funding agencies
  - accounting db also useful to charge costs to projects locally
  - but remember to consider DPA restrictions
    - *define data usage explicitly*
    - *make users agree, and make sure your click-through actually holds up*
    - *don't expose if you don't need to*

Nagios display

Nagios

General
- Home
- Documentation

Monitoring
- Tactical Overview
- Service Detail
- Host Detail
- Hostgroup Overview
- Hostgroup Summary
- Hostgroup Grid
- Servicegroup Overview
- Servicegroup Summary
- Servicegroup Grid
- Status Map
- 3-D Status Map

- Service Problems
- Host Problems
- Network Outages

Show Host:

- Comments
- Downtime

- Process Info
- Performance Info
- Scheduling Queue

Reporting
- Trends
- Availability
- Alert Histogram
- Alert History
- Alert Summary
- Notifications
- Event Log

Configuration
- View Config

Current Network Status
Last Updated: Wed Jun 6 11:53:57 CEST 2007
Updated every 90 seconds
Nagios® - www.nagios.org
Logged in as *nagiosadmin*
- Notifications are disabled

View History For all hosts
View Notifications For All Hosts
View Host Status Detail For All Hosts

Host Status Totals

Up	Down	Unreachable	Pending
25	0	0	1

All Problems	All Types
0	26

Service Status Totals

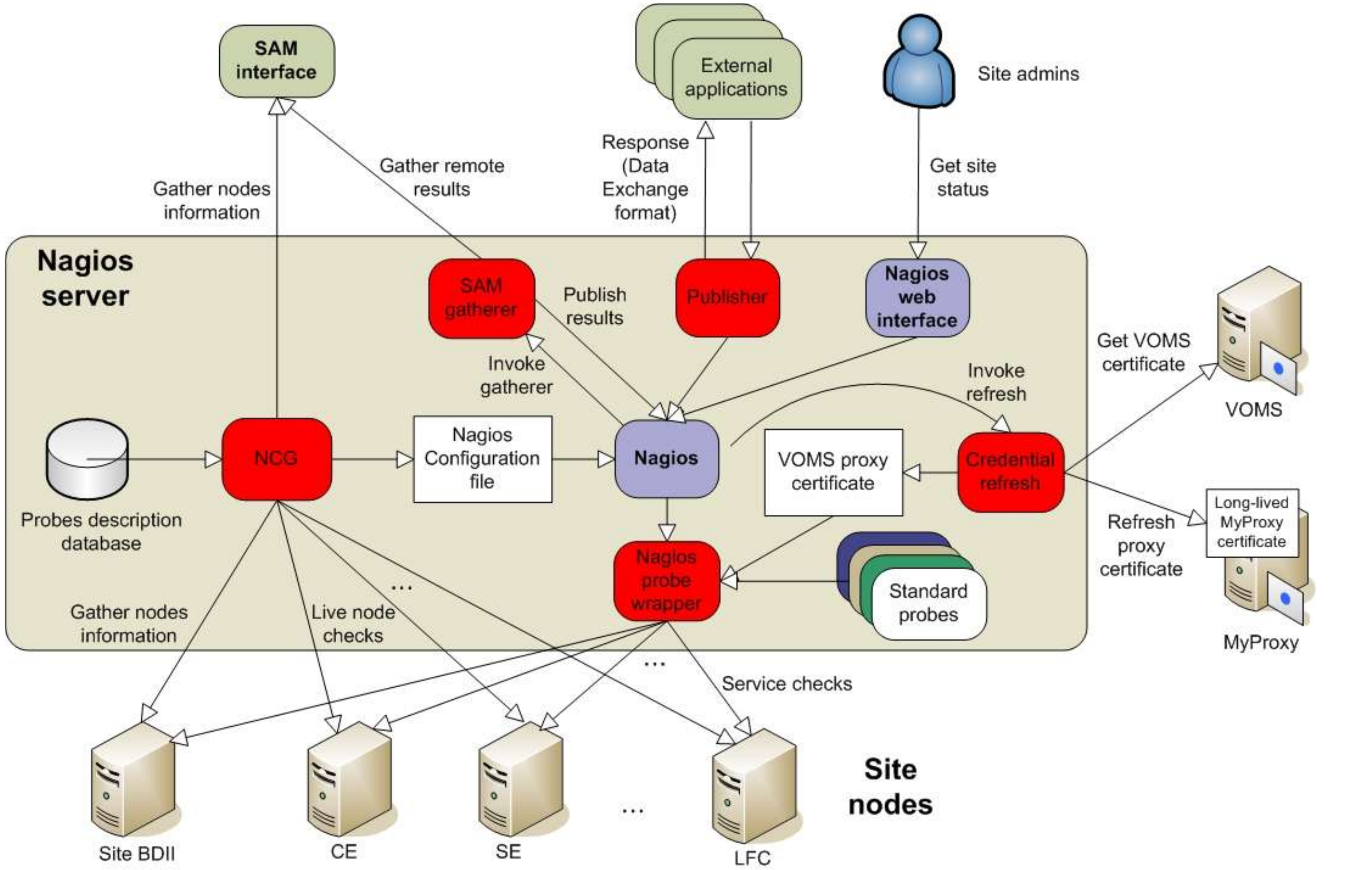
Ok	Warning	Unknown	Critical	Pending
158	0	137	24	0

All Problems	All Types
161	319

Service Status Details For All Hosts

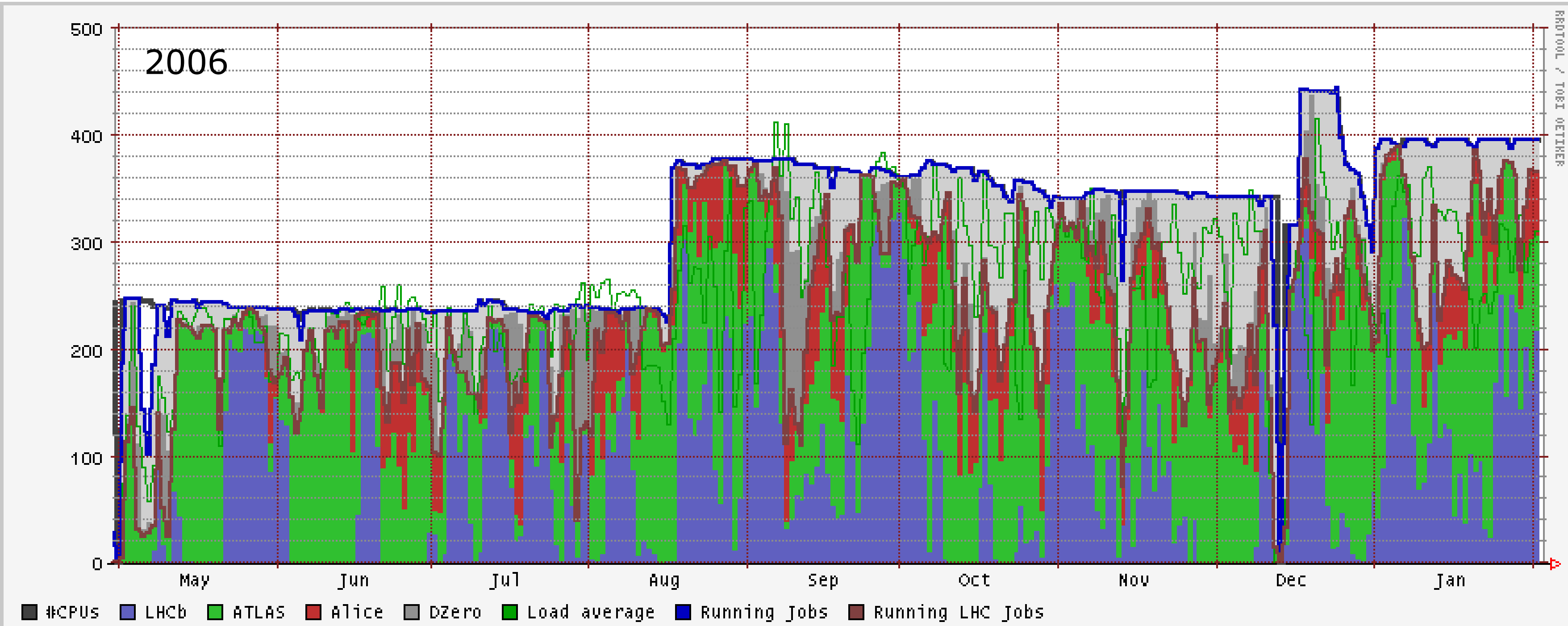
Host	Service	Status	Last Check	Duration	Attempt	Status Information
castorgrid.cern.ch	GridFTP-Ping	OK	06-06-2007 11:53:33	11d 20h 16m 10s	1/4	FTP OK - 0.039 second response time on port 2811 [220 castorgrid04.cern.ch CASTOR GridFTP Server 1.12 GSSA Globus/GSI wu-2.6.2(cern-2) (gcc32dbg, 1069715860-42
	GridFTP-Transfer	OK	06-06-2007 11:16:03	0d 0h 37m 54s	1/4	Upload to remote computer succeeded. Download from re computer succeeded. File successfully removed from rem computer. Received file is valid.
	SE-host-cert-valid-OPS-remote	OK	06-06-2007 11:38:08	0d 18h 29m 35s	1/1	SAM status: ok
	SE-lcg-cp-Atlas-remote	OK	06-06-2007 11:03:53	0d 18h 49m 54s	1/1	SAM status: ok
	SE-lcg-cp-CMS-remote	OK	06-06-2007 09:59:00	0d 1h 54m 57s	1/1	SAM status: ok
	SE-lcg-cp-DTeam-remote	OK	06-06-2007 11:47:54	0d 18h 21m 11s	1/1	SAM status: ok
	SE-lcg-cp-OPS-remote	OK	06-06-2007 11:00:03	0d 19h 2m 36s	1/1	SAM status: ok
	SE-lcg-cr-Atlas-remote	OK	06-06-2007 11:03:50	0d 18h 49m 59s	1/1	SAM status: ok
	SE-lcg-cr-CMS-remote	OK	06-06-2007 09:58:48	0d 1h 55m 9s	1/1	SAM status: ok
	SE-lcg-cr-DTeam-remote	OK	06-06-2007 11:47:51	0d 18h 21m 14s	1/1	SAM status: ok
	SE-lcg-cr-OPS-remote	OK	06-06-2007 11:00:00	0d 19h 2m 39s	1/1	SAM status: ok
	SE-lcg-del-Atlas-remote	OK	06-06-2007 11:03:56	0d 18h 49m 51s	1/1	SAM status: ok
	SE-lcg-del-CMS-remote	OK	06-06-2007 09:59:05	0d 1h 54m 52s	1/1	SAM status: ok
	SE-lcg-del-DTeam-remote	OK	06-06-2007 11:47:56	0d 18h 21m 8s	1/1	SAM status: ok
	SE-lcg-del-OPS-remote	OK	06-06-2007 11:00:05	0d 19h 2m 34s	1/1	SAM status: ok
	SE-seavail-OPS-remote	OK	06-06-2007 11:38:13	0d 18h 18m 53s	1/1	SAM status: ok
	SE-seused-OPS-remote	OK	06-06-2007 11:38:13	0d 18h 18m 53s	1/1	SAM status: ok

Nagios implementation at a site

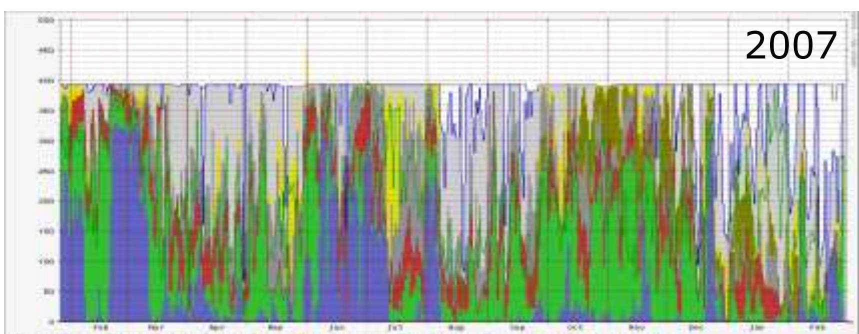


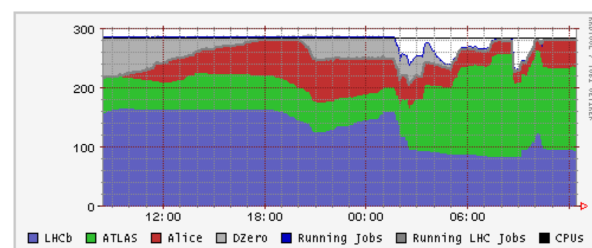
Graph: Ian Bird

NDPF Occupancy

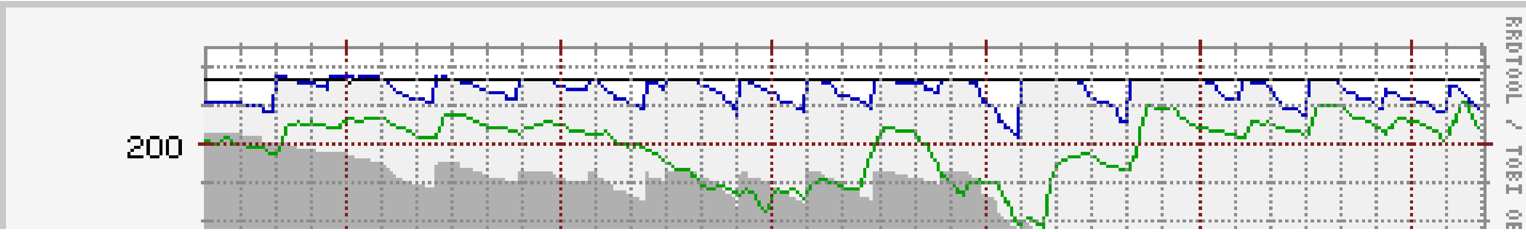


Average occupancy 2006, 2007 > 90%

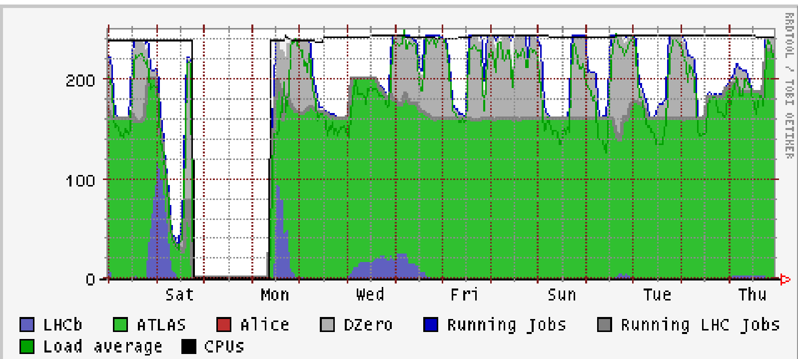




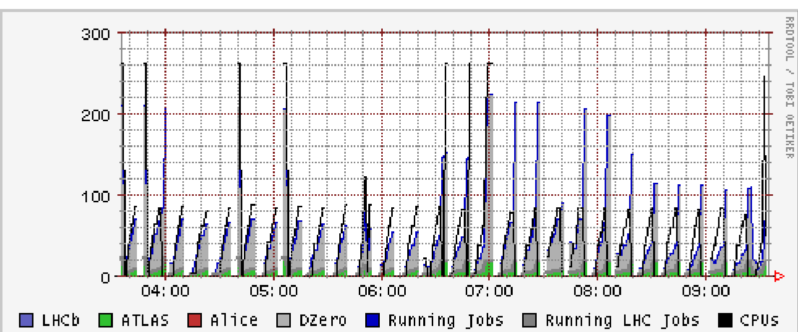
each colour represents a grid VO, black line is #CPUs available



An unresponsive node causes the scheduler MAUI to wait for 15 minutes, then give up and start scheduling again, hitting the rotten node, and …

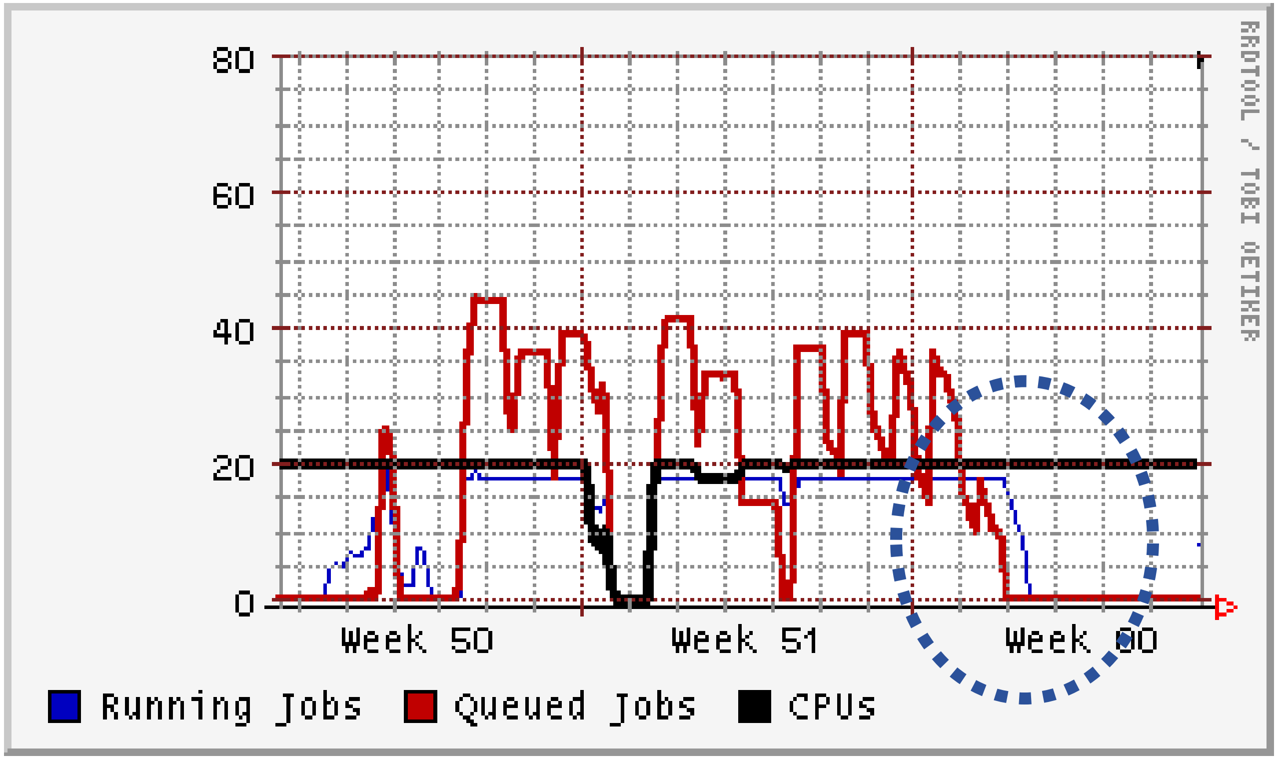


Auditing Indicent: a disk with less than 15% free makes the syscall-audit system panic, new processes cannot write audit entries, which is fatal, so they wait, and wait, and …
a head node has most activity & fails first!



PBS Server trying desparately to contact a dead node who’s CPU has turned into Norit … and unable to serve any more requests.

Black Holes



A mis-configured worker node accepting jobs that all die within seconds.
Not for long, the entire job population will be sucked into this black hole…

Clusters: what did we see?

- the Grid (and your cluster) are error amplifiers
  - "black holes" may eat your jobs piecemeal
  - dangerous "default" values can spoil the day ("GlueERT: 0")
- Monitor! (and allow for (some) failures, and design for rapid recovery)

- Users don't have a clue about your system beforehand
  (that's the downside of those 'autonomous organizations')
- If you want users to have clue, you push publish your clues correctly
  (the information system is all they can see)
- Grid middleware may effectively do a DoS on your system
  - doing *qstat* for every job every minute, to feed the logging & bookkeeping …

- Power consumption is the greatest single limitation in CPU density

- And finally: investment in machine room tidiness pays off,
  … or your colleague will not find that #$%^$*! machine
  in the middle of night…

Gluing things together

- Having flexible cluster management tools is one thing
  - *keeping it manageable with a staff shortage is something different*
- Hardware standardization helps
  - ensure strict HW compatibility with the chosen OS, and both it's next **and previous** version
  - no custom drivers
  - no custom kernels &c
  - standard management interfaces (IPMI)
- this needs careful wording in a RfP
  - or vendors will sell you quite wonderful but very labour-intensive stuff!
  - which either works only with Windows, or with their custom kernel

Breaking the egg shell approach
Towards policy harmonization
Open Issues: personal data in the grid

SECURITY IN A DISTRIBUTED WORLD

Hardening your cluster

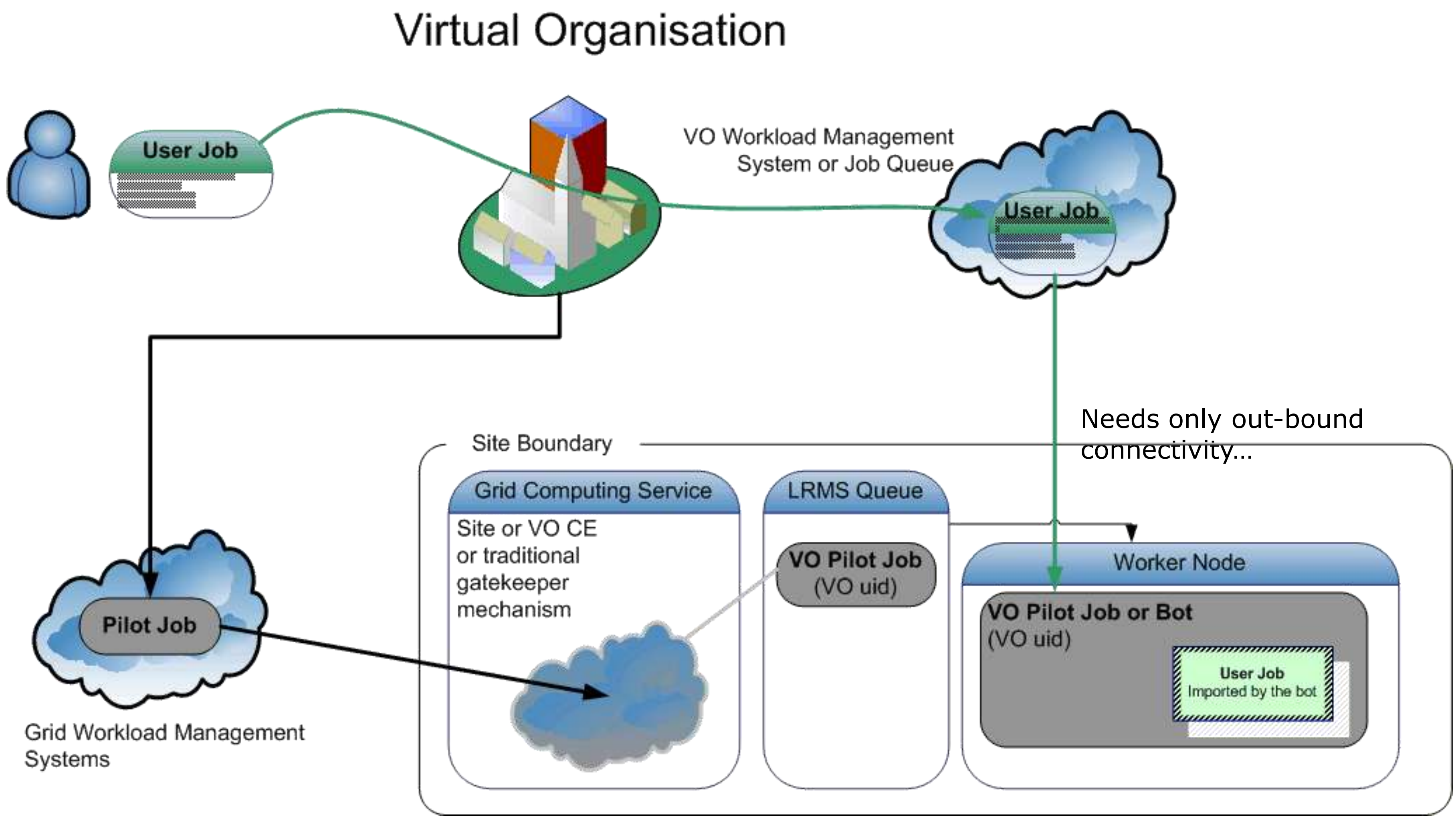
A firewall feels quite secure … initially …

- with grid resource sharing, the 'eggshell' approach breaks
  - local users are no longer local:
    - local exploits will be used
    - malicious users will try to 'escape' from the worker nodes
    - anyway, O(10k) systems in one go is quite attractive ;-) and has real market value
  - if you support an 'user interface' system for some remote users to get onto the grid, they *will* the same password as everywhere else
  - the most common attack on distributed clusters today is still the ssh trojans and password sniffers

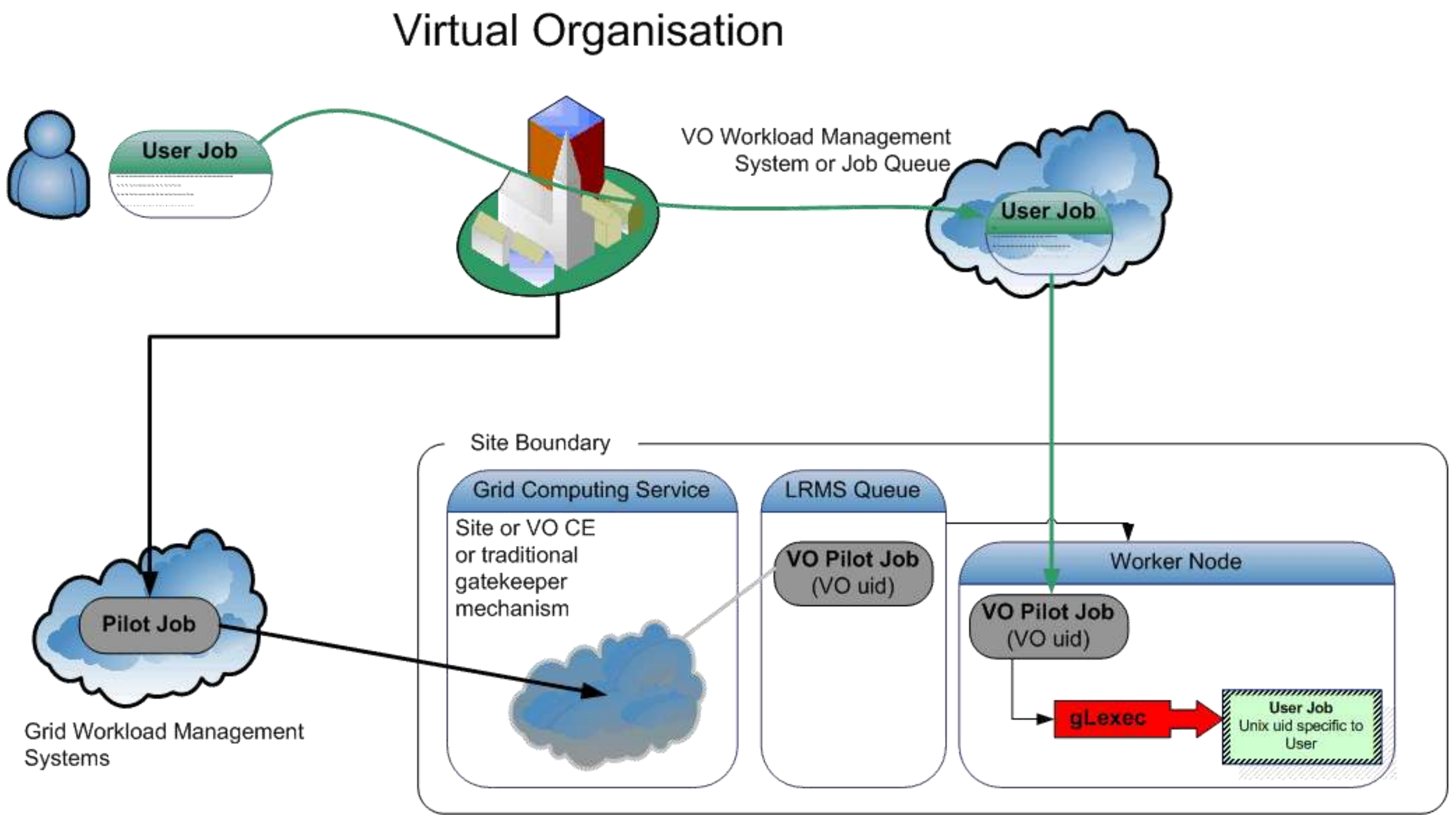
- you need global coordination, or you will be re-compromised from other 'partner' sites

read **http://www.nsc.liu.se/~nixon/stakkato.pdf** *for some real-life experience*

What A VO Community Can Do To You



Working with VO to respect policy, isolation

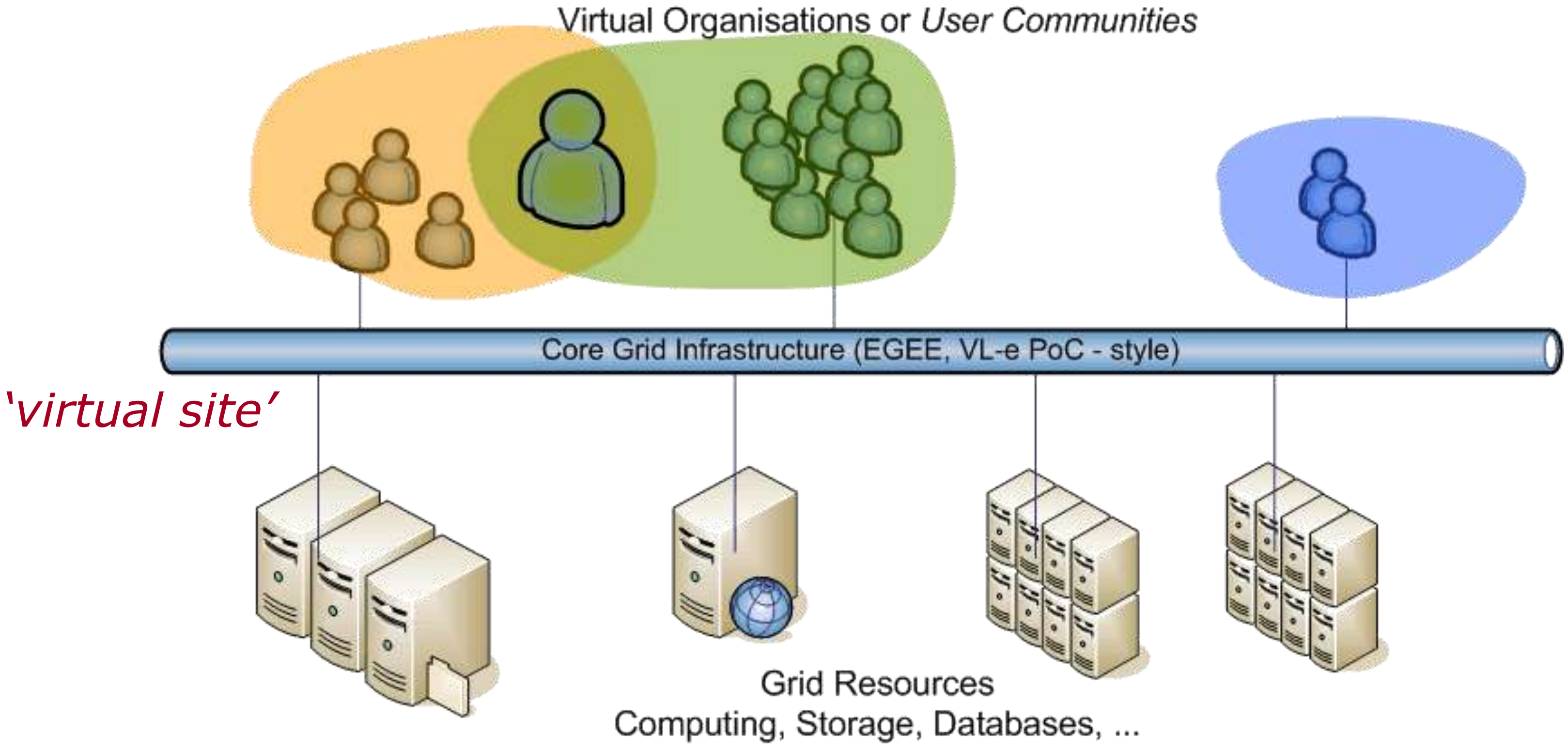


- At least prevent stealing VO pilot job credentials
- Allow cooperative policy compliance

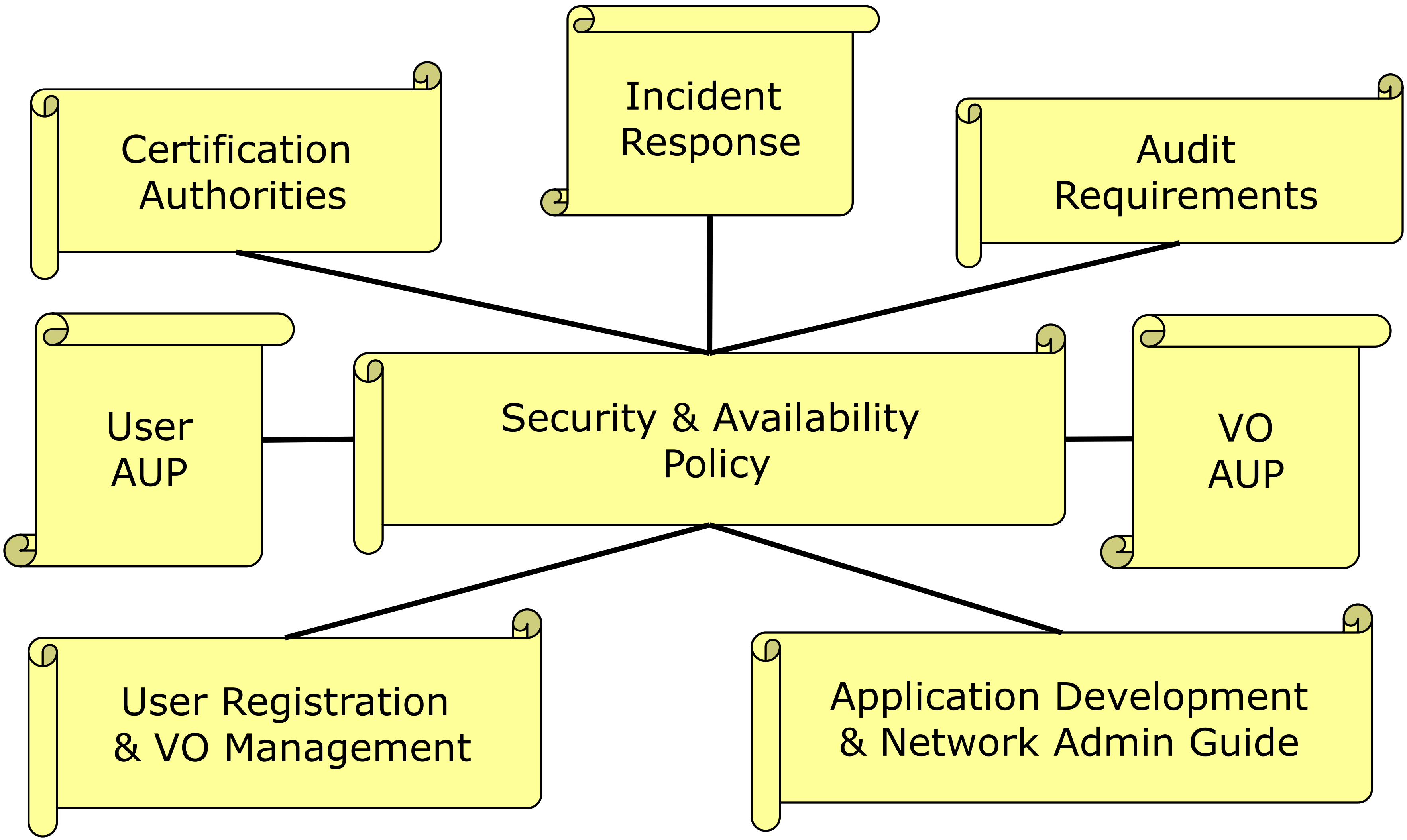
Operational model for the eInfrastructure

- Facilitating sustainable infrastructures (*e.g. science*)
  - decouple base resources from the VOs
  - provide “hosting” of core (central) services for smaller VOs

but now, policy is needed to create trust between users and ‘grid service providers’ (resource owners)



EGEE/LCG Security Policies



picture: Ian Neilson

http://cern.ch/proj-lcg-security/documents.html

strike balance between security and usability …

Issues

- Distributed security
  - any computer, desktop and laptop, must be assumed compromised
  - identity vetting and community membership assertions needed in cross-domain grids
  - trust between organisations needed
    - we demonstrated this in science – globally!
    - federated access to a wide range of resources coming
  - security, privacy policies must be coordinated
    - essential for a mainstream, sustained, infrastructure

- Hardware-supported security is gaining momentum

Balancing incident response to privacy

- There are a couple of exemption clauses, for
  - Computer, communications and access control
  - but limited to max 6 months

  … otherwise, you actually ought to register your administration or accounting database
- Write down what you keep, why, and for how long
- Keep as little data as possible
- Limit logs to traffic analysis, not content
- But
  - keep enough to trace people in case of incidents
  - and to support your peers in dealing with incidents

See e.g.

- http://www.cbpweb.nl/documenten/av_21_Goed_werken_in_netwerken.stm
- http://www.cbpweb.nl/HvB_website_1.0/i1.htm

What's in a Policy

- What do lawyers typically look for
  - Consistency of Terminology
  - Describe in exact and limitative terms
- How binding is it?
  - The signer must be explicitly aware of his or her action
  - Use default-deny
  - On web forms: at least use a pop-up box
  - *But this has only been marginally tested in court*
- What about the subjects
  - Keep it simple and short
  - 'Separate the policy from the actions' – but, indeed, then they'll never read the policy
  - Short lists work best (also for agreeing on policy)

Distributed Systems Architecture

It is all about scaling

Managing Complexity Challenges and standards

PUTTING TOGETHER THE GRID INFRASTRUCTURE

Grid Infrastructure

Realizing ubiquitous computing requires a *persistent infrastructure*, based on standards

Organisation

resource providers, user communities and virtual organisation

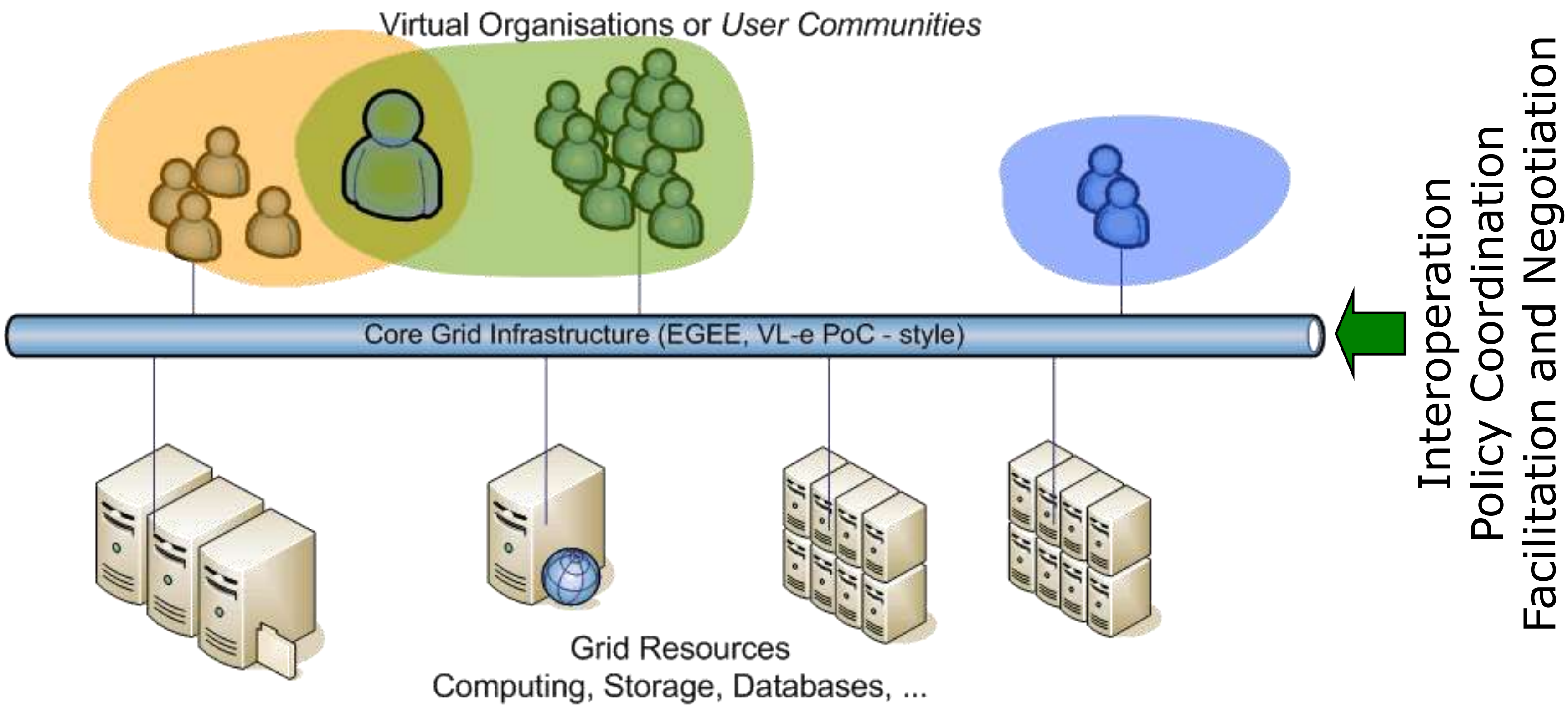
Operational Services

execution services, workflow, resource information systems, database access, storage management, meta-data

Support and Engineering

user support and ICT experts … with domain knowledge

Building Grid Infrastructures



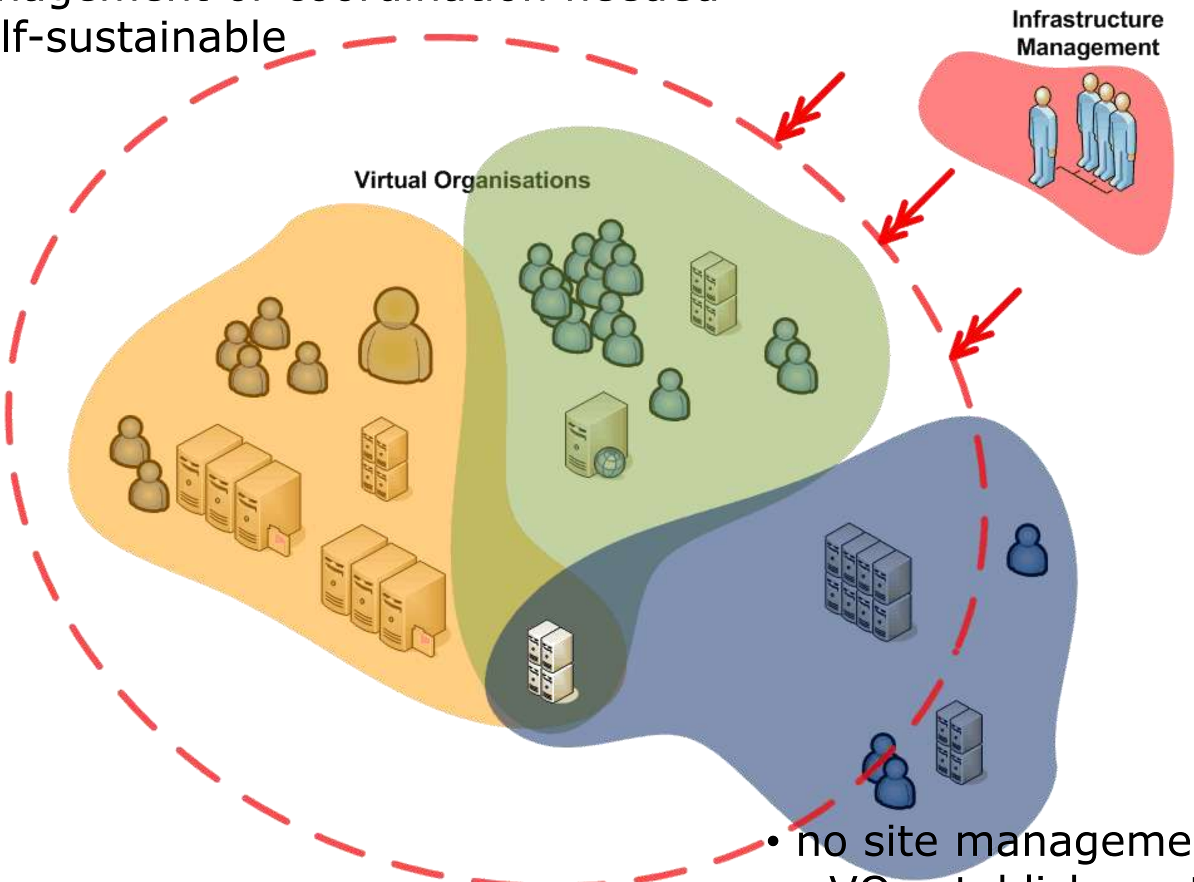
- Interop: common syntax and semantics for grid operations
- Policy Coordination: User and VO AUPs, operations, trust
- Facilitating negotiation: VO meta-data, SLAs, op. environment

VO-centric infrastructure (‘OSG style’)

What happens if you do not coordinate infrastructure from the beginning …

Advantages

- no site management or coordination needed
- VOs are self-sustainable



Disadvantages

- no site management or coordination
- VO establishment is more complex
- infrastructure itself is transient and harder to sustain

Interoperation and standards

- Standards are essential for adoption
  - as resource providers are not inclined to provide *n* different interfaces
- It's an 'emerging system'
  - GIN (Grid Interoperation Now)
    leverage existing de-facto agreements
  - different grid infrastructures need to interoperate
  - stability and consistency vary widely
  - self-healing and verification are largely absent

Example: GlueServiceAccessControlRule

```
For your viewing pleasure: GlueServiceAccessControlRule
261 distinct values seen for GlueServiceAccessControlRule

(one of) least frequently occuring value(s):
1 instance(s) of GlueServiceAccessControlRule:
   /C=BE/O=BEGRID/OU=VUB/OU=IIHE/CN=Stijn De Weirdt

(one of) most frequently occuring value(s):
   310 instance(s) of GlueServiceAccessControlRule: dteam

(one of) shortest value(s) seen:
   GlueServiceAccessControlRule: d0

(one of) longest value(s) seen:
   GlueServiceAccessControlRule: anaconda-ks.cfg configure-
   firewall install.log install.log.syslog j2sdk-1_4_2_08-
   linux-i586.rpm lcg-yaim-latest.rpm myproxy-addons myproxy-
   addons.051021 site-info.def site-info.def.050922 site-
   info.def.050928 site-info.def.051021 yumit-client-2.0.2-
   1.noarch.rpm
```

Example: GlueHostOperatingSystemRelease

```
Today's attribute: GlueHostOperatingSystemRelease
      1  GlueHostOperatingSystemRelease: 3.02
      1  GlueHostOperatingSystemRelease: 3.03
      1  GlueHostOperatingSystemRelease: 3.2
      1  GlueHostOperatingSystemRelease: 3.5
      1  GlueHostOperatingSystemRelease: 303
      1  GlueHostOperatingSystemRelease: 304
      1  GlueHostOperatingSystemRelease: 3_0_4
      1  GlueHostOperatingSystemRelease: SL
      1  GlueHostOperatingSystemRelease: Sarge
      1  GlueHostOperatingSystemRelease: sl3
      2  GlueHostOperatingSystemRelease: 3.0
      2  GlueHostOperatingSystemRelease: 305
      4  GlueHostOperatingSystemRelease: 3.05
      4  GlueHostOperatingSystemRelease: SLC3
      5  GlueHostOperatingSystemRelease: 3.04
      5  GlueHostOperatingSystemRelease: SL3
     18  GlueHostOperatingSystemRelease: 3.0.3
     19  GlueHostOperatingSystemRelease: 7.3
     24  GlueHostOperatingSystemRelease: 3
     37  GlueHostOperatingSystemRelease: 3.0.5
     47  GlueHostOperatingSystemRelease: 3.0.4
```

The Most Popular Site Location

Today's attribute: GlueSiteLatitude

Working at scale

Grid is an error amplifier …

'passive' controls are needed
to push work away
from failing resources

Failure-ping-pong – or *creeper and reaper* revisited

Resource information systems are the backbone of any real-life grid

Grid is much like the 'Wild West'

- almost unlimited possibilities – but as a community plan for scaling issues, and a novel environment
- users and providers *need to interact* and articulate needs

Monitoring Tools

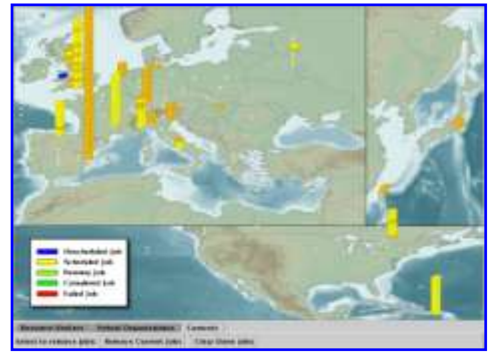
1. GIIS Monitor

2. GIIS Monitor graphs

3. Sites Functional Tests

4. GOC Data Base

5. Scheduled Downtimes



6. Live Job Monitor

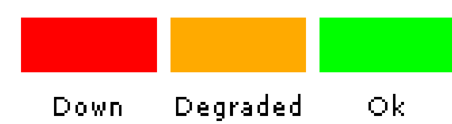
7. GridIce – VO view

8. GridIce – fabric view

9. Certificate Lifetime Monitor

Graphs: Ian Bird, EGEE SA1

Latest SAM results, Site Status, for 'OPS' VO, 27 Sep 2007 13:39 GMT.
Size of site rectangles is number of CPUs from BDII.
Certified Production sites, grouped by regions.



Graphs: Ian Bird, EGEE SA1, EGEE07 Conference October 2007

vl-e virtual laboratory for e-science

BiG Grid the dutch e-science grid

NIKHEF pdp

Managing the complexity

- Whatever the internals:
  the different implementations offer the same *service*

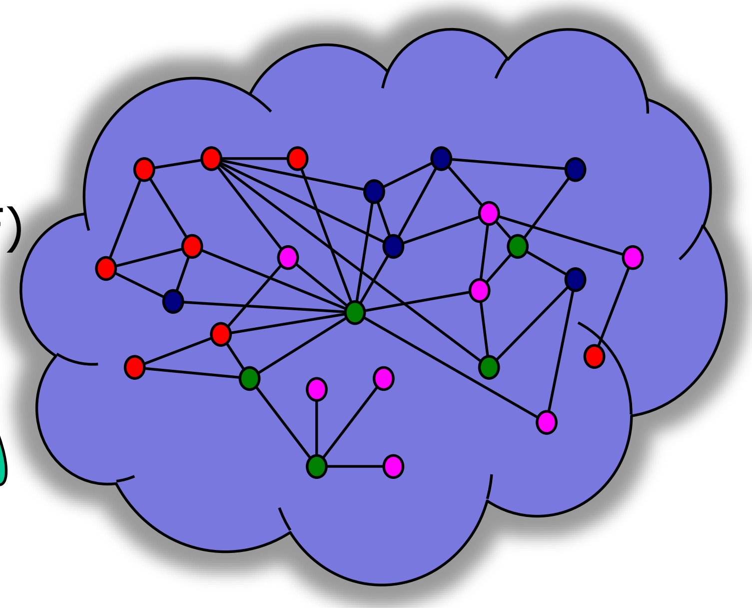
- *Composition of services* in to applications
  will be deciding factor for adoption

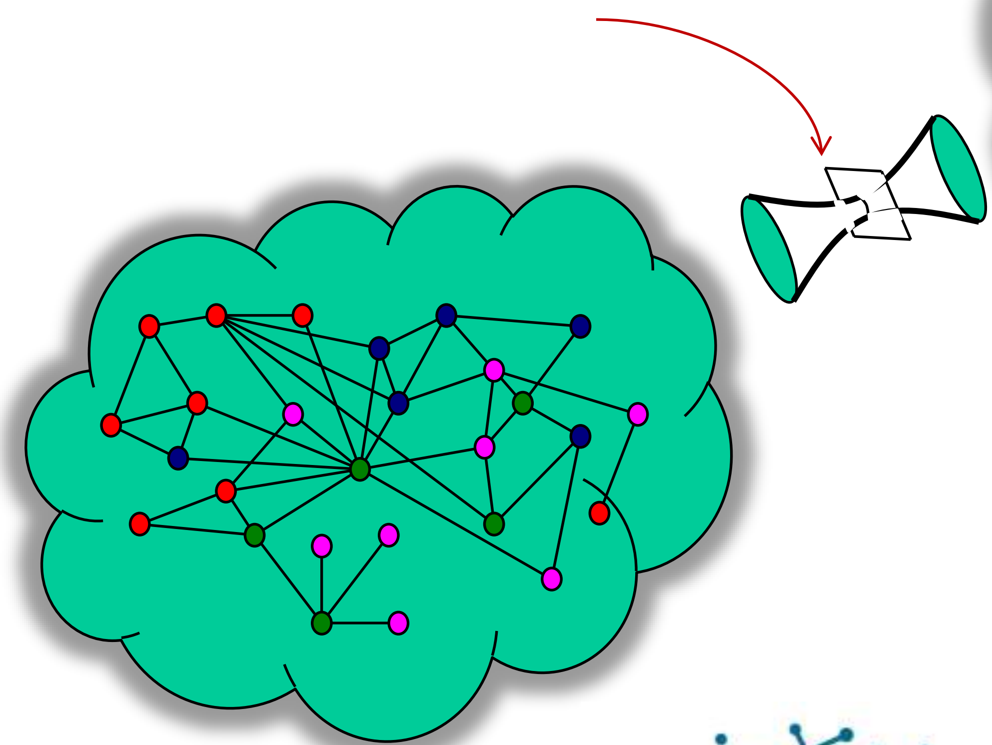
↳ Focus on service *interfaces* at the edge of proprietary fabrics

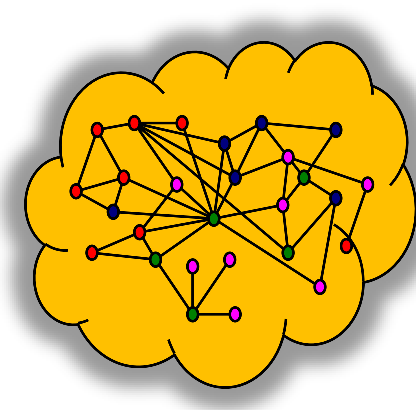
Interoperation – between the clouds?

Open protocols, today mostly

- web services over TLS
- with specific management extensions (WS-Addressing, WS-Notification, WS-RF)

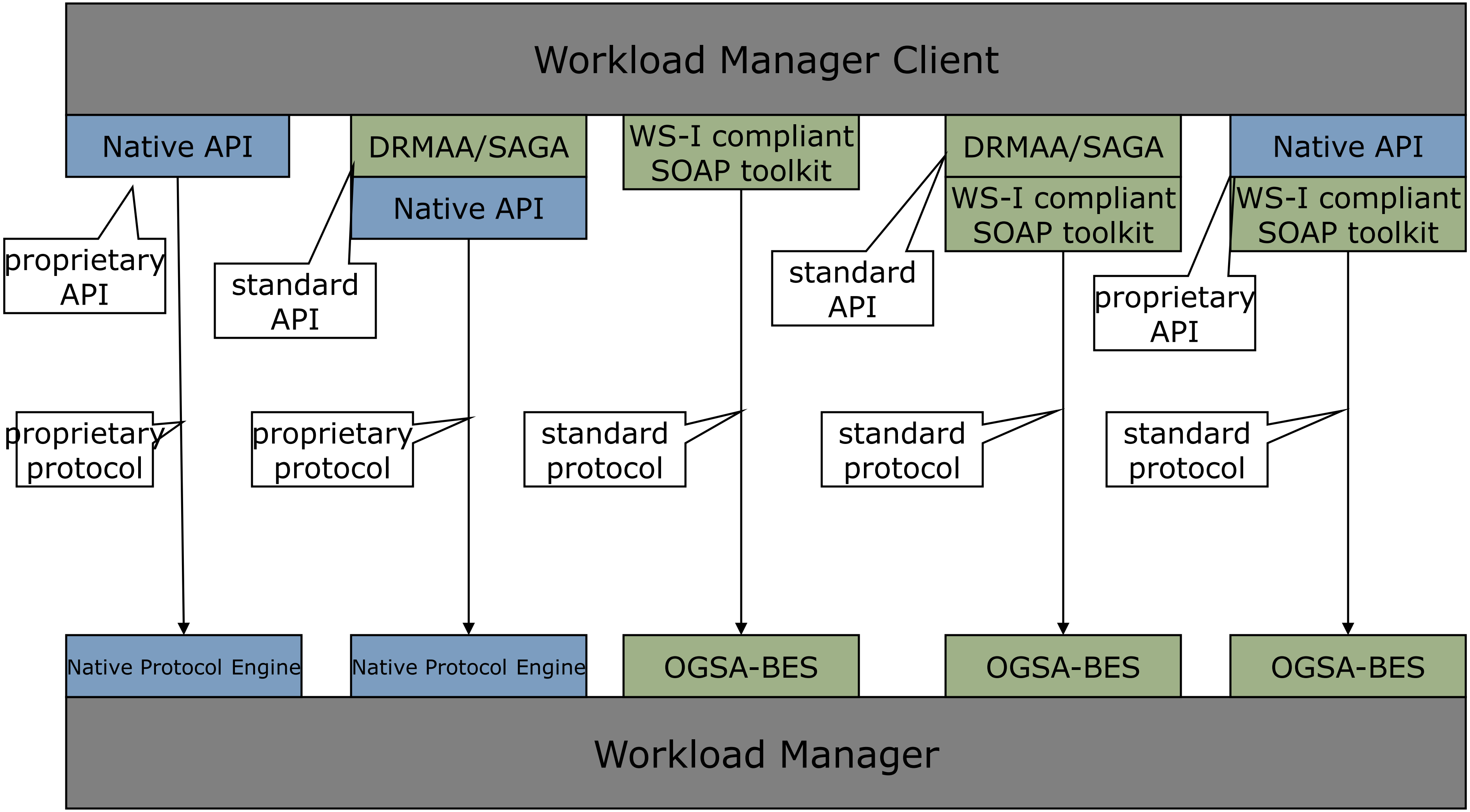






OpenGridForum

Introducing standards



Graphic: Open Grid Forum 2007

Standards

- Standards, such as those by IETF, OASIS, OGF, &c aid interoperability and reduce vendor lock-in

- as you go higher up the stack, you get less synergy
  - Transport: IP/TCP, HTTP, TLS/SSL, &c well agreed
  - Web services: SOAP and WS-Security used to be the solution for all … but 'Web 2.0' shows alternatives tailored to specific applications gaining popularity
  - Grid standards:
    low-level job submission (BES, JSDL), management (DRMAA), basic security (OGSA-BSP Core, SC) there
  - higher-level services still need significant work …

see also http://www.ogf.org/

Why not standardize?

- A technology might be "too new"
  - you stifle innovation with standardization, which focuses on commonality
- A technology might be very niched
  - De-facto standards will emerge in this case and in perhaps not so niche areas like KML in Google maps
- Standards take too long;
  get your product out today and grab market – then your API is the de facto standard
- Organizations with a strong proprietary product might try and succeed derailing standards that would enable competition

Growth of the Infrastructure in Europe

GROWTH

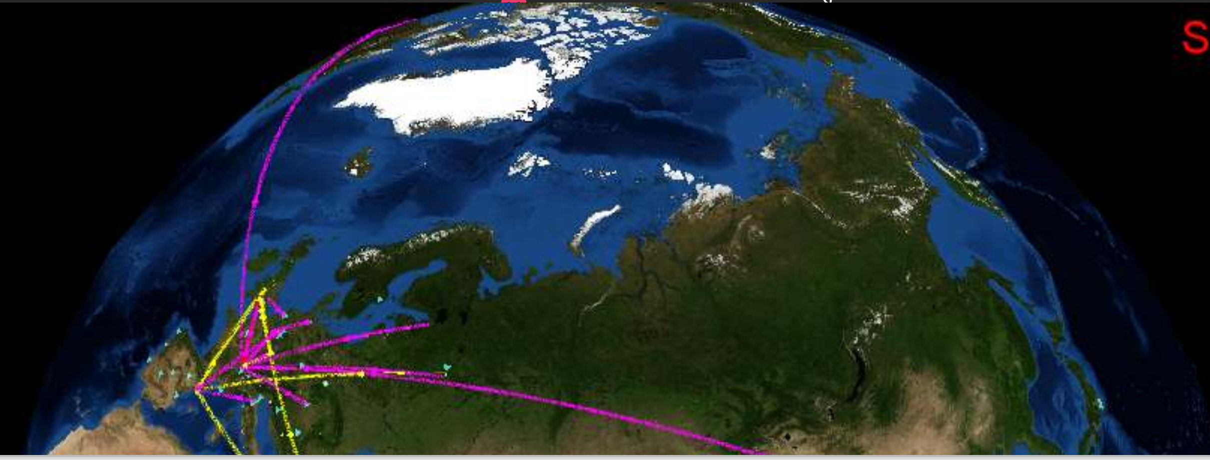
vl·e virtual laboratory for e-science

BiG Grid the dutch e-science grid

NIKHEF pdp

eGee Enabling Grids for E-sciencE

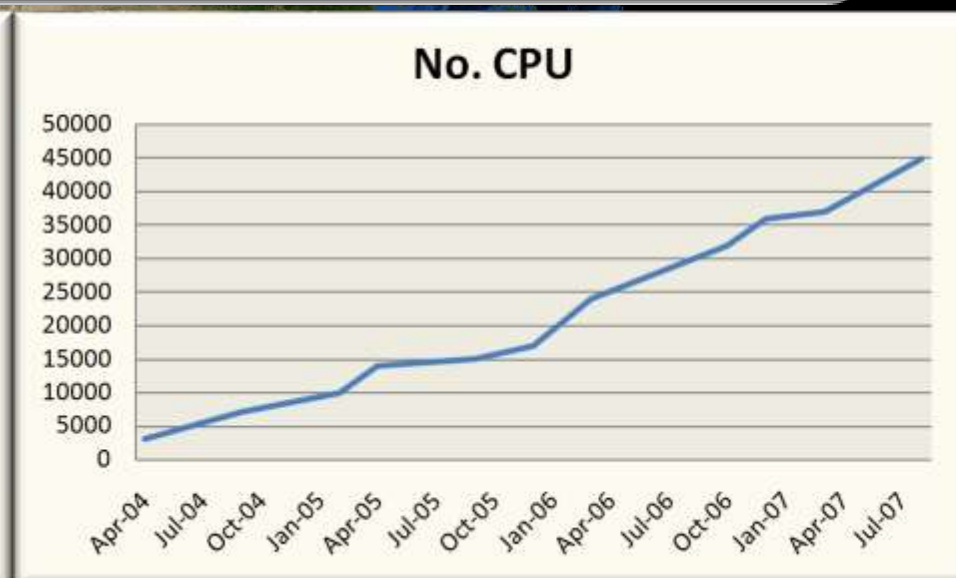
Scheduled = 17356
Running = 18359



EGEE: ~250 sites, >45000 CPU

24% of the resources are contributed by groups external to the project

~>20k simultaneous jobs

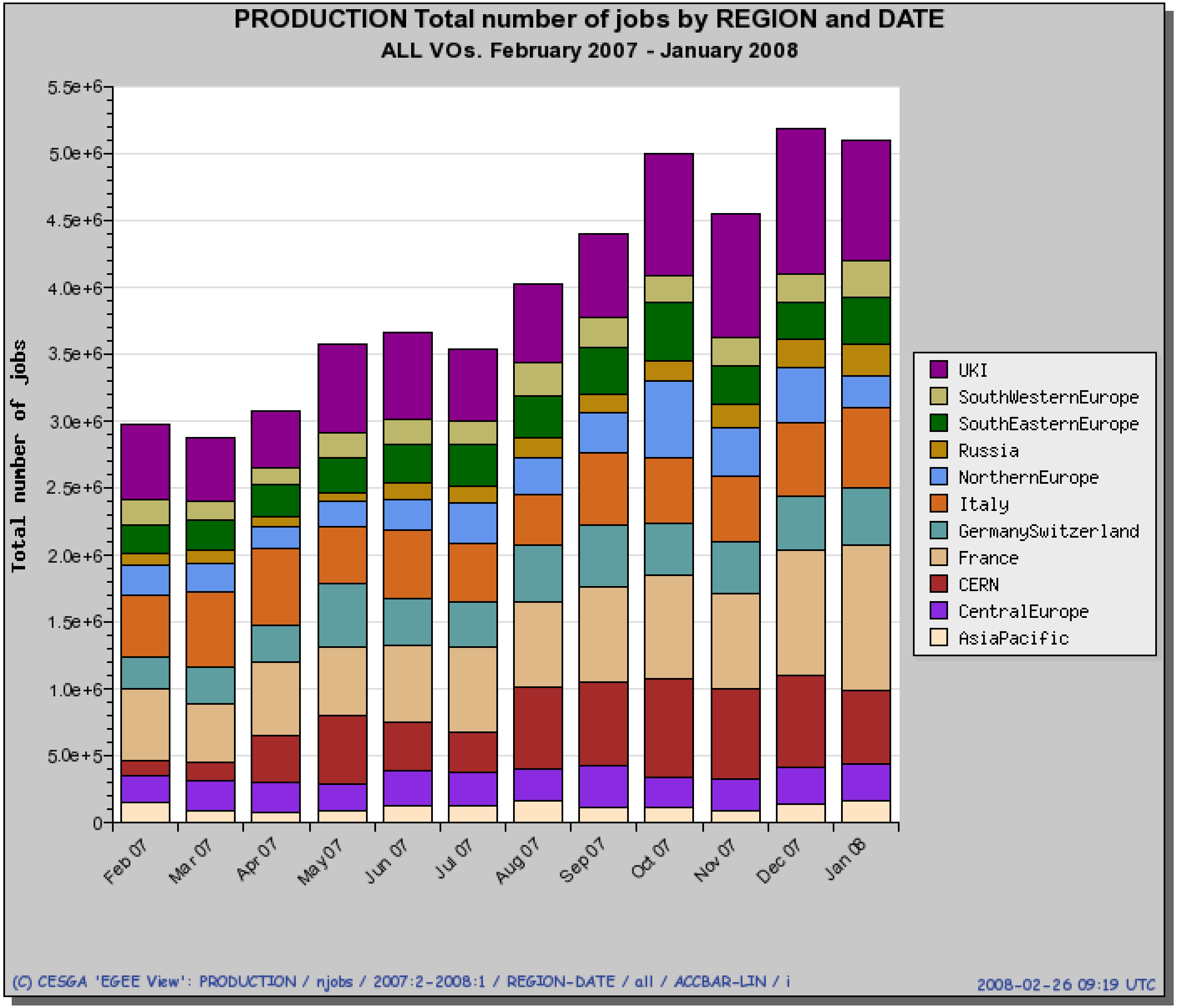


Graphs: Ian Bird, EGEE SA1, EGEE07 Conference October 2007

vl·e virtual laboratory for e·science

BiG Grid the dutch e·science grid

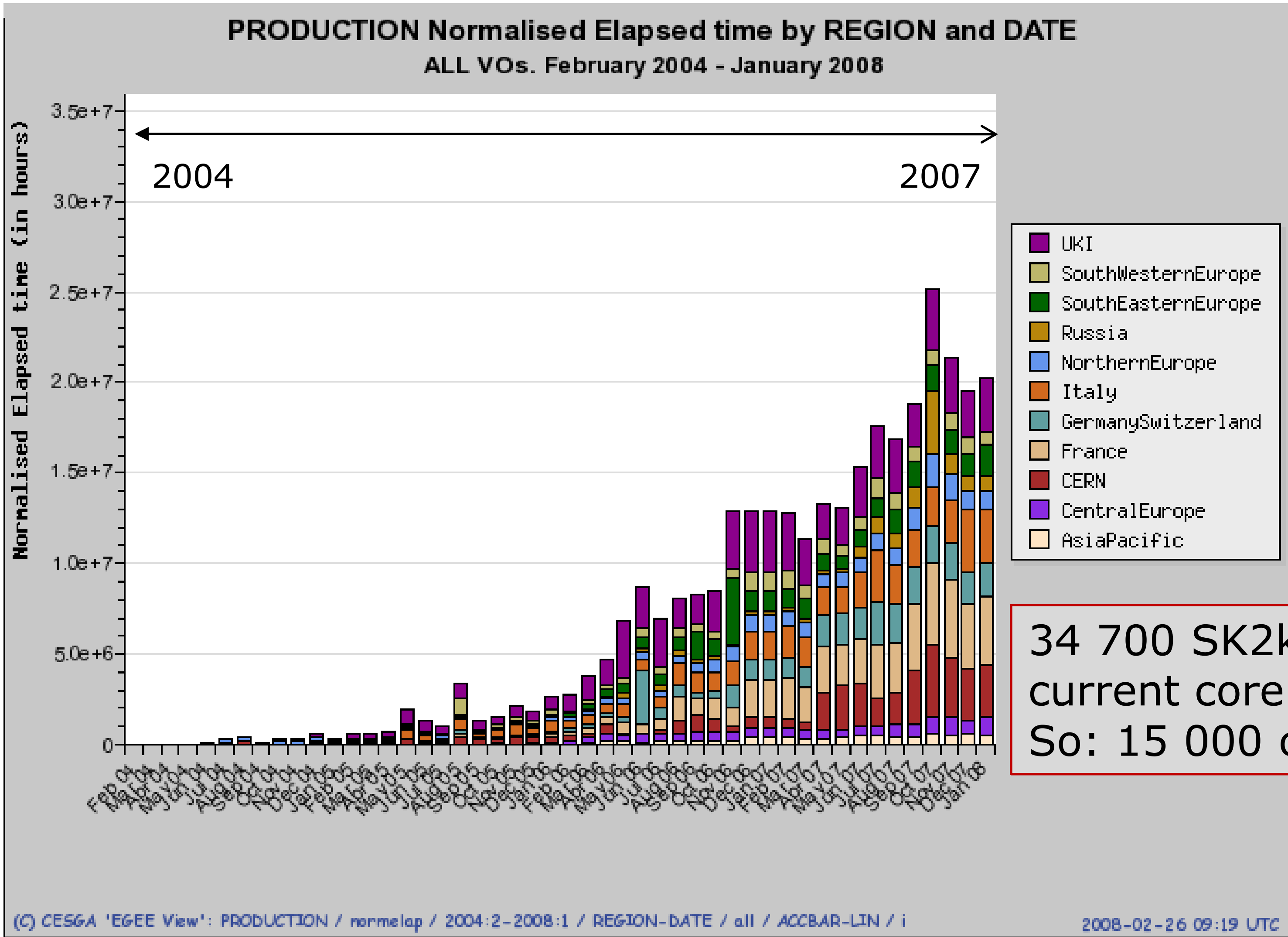
NIKHEF pdp



egee Enabling Grids for E-sciencE

data: EGEE monitoring, RAL and CESGA, http://goc.grid-support.ac.uk/gridsite/accounting/

25 million SI2000 CPU hours reported/month



egee Enabling Grids for E-sciencE

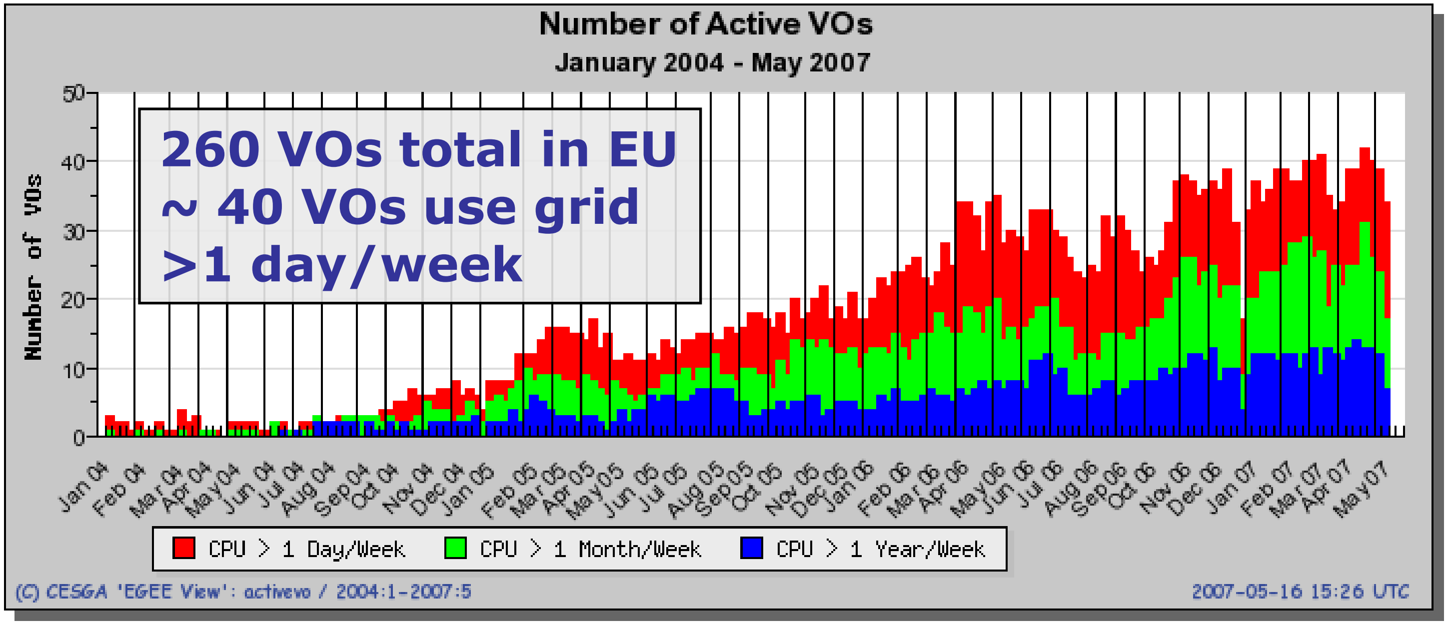
34 700 SK2k hr/hr
current core ~ 2 SI2k
So: 15 000 cores avg.

data: EGEE monitoring, RAL and CESGA, http://goc.grid-support.ac.uk/gridsite/accounting/

LHC and non-LHC

Graphs: Ian Bird, EGEE SA1, EGEE07 Conference October 2007

Grid Infrastructures Works!



egee Enabling Grids for E-sciencE

Number of **active** Vos in EU since 2004

over 20 VOs hosted in NL

A reliable Grid Infrastructure needs operational support:

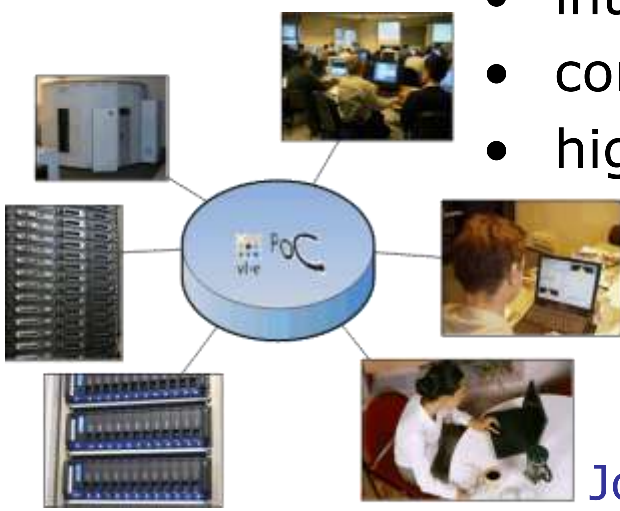
- availability monitoring
- reporting and follow-up
- user support

data: EGEE monitoring, RAL and CESGA, http://goc.grid-support.ac.uk/gridsite/accounting/

Common environment

vl-e PoC

Common infrastructure for e-Science in NL
provided by BiG Grid and the *VL-e Proof-of-Concept*



- interoperable interfaces to resources
- common software environment
- higher-level 'virtual lab' services

Central Facilities:
SARA, NIKHEF, RUG-CIT, Philips

Join yourself: user-interfaces,
distributed clusters, storage

http://poc.vl-e.nl/distribution/

Real Time Monitor 2D link

Does it work

How can we make it better

GOING FROM HERE

Going from here

Many nice things to do:

- In many cases, a single OS (mostly EL4) is a nice feature for users, since they know what they get
  - but users will need SLES, Debian, Gentoo, … or specific libraries
  - and sites don’t want to change OS
- Virtualisation (Xen, VMware) to hide user OS from system OS?

- Scheduling user jobs
  - both VO and site wants to set part of the priorities …

- Auditing and user tracing in this highly dynamic system
  can we know for sure who is running what where? Or whether a user is DDoS-ing the White House right now?
  - Out of 221 sites, we know for certain there is a compromise!

More things to do …

- Sparse file access: access data efficiently over the wide area
- Can we do something useful with the large disks in all worker nodes?
(our 425 cores share ~12 TByte of unused disk space!)
- Transparent (and cheap!) storage access is unsolved!
- There are new grid software releases every month, and the configuration comes from different sources …
how can we combine and validate all these configurations fast and easy?

A Bright Future!

Imagine that you could plug your computer into the wall and have direct access to huge computing resources immediately, just as you plug in a lamp to get instant light. …

Far from being science-fiction, this is the idea the [Grid] is about to make into reality.

The EU DataGrid project brochure, 2001

vl·e

http://www.vl-e.nl/

BiG Grid
the dutch e·science grid

NIKHEF

pdp