Notes from HEPiX Spring 2017, Hungarian Academy of Sciences

Jérôme Belleman, CERN

24-28 April 2017

Hosted for the first time at the Hungarian Academy of Sciences, HEPiX Spring 2017 attracted an astonishing 125 participants from 48 institutes across 23 countries worldwide. There were some newcomers, as well as some participants how hadn't attended HEPiX for a long time. The meeting took place at the Budapest seat of the Academy of Sciences, a neo-renaissance palace built in 1865 on the bank of the Danube. The seminar room was prestigious, large and comfortable, offering a stunning view of the river. The local organisers had set up a reliable audio/video system with live streaming, which ran seamlessly throughout the week. They had planned an abundant number of power sockets, conveniently located across the room.

As always, there was enough time dedicated to coffee breaks, for valuable face-to-face discussions. They were held in a large corridor, just outside the meeting room. Lunch breaks took place in the wine cellar of the same building, as did the Monday evening reception. The workshop dinner was organised on a boat, as part of a 3 hours cruise on the Danube, showing even more breathtaking views of the rich architecture that is prevalent in Budapest.

 DOI: 10.17181/CERN.NZLN.L4HW

Contents

Monday 24 April 2017 . . . 4
Welcome (Balázs Bagó) . . . 4
Site Reports . . . 4
CERN Site Report (Jérôme Belleman) . . . 4
INFN-T1 Site report (Stefano Bovina) . . . 4
DESY Site Report (Andreas Haupt) . . . 5
LAL/GRIF Site Report (Michel Jouvin) . . . 5
RAL Site Report (Martin Bly) . . . 6
Swiss National Supercomputing Centre T2 Site report (Dino Conciatore) . . . 6
T2_FI_HIP Site Report (Johan Guldmyr) . . . 7
Site Report for Max Delbrück Center for Molecular Medicine (Alf Wachsmann) . . . 7
NDGF Site Report (Mattias Wadenstein) . . . 8
BNL RACF Site Report (Ofer Rind) . . . 8
AGLT2 Site Report Spring 2017 (Shawn McKee) . . . 9
Nebraska Site Report (Brian Bockelman) . . . 9
End-User IT Services & Operating Systems . . . 10
CERN Linux services status update (Ulrich Schwickerath) . . . 10
SW & Computing for Big Science Journal (Michel Jouvin) . . . 10
Security & Networking . . . 10
IPv6 at the RAL Tier 1 (James Adams) . . . 10
Basic IT Services . . . 11
A Hard Puppexit from 3 to 4 (Jérôme Belleman) . . . 11
salt stack - Using, Extending, and programming (Owen Synge) . . . 12
Site Reports . . . 13
CEA Site Report (Arnab Sinha) . . . 13
Tuesday 25 April 2017 . . . 14
Site Reports . . . 14
Experience on the operations at new KEKCC (Tomoaki Nakamura) . . . 14
Tokyo Tier-2 Site Report (Tomoe Kishimoto) . . . 14
KR-KISTI-GSDC-01 Tier-1 Site Reports (Jeongheon Kim) . . . 14
Status of IHEP site (Jingyan Shi) . . . 15
Security & Networking . . . 15
Computer Security Update (Liviu Vâlsan) . . . 15
Security and networking: Security workshop (Liviu Vâlsan and Romain Wartel) . . . 16
WLCG/OSG Networking Update (Shawn McKee) . . . 17
ESnet Update (Joe Metzger) . . . 17
Network related updates in IHEP (Shan Zeng) . . . 18
Deployment of IPv6-only CPU on WLCG – an update from the HEPiX IPv6 Working Group (Andrea Sciabà) . . . 19
KEK Computer security update (Tadashi Murakami) . . . 19
Building and operating a large scale Security Operations Center (Liviu Vâlsan) . . . 20
Wednesday 26 April 2017 . . . 22
Storage & Filesystems . . . 22
CERN IT-Storage Strategy Outlook (Luca Mascetti and Julien Leduc) . . . 22
EOS and CERNBox Update (Luca Mascetti) . . . 22
BNL Box (Hironori Ito) . . . 23
Federated data storage system prototype for LHC experiments and data intensive science (Andrey Kirianov) . . . 24
RAL Tier-1 Evolution as a Global CernVM-FS Service Provider (Catalin Condurache) . . . 25
An update to Ceph at RAL (Tom Byrne) . . . 25
Data-NG: A distributed Ceph infrastructure (Guillaume Philippon) . . . 26

Site Reports . . . 27
PIC Report (Jose Flix Molina) . . . 27
Computing & Batch Services . . . 27
CosmoHub on Hadoop: a web portal to analyze and distribute massive cosmological data (Jordi Casals Hernandez) . . . 27
HammerCloud extension for Data Centre commissioning (Jaroslava Schovancova) . . . 28
Experiences With Intel Knights Landing, OmniPath and Slurm (William Strecker-Kellogg) . . . 28
JLab's SciPhi-XVI Knights Landing Cluster Update (Sandy Philpott, remotely) . . . 29
Updates from HEPiX Benchmarking Working Group (Domenico Giordano) . . . 29
The scheduling strategy and experience of IHEP HTCondor Cluster (Jingyan Shi) . . . 30
The search for new traceability and isolation approaches (Brian Bockelman) . . . 30
Understanding the performance of benchmark applications (Luca Atzori) . . . 31
Thursday 27 April 2017 . . . 32
Grid, Cloud & Virtualisation . . . 32
The Computing Resource Information Catalog (Alessandro Di Girolamo) . . . 32
ElastiCluster – automated deployment and scaling of computing and storage clusters on IaaS cloud infrastructures (Riccardo Murri) . . . 32
CERN Cloud service update: Containers, migrations, upgrades, etc. (Luis Pigueiras) . . . 33
Container Orchestration – Simplifying Use of Public Clouds (Ian Collier) . . . 34
System testing service developments using Docker and Kubernetes: EOS + CTA use case (Julien Leduc) . . . 34
Distributed computing in IHEP (Xiaomei Zhang) . . . 35
Understanding performance: optimisation activities in WLCG (Andrea Sciabà and Andrey Kirianov) . . . 36
Storage & Filesystems . . . 37
Advances in storage technologies (Joe Fagan) . . . 37
Basic IT Services . . . 37
Centralising Elasticsearch (Ulrich Schwickerath) . . . 37
The evolution of monitoring system: the INFN-CNAF case study (Stefano Bovina) . . . 38
Unified Monitoring Architecture for CERN IT and Grid Services (Jaroslava Schovancova) . . . 39
Data Collection and Monitoring update (Cary Whitney) . . . 39
Flexible, scalable and secure logging using syslog-ng (Péter Czanik) . . . 40
Typical syslog-ng use-cases at our Tier-1 (Fabien Wernli) . . . 40
Friday 28 April 2017 . . . 42
IT Facilities & Business Continuity . . . 42
Wigner Datacenter cooling system upgrade (Gábor Szentiványi) . . . 42
CERN Computing Facilities' Update (Wayne Salter) . . . 42
P2IO/LAL Datacenter Extension (Michel Jouvin) . . . 43
Manage your hardware failures in an (almost) automated workflow (Mattieu Puel) . . . 43
Miscellaneous . . . 44
Unscheduled Security Demo (Liviu Vâlsan) . . . 44
Workshop wrap-up (Tony Wong) . . . 44
The Wigner Data Centre Visit . . . 45

Monday 24 April 2017

Welcome (Balázs Bagó)

Gábor Pető briefly welcomed us to Budapest and HEPiX Spring 2017. He gave the floor to Balázs who, after covering some logistics, presented the Wigner Datacenter. Wigner RCP belongs to the Research Network of the Hungarian Academy of Sciences. It was established in 2012, as a result of a fusion of the Institute for Particle and Nuclear Physics and the Institute for Solid State Physics.

In 2004, they started contributing to WLCG, which was a significant milestone. The centre later won CERN's T0 tender in 2012. Three different kinds of services are offered: pure infrastructure, services to CERN (with an availability of 99.99%, which they reached) and cloud services for the academic sphere in Hungary and abroad. They now wish to contribute to R&D projects, too.

Szilvia Rácz took the word and further presented the services provided by the data centre, namely cloud solutions – public and private. The most important challenge is security isolation in public and private clouds. They also focus on scientific application such as big data visualisation. They're looking into GPU simulations, together with Wigner scientists. Szilvia emphasised the interest of the centre in coming up with some partnership with the HEP community in this area.

Site Reports

CERN Site Report (Jérôme Belleman)

After presenting CERN and its IT department, notably highlighting the commissioning of a third dedicated link between the Meyrin and Wigner data centres, Jérôme went through the latest of the major activities from each team.

CERN IT performed the AFS disconnection test, and saw to various updates related to storage. The CERN tape archive now stores 190 PB, comprising 46 PB of LHC data in 2016, with a peak growth of 11 PB in July 2016. EOS and CERNBox are becoming increasingly more popular, currently keeping 1.3 billion files, amounting to 170 PB, accessed by 9 500 users worldwide. Batch services are evolving quickly, with the ever increasing resources brought to the HTCondor cluster, and developments in the areas of high-performance computing with SLURM and volunteer computing with BOINC. The rise of virtualisation carries on with more than 25 000 VMs and impressive hypervisor hardware updates. The database team continues to investigate new technologies. The Database-on-Demand service, which previously already offered MySQL and PostgreSQL services with administration privileges for the users, now adds InfluxDB to the portfolio. The Hadoop service was further improved, supporting a variety of technologies such as HDFS, YARN, Hive, Impala, HBase, ZooKeeper. Kafka was mentioned to work with big data streams, also increasingly used by the monitoring team, as part of their monitoring service. On the security front, SSO supports trusted organisations: eduGAIN covers thousands of them and CERN supports Sirtfi-compliant ones. An authorisation service is coming up, which will help users manage authorisations for applications they develop, enabling them to set up application-specific roles. The TWiki service, mainly used by LHC experiments, has been used for nearly 15 years at CERN. It is becoming more popular than ever for its collaboration features. The latest TWiki 6 provides a better editor, dashboards and column layouts. The TWiki team are working on topic (i.e. TWiki page) archival.

Jérôme referred to the other talks from CERN this week, e.g. notably mentioning Puppet 4, the growing Elasticsearch service and an update on Linux at CERN.

Questions and comments: Brian asked about the new authorisation service. Liviu was happy to provide details offline.

INFN-T1 Site report (Stefano Bovina)

Bologna was chosen to host the new ECMWF data centre. Set up in an existing structure, it will provide resources to other users, too. The INFN-T1 site uses several connectivity solutions and underwent some recent evolutions in their LAN. They use GPFS where they currently store 23 PB of data. They only have

one tape library, expected to become full end of next year – currently 39 PB. They're running a tender for disks this year, namely to replace a 7 years DDN system of about 7 racks. They're expanding their fibre channel infrastructure. InfiniBand is used e.g. in the storage cluster they run for the LHC experiments.

Computing resources amount to 250 kHS06. They just installed their 2016 CPU tender last month, replacing 7 racks of old CPUs. The Ethernet connections are on the front side, which made the installation somewhat troublesome. The virtualisation infrastructure is managed by VMware. 8% of their computing resources are 600 km away. Their new challenge is to run another centre closer by, so they get easier access to the storage. The 2017 tender is almost ready. They have a concern around pricing, which has raised, and it's not clear why. They're about to start running an OpenStack pilot and are working on an HTCondor test bed.

DESY Site Report (Andreas Haupt)

Andreas started by covering the status of XFEL, where everything goes well. The Innovation Centre close by the Hamburg campus will start being built this year. The new research halls at the high-brilliance X-ray source were inaugurated.

Zeuthen was selected as Science Data Management Centre. New modern-looking buildings are to be erected on the campus for this reason. Zeuthen celebrated its 25 years anniversary. It's involved in the gamma-ray telescopes. Wolfgang Friebel retired and the team hired two new colleagues, Timm and Fabian, here at HEPiX this week.

DESY seems to be one of the largest off-site AFS users and took the disconnection test seriously. They perform checks every 5 minutes about opened files on the CERN AFS cell. The number of users/day is now gently decreasing. In the beginning, some power users still used the CERN AFS cell. The DESY grid worker nodes showed activities from grid jobs, too. Different LHC experiments have different AFS usage patterns. But nobody really complained. Future disconnection tests should be longer to make sure that users become aware.

The current server procurement contract ends this year, hence a new call for tender is in order. The new contracts will cover a 5 years period. Another call for tender will cover mass storage servers, to be launched later this year. DESY run an OpenStack pilot, based on KVM. A 1 TB Ceph storage instance is connected. They use Neutron. DNS names of the OpenStack nodes are fed via a REST API. Core services are kept in separate XenServer nodes.

DESY have also been working on a new identity management system, a mail quarantine framework and a web-based training software. They started a Windows 10 pilot phase. Some investigations are being carried out on privacy settings. Let us not forget the HTCondor workshop at DESY this year, early June.

Questions and comments:

- Ian commented on the 24 hours AFS disconnection test, emphasising that the test will be repeated, for longer.
- Michele asked whether DESY have plans to completely phase out AFS. Peter van der Reest said that they'll keep it running until 2019 at least.
- Which product is their identity and access management system based on? – It's mostly home-grown. The quarantine system is partly home-grown too, and made from open-source building blocks.

LAL/GRIF Site Report (Michel Jouvin)

They're very happy with the current infrastructure, which is shared by several labs. Their extension is finally starting, after some delays. They were hoping for a better situation in terms of manpower – and it happened. The cloud services are managed by 3-4 people, they hired apprentices and there was no retirement – none foreseen either.

Michel said that some of their old compute hardware is not willing to die. They intend to increase the virtualisation of their services. On the storage front, LAL use Dell systems as well as some very reliable NetApp ones. They're willing to look into Ceph: they're about to deploy a 1 PB system.

SL5 was almost eradicated – only 5 machines running. CentOS is growing, mainly for service machines. User nodes aren't using CentOS very much. LAL use Ubuntu for accelerator controls. They run a mix of Windows systems, even Windows XP nodes needing special drivers. They're faced with a strange problem with Windows 10 and eduroam: there seems to be a problem with the French CA and requirements for certificates demanded by Windows 10; LAL are asking the HEPiX community for ideas.

Some major changes took place in the mail service: Zimbra will be operated centrally by CC-IN2P3, which means that about 600 accounts need to be migrated. A user even has 1M mails in his inbox. Some months ago, Michel described their involvement in H2020. LAL will have some responsibility in operating a test bed. They've been working on a unified HTCondor pool, now looking into doing something similar for storage. They took the major decision to stop StratusLab and migrated to OpenStack. Their cloud is managed by Quattor for services. Cloud resources are funded by the Paris Sud university, a key component of the computing infrastructure. Since the migration, LAL got very good feedback from users. Since one year, new use cases have been explored, e.g. Spark on demand, where storage performance is key. There's a growing interest in this technology in all scientific fields. They're still considering virtualisation of their worker nodes, which they expect to be made easier with HTCondor. They're considering containers, too.

The Refondation Project is a major project being currently discussed, a merge of five P2IO labs from the Paris Sud university. The goal is to cover nuclear physics, HEP and astrophysics. There will be a potential "computing division" of 60 people. Half the people concerned are taking part, and it's a time-consuming process. They had started working on all this already 5 years ago.

RAL Site Report (Martin Bly)

Martin presented updates on hardware: 240k HS06 (24k cores), 16.5 PB in CASTOR, 13.3 PB in Ceph. They're still using T10KD tapes – 50 PB today. They performed a large tape migration which showed absolutely no errors. They increased their OPN link to 30 Gb/s. They added Mellanox switches, with Cumulus Linux on switches used in VxLAN. They finally got to IPv6 on the T1 network.

RAL run jobs in containers with HTCondor and have been investigating Kubernetes to provide portability between on-premises resources and multiple public clouds. They still run Ganglia for monitoring but it's going away. They use Telegraf instead.

They performed a CASTOR upgrade to 2.1.15 and are about to do so again to 2.1.16. The SRM update to the latest version was carried out, then they had to roll back because of performance problems identified by LHCb. On the tape front, they're wondering about the long-term support for drive and tape libraries and are waiting for announcements from Oracle. On a brighter note, they're running Ceph (Echo) smoothly, already storing 1 PB of data for ATLAS. The usage of Echo has been increasing ever since end March.

They installed new chillers. The PUE reduced from >1.64 to 1.35. The photo on the last slide depicted the new chillers being installed.

They got some new hardware for the CVMFS Stratum 0 service. They're looking into VMware to replace Hyper-V, because of crashes they had – still unclear what the cause is. Databases still run under RHEL5. They are retiring SL5 systems for other services and had to patch for security vulnerabilities. An interesting note for Windows systems, the administration privileges were removed from all federal IDs, following an STFC policy.

Questions and comments: Michele asked about how long a time for a return on investment with their spend-to-save policy? Martin expected 4 to 5 years.

Swiss National Supercomputing Centre T2 Site report (Dino Conciatore)

The centre is located in Lugano. Their flagship machine is called Piz Daint, a hybrid Cray XC50/XC40 system. There is a picture of it on the slides. The cooling system pulls water from the lake, 6°C all year long.

All resources are shared in the T2 cluster. They stopped using CREAM in favour of ARC. All is Puppetised. They use two InfiniBand bridges connected at 80 GB/s. They run dCache and GPFS for the `/scratch` system. Each server receives the same configuration, with fully-redundant fibre channel connections. Dino presented an overall view of the services. BDIIs, ARC CEs, Argus, two VOBoxes (one for ATLAS, another one for CMS). Nagios is used against central logs. Elasticsearch and Grafana for dashboards. One year

ago, they have started a project to collect all the information from the scheduler and job completion into Elasticsearch, giving a nice view of the cluster. They use PySlurm to get that information from the scheduler.

The LHConCray project aims at consolidating everything on the big machines – all LHC jobs. The goal is to run the jobs without having to change anything in the workflows. Containers will be run, and CVMFS be used. The main issue was to run CVMFS on the machines and they now succeed in running it for production jobs. Every time a container runs, it gets data via `/scratch`.

Everyday they deal with massive file creations, which is becoming a problem for them. CMS was generating about 1M files/minute and quickly ran out of quota – only them seem to be doing this. The other VOs aren't affected. They've been working on SLURM fairshare, running 10k jobs in queue.

Questions and comments:

- Do worker nodes have outgoing network access? – Yes.
- Could you provide some details about the Elasticsearch cluster? – It's an 8 nodes farm with around 32 GB each node. It stores 2 billion messages, keeping 3 months' worth of data. The team will expand the cluster to 12 nodes to keep 1 year of information. They're working on consolidating everything on Elasticsearch – metrics, load information.

T2_FI_HIP Site Report (Johan Guldmyr)

The site provides managed hosting for various institutes. Johan started by telling the tale of their hardware failures. They have a large RAID configuration. Three disks once decided to disappear, causing the system to shut down. NDGF started draining other pools. They first decided to replace the disks, restart and wait. It didn't help. They wrote a report parser to understand the logs better. Another three disks failed. They rebooted again, but the filesystems wouldn't mount anymore. They had the I/O module and cables replaced. This time, they couldn't restart as the local metadata was corrupted. They replaced the controller, as well as all the metadata. NDGF moved files to other sites. Pools were emptied and the array and filesystems recreated from scratch.

In other news, they moved a physical rack, with all the servers kept inside. They no longer have a private OPN to the Helsinki University for dCache traffic. They tried the new Ceph back-end for dCache pools. Basic functionality is there.

They also took part in the Finnish Grid and Cloud Infrastructure and contributed to the code, especially related to SLURM (high availability, mail on job completion showing reserved and used resources by the job). They designed dashboards with Grafana, namely by using a Grafana heat map plugin.

Questions and comments:

- Do you use Ansible for everything? – We do for the grid infrastructure.
- How do you cope with errors? – If the Ansible pool fails, red blinkers appear on the dashboards.

Site Report for Max Delbrück Center for Molecular Medicine (Alf Wachsmann)

Last year at HEPiX, Alf already gave a presentation about life science. He explained the benefits of learning from HEP computing. The Max Delbrück Center is part of the Helmholtz Association (DESY is part of it too). There's a lot of green on their campus. They've got buildings where they do science, other ones for translational medicine and some buildings host start-up companies related to medical technologies. The IT department provides a large variety of services for the entire campus. There are some vacancies and Alf asked for any interest.

They use a redundant connection to internet and another one to BRAIN (the Berlin Research Area Information Network), allowing the exchange of large quantities of data. Some institutes part of the Helmholtz Association aren't located in Berlin. On campus, they've got a 10 Gb/s backbone, being upgraded in some buildings to reach 40 Gb/s. Given that they have the facilities and the computer centre on campus, this fast backbone makes a lot of sense. They will need to think about what to do with their firewall when they upgrade the BRAIN connection, too. They're working on deploying wireless networking on campus.

Access points are Cisco – they recently renewed controllers. HPC uses Dell/Force 10 hardware and this may change in the future. It's not an IPv6 site yet, for lack of manpower and it's not really an issue for them at the minute anyway.

HPC runs through a standard Beowulf cluster connected to a 40 GB/s backbone, with a tender open for 70 compute nodes. They use GPFS and an Oracle NFS on ZFS appliance which causes them some grief – they're in touch with Oracle. Everything is Puppetised.

It used to be a Windows shop, with no support for scientific computing. This is why they still use Active Directory, Microsoft Exchange, …. They just replaced 4 Oracle SL5000 with IBM systems for backup and archival, with more space to ultimately reach 30 PB. They moved from Oracle SAM-FS to Versity to be able to use Linux. VoIP is available in some buildings, but it's otherwise mostly regular telephones. Telephony is not yet run by IT.

Questions and comments: How many mailboxes does your Microsoft Exchange manage? – There's about 1700 people working here, including guests and students. This results to between 2000 and 2500 mailboxes.

NDGF Site Report (Mattias Wadenstein)

They're a distributed T1 site for WLCG, spread out over 6 HPC sites and federated disks from a Slovenian T2. They're the second largest ATLAS data disk pool (behind BNL). A new dCache developer will start next week and NDGF is still recruiting. Last time, they were upgrading to work on an HA dCache. A hardware update typically takes a day. Mattias described the procedure to reboot their head nodes. They do so very frequently with no user impact. He then described the head node upgrade procedure, involving disabling and re-enabling HAProxy. They had a few bugs in their dCache Zookeeper queuing – now fixed. They followed a rather agile approach involving frequent kernel updates with reboots over the last 6 months. They're running on Ubuntu that releases kernel patches early and often.

NDGF bought more Dell 730XD dCache pool servers and upgraded Ubuntu in their Abisko cluster. They plan to run Singularity to provide ATLAS environments. Mattias described a performance regression on HPE RAID controllers, from 1.8 GB/s to 0.6 GB/s, well below the minimum required. Whichever kernel currently running on CentOS/Red Hat/SL seems to be involved in the cause of this. It could be a driver bug. They set up a udev rule to set the maximum kB size for single I/O to 512 to mitigate the problem.

IJS got more storage resources for dCache pools. Ceph is used for cluster cache at SiGNET, showing good performance. Network upgrades for LHCONE took place. They're pioneering on the Singularity front for ATLAS as a generic container for their clusters to execute jobs. Computing resources, or equipment, keep being moved to the new centre at the JSI Institute.

NSC suffer from funding problems, due to a reorganisation. Delays were introduced to fund replacement clusters. They reduced the number of nodes but finally got the necessary help. Four PDUs burned and were replaced. They reached an agreement with the Norwegian meteorology office, acquired more disks for WLCG and upgraded their network (SUNET). They suffered from power outages, for a large part of their system. Their worker nodes went down – they were covered by a non-prioritised UPS. A detector for electric arcs kept triggering. This was because of a lose nut.

Mattias ended his talk announcing the NeIC 2017 Conference, that will take place from 29 May to 1 June 2017. There will be two days of workshops and two days of conference.

Questions and comments: We started discussing new computing models, getting rid of small sites and running bigger ones. What about NorduGrid? – It's a way to consolidate storage. The majority of the money goes to local hardware.

BNL RACF Site Report (Ofer Rind)

There's a new branding at BNL, showing that it was formed about a century ago, army-based, back in the days. Only later on did it become a scientific laboratory. It's located on Long Island, NY. It provides full service computing, mainly for the 2 RHIC experiments and ATLAS. There's two new computing clusters in RACF. They're looking into hiring more manpower.

The proposal to refurbish an existing building as a new data centre was approved. The Institutional Cluster covers the entire BNL community, 108 compute nodes, running on a non-blocking InfiniBand connection. Initial problems were encountered, since then solved. Their other cluster is the Knights Landing cluster, 144 nodes.

Their network was reconfigured. The network team are working on migrating to the 25/50/100 GbE standard. The US ATLAS T1 migration required preparation to keep it transparent.

The upgrade of the T1 dCache took place just last month. They ran into some issues, such as memory leaks. The IPv6 dual stack is now fully functional, a big milestone. They're testing new features of GridFTP and the integration with Ceph. HPSS, their hierarchical storage manager, was upgraded. It amounts to 90 PB, 65k tapes. STAR now uses LTO-7. They're expecting more tape mounts and so make two copies of the data. In light of the AFS Phaseout, ATLAS T1 no longer use AFS, in favour of CVMFS. The remaining functionality is being replaced by a multi-tiered solution. A major Ceph migration started in March. They're deploying their first Ceph cluster on entirely new hardware. The old ATLAS Ceph cluster is being repurposed as test cluster.

New ATLAS systems were brought up this month: 48 Dell R430 systems, each one with 128 GB DDR4, 4 3.5" 2 TB SATA drives. They're rolling out collectd for monitoring and are very happy with Grafana. They use Singularity, to address the issue of HTC/HPC job mobility. It's very lightweight, no daemon and a single package to install. They have an ATLAS SL6 container for their environment with all the necessary bind-mounts.

AGLT2 Site Report Spring 2017 (Shawn McKee)

They run close to 10k cores and almost 7 PB of storage. All the services are virtualised on VMware – going now to version 6.5. On the networking front, they use 2× 40 Gb inter-site connectivity, 40 Gb between T2 and T3 sites. AGLT2 developed their own monitoring to preemptively address problems, based on Ganglia, Kibana, and an ELK stack. They underwent a personnel reorganisation. They're very happy with their good students.

They received some funds to ensure all is under warranty. They purchased some new Dell hardware, adding 24 kHS06. T2 VMs were rebuilt to use SL7, they updated dCache, HTCondor, OSG CE. They still run Lustre. They had some challenges with running it over ZFS.

Shawn previously reported they may need to move to a new location. Renovation works took place around the T2 centre space. They're taking part in SC17, working on the integration of Open vSwitch and setting up IPv6 dual-stack for all their nodes.

Nebraska Site Report (Brian Bockelman)

Nebraska bought new computers, the facility now reached 35k cores. Their heavy-duty usage of network pushes it to 30 Gb/s. The site offers services based on OpenStack and Ceph. They set up containers all the way. CVMFS is used for data federations (requiring x509 authentication), most of which work is carried out on OpenStack. They're looking into expanding their use of CVMFS for data federations.

Brian described their computing turducken with containers. It allows them to run pilots with the OS version of their choice, and do so too with the actual payload. Some OSG software components were removed: GRAM, GIP/BDII, Gratia, bestman2 – replaced in favour a simpler solutions. Some more components are planned to be retired, too. Singularity will replace gLExec. Without gLExec, they no longer need pool accounts. Without pools accounts, they no longer need GUMS. VOMS-Admin was used to keep track of the pilot service, and they found they didn't need a large web app to do so. Using more and more containers, running more and more smaller services, it's expected they'll soon do service orchestration – Kubernetes was mentioned.

End-User IT Services & Operating Systems

CERN Linux services status update (Ulrich Schwickerath)

Ulrich listed the different versions of Linux run at CERN. The workhorse version is SLC6, the latest one CentOS 7.3. CERN contributes to some Special Interest Groups and took part in the first CentOS Interlock. An interesting challenge is the support for alternative architectures, such as ARM. CentOS 7 CERN is based on upstream RPMs. The CERN Linux Team provide some additional software. There are snapshots made available for users needing to not move at the same pace as others. The LocMap tools replaces the LCM tools which used to be Quattor-based to configure desktops. LocMap was developed by Aris Boutselis, a technical student at CERN. It's working well and will receive new features soon.

The CC7 release cycle involves the CentOS 7 release, a testing phase, then the actual CC7 release. Almost 31k machines request updates from the Linux services, increasingly more so from CC7. CERN run Koji for building RPMs and cloud images, thanks to the Image Factory plugin.

Questions and comments: Is Red Hat 8 Enterprise coming along? Or do containers make this irrelevant? Ulrich will ask the Linux Team.

SW & Computing for Big Science Journal (Michel Jouvin)

Yves Kemp (DESY) had offered the idea of a journal in a previous HEPiX. It was presented in a variety of other fora as well. The feedback was always positive. The scope of the journal was unclear in the beginning. The main motivation is that several scientific communities produce large quantities of data. These experiments are not run by individuals, but whole collaborations. There's also astrophysics, biology, … All these communities share similar technologies and challenges. If there were more presentations from these communities (see for instance Alf's contribution), we could all get something out of these.

Sharing the information is the best way to avoid reinventing the wheel. Yet, the publication challenge is that there's not place to publish our R&D and solutions. The other challenge is that it's not enough to just publish something on a blog. If we want the information to be useful to others, it needs validation. The context needs validation too, to be recognised, the impact factors clearly identified. It has to be indexed. The goal is to publish a few tens of articles/year, which entails some considerable work.

The *Computing and Software for Big Science* journal is already being published. Christian Caron, who was previously at HEPiX, was enthusiastic about this and made it happen. The journal is being published in partnership with Springer. We don't plan to publish proceedings, which could cause trouble for indexing. The list of topics is basically the one presented by Yves a year ago. The journal was launched this winter. Some recognised people (e.g. John Harvey) have contributed. The article submission has been opened. We need to consider what we want to publish. What's presented in HEPiX is probably worth publishing. A minimum would be a few pages, but it's happened before that much longer articles were published (as long as 900 pages).

This is our – the HEPiX Community's – journal. There's associate editors from all continents – about 15 of them. We are amongst the potential authors and we should contact the editorial board if we've got any suggestion. It's not only about particle physics, it's also for big science.

Questions and comments: Tony emphasised that HEPiX have a role to play – is this something to advertise? – Yes, absolutely. We could add pointers from our website to remind people that contributions are welcome. Certainly we shouldn't spam our community all the time about it but a reminder from time to time would be welcome.

Security & Networking

IPv6 at the RAL Tier 1 (James Adams)

James is a system architect involved in Quattor and a technical consultant for Ceph and the IPv6 configuration. The WLCG T1 was expected to provide IPv6 to support IPv6-only sites, not so much because of

the address space limitation. Since the T1 relies on the STFC network, the STFC network had to turn to IPv6, too. There was some legacy infrastructure to work with too, some of it not supporting IPv6 at all. The first step was to build a test bed and identify those services to be using it. The network components which were there underwent firmware updates. There was the question of how the T1 IPv6 network was going to be structured, which James quickly went through with diagrams. Today their work resulted into an IPv6-capable network. Allocating subnets and addresses came next, with JANET at the top level. They looked into the STFC and the T1 addressing schemes separately.

They've got a perfSONAR nodes or two which seem to work out of the box. CASTOR, however, will not support IPv6, and CERN have no intention to make it dual-stack either. CTA will be IPv6-accessible, however. The Echo release of Ceph currently has an IPv4-only endpoint. Dual-stacking is currently being tested. Future plans include supporting other services such as FTS, CVMFS Stratum 1, Squid, …

Questions and comments:

- Brian asked, how do you know the firewall at the host-level? – It's something to check on a service-by-service basis. The idea is still to close firewall traffic down and then open as needed.
- Mattias asked, in this an IPv4 mapping scheme, are there IPv6-only hosts? – No.

Basic IT Services

A Hard Puppexit from 3 to 4 (Jérôme Belleman)

CERN upgraded to Puppet 4. Jérôme first presented the motivations. Among the strategic ones, he recalled that Puppet 4 had already been released in April 2015, that Puppet 3 reached its end of life in December 2016. While it's true that many sites still use Puppet 3, he noted that other ones have already migrated. Moving to Puppet 4 would also open the opportunity to update other configuration management components such as PuppetDB and MCollective afterwards. Finally, some interesting modules only support Puppet 4. There are technical motivations too, such as the stricter compiler, syntactic improvements and the hope for a performance improvement.

CERN organised the migration in three phases, over a period of three months. The first phase – January 2017 – was referred to as the volunteering phase where everybody was invited to run their Puppet agent in dry-run mode against a specific port, talking to a Puppet 4 master. The second phase in February entailed running the Puppet 3 agents against Puppet 4 masters. The third and last phases in March was about upgrading the Puppet agents to version 4. A new single RPM – `puppet-agent` – was provided by Puppet Labs, bundles Puppet 4, MCollective, Facter, Ruby, … and installed the files in an all-in-one layout, under `/opt/puppetlabs/`. The advantage is that it will become easier for the CERN team and Puppet Labs to manage. The drawbacks are that it took 2 weeks and several bug reports to successfully build the package first time. Also, paths and service names all changed, requiring some more work. The approach was to deploy configuration files in both locations and use an intermediate, empty, transitional package to perform the installation.

With the upgrade to Puppet 4 came the opportunity to perform some validation and testing with ModuleSync, to synchronise templates to all module and hostgroup repositories to make it easy for users to run `puppet-lint` and perform validation and RSpec tests. The tests can be run locally or as part of GitLab CI. The purpose is to catch regressions during significant migrations such as this one, and save time as fewer Puppet runs should be required, and there's a lesser need for environments. This will also help testing future Puppet versions. In effect, users who checked their code with CI cleaned up their code a lot and now pass `puppet-lint`.

Collaboration was key during this migration. The main users were invited to test their code early on (see the first, volunteering phase). Reinstallation tests (on just one node) were advised. Users were invited to report on any central problems. The configuration team often received similar questions, which made them realise they should have worked on their FAQs. Jérôme showed the number of support tickets increase in correlation with the three phases, noted a definite increase with a peak when everything was deployed and a sudden drop afterwards. Time will tell how this will evolve in the future. Would longer

phases have made for a smoother user transition? Some say it wouldn't have helped and that some users would have still waited until the last moment to start moving.

As the old and new masters were kept in separate sub-hostgroups, and listening to different ports, and as agent configuration files were located in different places, the switch was a smooth one. However, some nodes were left behind and kept on talking to Puppet 3 masters, or kept on running Puppet 3 agents. This was sometimes due to broken catalogues or to the Puppet agent having been intentionally disabled (sometimes when one needs to perform manual changes and doesn't wish to have Puppet in the way). The number of requests to `/node` showed that the greatest part of the move happened within a day. For those nodes left behind, the configuration team regularly sent node lists to users, helping them out wherever needed.

An embarrassing issue was that user crontabs were lost, because Puppet 4 started managing all user crontabs, in addition to system-wide ones. If users had any user crontabs, the upgrade would purge them. This is when the configuration team realised how many crontabs users have. Another problem is that relative paths are no longer searched, and the scope must be explicitly specified. More annoyingly still, the empty string now evaluate to false, which is what bit everybody the most. Puppet Labs provided a page listing these changes of behaviour. Finally, while it was expected that performance would improve, it did not, as can be seen with the catalogue compilation time which notably increased as the upgrade was rolled out.

In conclusion, CERN now have better tools for their everyday life and hope the Puppet 4 to 5 upgrade to be even smoother. They now have the opportunity to update the MCollective shell agent as well as PuppetDB, which currently suffers from PostgreSQL bloat, log clutter and a garbage collector timing out. The team are very thankful for their colleagues in IT and users for their collaboration.

Questions and comments:

- William expressed his surprise at the fact that the scope had to be specified even from within e.g. the same module.
- Regarding performance, the question was asked of whether JRuby was used, which is the case as it's shipped with Puppet Server.
- Tony asked if the user crontab problem could happen again during the Puppet 4 to 5 migration. – It won't, since user crontabs are now Puppet-managed.

salt stack - Using, Extending, and programming (Owen Synge)

Salt was built with the idea of how to run a submarine infrastructure. All CMSs are the same: they install packages, write config files, … Salt uses YAML with Jinja2. Other CMSs, in order of importance, are Puppet, Chef, Ansible. Ansible and Salt push to the node. Puppet and Chef are Ruby-based, Salt and Ansible Python-based. Salt has got a steep learning curve.

To configure your system, there's many things you can do. Salt is event-driven and that's what it excels at. A pull-based system is very useful. A push-based system means a master must be running. Otherwise events can be lost. This is what Salt isn't good at. Salt and Ansible still seem to go through maturity problems which Puppet and Chef already solved.

There's a Salt master with a host event bus. The Salt formula is what the custom DSL is called, which is YAML with Jinja2. The programming power Jinja2 offers should be used sparingly. Upon exceptions, the information is raised to the master. But errors are easily lost in the noise. Logging is done on the nodes. Modules simply use Python logging. So-called state modules are more user-friendly. They're meant to be reusable execution modules. You can use Python unit testing frameworks and the other facilities Python comes with. Owen wrote Ceph components this way. Salt is annoying with scope. Variables are best be made global within a module. Salt formulas take their variables in the form of Jinja2, which can get their variables from pillars, which can get the values from a database – until it breaks. Owen even added inventory DBs. He also wrote a function to find the master. Salt is completely reconfigurable, which is precisely what makes it rather powerful.

Questions and comments:

- Is there a possibility to manage e.g. storage sensibly, are there best practices? – With salt you can send an event at the end of a stage, to trigger the next stage, which is often a sensible paradigm.
- Is there a way to cache the pillars (i.e. tree-like structures of data passed through to the minions) on the minion (i.e. clients)? – It is in fact cached on the minion, but what's now obvious is how to send the event that it's done receiving the pillar?
- The audience showed interest in Owen's database work, namely in order to build YAML from data coming from a database.

Site Reports

CEA Site Report (Arnab Sinha)

CEA work on UNIX, the Grid, infrastructure and security. They use Puppet 4 and Ceph. Arnab described their Ceph instance of 5 servers. They use the Calamari administration tool for managing it. The Grid Ceph instance was reinstalled with the BlueStore back-end. They suffered from high memory usage. They got rid of EC pools and currently run a single VM in BlueStore. CEA monitor their Ceph instance with a Grafana dashboard.

There were some issues with HTCondor too, notably scaling with ARC. The site is fully dual-stack. They're moving towards cloud computing, collaborating on a variety of cloud projects. They have an agenda of installing new cooling systems, as well as some monitoring to avoid repeating a major air conditioning problem they had in 2016. They will have a new real-time power usage system, which will enable them to calculate their PUE precisely.

They're setting up some new security policies, following a number of norms. They updated security for most of their Windows machines – slowly catching up for Macs and Linux workstations. They recently launched the GLPI tool. Their ongoing projects include deploying an inventory system – SCCM for Windows, FusionInventory and GLPI (a recently-launched help desk tool) for Linux and Macs. They're planning to have software inventory too, as well as hardware inventory later on.

Tuesday 25 April 2017

Site Reports

Experience on the operations at new KEKCC (Tomoaki Nakamura)

"The Belle II detector was rolled-in to the collision point of the SuperKEKB accelerator", a press release announced. Tomoaki showed a schedule of the accelerator. The upgrade was completed. Data taking could then commence. The number of concurrent running jobs increased. Recently, the processing of Monte Carlo simulation started. It is expected that the computing system will soon face some increased load. They currently run 10k cores, 10 PB of storage, 70 PB on tape. They upgraded Grid services for the sake of robustness and to ensure uninterruptible operations. The network connectivity was improved, too. In the new KEKCC, the computing load has already become higher than where the old system culminated – and this still doesn't include local jobs. When including local jobs, the fraction of the Belle contribution is currently quite small. It will increase as more data comes in.

KEK have a tape archive system managed by HPSS. All users can access data via GPFS. With the start of the new system, they manually staged some high-priority data from user lists. There are some spikes of staged files/minute which are manual stagings. According to the specifications, mount and unmount operations take 15 s. There have been political issues when many concurrent stagings took place, as many jobs came in, at which point as few as 4 files/minute could be served in the worst case. They recently worked on their private cloud with an OpenStack deployment. Local and Grid jobs use LSF. They wrote some base images for the VMs. The team are looking forward to welcoming the HEPiX community at KEK next workshop.

Tokyo Tier-2 Site Report (Tomoe Kishimoto)

Tomoe presented the International Center for Elementary Particle Physics. It's the only ATLAS WLCG site in Japan. Hardware is leased and replaced every three years. It currently amounts to 10k cores. With a new arrival of CPU cores, the number of completed jobs increased again after a previous drop. They achieved a >99% availability. They run up-to-date versions of their HTCondor and CE software. They are in the process of replacing their Torque/Maui farm with HTCondor. They evaluated ARC and still run CREAM with Torque/Maui. They introduced dynamic partitioning with HTCondor. Tomoe compared static and dynamic partitioning, with some idle cores in the former case. The HTCondor dynamic partitioning showed a better CPU utilisation than Torque/Maui with static partitioning.

The site recently updated their database software. Their disk storage is managed by DPM (>6 PB available) and the database MySQL (whose size reached 80 GB). They currently have no redundancy in the MySQL database. They do however have some semi-synchronous replication, with daily backups from the slave server. They use a Fusion-io ioDrive to reduce maintenance times.

SINET5 is the academic backbone network in Japan. It's connected to the US sites with 100 Gb/s links and to the European ones with 20 Gb/s links. The campus network was upgraded from 10 Gb/s to 20 Gb/s, and some transfers were observed to have used the larger bandwidth.

Questions and comments:

- Why use ARC CEs and not CREAM or HTCondor-CEs? – This is due to ATLAS requirements, and they already use ARC CEs.

KR-KISTI-GSDC-01 Tier-1 Site Reports (Jeongheon Kim)

KISTI is a government-funded institute inaugurated in 1962. The GSDC – Global Science experimental Data hub Center – has been a government-funded project since 2009: the purpose of the data centre is data-intensive fundamental research. They run 25 storage racks with 6 different models of servers. Their T1 operations involve almost 4M jobs in the last 6 months, 3.9% of the ALICE payload. They provide 1.5 PB of disk space, 3 PB on tape.

They upgraded FreeIPA from 3 to 4. They moved to Puppet 4, updated Foreman and deployed it all to CentOS 7. The wrote a Katello install script which makes a secret vault, bootstraps Ansible, checks requirements, upgrades packages, sets up components and runs the Katello installer. Jeongheon's team investigated the use of Kubernetes, with the Atomic Host container OS. They evaluated different container network technologies such as Flannel, Weave and Calico.

Questions and comments: Are the many different models you use in your storage racks from the same manufacturers? – Yes, mainly Dell and IBM.

Status of IHEP site (Jingyan Shi)

Jingyan presented the dedicated links to their three sites, with another two coming up. Connections to Europe and the US are of 10 Gb/s. They migrated to HTCondor for HTC (10k CPU cores) and created a SLURM cluster for HPC (3k CPU cores). They still use Torque in their Grid site. The local cluster offers 9.3 PB (covered by Lustre, EOS and more). There's 5 PB on tape (CASTOR). Their provide 400 TB in DPM, 540 TB in dCache for the Grid site.

They have plans for networking, to isolate failures by performing some physical division, to be deployed in August. They performed the migration to HTCondor in 3 steps to limit risks. They use a new shared scheduling strategy to provide a higher resource utilisation. They provided their users with the `hep_job` tool set and will add more features to it. They intend to use more virtualisation and containers. Their cloud computing is based on OpenStack Kilo for two main use cases: IaaS on the one hand, user self-service on the other hand. Jingyan mentioned the VCondor tool they developed at IHEP. Their plan for Lustre is to upgrade to the next stable community version, with intentions to improve on data reliability, availability and I/O performance. For a while, they've been using EOS for batch computing, with 5 servers – 1.1 PB of raw capacity. They've developed IHEPBox based on ownCloud and EOS – 192 TB of raw capacity. They've been doing some work on CVMFS too for distributing IHEP experiment software.

Questions and comments: Tony noticed some interest lately in IHEPBox, CERNBox, … Is this initiative part of the AFS phaseout at IHEP? – No, IHEPBox doesn't have any link to the AFS phaseout. It's more for the comfort of users, to provide them with some home space.

Security & Networking

Computer Security Update (Liviu Vâlsan)

Liviu described the discovery of a new malware. It's a new rootkit, called VENOM and never seen before. It comprises 2 components, a user-land back-door with remote code execution, and a kernel part featuring an additional port-knocking mechanism. Binaries are compiled on the victim's machine in `/dev/shm`. Paths are changed to resemble legitimate Linux components. Log files are erased, filesystem timestamps manipulated. Liviu then went into some details, confidential information omitted from the live stream and this report.

E-mail is still the main infection vector. There's all kinds of malware distributed this way, including ransomware. Payload used to be embedded in the mail. Nowadays we mostly see malicious attachments and URLs. Users seem to be willing to jump into traps. How many users will click? CERN regularly runs campaigns to find out. Some e-mail messages designed by externals students without any inside knowledge of CERN – and the least well crafted ones – nonetheless caused click rates of nearly 19%. And some users don't seem to learn as many of the same people will fall into a similar trap the next year. The CERN security team sometimes uses nodes outside CERN, sometimes inside CERN but outside of the CERN network to run their campaigns. On the plus side, many users reported phishing mails to the security team, sometimes asking what to do after they clicked anyway, rather surprisingly. Liviu presented a typical mail they send and described the giveaways: cern.com, the message isn't signed, blunt typos, … Some messages are more effective than others – the successful login attempts from China scam being the most successful one. Most people click from outside CERN. Most users click from within 10 minutes after the mail was sent. The way CERN filters incoming mail involves cloud provider rules, organisation rules, sandboxing, antivirus on

endpoint systems and network rules. Sadly, attackers use randomisation, multiple payload URIs and various obfuscation techniques. It's sad that .science is the most-abused top-level domain. Threat intelligence plays a key role in daily operations.

Questions and comments:

- Michele described a commercial solution whereby e.g. `.doc` attachments are converted to PDF before being sent to the user, and the `.doc` file sent to a sandbox. The user must then explicitly ask for the original `.doc` file. This costs 100 €/mailbox. – This is a good approach, but one problem is the privacy – and the cost.
- We need to establish more communication links between our sites.

Security and networking: Security workshop (Liviu Vâlsan and Romain Wartel)

Why share indicators of compromise? For detection, for blocking, for performing intelligence, all of which objectives can be conflicting. It's mainly done for detection and blocking at CERN. MISP is an open source project for sharing indicators of compromise. It's about sharing threat intelligence. Everyone can be a consumer and/or a producer. There's no obligation to produce. MISP offers flexible sharing group capabilities, automatic correlations, … and most importantly a free-text import helper (e.g. to create events by dumping the contents of some security report). Attributes are indicators of compromise which contain a pattern that can be used to detect suspicious or malicious activity. It always belongs to a category (e.g. payload delivery which tells a story and a context). It's got an IDS flag which tells whether it can be used for detection. Sometimes people produce intelligence without changing the IDS flag, which is a problem. A MISP event is a container for grouping attributes, related to a security event (e.g. a malware variant).

We then moved to the hands-on part. MISP can be set up with a Puppet module, nowadays. Being a security tool, it requires HTTPS. Some network configuration work for the VM was necessary with respect to the conference network setup. A MariaDB instance needed installing, setting up and pre-populated. MISP offers a web interface to log in into. The workshop involved creating some intelligence, i.e. security event. The metadata includes the distribution level and a description. This will create a container where attributes can be populated. Liviu invited us to add as many attributes as we wished. A category must be chosen (e.g. network activity) and the type (e.g. URL, IP). You can't use patterns. The *for Intrusion Detection System* option enables the event to be acted upon.

We could then synchronise between different instances. An organisation for HEPiX was set up to do so and we hence shared the UUID. MISP features an ACL system and a separate user had to be created for the purpose of synchronisation. Each user comes with an authentication key. We then tried to find other workshop participants by sharing the authorisation key and the IP address. Afterwards, we needed to add a server, i.e. the other participant's machine – which was problematic due to the VM NAT setup.

Liviu had a number of VMs prepared, namely a victim VM talking to the internet via a so-called security onion VM running Bro IDS, which pulled intelligence from MISP. In the event list, we saw there was already one published, already populated with all the necessary intelligence. Liviu pulled intelligence from all of our contributions (which could be done with the `curl` command from a specific URL offered by MISP). Liviu brought up the victim VM from which he accessed the services registered in our events. In a tool called BroTop, we could see that various information was recorded, related to the event definitions.

Questions and comments:

- Must we set up our own servers? – It'd be welcome but it's not necessary.
- If we want to use the WLCG instance, can we pull events from it? – Definitely, we're welcome to do so.
- Is there an interface with dynhost? – Not a native one, but there's so many export formats there should be a way.

WLCG/OSG Networking Update (Shawn McKee)

Shawn gave an overview of what's been happening in the WLCG Network Throughput WG. It's about ensuring we understand bottlenecks better. The deployment of the perfSONAR infrastructure is part of it, as is network analytics and the network performance incident response team. Just yesterday, perfSONAR released the new major 4.0 version. The recommendation is to keep auto-update on. A bunch of features is coming up with this release. There's for instance a new mesh configuration agent, a convenient way to organise testing for various communities. With new analytics capabilities, they will be able to look at data with an ELK stack and Jupyter. There's a number of instances which reportedly don't have auto-update turned on. The initial deployment was coordinated in 2013-2014.

perfSONAR 4.0 focuses on control and stability. There's a new stand-alone mesh-config and a new test scheduler providing a lot more understanding of what the node is doing. This version still supports SLC6, but 4.1 won't anymore. A new endpoint selection capability provides better topology information. For latency nodes especially, there are new minimum hardware requirements. It's worth noting there's a lot of I/O load, too. Support for VMs and docker is coming up.

OSG runs a collector which gathers data into a data store so that end users can access the data, either by using e.g. the ELK stack or work with it differently. MCA gathers information on hosts from several sources. It provides a GUI to organise meshes, providing autocompletion – very easy to use. You can filter, create dynamic hostgroups. This whole platform is handed up to perfSONAR developers, and in the future it will be handled by them. There's even a Google Maps plugin. ETF PS is a new monitoring tool based on WLCG ETF using a JSON interface on each node.

The analytics services index historical network data. Jupyter enables us to use Python to work with the data. Anybody who wants can subscribe to a topic. Shawn showed a sample Kibana dashboard of sites where the average packet loss is above 2% over an hour. There's still quite a few problems, about 5% of all paths have significant losses. To address this, alerts and notifications have been requested for a long time. What they came up with is to enable people to self-subscribe to receive mails kept to a reasonable amount. This feature is still in beta but we're invited to give it a try.

The WG want to improve transfer efficiency. A part of this is responding to tickets. Shawn presented a ticket where connectivity to and from ASGC was difficult. The investigation took less than 30 days. perfSONAR contributed a lot in managing to gain a factor 10. In another case at TRIUMF, all Canadians sites suffered huge packet losses. Rolf Seuster looked into this. He looked at all the events in the database, for each route to DESY Zeuthen he classified if it was in the "good column" or the "bad column" and identified the faulty router. This allowed to isolate the problem.

Future plans comprise the way to perfSONAR 4.1, the user-friendly OSG Network Measurement Platform and improving analytics tools – they've been looking at Grafana for better time-series visualisation. Work on notifications, predictive capabilities, capacity planning is underway.

Questions and comments: Brian asked, if a network generates problems quickly enough, can we fix them at the same rate they appear? – We now have the potential to do faster finding and fixing. In the past 6 months there were recurring problems which could be addressed more efficiently. There's always going to be unusual things popping up but we're improving.

ESnet Update (Joe Metzger)

ESnet is a mission network funded by the US Department of Energy to enable and speed up scientific discovery by delivering network infrastructure tools. ESnet has been around for around 30 years. One of the challenges is that it'll be the first part of the DOE planning. How to specify what is the technology, what is the network and how soon will it be ready? Many organisations are moving to cloud. If you move your data to the cloud, it changes what kind of network you need. Biology sciences improved automated workflow systems, other scientific areas are catching up. Not all the areas have the manpower to build the tools, define the workflows and have the networking and computing expertise. The networking industry develops mainly at the web scale. Some planning is necessary. Joe showed what might be needed by 2020. The aggregate capacities will be significant, possibly with some splicing. There's a lot of assumptions, but there are questions remaining on redundancy, capacity overhead, …

How bursty is the traffic in the HEP community? Most networks do their capacity planning based on percentiles >90, when the network is busiest. Over 80% of the time it gets half as much utilisation. They got good data for average utilisation, not for peaks. They wish to see what the HEP usage profile looks like. We're going to move from a model where large computing is all squeezed through a single channel, to a model where there's many more parallel links.

In the queuing systems of computer centres, there's high-priority and low-priority jobs, and we're talking about minutes or days. From the networking perspective, there's no telling about high-priority or low-priority packets. How the LHC computing uses network has an impact. We need to keep each other informed to avoid making these impacts unexpected.

Questions and comments:

- Can we really expect to have 400 Gb/s components this year? – The first generation of an equipment makes it work, the second generation makes it work well, the third one makes economies. But there won't be three generations by 2020. We might see 400 Gb/s components by then, but there's also a chance that we might see more lower-speed links.
- Is there a particular area in HEP where we can expect a jump? – A number of sites have a couple of 100 Gb/s links. Looking at the growth in the last 3 decades, we observed a factor 10 every 48 months. Other sites might need this deployment by 2020. There's no expectation for any significant growth, but looking at LHC Run 3, the need to scale is there.

Network related updates in IHEP (Shan Zeng)

There have been network architecture upgrades at IHEP, some work on IPv6, new monitoring tools and developments on SDN. Both links from IHEP to the US and IHEP to Europe are 10 Gb/s. Shan showed the outbound peak in the last 6 months: 7.41 Gb/s. There are two internal networks, the campus network and the data centre network. There's dedicated links for the experiments, some sort of P2P links. There's wireless networking and cabled networking. But if there's a problem with the cabled network, the wireless network will be affected, too. The architecture will be upgraded to make the two independent and to bring a number of other improvements. IPv4/IPv6 will be supported. Another advantage of this new architecture is physical division, resulting in easier management and debugging. It's planned for August this year.

The goal with IPv6 is to provide dual-stack at IHEP. They've got perfSONAR for IPv6 ready. Firewall and switches also support IPv6. They now add AAAA records to DNSv4. Status was monitored and functional tests run to show that it works fine. Transfers over SRM/GSIFTP are successful, too. The problem is that there is no IPv6-only DNS at IHEP. A static IPv6 address is used to perform the tests. DHCPv6 is under investigation. They're planning to test commercial DNS solutions, perform some integration with Puppet and test dual-stack or IPv6-only in cloud environments.

They're thinking of setting up monitoring by selecting a collection point in each network zone and deploy perfSONAR. They also measure latency and bandwidth and plan to set up a warning mechanism via e-mail.

They're deploying SDN at the WAN level as well as at the data centre level, which was presented in previous HEPiXes. They wish to set up an elastic network in cloud environments, involving distributed data storage, load monitoring and load balancing by using different weight parameters. They perform traffic analytics with sFlow/NetFlow and transfer the results to the SDN controller. Results can be transferred through a REST API. Network security is another SDN application and Shan's team investigated on ARP attack detection and prevention. They use OpenDaylight as SDN controller. She described the algorithm based on the source MAC address in the Ethernet header and the sender's MAC address in ARP messages, as well as the number of ARP messages.

Questions and comments: You're upgrading to SDN to centralise your network into central switches. Did you perform tests? – Yes, we have a test bed.

Deployment of IPv6-only CPU on WLCG – an update from the HEPiX IPv6 Working Group (Andrea Sciabà)

The HEPiX IPv6 WG holds meetings on a monthly basis. All LHC experiments take part, T1s and T2s. More sites are welcome, especially from the US. Technical issues and progress are discussed. Andrea showed a dashboard of FTS transfers displaying some statistics.

The plan is to have IPv6-only CPUs. T1s must provide dual-stack storage. Most FTS servers should be dual-stack by 1 April 2018 (and it is already mostly the case). There's a number of other requirements foreseen for other services. ALICE central services have been dual-stack for more than a year. The ATLAS computing infrastructure already supports IPv6-only CPUs. CMS services are being upgraded to dual-stack. Central LHCb services fully support dual-stack and IPv6-only CPUs.

At CERN, the latest version of EOS was already validated for IPv6. CVMFS was fully tested to work with IPv6. CASTOR won't support it. Almost all T1s have a good IPv6 adoption. Andrea reviewed each T1 and the extent of their IPv6 support. IPv6 connectivity is tested with perfSONAR, making it easy to identify compliant meshes. There isn't much information about T2s, apart from Brunel having IPv6-only CPUs, QMUL a mix of IPv6-only and IPv4 nodes.

There are some issues with DPM, recently identified by Ulf. Transfers from NDGF (dual-stack) to Marseilles (IPv4) via FTS3 (dual-stack) fail. It's due to the way GridFTP redirection is implemented in DPM. It's got a low impact on WLCG sites because the redirection normally happens via SRM. Ulf keeps a page listing certification CRLs supporting IPv6.

The priority is now to push T2s to deploy IPv6. The common understanding is that we need good documentation for those sites without expertise. The WG started collecting documentation. This documentation is not yet finalised but something should be there in time for the WLCG workshop in Manchester.

The ETF IPv6 instance provides dual-stack testing as part of monitoring. It's designed to work for all experiments, even if it's not configured for all of them yet. It uses experiment production topologies and will help sites understand status and availabilities of IPv6 resources. The instance can be added to the central ETF.

Questions and comments:

- On the FTS transfers dashboard, do T2s seem to be driving transfers? – No, actually the plot should be sorting them by size instead.
- Is the FTS dashboard showing production traffic? – Yes.

KEK Computer security update (Tadashi Murakami)

At the previous HEPiX, Tadashi had presented KEK-CSIRT. In this talk, he introduced the KEK security infrastructure, in the context of recent events. This security infrastructure is part of a lease contract. They are reviewing the rental conditions. The infrastructure comprises firewalls, IDS, a security operation centre (for traceability) and a vulnerability management system. They provide services to users inside KEK.

They had introduced a common firewall in the zone boundary in 2002 and now introduced other ones to keep wired networks separate. There are other internal firewalls, too. They perform URL filtering as a proactive measure, from a list of malicious URLs supplied by the firewall vendor on a daily basis. They are able to add URLs and IPs manually themselves and have done so for over 500 of them. They believe that it makes a difference. KEK wish to hear how other sites do it.

They mirror packets and monitor with IDS. They suffered from too many alerts. Now they resort to a commercial analysis service. They run 350 hosts (including grid services) in their DMZ networks. They use a vulnerability management device, enabling them to find vulnerabilities and score them, sending out an alert depending on the score. They developed a DMZ user portal, which host administrators use themselves, check their hosts and submit reports reviewed by their security management committee. Scores have gradually been decreasing. User-based quality management seems successful, especially thanks to the portal.

They were faced with a ransomware incident. The vector used was targeted e-mails and one of them was opened. Luckily, it occurred during the long summer shutdown and the network was isolated thanks to the firewall. This confirmed that network separation is important. They perform statistics to fight against

targeted mail attacks. Some mails are very sophisticated, written in good Japanese, apparently coming from well-known companies. KEK organise an annual user education drill, with mails sent twice in a few weeks' interval to about 1 600 users of PostKEK. It's fair to say that the open rate itself isn't as important as the contents. Tadashi showed some incident statistics. The number of incidents is kept relatively small thanks to the efforts of the security management committee, although there's a common demand in Japan of zero risk. Some serious incidents occur sometimes, nevertheless.

Inside Japan, they actively exchange experiences with corporations, universities and other research institutes. They hold meetings, join workshops and are in touch with security staff. They're starting to do the same on an international level.

Their future plans are to carry on working on network separation and improve monitoring, logging and control. They will also set up a separation for wireless networking, enforce short-term connection times for visitors and exclude them from the KEK intranet. ACLs will be applied depending on users roles.

Questions and comments:

- In answer to how other sites set up their firewall with URL filtering, CERN are discussing with the network team for automatically blocking URLs.

Building and operating a large scale Security Operations Center (Liviu Vâlsan)

The Security Operations Centre is a centralised system for detecting, containing, addressing IT threats. It's a unified platform with multiple data access points and using multiple view patterns. There's a command-line interface for easy scripting. It's designed as a modular architecture. Most importantly, there's unified data access control policies.

The main source of data are IDS systems. There are also logs being transferred, something ensured by Puppet modules. The sessions used, process information and some other data is sent. Other sources are controlled by other groups in the department. Flume normalises and transfers the data. What's important is to have the same format as the original one may differ from source to source. There's another Flume gateway down the line, to make the bridge between Kafka and HDFS or Elasticsearch (where they keep data for the last 3 months). Kafka, used jointly with Spark, performs fast data enrichment, aggregation, correlation, whereas the Spark instance behind HDFS takes the time to go into more details. The alerting component is yet to come.

Log collection is scaling up. They started with Bro which is completely under the security team's control. They gather 1 TB of logs/day. They're currently adding syslog data. Other third-party logs will be added later on. Kafka and Flume are used for transport, the former as a central data backbone. They're currently moving to the centralised Kafka service, which wasn't there when they originally designed the architecture.

They perform some threat intelligence with MISP as the only platform. CERN run 3 instances: the main one with close to half a million IoCs, the WLCG central one with more than 160k IoCs and the development one for MISP development, as CERN is an active contributor. Detection is carried out at the network level with Bro and Snort. Detection in logs is carried out with Spark streaming for processing of data quickly. At the previous HEPiX, the security team covered the network traffic aggregator and splitter. It's worth saying that the central router is configured to only mirror traffic.

Lightweight enrichment is not necessarily 100% accurate, something which is taken into account. It's most of the time accurate, however. Additional enrichment is carried out for malicious activity only, adding more accuracy – aiming for 100%. This is a backlog process which has more time to perform. It's used for aggregation and correlations. Given the size of the data processed, they hit bottlenecks.

Liviu showed a piece of code from Flume, asking the audience what could go wrong with it. It's a Flume source to monitor a file to collect data as it gets modified. It reads data character by character, which won't scale. It's been since then fixed, now doing buffering. They use Cloudera Hadoop which is a bit behind.

The WLCG SOC WG mandates to investigate different models for SOCs and advises the WLCG sites on best practices. It was initially focused around the development of a minimum viable product – Bro IDS & MISP. Additional components to add to the stack will be investigated next.

Questions and comments:

- What size has the Bro cluster got? – 16 nodes (not all of them used, 8 should be enough).
- A lighter-weight solution might be desirable for smaller sites, e.g. OSSEC. What are your suggestions? – Please join the working group, whose initial purpose is to provide recipes. For smaller sites, an idea could be to come up with a perfSONAR-like solution.
- Do you keep all data? – We need to keep as much as possible in case of incidents, to be able to go back in time. We nominally keep one year. We had security incidents which were older than one year and it was problematic. More data helps to get an overview but we need to draw a line somewhere.
- Is this one-year limit there for complying with a law? – No, it's a choice we made.

Wednesday 26 April 2017

Storage & Filesystems

CERN IT-Storage Strategy Outlook (Luca Mascetti and Julien Leduc)

Luca presented the various services their group run – EOS, CASTOR, … The idea is to build a uniform storage layer. There's a need to scale at the exabyte scale. It involves EOS disk technologies and a tape back-end. There's also a need for a generic home directory service for users who need to share data between them – something to be implemented with EOS and CERNBox. The key of this development is based on EOS. The filesystem interface had been improved over the past few years and efforts need to go forwards. The number of files is increasing quickly, again emphasising the need for scale. The challenges EOS is facing has to do with remote access APIs and a metadata scale-up. CERNBox is interesting in this respect. It also integrates analysis software, Office Online, etc.

Julien took over presenting the evolution of data archiving. CERN now collect 100 PB/year. There are greater challenges ahead, with the need for data preservation and the fact that tapes are becoming more sensitive to dust. Some of the data is 40 years old. The team are now anticipating the future, expecting 150 PB/year after 2021. EOS is the central strategic platform for CERN, replacing AFS and other services. Tape is the strategic long-term archive medium. The idea is now to streamline the interface from experiment to tape. CTA is a tape back-end for EOS, a separate infrastructure taking care of all the scaling, where we used to rely on staging which was more complex. There's a clear separation between disk and tape. EOS and CTA act as a drop-in replacement for CASTOR. As the tape format is the same, migration should be seamless. CTA should first be released mid-2017, operational for small experiments mid-2018, ready for all of them end 2018. CTA will be available everywhere EOS is already used. It can work from behind another disk system – for which the CERN storage team would ask for help.

They're trying to reduce the backup volume, which they're moving to CASTOR, i.e. eventually to CTA. Since they currently use a proprietary IBM solution, this migration seems advisable.

Questions and comments:

- The market for tape systems is shrinking. What's plan B? – Tape isn't going away just yet. For instance Amazon and Google still use it, with LTO technology. Julien is confident that for that reason it will stay for some time still. But they keep looking out for alternatives anyway.
- Which NFS do you use? And why use NFS anyway? – This is for Oracle, which requires NFS. It's not needed for other services.
- Wouldn't you prefer per-server licence costs for TSM, instead of per-volume? We have some experience that it's much cheaper. – The road map linked to the client-based licence wasn't clear. And we have more than 1 000 clients. We prefer monitoring packet consumption and try to contain it.

EOS and CERNBox Update (Luca Mascetti)

The EOS project started in 2010 – it's still a young one compared to some other projects. It's got a free licence, offers a simple and scalable solution. It's secure and supports Kerberos. They use network RAID and it's possible to use chunks across disks. Release cycles are short, allowing for agile development. It's got 165 PB's worth of space, comprising LHC, physics data and CERNBox. The third link to Wigner was felt as an improvement by our users. EOS is split inside CERN for easier management. There's an instance for LHC experiments on the one hand and end-users on the other hand. Users come from all over the world. The CERNBox offering is 1 TB/user. It's easy to synchronise files, share files with other users or groups (CERN e-groups). The popularity is rapidly increasing, with 60 new users/week. There once was a workshop on statistics which caused a sudden bump, from new users wanting to use Jupyter notebooks.

The client is not just a FUSE-mount and it's not using the ownCloud PHP code, but it directly talks to EOS for the sake of performance. A Samba gateway was recently added for Windows clients. CERNBox should be the first entry point, which is why it's integrated with other services such as Microsoft Office 365,

Jupyter notebooks, … and more is to come. Uniform hardware is used to make scaling out easier. The plan is to pack more trays per server to pull the costs per GB down.

The team are working on a generic home directory service. Linux machines have AFS, Windows DFS, some more private clusters use NFS. The idea is to federate everything with EOS and CERNBox, also in light of the slow AFS phaseout. The idea is to present the same home directory on Windows, Macs, on lxplus. The purpose is to avoid running small storage clusters too, improving also the synergy with DFS. There's a lot of development effort needed in the EOS FUSE client. The team are working on cache performance and fast synchronisations. There were several implementation versions already improving speed by several orders of magnitude. A server upgrade will help, too. The migration should be complete by the LHC Run 3.

The flexibility offered by EOS makes it an interesting solution. It's the largest low-cost HEP storage installation site today. On top of EOS, CERNBox is a natural extension of the desktop, offering new ways to interact with data.

Questions and comments:

- Is `http://cernbox.cern.ch` IPv4-only? – Yes. There is no strict need for IPv6 at the minute. The move should be easy.
- When we use EOS for home directories, can we run checksums, create snapshots, enforce quotas? – Yes. In fact checksums were not possible with AFS. 10 versions of each file are currently kept. There is a quota system.
- AFS has been successful in HEP in the past, and many site have AFS servers. How do you see EOS in this perspective? – We need to understand the traffic with respect to user needs. We don't see any problem as such but that will depend on the site itself. EOS is opened to everybody.
- It's not clear how authentication works. Do CERNBox, lxplus, Windows machines share the same mechanism? – Yes, it's Active Directory.
- Home directories typically follow POSIX. What are plans for applications that are inode-specific? – AFS was actually not totally POSIX-compliant. New implementations will follow POSIX more closely.
- FUSE performs lots of context switches, a cause of serious performance degradations compared to NFS. Has this problem gone away? – For the current kernel, the real problem is the pipe at the FUSE level which seems to be improving.
- How do file permissions and ACLs work with FUSE? – The CERNBox web interface, which works with e-groups and accounts, lets you adjust permissions and ACLs. A command-line interface is coming up, too.
- How does POSIX compliance agree with Windows permissions? – We checked what users need and the two seem to work well together.
- Windows has a tendency of leaving junk files behind. – Yes. Yet, users don't seem to be bothered by this.

BNL Box (Hironori Ito)

Hironori described BNL's attempt at cloud storage. The idea is to use it as a way to transfer files between different systems and hosts. System administrators need to do so for software installations, too. Again, this started from the upcoming AFS phaseout. What's more, AFS isn't always that easy to use, e.g. on Windows. They considered commercial cloud storage, but the cost, performance, archival facilities put them off. They're not talking about the worldwide audience, but only BNL users. There are other science domains at BNL who could use a good cloud storage solution, too. They have different use cases than the ones we've got in HEP. The usage targets small and large data transfers, access to computing farms and archival.

So they came up with the BNL Box prototype. It's based on ownCloud. Clients are available for a large variety of OSs. It's very easy to use. They use Ceph storage as back-end, currently Ceph Infernalis, targeting Ceph Kraken. They like the reliability, scalability and performance (40 Gb/s for BNL Box). They consider using a tape restore feature with a dCache HPSS tape archive. They wish to offer an XRootD interface and WebDAV access, too. The default synchronisation application is enough for small data. But a higher volume of the order of 10 TB could be required for some use cases – with the ownCloud support of the standard WebDAV protocol, 150 MB/s transfers can easily be achieved. And concurrent transfers will result in higher throughputs still. There's a desire to keep data synchronisation operations from data read accesses. What they did was to use XRootD to map user data in BNL Box in a simple way, involving omitting the username from the URL pointing to a file.

Some users requested to be able to archive data. The first question is, how often they'll need to read back data, the issue being that mounting and seeking is slow. Archives also work best with large files and small files will be tarred up and only copied to tape when the tar files reach a certain minimum size. Index files will be created to help users find their archives.

Users can decide what to share, protect files or directories with a password and make them completely public, setting a deadline after which sharing is automatically disabled. Not all files need synchronising, which can be useful depending on the available size on a client device. These settings can be adjusted with the provided application.

Questions and comments:

- Was there any problem related to using CephFS as back-end? – It's been in production now since last summer, and we haven't seen any problem, although it's fair to say that it hasn't been heavily used yet.
- How to share files in BNL with e.g. CERN? – ownCloud seems to provide a way to federate between sites.
- Why use Kraken instead of a longer-term release? – Because we wanted to see how far we could push our prototype.
- How does ownCloud interact with CephFS? – CephFS is mounted where ownCloud services run.
- Do you plan to provide direct access to CephFS to users? – Let's wait and see. Technically, nothing prevents us from doing so.

Federated data storage system prototype for LHC experiments and data intensive science (Andrey Kirianov)

A year and half ago, NRC got the green light to evaluate federated data storage technologies. The goal is to create a storage federation which is geographically distributed. There should be a single entry point and the federation should be able to scale and make it easy to add new resources. Transfers should be optimised, stable and fault-tolerant. They initially chose EOS (because users use XRoot) and dCache (which has become very popular). The test bed comprise 5 centres in Moscow, 2 in Sankt Petersburg, one at CERN, one at DESY-Hamburg. This isn't supposed to be production quality.

They had no experience with EOS in the beginning. They started with a small EOS test bed. They didn't want to only run synthetic tests but also run production payload. The authentication is based on X.509 – no Kerberos. Now they know how to deploy new sources. They chose a number of tests they were familiar with, testing local filesystems to evaluate the EOS FUSE mount. They also used a pure XRootD test (`xrdstress`), another one specific to ATLAS and one to ALICE. The software component was CentOS 6 (CentOS 7 wasn't available at the time). They deployed perfSONAR. As all the links are shared, some sites in Russia have complicated connections for historical reasons and they weren't sure links between 2 sites were always OK. They wanted to make sure of this before running tests.

In the beginning, they started with Bonnie++. What they wanted to check in a federated scenario is that the metadata management and stream transfers worked OK. Then they wanted to try ATLAS and ALICE software. It turns out different experiment software have very different ways of accessing data. ALICE performed much better with pure XRoot than FUSE mount. It was the other way around for ATLAS.

But federated storage otherwise worked as expected. There were some problems with EOS, e.g. the synchronisation between 2 masters.

Then they wanted to check data placement policies. Different data types have different preciousness, affecting the number of replicas. With EOS there are three scenarios: a random distribution of data, data located as close to the client as possible (EOS support geotags, with a more advanced support, recently), data located as close to the client as possible with replication. They performed a data population performance test from CERN.

The ALICE test showed the funny result that for the SINP institute, reading from remote storage was faster than from local storage. This showed the importance of deploying an optimal federation, making sure the infrastructure is fine in all sites, otherwise keeping data close to the client may make no sense.

The first experience with XRoot was very positive. They had to make sure it wasn't the only software that performed adequately, so they looked into dCache, too. It's completely written in Java. It's got an implementation of its own for transfer protocols. The dCache XRootD implementation doesn't support FUSE mounts, something which may be fixed from a collaboration between JINR and DESY. There's no built-in security, whereas EOS offers authentication out of the box – a very strong point. There is no built-in manager redundancy either.

Andrey compared the performance of EOS and dCache – there was no striking difference. Both can be used in the federation, as both work reasonably well. Looking at the ATLAS tests, EOS was consistently scalable in performance; dCache performance goes up and down but is also very fast overall.

Questions and comments:

In fact dCache *does* support multiple managers since version 3 – Yes, the older version we tested (2.16) didn't yet.

RAL Tier-1 Evolution as a Global CernVM-FS Service Provider (Catalin Condurache)

CVMFS is a globally-distributed filesystem optimised for software distribution, built using standard technologies. It needs a single installation and is mounted into the universal `/cvmfs` mount point. Catalin described the history of CVMFS at RAL. Today, RAL's CVMFS infrastructure offers 32 repositories (780 GB) for the Stratum 0 service and a Stratum 1 offering 65 repositories (16 TB). The CVMFS Uploader service is an in-house implementation providing an upload area for `egi.eu` and `gridpp.ac.uk` repositories. About 60 people upload the files via a GSI interface, then rsync'd to the Stratum 0 at RAL, before being replicated. There's an acceptable rate of traffic and number of requests, between 400-500 GB/day.

Recently, they protected CVMFS repositories. Normally, they're designed to be public with unauthenticated access. Some users want to distribute licensed software. As a result of some work done within OSG, managing authorisation and authentication with X.509 proxy certificates became possible. They have a working prototype at RAL. The Stratum 0 has `mod_gridsite` and `https` enabled and clients connect to Stratum 0 directly. The cloud environment was a good place to start as VMs are instantiated in various places. Another good use case was that of worker nodes. The West-Life (H2020) biology project was the first use case at STFC. They deployed VMs in various places. Downloading a valid proxy was the way forward to access the Stratum 0 via HTTPS.

In new developments, the new `CVMFS_CONFIG_REPOSITORY` environment variable can be used to centrally maintain public keys and repository configuration. The `africa-grid.org` namespace was added. IPv4/IPv6 dual-stack is coming up. CVMFS now supports Web Proxy Auto Discovery (useful e.g. when users who don't know what the local Squid is).

An update to Ceph at RAL (Tom Byrne)

RAL presented their work on Ceph in several previous HEPiXes. They've got two main clusters: Sirius (600 TB) and Echo (9.9 PB). Echo emphasises more on TB/£ cost economy. The main change since last time is that they're now accepting production data. They provide GridFTP and XRootD access. They pledged 7.1 PB to WLCG. It's a similar level of usage than with their CASTOR instance.

There were operational issues. In February, they've been through a routine reboot of nodes. One node didn't come back up in a healthy state and got stuck. There were 2000 files affected by this. It should have

been as simple as restarting the OSDs. But in this case, there seemed to be a more serious problem – two OSDs couldn't talk to each other. There were jobs failing in the meantime. It was decided to recreate the placement group manually. Purging it from the set was revealing: one of the original OSDs failing to peer seemed to have corruption trying to access the LevelDB and was reformatting the OSD. Recovering from this kind of situations with Ceph has always been a concern. Data was ultimately lost, which could have been avoided if the problem had been identified earlier on, before they manually removed the placement group from the set.

When they started looking at Ceph, CephFS wasn't production-ready (there still are scaling concerns). Ceph provides an S3 server but they needed GridFTP and XRootD. The XRootD plugin was developed at CERN. The problem they faced is that direct I/O was slow, something they addressed at the caching level. There were other bugs which have since then been fixed. The GridFTP plugin was started at CERN, completed by STFC. They improved the plugin, solving transfer time-outs, adding check-summing and multi-streamed transfers. They used grid map files for authentication. Instead of having worker nodes talk to gateways, they turned worker nodes to gateways, running an XRootD gateway inside containers.

Tom presented his team's plans. They believe that they should be supporting S3/Swift. They're expecting that most new users will need help with this. They've been looking into setting up a DynaFed, which they believe to be the best tool for allowing small VOs to access data securely. It provides a filesystem-like structure and supports transfers to existing Grid storage. The DynaFed service is set up behind a high availability proxy. It will be moved to production in 6 months.

Questions and comments:

- Owen asked, how much does turning worker nodes into gateways increase performance? – We haven't got to the point of doing load tests. We didn't try with a significant number of worker nodes. It should allow you to get an awful lot more bandwidth.
- Regarding the use of S3/Swift for future VOs, are there any coming VOs wanting file stores instead of object stores? – One of the nice things about DynaFed is that it provides a filesystem-like interface. It's not POSIX, but good enough if you have a directory-based workflow.

Data-NG: A distributed Ceph infrastructure (Guillaume Philippon)

Guillaume presented his team's work for 8 physics laboratories from Orsay/Saclay. They've got 3 hosting facilities. All laboratories manage a large storage infrastructure, with 7 PB distributed over all sites – it's becoming harder to provide efficient access. Most sites have built either a test or production Ceph infrastructure.

Agata is a nuclear physics experiment. They use Ceph for acquiring data – 300 TB. It's too small an instance to justify a resilient infrastructure. Another instance they have is for their OpenStack persistent volume service (Cinder). They've got 4 OSD servers, 3 monitors, which seems to be a minimum for such an infrastructure. They use Ceph for Proxmox too and have a test bed at CEA, where they perform benchmarking. They experienced that Ceph can make good use of large infrastructures. Yet, they want to avoid duplicating infrastructures to optimise manpower as running Ceph is a common effort from non-dedicated people.

The Data-NG project is about building a 1 PB storage infrastructure which is scalable, featuring resiliency and efficiency. Efficiency also in terms of power supply: if a site is down, another one can take over. They don't wish to provide storage services. There's no backup done. The initial use case will be to move their OpenStack Cinder there. They wish to provide distributed data storage for nuclear physics analysis, astrophysics data and data acquisition for P2IO facilities (e.g. accelerator projects). An on-demand Spark infrastructure is another use case – they're wondering if they can access data directly via HDFS or by using S3). They may also consider setting up a resilient unified data back-end for GRIF grid storage.

They chose Ceph because they want to have different replication policies, as experiments have different needs. Replication will be across sites. They wish to use a homogeneous hardware configuration on the 3 sites at first. They won't do journaling on SSDs, as they're only interested in the high performance of the pool. They'll use Dell R730XD servers for OSD and basic Dell R630 ones for monitors with no special requirements in terms of memory and CPU.

It's not so much a technical challenge than a human one. The instances will be managed by different laboratories, with different policies, and preferences for different configuration management systems. Up next, they will order hardware, run the first tests, and upgrade the network between sites to 100 Gb/s.

Questions and comments:

- You said different laboratories use different configuration management systems. Will every site manage hardware with their own? – It isn't clear yet. They will each have their own tools to install hardware, certainly. We'll converge to a common set of tools later on.
- You'll likely see that the bottleneck will be Ceph more than the SSDs. You may want to assign SSD nodes to metadata.
- As soon as you'll start using tiering, RAM usage will go up and you might become tight on memory.

Site Reports

PIC Report (Jose Flix Molina)

Pepe presented recent developments in his site: they reached 75 kHS06, 7.1 PB on disk, 16.8 PB on tape. They made recent purchases (10.8 kHS06's worth of Dell servers to immerse in oil). They also acquired 1 728 TB of disks, 843 T10K cartridges. All the new CPU servers they buy are to be immersed in oil, in large tanks. They migrated away from IBM LTO tapes as they retired the IBM robot. They migrated LHC data to STK tapes instead. The number of slots occupied decreased by moving to denser technologies. The space allocated (now in a single library) is going up.

Recent tape staging tests from ATLAS and CMS recently took place at T1s. They resulted in good rates at PIC. But they noticed that there wasn't much load from CMS, as data retrieval from tape systems is rather inefficient. There is some work to be done in this area.

They upgraded their WAN to 20 Gb/s. They're testing many services against IPv6. For instance, they have an IPv6 dCache instance. They've got several production services running in dual-stack. They also have worker nodes running IPv6, over 50% of them in dual-stack. HTCondor is being deployed in production (10%). They developed APEL accounting for HTCondor-CE, now pushed in production. It's being validated by APEL experts. PIC are active members of the HEPiX CPU Benchmarking WG.

Questions and comments:

- How do you upgrade production servers when they're immersed in oil? – Before immersing them, we take out the fans. We can't carry out interventions ourselves and we leave it to the vendor. Manipulating the machines is less comfortable than when they're mounted in racks, certainly. On failures, we bring in the vendor. We accumulate several failures before planning an intervention. You can't immerse storage servers, as disks need replacing more often.
- On the IPv6 work, how do you track failure rates? – We run monitoring and we noticed there aren't too many. We tested transfers and didn't notice more failures with IPv6 than with IPv4.

Computing & Batch Services

CosmoHub on Hadoop: a web portal to analyze and distribute massive cosmological data (Jordi Casals Hernandez)

CosmoHub lets you register galaxies with positions, colours, etc. and perform analytics on them. Jordi's team started running it with a PostgreSQL back-end. Data was growing too fast for the database to scale. So they moved to Hadoop with Hive to query the data with an SQL-like language. They use Tez as an alternative to MapReduce. They have input files, which are distributed across nodes. Hardware demands from PostgreSQL to Hadoop dropped (even though the number of nodes increased, but they could use old servers). The speed-up is considerable (up to 100× faster). It even enabled them to perform interactive

catalogue analysis. They've got 450 users using the platform, 1 500 custom catalogues, 6 TB of hosted data, 10^{10} objects.

Jordi gave a demonstration of the CosmoHub. He selected some columns he wanted to work with, chose a random sample and built a heat map in a matter of 2 minutes, with which he represented the Milky Way. It's easy to download the data in a variety of formats, save plots to PNG or even download the whole catalogue – a long process whose progress can be monitored and notified by mail. There's also a convenient list view to track running and completed queries. With the speed improvements brought by Hadoop, they're willing to explore more of the Hadoop ecosystem and find use cases other than cosmology.

Questions and comments:

- Can anybody access cosmology data? – There's a public catalogue made available to play with, without having to belong to any affiliation.
- Did you consider qserv as an alternative to Hadoop?
- Which format do you use for storing data in Hadoop? Is it Parquet? – We use ORC files (initially CSV).

HammerCloud extension for Data Centre commissioning (Jaroslava Schovancova)

They test services by submitting jobs, on demand or continuously, at a lower frequency. Reporting comes with summaries. They run >180k jobs/day. This saves manpower. Automatic actions can be programmed. HammerCloud is used to exclude and recover sites. The tests are used by site administrators or LHC experiments. The web interface is a Django application. MySQL is the data back-end. The application provides monitoring views. The testing infrastructure is based on OpenStack VMs. Originally, HammerCloud was based on Ganga for submissions. Now a combination of Celery and Redis is used, opening doors for using containers.

Originally, HammerCloud could only submit via submission frameworks. Now the ability to submit to HTCondor-CE has been added, to test resources not yet commissioned to experiments. It allows to run benchmark tests and check capabilities, too. Next steps will entail broadening the family of available workflows and performing analytics. In the further future, better packaging could help you run your own instance of HammerCloud. The ultimate goal is to set up ever more realistic experiment workflows.

Experiences With Intel Knights Landing, OmniPath and Slurm (William Strecker-Kellogg)

They set up a 142 nodes Knights Landing (KNL) cluster, running 64 physical/256 logical cores per node. This makes it a many-core architecture. KNL excels at e.g. prefetching, improving pipe-lining if you code it appropriately. It uses MCDRAM. It operates in different modes (flat, cache or hybrid). You can change the way memory is mapped to the die. The OmniPath (OPA) Interconnect is an aggressive competitor against Mellanox and InfiniBand.

The first thing they hit when trying to run code is that it limited bandwidth for MPI jobs under some circumstances. The LQCD solution of using 4 MPI ranks/node helped. There is a Fabric Manager on managed switches. GPFS storage exists for Institutional Cluster and integrates with the OPA fabric. The aggregate performance is of the order of 20 GB/s through gateways. Initial assembly revealed that a few CPUs needing replacing. Several BIOS updates were deployed. The system image provided performed as expected. BNL's RHEL 7.2 image showed a 10-20% performance reductions. RHEL 7.3 brought the performance back to nominal levels. They use Kickstart and Puppet for provisioning. They were provided with tarballs where an `INSTALL` script had to be run. Switching cache modes dynamically is supported by SLURM. But this proved to be unstable. Intel later confirmed that a power cycle between mode switches is necessary. A BIOS update should fix this.

The team put together SLURM monitoring, custom scripts dumping data into their Graphite instance, which Grafana displays. The data collections takes place using PySlurm bindings and parsing the output of CLI tools. They're welcoming a discussion on LQCD-wide monitoring, possibly to come up with something similar to Fifemon.

Questions and comments:

- Michele expressed his surprise that HEP-SPEC could run on the KNL cluster.
- Make sure that running with many threads doesn't cause swapping. – Yes, in fact the system is configured with very little swap.
- Did you test the Intel compiler? – Not for HEP-SPEC.

JLab's SciPhi-XVI Knights Landing Cluster Update (Sandy Philpott, remotely)

At the previous HEPiX, Sandy presented the Knights Landing (KNL) cluster they installed. Since then, they made it to rank #397 on TOP500. After installing racks, they had to relocate them. Users reported the system hanging and MCDRAM flat/cache reboots were necessary. They almost scored 426 TFLOPS and made it to #10 on Green500. Their systems amounts to >16k cores, with >50 TB of memory. They run CentOS 7.2. Sandy's team went through changing nodes with a reboot to update the BIOS in order to set MCDRAM for flat or cache mode per user tag. They've got OmniPath, a single interface. You need more than a single core to drive OmniPath. The black box on the diagram she showed is the only OmniPath they've got in their fabric. OmniPath nodes need to reach Lustre and NFS file services on the InfiniBand QDR fabric. The team still wish to benchmark users' USQCD codes to compare performance with conventional code. The test bed will be moved into production by the end of June, with the software and BIOS updated. They intend to automate cache and flat memory reboots based on job demands. They plan to investigate the Zonesort kernel module and the Intel Cluster Checker tool.

Updates from HEPiX Benchmarking Working Group (Domenico Giordano)

The mandate of the working group is to investigate scaling issues between HS06 and CPU-intensive HEP workloads, study the next generation of long-running benchmarks and evaluate fast benchmarks. They have recently reported their latest work in GDB meetings. There is the debate around adopting fast benchmarks. There's a consensus that experts want to include fast benchmarks and get real-time information. HS06 is getting quite old and we're still waiting for a new version. There were 5 fast-benchmark candidates in the beginning. Systematic studies converged towards DB12 and ATLAS KV (Kit Validation). Domenico showed a correlation between KV and DB12 from ATLAS simulation jobs. LHCb showed that DB12 performed better than HS06. ALICE saw good correlation between DB12 and Monte Carlo, but a large discrepancy with respect to HS06.

CMS have been working on instrumenting pilots to run fast benchmarks and collect results. Similarly, ATLAS prepared HammerCloud reference jobs (single and multi-core) running on benchmarked resources. The working group benchmarked Haswell servers in virtual environments with VMs running various number of cores. Given the diverging results on DB12, they decided to continue studies by performing application profiling, checking the reproducibility under different Python versions and the effect of using different implementations (C++ or Python NumPy). Processes in DB12 are spawn by the `multiprocessing` module. Each process loops around a random number generation. It is assumed that DB12 represents Monte Carlo jobs well because it is mostly dominated by random number generation, as are Monte Carlo jobs.

There are two other implementations of DB12 available: an optimised version with NumPy, and a C++ implementation. Profiling studies showed that these two versions were dominated by calls to the `math` and `rand` modules. Manfred Alef compared the benchmark on different hardware models. The C++ and NumPy DB12 versions scale better than the original DB12 Python script.

In other studies, the use of real jobs to measure the relative speeds of different CPU models was investigated. This was tried at the ATLAS T0, mainly with reconstruction jobs. The plan is now to start drafting requirements to validate the successor of HS06. Several effects (such as differences between VMs and physical nodes) remain to be disentangled. A test bed with a representative set of hardware models must be defined.

The scheduling strategy and experience of IHEP HTCondor Cluster (Jingyan Shi)

They migrated to HTCondor away from PBS because they suffered from the limited scalability. They found the active community attractive, too. The migration involved a few milestones with risk control. With 28 submitting nodes, 2 schedulers, 2 central managers and 10k cores, they can process 100k jobs/day. Most of them are serial and single-core jobs. Jingyan's team support several HEP experiments, such as BES, Daya Bay, Juno, Lhaaso, HXMT. They used to have resource separation which they wanted to break to reduce idle capacity.

They set up their scheduling to guarantee fairshare. Their resource sharing is based on job slots. Some are partly shared, some kept exclusive to specific experiments. In their shared resource pool, at least 20% of the slots are shared by each experiment. The preference is to run jobs on exclusive slots of their respective experiments. Shared slots are kept for busy experiments. Experiments are aggregated into UNIX groups. An initial quota is set, but it can be exceeded if there are idle slots. They put some error detection and recovery in place. Health status is collected and stored in a central database. Central controllers update worker attributes regularly.

IHEP developed the `hep_job` toolkit, to help users migrate from PBS to HTCondor. It helped the team implement their scheduling strategy. It's based on the HTCondor Python API and integrated in the IHEP computing platform. Nagios and Ganglia are used for monitoring. Detailed accounting information for each group and user is collected, weighing slots according to their CPU speed, memory size, disk space and a number of other parameters.

They faced a problem with a dishonest user, incorrectly claiming he belonged to a group. He started SSH daemons on worker nodes and ran MPI tasks. He occupied more CPU cores than advertised. As a result, IHEP added a group priority check at the worker node level. Zombie processes are also now checked. Another problem they had was hanging jobs, as a result of scheduler daemons losing connectivity for a short period of time. The reason was due to the default limit on the number of open files. Finally, they found themselves in a situation where the `condor_sched` owner changed. This was due to a disk mounted in the scheduler which became inaccessible. Disk checks are now performed.

Concerning future work, they plan to automatically tune the resource sharing ratio according to the overloads of each group, integrate job monitoring and the central controller, and come up with a union of HTCondor sites because of their little capacity and small manpower.

- Did you consider enabling cgroups to reduce the risk of having dishonest users? – They considered this too, and Jingyan requested advice on how to do so.
- The newest version of HTCondor adds server-side requirements to prevent dishonest users.
- At BNL, they had to stop dishonest users too with policies.

The search for new traceability and isolation approaches (Brian Bockelman)

WLCG experiments have heavily used the multi-user pilot job model. We need isolation so user payloads cannot interact with each other or with the pilot. gLExec is how we do traceability so sites can identify who uses a given computing resource at any time. gLExec never was popular, possibly because our environment wasn't built with user switching in mind. And it requires configuration files – system administrators are very unresponsive when asked to change a line.

So Brian presented a different approach, using Singularity for isolation. It's a container solution tailored for HPC. It's simple isolation, there's no daemons, no UID switching, no editing configuration files – it's just a matter of installing an RPM. There's three options when using containers: either the batch system starts the pilot inside a container, or the pilot starts each payload inside its own container, or a combination of both. They combine both in Nebraska. Brian showed a process view showing, interestingly, that Singularity runs from within Docker. From the pilot's point of view, we can't tell what the OS is, what other users there are, which other pilots are running on a given worker node. From the payload's point of view, only the payload processes themselves are visible. The OS environment must be delivered to the CMS pilot. Singularity can read from a directory. CVMFS is used to distribute a directory of software (we're welcome to take a look at `/cvmfs/singularity.opensciencegrid.org`).

Singularity doesn't provide traceability features. They want to keep the information on site. HTCondor-CE is used for this. Brian's team submitted a patch to HTCondor to be able to log the payload.

Questions and comments:

- What's the relationship between Singularity and Shifter? – Shifter is more Cray-centric. Singularity allows you to create your own images. Shifter is also more convoluted to set up for a site.
- For anyone running Singularity, you might encounter interesting problems with the auto-mounter and older kernels. If you have solutions, please contact the Jozef Stefan Institute. – We, at Nebraska, are aware of this, and we put workarounds in place.
- What goes into images? – It depends on the VO, the CMS one is small. For instance, some VOs need OS libs.
- The ATLAS experiment is looking into integrating Singularity into their workflows. In which way to run it (pilot in the container or the other way around) is still under discussion.

Understanding the performance of benchmark applications (Luca Atzori)

How to approach performance problems with real world examples? Luca presented an attempt with the Dirac Benchmark 2012 (DB12), currently investigated by the Benchmarking WG. They noticed a performance difference between Ivy Bridge and Haswell architectures. They worked with a function profiler, breaking down the execution time, then breaking down instructions, but they couldn't see any significant difference between Ivy Bridge and Haswell. They tried to have a look at basic performance counters. There seemed to be a difference in `branch-misses` – a mistake a CPU makes when trying to jump instructions. When a pipeline is running an instruction, it tries to guess the next one. If there's a mistake, everything must be flushed and started from scratch – a significant performance penalty. Another consequence is cache pollution (`L1-dcache-load-misses`). How could Haswell never make wrong predictions? There is a history table with jump addresses. If the address where a branch leads to is indexed, performance will be better – what Haswell pretty much does. The better branch prediction on Haswell was the cause of its faster performance. It's worth noting that most of the time is spent in DB12 – does this really make it a good candidate? On an unrelated note, Luca mentioned the TechLab Benchmarking Website, a collective database of extensive benchmarks.

Thursday 27 April 2017

Grid, Cloud & Virtualisation

The Computing Resource Information Catalog (Alessandro Di Girolamo)

Alessandro's team support experiments in the WLCG. There are 200 computer centres in WLCG, most of them providing different services and solutions. There's many disks, tapes, jobs and users to accomplish our mission in HEP. All four experiments have different workflow management systems. We rely on different middleware. We also have different computer resources – standard grid sites, cloud resources, opportunistic HPC resources. We want to use all the available resources. All these different bits and pieces need some underlying component to make them work together.

The information system is a big world and *word*, because people understand different things behind the name. For instance, GOCDB, where services are defined; or BDII; or REBUS, where computer centres list how much they pledged and installed. Each experiment have set up their own collectors. Inside experiments, there are lots of frameworks to accomplish our mission. These two worlds are separate from each other. Since some time ago, AGIS has been the central information system for ATLAS. With their experience, it was thought useful to extend AGIS, even start from scratch to support other experiments too: this gave way to CRIC.

They believe the information system is a key component. It doesn't matter how big the site is, especially as sites keep growing in services. There quickly comes a need to have a unified structure to advertise those services. In CRIC, a site defines the topology, e.g. how computing resources are connected to storage. The resource status information is integrated. The configuration is declared. Through a REST API, this information can be distributed. The basic, fundamental concept brought in with CRIC is a clear split between physical resources (i.e. servers) and logical resources (i.e. what the experiments, the shifter, see). For instance, perfSONAR, FTS, […] as opposed to e.g. endpoints.

The system is Django-based. It's got a modular architecture. There's a client-server model, providing a GET/POST REST API. The web portal uses AJAX (with Bootstrap). It's database-backend agnostic. There's various collectors run by crons. The system supports information protection through SSO. There are plug-ins for each experiment, describing the views the experiment needs, as each of them have different requirements. One of the challenges not yet solved is the storage. People still send mails telling which storage to use. The problem is that there's either too much or too little information. To try and address this, there's a clear separation between dynamic and static information in CRIC. An iterative approach will be needed to find the right balance. For instance, on the one hand, in the service world, they will model a storage service in which there are storage areas using different protocols; on the other hand, in the experiment world, there are attributes and collection endpoints.

Questions and comments:

- Dennis appreciated the effort. He wondered how this will work for a site. They went through the same effort. How much effort will be necessary from the site to support CRIC? – There will be some effort involved, certainly. BDII somehow didn't succeed. CRIC tries to simplify the information by decreasing the number of attributes, or avoiding too frequent updates. They might keep the BDII running, pulling information from it.

- Are you planning to add volunteer computing resources? – They do not plan to have monitoring inside CRIC, which only describes the topology. It will e.g. show backfill queues but not whether resources are being wasted or not. It's the difference between static and dynamic information.

ElastiCluster – automated deployment and scaling of computing and storage clusters on IaaS cloud infrastructures (Riccardo Murri)

The University of Zurich would not have a batch system and this is where ElastiCluster came in. It's a command-line tool. A single command should be able to deploy a fully-functional cluster. But you need to describe what you want. You can specify the cloud, authentication information, how many compute

nodes should be running, whether there is any need for SSH, which image to use. Riccardo showed a video demonstrating how the cluster is set up. Ansible is used behind the scenes, here to set up a SLURM cluster with Ganglia (which is an add-on, it can be omitted too). He then showed how to resize the cluster, which is just another ElastiCluster command. When you are done with the cluster, since it's all virtual, you can decommission it with just one command.

SLURM, Grid Engine, HTCondor all work, as do Hadoop, HDFS and other parallel filesystems. The most important feature is that ElastiCluster is agnostic to the cloud infrastructure and the operating system, as all of these aspects are delegated to Ansible. Riccardo reminded us that Ansible is a software orchestration system. It works with playbooks which we can run over and over again. It can run in a client-only mode.

There's currently a problem of scale, the setup time increasing linearly with the number of nodes. It's not clear why, and ideas from the community are welcome. To speed up the setup, they resorted to snapshots. Ansible has a conservative setting for SSH – only 5 connections at a time. Increasing this will also boost performance. Scaling depends on the number of nodes, but if you're only interested in the number of connections, you can deploy larger worker nodes.

Riccardo showed how people currently use ElastiCluster in the wild. They use it for provisioning temporary clusters, e.g. for teaching and testing. They use the scaling feature on permanently-deployed clusters to grow them to face a temporary peak and then shrink it back. Google Genomics is one of the main users of ElastiCluster. More on teaching use cases, the University of Zürich use ElastiCluster for Jupyter/Spark courses where short-lived events are necessary. ATLAS in Switzerland use it to scale their permanent cluster on SWITCHengines up and down depending on the load.

Questions and comments:

You're covering a huge variety of software. Testing it all must be very hard. How do you do it? – We rely a lot on community reports. For frequently-used clusters such as SLURM and Grid Engine, if something goes wrong, somebody will notice quickly.

CERN Cloud service update: Containers, migrations, upgrades, etc. (Luis Pigueiras)

The policy in CERN IT is to run all servers virtually. They should be based on OpenStack. It's a project that started in 2013. OpenStack is a collection of tools to manage a cloud infrastructure. They're currently halfway through Mitaka and Newton. There are two data centres, in Wigner and Geneva. There's a single OpenStack region, i.e. a single API. Cells separate compute nodes for scalability reasons. There are three types of cells: shared (across 5 availability zones), project (for special requirements) and batch (optimised for batch computing). Luis showed a picture of their architecture. There are 7k hypervisors in production with another 2k coming up. There's 220k cores with another 86k being added. They record 400 operations/min on the API and 27k VMs are running, with an increase of 5k just in the last 6 months.

Regarding operations, they upgraded from Liberty to Mitaka and will move to Newton next week. It involved a database schema upgrade, which they had to back up beforehand. They went through a validation process. The compute nodes were upgraded with Puppet and Yum. A Nova update takes 3 hours, typically. They upgraded from CC 7.2 to CC 7.3, on a cell-by-cell basis. They lost the connectivity of a few VMs in the process. They also suffered from the kernel soft lockup bugs. They're trying to run a homogeneous cloud to reduce the complexity of running the service. Neutron is the networking service. It's available in 10 cells. They're moving from nova-network to Neutron this year. On the Keystone front, no tokens need being stored in a database anymore, allowing for better performance. They started accounting for their resources, for which they use cASO and store data in S3. Reports are published on a web page. Rally is a cloud benchmarking system. They use it for performing continuous testing in their cloud. They've got a dashboard showing any problem with a red light.

Magnum is a container orchestration engine. The current release is in Newton. There's a Magnum client which interacts with the OpenStack cloud to create a cluster. GitLab CI, SWAN, the batch service, FTS are some of the use cases using Magnum at CERN. ATLAS RECAST uses Magnum for software distribution in HEP, providing reproducible development environments and continuous testing. For their future plans, they will roll out cluster upgrades, set up heterogeneous clusters, monitor containers, use them for load-balancing as a service and improve storage support.

The cloud team will provide new services soon, such as Ironic (bare metal provisioning), Mistral (workflow service), Manila to provide file shares (using CephFS as back-end) – already a pilot service. It offers off-the-shelf integration with Kubernetes and Swarm. There's a need for a highly-available filesystem to replace the NFS filer service, with whose team they're in collaboration. This is used for instance for sharing configuration files and certificates.

Questions and comments:

- Did you set up Neutron with Linux Bridge or Open vSwitch? – Linux Bridge.
- Concerning the container service based on Magnum, do they run on dedicated hardware? – They run on top of virtual machines.
- You mentioned the Ironic service. Could this be exposed to experiments to use in their central services? – Yes.

Container Orchestration – Simplifying Use of Public Clouds (Ian Collier)

Andrew decided to try a new approach. What can be done to simplify running things? On the cloud platforms, the common factor was Kubernetes, already used for running LHC jobs. The cloud providers stress loudly that we're not allowed to talk about performance. They used clouds in HEP for many years. It generally involved different efforts for provisioning resources. There were limitations, with each cloud provider offering a different API. There's not much portability either. Kubernetes is an open-source cluster manager originally deployed by Google. It provides service discovery configuration and secrets. A *pod* is the smallest deployable unit of compute. Kubernetes can be run anywhere. The idea is to use it as an abstraction layer. Why not Mesos? In earlier talks, Andrew had been focusing on it but the solution suffered from a security limitation – not ideal for running jobs from third-parties (e.g. LHC VOs). Kubernetes has really caught up fast.

What they'd like to have is a single API to provision resources. Not all cloud providers support all of Kubernetes features. There's a variety of command-line tools available e.g. to automate the deployment of Kubernetes. To run LHC jobs, Squids for CVMFS and Frontier are needed, as well as auto-scaling pilot pools and worker nodes, and credentials for joining the HTCondor pool. They need to have a pool of worker pods which scales up if there's work, scales down when there's not much. Ian showed a plot where CMS Monte Carlo jobs were submitted, then killed, as a result of which the number of pods decreased. Kubernetes won't necessarily kill idle jobs when scaling down, however. An alternative approach was to write a custom controller to create worker pods – essentially a Python script. There are GitHub issues open to address these problems, so there's activity in this area. There's a need to provide an X.509 proxy to authenticate. CVMFS presented Ian's team with a problem because Kubernetes only allows containers to have private mount namespaces. For running LHC jobs on Kubernetes, they deployed a proxy renewal pod, a custom controller pod, a pilot pod, a Squid replication controller and a Squid pod. Initial tests with CMS analysis jobs were successful. They ran real ATLAS and LHCb jobs on Kubernetes at a small scale, too. There are plans to perform larger-scale tests on Azure, first using RAL storage (Ceph Echo), then using Azure Blob storage via the RAL DynaFed. Federations provide the standard Kubernetes API, but apply across multiple Kubernetes clusters. It should make it easy to overflow from on-premise resources to public clouds.

Questions and comments:

- Which scale are we talking about? – The order of 1000s of jobs. Soon we'll try at a larger scale.
- How much are you going to use this logic? – It will be useful in case of peaks, when new resources will become available and when a larger scale will be reached.

System testing service developments using Docker and Kubernetes: EOS + CTA use case (Julien Leduc)

This talk is in continuation with the previous one. It's about how we can use Kubernetes for testing a service. Data archiving at CERN increases exponentially, there's many tape libraries, with a current capacity

of 0.6 EB. EOS and tapes are the strategy, which justifies the CERN Tape Archive (CTA). CTA and EOS developments are tightly coupled. There's a need for extensive and systematic testing to limit regressions. When Julien started working on system testing, he kept all components within a single git repository. Puppet deploys development instances. Among extra dependencies, he noted a database and a tape library. But deploying a developer instance takes a long time. Code changes in CASTOR also often require Puppet manifest changes. Real tape hardware cannot be triggered automatically.

They've got a developer environment, a developer test environment, nodes in QA and nodes in production. To avoid manual operations and streamline testing, continuous integration was looked into. Tests must allow involving real tape hardware. The continuous integration environment was implemented in GitLab CI. They make CTA RPMs, build and publish a generic Docker image to the GitLab registry and run the system tests in a custom Kubernetes cluster. All the required versioned RPMs are available. Julien showed the workflow on a diagram. They deploy their Kubernetes cluster with Puppet. The resources used are an Oracle database, a Ceph object store and virtual tape libraries to test workflows. Instantiating a test involves creating a namespace – if something goes wrong, it only takes to destroy it and start again. Then they instantiate all the services within, then all the pods. Pods they use include the CTA front-end, EOS, the CTA command-line interface, the tape server and a KDC. Real tape drive tests involve deploying a Puppet manifest on real hardware, adding a physical tape library source in Hiera and increasing time-outs. Julien showed a web interface visualising activity in the Kubernetes cluster, demonstrating its deployment and clean-up.

Questions and comments:

- How long does it take to run all of the tests? – Between 20 and 30 minutes. They can be parallelised.
- How do you decide which tests are run in parallel and in series? – They've got basic tests at the moment. Future tests will be coordinated with gtest.
- Would the idea be to use Magnum on OpenStack? – We couldn't do so because we need a specific kernel driver. And we need to deploy to real hardware and laptops.

Distributed computing in IHEP (Xiaomei Zhang)

Experiments are producing larger data volumes in IHEP, putting load on their single data centre. They can get resources from a wide international cooperation with experiments. There's various heterogeneous opportunistic resources available. Distributed computing is the way to go. It was first set up in 2012, to meet the needs when peaks happen. It was put into production in 2014 and integrated in the cloud in 2015. More and more experiments are joining IHEP – Xiaomei mentioned UNO, LHAASO, CEPC. IHEP process raw data, perform bulk reconstruction and analysis. Remote sites run Monte Carlo jobs and analysis. They can't afford to have a storage element. So data generated in IHEP is transferred to the sites, then back to IHEP for backing up. They've got sites from all over the world. Their network reaches 10 Gb/s and they plan to join LHCONE to improve it. They use workload managers such as DIRAC, Ganga and JSUB, and CVMFS for deploying experiment software to remote sites. IHEP recently created Strata 0 and 1 to speed up LHC and non-LHC software access in Asia.

Xiaomei presented JSUB as a lightweight and general-purpose framework developed to take care of the life cycle of tasks, defined as a bunch of jobs. Its extensible architecture makes it easy for experiments to create their own plug-ins. It allows job workflows to be customised. They use Frontier/Squid for offline database access. They've got static SQLite databases on CVMFS and mirror those which are used in the data centre. They would like to add the support of MySQL. They support multiple VOs in DIRAC and use VOMS to group experiments. They schedule and control jobs based on user, groups and tagged resources. Metadata and the file catalogue are built from the DIRAC File Catalogue (DFC), combining replica, metadata and datasets – currently 300 GB.

Their storage element originally used dCache. Now they prefer StoRM. The performance is good with the current load and the capacity reached 2.5 PB. Their instance supports multiple experiments. Their massive data transfer system is developed as a DIRAC service to share data across sites and between storage elements. Each year, 100 TB data transfers take place. They integrated it in their cloud, extending

VMDIRAC with a VM scheduler. Different cloud types are supported, such as OpenStack, OpenNebula and AWS, using different interfaces. It's not easy to meet all requirements. The IHEP cloud has become an important part of their data centre. More than 700k jobs were processed in the last two years. There were 5% of failures related to VM performance issues. They're considering commercial clouds. They ran tests and evaluated the prices, which they consider to be a bit high compared to managing their own one.

They set up action-based monitoring to improve the overall stability, ease the life of administrators and provide a global site status view. It's made of collectors and dashboards. It was designed with decisions and actions in mind. Policies are defined for taking automatic actions, e.g. sending warning messages or banning sites. The maximum number of running jobs reached 2 000. There's 300 TB exchanged from jobs each year. With multithreading jobs being very popular in HEP, IHEP are considering multicore jobs. They first looked at multicore pilots pulling multicore jobs – easy to implement but often starving. The second approach is to use standard-sized pilots with dynamic partitionable job slots – more complicated to implement. They're now planning on some HPC federation to build a grid of HPC computing resources. They're looking into scaling even more, too.

Questions and comments:

For multicore jobs, if you have a look at the dashboards in WLCG, there were many things discussed that can be useful to better optimise.

Understanding performance: optimisation activities in WLCG (Andrea Sciabà and Andrey Kirianov)

So far, LHC computing was able to meet requirements in terms of resources but there is an increasing pressure to achieve more with less. With high-luminosity LHC, the demand is soaring. Revolutionary changes are required in computing. Moore's law is slowing down. While changes in experiment software need to take place, WLCG can provide optimisation and performance tools. Computing efficiency should be investigated. New cost models need developing. Experiment workflows are being studied. There is some work in understanding the performance of experiment workflows in commercial clouds (extension of CERN batch).

The main focus is on LHCb's GaudiHive multithreaded framework. There are plans to work on LHCb software activities with Intel tools. In software performance studies, the objectives are to investigate bottlenecks and give suggestions to experiments on how to improve performance. FOM-tools are used for memory usage analysis and helped discovering that large fractions of allocated memory are not used at all. The I/O performance of ATLAS production jobs was investigated, too. Hardware counters were used. Some results were already achieved in saving large quantities of memory (e.g. of the order of 900 MB) and reaching 10% speed-ups, e.g. with Sandy Bridge. We need to understand experiment workflows to optimise them – what types of jobs they run, how many resources they require, etc. Different job types have different efficiency ratios. Speed factors on different types of jobs can be compared. Andrea's team analysed the time wasted on failed jobs, too. CERN have been working on infrastructure analysis to find the origin of inefficiencies and bottlenecks, performing passive benchmarking at the T0 and using machine learning. Experiments should now use what was discovered to work.

In the second part of this topic, Andrey presented the attempt at *Harvesting Cycles on Service Nodes*. The estimated compute resource needs for LHC Run 3 and high-luminosity are at least two orders of magnitude higher than today. There's low CPU load on storage nodes – often bound to I/O activities. Can we make use of some of these cores for additional computational tasks? Andrey showed that most of the CPU is idle.

In their first test bed, they set up I/O load generators and worked with an EOS head and disk servers. They first tried VMs, which was a bad idea because results deviated. They then used physical machines. The load was generated with `xrdstress` and compute payloads managed by HTCondor. LHC@Home Theory applications (mostly Monte Carlo) were used as compute payload. They performed accounting with `psacct` and normalised CPU. They saw no significant difference in I/O numbers. They found they were limited by network, not disk. Stressing storage, they noticed no performance degradation (maybe just a little bump at the start when the job was loaded). There was little difference in memory footprint either. In their second test bed, they used a single, modern disk server running more disks. Still, 80% of

CPU resources remained available to compute payload. Without payload, these resources were wasted and I/O performance did not improve. Compute payload doubled the interrupt rate but modern CPUs can cope with this.

This study resulted in the CERN BEER Pilot to run Batch on Extra EOS resources. They partitioned the system, reserving cores/resources for EOS and guaranteed them for EOS with cgroups. Extra resources were used by the CERN HTCondor cluster. In the worst case, 40% of CPU resources can be used. Based on the average load over 80% of CPU resources can be used.

Questions and comments:

- In the market, do you see the tendency to combine CPU and storage in one box? – Yes.
- Can we expect to have one box for everything? – It's worth saying that real life is more complicated. EOS is dynamic and it's hard to predict what will happen.

Storage & Filesystems

Advances in storage technologies (Joe Fagan)

This is an industry talk from Seagate. Seagate make 470k units/day all year long. Flash is eating the HDD market and has been for 15 years. The last standing bastion is in price. Joe described a spinning disk on diagram. Trying to add more capacity, they can add more tracks in an inch. They already doubled capacity about 21 times. Adding more spindles is another dimension to work on. Stretching tracks to a TB results in a 113 km line, giving us an idea of the head speed – 128 km/h. This is another aspect they're trying to improve, also on an aerodynamic perspective. Disk drives have to live horrifically hostile environments, and the worst they have to face is another disk drive.

Joe presented the Advanced Storage Technology Consortium (ASTC) road map. Shingled Magnetic Recording (SMR) technology allows more narrow tracks to be written. It involves overlapping tracks. Another technology is helium, the disadvantage being that the disk must be sealed. And they have to tolerate atmospheric changes. Seagate just started shipping their Multi-Sensor Magnetic Recording technology – two or more readers on the same tracks or two adjacent ones. It lets them pack tracks more and stick more bits in a row. The real challenge then is signal processing. This is shipping 12 TB drives. Heat-Assisted Magnetic Recording (HAMR) – can we reduce the dimension of the bit smaller than the head? The only bit that gets written is the one heated. This is achieved by shining a laser, producing a plasma. They now need to reach the typical disk lifetime of 5 years. Combining Bit-Pattern Media and HAMR technology, they're working on creating multiple grains per bit to a single magnetic island per bit. The increase in capacity requires them to work on the IO/s front.

Questions and comments:

SMR – is it a short-gap solution or the base for future evolution? – It is for most applications. If there's a 10 TB drive, at 40% utilisation there's a performance penalty without enjoying that capacity. If the OS and filesystem people become more careful with how they write data, there will be support for SMR for many applications.

Basic IT Services

Centralising Elasticsearch (Ulrich Schwickerath)

This topic was mentioned in previous CERN site reports. The policy at CERN used to be for users to spawn their own VMs and install an Elasticsearch instance themselves. The goal was for unification. There were many different use cases, sometimes incompatible with each other. Consolidating everything into the same cluster couldn't work. So they went for centralised management while sharing resources. Some users have special requirements in privacy, security, performance and scalability. Nevertheless, they tried

to put users with similar needs into the same clusters. Some users had special requirements, e.g. the technical network.

They've got 3 node types: master nodes, search nodes, data nodes. They used to have combined nodes running all three services, which they're now phasing out. Hardware isn't dedicated but virtualised, standard flavours. This offers some flexibility, allowing for various sizes. Their workhorses for data nodes run on special hypervisors, spinning disks with an SSD cache. The security cluster was set up for a large quantity of data. Their clusters are Puppet-managed, part of their central monitoring, documentation is available, and there are automated workflows. Accounting isn't yet in place. There are 21 clusters up and running. They don't see each other, thanks to firewall rules. They support 39 use cases, of varying sizes. They started with Elasticsearch 2 and are now deploying Elasticsearch 5. Clusters for security and monitoring are the largest in terms of disk usage. Clusters for monitoring are also demanding in terms of cores. Ulrich showed the increase in space capacity, which plateaued before they changed flavour and it increased again. The number of users is increasing, too.

Some users don't trust each other and ACLs are necessary. There are commercial plug-ins to put them in place. They also evaluated search-guard which didn't look promising in terms of performance. In the end, they went for a model involving Apache proxies, the ReadonlyREST Elasticsearch plug-in and the Kibana Own Home plug-in. There's little performance impact, the solution is based on open-source solutions and deployment is only needed on search nodes. Access is only possible with SSL and REST (the Java API was disabled). Depending on the endpoint, they use different authentication technologies (e.g. SSO or Kerberos), allowing only reads or also writes. They try to avoid patching upstream code. This model also requires them to run 2 instances of Kibana per search node. They plan to deploy ACLs, automate workflows even more, improve the stability of the service. They wish to perform some anomaly detection to catch potential problems early on.

Questions and comments:

- With commercial ACL systems coming from the authors, is there any hope to push changes upstream? – We're not having much problem with this at the moment.
- Maybe you can avoid using SSL with search-guard.

The evolution of monitoring system: the INFN-CNAF case study (Stefano Bovina)

CNAF is an Italian T1 for the WLCG infrastructure, a computing facility for 4 LHC experiments. Their monitoring system needs to take into account their heterogeneity. They needed to review the previous monitoring system based on Nagios and Lemon. They aimed at creating a cloud-oriented monitoring system, scalable, available, manageable with a CMS, supporting Lemon and Nagios scripts. They needed interaction with an API, a modern UI and a separation of contexts. Stefano showed the architecture of their system, involving RabbitMQ, Redis, InfluxDB and Sensu.

Sensu is a monitoring framework, scheduling checks on clients and managing event actions. It can reuse Nagios scripts, offers a REST API and modern dashboards. It allows the registration of clients and can scale far enough for CNAF's numbers. InfluxDB is a time-series database, it needs no external dependencies and allows downsampling data. The freedom to define retention policies is of interest to CNAF. Uchiwa gives dashboards to Sensu, showing e.g. check status, last execution times, offering the administrators the freedom to trigger new checks or silence them. They also use Grafana, using InfluxDB as source. In the future, they'll monitor their network, optimise InfluxDB, finish decommissioning Nagios and Lemon and integrate it all in an ELK stack.

Questions and comments:

Can you tell us about the hardware, how many nodes, etc.? – InfluxDB is optimised for a single-node instance. The cluster solution was previously open-source and has become commercial. Low-I/O disks aren't advisable for continuous queries. Single-node performance is going to be improved in the future.

Unified Monitoring Architecture for CERN IT and Grid Services (Jaroslava Schovancova)

This talk covers monitoring *and* accounting. There used to be data centre monitoring and experiment (WLCG) monitoring. CERN wished to unify the two. Data centre monitoring covered storage, hardware, notifications. WLCG monitoring covered job monitoring, data transfers, accounting reports. While both were hosted in CERN IT, they were run by different teams. Jarka showed a snapshot of the overall data centre monitoring, with Grafana. WLCG monitoring covered all the activities of WLCG, used by 500 users/day. Jarka showed a few more dashboards. The mandate of the new team was to find synergies using common tools and review needs.

The previous monitoring showed many different components. The new, unified monitoring united common components, even though data sources basically stayed the same. We identify here transport, storage/search, processing/aggregation and data access. Another diagram showed how these different components are linked to each other. The monitoring processes 500 GB/day, 48 hours' worth of Kafka data. Data sources are transformed and channelled by Flume. This is where data is normalised and validated, an important step given the sheer number of different sources. Kafka buffers data, Spark offers processing facilities. This step enables users to enrich data, aggregate it, perform correlations. Data access is offered by Kibana, Grafana, Zeppelin, command-line interfaces and APIs.

An activity recently undertaken is to replace the Lemon Agent with collectd to gather system and service metrics, handle thousands of metrics and be modular. This is a new data source in the overall unified monitoring architecture. Just by adding this component, very interesting operations can now be performed. It's on the host that collectd runs to generate samples. Flume enriches samples before passing them on to the monitoring infrastructure. There's already many metrics and plug-ins available. The replacement strategy involves using existing plug-ins, extending them if needs be and running Lemon sensors inside a collectd wrapper.

Services provided include monitoring, collection, visualisation, processing, aggregation and alarming. They enable infrastructure operations and scaling. The team can help and offer support to users. Jarka offered a few links showing dashboards demonstrating this work, URLs such as `http://monit.cern.ch` and `http://monit-grafana.cern.ch`.

Data Collection and Monitoring update (Cary Whitney)

The overall architecture of their monitoring system pretty much stayed the same since last time. In Berlin, Cary presented data collection. Data collection and monitoring are two different things. Putting the data to use was the hard one to do. They're moving into the monitoring phase. Cary invited us to add ideas to the HEPiX monitoring TWiki page. They hired students to work on dashboards, but it was hard because they didn't understand the data. He who collects the data knows the data. Stakeholders need to partner with whoever works on collecting data. Some people ask to copy data into another format. Is there a way to export part of the data, while keeping the original data set? They also had trouble with RabbitMQ, where they could have speed or monitoring but not both. They worked on SEDC which is for power environmental data, a plug-in for the Cray systems. They looked into Elasticsearch on Docker, upgraded to Elasticsearch 5, which broke kopf.

Cary showed a netdata dashboard, which is great for viewing data *now*. It does monitoring. It has the ability to put in monitoring based on different plug-ins. He then presented openDCIM, showing views of various NERSC Cray systems. They're going to tie it into Nagios to work with ownership, network paths, … openDCIM lets you drill down information.

Cary mentioned Cori, their Cray XC40-based system. They looked into using Shifter on it. At the time Shifter was introduced, this Cray was a glorified BusyBox, it didn't run a full-fledged Linux. Shifter is trying to do complete Docker. He showed the various systems from a power management dashboard's point of view. He created a dashboard of Cori with many charts, all 11k nodes. He can display the overview of a single node, e.g. temperature, power, memory, … They once had an issue. This gave system administrators the information they needed in a single page to solve it. But this is after the event happened. They collect from MODBUS, collectd, SEDC, syslog, amounting to 160 GB, 1.2 billion documents.

Questions and comments:

- Do you really have 1.2 billion documents? – Yes, mostly from collectd.
- How do you feed data to openDCIM? – With Python plug-ins.
- collectd can use various protocols to transfer data.
- Do you compact data? – They're collected into SSDs (a week to a month). The next day, the index is snapshot to HPSS (where all is archived). There's a manual pruning process which keeps different time windows according to the collected size. Data in HPSS remains accessible.

Flexible, scalable and secure logging using syslog-ng (Péter Czanik)

syslog-ng is for logging, recording events. It focuses on high-performance central logging. It's easy to use, with a single place to check everything. Data is still available after the collector node is down. It's secure. It can log from any logs or streams, through files, sockets, pipes, etc. It's supported on many platforms. Most importantly, it can classify, normalise, structure logs. Messages can be rewritten, reformatted and data enriched. Filtering is used for discarding irrelevant messages. It works on a message contents basis, using comparisons, wild cards, regular expressions. Traditionally collecting data into text files, syslog-ng can now record to distributed filesystems, NoSQL databases and messaging systems such as Kafka.

Most log messages have a date, a host name and a message text. They're easy to read by humans, but it's harder with scripts. Structured logging is the solution, where events are represented as name-value pairs. Parsers can turn unstructured message data into name-value pairs. One of them, the Pattern DB parser, does so by using XML message descriptions. Enriching log messages means adding additional name-value pairs based on the message contents. The `inlist()` filter is based on white- or blacklisting.

Don't panic about the syslog-ng configuration. It's simple and logical, even if off-putting at first. You can set global options, define sources (e.g. listening to a port), define destinations (e.g. Elasticsearch), filters and parsers (e.g. referring to an XML definition). Péter showed a dashboard he built from data he'd syslog'd.

The traditional syslog client-server approach can suffer from too much processing. The client-relay-server model works best, allowing namely to distribute some of the processing. Log routing is another way to scale. It's based on filtering and is about sending the right logs to the right places. There can be requirements for anonymising messages for a large variety of reasons. The problem is that regular expressions are slow and Pattern DB only works for known log messages. Overwriting with a constant or a hash is a better approach to anonymising. Péter presented a way to work with GeoIP data.

The newest syslog-ng 3.8 and 3.9 versions offer disk-based buffering, group-bys, Elasticsearch 2 and 5 support, HTTP destinations and performance improvements. Parsers can be written in Python. This adds to syslog-ng's strengths – high-performance log collection, a simplified architecture, data that's easier to work with and a lower load on destinations. Péter invited us to join the community.

Questions and comments:

Is there a way to cache data locally? – Yes, but the recommendation is to use a relay server.

Typical syslog-ng use-cases at our Tier-1 (Fabien Wernli)

They have 1.5k clients running syslog-ng sending to 3 central servers. They have other servers for real-time analysis, alarming and indexers. They're lightweight VM systems (apart from Elasticsearch which gets heavy-duty bare metal). Fabien added the friendliness of the community to the list of advantages Péter previously mentioned. They had looked at rsyslog (not flexible), Logstash (slow) and Elastic Beats (which didn't exist at the time they started).

The Elasticsearch destination supports various protocols. HTTPS for search-guard was implemented by CC-IN2P3. You need to know your `libjvm.so` to do debugging. They still use Nagios and do so with syslog-ng. Note the convenient Nagios-related variables in command templates that you can use. Variables can equally be used to send e-mails or use the Riemann monitoring system for alerting. Routing

is performed with pattern matching (Pattern DB). Filters match messages with a given flag, we connect sources with the parser and the log path links it all up. They use alerting on GPFS messages, node reboots and filesystem events, too. For each message they send, they enrich it with the role of the node, its OS and various other aspects. One type of messages they collect that's not syslog is e-mail. They're able to tell syslog-ng to read e-mail with a bit of Python. Mainly because there are still some tools only sending mail. For instance, you would find most of a cron job's information goes by mail. For HPSS, they use correlations, useful when different related messages are in different places in a log file. They use several syslog-ng modules for Puppet.

How to monitor syslog-ng itself? It's got a control socket to query statistics (e.g. the number of processed messages per source or destination). Fabien showed a peak incident where a message queue started to grow and eventually flushed itself. A few problems we might run into is that the EPEL versions aren't always available. Unofficial packages seem to work fine, however. There are some dependency problems, such as packages requiring rsyslog. When running LDAP on NSS, owner resolutions taking time can cause `syslog()` calls to block. Other destinations of interest are HTTP, Kafka, HDFS, SQL and collectd (work in progress).

Questions and comments:

- How does syslog-ng deal with a relay hanging? – This is mitigated with e.g. memory caching or disk caching.
- If a queue is full, does it slow down other destinations too? – This can be configured.

Friday 28 April 2017

IT Facilities & Business Continuity

Wigner Datacenter cooling system upgrade (Gábor Szentiványi)

The data centre comprises 4 computer rooms, 2 cooling circuits, 7 chillers each. The 1 200 kW heat load is unevenly distributed across the 4 rooms, which is a problem. The coolant flow in manifolds wasn't designed well. It's uncontrolled because there is no control panel. They've got software problems, too. They switched down all control systems and they've been controlling cooling manually since 2015. They collect data from the system to mitigate these problems. They changed the regulation system, to regulate by power consumption. They installed new software for chillers too and run under a new operation mode, a good idea from their contractor: they used to have 3 cooling modes where they only regulated fans. The new mode regulates the pump while keeping the fans at maximum speed. It's all manageable from the old EBI system. They first uploaded the software for one chiller to test it. It's now deployed on 3 chillers. If all goes well, they'll upload it to the second cooling circuit. Gábor showed a graph of the ambient, inlet and outlet coolant temperatures, which looked promising. They hope to optimise the PUE even more in the future (1.5 today). Safety is their priority – if they keep it at 100% they'll look into efficiency. They heard news that manufacturers allow for higher temperatures, too.

CERN Computing Facilities' Update (Wayne Salter)

There were two incidents. There was a water leak into their electrical room. The room is located below large transformers. They discovered on 4 February a large quantity of water which they found by chance during an unrelated intervention. They started pumping it out. Some cables were under water. One of the evacuation pipes wasn't draining water, being full up. They pumped it out, too. On 13 February, the room was dry, no obvious damage observed. On 14 February, they found a blockage in the pipe, a limestone deposit in the meter wide pipe. What made removal difficult is a bend in the pipe, where the deposit was. It was also difficult to access. On 28 February water appeared again in the false floor, after heavy rain. On 2 and 10 March the limestone was finally removed. They checked the rest of the pipe with a camera. After another heavy rain storm, there was no further leak. They then looked for an explanation as to why water came into the room. There is a known issue that the waterproofing under the transformers isn't perfect. Wayne showed photos of the concrete slab with its layer of waterproofing that's degrading. Water might come through the facade too, through ventilation grilles.

A site power cut on 9 March occurred. It was believed there was a problem from French power, and the site switched over to Swiss power. The data centre didn't suffer at all. And the UPS systems helped. In the new data centre, they considered having only limited UPS coverage – which wouldn't have been a good idea in such an event. The loss of French power was a human error, during the commissioning of a server controlling the electrical infrastructure – an incorrect procedure, which they have since then changed.

The second data centre project is for two experiments which have needs of high-level trigger farms. Feasibility studies had been made. An informal cost estimate was prepared for a Green IT Cube at CERN. They just received results from the container tender. At the time, the project was seen as positive by the CERN higher management. But experiments were very negative, as were technical experiments. They're used to running everything themselves, they don't like the idea of relying on IT. A decision was made before Xmas to go out for a tender. First they carried out a market survey, limiting the number of companies to send a tender to. They weren't sure how many responses they were going to get given the constraints that it had to rely on an existing design with a PUE of 1.1. In the end they got 16 replies, some of them not really compliant. Nonetheless they went out to tender with 4 of these companies. They got many questions for clarifications. Companies needed time to reply. During visits they had with them on site, it became clear a deadline extension was needed. There was a large price variation (up to 70%). They asked for an initial 4 MW configuration. A Finance Committee paper needed to be produced by 11 May. A report is being written with full costing information. A decision will then be made, whether they'll go ahead or not. They met with the experiments to try and look at what the benefits could be, and make it more palatable.

The second network hub project is there in case there is a major problem with the computer centre, which would cause them to be down for months. It will be sized for 48 racks and 4 rows with 120 kW. It's being built in the Prévessin site, due to be ready soon. The real building now exists. Wayne showed photos of the outside, the computer room with racks and the ventilation room. He then showed a video of the construction.

Questions and comments:

- Does the 32°C inlet temperature correspond to what the equipment can take? – It follows the ASHRAE 1 standard. All of the equipment supports this temperature.
- Can you comment on the price difference between this solution and what the experiments would like (containerised solution)? – It's not a straight comparison. There's a lot of data from the detectors to the farm, needing lots of fibre connectivity. The building costs are comparable, the connectivity makes the difference.

P2IO/LAL Datacenter Extension (Michel Jouvin)

This project is part of the P2IO initiative by HEP, nuclear physics and astrophysics laboratories. The goal is to foster synergies between 8 laboratories. It covers a small geographical area. When they started the project, they didn't have much resources. GRIF is a 10 years old multi-laboratory experience. It runs a production OpenStack cloud operated by LAL, a 2k cores setup. They managed to convince their director to stop spending money on small, badly-designed computer rooms. The focus for the Orsay data centre is to target a PUE<1.3. The return on investment for 1 M€ is estimated to 5-6 years. Chilled water is produced by chillers with a free unit. There's no UPS but a reliable power feed. The room will be rather long. It used to be a technical building. The ground floor is 1 meter over the ground floor, which saved them trouble when they once had a flood.

In the initial phase, they equipped the room with 30 racks. They've got a dense rack occupation. The feedback from these 3 years' operation is that they suffered no particular problem. The average manpower for operation is estimated to 0.15 FTE. It made the data centre very attractive for other laboratories. An active cooling door is a must. They've got fans to compensate the overpressure due to the exchanger, causing less work done by the server fans. It helps the airflow, causing less consumption. The initial room is basically full, with only two 42U racks free.

Resilient cooling is key. If a rear door exchanger fails, just open the door. Heat will be absorbed by other racks. It will have little impact on the overall room temperature. The room temperature could be higher – they're running at 23°C, they're planning to run at 25°C. Chiller redundancy is another must. Power redundancy is more critical than anticipated. Construction work in the area caused some power cuts. Cooling door monitoring and alarming is instrumental. The goal of the extension is to increase the data centre capacity. The plan is also to prepare the development required for further extensions. They got funding, but it's now managed by a public agency which isn't exactly flexible.

Power and cooling is organised in technical poles, each of which provides 300 kW. They've got an option for double-attaching all machines, although not doing so means more power is available. They still want to avoid UPSs. They currently check the PUE only manually. This isn't good enough. An automated calculation was part of the plan but they lacked resources. All the equipment is ready to be monitored for PUE. It's also necessary for identifying the optimal temperature regime – it's one of the reasons they don't run at 25°C yet. The extension work should start in March 2018, reaching production in September 2018.

Manage your hardware failures in an (almost) automated workflow (Mattieu Puel)

This becomes useful if your team are in charge of handling failures (from detection to resolution) and your interactions with support teams are time-consuming. Dell set up a support service platform where you provide your own diagnostic and describe what you want through SOAP APIs. Mattieu described a workflow for changing parts, taking the example of a disk failure, with Redmine at the heart of it. Metadata is necessary to make the process successful. It comes from their CMDB. It's forwarded to the host from Puppet. A Python-Redmine process queries hardware status on the node, generates a diagnostic file and

the issue is tracked in Redmine. Some thoughts: whether or not to send full diagnostics to the vendor or only targeted information; how to manage issue assignment (self-assignment or round-robin); how to handle named deliveries and front-desk interactions; which probes to rely on; how to deal with flapping alarms and false positives. Once all is in Redmine, statistics can be run. With Redmine, you can draw plots of issues by service and by category. Mattieu showed that the top two categories are batteries and disks. In the future, they'll want to use the dispatch API for incidents requiring human diagnostics, integrate non-Linux boxes in the workflows (e.g. storage arrays) as well as worker nodes, and enrich logs to let an ELK stack do the plotting. It's estimated that, with their current approach, Mattieu's team saved 54 hours for the 242 automated cases that took place last year.

Questions and comments:

- Do you keep all this data in a special database (failure rates, …)? – Not yet, it's extracted from Redmine. It's also why we want to push it to an ELK stack.
- How many vendors besides Dell have this kind of automation? – It's a question Mattieu wished to ask the audience, in fact. Tony said HPE has it, but you have to pay extra for this (and they decided not to do that).

Miscellaneous

Unscheduled Security Demo (Liviu Vâlsan)

Liviu gave an unscheduled presentation on a rather inconspicuous security demo. He showed pictures of UBS keys which were lost and found throughout the week. Some of us were curious enough to open them (physically). Swapping the board, we could see a small SD Card reader. It's called a USB Rubber Ducky. It will emulate a keyboard, issuing key presses at blistering speeds. It's multiplatform. Soon after connection, it will start running things. You wouldn't have seen much. Nobody actually connected it. Every time it was found, participants would return it to the local organisers. Well done, HEPiX.

Workshop wrap-up (Tony Wong)

There was an overwhelming 125 participants and 66 contributed presentations. The presentations were well distributed. There were some newcomers and participants who hadn't come to HEPiX for a long time.

A definite trend is the migration away from AFS. There were several facility constructions presented, focusing on more efficient operations. CentOS is being adopted increasingly. More and more are migrating to Grafana. There's a continuing presence by non-traditional fields, other than HEP. IPv6 is being adopted more and more. There's ever more sites moving to HTCondor. There are several open positions at various sites.

There were talks on various network aspects and presentations on security incidents and threats. Several configuration management and monitoring talks were given, with the Elasticsearch stack mentioned several times. Some new monitoring tools were presented. Concerning storage and filesystems, more and more sites implemented box-like services. The Seagate technology presentation sparked a lot of interest. Knights Landing clusters are being deployed more and more. Our community investigate future benchmarks. While several sites add resources to their clouds, container solutions are in the spotlight. In coming years, the growth in resource requirements by LHC experiments will exceed the community's ability to provide them. The interesting idea of using storage nodes for computing may help. Several presentations described computer centre extensions, management and incidents.

Tony congratulated Balázs and his team for a well-run meeting and thanked Seagate, Super Micro and Dimension Data for sponsoring the event. The next meeting will take place at KEK, Japan. KEK have been long-time participants in HEPiX. HEPiX Fall 2017 will be co-located with the HUF and LHCOPN-LHCONE meetings. Takashi Sasaki and Tomoaki Nakamura are the local organisers. Tony thanked us all for coming and wished us a safe trip home.

The Wigner Data Centre Visit

A bus took us from the workshop venue, through the rush-hour traffic, to the Buda side and up the hills, which showed contrasts of seemingly abandoned buildings and large luxury houses. Access to the site was heavy controlled which is why the organisers had repeatedly asked us to ensure we had correctly registered and had our passport with us. On our way to the site, armed security guards were picked up to give them more time to go through the lengthy process of checking each of our IDs against a participant list. Once on site, we walked a short distance in a rather natural setting, with the smell of pine trees and the sound of birds, mixed with the contrasting combination of derelict buildings and modern-looking, well looked-after ones. CERN's sister data centre was one of those, located near the walled boundaries of the site, next to a rusty watchtower dating back from the communist era. Its location gave a lovely view of the city below, sitting high up on the hill, where 30 cm of snow had fallen just the week before.

The organisers kept on emphasising the fact we weren't allowed to take pictures of either the outside or the inside of the building. The inside of the building still gave the impression that it was brand new. We arrived in a rather empty hall with a wide green stripe painted on the floor and wall, and small plants by the windows. Our guide explained that there are 4 computer rooms. One room hosts equipment for the Hungarian government, and access was blocked by a heavy gate. We were unfortunately not allowed to access any of the three CERN rooms either, and the reason we were given is that it is Swiss territory. We could only glimpse through a window one or two 25°C cool aisles using in-row cooling. In the rest of the room, a temperature of between 35°C and 40°C is sustained at all time. The building uses liquid chillers which are located on the roof. The target PUE is <1.5. A 72 cm raised floor gives space for cabling.

We visited the UPS room, which uses a hot aisle layout. Equipment is connected to UPS units providing 2 MW/room, giving 8 minutes in the event of power cuts. It was interesting to note that everything is labelled in Hungarian, clearly suggesting that the on-site team are exclusively locals. There is a diesel generator for each room, too. The building structure, walls and doors, can sustain a 1000°C fire for an hour.

Our guide showed us the so-called optical room, where two of the three 100 Gb/s links from CERN arrive. They were hidden inside a rather opaque metal cage and we could only see a few orange cables coming out of its top. Behind it, a small door lead to a set of servers used for the general-purpose maintenance of the facility. Next to this room, another one hosted large, green tanks of nitrogen gas.

CERN use 40% of the data centre computing capacity. The facility is monitored 24/7. The local staff have normally no idea of the services running on the servers, a responsibility which is left to CERN staff back in Geneva. The Hungarian government servers are managed following a different model, where it's government IT staff who work on-site to manage their services, from the office building next door. The building can offer 8 MW in total. There is room for expansion. In fact, one of the 3 CERN rooms is only half full.