# CURRENT STATUS OF FABRIC MANAGEMENT AT CERN

G. Cancio, T. Kleinwort, W. Tomlin*, M. Siket, V. Bahyl, S. Chapeland, J. van Eldik*, V. Lefebure, H. Meinhard, P. Poznanski, T. Smith, M. Stepniewski, D. Waldron, CERN, Geneva, Switzerland
D. Front, Weizmann Institute of Science, Rehovot, Israel

*Abstract*

This paper describes the evolution of fabric management at CERN's T0/T1 Computing Centre, from the selection and adoption of prototypes produced by the European DataGrid (EDG) project [1] to enhancements made to them. In the last year of the EDG project, developers and service managers have been working to understand and solve operational and scalability issues.
CERN has adopted and strengthened Quattor [2], EDG's installation and configuration management tool suite, for managing all Linux clusters and servers in the Computing Centre, replacing existing legacy management systems. Enhancements to the original prototype include a redundant and scalable server architecture using proxy technology and producing plug-in components for configuring system and LHC computing services. CERN now coordinates the maintenance of Quattor, making it available to other sites. Lemon [3], the EDG fabric monitoring framework, has been progressively deployed onto all managed Linux nodes. We have developed sensors to instrument fabric nodes to provide us with complete performance and exception monitoring information. Performance visualization displays and interfaces to the existing alarm system have also been provided. LEAF [4], the LHC-Era Automated Fabric toolset, comprises the State Management System, a tool to enable high-level configuration commands to be issued to sets of nodes during both hardware and service management Use Cases, and the Hardware Management System, a tool for administering hardware workflows and for visualizing and locating equipment.

## ELFMS: QUATTOR, LEMON AND LEAF

With the scale up needed to serve LHC's Tier-0/1 centre, CERN's computer centre will increase from currently 2700 to approximately 8000 nodes within less than 3 years. With the end of the EDG project, remaining developments as well as the maintenance and deployment of Quattor and Lemon were taken over by the ELFms (Extremely Large Fabric management system) project, coordinated by CERN's IT department. ELFms also accommodates the LEAF system, developed with support from the UK's GridPP [5] as part of the LCG project [6]. The three toolsets have been integrated and form a fully modular, interoperating framework. With ELFms, we manage a large, heterogeneous environment in several dimensions. First of all, in terms of functionality and size, it includes large general purpose batch and interactive farms, disk and tape servers, database and web servers, etc. Also, various generations of hardware co-exist leading to a multiplicity of setups for example in terms of CPU types, memory and disk sizes. Finally, four supported Operating Systems are in use: Three RedHat Linux based (RH73, RHES2.1, Scientific Linux 3) and Solaris 9.

## QUATTOR

In 2003, we decided to migrate the dispersed Computer Centre configuration information into the Quattor Configuration Database (CDB). Data sets coming from over 20 different places (databases, shared files, installation scripts) had to be identified and appropriate CDB data models and schemata defined. The resulting configuration templates were arranged into hierarchical template-based structures matching service and cluster descriptions, and utilities written for automating the actual data migration. As of today all legacy databases have been shut down and CDB now holds configuration information (like networking parameters and physical location) for 95% of all systems in the computer centre. For systems managed by Quattor, all configuration details including the precise hardware setup, list of software packages, running services, etc. are stored in CDB. CDB's relational backend, "cdb2sql", is being used for general-purpose access to the database contents. A KickStart generation tool interfacing to CERN's legacy RedHat Linux installation server has been written, which might eventually be replaced by Quattor's AII solution (which is itself KickStart based).

### *Software deployment and node configuration*

With CDB interfaced to KickStart, we have the means to perform centrally configurable and reproducible installations. In addition nodes should be managed during their runtime in order to maximise availability without needing to reinstall due to functional or security updates, or service reconfigurations. Unlike systems like ROCKS [7], Quattor addresses this requirement by providing two subsystems, SPMA (Software Package Manager Agent) for software deployment and NCM (Node Configuration Manager) for system configuration.

The NCM is a framework, where service-specific plug-ins, called 'Components' are responsible for making the necessary system changes to bring the node to the desired state as defined in its CDB configuration profile, regenerating or updating local service configuration files and restarting services if needed. Components are written to be idempotent, and multiple invocations result in no

* With support from the Particle Physics and Astronomy Research Council (PPARC), Swindon, UK.

change unless their configuration is changed. Configurations of NCM components are stored in CDB, making full use of its hierarchical inheritance and overwriting mechanisms in order to define service definitions and software environments.

SPMA manages the software packages installed on a node and can handle multiple package formats, including RedHat's RPM and PKG for Solaris. SPMA's main advantage over other software distribution systems (like `apt-get` or `yum`) is the clear separation of node configuration and repository contents: adding a new version of a package to a repository does not imply its deployment on client nodes without an explicit configuration change in CDB. This way, external drifts are avoided and reinstallations of systems result in exactly the same setup as defined in the SPMA configuration. It is possible to configure nodes with different package version selections, which is sometimes critical e.g. due to backward incompatibilities on production services which would otherwise block a package upgrade required elsewhere. SPMA also allows rollback to older well-known software configurations. In addition, SPMA supports the simultaneous installation of multiple package versions. This is useful for e.g. kernel upgrades, as new 'dormant' kernel versions can be installed in advance and activated at a later moment in time.

SPMA replaced ASIS [8], the legacy application software distribution system developed and used previously at CERN, in spring 2003. CERN's legacy UNIX configuration system, SUE [9], was next on the phase-out list. The main advantages of NCM over SUE are its access API to CDB configuration and the availability of improved system management libraries. Replacing all SUE system configuration modules by NCM components did require a substantial amount of work. The organization of training sessions and generation of appropriate documentation has been critical in order to achieve this task. Around 90 components have been produced, ranging from basic system configuration up to setup of Grid services (see below under "Grid Integration").

## *Scalability via proxy caching*

Uniform deployment of large software updates onto clusters with thousands of nodes requires provision for scalable solutions. By default, Quattor uses HTTP as base transport protocol for RPM packages and XML configuration profiles. Since HTTP requests can be cached via proxy mechanisms we have deployed a replicated "reverse proxy" server architecture, using apache `mod_proxy`. Our architecture contains two proxy levels. A server cluster, consisting of two back-end servers and a variable number of front-end servers, handles requests for configuration profiles and software packages. A second level of proxy nodes mediate and cache requests between racks of client nodes and the front-end servers, and provide also serial line based console access to the rack nodes. This second level avoids wasted network bandwidth due to multiplication of identical transfers and it increases reliability by reducing the ratio of clients per server from hundreds or thousands to the number of nodes in a rack (around 40).

## *Grid Integration*

Integrating fabric resources into Grid environments requires deploying additional software and configurations to cluster and server nodes - if possible using automated tools for minimizing human intervention which is costly and error prone. Configuration errors may affect not only a site but the whole Grid response and stability. An example is the creation of 'resource holes' attracting too many jobs due to incorrect resource announcements or wrong node configuration causing jobs to fail. For achieving the integration of Quattor with the LCG Grid, NCM configuration components needed to be developed for multiple grid services and server types. At the same time a number of Tier1/Tier2 centres are evaluating or starting to use Quattor for managing their grid and/or production clusters. LCG and EGEE [10] are adopting Quattor for managing their certification and testing environments, respectively.

In order to minimize work duplication and advance towards a uniform and end-to-end Grid fabric management solution a community task force has been set up and configuration components and templates have been produced for most of the LCG-2 services. In the CERN computer centre, we are in the process of progressively reconfiguring our batch nodes for converting them into LCG-2 Worker Nodes; other service nodes will follow in the future. With the deployment on other sites, site assumptions and dependencies in the core software and configuration components have been identified and replaced by more general solutions, and the quality and robustness of the software has consequently been increased. Generic installation guides and tutorials have been produced, and a Savannah based bug and task tracking system put in place to coordinate the developer team with the user community.

## *Status and next steps*

Quattor is currently managing around 2100 Linux nodes in the CERN computer centre, and providing a uniform management solution across several dimensions of heterogeneity. All nodes in the computer centre (managed or not by Quattor) are declared in the Configuration Database. We have ported Quattor from RedHat 7.3 to RedHat Enterprise Server 2.1 and Scientific Linux 3 (IA32 and IA64). In parallel, the CERN Solaris support team is working on a port to Solaris.

SPMA/NCM are used on a daily basis to deploy security, functional and configuration updates. One example of a large deployment (which used to take days) was the upgrade of the CERN batch system from LSF 4 to LSF 5, which took place in less than 15 minutes on more than 1000 nodes, and without service interruptions.

The inclusion of the EGEE and LCG testbeds into the existing infrastructure, as well as new deployments outside CERN, will imply significant functional growth. For example, a 'light' version of Quattor will be released in the near future, containing NCM and configuration templates for configuring grid services on top of existing installations.

Also, the ramp-up for LHC computing will require optimising potential bottlenecks. For example, usage of parallelism for XML profile generation inside CDB in order to reduce recompilation times.

Security is another critical aspect: a prototype CDB fine-grained authorization module needs to be tested and deployed, as well as an improved mechanism for handling host certificates which would allow using encrypted HTTPS for configuration profile transport.

## LEMON

Lemon (LHC EDG monitoring) is a client/server based monitoring system. On every monitored node, a monitoring agent launches and communicates using a push/pull protocol with sensors which are responsible for retrieving monitoring information. The extracted samples are stored on a local cache and forwarded to a central Measurement Repository. Sensors can collect information on behalf of remote entities like switches or power supplies. The Measurement Repository can interface to a relational database or a flat-file backend for storing the received samples, and can be queried via a SOAP based API.

Thanks to the modular agent design, it has been possible to add to the sensors provided by the EDG project (mostly performance related) new sensors for providing a broader fabric view and accurate exception and inventory information. Some 120 new metrics have been provided, ranging from general information (for hardware information, status of daemons) over batch system specifics (for LSF) to service specific metrics (eg. for disk and tape servers, Oracle databases). In addition to the existing C++ API, a Perl sensor API has been provided to ease the task of writing sensors. Some of the metrics and sensors have been external contributions, e.g. coming from IBM (for StorageTank) and BARC India (LSF quality of service). In total, we collect around 150 metrics organized in 70 metric classes and implemented in 12 sensors.

For reliable archiving of monitoring information we adopted and enhanced the 'OraMon' Oracle back-end based instance of the Measurement Repository. OraMon (developed with support from Weizmann Institute and LCG) is in production at the CERN computer centre since November 2003 collecting information from approximately 1800 monitored nodes totalling around 1.2 GB of daily data.

### *Fault recovery and derived metrics*

In fabric environments with thousands of nodes, the ability to perform automated fault recognition and self-healing recovery becomes paramount, in order to free operators and technicians performing repetitive operations (like cleaning up temporary space or restarting daemons). We re-engineered the initial EDG fault-tolerance prototype and now provide two independent modules for global and local scope action. A light-weight self-healing module running on every monitored node can compare cached metric samples to absolute or relative reference values and trigger recovery actions or activate exceptions which lead to triggering an operator alarm. For correlations involving several nodes, a Perl plug-in based framework allows execution of advanced user-defined correlations, runs recovery actions and injects derived metrics back to the Monitoring Repository.

### *Server Redundancy*

For enhancing server robustness, a high-availability solution is being developed, providing fail-over recovery at the database and OraMon level. At the database level, an Oracle Streams based replication between two database servers has been enabled using a master-master scenario.

### *Visualization*

In order to present the monitoring information in an easily understandable way, we have created a graphical web interface using PHP and RRDTool [11]. Data from the Monitoring Repository (and other data sources like Quattor CDB, or LSF) is retrieved via a Python-based application and stored into RRD on a per-node, per-cluster or per-service basis (eg. power supply consumption).

### *Status and next Steps*

The Lemon agent and the OraMon repository have been in production for 2.5 years and one year respectively on both Linux and Solaris nodes. Lemon is also used outside CERN's Computer Centre: The GridICE project [12] uses it for monitoring Grid services, and we are currently discussing how to interface Lemon with CalTech's MonALISA system [13]. Recently, the CMS experiment decided to use Lemon for monitoring their DAQ prototype nodes. Lemon has been integrated to the current operator alarm system at CERN, SURE [14], and we are working on a gateway into the future LHC controls alarm system, LASER [15].

## LEAF

The successful deployment of Quattor at CERN has provided a platform for some advanced components to be added to the fabric management stack. These components, known collectively as the LHC-Era Automated Fabric (LEAF) toolset, consist of a State Management System (SMS), which enables high-level commands to be issued to sets of Quattor-managed nodes, and a Hardware Management System (HMS), which manages and tracks hardware workflows in the Computer Centre and allows visualization of equipment location.

### *State Management*

SMS enables sets of nodes to be automatically re-configured to be in production or on standby during the execution of operational and service management Use Cases. By leveraging the Quattor framework, a set of machines may be removed from production during, for instance, a kernel upgrade or a physical move, undergo the intervention and then be seamlessly put back into production once the activity is complete. Concurrent events, such as a simultaneous kernel upgrade and physical move, are correctly handled, the machine not going into production until both interventions are complete. All parties can see who is doing what, when and why. SMS ensures authentication, authorization, validation and auditing.

At a lower level, each node is configured to execute a specific script when asked to perform a particular state transition. For example, when the desired state of an interactive node is changed from 'standby' to 'production', logins are enabled and monitoring alarms switched on. Given this encapsulation, it becomes trivial to issue high-level configuration commands to a heterogeneous set of nodes, such as "go into production", because the caller needs no knowledge of how this is achieved.

### *Hardware Management*

By LHC start-up, mass installations, moves, renames and retirements are to be expected, along with daily hardware failures. A product of extensive workflow analysis and process re-engineering, HMS facilitates predictable, consistent, traceable and automatic workflows that are designed to scale up to the future needs of CERN's Tier 0/1 facility. The system:

- Automates the update of all databases and repositories participating in its Use Cases
- Issues formal work orders wherever people are required to perform an action
- Provides statistics for management reporting
- Removes the dependency on specific individuals or informal communication for fulfilment

### *Status and next steps*

Since its first production release in late 2002, where it was used to manage the installation of 400 new machines, HMS has evolved rapidly with 16 new releases last year. With nearly 1,500 machines relocated in the last four months, HMS has been well tested and has proven successful. The first full production release of SMS was in January 2004 to coincide with a stable configuration database schema and is currently deployed for all Quattor-managed nodes. HMS will continue to evolve smoother, more automatic processes and to handle more secondary scenarios on demand. It may be necessary in the future to track other types of hardware and to integrate new or modified components.

SMS is currently deployed for all farm PCs but needs to be extended to other types of node. In addition, more service specific SMS clients need to be written to allow service managers and system administrators to perform maintenance tasks and upgrades more easily. One such client will be a more advanced GUI which is currently being developed to allow easy invocation of service management and operational interventions across sets of selected nodes, automatically initiating HMS workflows and invoking SMS state changes as necessary.

## CONCLUSIONS

The ELFms tool suite is in full production at CERN's Computer Centre. We are benefiting from the results of 3 years of development, in which the components have been stabilized and hardened from prototype to production quality. Thanks to their modular architecture, Quattor and Lemon have been integrated into the Computer Centre's site specific setup and procedures, and have allowed to incrementally phase out legacy solutions. In their quality as framework systems, it has been possible to enhance the original set of plug-ins for system configuration and monitoring, therefore increasing the fabric's automation level. With LEAF as "glue" layer on top of Quattor and Lemon we now are able to define complex high-level workflows for full lifecycle management. While a number of non-core developments and integration steps remain to be done, we have already now achieved a high level of management automation. This is a requisite for providing quality fabric services in Grid environments, where the correctness of configuration is critical and may affect the stability of the complete Grid workload system.

Other projects and sites are starting to deploy ELFms modules. This is leading to an improved software portability and quality, and also increased functionality, in particular for configuring and managing LCG and EGEE Grid services. In this shared collaboration context, the coordination and standardization of developments and configuration schemas will become very important for streamlining new developments aiming to instrument Grid services, and avoiding incompatibilities and work duplication.

## REFERENCES


[1] http://www.edg.org
[2] http://www.quattor.org
[3] http://cern.ch/lemon
[4] http://cern.ch/leaf
[5] http://ww.gridpp.ac.uk
[6] http://cern.ch/lcg
[7] http://www.rocksclusters.org
[8] http://cern.ch/asis
[9] http://cern.ch/proj-sue
[10] http://cern.ch/egee
[11] http://www.rrdtool.org
[12] http://server11.infn.it/gridice
[13] http://monalisa.cacr.caltech.edu
[14] http://cern.ch/service-sure
[15] http://cern.ch/proj-laser