

Monitoring and Messaging

Introduction and Plans

Alberto AIMAR, IT-CM-MM

People (it-dep-cm-mm-all)

Alberto Aimar

Borja Garrido Bear

Daniel Zolnai

Edward (**Eddie**) Karavakis

Hassen Riahi

Jaroslava (**Jarka**) Schovancova

Javier Rodriguez Martinez

Lionel Cons

Luca Magnoni

Marian Babik

Maria-Varvara (**Mary**) Georgiou

Pablo Saiz

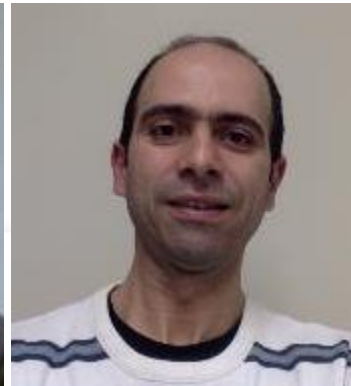
Pedro Rodrigues De Sousa Andrade

Rocio Rama Ballesteros

Sergey Belov

Valentina Mancinelli

People (mugshots 😊)



Mandate at a Glance

- Regroup monitoring activities
- Continue (and review) existing services
- Be more integrated with IT practices
 - services, communication, management, tools

Starting with:

- Merge IT and Grid monitoring
- Increase transparency and work across services, projects, activities
- Collaborate with other services in the group (and beyond)

Mandate as Functional Elements

2015

Monitoring tools
Data Centre Tools
Evaluation and Consultancy on Third Parties Messaging Software
Message Broker Infrastructure
Messaging Software Development and Customization
WLCG HammerCloud

WLCG Grid Monitoring
WLCG Job Monitoring
WLCG Data Management Monitoring
WLCG Google Earth Dashboard
WLCG Infrastructure Monitoring
WLCG Network Monitoring
WLCG Experiment Probe Submission Fmk (ETF)

2016

Monitoring
Messaging
HammerCloud
ETF

WLCG Development, glExec, Elog Operations, REBUS,
Commercial Cloud Provisioning
CMS Integrated Data Transfers and CMS DM (ASO)

Outline

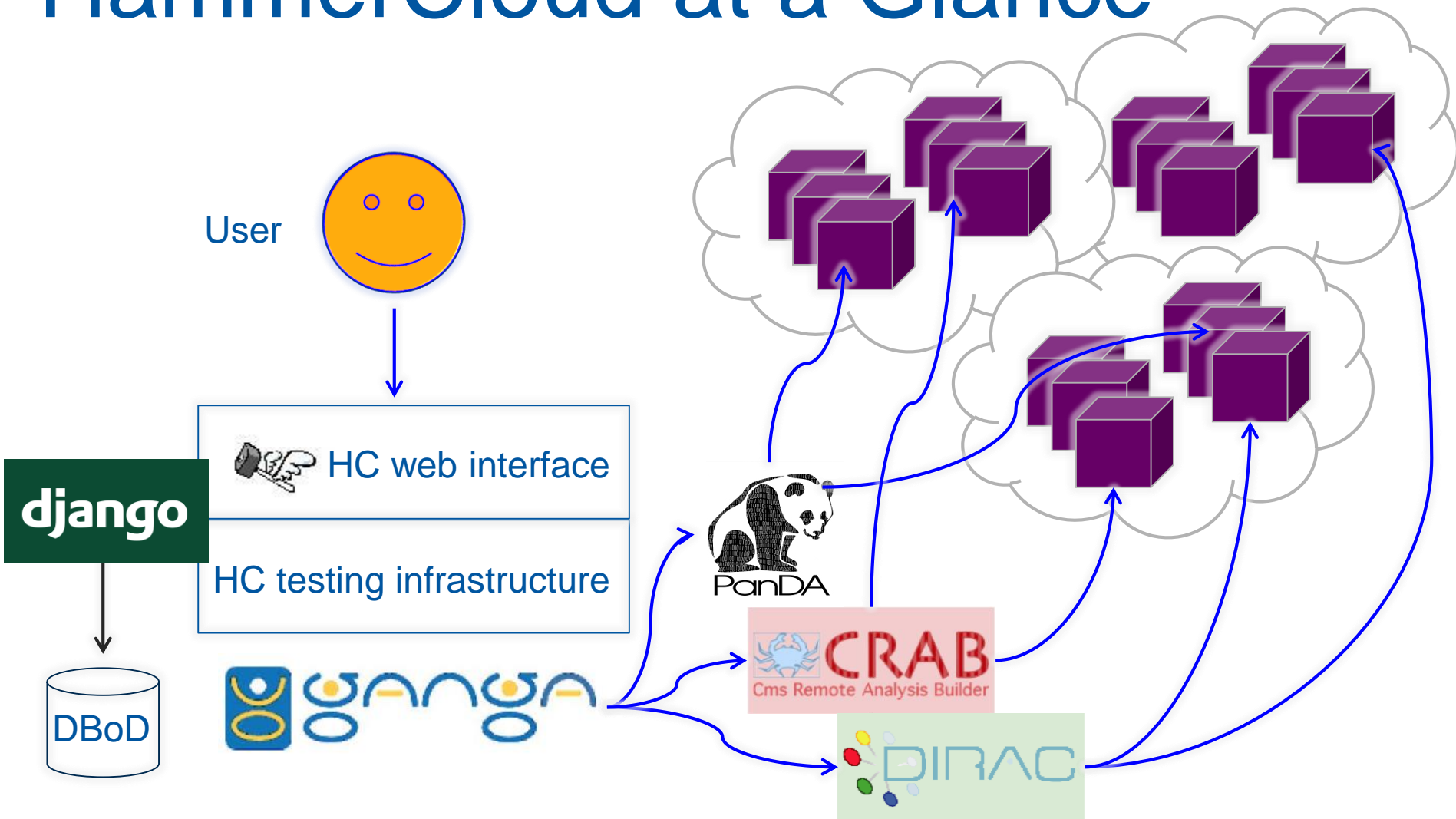
- HammerCloud
- Messaging
- ETF
- Network and Transfers WG
- ASO
- Monitoring
 - IT Monitoring
 - WLCG Monitoring

HammerCloud



- Functional and stress tests
 - WLCG resources: ATLAS, CMS, LHCb
 - Functional: steady flow of test jobs
 - Stress: on demand tests, configure load intensity
- Testing the full chain of an experiment job
 - Same environment as the “real” analysis/prod jobs
- Numbers
 - ATLAS: 52k jobs/day, 65 tests/day
 - CMS: 39k jobs/day, 36 tests/day
 - LHCb: limited usage, being restarted

HammerCloud at a Glance



HammerCloud Usage

Essential test and automation tool

- for the experiments' computing operations

Commission new systems and technologies

- XRootD, FAX, pilot, compare Meyrin and Wigner, ...

ATLAS: functional and stress tests

- blacklisting/recovery of sites
- software release testing
- benchmarking

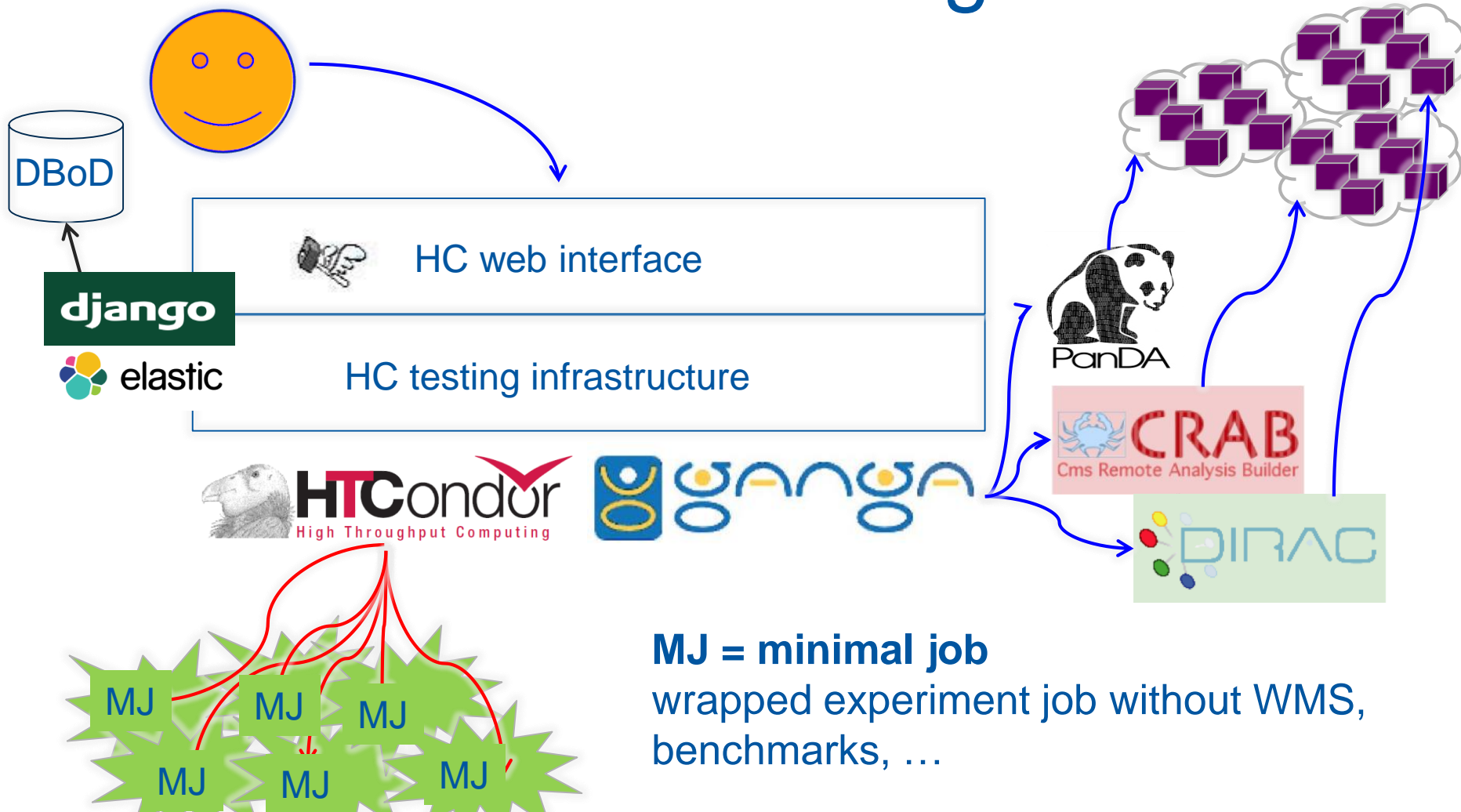
CMS: functional and stress tests

- Site readiness report
- XRootD tests

HammerCloud Plans

- Standard service operations → SNOW, rota
- System and tools upgrades (CC7, git → GitLab)
- Review privacy and security (log visibility, SSO)
- Experiment specific updates
 - ATLAS: unify topology and status sources, facelift, blacklisting
 - CMS: python client replaced REST API, provides more monitoring info
 - LHCb: re-activate HC activity
- Review data store, visualization → copy to ELK
- HC for “grid sites” → HC for “grid sites and generic computing resources”

Resource Commissioning with HC



MJ = minimal job
wrapped experiment job without WMS,
benchmarks, ...

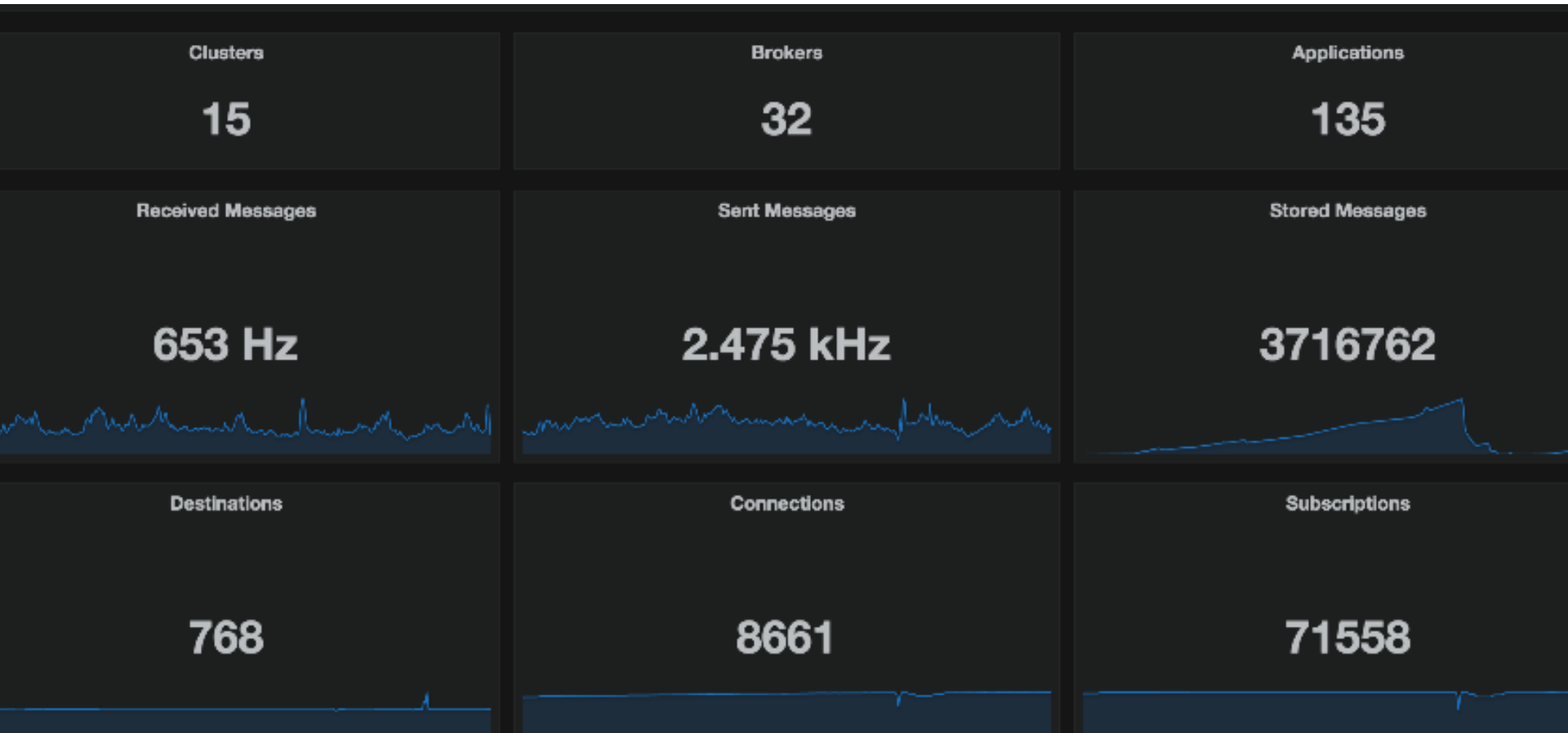
Commercial cloud VMs, new HW, check versions, etc

Messaging

Messaging Service for a large variety of different use cases, such as:

- AI: events, monitoring notifications
- ATLAS: Rucio, event indexes
- Dashboard: FTS, Transfer, SAM, XRootD
- DSS: CASTOR/EOS logs
- MCollective
- WLCG Network Monitoring: perfSONAR
- Messaging monitoring

Messaging Services Numbers



Hardware and Software Layer

Physical machines for the brokers

VMs for the monitoring nodes

Currently Java 7 on SL 6

Red Hat's A-MQ which is basically Apache ActiveMQ patched, tested and supported by Red Hat.

A-MQ 6.2.1 which is based on ActiveMQ 5.11.

Messaging Plans

1. Prepare for the upcoming system upgrade
 - CentOS 7 and Java 8
2. Evaluate ActiveMQ Artemis
3. Investigate Kafka-based messaging services
4. Improve and extend our building blocks
5. Test high-end VMs with better I/O
 - fast Ceph volumes
 - local SSD drives
6. Extend Metis beyond messaging monitoring

WLCG Network and Transfers

WLCG Network and Transfer Metrics WG ([link](#))

Define and understand slow transfers

- Established and operating large perfSONAR network (264 sonars)
- Measuring achievable bandwidth, mapping links and latency
- Open to all sites and experiments via perfSONAR interfaces

Baseline existing links, help commission new links

- Commissioned full mesh testing of latencies, traceroutes and throughput
- Running core networking meshes (LHCOPN/LHCONE)

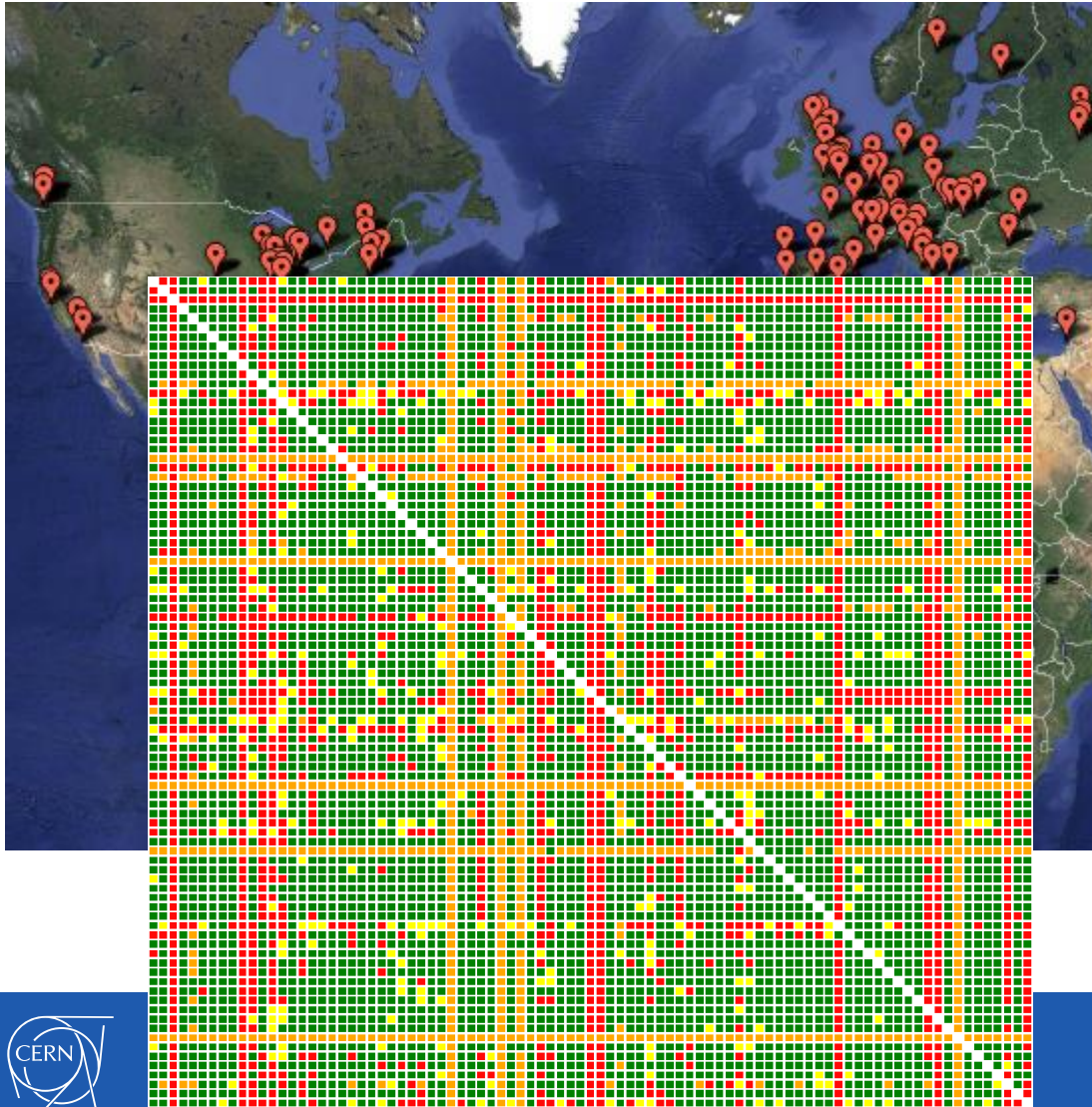
Uniform way to access and integrate network measurements

- Proximity service providing common topology to match sonars and storages
- Providing public interfaces to all perfSONAR measurements

Coordinate response to network performance issues

- WLCG Network Throughput SU in GGUS and underlying procedure

WLCG perfSONAR network



WLCG perfSONAR network

- 264 sonars world-wide
- 95 sonars testing latency/loss
8930 links measured at
10Hz
packet-loss, one-way
latency, jitter, ttl, packet-
reordering
- 115 sonars testing traceroutes
13110 links
hourly traceroutes, path-mtu
- 102 sonars testing bandwidth
10920 links
iperf3 throughput,
tcp-retransmits

**Long-term efficiency now at
90%**

**Consistent and stable
operations**

WLCG Network WG Plans

- Integration with transfer systems and pilot projects in collaboration with LHC experiments
- Integration of the higher-level infrastructure services (network alerting/notifications)
 - With existing projects in this area (MadAlert, Pundit)
- Enabling easy integration with analytical platforms
 - ElasticSearch, Hadoop/Spark
- Follow up on production deployment of the base infrastructure
- Upgrade WLCG network to perfSONAR 3.6

Experiments Testing Fmwk (ETF)

- Test framework = measurement service
 - Responsible to actively check status of services
 - Focusing on functional (remote) testing of core infrastructure
 - Direct job submission
 - Worker node environment
 - Basic storage operations
 - Generic framework based on open source tools
- Common to all experiments
- Main source for WLCG Availability/Reliability monthly reports

ETF Structure and Numbers

Core Framework

- based on Nagios-core and check_mk
- includes configuration, scheduling, notifications, APIs

Probes/plugins

- based on nagios plugins standard
- wide range of plugins available - covering broad range of services (including cloud/grid services)
- contributed by experiments, product teams, task forces, Nagios community, etc.

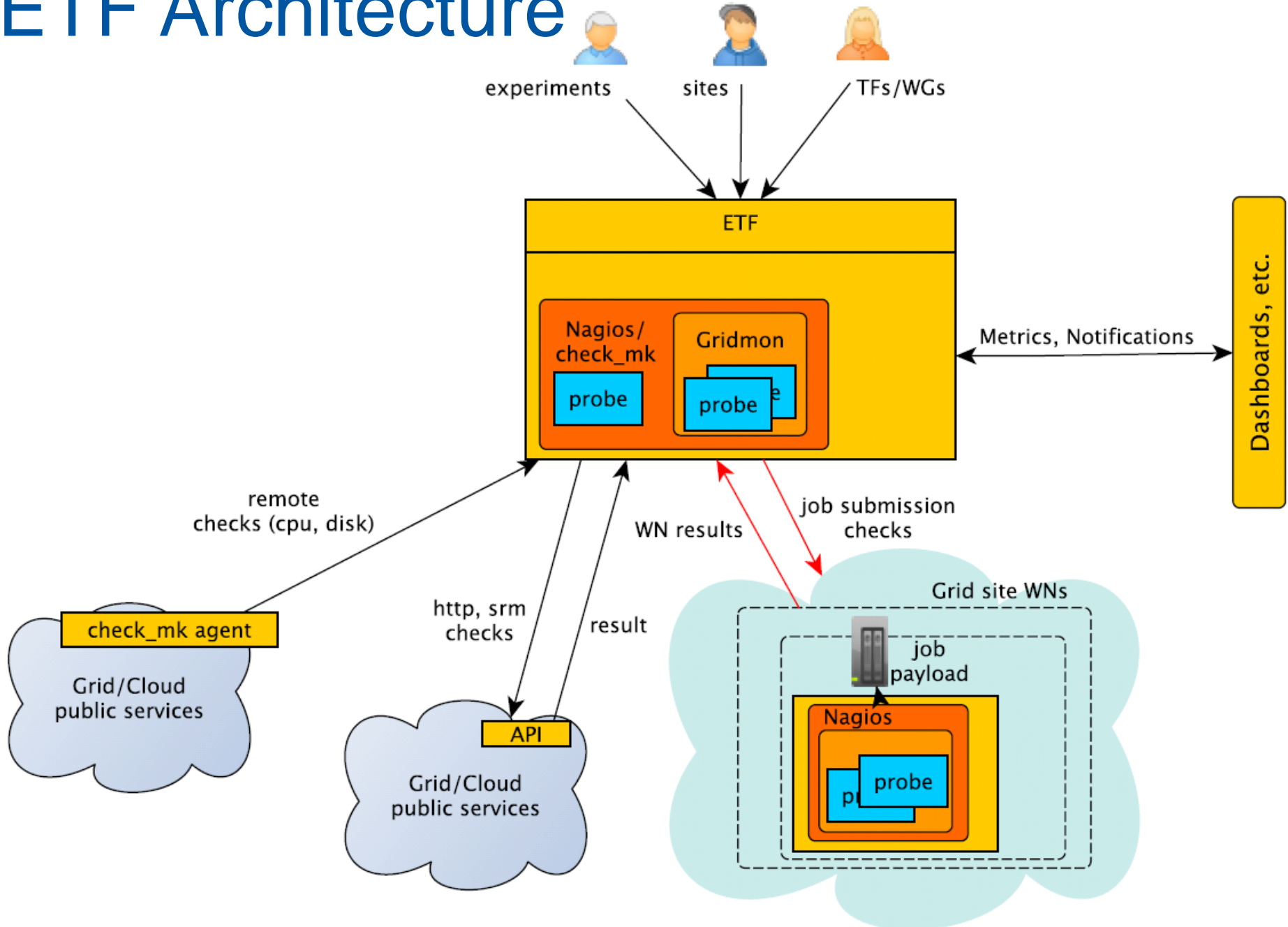
Worker node framework

- micro-framework to run tests on the worker nodes

Numbers

- 7 clusters: 4 clusters, one for each experiment (HA for production), 1 cluster for Data Centre, 1 cluster for perfSONAR (hosted by OSG), 1 cluster for IPv4/IPv6 WG

ETF Architecture



ETF Plans

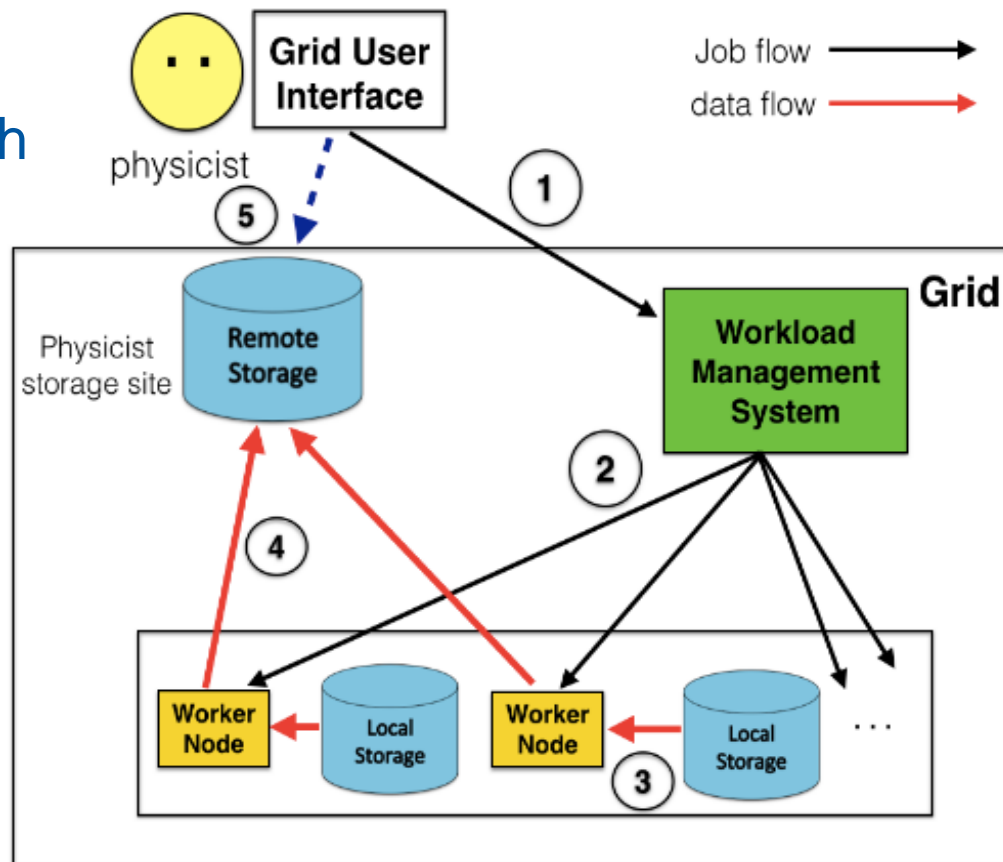
- Evolve ETF to MaaS platform
 - provide it as an on-demand service aimed for core infrastructure monitoring (remote probing)
- Integration with Agile Monitoring
 - SNOW notifications, publishing to Elasticsearch
- Refactoring of plugins
 - consolidate existing code-base
- Refactoring of worker node framework
 - investigate if we could migrate to collectd/ganglia
- Prepare for upcoming CentOS 7 and IPv6

Asynchronous Stage-Out (ASO)

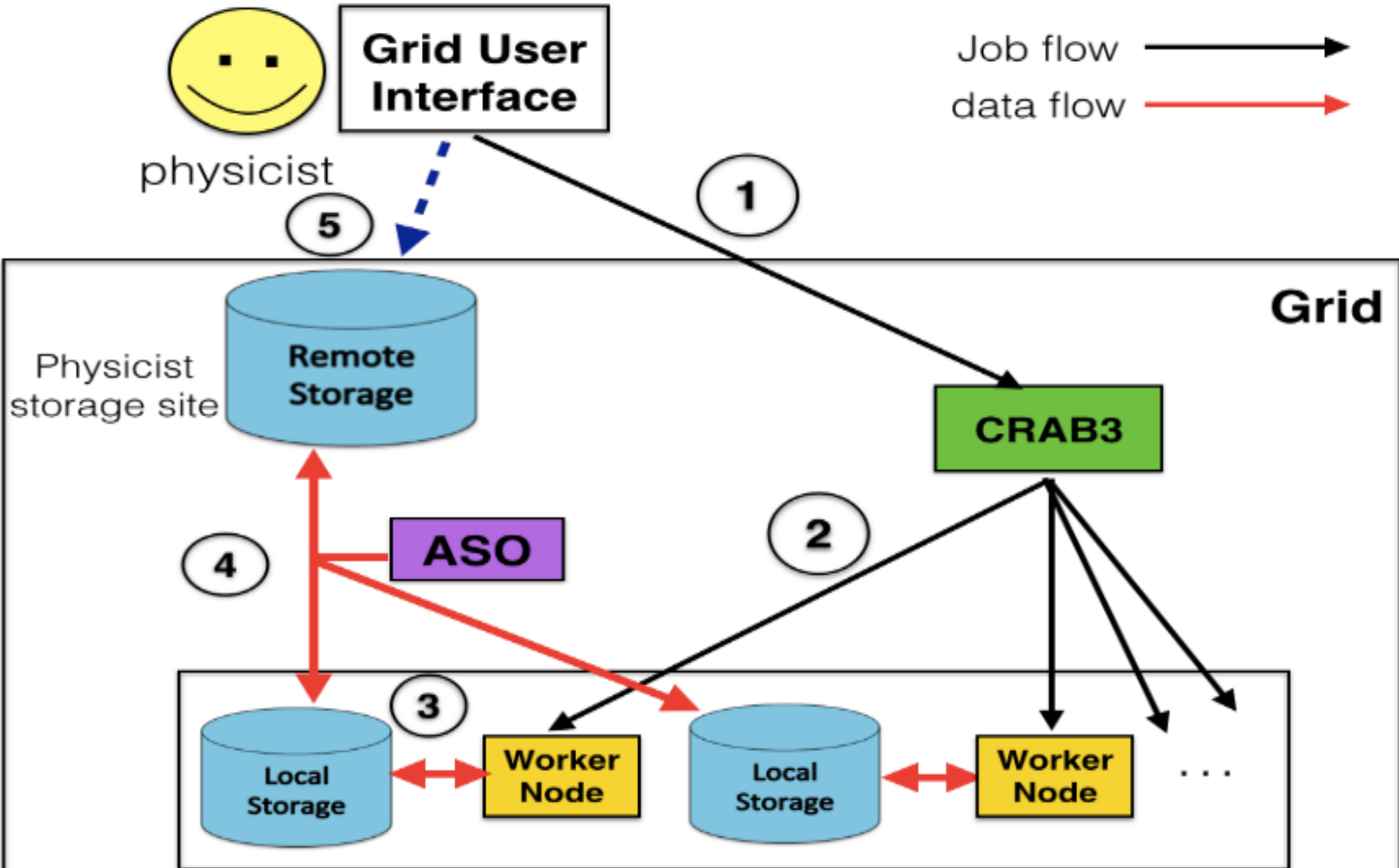
Distributed Data Analysis in CMS

- +1000 individual user/month
- +50 sites
- +20k concurrent jobs
- Typically 1 output/job
 - Files vary in size
- +200k completed jobs/day
- Chaotic environment

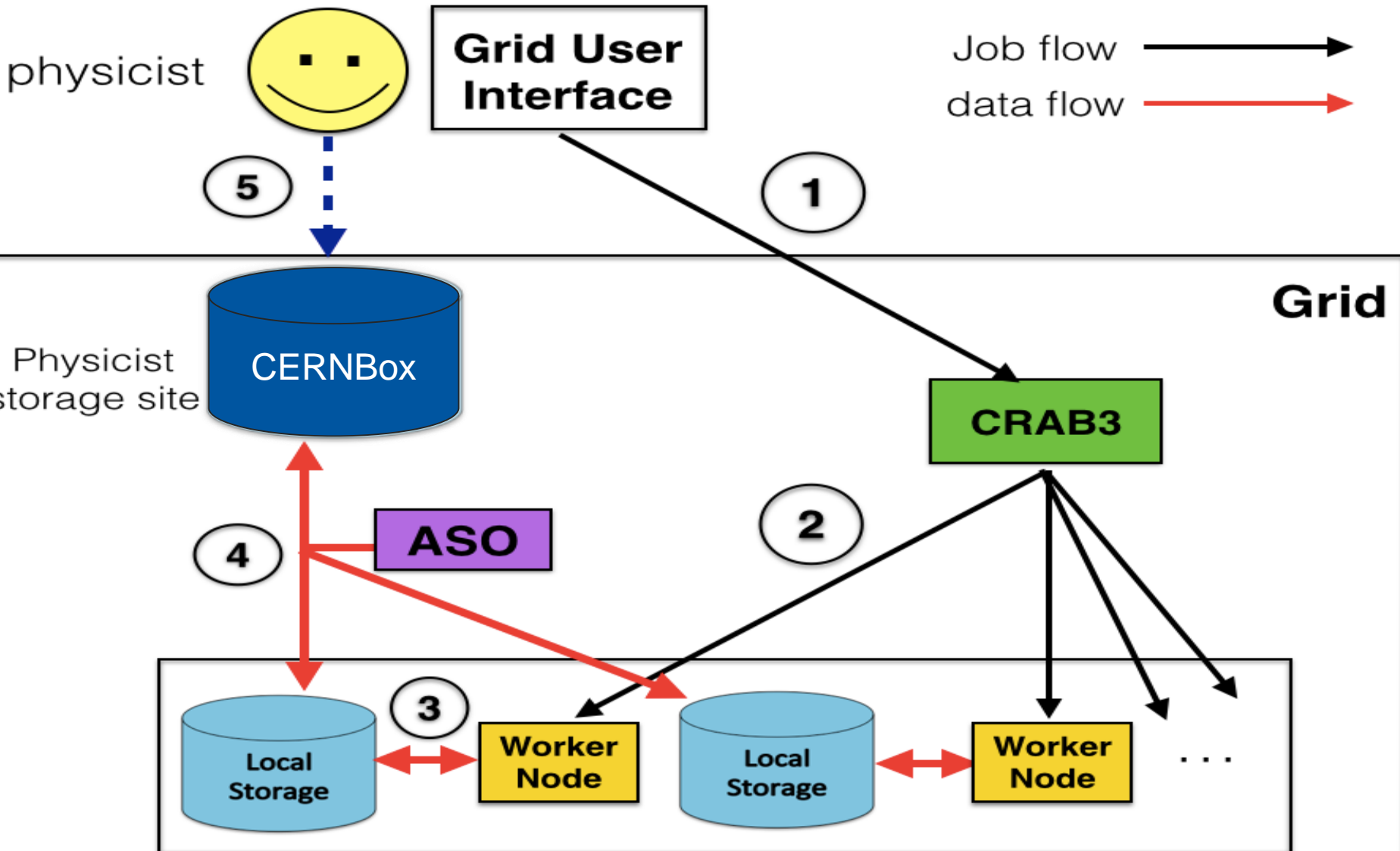
THE PROBLEM
15% CPU lost



Asynchronous stage-out for CMS



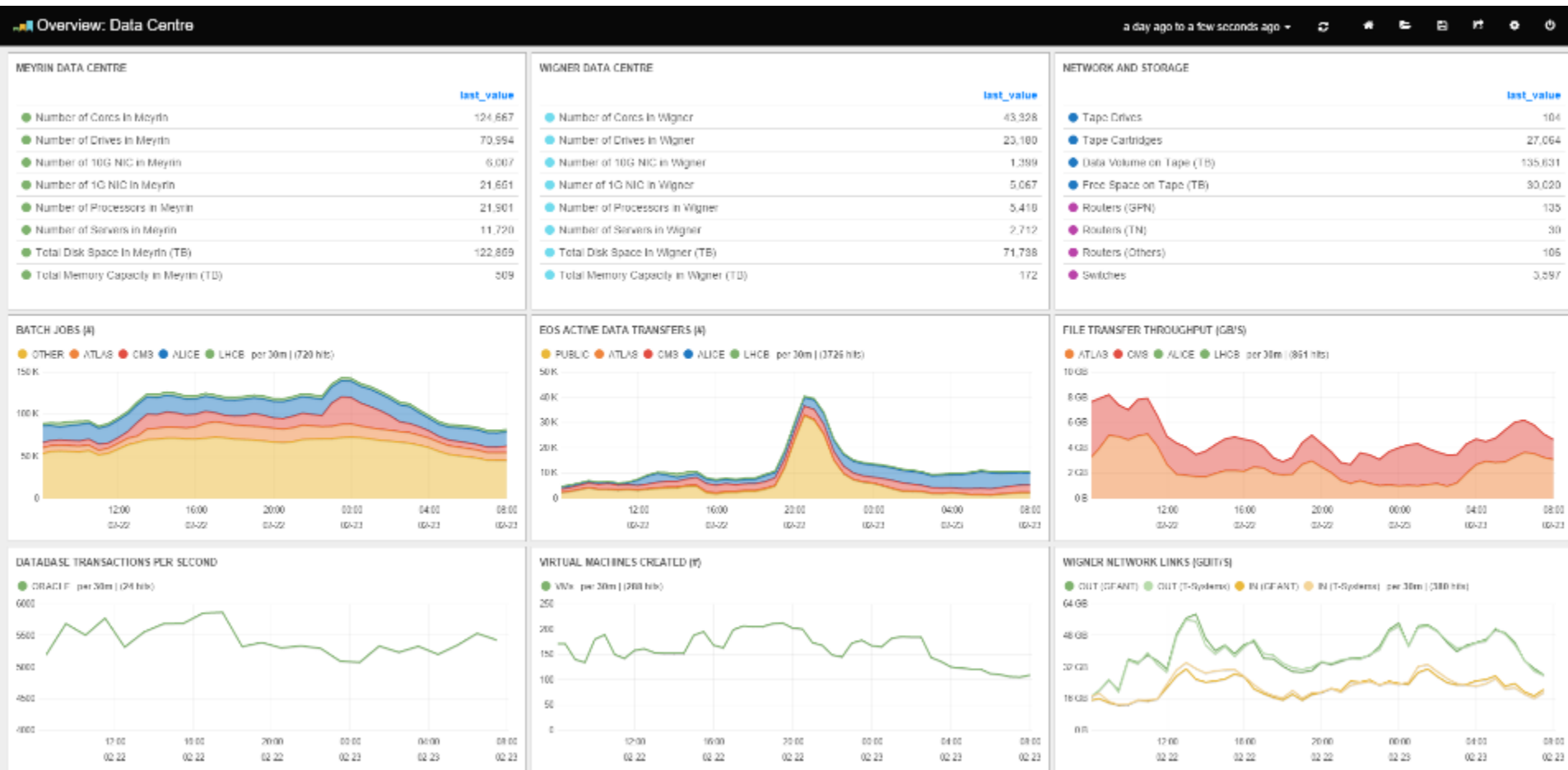
Integration of Grid and Cloud Resources



Monitoring

- Provide monitoring solutions for CERN and the **WLCG**
 - Hardware, operating system, and services
 - Other equipment (PDUs, temperature sensors, etc.)
 - **Sites availability, data transfers, job information, reports**
- Production operation of new tools and services

Data Centre Monitoring (Meter)



Streaming (Foz, Spark)

jupyter Untitled5 Last Checkpoint: 11/05/2015 (autosaved)

Control Panel Logout

File Edit View Insert Cell Kernel Help

Kernel starting, please wait...

Python 2 with Spark

Code Cell Toolbar: None

```
In [1]: from pyspark import SparkContext, SparkConf
        from pyspark.streaming import StreamingContext
        from pyspark.streaming.kafka import KafkaUtils
        import json

        def parse_everything(raw_metric):
            parsed_metric = json.loads(raw_metric)
            parsed_metric['body'] = json.loads(parsed_metric['body'])
            return parsed_metric

        zkQuorum = 'itmon-zookeeper-dev.cern.ch:2181/kafka/river/dev'
        topic = 'default-flume-topic'
        consumer_name = 'pedro3_test_consumer'
        binSize = 30 # seconds
        ssc = StreamingContext(sc, binSize)
        kafkaStream = KafkaUtils.createStream(ssc, zkQuorum, consumer_name, {topic: 1})
```

```
In [2]: def parse_everything(raw_metric):
        # We parse the metric as a JSON
        parsed_metric = json.loads(raw_metric)

        # The body is a `string`, so it needs to be parsed again as a JSON
        parsed_metric['body'] = json.loads(parsed_metric['body'])

        # Python dictionary
        return parsed_metric

        metrics_in_bucket = kafkaStream.map(lambda x: x[1]).map(parse_everything)

        result = metrics_in_bucket \
            .filter(lambda x: x['headers']['toplevel_hostgroup'] == 'aimon') \
            .filter(lambda x: x['headers']['metric_id'] == '9011') \
            .count()
        result.pprint()
        ssc.start()
```

Time: 2015-11-05 14:38:30

Time: 2015-11-05 14:39:00

Time: 2015-11-05 14:39:30

27

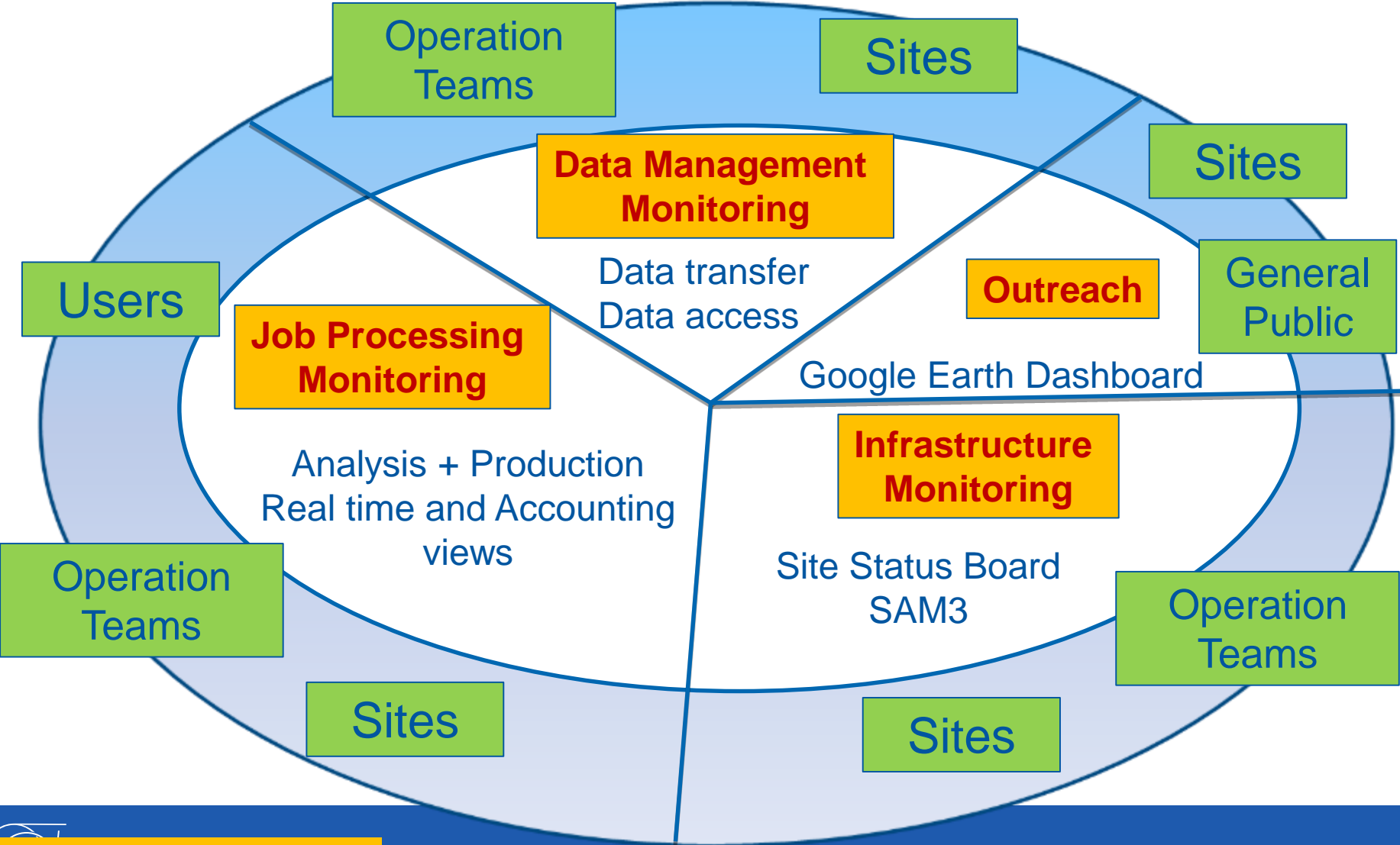


Current IT AI Monitoring

Monitoring	Metric Manager
	Lemon Agent
Transport	Gateway (Flume)
	XSLS (Flume)
	Messaging (AMQ)
	River (Kafka)

Archive	HDFS
Displays	Meter (ES)
	Timber (ES)
	Meter Proxy
Streaming	Foz (Spark)
Alerts	GNI

Experiment Dashboards



300-500 users per day

WLCG Job Monitoring

More than 170 computing centres

- 250K parallel running jobs all the time

3 million jobs per day for ATLAS and CMS

- Production (simulation) jobs: Well-organised activity performed by a group of experts
- Analysis jobs: Fully chaotic activity conducted by diverse members of the physics community

ATLAS

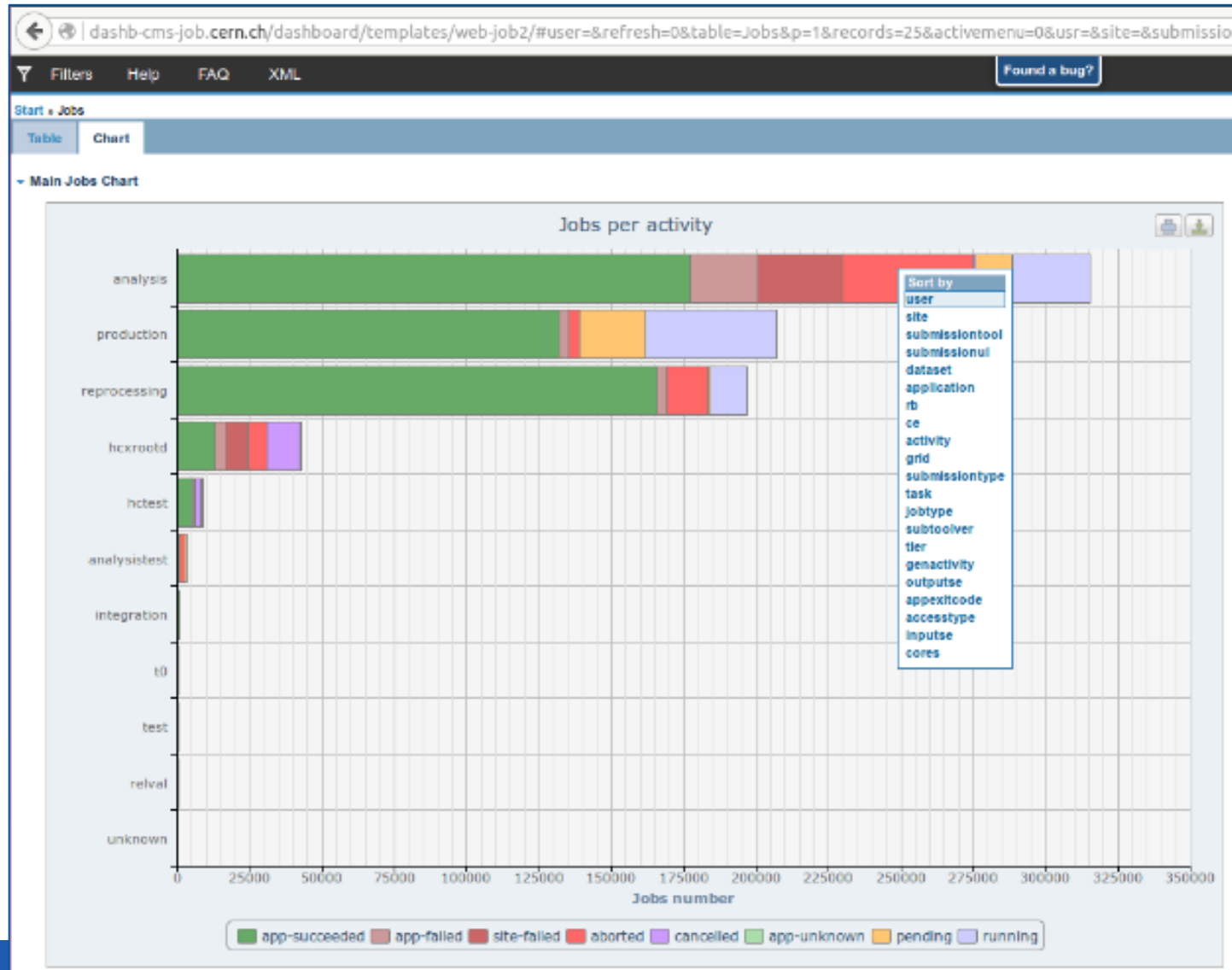
- ~11 to 15M rows per day for job status and updates
- ~200K for the daily summarised statistics (accounting)

CMS

- ~ 6M rows per day for job status updates
- ~100K for the daily summarised statistics (accounting)

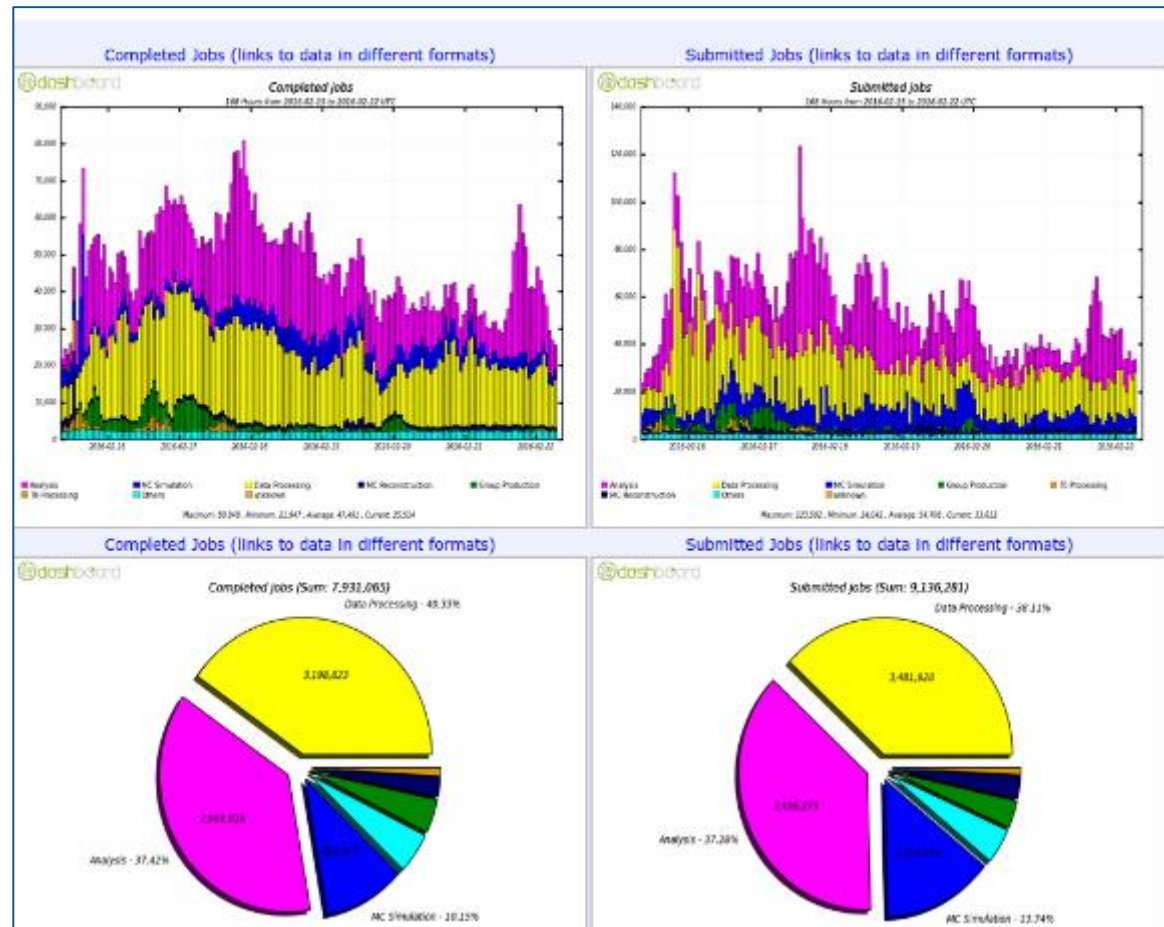
Real-Time: Jobs Interactive View

- Shared between ATLAS and CMS
- Targets site admins and operation teams
- Many filters



Accounting: Jobs Historical Views

- Many different filtering and grouping by options
- Many different plotting categories
- Utilisation over pledges
- Pledged / unpledged resources
- Usage per Resource Type (grid, cloud, HPC, local), per CPU Core, per Project, per Group, per Cloud, ..



Accounting: CMS Condor Monitoring



DAILY JOB SUMMARY

DEFINE PARAMETERS

Select Sites or Servers ▾ Select Job Types Select Cores Last Week ▾ ? Hourly ▾ All ▾ ? Group by Sites ▾ **Info**

PLOTTING CATEGORY
 Condor Pending
 Running Jobs

CHOSEN PARAMETERS

Sites:
All T1s + T0

Job Types:
all

Cores:
all

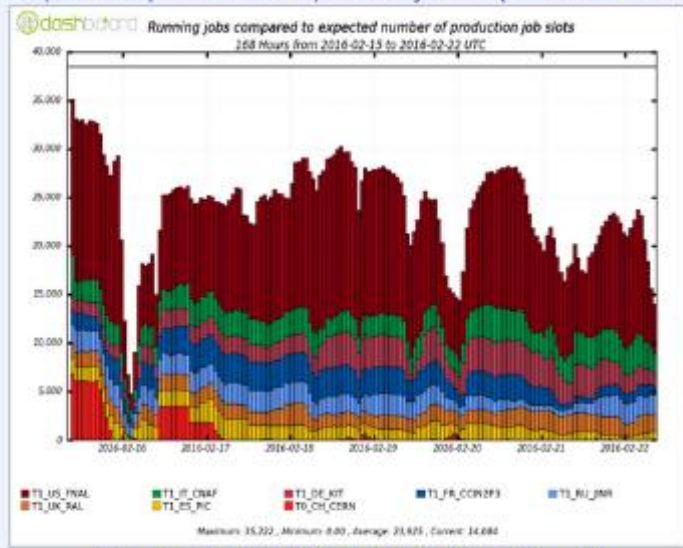
Time Range:
Last Week

Granularity:
Hourly

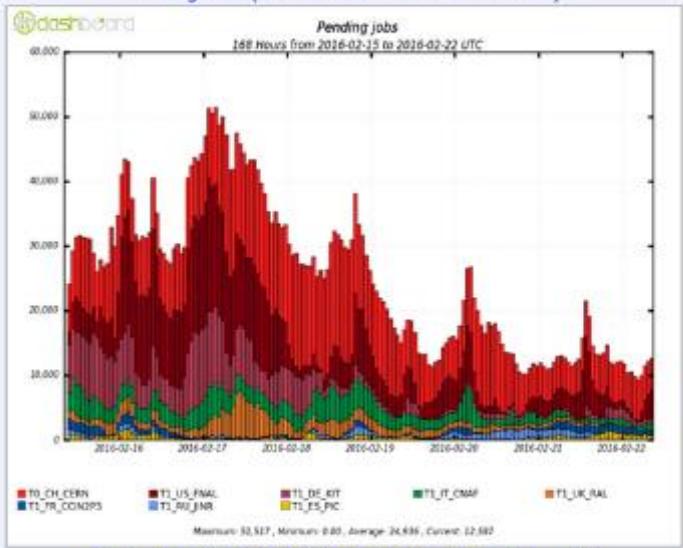
Series:
All

Group by:
Sites

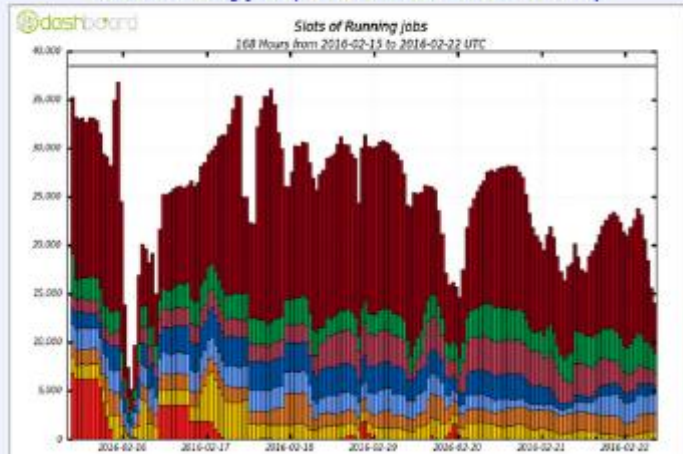
Running jobs compared to expected number of production job slots (links to data in different formats)



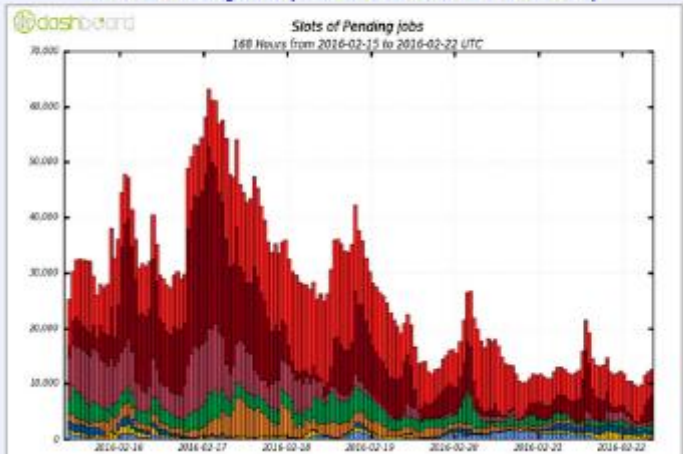
Pending Jobs (links to data in different formats)



Slots of Running jobs (links to data in different formats)



Slots of Pending Jobs (links to data in different formats)



Report a bug or a suggestion

Accounting: ATLAS DM Accounting

DEFINE PARAMETERS

Select Sites
Select Countries
Select DDM Endpoints
Select Projects
Select Datatypes
Custom
? Weekly
? All
? Group by: Grouped Datatype

DEFINE ADVANCED PARAMETERS

Select Streams
Select Replica Groups
Select Dataset Groups
Hidden Flag
Select Provenance
Select Campaign
Select Replica Class
Select ProdStep

PLOTTING CATEGORY

CHOSEN PARAMETERS

Sites: All Sites

Category of Sites: All Countries

DDM Endpoints: all

Projects: all

Datatypes: all

Time Range: From: 2015-01-01 To: 2015-12-31

Granularity: Weekly

Series: All

Streamnames: all

Replica Groups: all

Dataset Groups:

Physical Definitions

No. of Dataset Replicas (links to data in different formats)

No. of Physical Files (links to data in different formats)

No. of Physical Bytes in TBs (links to data in different formats)

No. of Physical Bytes in TBs over Pledges (links to data in different formats)

Rucio, SRM usage (links to data in different formats)

■ aod	■ cbnt	■ d2aod	■ d2aodm	■ d2exrd
■ daod	■ desd	■ desdm	■ dntup	■ dpd
■ draw	■ esd	■ evnt	■ expired	■ group
■ hist	■ hits	■ log	■ munt	■ ntup
■ raw	■ rdo	■ tag	■ txt	■ user
■ 109ff22-8235-49e5-81fc-c86fbab8c88d	■ ci			
■ secondary	■ USED	■ TOTAL		

Average number of Replicas per Replica Class (links to data in different formats)

Average number of Replicas per Datatype (links to data in different formats)

WLCG Data Transfer Monitoring

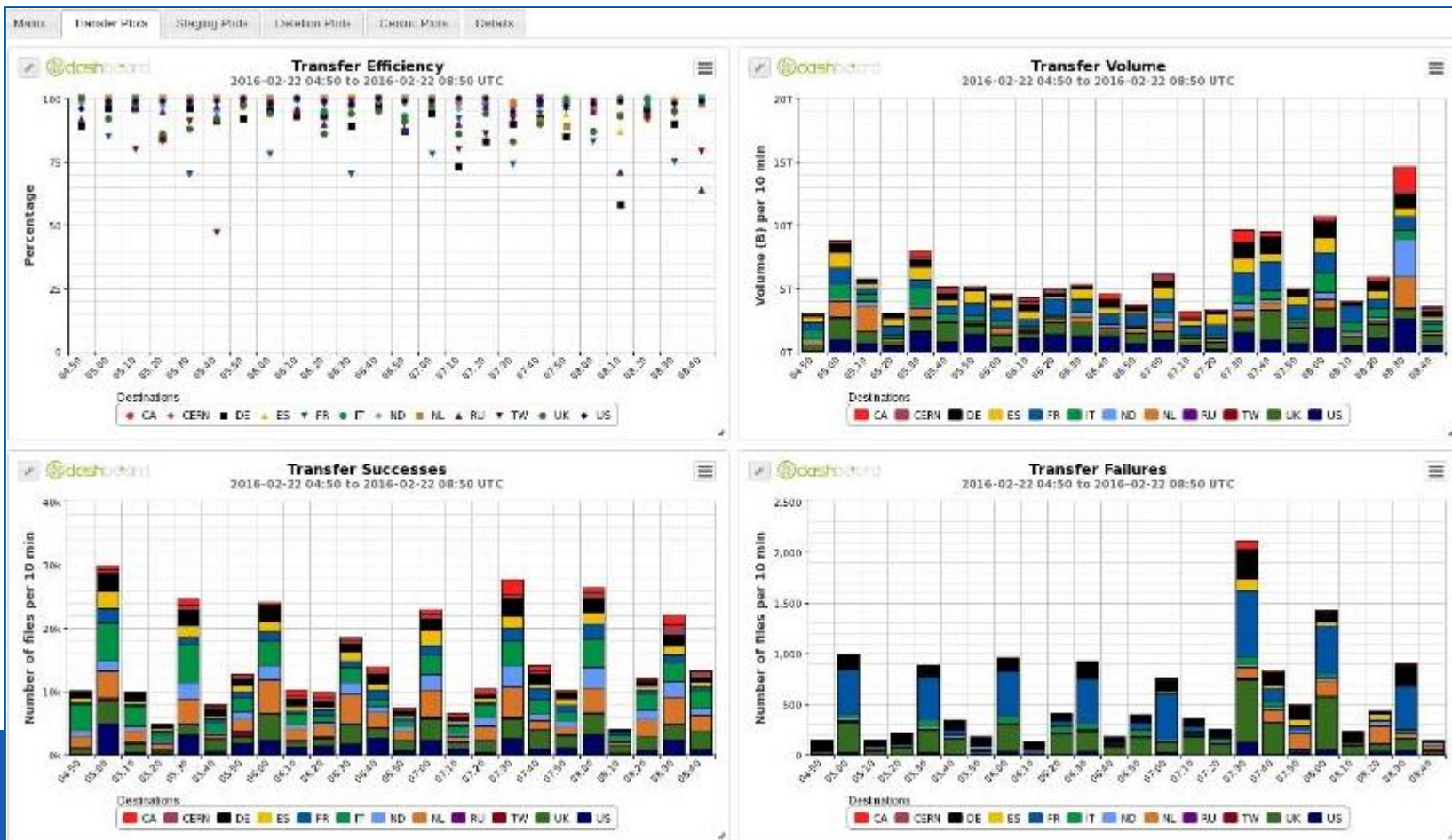
- Collect, archive, process and display data for transfers and access

ATLAS
DDM

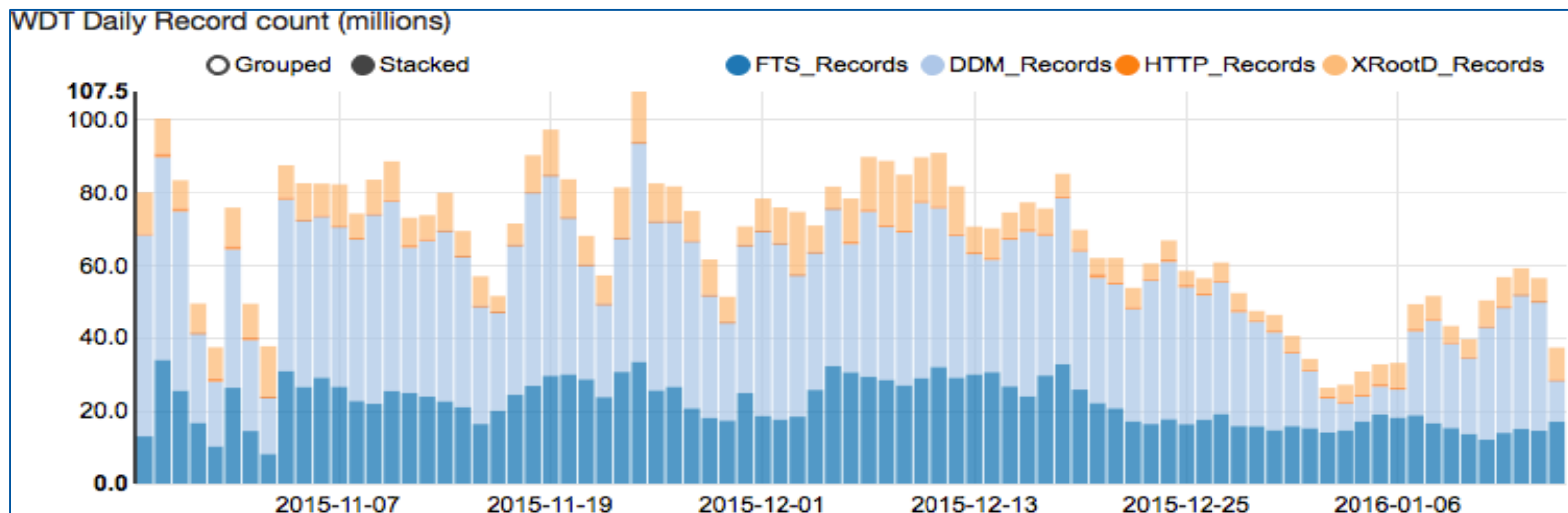
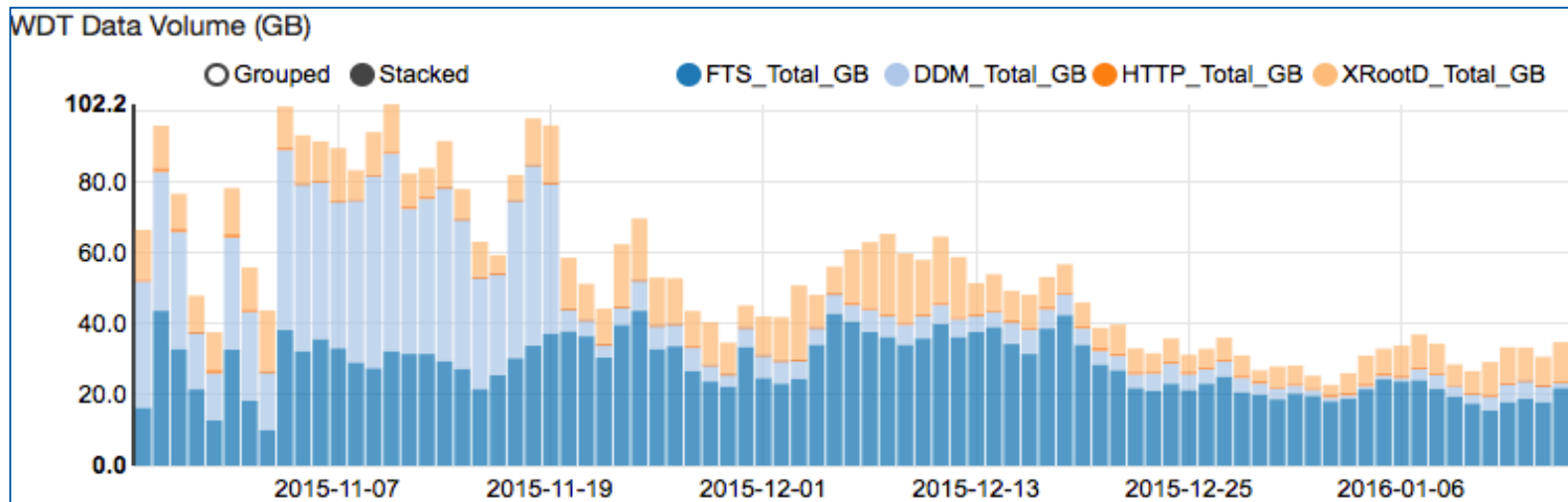
FTS

XRootD

HTTP



Challenges – Volume & Rate



WLCG Infrastructure Monitoring

- SSB (Site Status Board)
 - Collect metrics for entities
 - Numerical, string, status
 - Metrics can be combined to create new metrics:
 - AND, OR, ANY, ALL, FILTER, OVERWRITE
 - 5 instances: CMS, ATLAS, LHCb, ALICE, WLCG
- SAM3 (Service Availability Monitoring)
 - Dedicated UI on top of SSB
 - Monthly availability/reliability reports for WLCG office
 - Using same database structure

SSB Summary page



Index Expanded Table

View: default



T0 +T1		T2		T2		T2	
Status	Site Name	Status	Site Name	Status	Site Name	Status	Site Name
	T0_CH_CERN		T2_AT_Vienna		T2_FR_GRIF_IRFU		T2_RU_JINR
	T1_DE_KIT		T2_BE_IHHE		T2_FR_GRIF_LLR		T2_TH_CUNSTDA
	T1_ES_PIC		T2_BE_UCL		T2_FR_IPHC		T2_UA_KIPT
	T1_FR_CCIN2P3		T2_BR_SPRACE		T2_GR_Ioannina		T2_UK_London_Brunel
	T1_IT_CNAF		T2_BR_UERJ		T2_HU_Budapest		T2_UK_London_IC
	T1_RU_JINR		T2_CH_CERN		T2_IN_TIFR		T2_UK_SGrid_Bristol
	T1_UK_RAL		T2_CH_CERN_AI		T2_IT_Bari		T2_UK_SGrid_RALPP
	T1_US_FNAL		T2_CH_CERN_HLT		T2_IT_Legnaro		T2_US_Caltech
			T2_CH_CSCS		T2_IT_Pisa		T2_US_Florida
			T2_CN_Beijing		T2_IT_Rome		T2_US_MIT
			T2_DE_DESY		T2_KR_KNU		T2_US_Nebraska

Found a bug?

SSB expanded table

Site Name	Waiting Room	Waiting Room - manual changes	Morgue	Prod status	Prod status - manual changes	Usable_Analysis	SiteReadiness ranking	SiteReadiness Status	Site SAM availability
T1_US_FNAL_DISK									
T2_AT_Vienna	out		out	on	on	usable	100	n/a	100
T2_BE_IIHE	out		out	on	on	not_usable	83	n/a	0
T2_BE_UCL	out		out	on	on	usable	100	n/a	100
T2_BR_SPRACE	in		out	drain	on	usable	53	n/a	18
T2_BR_UBERJ	out	out	out	drain	drain	usable	100	n/a	0
T2_CH_CERN	out	out	out	on	on	usable	100	n/a	100
T2_CH_CERN_AI	out	out	out	tier0	tier0	not_usable			
T2_CH_CERN_HLT	out	out	out	on	on	not_usable			100
T2_CH_CERN_Wigner									
T2_CH_CSCS	out	out	out	drain	drain	usable	100	n/a	100
T2_CN_Beijing	out	out	out	on	on	usable	100	n/a	100
T2_DE_DESY	out	out	out	on	on	usable	100	n/a	100
T2_DE_RWTH	out	out	out	on	on	usable	80	n/a	8 0102050505050505
T2_EE_Estonia	out	out	out	on	on	usable	100	n/a	100
T2_ES_CIEMAT	out	out	out	on	on	usable	100	n/a	100
T2_ES_IFCA	out	out	out	on	on	usable	100	n/a	95
T2_FI_HIP	out	out	out	drain	on	usable	71	n/a	64
T2_FR_CCIN2P3	out	out	out	on	on	usable	100	n/a	100
T2_FR_GRIF_IRFU	out	out	out	on	on	usable	100	n/a	100

SAM3

ATLAS dashboard

SAM VISUALIZATION | ATLAS

Latest Results Historical View Generate Report Help

View: Site Availability Time range: Last 12 Hours Site Groups: Tier1s Profiles: ATLAS_CRITICAL

Plot: Quality Plot Granularity: Default Sites: All Sites (BNL-ATLAS, FZK-LCG2, IN2P3-CC)

Show values before corrections

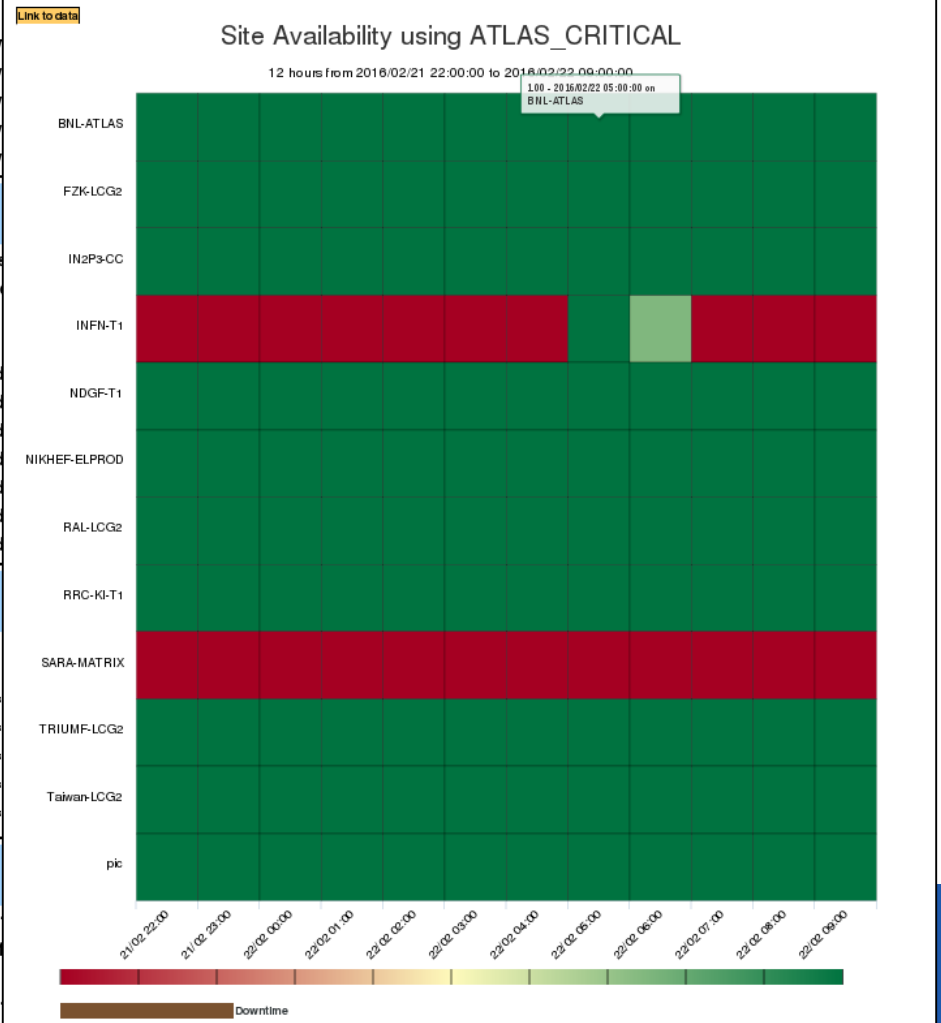
Algorithm: (OSG-CE + CREAM-CE + ARC-CE) * (all SRMv2 + all OSG-SRMv2)

Sitename	Flavour	Hosts
BNL-ATLAS	OSG-SRMv2	OK
	OSG-CE	OK
		OK
		OK
		UNKNOWN

Sitename	Flavour	Host status in profile	Hosts
FZK-LCG2	SRMv2	OK	atlassrm-fzk.gridka.de
		OK	dgridsrm-fzk.gridka.de
		MISSING	ppssrm-kit.gridka.de
	CREAM-CE	OK	cream-ge-2-kit.gridka.de
		OK	cream-ge-3-kit.gridka.de
		OK	cream-ge-4-kit.gridka.de
		OK	cream-ge-5-kit.gridka.de
		OK	cream-ge-6-kit.gridka.de
		OK	cream-ge-7-kit.gridka.de
		OK	cream-ge-8-kit.gridka.de

Sitename	Flavour	Host status in profile	Hosts
IN2P3-CC	SRMv2	OK	ccsr.in2p3.fr
	CREAM-CE	OK	cccreamceli01.in2p3.fr
		OK	cccreamceli02.in2p3.fr
		OK	cccreamceli04.in2p3.fr
		OK	cccreamceli05.in2p3.fr
		OK	cccreamceli06.in2p3.fr

Sitename	Flavour	Host status in profile	Hosts
INFN-T1	SRMv2	OK	storm-fe.cr.cnaf.infn.it
		MISSING	vm-storage-06.cr.cnaf.infn.it
	CREAM-CE	CRITICAL	ce01-lcg.cr.cnaf.infn.it



WLCG on Google Earth

Feb 15, 2016 3:41:00 pm

Running jobs: 279977
Active CPU cores: 411389
Transfer rate: 11.65 GiB/sec



dashboard

Imagery Dates: Apr 10, 2013 - Dec 14, 2015

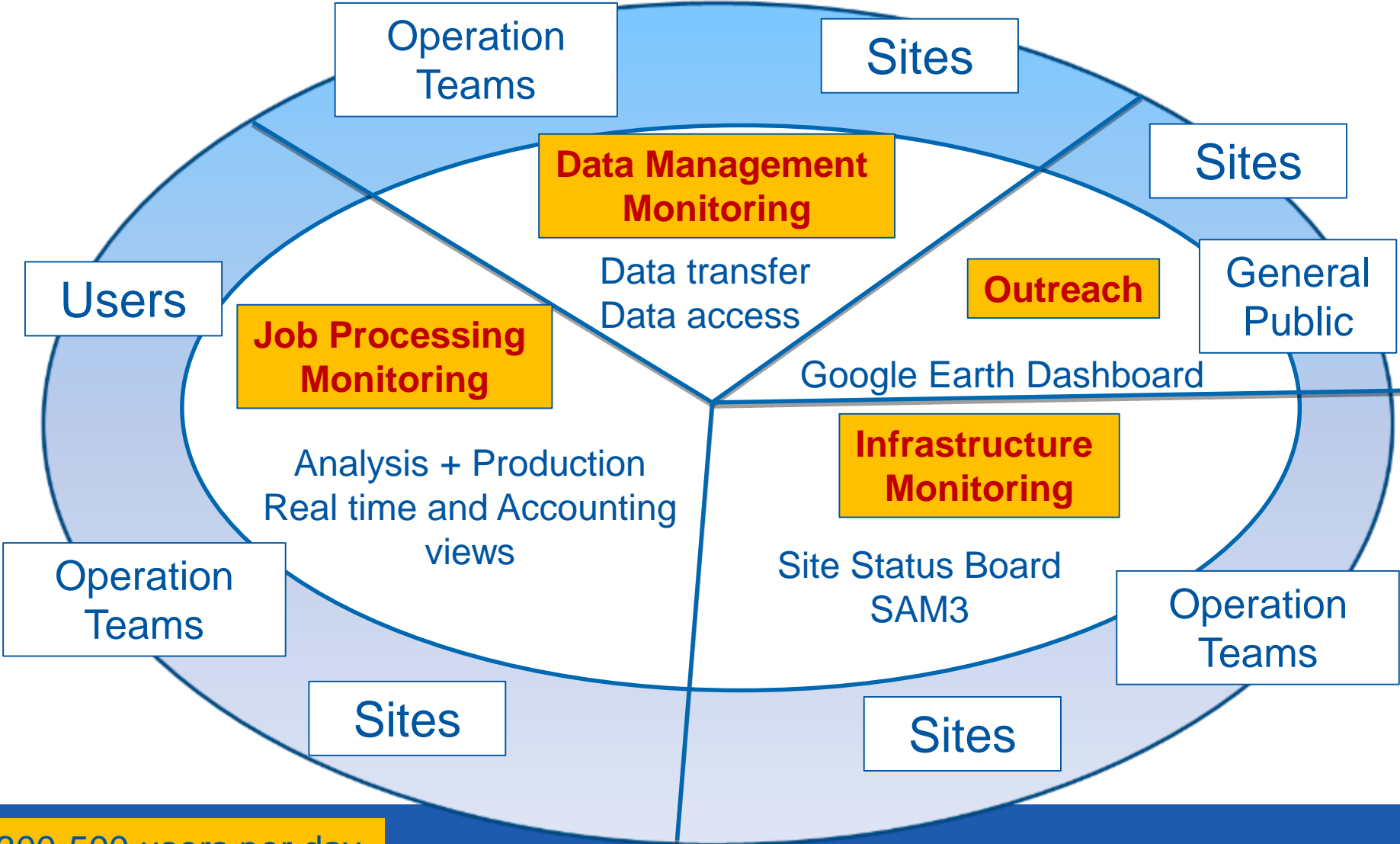
Data SIO, NOAA, U.S. Navy, NGA, GEBCO
Image Landsat

42°20'51.05" N 0°14'51.72" E elev: 322 ft

©2010 Google

Eye alt: 8985.83 m

Experiment Dashboards

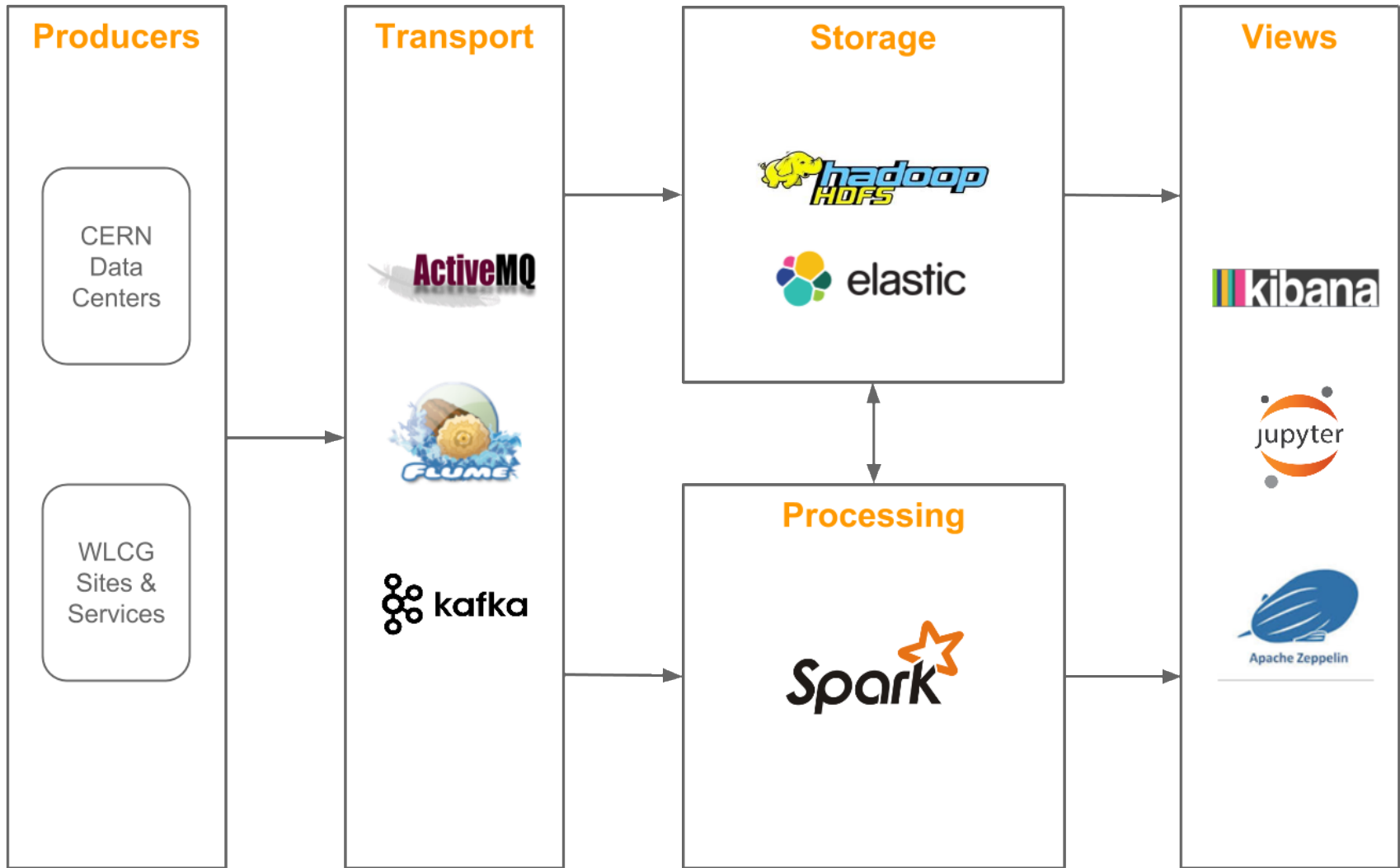


300-500 users per day

Plans

- Focus on migrating to a common architecture
- Review the existing architecture and areas
- Look into other technologies as needed (Grafana, collectd, Kafka, etc.)
- Use the feedback received from the users (Experiments and IT groups)
- Move to new ES central service
- Continue with standard operations and upgrades

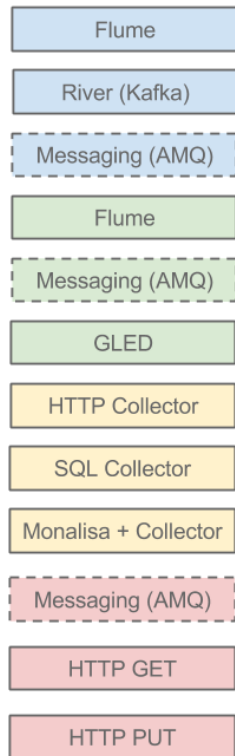
Common Architecture and Areas



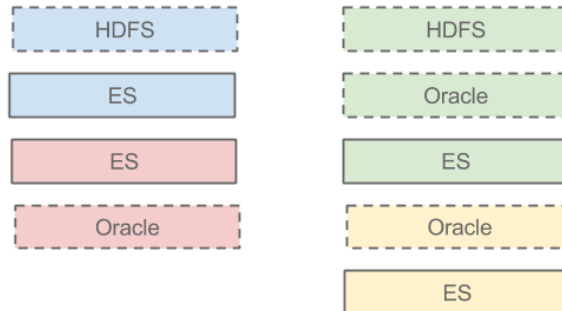
Producers



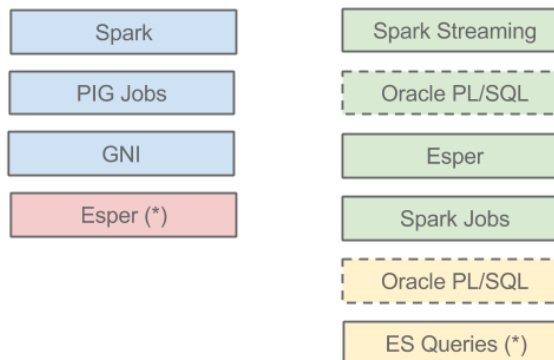
Transport



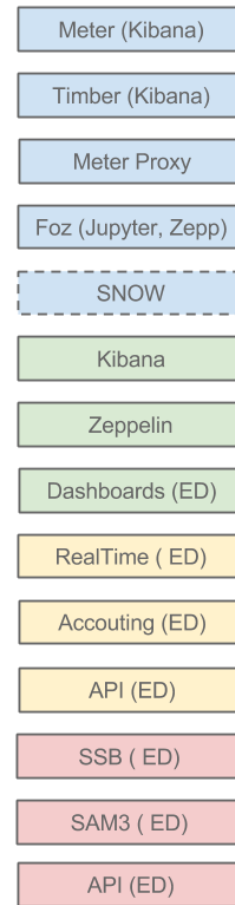
Storage and Archive



Processing (Batch and Streaming)



Serving



Monitoring Scrum

SPRINT: Keep Calm and Merge ! ▾

 QUICK FILTERS: Only My Issues Recently Updated

Open	In Progress	In Review	Closed
<p>MONIT-8 ↓ Add ES output options to DDM Spark job DDM processing on Spark</p>	<p>MONIT-4 ↓ Implement the algorithm to aggregate raw data into ddm statistics in spark from HDFS DDM processing on Spark</p>	<p>MONIT-31 ↑ Define the endpoint, index names for ES Store AGIS/DDM/Job Monit...</p>	<p>MONIT-67 ↓ Document the current WDT job deployment and scheduler solution Improve Spark jobs deploy...</p>
<p>MONIT-22 ↓ Investigate FTS views in Kibana Prototype early Dashboards</p>	<p>MONIT-11 ↓ Investigate how to read AGIS metadata using flume Transport initial data produc...</p>	<p>MONIT-58 ↓ Investigate and define template for AGIS data in ES Store AGIS/DDM/Job Monit...</p>	<p>MONIT-47 ↑ Document the usage of APIs from internal Dashboard apps Investigate the usage of AP...</p>
<p>MONIT-52 ↓ Investigate current Job Monitoring views Prototype early Dashboards</p>	<p>MONIT-42 ↓ Deploy flume setup for es sink Setup initial new infrastructure</p>		<p>MONIT-49 ↑ Document the usage of APIs from external users and third party systems Investigate the usage of AP...</p>
<p>MONIT-54 ↓ Investigate Job Monitoring views in Kibana Prototype early Dashboards</p>	<p>MONIT-21 ↓ Deploy zeppelin Setup initial new infrastructure</p>		<p>MONIT-53 ↓ Investigate current FTS Transfers views Prototype early Dashboards</p>
<p>MONIT-77 ↓ Investigate and define template for Panda Job Monitoring raw data in ES Store AGIS/DDM/Job Monit...</p>	<p>MONIT-17 ↓ Read FTS data from AMQ using flume Transport initial data produc...</p>		<p>MONIT-62 ↑ Guidelines for hostgroup management and structure Define DevOps guidelines</p>
<p>MONIT-27 ↓ Investigate and define template for Job Monitoring aggregated data in ES Store AGIS/DDM/Job Monit...</p>	<p>MONIT-14 ↓ Write FTS raw data to ES and HDFS using flume Transport initial data produc...</p>		<p>MONIT-66 ↓ Guidelines for VMs, accounts, and documentation repos Define DevOps guidelines</p>
<p>MONIT-30 ↓ Create guidelines for storing data in ES</p>	<p>MONIT-59 ↓ Investigate and define template for FTS raw data in ES Store AGIS/DDM/Job Monit...</p>		<p>MONIT-63 ↓ Define common mechanism to add a "producer" key/identifier</p>

Metrics Improvements

Metrics producer

- Lemon-agent is the remaining tool of old toolset
- Host monitoring
 - Study new tools (e.g. collected, statsd)
 - Start by looking at OS sensors/metrics
- Cgroups/containers monitoring
 - Deploy initial prototype using cAdvisor+collectd

Metrics Management

- Enhance metrics registration and storage
- Registration for host and service metrics
 - Host namespace not enforced, no namespace for services
 - Introduce common namespace for hosts and metrics

Test distributed time-series databases

- New dedicated solutions for metrics indexing/storage
- Some examples: InfluxDB, OpenStack Gnocchi

Other Improvements

Stream Processing

- Get feedback from service early adopters
- Anomaly detection
- Actionable aggregation of multi-node alarms

Security

- Integrate security data into monitoring workflow/tools
- Several meetings and prototypes completed
- Security team member joined Monitoring sprints
 - Integrate transport and archive of security data
 - Dedicated streaming workflow for security data

Summary

- Merge IT and Grid monitoring
- Converge in ES for other monitoring data (HC, ETF)

- Continue (and review) existing services
- Increase transparency and work across services, projects, activities

- Collaborate with services in the group and in IT
- Be more integrated with IT practices
 - services, communication, rota, tools

<https://twiki.cern.ch/twiki/bin/view/CMgroup/CmMmSection>

Thank you

Alberto Aimar

Borja Garrido Bear

Daniel Zolnai

Edward (**Eddie**) Karavakis

Hassen Riahi

Jaroslava (**Jarka**) Schovancova

Javier Rodriguez Martinez

Lionel Cons

Luca Magnoni

Marian Babik

Maria-Varvara (**Mary**) Georgiou

Pablo Saiz

Pedro Rodrigues De Sousa Andrade

Rocio Rama Ballesteros

Sergey Belov

Valentina Mancinelli

