



Monitoring Evolution

WLCG collaboration workshop

7 July 2014

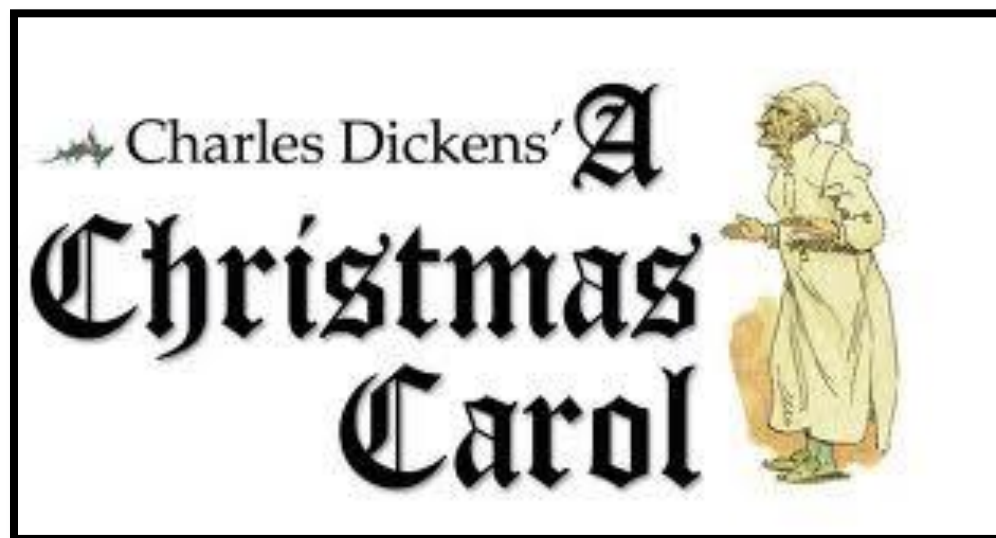
Pablo Saiz

IT/SDC



Monitoring evolution

- Past
- Present
- Future



The past

- **Working** monitoring solutions
 - **Small overlap** in functionality
 - Big diversity in tools/operations/development
 - **Large amount of people** to support it
- Some parts based on **old technologies**

Unsustainable, due to personnel reduction



WLCG Monitoring consolidation group

- Create a team:
 - Mon + Exp. Rep + Ops. Rep + AI Mon
- Main goals:
 - Reduce complexity, modular design
 - Simplify operations, support and service
 - Common development and deployment
 - Unify, where possible, components
- We know we can monitor with the current system:
 - How can we do it with **less resources**?
- <http://go.cern.ch/B6NS>
- 3 months evaluation + 3 months design + 9 months implementation

Design report and plan

- By the end of 2013:
 - Report with summary of current status, and tasks to improve it
 - <https://indico.cern.ch/event/287653/material/2/1.pdf>
- Tasks:
 - Deprecate unused applications
 - Combine applications/tools
 - Evaluate technologies
 - Improve deployment/operations
 - <https://its.cern.ch/jira/browse/WLCGMONCON>

We have a plan!

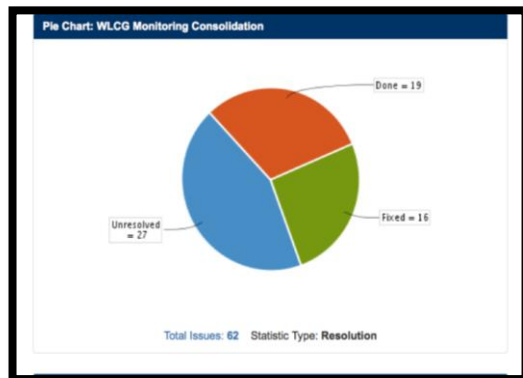


Present

- **Working** monitoring solution(s)
 - In the process of unification
- Huge **improvement** on deployment/operations
 - Using CERN Agile infrastructure
- WLCG SAM decoupled from EGI

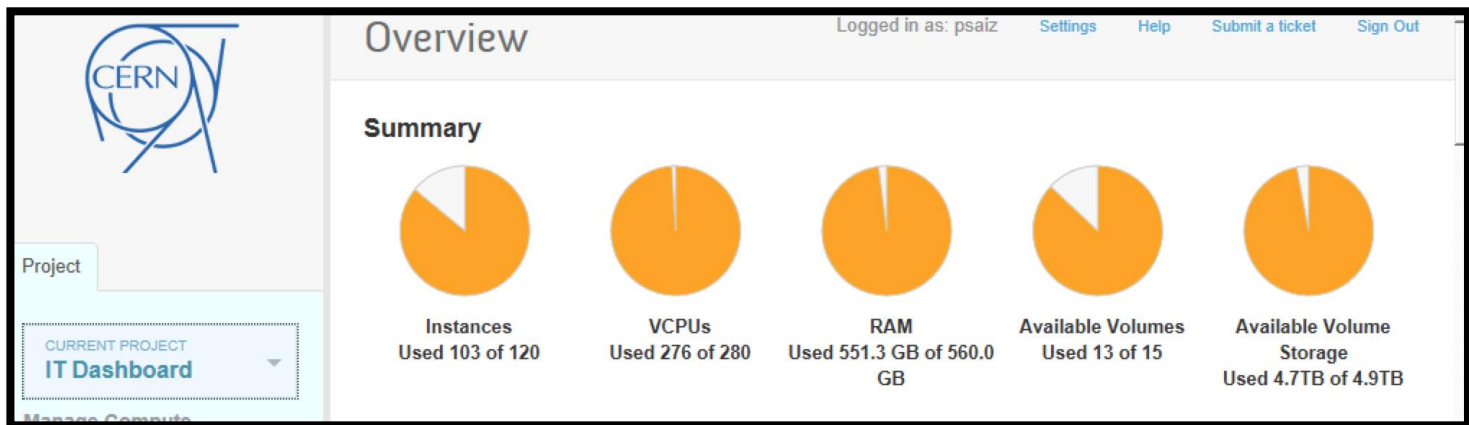


- Using a **common metric store** for multiple applications
- Evaluating different technologies for storage/visualization

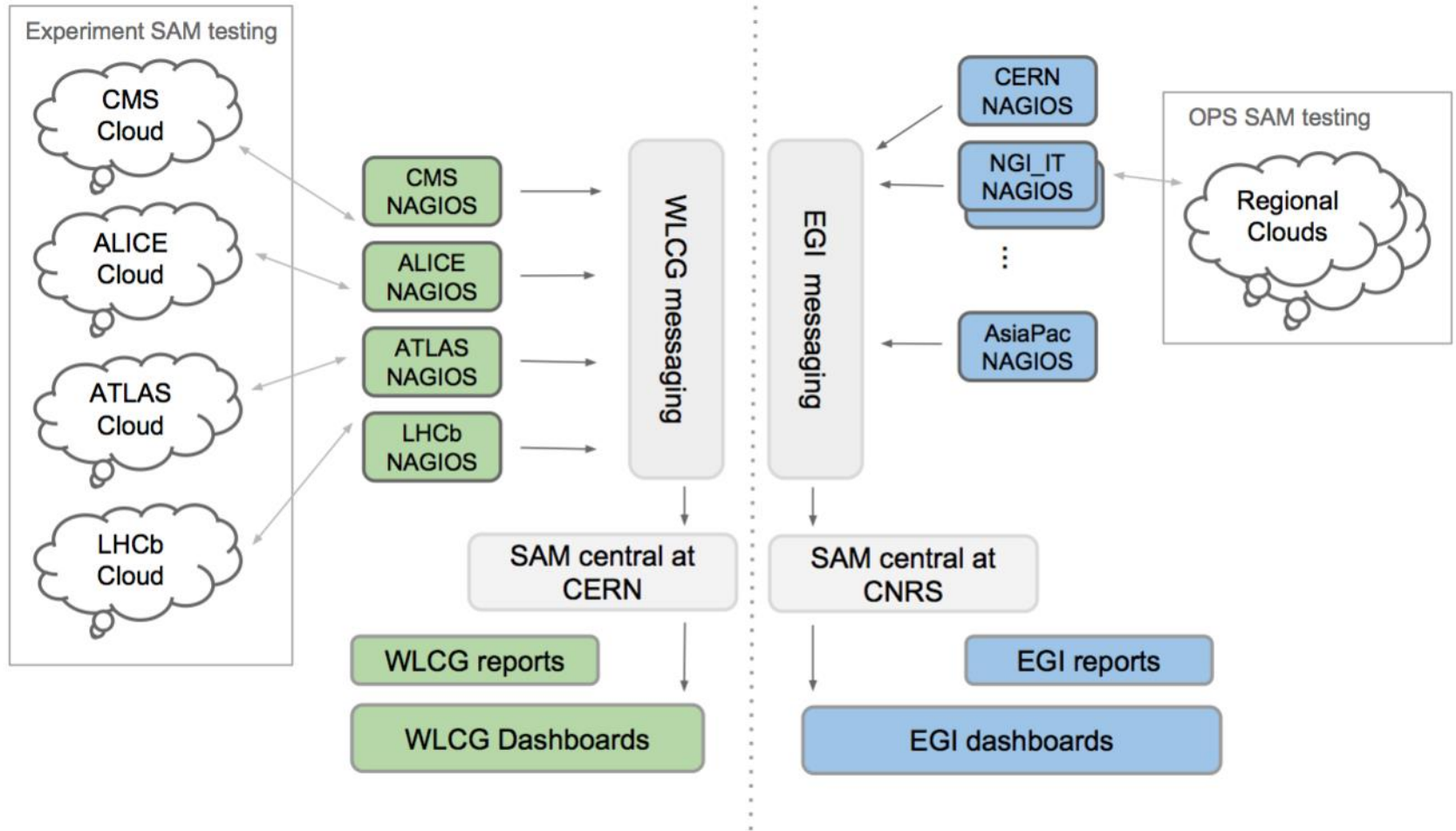


Agile Infrastructure

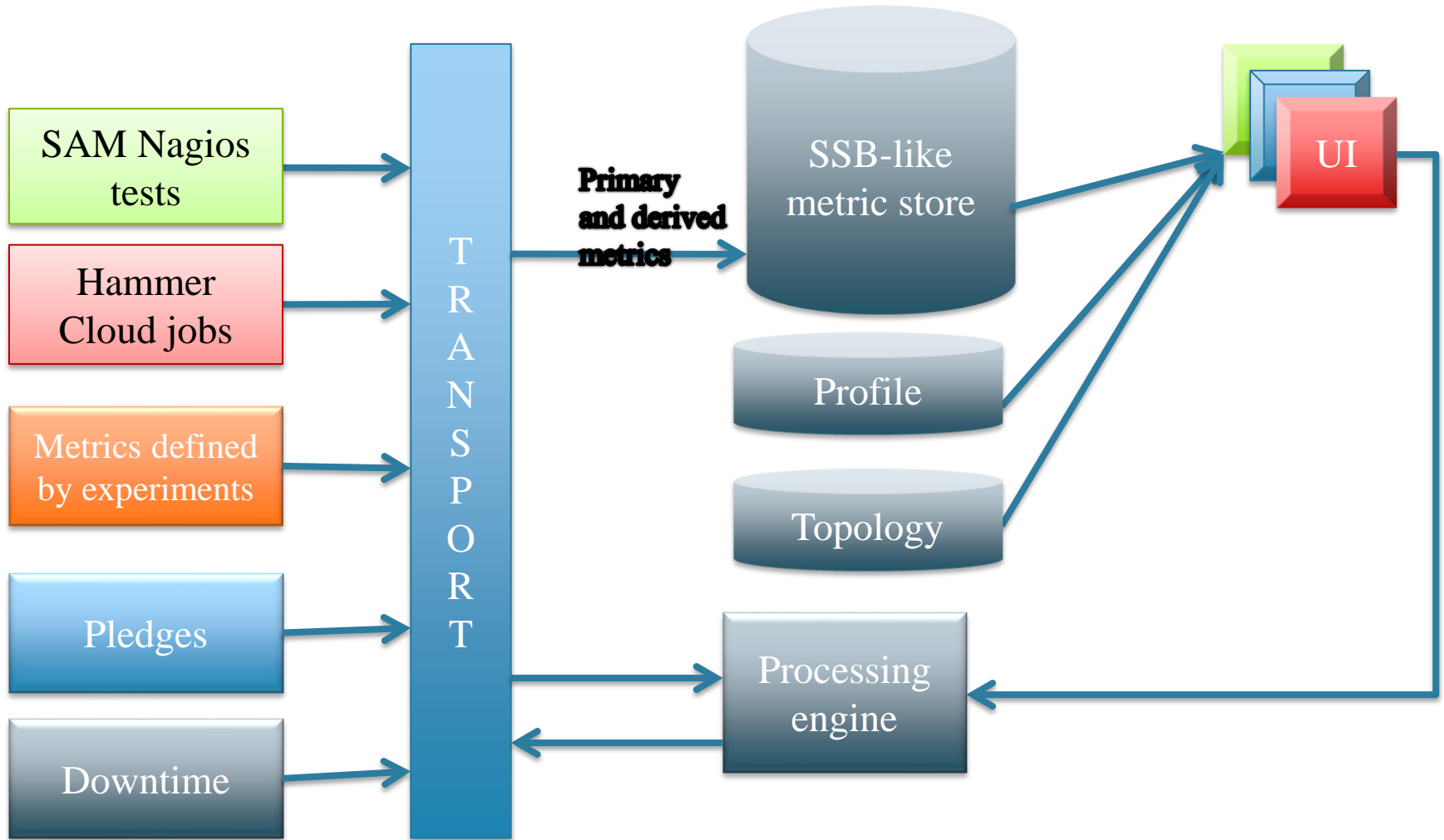
- Over 100 hosts puppetized (into almost 100 hostgroups!)
 - Web portals:
 - SAM3, SSB, Job Monitoring, DDM, REBUS, Hammercloud
 - Services:
 - Dashboard collectors, HC submission nodes
 - Other:
 - Development nodes, build servers (deprecated for koji)
- Using CERN Koji, git, jira, SLC6



Decouple of EGI/WLCG SAM

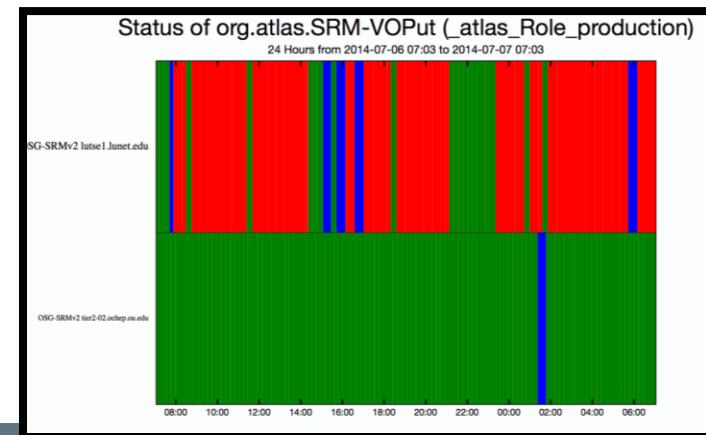


Towards a common metric store



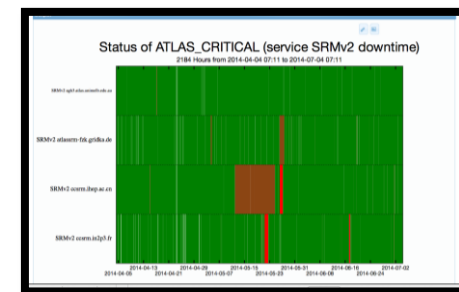
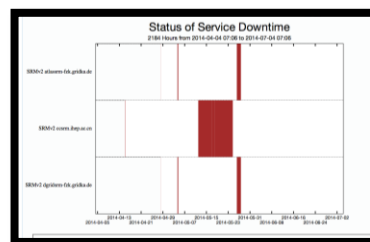
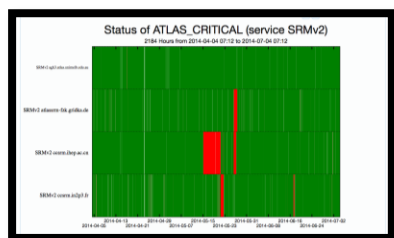
Common format metric store

- Using the Site Status Board as a metric store
- Stores values for instances over a period of time
 - Instances can be sites, services, channels, clouds...
- Provides possibility to combine metrics
 - And also extract them and publish new ones
- Used by
 - Experiments SSB (CMS, ATLAS, LHCb)
 - SAM (test status, availabilities)
 - Downtimes
 - REBUS (pledges, capacities)
 - Glue validator
 - Cloud accounting



Creating Derived Metrics

- Based on some metrics, create new ones:
 - Basic operations provided by SSB
 - AND, OR, OVERWRITE
 - Possibility to fetch data and provide complex calculation
- Derived metrics can be used as input for even more metrics

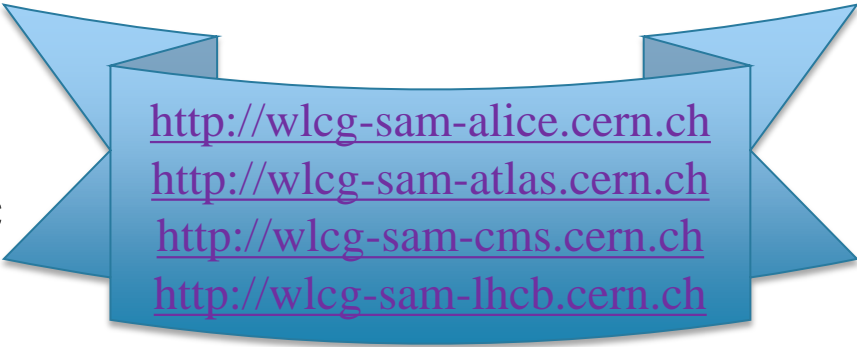


Site usability

- Combine results of tests with real utilization of the site
 - Including metrics published by experiment
- Similar to ‘CMS Site readiness’ and ‘ATLAS site blacklisting’
- ATLAS interested in using this approach for monthly reports
 - To be followed up...

SAM3 interfaces

- Very similar interface to SUM
 - With **FQAN and any service**
 - Using SSB as a common metric store
- Using the tests results from production
- Including all current profiles
 - And even more!
 - Profiles could be based on any metric
- Availability and reliability** calculated
 - In the process of validation



Site Usability Monitor v0.4.0_rc21

VO view Historical View Feedback Help Bugs

Site-Groups Profiles Metric Exit Status

Site-Groups	Profiles	Metric Exit Status
All Groups: CMS_CRITICAL	All Service Flavours	All Exit Status: OK, WARNING, UNKNOWN, CRITICAL
All Sites: TO_CH_CERN, T1_CH_CERN, T1_CH_CERN, T1_CH_CERN, T1_CH_CERN, T1_CH_CERN, T1_CH_CERN	All Metrics: org.cms.SRM-VOGet, org.cms.SRM-GetPFNFromTFC, org.cms.SRM-VOPut, org.cms.SRM-GetPFNFromTFC, org.cms.SRM-VOPut, emi.cream.CREAMCE-JobSubmit, org.cms.WN-env	

Algorithm for calculating the Site and Service Availability

Status:	NA	OK	WARNING	CRITICAL	UNKNOWN	MISSING	MAINTENANCE	REMOVED
Legend:	NA	OK	W	C	U	M	MT	RM

Note: brightest colors: test is 0 - 12 hours old, ... lightest colors: test is more than 12 hours old

Legend	Metric Name	Legend	Metric Name	Legend	Metric Name	Legend	Metric Name	Legend	Metric Name
1	emi.cream.CREAMCE-JobSubmit (/cms/Role_production)	2	emi.ce.CREAMCE-JobSubmit (/cms/Role_production)	3	org.cms.SRM-VOGet (/cms/Role_production)	4	emi.cream.CREAMCE-JobSubmit (/cms/Role_production)	5	emi.ce.CREAMCE-JobSubmit (/cms/Role_production)
6	org.cms.WN-env (/cms/Role_production)	7	org.cms.SRM-VOPut (/cms/Role_production)	8	org.cms.SRM-GetPFNFromTFC (/cms/Role_production)	9	org.cms.glexec.WN-gLExec (/cms/Role_production)		

Link to data

SiteName	Flavour	Host status in profile	Hosts	9	4	1	6
TO_CH_CERN	CREAM-CE	CMS_CRITICAL	ce205.cern.ch	OK	OK	OK	OK
		UNKNOWN	ce202.cern.ch	OK	OK	OK	OK
		OK	ce201.cern.ch	OK	OK	OK	OK
		OK	ce302.cern.ch	OK	OK	OK	OK
		OK	ce301.cern.ch	OK	OK	OK	OK
		UNKNOWN	ce207.cern.ch	OK	OK	OK	OK
		UNKNOWN	ce204.cern.ch	OK	OK	OK	OK
		OK	ce203.cern.ch	OK	OK	OK	OK
		UNKNOWN	ce206.cern.ch	OK	OK	OK	OK



New Nagios Plugin

- Bash script downloads JSON files for each of the experiment's SAM (SUM) dashboard
- After parsing it, it get status of each individual check in a given profile
- It sends status to local Nagios system

Slide from
Jordi Casals
(PIC)

<http://cern.ch/go/N8r7>

Link to data

Dashboard JSON

```
{
  "data": {
    "siteName": [
      "T1_ES_PIC"
    ],
    "results": [
      {
        "sitename": "T1_ES_PIC",
        "flavours": [
          {
            "hosts": [
              {
                "metric": [
                  {
                    "status": "OK",
                    "timestamp": "2014-04-09T09:43:15Z",
                    "age": 0,
                    "Abbr": 2,
                    "metric_name": "emi.cream.CREAMCE-JobSubmit"
                  }
                ]
              }
            ]
          }
        ]
      }
    ]
  }
}
```

Nagios Plugins LCG SAM

```
...
...
# Get JSON content using input
args
json=$(wget -qO - "http://dashb-
$vo-
sum.cern.ch/dashboard/request.py/1
atestresultssmry-sumjson?
profile=$profile&flavour=$flavour&
site=$site")
...
```

PIC Nagios

```
WARNING (for 0d 23h 55m 53s)
CREAM-CE: WARNING
HOSTS
=====
CREAM-CE
ce07.pic.es: WARNING
ce08.pic.es: WARNING
ce09.pic.es: WARNING
ce10.pic.es: WARNING
ce11.pic.es: WARNING
PROBLEMS
=====
CREAM-CE - ce07.pic.es - org.cms.WN-xrootd-access: W
CREAM-CE - ce08.pic.es - org.cms.WN-xrootd-access: W
CREAM-CE - ce09.pic.es - org.cms.WN-xrootd-access: W
CREAM-CE - ce10.pic.es - org.cms.WN-xrootd-access: W
CREAM-CE - ce11.pic.es - org.cms.WN-xrootd-access: W
```


And even more displays

Livestatus

- mk_livestatus creates socket to which you can attach to query Nagios data
- For Naemon addressable at
`/var/cache/naemon/live`
- Access socket via normal methods: used Python



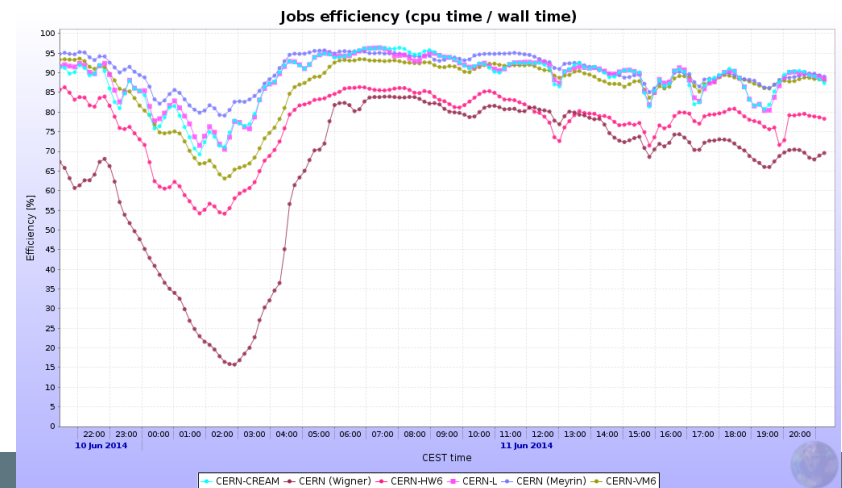
Slides from David Crooks (Glasgow)
<http://cern.ch/go/7DSgc>

Final result



Wigner efficiency analysis

- Effort involving multiple groups
 - <https://twiki.cern.ch/twiki/bin/view/PEsgroup/JobEfficiencyTestsMeyrinVSWigner>
- Correlate job (cpu/wall), events/second, grid id, local batch id, SLCX, virtual/physical, EOS access, network status
- ATLAS will do functional tests in Geneva and Wigner, passing job Id and Panda id



CVMFS Monitoring

Prototype

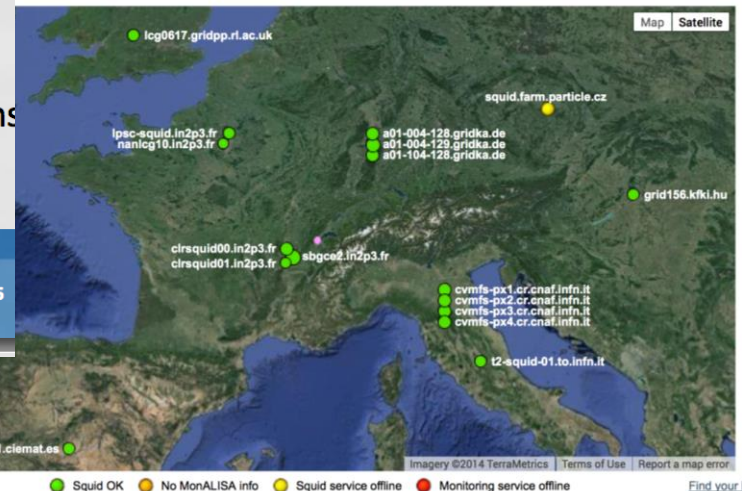
- <http://cvmfsmon.cern.ch/>
- Very simple package to deploy on the Squid proxy server
 - Either as RPM or as a script to run locally
- Based on MonALISA monitoring service
 - All host monitoring modules enabled
 - New Squid module querying the status 1/minute
 - Network topology discovery enabled for the group
- Installed on 35 servers already (5 T1 sites)
 - Many thanks for the help of Subatech, BITP and ISS site admins during the initial iterations and to all who have deployed and provided feedback!

Slides from
Costin Grigoras (CERN)
See ALICE talk tomorrow
for more details

CVMFS infrastructure monitoring

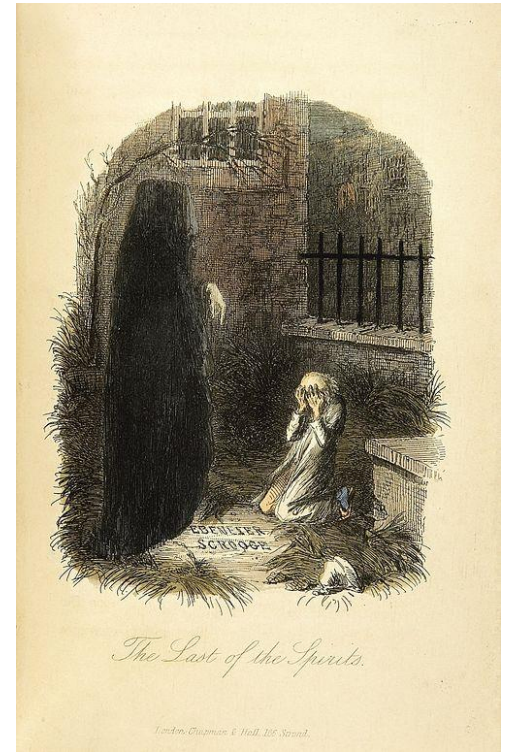


Repository home | Download monitoring package | MonALISA GUI



The future

- One monitoring solution
- Easy to maintain/evolve
- Modular design
 - Plugin/replace components
- Possibility to correlate data
- Providing all required information to end user
 - Including Real Time Analytics



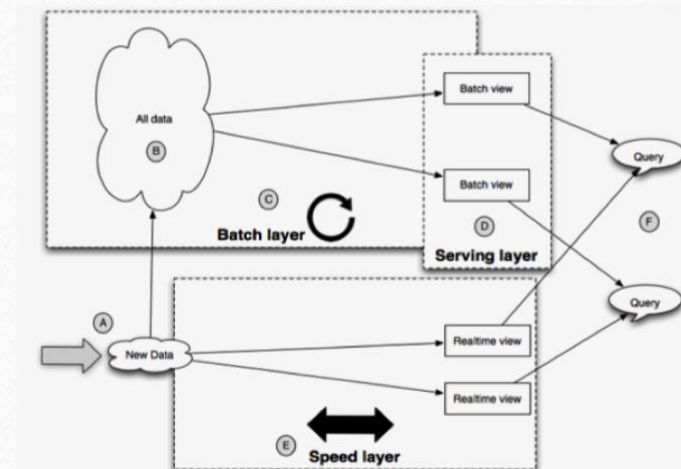
Data Analytics

Slide from
Uthay Suthakar (Brunel)
Luca Magnoni (CERN)
<http://cern.ch/go/67Gx>

Lambda Architecture

Three layers architecture:

- **Batch Layer** – for batch processing on Big Data and producing queryable views.
- **Serving Layer** – for ad-hoc query (ideally from views generated by the batch layer).
- **Speed Layer** – for real-time views based on incremental algorithms.



Lambda Architecture for handling Big Data: Proposed by Nathan Marz (Twitter)

28/04/2014

6

Prototype with xrootd
monitoring data

Correlations

- Are low-efficiency jobs related to storage overload, remote access, hardware type, site?
- Does the ATLAS transfer rate depend on the number of jobs of another experiment?
- What is the percentage of pledges used for MC at a particular site?
- Do the jobs of the same user have different efficiency depending on date/site/concurrent jobs?

Technology evaluation

- Keeping an eye on technology evolution
- Storage:
 - ElasticSearch, hdfs, Impala, Riak
- Visualization:
 - Ember, Angular
- Real time analytics:
 - Storm, ESPER

Summary

- Consolidating WLCG monitoring
 - Working towards a unified monitoring solution
- Plenty of work already done:
 - Service deployment/operations on AI
 - Using common tools: Koji, git, jira
 - Using common metric store
 - With different UI tailored to the needs
- And more to come
 - Data analytics
 - Correlations
 - Technology evolution