# The Detector Monitoring System

F.Berni, F. Carbognani, V. Dattilo, F. Gherardini, G. Hemming, D. Verkindt

**VIR-0191A-12**

https://tds.ego-gw.it/ql/?c=9005

# Change Record

| Version | Date | Section Affected | Reason / Remarks |
|---|---|---|---|
| **VIR-0191A-12** | | All | |

# Table of content

# 1 Introduction

## 1.1 Scope

This document describes the features, the software implementation and the upgrades of the Detector Monitoring System, which, in its first implementation, was used in the control room to monitor the detector status and to give in an easy way any information about possible problems and malfunctions. The evolution of the system improved a lot the first implementation, with the development of a new architecture and new functions. The upgrades were mainly done to add new functions and to improve the reliability and the flexibility of the whole system.

## 1.2 Abbreviations and Acronyms

| **Abbreviation/Acronym** | **Description** |
|---|---|
| DMS | Detector Monitoring System |
| CDB | Connection Database (https://tds.ego-gw.it/ql/?c=8774 , https://tds.ego-gw.it/ql/?c=8905) |
| AJAX | Asynchronous JavaScript And XML |
| CSS | Cascading Style Sheets |
| HTML | HyperText Markup Language |
| HTTPS | HyperText Transfer Protocol Secure |
| GPL | General Purpose Licence |
| PHP | Pre-Hypertext Processor |
| UI | User Interface |
| W3C | World-Wide Web Council |
| XHTML | eXtensible HyperText Markup Language |

## 1.3 Glossary

| **Term** | **Description** |
|---|---|
| Active Directory | Microsoft user management service |
| LDAP | Lightweight Directory Access Protocol (http://en.wikipedia.org/wiki/Lightweight_Directory_Access_Protocol) |
| AJAX | Browser-server asynchronous communication kit |

| | |
|---|---|
| Apache | Open source web server (http://httpd.apache.org/) |
| DIV | Block-level HTML element (http://www.w3.org/TR/html401/struct/global.html#h-7.5.4) |
| IIS | Microsoft web server |
| JavaScript | Browser-side scripting language (http://en.wikipedia.org/wiki/JavaScript) |
| jQuery | Javascript library (http://jquery.com/) |
| JpGraph | JpGraph is an Object-Oriented Graph creating library for PHP >= 5.1 The library is completely written in PHP and ready to be used in any PHP scripts (both CGI/APXS/CLI versions of PHP are supported) (http://jpgraph.net/). |
| MySQL | Open source database software (http://dev.mysql.com/) |
| PHP | Server-side scripting language (http://www.php.net) |
| Foxbox | Hardware gateway to send and receive SMS or MMS (reception only) messages using a commercial SIM card (http://www.smsfoxbox.it/). |

## 1.4 Related documents

- The Detector Monitoring System user manual – Authors: F.Berni, F. Carbognani, V. Dattilo, F. Gherardini, G. Hemming, D. Verkindt - (VIR-0192A-12) - https://tds.ego-gw.it/ql/?c=9006 .

# 2 What is DMS ?

The DMS is a tool based on a set of monitoring processes providing the needed information and a WEB tool intended to collect, further process, archive and display/notify, in a centralized way, the information regarding the status of the ITF, the technical infrastructures, the electronics equipment, etc... This information is displayed as a *"multi cell"* table, where each cell, named **flag,** denotes the status (codified by a color) of a specific item.

In case of problem the system can perform several types of notification. All the various information is stored in MySQL databases and can be retrieved through dedicated WEB interfaces.

## 2.1 A bit of history

### 2.1.1 General requirements

A dedicated on-line quality monitoring system was needed to display (and store) in a centralized way the on-line information concerning the detector's status. It was intended to summarize in a single graphical interface the status & quality of the whole ITF by using information provided by various monitoring processes. In 2004, the Moni library was developed in order to allow to connect such processes to the data acquisition system and to configure easily the data quality flags used by the DMS. Other processes, like BigBrother, provided their own information, with an output format aligned on the Moni processes.

The data quality flags were then used by the DMS to provide a warning or even alarm notification to operators whenever signals exceed the standard values.

In the graphical interface, the information had to be formatted as following:

| **Header section: general info about ITF and links** | | | | |
|---|---|---|---|---|
| 🟥 **Subsystem 1** | 🟨 flag | 🟩 flag | 🟩 flag | 🟨 flag |
| | 🟥 flag | 🟥 flag | 🟨 flag | ⬜ flag |
| 🟥 **Subsystem 2** | 🟨 flag | 🟩 flag | 🟨 flag | 🟩 flag |
| | 🟥 flag | 🟩 flag | 🟩 flag | 🟥 flag |
| ⬜ **Subsystem 3** | 🟩 flag | 🟩 flag | ⬜ flag | 🟨 flag |
| | 🟩 flag | 🟨 flag | 🟩 flag | 🟩 flag |
| 🟥 **Subsystem n** | 🟨 flag | 🟥 flag | 🟥 flag | 🟩 flag |
| | 🟩 flag | 🟩 flag | 🟨 flag | 🟩 flag |

A specific item had to be represented by a *"flag"* and the message level had to be codified through a set of colors:

| **Flag Color** | **Denoted State** |
|---|---|
| 🟥 | Problem on that item; it can be the real source of the problem or it can be induced by another item. |
| ▨ | The problem is induced by another item. |
| ⬛ | At least one channel used to compute the flag has corrupted data. |
| ⬜ | At least one channel used to compute the flag is missing. |
| 🟨 | Warning on that item: it is almost outside the standard state. |
| 🟩 | Standard state. |

The information had to be provided in a hierarchical way and with more features: by clicking on each flag a new page displaying additional info should be opened.

Some kind of efficient notifications of the event, besides the displaying on the screen, like SMS, email, or sound were necessary to avoid that sometime the operator may not realize in due time an event occurrence, thus delaying the triggering of some action or to request the intervention of the on-call person.

The Detector Monitoring System had to be configurable in a flexible and modular way by the Operations Group through a dedicated configuration file and trough dedicated WEB interfaces.

Some web interfaces had to be implemented to easily retrieve information, to set some parameters e to display auxiliary information.

### 2.1.2 Technical requirements

The DMS had to be able to operate using the standard Linux operating systems common to the Virgo host machines.

We decided that the DMS had to be written in PHP thus at server side it required PHP running on a standard web server (Apache, IIS, etc.) in conjunction with a MySQL database; at browser side it was decide that DMS had to use JavaScript taking advantage of the jQuery library (http://jquery.com/).

Development and testing of the application was undertaken mainly with PHP version 5 and above and MySQL versions 4 and 5.

User authentication had to be undertaken in the standard manner for applications in the Virgo PHP environment - against the central Active Directory user database, making use of the PHP LDAP library (http://php.net/manual/en/book.ldap.php).

It was decided that the plots had to be generated using the PHP library called JpGraph (http://jpgraph.net/).

A SMS server was necessary to send SMS in case of alarm; this server had to be able to interact with the DMS PHP scripts.

## 2.2 The current situation

The DMS is a now a tool largely used in control room to get very low latency information about the status of the interferometer and its environment, the status of the various electronics equipment or online processing servers, etc…

As requested, the information is provided in a hierarchical way: the main page is a graphical interface showing information in predefined colored boxes called **"flags"**.

The system is completely configurable through configuration files and through devoted web interfaces available after Active Directory login.

The system can perform alarm notifications in different ways: by email, by sms and by playing a sound from the main page.

All the information like the evolution of the flag status, the change of configurations or the notifications of the alarm are stored in a MySQL database and can be retrieved through specific WEB interfaces which have been developed during the years; among these interfaces there are:

- Plot: interface to show the evolution of the status/signal of the flag;
- Shelving/Muting/Comment: interfaces to act on the configuration of the flag;
- DMSLog: interfaces to retrieve DMS pictures stored in the database specifying also some search criteria;
- AlarmLog: interface to retrieve notifications event providing detailed information ;
- FlagLog: interface where it's possible to have the information displayed in chronological way. The information might be: changing of a flag/subflag status, alarm notification, shelving/muting and comments ordered by time
- Flag list: interface which lists all the flags and all the information associated to that flag.
- Admin interface: interface reserved to the system administrator and from which we can set the user permissions.

From the main page we can open another web page, called "Level Two page", showing more detailed information about the flag. The "Level Two page" information is codified through multicolor boxes called "**subflags".** From this page, it is then possible to open the web interfaces through devoted icons. The user is automatically redirected and the parameters are automatically set.

# 3 The DMS monitoring processes

The DMS part dedicated to the collection of information and the comparison of signals to thresholds is made of a set of various monitoring processes, which create data quality flags whose value depends on the results of the comparison.

## 3.1 The Moni processes

The main monitoring processes are called “Moni processes” and are linked, through a shared memory, to the online data flow coming from the Virgo data acquisition system. All those processes use the Moni library which allows to do simple signal processing, to build the data quality flags and to output the results in a xml file to be read by the WEB tool part of the DMS. Each process has its own configuration, where, according to the lock status of the interferometer, a set of data quality flags are defined on the basis of a set of conditions on well chosen processed signals.

For instance, a configuration line like

*MONITOR 0-11.5 MC_AA 7 "rms(Sc_IB_MC_tx,10)<0.7" "MC autoalignment tx error signal out of range"*

will indicate that, if the ITF locking step is between 0 and 11.5, the bit 7 of the flag MC_AA will be set to 1 if the RMS of the signal Sc_IB_MC_tx, computed over the last 10s, is over the threshold value 0.7. More details on the configurations and the architecture of the Moni processes can be found in the Virgo note VIR-0074A-08.

The online architecture is one Moni process per Virgo sub-system, each taking input data from a shared memory connected to DAQ, each producing a XML file and sending in parallel the DQ flags back to DAQ for storage purpose.

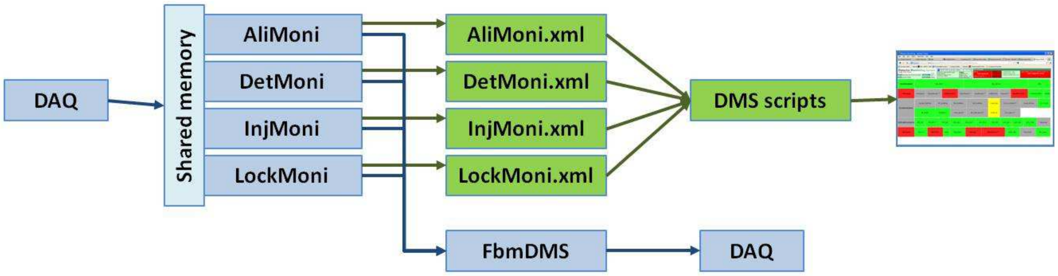


*Fig.1: Architecture of the online Moni processes in the DMS*

## 3.2 BigBrother

An other tool, called BigBrother, is in charge of monitoring several parameters of the various machines used to run the data acquisition, online processing and control of the ITF. It was adapted to provide also this information with data quality flags and in the xml format used by the Moni processes.

## 3.3 AlpDMS

A set of processes, based on the Alp library, are in charge of the automatic control of the interferometer, especially of the control of the automatic lock procedure. One of those processes, called AlpDMS, get information on the status of this control and on the status of the associated Alp processes. It was adapted to provide also this information with data quality flags and in the xml format used by the Moni processes.

## 3.4 Mailbox polling agent

It is a tool set up to be notified from external triggers like:

- Astrophysical trigger alarms from:
  - Gamma Ray Bursts Coordinates Network (GCN) ;
  - SuperNova Early Warning System (SNEWS) neutrino detectors network
- Triple coincidence triggers generated by the LIGO-Virgo pipelines:
  - LUMIN triggers; they are sent to ground based telescopes: this network includes QUEST, ROTSE and TAROT.
  - SWIFT triggers: they are sent to Swift space satellites.

The last triggers are intended to be sent to some optic and radio telescopes to make prompt (ie within ~30 minutes) EM follow-up observations of the interesting triple coincidence GW event candidates. If a counterpart is found, that will significantly increase our confidence in the reality of the event.

Two kind of distinct triple coincidence trigger alerts are available:

The tool was adapted to read this information from specific mailboxes and to provide this information with data quality flags and in the xml format used by the Moni processes.

More details can be found on the "ITF procedure":

- Alert from LUMIN and SWIFT (triple coincidences);
- Alert from SN and GRB warning systems;

In the appendix section there are useful links to know more abut this external triggers.

# 4 Alarm handling philosophy

The proper handling of alarms is crucial to any automated process control. In practice, many alarms are only distractive and do not represent a potentially dangerous situation. In principle, every alarm that is not followed by an action is a nuisance, but it is mainly up to the operator to make that judgment. Many alarms are activated due to too narrowly set alarm limits, and if this feature is detected many alarms can be removed by simply adjusting the alarm limits. An alarm system that generates too many alarms brings situations where the operators do not get enough time for preventive supervision and control. Instead they react only on the alarms that are activated. An alarm that constantly activates incorrectly, may lead to the point where the operators ignore it, and thus fails to bring attention to a hazardous situation. The easiest and most common way to reduce alarms is an appropriate tuning of the alarm limits; there could be also some other techniques to reduce the alarm activation.

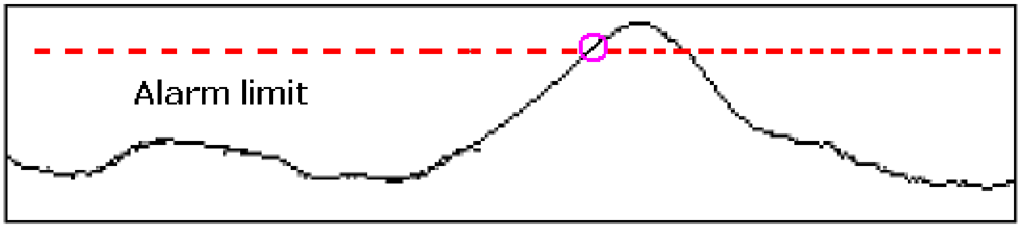


*Fig. 2: Badly tuned alarm limit can cause alarms even in the state of normal operation*.

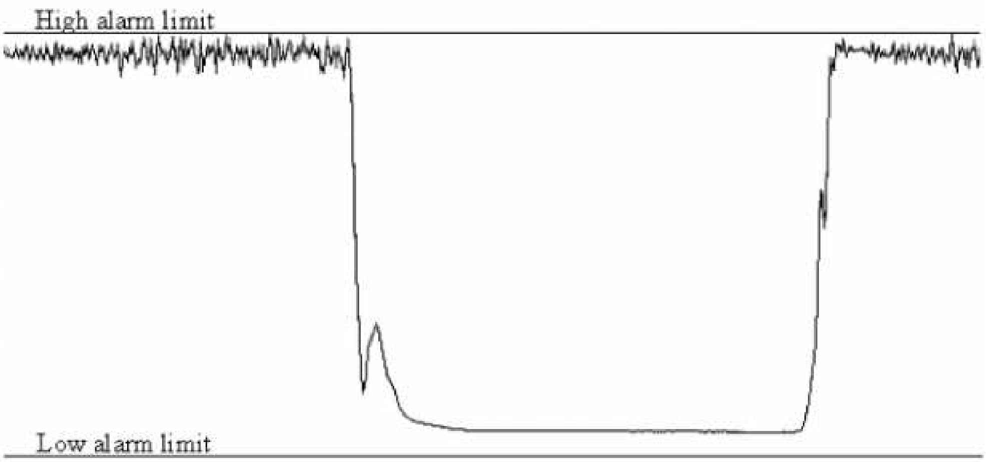


*Fig. 3: The high alarm limit is set too tight and is therefore sensitive to noise. The low alarm limit on the other hand is set too wide, and does not activate even though a problem occurs.*

This requires to process and filter the alarms before they come to the users. We decided to do this at three levels: a pre-filtering, a filtering for the display and a filtering for the notification.

## 4.1 Alarm pre-filtering: "Aggregated flags"

It is a new flag created as a result of a logic expression among other flags. It can be applied only to the flags provided by the various providers (see paragraph 6, *"Diagram block"*). The available logic operators are: *"OR"*, *"AND"*. The new created flag is treated as a standard flag and therefore it can have associated all the functionalities. The status of the new flag is computed following this multicolor truth table:

| Input flag | | Output flag | |
|---|---|---|---|
| **Flag 1** | **Flag 2** | **Flag 1 AND Flag 2** | **Flag 1 OR Flag 2** |
| 🟩 | 🟩 | 🟩 | 🟩 |
| 🟩 | 🟨 | 🟩 | 🟨 |
| 🟩 | ⬜ | 🟩 | ⬜ |
| 🟩 | ⬛ | 🟩 | ⬛ |
| 🟩 | 🟥 | 🟩 | 🟥 |
| 🟨 | 🟨 | 🟨 | 🟨 |
| 🟨 | ⬜ | 🟨 | ⬜ |
| 🟨 | ⬛ | 🟨 | ⬛ |
| 🟨 | 🟥 | 🟨 | 🟥 |
| ⬜ | ⬜ | ⬜ | ⬜ |
| ⬜ | ⬛ | ⬜ | ⬛ |
| ⬜ | 🟥 | ⬜ | 🟥 |
| ⬛ | ⬛ | ⬛ | ⬛ |
| ⬛ | 🟥 | ⬛ | 🟥 |
| 🟥 | 🟥 | 🟥 | 🟥 |

## 4.2 Alarm Suppression: *"Shelving"*

Generally alarm shelving is used in dealing with repeating alarms and is a facility for manually or automatically removing an alarm from the main list and placing it on a shelve list, temporarily preventing the alarm from re-occurring on the main list until it is removed from the shelf.

In our case shelved flag will go into a dedicated shelved list and automatically removed from the main page of DMS.

This kind of feature can be applied both at the level of overall flag and at level of subflags.

## 4.3 Alarm filtering for displaying

### 4.3.1 Filtering 1: "*Timedelay"*

Time-delay is one of the most commonly used alarm reduction methods. It is available in many control systems. Generally this function holds back the alarm for a fixed time, and if the signal does not return to normal operation within that time, the alarm is activated.

In our case the alarm is the red flag and the fixed time is the time before displaying the flag as red. This functionality avoids cases of blinking red flags in the main page of DMS which generally depreciate the user attention.

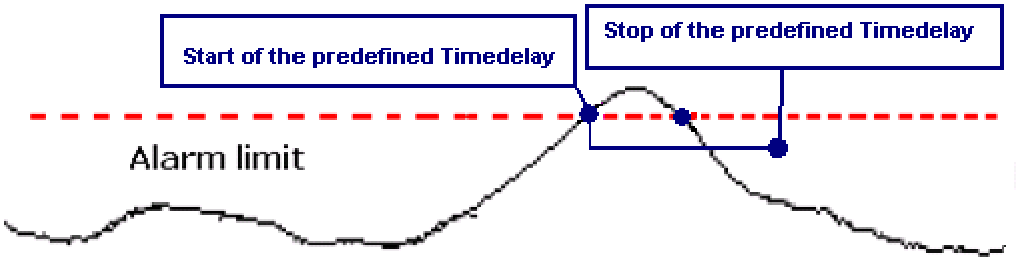

Start of the predefined Timedelay
Stop of the predefined Timedelay
Alarm limit

*Fig. 4: Badly tuned alarm limit can cause alarms even in the state of normal operation; this kind of alarm activation can be eliminated using the Timedelay filtering.*

### 4.3.2 Filtering 2: "*Depend"*

The "Depend" functionality is a method to handle alarm flood; such method is based on cause-consequence analysis. This functionality introduces alarm priorities which are represented trough different colors. The alarms which are the real cause of the problem are displayed as red flag, the alarm which are consequence of the previous problem are displayed as 'semitransparent' red flags.

The aim of this feature is to make easier the identification of the source of the problem focusing the user attention on the red flags considering them as the source of the problem and considering the 'semitransparent' red flags as an induced problem.

### 1.1.3 Filtering 3: "Group"

This functionality allows grouping a set of flags with different names and different associated functionalities inside a group. The status of the group depends by the status of the flags which constitutes that group: the status of the group is computed as the logic "OR" between the flags.

## 4.4 Alarm filtering for notification

### 4.4.1 Filtering 1: *“Persistence”*

The persistence time is a parameter which defines the amount of time that an alarm must be constantly active (the flag must be constantly red) before being notified. This parameter is completely independent from the others filtering (in particular from the Timedelay) used to display the flags.

### 4.4.2 Filtering 2: *“Muting”*

When a flag is muted the alarm notification is not executed even if the flag would have been triggered the notification. The filtering avoids receiving useless notification.

## 4.5 Alarm notification

Once the alarm has been activated (the flag has became red) there is the possibility to notify this alarm in different way:

- through E-MAIL;
- through SOUND;
- through SMS;

The first two ways are used to notify a remote user while the last one is used to notify the user from its local PC.

### 4.5.1 E-MAIL notification

The email notification is performed using the PHP function: “*mail*” (for more details about this function look to the official web site: http://it1.php.net/manual/en/function.mail.php). The address, the subject, and the text of the mail are built by a PHP script (see next paragraphs for more details) : the address is read from dedicated configuration files; the subject and text of the mail report the problem of the red flag.

Below there is an example of a sent message.

```
From:       DMS@olserver13.virgo.infn.it
Subject:    WARNING - Ce_Building_ALARM
Date:       Mon, January 28, 2013 13:50
To:         xxx.xxx@ego-gw.it,yyy.yyyi@ego-gw.it,zzz.zzz@ego-gw.it
________________________________________________________________________________

* "24 > mean(Em..TEBDCE11,10) > 18" ; v=17.411 ; "Temperature_out_of_range"
```

### 4.5.2 SOUND notification

A sound is executed by loudspeaker of PC, together with BLINKING TEXT of the flag on the main page of the DMS. The sound to be played is defined in the dedicated configuration file and played by a flash mp3 player: Dewplayer **(**http://www.alsacreations.fr/dewplayer-en.html**).**

### 4.5.3 SMS notification

The SMS notification is executed by a dedicated SMS-server called *"FoxBox G20"* ( for more details on this SMS server look at the WEB site: http://www.smsfoxbox.it/) installed on site. This SMS-server was configured and equipped by an own SIM card to send SMS. The phone number and the text of the SMS are built by a PHP script (see next paragraphs for more details): the phone number is read from dedicated configuration files; the text of the SMS reports the problem of the red flag.

Below there is an example of a sent message.

```
From: DMS
Text: WARNING - Ce_Building_ALARM 24 > mean(Em..TEBDCE11,10) > 18" ; v=17.411
      ; "Temperature_out_of_range"
```

# 5 The system architecture

## 5.1 How does it work

As previously described, specific monitoring tools (Moni processes, BigBrother, etc …) read information from the DAQ and perform well defined mathematical/logic operations on data to produce information written in XML files with a predefined structure.

A dedicate PHP script (named *server_db.php*), according to its configuration file, reads these XML files and it translates the information written in the XML files into a MySQL database.

Another dedicated PHP script (named *server_DMS.php*) reads the data stored inside the db and according to its configuration file it implements a set of functionalities and it performs all the necessary operations to fit the requirements.

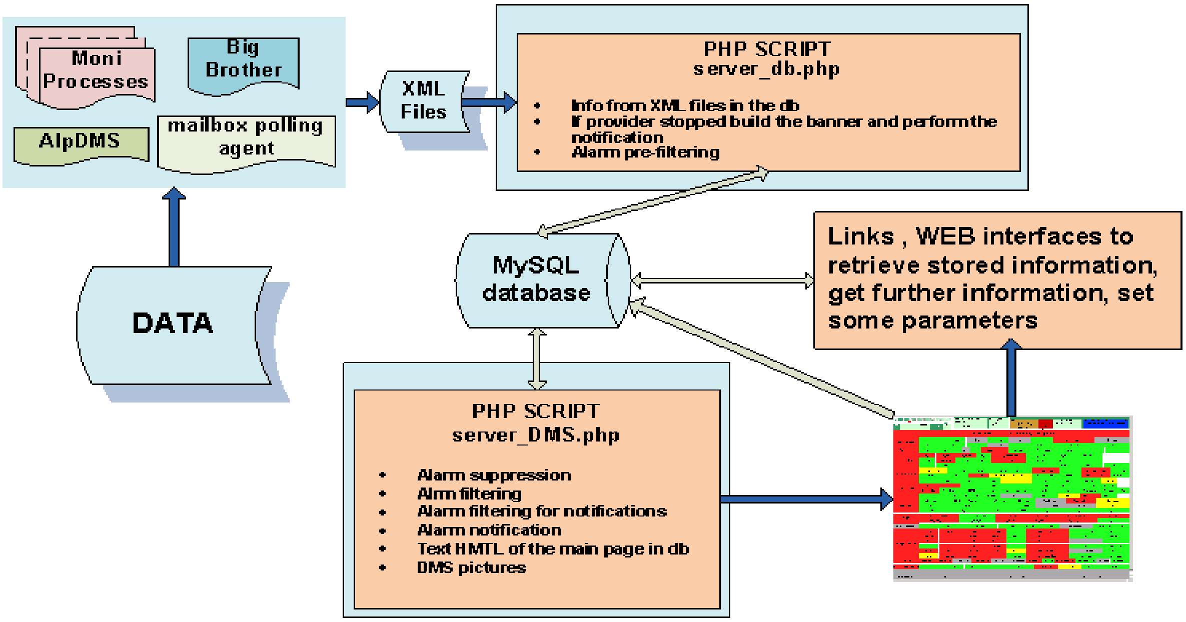

Moni Processes
Big Brother
AlpDMS
mailbox polling agent
XML Files
PHP SCRIPT
s erver_db.php
Info from XML files in the db
If provider stopped build the banner and performthe notification
Alarm pre-filtering
DATA
MySQL database
Links , WEB interfaces to retrieve stored information, get further information, set some parameters
PHP SCRIPT
server_DMS.php
Alarm suppression
Alrm filtering
Alarm filtering for notifications
Alarm notification
Text HMTL of the main page in db
DMS pictures

*Fig. 5: General structure of the system.*

## 5.2 Diagram block

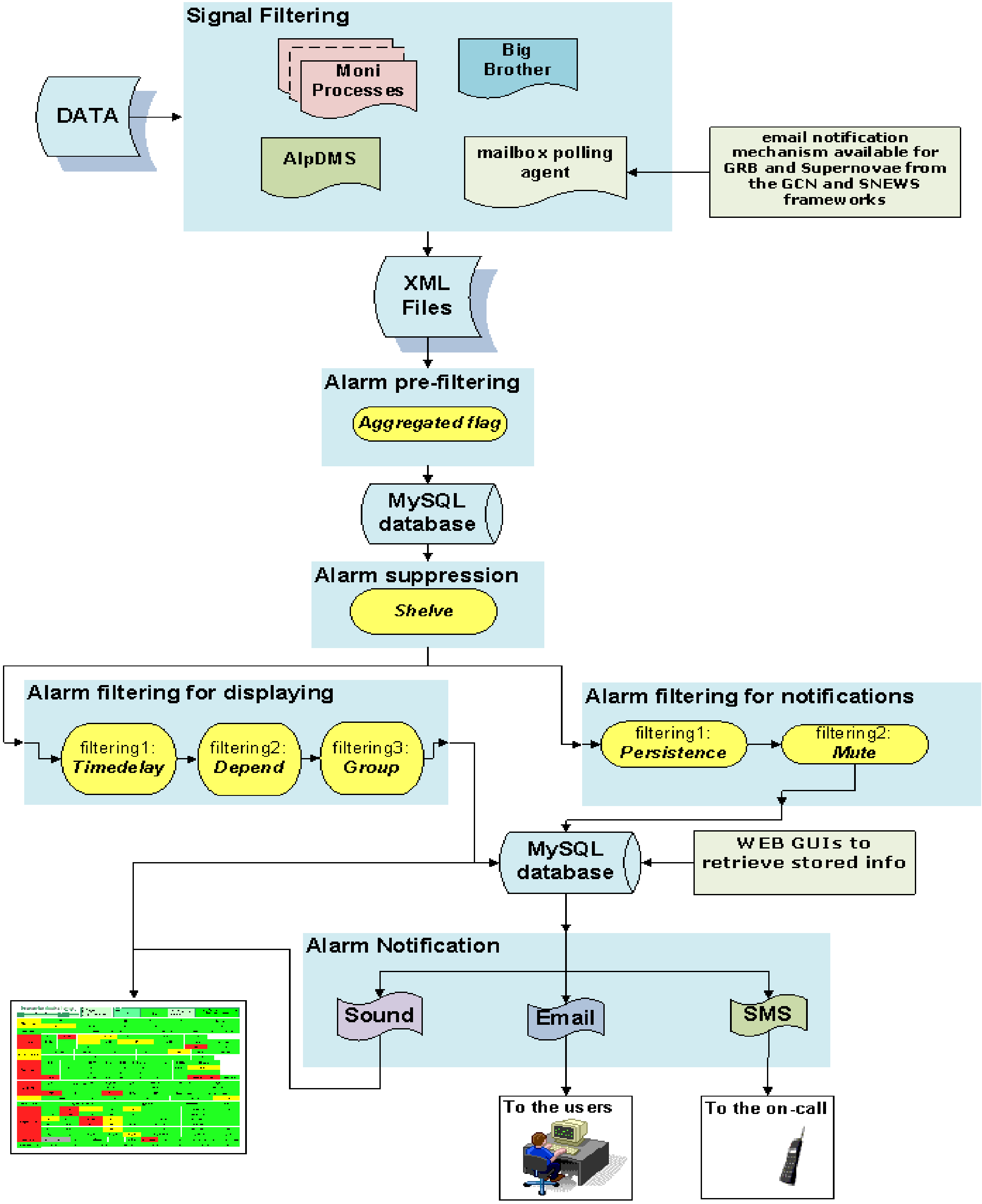


*Fig.6: Sequence of the implemented techniques in order to have a better flag displaying and to reduce as much as possible false alarm activations.*

## 5.3 Setting the system configurations

### 5.3.1 Configuration file interpretation

The keywords used to identify the various functionalities are:

- ROWS_HEIGHT(arg)
- TIMEDELAY(arg1, arg2, …., …)
- DEPEND(arg1, arg2, …., …)
- GROUP(arg1, arg2, …., …)
- EMAIL(arg1, arg2, …., …)
- SOUND(arg1, arg2, …., …)
- SMS(arg1, arg2, …., …)

The parameters related to a keyword are passed as arguments in brackets just after the keyword name and are separated by a comma. This kind of structure makes the system very flexible because in the future a new feature, new parameters can be added in a modular way. Moreover at the same flag can be associated more than one feature: a flag can be inside a group, can have both "alarm sound" and "alarm e-mail", etc…

The *ROWS_HEIGHT* functionality is used to specify the height expressed in pixel of the rows which constitute the subsystem.

The *TIMEDELAY*, *DEPEND*, *GROUP* functionalities are used to filter the flag displaying; the *EMAIL*, *SOUND*, *SMS* functionalities are used to notify the alarm events.

### 5.3.2 Shelving and Muting

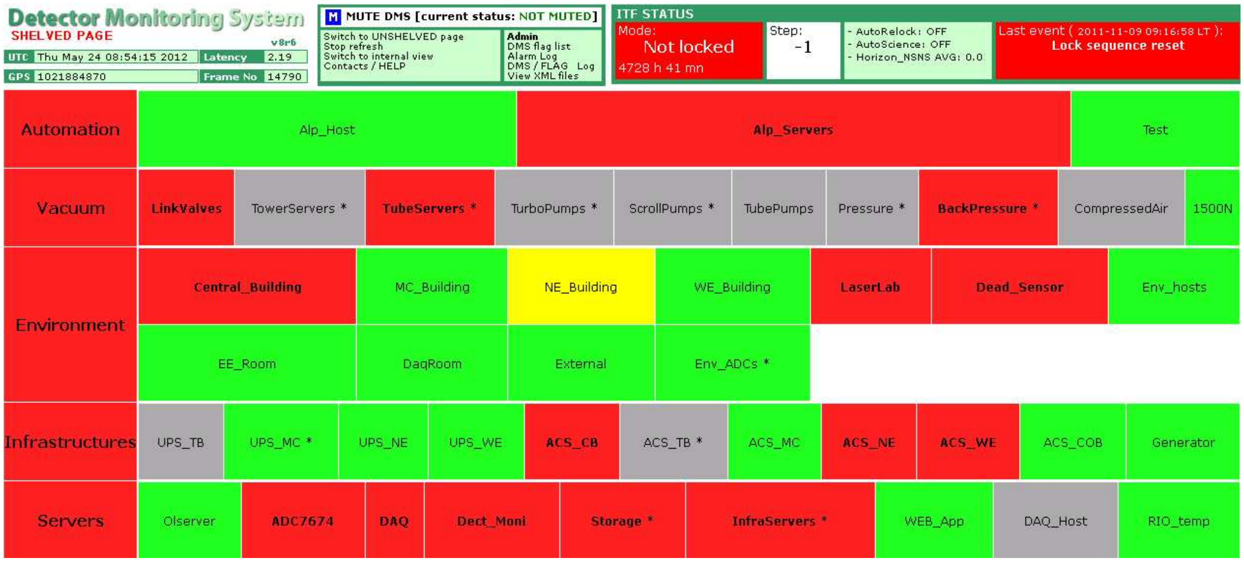
The *SHELVING* functionality is used to reduce false alarm activations; the *MUTING* functionality is used to filter the alarm notifications.

# 6 The "Main Page"

https://pub3.ego-gw.it/itf/detOp/DMS/index_DMS_ShelvedVersion.php?s=&pause=0

The DMS main page is composed by two sections:

- the header which shows general information, links to the other WEB interfaces and information about the ITF status (ITF Mode, step, events…);
- the body which shows the information codified with colored boxes.



Detector Monitoring System
SHELVED PAGE
v8r6
UTC Thu May 24 08:54:15 2012 | Latency 2.19
GPS 1021884870 | Frame No 14790

M MUTE DMS [current status: NOT MUTED]
Switch to UNSHELVED page
Stop refresh
Switch to internal view
Contacts / HELP

Admin
DMS flag list
Alarm Log
DMS / FLAG Log
View XML files

ITF STATUS
Mode: Not locked 4728 h 41 mn
Step: -1
- AutoRelock: OFF
- AutoScience: OFF
- Horizon_NSNS AVG: 0.0
Last event ( 2011-11-09 09:16:58 LT ): Lock sequence reset

Automation: Alp_Host, Alp_Servers, Test
Vacuum: LinkValves, TowerServers *, TubeServers *, TurboPumps *, ScrollPumps *, TubePumps, Pressure *, BackPressure *, CompressedAir, 1500N
Environment: Central_Building, MC_Building, NE_Building, WE_Building, LaserLab, Dead_Sensor, Env_hosts, EE_Room, DaqRoom, External, Env_ADCs *
Infrastructures: UPS_TB, UPS_MC *, UPS_NE, UPS_WE, ACS_CB, ACS_TB *, ACS_MC, ACS_NE, ACS_WE, ACS_COB, Generator
Servers: Olserver, ADC7674, DAQ, Dect_Moni, Storage *, InfraServers *, WEB_App, DAQ_Host, RIO_temp

*Fig. 7: DMS main page.*

As described in 4.1 the page is built through two PHP scripts: one named server_db.php and the other one named server_DMS.php.

The server_db.php script, according to its configuration file, reads the XML files and it translates the information written in the XML files into a MySQL db. In case the XML file is not being updated by the provider, the PHP script builds a text banner to denote the problem and in case it can perform the alarm notification.

In this section the pre-filtering of the aggregated flag is also implemented.

The script server_DMS.php, according to its configuration file, reads the information about flags and subflags written in the MySQL db by the script serve_db.php.

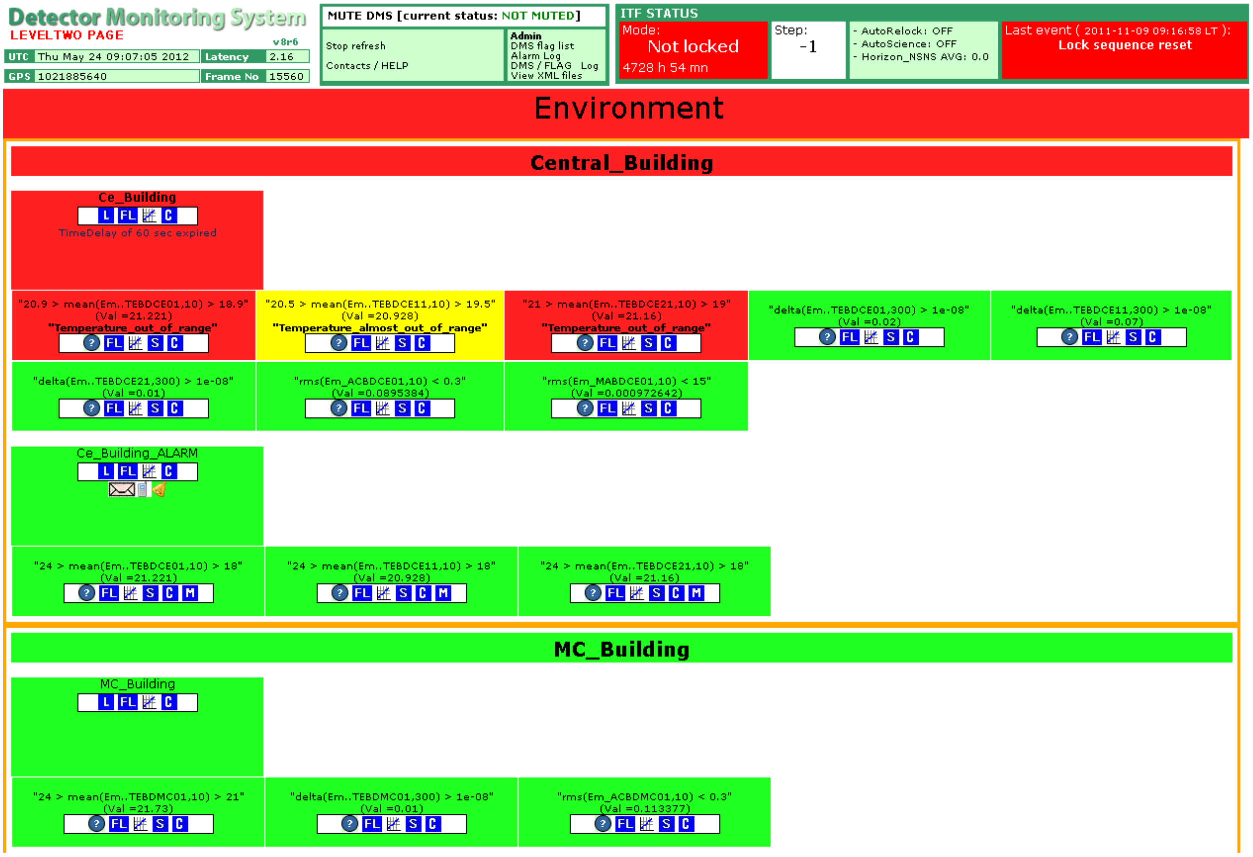
Once read, the server_DMS.php script performs all the necessary operations to implement the functionalities, to build the HTML text which constitutes the main page of the DMS and to manage the alarm notification.

The HTML text written by the script server_DMS.php is then read from a client side.

The client takes information about ITF Mode, Step, etc … from another database and it adds this kind of information to the previous HTML text. The WEB page, opened by any WEB browser, it is thus main page of the DMS.

## 7 The "Level Two Page"

The Level Two page has the same structure of the Main Page, the header and the main page that shows the flags in detail: status of the flag, name of the channel monitored, thresholds and current value; it is possible to interact with the single flag using the graphical link below its to perform the shelving/muting, to leave a comment or to display a plot.



Detector Monitoring System
LEVELTWO PAGE
v8r6
UTC Thu May 24 09:07:05 2012 | Latency 2.16
GPS 1021885640 | Frame No 15560

MUTE DMS [current status: NOT MUTED]
Stop refresh
Contacts / HELP
Admin
DMS flag list
Alarm Log
DMS / FLAG Log
View XML files

ITF STATUS
Mode: Not locked
4728 h 54 mn
Step: -1
- AutoRelock: OFF
- AutoScience: OFF
- Horizon_NSNS AVG: 0.0
Last event ( 2011-11-09 09:16:58 LT ): Lock sequence reset

Environment

Central_Building

Ce_Building
TimeDelay of 60 sec expired

"20.9 > mean(Em..TEBDCE01,10) > 18.9"
(Val =21.221)
"Temperature_out_of_range"

"20.5 > mean(Em..TEBDCE11,10) > 19.5"
(Val =20.928)
"Temperature_almost_out_of_range"

"21 > mean(Em..TEBDCE21,10) > 19"
(Val =21.16)
"Temperature_out_of_range"

"delta(Em..TEBDCE01,300) > 1e-08"
(Val =0.02)

"delta(Em..TEBDCE11,300) > 1e-08"
(Val =0.07)

"delta(Em..TEBDCE21,300) > 1e-08"
(Val =0.01)

"rms(Em_ACBDCE01,10) < 0.3"
(Val =0.0895384)

"rms(Em_MABDCE01,10) < 15"
(Val =0.000972642)

Ce_Building_ALARM

"24 > mean(Em..TEBDCE01,10) > 18"
(Val =21.221)

"24 > mean(Em..TEBDCE11,10) > 18"
(Val =20.928)

"24 > mean(Em..TEBDCE21,10) > 18"
(Val =21.16)

MC_Building

MC_Building

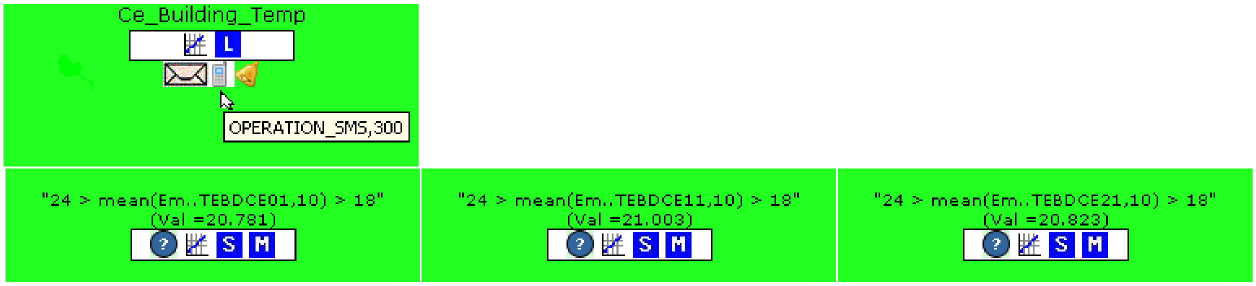
"24 > mean(Em..TEBDMC01,10) > 21"
(Val =21.73)

"delta(Em..TEBDMC01,300) > 1e-08"
(Val =0.01)

"rms(Em_ACBDMC01,10) < 0.3"
(Val =0.113377)

*Fig. 8: DMS level two page. This page is opened by clicking on the flag "Central_Building" in the main page. In this page we can see additional info of the flag "Central_Building".*

The WEB page can be opened by clicking over a flag on the main page of DMS; from the opened WEB page it is possible to have additional info about the flags and clicking over the icons it is possible to reach others WEB pages.



Ce_Building_Temp

OPERATION_SMS,300

"24 > mean(Em..TEBDCE01,10) > 18"
(Val =20.781)

"24 > mean(Em..TEBDCE11,10) > 18"
(Val =21.003)

"24 > mean(Em..TEBDCE21,10) > 18"
(Val =20.823)

*Fig. 9: Detail of flag in the DMS level two page. We can see the subflags which constitute the main flags and all the possible icons that allows to reach other WEB interfaces.*

Details of the icons:

- 📈 clicking on it is possible to open a plot which show the evolution of the status (the color) of the overall flag or the evolution of the signal of the subflags.
- L clicking on it possible to open the main page of the *DMSLog* (see paragraph 9.3, *“DMSLog”* for more details) and the parameters are automatically to set to search the red status of the flag in the last five days.
- ✉ this icon indicates that the flag is alarmed and it can perform the alarm notification sending an email; keeping the mouse cursor over this icon a *JavaScript tooltip* is opened and it shows information about the alarm configurations.
- 📱 this icon indicates that the flag is alarmed and it can perform the alarm notification sending SMS; keeping the mouse cursor over this icon a *JavaScript tooltip* is opened and it shows information about the alarm configurations.
- 🔔 this icon indicates that the flag is alarmed and it can perform the alarm notification playing a sound and blinking; keeping the mouse cursor over this icon a *JavaScript tooltip* is opened and it shows information about the alarm configurations.
- ? clicking on this icon a link we are automatically redirect to the *“ITF Procedures*” (https://pub3.ego-gw.it/procedures/index.php?areaid=1); in particular we are automatically redirected to the procedure of that flag.

  If the procedure is not present the user can create it after authentication.
- S clicking on this icon it is possible to open a popup-window to set shelving parameters;

- M clicking on this icon it is possible to open a popup-window to set shelving parameters;

For more information about these WEB interfaces see the next paragraphs.

- C clicking on this icon it is possible to open a popup-window to leave a comment.
- FL clicking on it possible to open the main page of the *DMS Flag Log* and the parameters are automatically to set to search all the "events" for the flag in the last twenty.

# 8 WEB interfaces

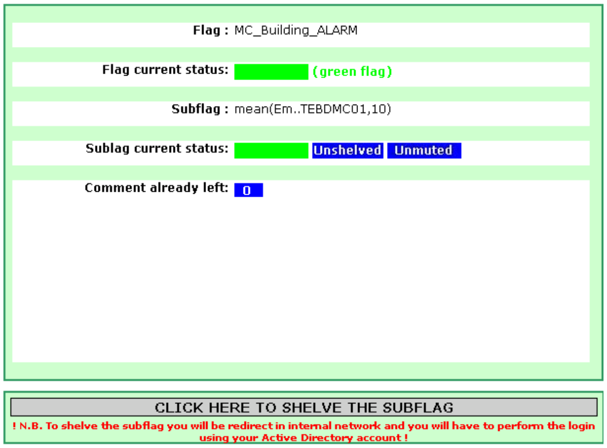
Some web interfaces have been developed to allow the user to quickly and friendly retrieve information.

## 8.1 Admin interface

The admin interface is reserved to the system administrator. After Active Directory authentication the administrator, using an internal database, can define groups of users "*allowed*", *"not allowed"* to act (shelve, mute) on the flag.

## 8.2 The "Shelving interface"

This interface shows the flag status, the shelving/muting parameters and comments; after the standard Active Directory authentication the user can modify the shelving start and stop time, the trigger (grey, red or all) and leave a comment.



**Flag :** MC_Building_ALARM

**Flag current status:** 🟩 (green flag)

**Subflag :** mean(Em..TEBDMC01,10)

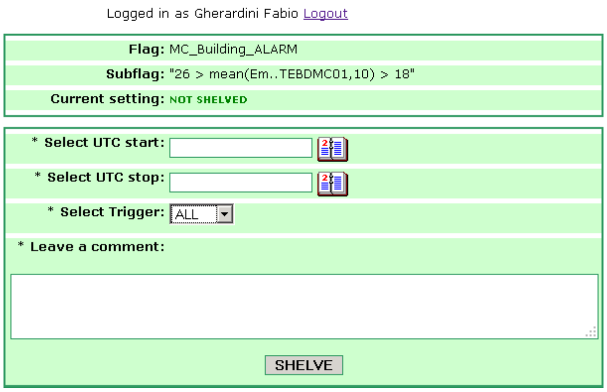
**Sublag current status:** 🟩 Unshelved Unmuted

**Comment already left:** 0

CLICK HERE TO SHELVE THE SUBFLAG

! N.B. To shelve the subflag you will be redirect in internal network and you will have to perform the login using your Active Directory account !

*Fig. 10a: The shelving interface. This page can be see by everyone (no login is necessary) and it shows info detailed info about the subflag: "mean(Em..TEBDMC01,10)*



Logged in as Gherardini Fabio Logout

**Flag:** MC_Building_ALARM

**Subflag:** "26 > mean(Em..TEBDMC01,10) > 18"

**Current setting:** NOT SHELVED

* **Select UTC start:**

* **Select UTC stop:**

* **Select Trigger:** ALL

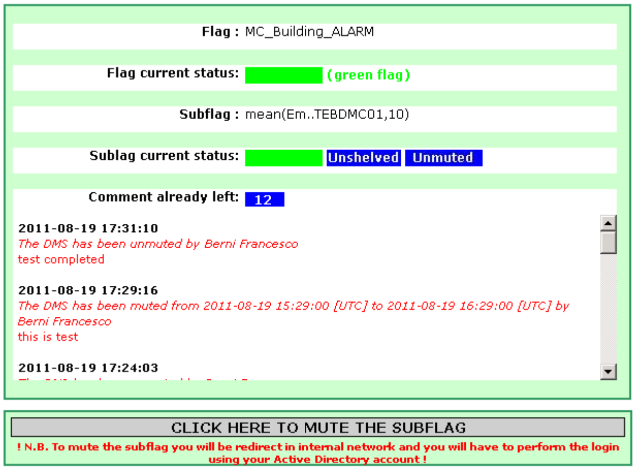
* **Leave a comment:**

SHELVE

*Fig. 10b : The shelving interface. After Active Directory authentication the user can act on the shelving settings.*

## 8.3 The “Muting interface”

This interface shows the flag status, the shelving/muting parameters and comments; after Active Directory authentication the user can see the alarm configuration, modify the muting settings (start and stop time, etc…), and leave a new comment.



**Flag :** MC_Building_ALARM

**Flag current status:** 🟩 (green flag)

**Subflag :** mean(Em..TEBDMC01,10)

**Sublag current status:** 🟩 Unshelved Unmuted

**Comment already left:** 12

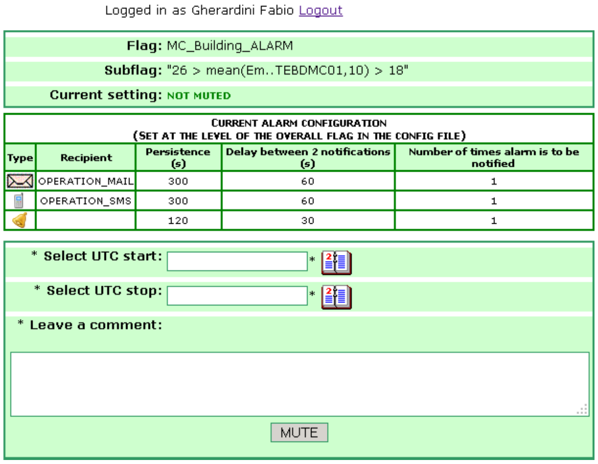
**2011-08-19 17:31:10**
*The DMS has been unmuted by Berni Francesco*
test completed

**2011-08-19 17:29:16**
*The DMS has been muted from 2011-08-19 15:29:00 [UTC] to 2011-08-19 16:29:00 [UTC] by Berni Francesco*
this is test

**2011-08-19 17:24:03**

**! N.B. To mute the subflag you will be redirect in internal network and you will have to perform the login using your Active Directory account !**

*Fig. 11a: The muting interface. This page can be see by everyone (no login is necessary) and it shows info detailed info about the subflag: “mean(Em..TEBDMC01,10) .*



Logged in as Gherardini Fabio Logout

**Flag:** MC_Building_ALARM

**Subflag:** "26 > mean(Em..TEBDMC01,10) > 18"

**Current setting:** NOT MUTED

CURRENT ALARM CONFIGURATION
(SET AT THE LEVEL OF THE OVERALL FLAG IN THE CONFIG FILE)

| Type | Recipient | Persistence (s) | Delay between 2 notifications (s) | Number of times alarm is to be notified |
|---|---|---|---|---|
| ✉ | OPERATION_MAIL | 300 | 60 | 1 |
| 📱 | OPERATION_SMS | 300 | 60 | 1 |
| 🔔 | | 120 | 30 | 1 |

* **Select UTC start:** *

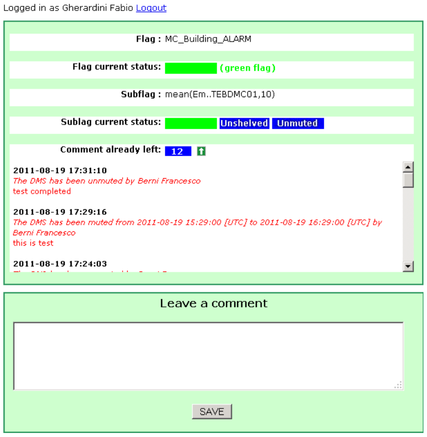
* **Select UTC stop:** *

* **Leave a comment:**

*Figure 11b: The muting interface. After Active Directory authentication the user can act on the shelving settings*

## 8.4 The “comment interface”

This interface shows the flag status, the shelving/muting parameters and comments; after authentication the user can leave a new comment.



Logged in as Gherardini Fabio Logout

**Flag :** MC_Building_ALARM

**Flag current status:** 🟩 (green flag)

**Subflag :** mean(Em..TEBDMC01,10)

**Sublag current status:** 🟩 Unshelved Unmuted

**Comment already left:** 12

**2011-08-19 17:31:10**
*The DMS has been unmuted by Berni Francesco*
test completed

**2011-08-19 17:29:16**
*The DMS has been muted from 2011-08-19 15:29:00 [UTC] to 2011-08-19 16:29:00 [UTC] by Berni Francesco*
this is test

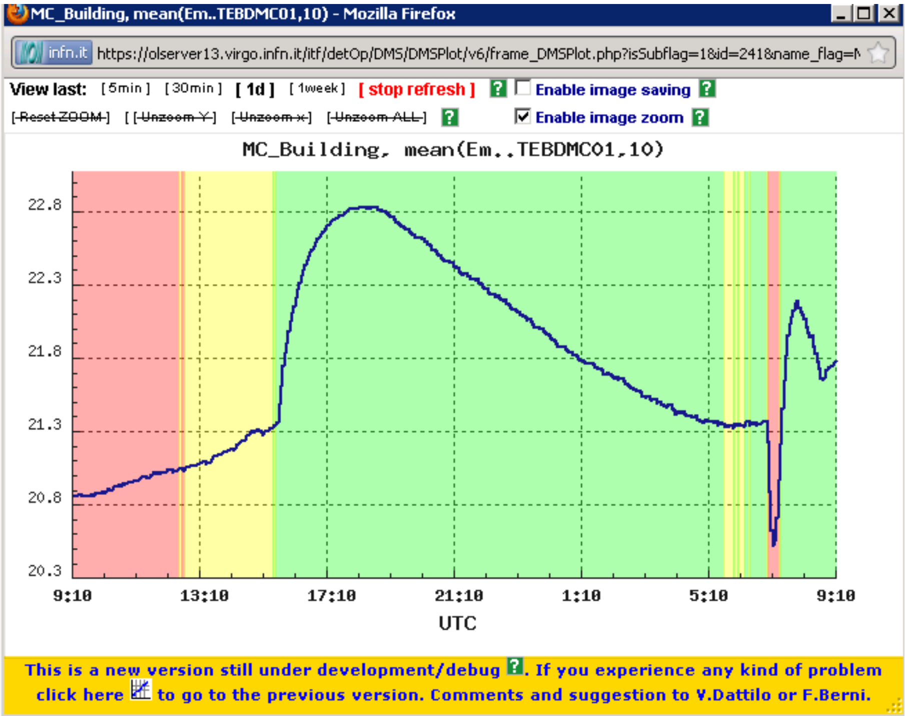
**2011-08-19 17:24:03**

Leave a comment

*Fig. 12: The comment interface.*

## 8.5 The "DMS Plot"

The plot show the evolution of the status (the color) and the signal of the flag, the background color follows the color of the flag; there is the possibility to interact with the plot: change the duration, zoom/unzoom the axis and save the image.



MC_Building, mean(Em..TEBDMC01,10) - Mozilla Firefox

infn.it https://olserver13.virgo.infn.it/itf/detOp/DMS/DMSPlot/v6/frame_DMSPlot.php?isSubflag=1&id=241&name_flag=N

View last: [5min] [30min] **[1d]** [1week] [stop refresh] ? □ Enable image saving ?

[~~Reset ZOOM~~] [~~[Unzoom Y~~] [~~Unzoom x~~] [~~Unzoom ALL~~] ? ☑ Enable image zoom ?

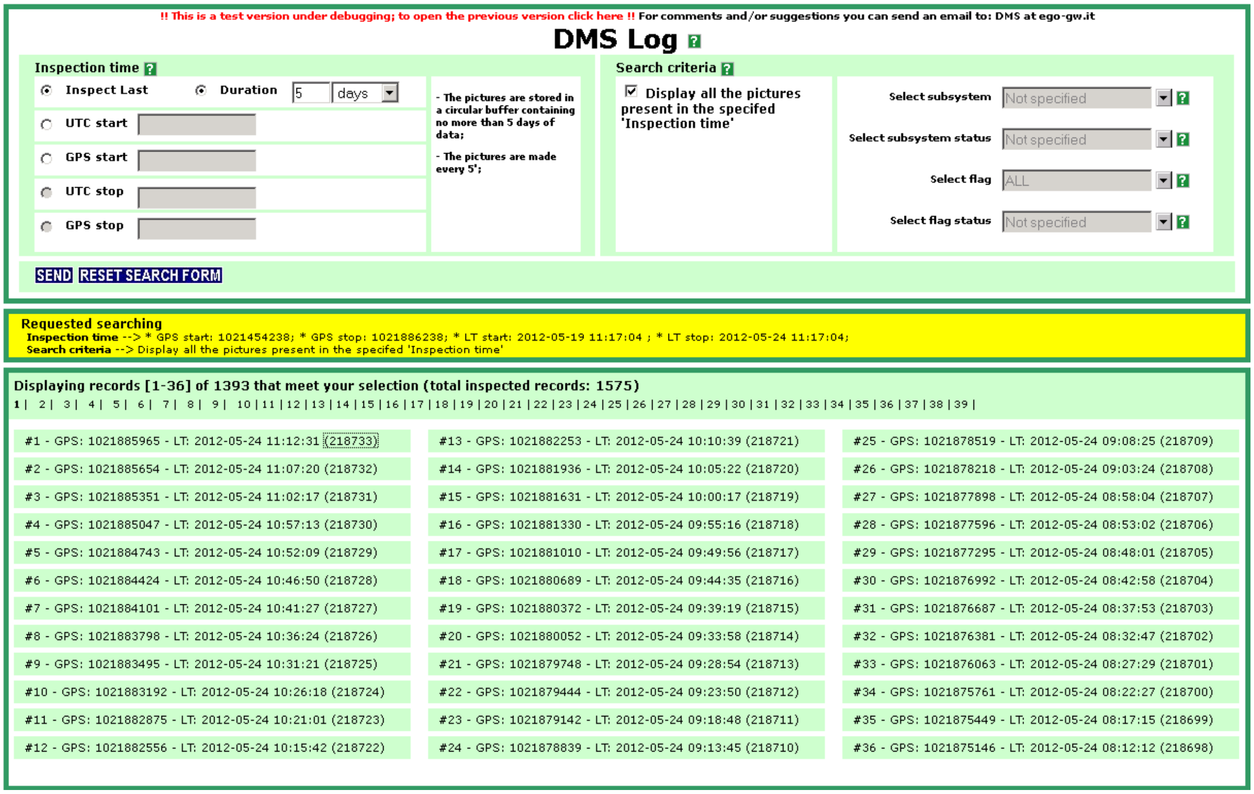
This is a new version still under development/debug ?. If you experience any kind of problem click here to go to the previous version. Comments and suggestion to V.Dattilo or F.Berni.

*Fig. 13: DMS Plot. This page shows the evolution of the signal "mean(Em..TEBDMC01,10) which is a subflag of the flag MC_Building. The background of the plot reproduce the evolution of the color of the subflag.*

## 8.6 The "*DMS Log*"

https://pub3.ego-gw.it/itf/detOp/DMS/web_interfaces/DMSLog/v1/DMSLog.php

The "DMS Log" interface allows retrieving the DMS pictures specifying also some search criteria; the detail search provides a variety of searchable options. These have been tailored in such a way as to produce easily manageable and modifiable result sets.



!! This is a test version under debugging; to open the previous version click here !! For comments and/or suggestions you can send an email to: DMS at ego-gw.it

DMS Log ?

Inspection time ?

● Inspect Last ● Duration 5 days

○ UTC start

○ GPS start

○ UTC stop

○ GPS stop

- The pictures are stored in a circular buffer containing no more than 5 days of data;

- The pictures are made every 5';

Search criteria ?

☑ Display all the pictures present in the specifed 'Inspection time'

Select subsystem: Not specified

Select subsystem status: Not specified

Select flag: ALL

Select flag status: Not specified

Requested searching
**Inspection time** --> * GPS start: 1021454238; * GPS stop: 1021886238; * LT start: 2012-05-19 11:17:04 ; * LT stop: 2012-05-24 11:17:04;
**Search criteria** --> Display all the pictures present in the specifed 'Inspection time'

Displaying records [1-36] of 1393 that meet your selection (total inspected records: 1575)

1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 | 21 | 22 | 23 | 24 | 25 | 26 | 27 | 28 | 29 | 30 | 31 | 32 | 33 | 34 | 35 | 36 | 37 | 38 | 39 |

```
#1 - GPS: 1021885965 - LT: 2012-05-24 11:12:31 (218733)
#2 - GPS: 1021885654 - LT: 2012-05-24 11:07:20 (218732)
#3 - GPS: 1021885351 - LT: 2012-05-24 11:02:17 (218731)
#4 - GPS: 1021885047 - LT: 2012-05-24 10:57:13 (218730)
#5 - GPS: 1021884743 - LT: 2012-05-24 10:52:09 (218729)
#6 - GPS: 1021884424 - LT: 2012-05-24 10:46:50 (218728)
#7 - GPS: 1021884101 - LT: 2012-05-24 10:41:27 (218727)
#8 - GPS: 1021883798 - LT: 2012-05-24 10:36:24 (218726)
#9 - GPS: 1021883495 - LT: 2012-05-24 10:31:21 (218725)
#10 - GPS: 1021883192 - LT: 2012-05-24 10:26:18 (218724)
#11 - GPS: 1021882875 - LT: 2012-05-24 10:21:01 (218723)
#12 - GPS: 1021882556 - LT: 2012-05-24 10:15:42 (218722)
#13 - GPS: 1021882253 - LT: 2012-05-24 10:10:39 (218721)
#14 - GPS: 1021881936 - LT: 2012-05-24 10:05:22 (218720)
#15 - GPS: 1021881631 - LT: 2012-05-24 10:00:17 (218719)
#16 - GPS: 1021881330 - LT: 2012-05-24 09:55:16 (218718)
#17 - GPS: 1021881010 - LT: 2012-05-24 09:49:56 (218717)
#18 - GPS: 1021880689 - LT: 2012-05-24 09:44:35 (218716)
#19 - GPS: 1021880372 - LT: 2012-05-24 09:39:19 (218715)
#20 - GPS: 1021880052 - LT: 2012-05-24 09:33:58 (218714)
#21 - GPS: 1021879748 - LT: 2012-05-24 09:28:54 (218713)
#22 - GPS: 1021879444 - LT: 2012-05-24 09:23:50 (218712)
#23 - GPS: 1021879142 - LT: 2012-05-24 09:18:48 (218711)
#24 - GPS: 1021878839 - LT: 2012-05-24 09:13:45 (218710)
#25 - GPS: 1021878519 - LT: 2012-05-24 09:08:25 (218709)
#26 - GPS: 1021878218 - LT: 2012-05-24 09:03:24 (218708)
#27 - GPS: 1021877898 - LT: 2012-05-24 08:58:04 (218707)
#28 - GPS: 1021877596 - LT: 2012-05-24 08:53:02 (218706)
#29 - GPS: 1021877295 - LT: 2012-05-24 08:48:01 (218705)
#30 - GPS: 1021876992 - LT: 2012-05-24 08:42:58 (218704)
#31 - GPS: 1021876687 - LT: 2012-05-24 08:37:53 (218703)
#32 - GPS: 1021876381 - LT: 2012-05-24 08:32:47 (218702)
#33 - GPS: 1021876063 - LT: 2012-05-24 08:27:29 (218701)
#34 - GPS: 1021875761 - LT: 2012-05-24 08:22:27 (218700)
#35 - GPS: 1021875449 - LT: 2012-05-24 08:17:15 (218699)
#36 - GPS: 1021875146 - LT: 2012-05-24 08:12:12 (218698)
```

*Fig. 14: DMS Log*

The possible searchable areas are:

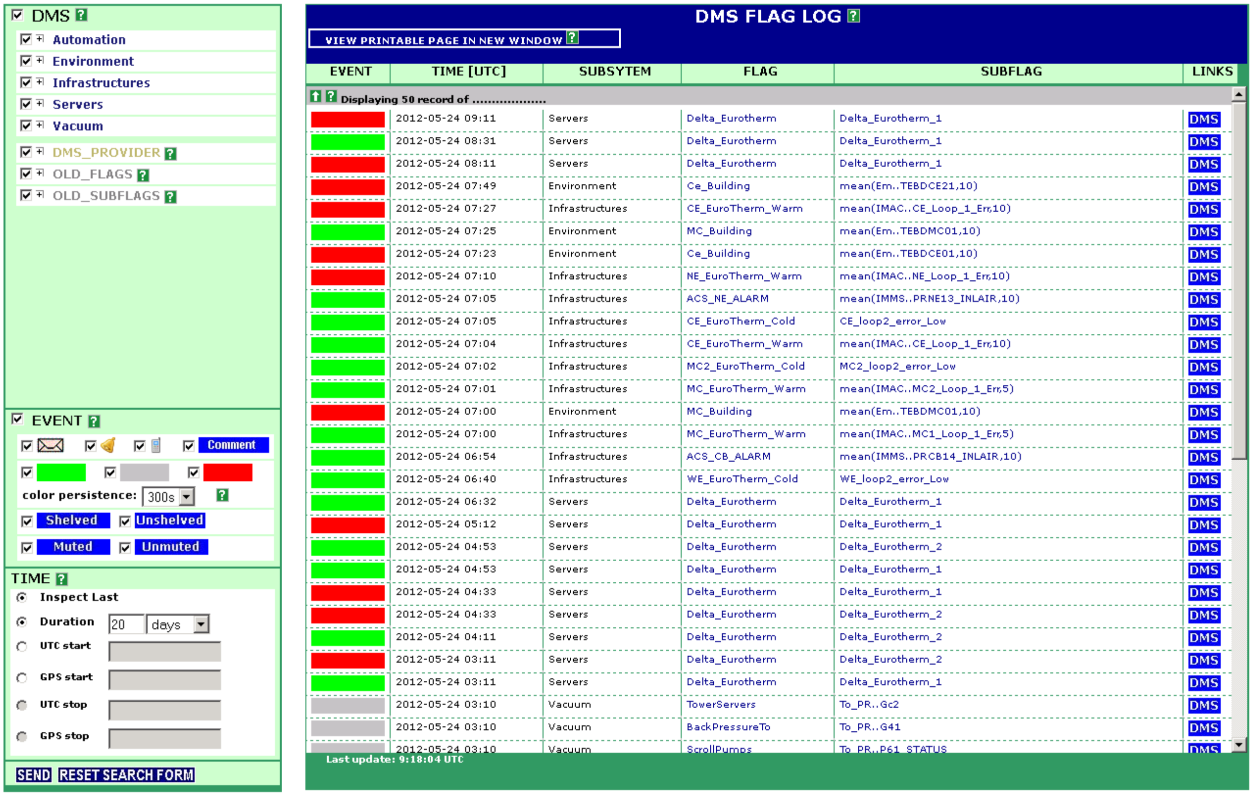
- *Start Time*: to restrict the searching of the snapshots starting from the specified time;
- *Time Inspection*: to express how much time forward (Duration) or backwards (Inspect Last) starting from the Start Time;
- *Search Criteria:* to looks for specific status of the Subsystems/Flags in the image of all photos in the database
- *Subsystem:* to specify a subsystem to be searched
- *Flags:* to specify flag/flags to be searched
- *Status:* to specify the status of the previously selected subsystems/flags

## 8.7 The *"Flag Log"*

The "Flag Log" page shows all the DMS events: events mean changing of a flag/subflag status, alarm notification, shelving/muting and comments ordered by time; clicking the link icon provide the actual

DMS flag status; clicking the event icon in case of shelving/muting and comment provide detailed information about the event.

Through the interface on the left it is possible to perform a search  selecting the flag, the kind of event or specifying the time.



☑ DMS ?

- ☑ + Automation
- ☑ + Environment
- ☑ + Infrastructures
- ☑ + Servers
- ☑ + Vacuum
- ☑ + DMS_PROVIDER ?
- ☑ + OLD_FLAGS ?
- ☑ + OLD_SUBFLAGS ?

☑ EVENT ?

☑ ✉ ☑ 🔔 ☑ 📱 ☑ Comment

☑ 🟩 ☑ ▨ ☑ 🟥

color persistence: 300s ?

☑ Shelved ☑ Unshelved

☑ Muted ☑ Unmuted

TIME ?

● Inspect Last

● Duration 20 days

○ UTC start

○ GPS start

○ UTC stop

○ GPS stop

SEND RESET SEARCH FORM

DMS FLAG LOG ?

VIEW PRINTABLE PAGE IN NEW WINDOW ?

Displaying 50 record of ..................

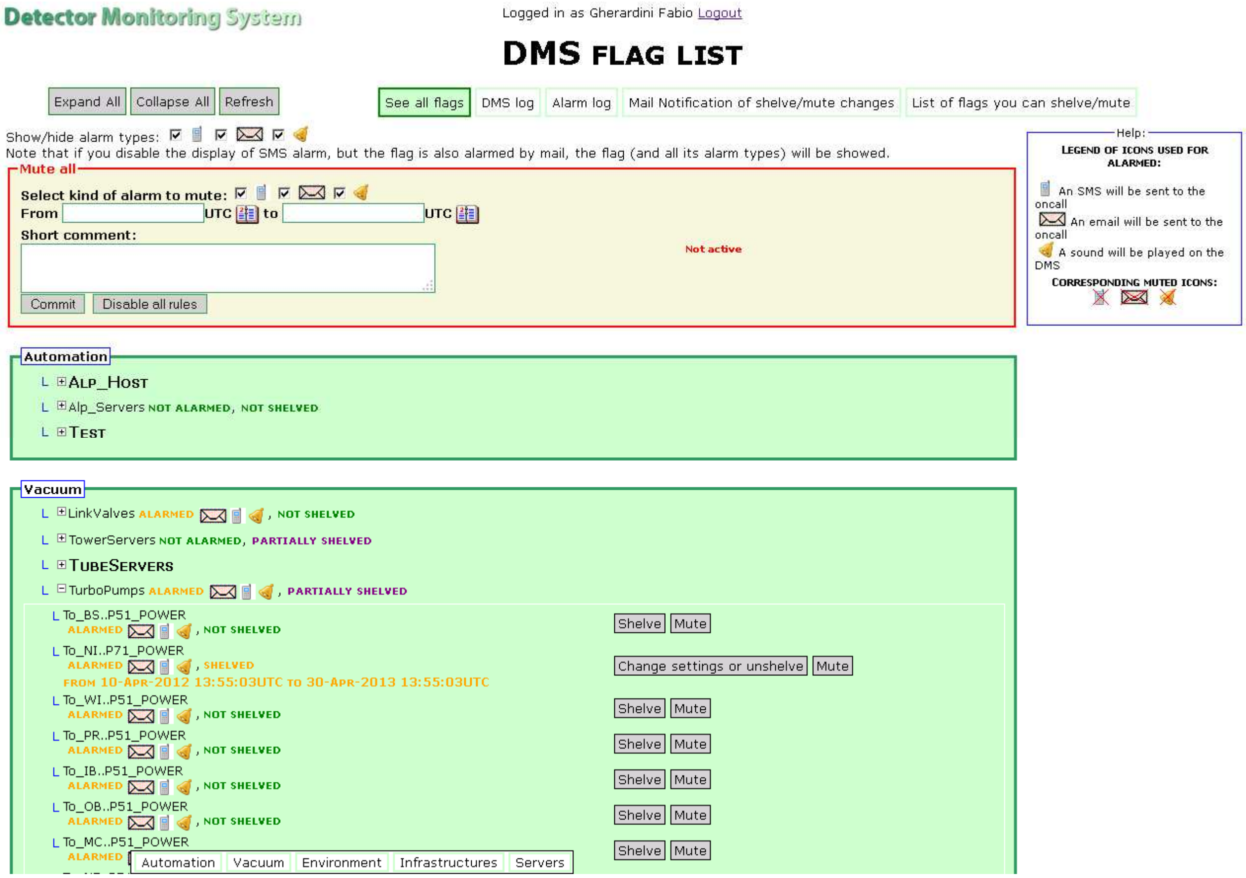
| EVENT | TIME [UTC] | SUBSYTEM | FLAG | SUBFLAG | LINKS |
|---|---|---|---|---|---|
| 🟥 | 2012-05-24 09:11 | Servers | Delta_Eurotherm | Delta_Eurotherm_1 | DMS |
| 🟩 | 2012-05-24 08:31 | Servers | Delta_Eurotherm | Delta_Eurotherm_1 | DMS |
| 🟥 | 2012-05-24 08:11 | Servers | Delta_Eurotherm | Delta_Eurotherm_1 | DMS |
| 🟥 | 2012-05-24 07:49 | Environment | Ce_Building | mean(Em..TEBDCE21,10) | DMS |
| 🟥 | 2012-05-24 07:27 | Infrastructures | CE_EuroTherm_Warm | mean(IMAC..CE_Loop_1_Err,10) | DMS |
| 🟩 | 2012-05-24 07:25 | Environment | MC_Building | mean(Em..TEBDMC01,10) | DMS |
| 🟥 | 2012-05-24 07:23 | Environment | Ce_Building | mean(Em..TEBDCE01,10) | DMS |
| 🟥 | 2012-05-24 07:10 | Infrastructures | NE_EuroTherm_Warm | mean(IMAC..NE_Loop_1_Err,10) | DMS |
| 🟩 | 2012-05-24 07:05 | Infrastructures | ACS_NE_ALARM | mean(IMMS..PRNE13_INLAIR,10) | DMS |
| 🟩 | 2012-05-24 07:05 | Infrastructures | CE_EuroTherm_Cold | CE_loop2_error_Low | DMS |
| 🟩 | 2012-05-24 07:04 | Infrastructures | CE_EuroTherm_Warm | mean(IMAC..CE_Loop_1_Err,10) | DMS |
| 🟩 | 2012-05-24 07:02 | Infrastructures | MC2_EuroTherm_Cold | MC2_loop2_error_Low | DMS |
| 🟩 | 2012-05-24 07:01 | Infrastructures | MC_EuroTherm_Warm | mean(IMAC..MC2_Loop_1_Err,5) | DMS |
| 🟥 | 2012-05-24 07:00 | Environment | MC_Building | mean(Em..TEBDMC01,10) | DMS |
| 🟩 | 2012-05-24 07:00 | Infrastructures | MC_EuroTherm_Warm | mean(IMAC..MC1_Loop_1_Err,5) | DMS |
| 🟩 | 2012-05-24 06:54 | Infrastructures | ACS_CB_ALARM | mean(IMMS..PRCB14_INLAIR,10) | DMS |
| 🟩 | 2012-05-24 06:40 | Infrastructures | WE_EuroTherm_Cold | WE_loop2_error_Low | DMS |
| 🟩 | 2012-05-24 06:32 | Servers | Delta_Eurotherm | Delta_Eurotherm_1 | DMS |
| 🟥 | 2012-05-24 05:12 | Servers | Delta_Eurotherm | Delta_Eurotherm_1 | DMS |
| 🟩 | 2012-05-24 04:53 | Servers | Delta_Eurotherm | Delta_Eurotherm_2 | DMS |
| 🟩 | 2012-05-24 04:53 | Servers | Delta_Eurotherm | Delta_Eurotherm_1 | DMS |
| 🟥 | 2012-05-24 04:33 | Servers | Delta_Eurotherm | Delta_Eurotherm_1 | DMS |
| 🟥 | 2012-05-24 04:33 | Servers | Delta_Eurotherm | Delta_Eurotherm_2 | DMS |
| 🟩 | 2012-05-24 04:11 | Servers | Delta_Eurotherm | Delta_Eurotherm_2 | DMS |
| 🟥 | 2012-05-24 03:11 | Servers | Delta_Eurotherm | Delta_Eurotherm_2 | DMS |
| 🟩 | 2012-05-24 03:11 | Servers | Delta_Eurotherm | Delta_Eurotherm_1 | DMS |
| ▨ | 2012-05-24 03:10 | Vacuum | TowerServers | To_PR..Gc2 | DMS |
| ▨ | 2012-05-24 03:10 | Vacuum | BackPressureTo | To_PR..G41 | DMS |
| ▨ | 2012-05-24 03:10 | Vacuum | ScrollPumps | To_PR..P61_STATUS | DMS |

Last update: 9:18:04 UTC

Fig. 15: DMS Flag Log.

## 8.8 The *"DMS Flag List"*

This interface lists all the flags and for each flag we can have information about alarm configurations (if present), shelving status, and muting status.



Detector Monitoring System

Logged in as Gherardini Fabio Logout

DMS FLAG LIST

Expand All
Collapse All
Refresh
See all flags
DMS log
Alarm log
Mail Notification of shelve/mute changes
List of flags you can shelve/mute

Show/hide alarm types: ☑ ☎ ☑ ✉ ☑ 🔔

Note that if you disable the display of SMS alarm, but the flag is also alarmed by mail, the flag (and all its alarm types) will be showed.

Mute all

Select kind of alarm to mute: ☑ ☎ ☑ ✉ ☑ 🔔

From UTC to UTC

Short comment:

Not active

Help:

LEGEND OF ICONS USED FOR ALARMED:

☎ An SMS will be sent to the oncall

✉ An email will be sent to the oncall

🔔 A sound will be played on the DMS

CORRESPONDING MUTED ICONS:

Automation

- ⊞ALP_HOST
- ⊞Alp_Servers NOT ALARMED, NOT SHELVED
- ⊞TEST

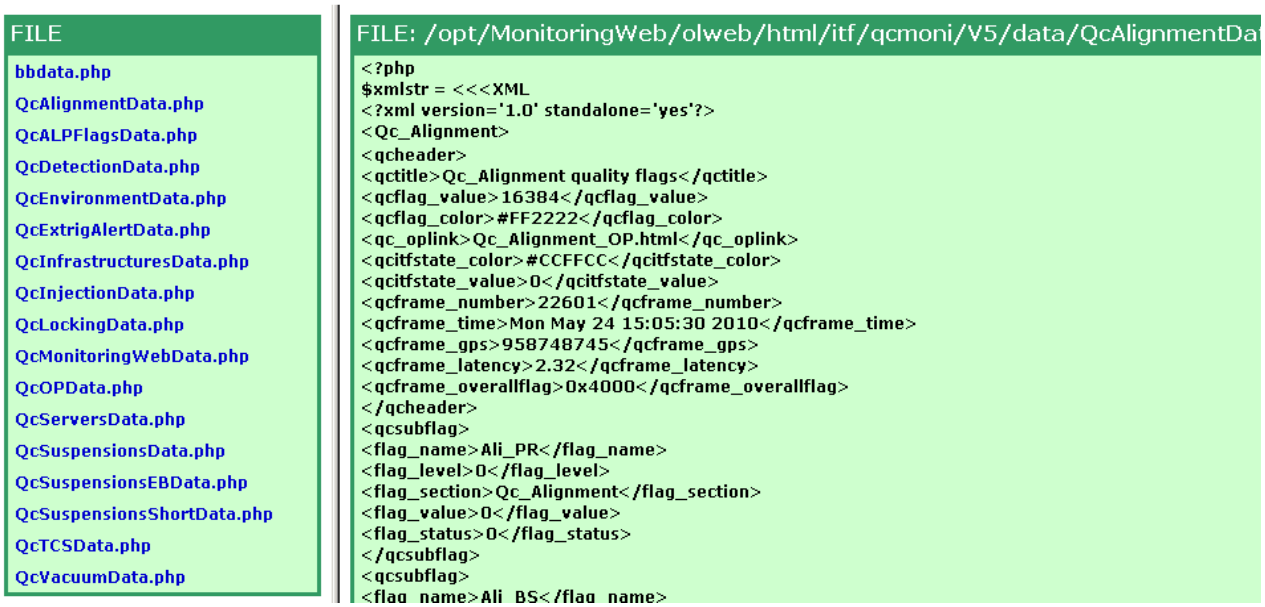
Vacuum

- ⊞LinkValves ALARMED ✉ ☎ 🔔 , NOT SHELVED
- ⊞TowerServers NOT ALARMED, PARTIALLY SHELVED
- ⊞TUBESERVERS
- ⊟TurboPumps ALARMED ✉ ☎ 🔔 , PARTIALLY SHELVED
  - To_BS..P51_POWER ALARMED ✉ ☎ 🔔 , NOT SHELVED — Shelve Mute
  - To_NI..P71_POWER ALARMED ✉ ☎ 🔔 , SHELVED FROM 10-APR-2012 13:55:03UTC TO 30-APR-2013 13:55:03UTC — Change settings or unshelve Mute
  - To_WI..P51_POWER ALARMED ✉ ☎ 🔔 , NOT SHELVED — Shelve Mute
  - To_PR..P51_POWER ALARMED ✉ ☎ 🔔 , NOT SHELVED — Shelve Mute
  - To_IB..P51_POWER ALARMED ✉ ☎ 🔔 , NOT SHELVED — Shelve Mute
  - To_OB..P51_POWER ALARMED ✉ ☎ 🔔 , NOT SHELVED — Shelve Mute
  - To_MC..P51_POWER ALARMED — Shelve Mute

*Fig. 17: DMS Flag List*

## 8.9 Interface to view XML files content

As described in the previous chapters the ‘input’ data are read from specific files having the information formatted following the XML structure; this interface allows choosing the input file and reading the content of that file.



- bbdata.php
- QcAlignmentData.php
- QcALPFlagsData.php
- QcDetectionData.php
- QcEnvironmentData.php
- QcExtrigAlertData.php
- QcInfrastructuresData.php
- QcInjectionData.php
- QcLockingData.php
- QcMonitoringWebData.php
- QcOPData.php
- QcServersData.php
- QcSuspensionsData.php
- QcSuspensionsEBData.php
- QcSuspensionsShortData.php
- QcTCSData.php
- QcVacuumData.php

FILE: /opt/MonitoringWeb/olweb/html/itf/qcmoni/V5/data/QcAlignmentDat

```
<?php
$xmlstr = <<<XML
<?xml version='1.0' standalone='yes'?>
<Qc_Alignment>
<qcheader>
<qctitle>Qc_Alignment quality flags</qctitle>
<qcflag_value>16384</qcflag_value>
<qcflag_color>#FF2222</qcflag_color>
<qc_oplink>Qc_Alignment_OP.html</qc_oplink>
<qcitfstate_color>#CCFFCC</qcitfstate_color>
<qcitfstate_value>0</qcitfstate_value>
<qcframe_number>22601</qcframe_number>
<qcframe_time>Mon May 24 15:05:30 2010</qcframe_time>
<qcframe_gps>958748745</qcframe_gps>
<qcframe_latency>2.32</qcframe_latency>
<qcframe_overallflag>0x4000</qcframe_overallflag>
</qcheader>
<qcsubflag>
<flag_name>Ali_PR</flag_name>
<flag_level>0</flag_level>
<flag_section>Qc_Alignment</flag_section>
<flag_value>0</flag_value>
<flag_status>0</flag_status>
</qcsubflag>
<qcsubflag>
<flag_name>Ali_BS</flag_name>
```

*Fig. 18: Content of the file provided by the process AliMoni; as we can see the information is formatted following the XML format.*

# 9 Future upgrades

The application is now fully tested and fully working indeed the last modification on the package is dated: June 2011. For this reason the upgrades are not yet completely planned.

A possible list of upgrades might be:

- improvement of the performances by using a better machine (faster, higher HARD DISK, etc…) to run the PHP scripts and to host the MySql db;
- replicate the db, in read-only mode on a public machine (i.e. pub3) to avoid the problem of the locked tables and to make the system usable in the CDB application;
- ask a backup machine to implement a mechanism of master/slave and make the system more reliable;
- interaction with FoxBox;

The functionalities of DMS plot has to fully debugged and maybe also on this aspect some improvements are necessary.

# 10 Program history

The software package of Detector Monitoring System is FlagMonitoringSystem; here are reported the major upgrades.

- version: **v1r0**, date: 2006 07 06 10:35:34

Warning system used in Control Room to check ITF situation

- version: **v2r0** date: 2006 07 06 10:54:57

Amendments made to sub-level.

- version: **v2r1** date: 2006 07 10 17:38:32

Amendments made to Alarm system

- version: **v2r2** date: 2006 07 10 18:10

Altered links to lower level because of flag manoeuvring and splitting of server.

- version: **v2r3** date: 2006 07 11 11:26:17

Old files backed-up.

Pause-Play functionality added to new version.

- version: **v2r4** date: 2006 07 11 18:34:02

Work on next version continues and amendments made to last group bug made.

- version: **v2r5** date: 2006 07 12 14:12:03

Lower level amendments completed.

Last group display resolved and links to recovery instructions included

- version: **v2r6** date: 2006 08 01 15:37:49

Archive minor modifications before switch to new version.

- version: **v3r0** date: 2006 09 13 10:34:05

Archived to allow for introduction of different group management system in next version.

- version: **v4r0** date: 2006 09 27 15:44:10

Groups managed in different ways, single lower level, and subflags declared via function

- version: **v5r0** date: 2006 10 09 12:31:57

New version online in the Control Room. Used since WSR3.

- version: **v5r1** date: 2006 10 09 15:40:20

Lght amendment made to refresh times to ensure immediate page refresh in event of problems.

- version: **v5r2** date: 2006 10 11 17:27

Modified to analyze bug from QcMoni server. Ultimately no problem with this code

- version: **v5r3** date: 2006 10 13 17:03:

Level two flags now bold.

Timeout of 2m kills second level automatically.

The run page can be opened from the page now.

The output buffer is cleaned every 5m.

- version: **v5r4** date: 2006 10 16 14:16:56

Stopped servers now report problems on index.php.

- version: **v5r5** date: 2006 10 16 15:08:38

IE refresh problem resolved. Pause no longer over-ridden in IE5.

- version: **v5r6** date: 2006 10 16 15:21:54

Header incorporated into level two as well.

- version: **v5r7** date: 2006 10 23 16:03:57

$alarmString removed from Detector Monitoring System logo to enable further testing of alarm system.

- version: **v5r8** date: 2006 12 07 17:56:49

Buffer flush reduced from 300s to 100s to attempt to resolve no refresh problem.

- version: **v5r9** date: 2007 03 23 14:52:41

Refresh moved to before styles.

- version: **v6r1** date: 2007 07 04 09:50:58

AND OR operators fully incorporated.

First version of DMS implemented.

- version: **v6r2** date: 2007 12 05 11:21

Altered email address for alerts from xmldms to DMS.

- version: **v6r3** date: 2008 02 04 15:42:

SPR #903: problem about wrong green flag fixed.

SPR #931: implemented the possibility to share flags with same name

- version: **v6r7** date: 2008 04 08 17:26

SPR #934: a MySql db is used to: store, and manage the alarm notification.

SPR #888: possibility to notify an alarm trough a sound.

- version: **v7r1** date: 2008 07 02 10:55:09

SPR #1067: configuration file structure changed to allow the possibility to specify all the functionality at each flag.

Optimization of the code using shared files.

Auto-restart mechanism improved.

- version: **v7r2** date: 2008 10 10 09:01:10

SPR #1092: notification in case of crash of the moni process;

SPR#1096: more details added in the text of the alarm notification;

alarm reset in case a flag turns from red to yellow in the persistence time;

flags alarmed pointed out with a small icon in DMS Level_2;

first interface to display alarmed flags

- version: **v7r3** date: 2008 12 10 13:51:44

SPR #1125: possibility to send SMS text with more 160 characters sending more messages (the original message is split); the error message specified in the cfg file of the various Moni processes added to the message; the additional message specified in the DMS cfg file added to the message;

SPR #1126: web interfaces fully developed;

function alarm.php rearranged in order to send sms both with Vodafone and with Servitel;

blinking test associated to SOUND alarm

date for the DMS pictures taken using the command "GPS"

- version: **v7r4** date: 2009 02 10 10:44:11

new Header section;

SPR #1140: alarm notification managed at the level of the subflags which constitute the overall flag.

- version: **v7r5** date: 2009 03 03 10:53:13

SPR#1143: TimeDelay functionality implemented;

new structure for all the WEB interfaces;

- version: **v7r6** date: 2009 03 11 11:28:43

SPR#1147: Depend functionality implemented;

- version **v7r7** date: 2009 04 14 08:59:13

SPR 1148: Shelve functionality implemented;

Possibility to create Aggregated flags;

Help implemented;

On the subflags added a link to the Java application to see the value of the channel on a plot.

- version **v7r8** date: 2009 05 11 08:50:35

SPR#1166: MoniProcess status in case of DAQ problems;

window to display only the desired flag;

- version: **v7r9** date: 2009 05 20 11:25:22

Priorities between functionalities implemented following the diagram block: AgF, Shelve, TimeDelay, Depend, Group.

- version: **v7r10** date: 2009 09 17 01:57:40

grouped flags shown with a border;

aggregated flags shown with a smaller border;

CSS applied also on the text of the header section;

Interface for XML files;

height of the rows for the subsystem specified in the configuration file through the keyword ROWS_HEIGHT;

functions for the output put inside a dedicated folder;

added two special flags: GRBAlert.Alert and SNAlert.Alert;

- version: **v7r12** date: 2010 03 31 09:59:53

archiving before new version

- version: **v8r0** date: 2010 04 26 14:47:50

NEW Version

- version: **v8r1** date: 2010 09 08 11:35

New version bugs fixed and further implementation; FOXBOX

- version: **v8r2** date: 2011 01 18 09:51:14

reason: AC login;

notification about shelving muting changes;

admin and notification interfaces implemented;

restyle of shelving muting interfaces;

- version: **v8r3** date: 2011 02 03 11:20:51

reason: Notification in case of SMS failure from FOXBOX

- version: **v8r5** date: 2011 04 08 10:37:43

EventLog ;

interface to add comment ;

plots improvement previous

- version: **v8r6** date: 2011 06 30 10:21:

EventLog->FlagLog: debug + improvement of configurable color persistence ;

script to copy locked tables to speed up the queries