Lessons Learnt from WLCG Service Deployment

J. D. Shiers

CERN, 1211 Geneva 23 Switzerland

Jamie.Shiers@cern.ch

Abstract. This paper summarises the main lessons learnt from deploying WLCG production services, with a focus on Reliability, Scalability, Accountability, which lead to both manageability and usability. Each topic is analysed in turn. Techniques for zero-user-visible downtime for the main service interventions are described, together with pathological cases that need special treatment. The requirements in terms of scalability are analysed, calling for as much robustness and automation in the service as possible. The different aspects of accountability - which covers measuring / tracking / logging / monitoring what is going on – and has gone on - is examined, with the goal of attaining a manageable service. Finally, a simple analogy is drawn with the Web in terms of usability - what do we need to achieve to cross the chasm from small-scale adoption to ubiquity?

1. Introduction

Ian Foster's famous paper "What is the Grid? A Three Point Checklist**"** [1] lists 3 criteria that are proposed for determining whether a given system is "a Grid" or not. The Worldwide LHC Computing Grid (WLCG) [2] is a system that uses resources provided by two major production Grids – namely the Enabling Grid for E-SciencE (EGEE) [3] in Europe and elsewhere, and the Open Science Grid (OSG) [4] primarily in the US. Thus, by definition, WLCG satisfies the first two criteria – these two major Grids are clearly separate management domains and at least a workable degree of *de-facto* standards is needed for successful production services to be offered. This paper addresses the third point in this checklist – quoted in full below – describing in detail the lessons learnt from offering world-wide production services across many sites for a number of years.

"... to deliver nontrivial qualities of service. (A Grid allows its constituent resources to be used in a coordinated fashion to deliver various qualities of service, relating for example to response time, throughput, availability, and security, and/or co-allocation of multiple resource types to meet complex user demands, so that the utility of the combined system is significantly greater than that of the sum of its parts.)"

2. The WLCG Computing Model

The purpose of the WLCG service is to satisfy the data processing and analysis needs of the LHC experiments at CERN. These needs have been described many times (see, for example [5]) and the WLCG service implementation is based on a hierarchical model that was first proposed by the MONARC (Models of Networked Analysis at Regional Centres for LHC Experiments) project [6]. In some senses, this model is an evolution and a formalisation of the move to distributed processing that has been underway in HEP for a number of decades, as described in [7]. The hierarchical model itself

is very similar to that proposed by Jim Gray et al. [8], with the sum of resources at each "tier" being approximately constant.

As a reminder, the main responsibilities of the different tiers of the WLCG computing model are as follows:

• Tier0 (CERN): safe keeping of RAW data (first copy); first pass reconstruction, **distribution of RAW data and reconstruction output (Event Summary Data or ESD) to Tier1**; reprocessing of data during LHC down-times;
• Tier1: safe keeping of a proportional share of RAW and reconstructed data; large scale **reprocessing** and safe keeping of corresponding output; **distribution of data products to Tier2s** and safe keeping of a share of simulated data produced at these Tier2s;
• Tier2: Handling **analysis** requirements and proportional share of **simulated event** production and reconstruction.

Sites that are members of the WLCG collaboration sign a Memorandum of Understanding (MoU) [9] that lists the specific responsibilities for that site, the resources that they will offer each supported virtual organisation (VO), the maximum time for intervening in the case of service degradation or loss, as well as the annual availability that should be provided. However, even in the simplest case, these "services" in fact involve numerous components – that sometimes involve other (WLCG) sites and/or third parties, such as network operations. An example is given below for Tier1 sites:

"Tier1 services must be provided with excellent reliability, a high level of availability and rapid responsiveness to problems, since the LHC Experiments depend on them in these respects.
The following services shall be provided by each of the Tier1 Centres in respect of the LHC Experiments that they serve, according to policies agreed with these Experiments. With the exception of items i, ii, iv and x, these services also apply to the CERN analysis facility:

i. acceptance of an agreed share of raw data from the Tier0 Centre, keeping up with data acquisition;
ii. acceptance of an agreed share of first-pass reconstructed data from the Tier0 Centre;
iii. acceptance of processed and simulated data from other centres of the WLCG;"

3. WLCG Service Challenges

In 2004 the WLCG Service Challenge programme [10] was launched, aimed at *"achieving the goal of a production quality world-wide Grid that meets the requirements of the LHC experiments in terms of functionality and scale."* Whilst most widely known for their contribution in ramping up data movement and data management services, an often over-looked but extremely important aspect of this programme was that of delivering full production services. Indeed, whilst the first two challenges focussed on basic infrastructure setup and network tuning, the bar was raised considerably for Service Challenges 3 and 4. Both challenges included not only tests performed using the *dteam* virtual organisation, but more importantly included extensive production use by all four of the major LHC experiments. As such – and for the first time – an attempt was made to identify and deploy all of the needed services at the participating sites. The target date for the deployment of these services at the Tier0 was May 2005. This was a highly ambitious goal – not only was this date well in advance of the delivery of the final "Baseline Services" working group report [11], but also a number of the middleware components behind the corresponding services had never previously been deployed in production conditions, nor had they been tested by the experiments, nor integrated into their data processing environments.

Realising that there were two distinct goals to be achieved and understanding the unlikelihood of deploying the new services perfectly the first time, two separate instances of the main new services

were deployed in the production environment. These were a so-called *pilot service* – the goal of which was to expose the new service to the experiments in order to allow them to gain experience with it, integrate into their software and to provide early feedback and the standard *production service* – to be used both for *dteam* and – after and urgent fixes or enhancements from the experience with the pilot system – for the experiments' production processing. The requirements on these two instances in terms of stability versus rapid updates were clearly different and this model continues to prove valid for making available new features in the production system today.

Other issues that compounded the task for initial service deployment were the lack of clear understanding of how these services would be used by the experiments – making resource estimation an impossible task – as well as the lack of available hardware resources. As is true for many laboratories, hardware resources at CERN are acquired through competitive tender and are typically over-subscribed. Thus, we had little choice but to deploy the services on the only boxes that were then available – typically batch worker nodes, lacking even dual power supplies and mirrored system disks – deferring the choice of suitable systems with which to target the needed availability and reliability to a later date.

4. WLCG Services – the *"a priori"* analysis

Starting in August 2005, and based on the service levels implied in the WLCG MoU, an *a priori* analysis of the Tier0 WLCG services was performed. This targeted not only the hardware needs, but also the middleware requirements, operational procedures and all other service aspects involved in setting up robust and reliable services. In addition, the feedback and experience from the early months of Service Challenge 3 called for a significant number of service updates. In order to perform these, a "long shutdown" of several days was scheduled during October 2005. It was well understood that such intervention could not normally be performed on a production service, but this was felt to be the least intrusive method available at that time to perform the numerous pending upgrades – including not only deployment of new middleware releases, but also network reconfiguration, hardware moves and reallocation. Unfortunately, insufficient hardware was still unavailable to redeploy the services in an optimal manner, and their redeployment continued over a period of many months. This was first done using a regular "intervention slot" – simplifying not only scheduling of such interventions with the experiments but also their production planning. However, it was soon realized that the coupling between the various services – not to mention their impact that in many cases extended way beyond the host site and was often Grid-wide – called for a less intrusive manner of performing such changes.

5. Expecting the (un-)expected

It is a truism to state that anything that can go wrong will do so – this is often referred to as "Murphy's law". Whilst this is even part of popular culture, it is still often ignored – who has not lost one or more files due to human error, hardware failure or even a combination, only to find out (or often to realise, in the case of a personal computer) that no adequate backup exists? However, do we systematically prepare for common failures or problems – let alone less likely scenarios? Experience from previous generations of HEP experiments – such as those at the LEP collider at CERN - remind us that there can be many causes of data loss or corruption, including software failures. Whilst naively some such scenarios would appear to be so unlikely that they can be readily dismissed, experience over more than two decades of running production services suggest that preparation for all eventualities is a much safer strategy. In the early days of attempting to deploy the European DataGrid (EDG) Replica Location Service (RLS) as a file catalogue, it was even claimed that if the release procedure were correctly followed, it would be "impossible" for a bug to appear in the production system. More valuable lessons that were (re-)learnt by a new generation of service providers were the length of time that it takes to deploy a full production service and the amount of detail that is required in the associated planning process. Some concrete examples of events that have taken place that are more or less expected, depending on one's viewpoint, are listed below:

- The Expected:
  - When services / servers don't respond or return an invalid status / message;
  - When users use a new client against an old server;
  - When the air-conditioning / power fails (again & again & again);
  - When 1000 batch jobs start up simultaneously and clobber the system;
  - **A disruptive and urgent security incident… (again, we've forgotten…)**
- The Un-expected:
  - When disks fail and you have to recover from backup – and the tapes have been overwritten;
  - When a 'transparent' intervention results in long-term service instability and (significantly) degraded performance;
  - When a service engineer puts a Coke into a machine to 'warm it up'…
- The Truly Un-expected:
  - When a fishing trawler cuts a trans-Atlantic network cable;
  - When a Tsunami does the equivalent in Asia Pacific;
  - When Oracle returns you someone else's data…
  - When mozzarella is declared a weapon of mass destruction…

6. Building Robust Services

Techniques for building robust distributed services are well understood. However, not only does the Grid take the challenge of overall service reliability to new heights, but also the associated middleware has not always been designed with robust and reliable deployment in mind. Operations workshops have frequently highlighted either – in the worst case – the total absence of error messages and logging, or else the obscurity of such messages or the inconsistency of the logging itself. These and other such issues are described in the "Grid Operations Cookbook" [12]. These issues are compounded further when trying to resolve cross-site issues, where to-date we have not even managed to consistently agree on the use of UTC (Coordinated Universal Time) for logging messages: in a worldwide Grid, being able to correlate events is clearly essential and there is no widely accepted alternative to the use of UTC, yet many services continue to log in local time – often in places that are inaccessible to those charged with problem resolution. These issues are high-lighted – but not yet resolved – in a presentation [13] prepared for the WLCG Collaboration workshop held in Victoria, BC, in September 2007.

We describe briefly the main techniques in use at the WLCG Tier0 below. However, experience shows that some simple "common-sense" rules need to be reiterated. The overall system is only as reliable as the "weakest link" – there is no point in using redundant power supplies and / or load-balanced servers if only a single power feed is used for all such systems. Similarly, a single network switch negates the investment in hardware and software redundancy. Whilst these guidelines seem obvious, they are not always followed. As a concrete example, the large majority of all Storage Resource Management (SRM) v1.1 load-balanced servers at CERN are deployed in a single rack and break both rules above. In the event of a power failure to this rack, the remaining systems rapidly entered overload and the entire service collapsed. (The argument that we should by now have migrated to SRM 2.2 can hardly be considered a justification in this case.) Even in the case when these simple guidelines are followed during the initial deployment of a service, all too frequently re-configuration of components of the service has resulted in later exposure. For example, replacement of network switches by newer models with more ports has meant that servers that were originally split over two or more switches end up behind a single-point-of-failure. In other words – and as will be re-iterated with further justifying arguments below – it is *essential* that an end-to-end service view is maintained: not just for interventions but for all service aspects.

7. Common Techniques

The basic techniques that are currently deployed are:

- Load-balanced, stateless middle-tier servers (where the main application logic resides);
- Database clusters, currently implemented on Oracle Real Application Clusters (RAC).

A description of the database services at the Tier0 can be found in [14]. The evolution of database services for physics in the last twenty five years can be found in [15].

An additional technique – less well understood and with significant restrictions, such as the requirement for all services to be on the same network switch – is that of H/A Linux. This should only be used when the middleware involved does not support one of the previous techniques. This is currently used at CERN for the VOM(R)S (Virtual Organisation Membership Service / VOM Registration Service) service.

However, it is important to stress that there is no free lunch – significant additional work is required at the middleware/application level to ensure that the software correctly supports these H/A techniques. Whist relatively straightforward for stateless middle-tier applications, it is far from trivial for database applications, let alone higher-level experiment services, which may well use multiple back-end services and where maintaining consistent state across eventual roll-back due to failover during update transactions needs to be carefully coded and tested. Particularly important in this respect is the developer / database administrator relationship, as described in [16]. Consider, for example, experiment-driven production data transfers. These involve numerous services, including storage management at the source and sink, possibly file and dataset catalogues, the File Transfer Service (FTS) and the experiments' own data management layers. In the middle of transferring a logically consistent set of files – opaque to the underlying generic services – one or more of the nodes hosting a database at source and / or sink fails. Is sufficient information passed up through the layers to the driving application to cleanly abort or preferably recover? Does the entire file set have to be aborted or can it be cleanly restarted? Is it even possibly to clearly identify all the possible failure modes and prepare for them? Most likely not.

8. Intervention Analysis

An analysis of the causes of service interventions since the time of the European DataGrid (EDG) Replica Location Service (RLS) – a period of some 5 years – shows that the main cause for user-visible downtime is *scheduled interventions*. This comes as a surprise, but is arguably good news. As they are scheduled, they can be readily addressed, whereas unscheduled interventions, which come not only from unstable middleware, but also from infrastructure instabilities, are harder to address. A first analysis of the times of unscheduled services – limited at this stage to the WLCG Tier0 and Tier1 sites – has been performed, revealing the following broad categories:

- Power, cooling and network problems, responsible in the worse cases for complete site downtime for several hours or even days;
- Problems with storage and related services, e.g. CERN Advanced Storage manager (CASTOR) or dCache (by definition);
- Problems with back-end database services or the interaction between the application layer and the database, affecting services such as LCG File Catalog (LFC), FTS, VOMS, Service Availability Monitoring (SAM) etc.

9. Scheduled and Unscheduled Interventions

As stated above, the main reason for service interruption is for scheduled interventions. The reasons include:

- Adding additional resources to an existing service;
- Replacing hardware used by an existing service;
- Operating system / middleware upgrade / patch;

- Similar operations on DB backend (where applicable).

Given the coupling between the various services and indeed their impact that is typically Grid-wide, early scheduling of these interventions is essential. The WLCG and even EGEE have agreed upon procedures for scheduling and announcing such interventions – including the availability of the service(s) at the end of the exercise. Interventions that are performed without following these procedures are classified as *unscheduled* and are counted against the corresponding service / site availability.

Equally important is a detailed intervention plan – with responsibilities, timeline, decision points and roll-back strategy – as well as a post-mortem analysis, listing any problems encountered and their resolution. This is particularly important not only within a site – where different people may be responsible for performing a subsequent intervention – but also for preparing reliable procedures for external sites to perform similar intervention (services are typically upgraded to a new major release, e.g. FTS 2.0, first at the Tier0 and at the Tier1s only after a minimum of several weeks' experience at the Tier0).

10. WLCG Services – the *"a posteriori"* analysis

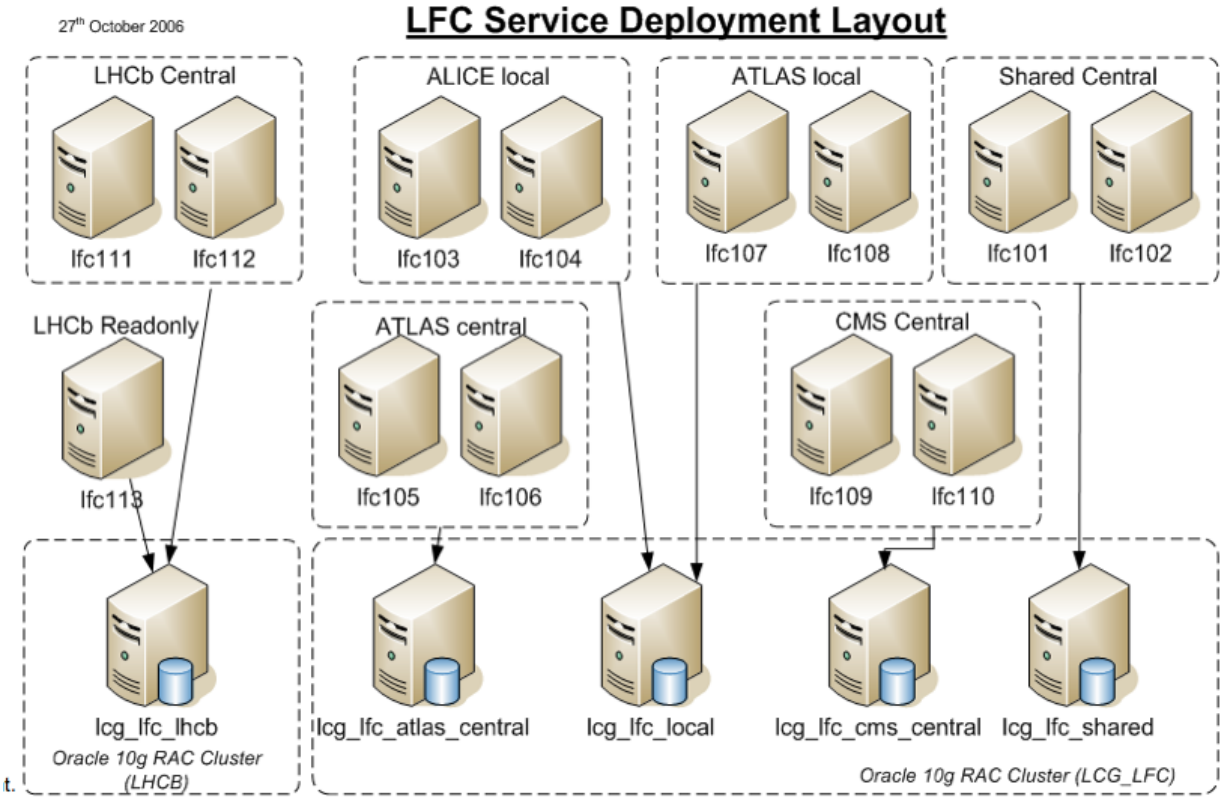
As described above, an analysis was performed in 2005 based on our understanding of how the services would be used and how service degradation or loss would impact ongoing production. However, as has been seen on a number of occasions, experiments may adjust their usage of a service based on experience and/or the service may evolve to address observed weaknesses. It was therefore decided to redo this analysis, profiting from several years of production experience. Whilst this is very much work in progress, an initial analysis has already identified a number of weaknesses, as well as varying degrees of software maturity with respect to resilience to glitches and the ability to support 'transparent' service upgrades. This work will continue with the WLCG Tier1 sites, as well as the major Tier2s, with the goal of supporting transparent upgrades in all of the main services well prior to CHEP 2009. The main points of this review were as follows:

- **Hardware**
  - Servers - Single or multiple, DNS load balanced, HA Linux, RAC, multiple power supplies / feeds, battery / diesel backup?
- **Software**
  - Can the middleware handle loss of one or more servers?
  - Does service reliability improve or degrade with the number of servers? (Used for load
  - What is the impact to other services and/or users of a loss/degradation?
- **Quiesce/Recovery**
  - Can the service be cleanly paused?
  - Is there built-in recovery when the service comes back?
- **Tests and Documentation**
  - Have these procedures been (regularly) tested in practice, are they used to provide transparent interventions?
  - Does the required documentation for operations and service staff exist?

Of these services, one of the most advanced is the (W)LCG File Catalog (LFC), shown schematically below. This is implemented using load-balanced servers in front of an Oracle RAC. In case of failure of one of more nodes (at either level) any remaining nodes take the load. Only in the case of failure of all nodes, or in the case of Oracle cluster-ware failure, is the entire service affected. The service has proven to be resilient to glitches and middleware upgrades – other than those requiring schema changes to the backend database – are routinely performed with zero user-visible downtime.

The advantages of making services both resilient to short-term glitches (say up to 10 minutes, the time for which systems are covered at the Tier0 by battery backup – only a few being additionally

covered by diesel Uninterruptable Power Supply (UPS)) as well as supporting transparent service interventions as listed above are huge. Firstly, the users see a greatly improved service. Secondly, service providers have much more flexibility in scheduling interventions, which no longer need to be confined to error-prone early morning or later evening slots. These two advantages greatly improve the service provider – user relationship, reducing stress levels and providing further positive feedback. However, it must clearly be supported by the associated middleware, which is not always the case today. Fortunately, the analysis that has been performed has identified a small number of areas where large improvements can be obtained for relatively little effort.



However, the LFC service is currently only used by two of the LHC VOs: by ATLAS as local file catalogue and by LHCb as a global catalogue, with a read-only replica at CNAF and eventually at all LHCb Tier1 sites.

On the other hand services such as CASTOR(SRM) and the FTS that are ranked as critical Tier0 services by all VOs currently have a number of single points of failure (SPOFs). Furthermore, CASTOR upgrades are intrusive and typically require a downtime of several hours.

Finally, whilst the recent efforts in monitoring are both very welcome and long overdue, one should not fall into the trap of thinking that monitoring alone will make systems more reliable. It will certainly help identify areas that need to be improved, but robustness can only be achieved by design and not empirically.

In summary, there is a relatively poor match between those services that are relatively well advanced in terms of resilience (such as Grid batch and the LFC), and those listed as critical by CMS below.

11. The User View

The service availability that is by default measured in the WLCG is that of the basis component services – Compute Element (CE), Storage Element (SE), LFC etc. Whilst this is important to sites for understanding the status of the services that they offer and is increasingly supplemented by experiment tests running in the SAM framework, the user view is essential. In the case of the CMS experiment, a

list of critical services and the impact of failure or degradation has been prepared [18]. The services have been ranked on a scale from 0 (service not used by CMS) to 11(!) (CMS stops operating) [19]. The most critical category (11) is not currently treated, the most severe that remains (10 - CMS stops transferring data from the data taking pit to the CERN computer centre) is assigned to the following services:

- Main Oracle backend service for CMS;
- CERN SRM & CASTOR;
- Data book-keeping system (CMS service);
- Batch queues;
- Kerberos;
- Networking from the pit to the computer centre;
- Campus networking;
- (Fixed?) telephones;
- Web backend system;
- (File) transfer system from the pit to the computer centre.

Classified as only slightly less critical are services related to data transfer from the Tier0 to Tier1s, including the WLCG File Transfer Service (FTS) and CMS' Physics Experimental Data Export (PhEDEx) system. Whilst some leeway can be bought by additional hardware – e.g. for transfer buffers both for pit to computer centre as well as Tier0-Tier1 transfers – these stringent requirements cannot currently be met and will have an impact on all aspects of service deployment, including hardware setup, middleware/storage-ware and operational procedures.

12. Preparing for Experiment-/Application-Driven Challenges

Given a backdrop of such complexity, it may come as no surprise that the most effective strategy for ramping up experiment-driven challenges has been to establish clear goals and metrics for the exercise in question, to prepare carefully – testing all components individually and later together – and to gradually ramp-up in both scope and scale. This approach is that adopted by the CMS experiment [20] and will also be used for the foreseen Combined Computing Readiness Challenge (CCRC08)[21], proposed at the WLCG Collaboration workshop prior to this conference. The scope of this challenge is given below – all of these tests should be run concurrently:

- Test data transfers at 2008 scale:
  - Experiment site to CERN mass storage; CERN to Tier1 centres;
  - Tier1 to Tier1 centres; Tier1 to Tier2 centres; Tier2 to Tier2 centres
- Test Storage to Storage transfers at 2008 scale:
  - Required functionality; Required performance
- Test data access at Tier0, Tier1 at 2008 scale:
  - CPU loads should be simulated in case this impacts data distribution and access

13. The Grid is Greater than the Sum of its Parts

The (W)LCG Technical Design Report (TDR) [22] list two motivations for adopting a Grid solution. These are as follows:

1. *Significant costs of [providing] maintaining and upgrading the necessary resources … more easily handled in a distributed environment, where individual institutes and … organisations can fund local resources … whilst contributing to the global goal*
2. *… no single points of failure. Multiple copies of the data, automatic reassigning of tasks to resources… facilitates access to data for all scientists independent of location. … round the clock monitoring and support.*

For funding reasons, the first argument is clearly extremely important – for the reason stated in addition to the fact that many of the institutes involved are multi-disciplinary. Thus, not only for resource sharing within a site but also to bolster the scientific and intellectual environment in the

collaborating countries, such a scenario is much healthier than one where all resources are concentrated at the host laboratory (and acquired locally).

The second argument needs further analysis. With the exception of services and processing that is performed at the Tier0 site, the fact that much of the data – e.g. with the exception of Monte Carlo data in a given Tier2's output buffer – is replicated at several or many sites, the partial of even total failure of a site should not stop the associated production or analysis. Similarly, some of the services – such as the FTS – are already designed to cater for service interruptions at source and/or sink site. For example, if the storage element (SE) at a given site is about to enter scheduled maintenance, the corresponding FTS *channels* that source or sink data in that SE can be paused. This still allows new transfer requests to be queued, but they will not be attempted until the channel is re-opened, avoiding wasting bandwidth on transfers that are bound to fail and potentially reducing the background load on support staff (analysing "fake" failures.)

14. Operations & User Support Issues

There is insufficient room in this paper to analyse these issues in detail. However, the solutions that are used in both areas today are limited in terms of the number of incidents that can be handled. For example, the current operations model is rather eyeball intensive and will not obviously scale to handle much larger numbers of VOs and/or sites, as would be expected to be the case in the future. Much can be done by making the services more robust by design and by automatic problem resolution and – where possible – correction. Furthermore, 24x7 coverage is not currently offered but will be needed – at least for the HEP VOs – in the not too distant future. Similarly, each problem ticket that requires human intervention has an associated cost. Problems that cannot be rapidly resolved and are passed to 3rd level support can take many hours, days or even weeks to resolve. Only an extremely small number of such problems is required to bring the support system to its knees. These issues must clearly be addressed by any long term sustainable e-Infrastructure project that is to offer a viable service.

15. Crossing the Chasm

Many analogies have been drawn between the Web and the Grid. What remains, in terms of challenges, for the Grid to cross the chasm from niche usage to ubiquity? One obvious answer is usability – had the tools for Web authoring remained as primitive as those offered even in the early days of Mosaic and other such browsers, it is hard to imagine it pervading all of our lives, as it does today. The bar is almost certainly much higher in the case of the Grid – although much work has done [23][24] to assist new applications to move to the Grid and to hide the complexity, it is still a complex, manpower intensive task. This will be a major challenge for future Grid initiatives, such as the European Grid Initiative [25] that is currently being studied.

16. Conclusions

Despite almost impossible constraints, WLCG has managed in an extremely short time to turn prototype services into widely used, production quality systems. Whilst by no means perfect and – in a number of cases, most significantly in the area of storage management – with outstanding issues still to be resolved, worldwide production Grid services are a reality. The focused effort that was needed to establish these services will still be required for some time to come, with the clear goals of deploying the remaining residual services, ramping up service reliability and reducing the overall deployment and operational costs. The work on improving service reliability needs to concentrate on those areas that are of most concern to the experiments. In particular, this includes all services related to data acquisition, first pass processing and distribution of data to the Tier1s.

Acknowledgements

Many people have contributed to and continue to contribute to the ramp-up and delivery of WLCG Services. It is impossible – for simple space reasons – to mention every individual involved in this

work. With apologies to those not explicitly mentioned, which includes regular participants in the LCG Service Coordination Meetings and the joint operations meetings – whilst gratefully acknowledging their work – the following warrant an explicit mention. The WLCG Service Coordination team consists of (in addition to the author): James Casey, Flavia Donno, Maarten Litmaath and Harry Renshall. Patricia Méndez Lorenzo coordinated and presented the *"a posteriori"* analysis of WLCG Tier0 services and Tim Bell coordinated the original *"a priori"* work. Ulrich Schwickerath, Miguel Coelho Dos Santos, Jean-Philippe Baud, Harry Renshall, Gavin McCance, Maria Girone and Dirk Düllmann all provided input to this recent analysis.

Glossary

ALICE: A Large Ion Collider Experiment
ATLAS: A Toroidal LHC ApparatuS
CASTOR: CERN Advanced Storage Manager
CE: Compute Element
CERN: Centre Européenne pour la Recherche Nucléaire
CMS: Compact Muon Solonoid: (A large general purpose detector and associated collaboration at CERN)
DNS: Domain Name System
EDG: European DataGrid
EGEE: Enabling Grids for EsciencE
ESD: Event Summary Data
FTS: File Transfer Service
LCG: LHC Computing Grid
LHC: Large Hadron Collider
LHCb: The Large Hadron Collider beauty experiment
MONARC: Models of Networked Analysis at Regional Centres for LHC Experiments
MoU: Memorandum of Understanding
OSG: OpenScienceGrid
PhEDEx: (CMS) Physics Experiment Data Export
RAC: (Oracle) Real Application Clusters
RLS: Replica Location Service
SAM: Service Availability Monitoring
SE: Storage Element
SRM: Storage Resource Management
TDR: Technical Design Report
UTC: Coordinated Universal Time
VO: Virtual Organisation
VOMRS: Virtual Organisation VOM Registration Service
VOMS: Virtual Organisation Membership Service
WLCG: Worldwide LCG (the collaboration, as opposed to former middleware release)

References

[1] I. Foster, Argonne National Laboratory and University of Chicago, What is the Grid? A Three Point Checklist, 2002.
[2] The Worldwide LHC Computing Grid (WLCG), http://lcg.web.cern.ch/LCG/.
[3] The Enabling Gri ds for E-sciencE (EGEE) project, http://public.euegee.org/.
[4] The Open Science Grid, http://www.opensciencegrid.org/.
[5] The Worldwide LHC Computing Grid (worldwide LCG), Computer Physics Communications 177 (2007) 219–223, Jamie Shiers, CERN, 1211 Geneva 23, Switzerland
[6] MONARC - Models of Networked Analysis at Regional Centres for LHC Experiments –

available at http://monarc.web.cern.ch/MONARC/.
[7] LHC Experiment Computing, to appear in the proceedings of the International Conferene on Computing in High Energy Physics, Victoria, BC, September 2007, Ian Fisk
[8] "Petascale Computational Systems: Balanced Cyber-Infrastructure in a Data-Centric World ," pdf, Gordon Bell, Jim Gray, Alex Szalay, Letter to NSF Cyberinfrastructure Directorate., IEEE *Computer*, V. 39.1, pp 110-112, January, 2006.
[9] Memorandum of Understanding for Collaboration in the Deployment and Exploitation of the Worldwide LHC Computing Grid, available at http://lcg.web.cern.ch/LCG/C-RRB/MoU/WLCGMoU.pdf.
[10] The LCG Service Challenges – focus on SC3 rerun, Jamie Shiers, in the proceedings of the International Conference on Computing in High Energy Physics, Mumbai, India, February 2005.
[11] LCG Baseline Services Group Report – available at http://lcg.web.cern.ch/LCG/peb/bs/BSReport-v1.0.pdf.
[12] The Grid Operations cookbook, Nicholas Thackray et al. – available at https://edms.cern.ch/document/726257.
[13] WLCG Collaboration Workshop, held in conjunction with CHEP 2007: operations "birds-of-a-feather" session.
[14] CERN Database Services for the LHC Computing Grid, Maria Girone, to appear in the Proceedings of the International Conference on Computing in High Energy Physics, Victoria, BC, September 2007.
[15] Databases in High Energy Physics – A Critical Review (to be published in "HEP Computing in the Era of the Grid") – available at http://cern.ch/jamie/Databases-in-HEP-v1.pdf.
[16] Effective Oracle by Design (Osborne ORACLE Press Series), Thomas Kyte
[17] WLCG Tier0 Service Review, presented by Patricia Méndez Lorenzo at WLCG Collaboration Workshop, held in conjunction with CHEP 2007.
[18] CMS Critical Services – available at https://twiki.cern.ch/twiki/bin/view/CMS/SWIntCMSServices.
[19] Spinal Tap – see http://www.spinaltapfan.com/.
[20] CMS Experience with Computing, Software and Analysis Chalenges, Ian Fisk(FNAL), to appear in the Proceedings of the International Conference on Computing in High Energy Physics, Victoria, BC, September 2007.
[21] Proposal for Combined Computing Readiness Challenge 2008 – available at http://indico.cern.ch/contributionDisplay.py?contribId=26&confId=3578.
[22] LCG Technical Design Report, CERN-LHCC-2005-024, available at http://lcg.web.cern.ch/LCG/tdr/.
[23] UNOSAT, Project Gridification: The UNOSAT Experience, EGEE User Forum, CERN, 1st March 2006. Session: "Earth Observation— Archaeology—Digital library", Author: P. Mendez.
[24] ITU, International Telecommunication Union Regional Radio Conference and the EGEE Grid, EGEE User Forum, CERN, 1st March 2006. Session: "Earth Observation—Archaeology—Digital library", Author: A. Manara, M. Cosic, T. Gavrilov, P.N. Hai.
[25] The European Grid Initiative Design Study – website at http://www.eu-egi.org/.