LEMON - LHC Era Monitoring for Large-Scale Infrastructures

BABIK Marian, FEDORKO Ivan¹, HOOK Nicholas, LANSDALE Thomas Hector, LENKES Daniel, SIKET Miroslav, WALDRON Denis
CERN - European Organization for Nuclear Research CH-1211 Genève 23, Switzerland

E-mail: Ivan.Fedorko@cern.ch

Abstract. At the present time computer centres are facing a massive rise in virtualization and cloud computing as these solutions bring advantages to service providers and consolidate the computer centre resources. However, as a result the monitoring complexity is increasing. Computer centre management requires not only to monitor servers, network equipment and associated software but also to collect additional environment and facilities data (e.g. temperature, power consumption, cooling efficiency, etc.) to have also a good overview of the infrastructure performance. The LHC Era Monitoring (Lemon) system is addressing these requirements for a very large scale infrastructure. The Lemon agent that collects data on every client and forwards the samples to the central measurement repository provides a flexible interface that allows rapid development of new sensors. The system allows also to report on behalf of remote devices such as switches and power supplies. Online and historical data can be visualized via a web-based interface or retrieved via command-line tools. The Lemon Alarm System component can be used for notifying the operator about error situations.

In this article, an overview of the Lemon monitoring is provided together with a description of the CERN LEMON production instance. No direct comparison is made with other monitoring tool.

¹ To whom any correspondence should be addressed.
1. Overview

Lemon is a server/client based monitoring system (see Figure 1). On every monitored node, a monitoring agent is launched and this communicates using a push/pull protocol with sensors, also running on the node, which are responsible for retrieving the monitoring information. The monitored samples are stored in a local cache and also forwarded to the application server using UDP or TCP transport protocols with or without authentication/encryption of the data samples. Sensors can collect information on behalf of remote entities like network switches or power supplies. The application server stores the samples received in a central measurement repository which can interface to either a relational database or a flat-file backend. A web based interface, called Lemon-web, is provided for the visualization of the data.

Lemon is part of the Extremely Large Fabric management system (ELFms) tool suite [1] but has no functional dependencies on other ELFms components and can be used separately [2].

Lemon is available for Linux servers only, as the Windows servers use a Microsoft dedicated monitoring system.

2. Lemon Agent and Sensors

On every Lemon monitored machine, a lemon-agent gathers data from 1 or more sensors. From a sensor’s point of view the agent is in charge of launching and configuring the sensors, scheduling measurements and collecting samples.

2.1. Lemon Agent

By design, when the agent requests information from its sensors it expects NO reply. This allows the sensors to provide their own logic as to whether data should be recorded or not at the sensor level. As a result a sensor sends samples asynchronously and can send data at any time. The agent makes no distinction between measurements triggered by requests from the agent and measurements which are spontaneously generated. The agent is implemented in C code and provides an API for sensor implementation. The officially supported sensor APIs are for Perl and C/C++. 
2.2. Lemon Sensors
Sensors are separate processes connected to the agent via a bi-directional pipe and live for the duration of the agent’s lifetime. Each sensor is able to collect a given set of metrics.

Sensors implement metric classes (see Figure 2). These classes are then instantiated as metric instances for which each instance has its own metric id. The benefit of this approach is that the same piece of code can be instantiated many times, with each instance having a different set of configuration options.

Figure 2. Schema of the Lemon agent and sensor communication via a bi-directional pipe. The metric classes are instantiated as metrics with their own metric id. The exception sensor provides specific functionality as described in 2.2.2.

2.2.1. Supported Sensors. Lemon provides a set of basic sensors which are suitable to cover most of monitored cases. A full description of the implemented classes is available in Error! Reference source not found.. Here is a brief description of the basic sensors:

- The Monitoring Sensor Agent (MSA) is a built-in sensor in the Lemon agent. The sensor checks the health of the Lemon agent and provides information about an agent instance (e.g. heartbeat, version, number of running sensors etc.).

- The Linux sensor provides standard system performance monitoring (e.g. partition information, CPU load, etc.) based mainly on the information found under the /proc/ directory.

- The File sensor provides various file-related utilities such as checking the number of files in a given directory, checking if a file is writable, checking size parameters (e.g. size in bytes, usage of a file system as a percentage of total size, etc.), md5 checksum and the number of seconds from last the update of a given file.

- The Parse-log sensor provides a log parsing metric. A configured file (by default /var/log/messages file) is parsed for the presence or absence of a given expression.
• The Exception sensor is providing support for correlations between metrics.

Approximately 30 additional sensors have been implemented covering functionalities such as processing SNMP information, hardware scanning, IPMI or specific tape storage information and many others. More examples may be found by viewing the Lemon code repository [4].

2.2.2. Sensors Exception. The exception sensor is a C++ sensor developed in collaboration between CERN and the Bhabha Atomic Research Centre (BARC) [6]. The sensor is used for generating the exception metrics and launching recovery actions in response to any problems detected. The built-in correlation engine allows for exceptions to be raised based on the information reported by multiple metrics and supports mathematical expressions, string comparisons, regular expressions and the generation of alarms on behalf of other monitored entities. The sensor Exception is an officially supported lemon sensor and is mandatory if LAS (Lemon Alarm System, see Section 5.) is intended to be used.

2.3. Lemon command lines

2.3.1. lemon-cli. Lemon-cli is a command line interface tool for accessing metric information from the local cache as well as from the central measurement repository.

2.3.2. lemon-host-check. Lemon-host-check is a command line utility for reporting the state of the exceptions on a given machine as well as for the exception management (temporary disabling, enabling the exceptions or groups of them).

3. Lemon Measurement Repository

The Lemon Measurement Repository is made of two components. The first one is a Lemon application server. This application is responsible for processing data received from the lemon agents and storing it in the database backend. Data are sorted by metric id, i.e. every unique metric id is represented in the backend by table where metric values from reporting nodes are stored.

The second part of the repository is the database backend. The Lemon monitoring system currently provides two types of database backend: a flat file and an Oracle database. Except for minor differences the functionality is the same for both backend implementations:

• The Flat file backend provides standard functionality and is easy to install and manage. It is not advised to use this backend though for clusters bigger than 1000 computers. It is not that the server would not be able to handle this number of machines, but rather there may be a limitation due to I/O on the local file system. It is perfectly fine for smaller sites. The required file size is about 500 kB/day per monitored machine with the basic sensors (Section 2.2.1.).

• The Oracle based backend provides full functionality and features and is designed for large computer centres. A set of the Oracle database schema management tools is provided for the maintenance of the measurement repository (e.g. adding a table for new metric).

4. Lemon Web

The Lemon-web application is used to retrieve metric information from the monitoring repository and to store it into time serialized data structures. Round Robin Database (RRD, [5]) files are used for time series data storage. Every file holds information from all metrics measured for a given entity (e.g.
node). Data from the RRD files is then passed to the web interface for visualization. Lemon-web supports grouping of entities (e.g. nodes) into groups (clusters, racks, hardware models, etc.) and provides a summary or average view of each group even if certain machines are part of multiple groupings. Lemon-web supports the federation of Lemon instances. On Figure 3 is an example of web interface for a cluster entity.

![Figure 3. An example of the Lemon-web interface displaying information for a cluster. A user has access to entity details retrieved from the measurement repository as well as to exception/alarm information for the nodes.](image)

5. Lemon Alarm System

Exception information stored in the measurement repository with an Oracle database may be processed further by the Lemon Alarm System (LAS). The LAS business logic is implemented as a set of PL/SQL procedures which run on top of the Oracle database and use the exception metrics as input. The Lemon web server hosts the LAS GUI, accessible with a web browser. Operators and administrators go via a web browser to use and configure LAS. Two kinds of alarms are distinguished: raw alarms (sampled every 60 seconds) and visible alarms built from processed and combined raw alarms. Multiple raw alarms are grouped either per host or per alarm type into the visible alarm. Visible alarms must be acknowledged by the computer centre operator. An acknowledgement procedure may be linked directly to an external event management system. Lemon monitoring may be used without activating the LAS functionality.

6. Production Instances

Live instances of Lemon are working at CERN, CNAF in Bologna and the BARC institute and in a major financial institution hosting the largest instance.

6.1. CERN Computer Centre

At CERN, approximately 8000 physical nodes are monitored with Lemon. In addition to the nodes also various facility measurements (e.g. power, temperature) or measurements on behalf of other entities (e.g. disk arrays) are provided. In total more than 11000 unique entities are monitored in the computer centre with a single Lemon instance. Approximately 1.7 million metrics are reported to the
Lemon instance with a frequency of a few minutes to a couple of hours. 1000 unique metrics of 250 metric classes are in use. Table 1 shows the distribution of the number measured metrics. It should be stressed that 150-200 metrics are monitored on more than 5000 nodes.

| # of metrics monitored on node | # of nodes |
|------------------------------|------------|
| < 50                         | ~ 3000     |
| 50-100                       | ~ 50       |
| 100-150                      | ~ 250      |
| 150-200                      | ~ 5000     |
| 200-250                      | ~ 2600     |
| > 250                        | ~ 2        |

7. Short-Term Enhancements
Here is the list of on-going enhancement activities:

- Oracle backend: The current implementation of the Oracle backend for the measurement repository uses 2 tables per metric (one table for recent data, one for historical). The new implementation will use one table per metric class in order to improve the database performance.

- Sensors: The basic sensors and the exception correlation engine are continuously being enhanced.

- Exception on aggregated data: At the moment the exceptions are evaluated at the node level. In the new implementation, the exception will be evaluated using correlation rules based on any of the data collected by the instance, i.e. support for exceptions on aggregated objects.

- Lemon-web: A re-designed and re-implemented Lemon-web shall very soon provide a more flexible configuration and visualization of collected data.

References
[1] ELFms project information is available at http://elfms.web.cern.ch/elfms/
[2] Lemon project information is available at http://lemon.web.cern.ch/lemon/index.shtml
[3] A list of basic lemon sensors is available at http://lemon.web.cern.ch/lemon/doc/sensors.shtml
[4] The Lemon code is available at https://svnweb.cern.ch/trac/lemon/browser/trunk
[5] RRDtool is described at http://www.mrtg.org/rrdtool/
[6] BARC institute http://www.barc.ernet.in/