Contents

1	Intr	odu	uction5	5
	1.1	Th	e DMS system	5
	1.2	Fla	ag states	5
	1.3	Dia	agram block	5
	1.4	Ch	annels data are written to downstream JSON-formatted files	5
	1.4	.1	DMS data providers connected to a shared memory	5
	1.4	.2	DMS data providers not connected to a shared memory	5
	1.5	Da	ata JSON files	7
	1.6	Th	e DMS server	7
	1.7	Ala	arm suppression	7
	1.7	.1	Shelving Flags	7
	1.8	Ala	arm filtering for notification	7
	1.8	.1	Persistence	7
	1.8	.2	Muting Flags	7
	1.9	Ala	arm filtering for displaying	7
	1.9	.1	Visualization delay	7
	1.9	.2	Grouping Flags	3
	1.10	1	Alert notifications	3
	1.11		Snapshot JSON file	3
	1.12	,	Visualization of Flags and associated information via a WUI	3
	1.13		The DMS publisher	9
	1.1	3.1	Archive snapshot	Э
	1.1	3.2	Archive trend	9
2	Mor	ni p	rocesses	9
	2.1	Mo	oni process configuration file	Э
	2.2	Mc	oni processes mathematical functions10)
3	Cor	npu	itingDMS)
	3.1	Со	omputingDMS configuration file 10)
	3.2	Av	ailable information	1
	3.3	Со	omputingDMS mathematical functions 11	1
4	Dat	a JS	SON files details	2
	4.1	Ра	yload12	2
5	The	D١	1Sserver in detail	3
	5.1	D١	1Sserver configuration file 14	1
6	Sna	psł	not JSON payload	7
	6.1	me	etatron_info18	3
	6.1	.1	state_info	3

	6.1.	2	metadata	18
	6.2	sub	system	19
	6.3	acti	ve_comments	19
	6.4	acti	ve_muting	19
	6.5	grou	up_flags	19
	6.6	dms	s_info	20
	6.7	acti	ve_shelving	21
	6.8	prov	viders	21
	6.9	acti	ve_alerts	22
	6.9.	1	flags	22
	6.9.	2	DMS	22
	6.9.	3	condition_flags	22
	6.9.	4	group_flags	23
	6.9.	5	providers	23
	6.10	С	onfiguration_info	23
	6.11	fla	agsź	24
	6.12	C	ond_flags_conflicts	25
	6.13	Ν	otes on the Snapshot JSON payload	25
7	The	DMS	Spublisher in detail	26
	7.1	Con	npression of Snapshot JSON files	26
	7.2	Rea	ding raw data and writing it to Trend 2	26
	7.3	Con	npression of Trend JSON files	26
8	The	Web	o User Interface	26
	8.1	The	homepage	26
	8.1.	1	Subsystem states	28
	8.1.	2	The individual-flag-information section	28
	8.1.	3	The Dashboard	30
	8.1.	4	The Alerts section	30
	8.1.	5	The Shelving section	31
	8.1.	6	The Muting section	32
	8.1.	7	Associated Condition Flag plots	32
	8.1.	8	Responsiveness	33
	8.2	The	DMS event monitor WUI	33
	8.3	The	DMS playback WUI	34
	8.4	The	DMS archive WUI	36
	8.5	The	DMS currently shelved condition-flags WUI	36
9	DMS	5 wo	rkflow	37
	9.1	Mor	ni process configuration	37
	9.2	Mor	ni process output: JSON payload	37

9.3	DMSserver configuration	38
9.4	DMSserver output: snapshot JSON payload	38
9.5	Homepage	39
9.6	DMSpublisher: reading raw data and writing it to Trend	39
9.7	DMSpublisher: compression JSON snapshot	39
9.8	DMS event monitor WUI	40
9.9	DMS playback WUI	40
9.10	DMS archive WUI	41
10 H	low-to	41
10.1	How-to for the Moni processes	41
10.	1.1 How to configure a generic channel in the Moni process	41
10.	1.2 How to configure a channel computed with the mean() mathematical function	42
10.	1.3 How to configure a channel computed with the rms() mathematical function	42
10.	1.4 How to configure a channel computed with the brms() mathematical function	42
10.	1.5 How to configure a channel computed with the delta() mathematical function	42
10.	1.6 How to configure a channel computed with the exist() mathematical function	42
10. TTE	1.7 How to configure a channel with thresholds depending by the value of	17
10 2	LOCK_STATE	42 42
10.2	2.1. How to configure a channel computed with the ping()function	42 42
10.	2.2 How to configure a channel computed with the cpu user()function	42
10.	2.2 How to configure a channel computed with the load()function	43
10.	2.4 How to configure a channel computed with the mem use ()function	4J 43
10.	2.5 How to configure a channel computed with the mem_use ()function	7J //2
10.3	How-to for DMSserver	43
10.5	3.1 How to stop and start the DMSserver	43
10.1	3.2 How to configure a provider.	43
10.1	3.3 How to configure an alert recipient	44
10.	3.4 How to test an alert recipient	44
10.3	3.5 How to configure a flag in the DMSserver	45
10.4	How-to for the homepage	46
10.4	4.1 How to open the DMS homepage	46
10.4	4.2 How to open the individual-flag-information section	46
10.4	4.3 How-to for Associated Condition Flag plots	46
10.4	4.4 How to log into-out the system	49
10.4	4.5 How to open the dashboard	49
10.4	4.6 How to open the alerts section	49
10.4	4.7 How to open the shelving section	49
10.4	4.8 How to open the muting section	49
10.4	4.9 How to shelve-unshleve a condition flag	50
		3

10.4.10	How to mute-unmute a condition flag 51
10.5 H	low-to for DMS event monitor
10.5.1	How to open the DMS event monitor
10.5.2	How to know the last events in a specified time window
10.5.3	How to know a specific events for specific condition flags a specified time window 52
10.5.4	How to know details of the event 53
10.6 H	low-to for DMS playback 53
10.6.1	How to open the DMS playback53
10.6.2	How to start-pause-stop the playback 53
10.6.3	How to download the snapshot JSON payload53
10.6.4	How to upload the snapshot JSON payload 54
10.6.5	How to switch the display 54
10.7 H	low-to for DMS archive
10.7.1	How to open the DMS archive
10.7.2	How to edit summary plots 54
10.7.3	How to edit custom plots 54
10.8 H	low to for the DMS currently shelved condition-flags
10.8.1	How to open the DMS currently shelved condition-flags
10.8.2	How to order the results 55
10.8.3	How to log into the system
10.8.4	How to view the list of the reports
10.8.5	How to build the report
10.8.6	How to un-shelve a flag 56
11 Docu	mentation

1 Introduction

1.1 The DMS system

Fundamentally, the DMS should constantly cycle through the following core steps:

- draw channel data from the Data Acquisition System (DAQ);
- elaborate this data and produce relevant Flags;
- provide alerts in various formats in pre-defined conditions;
- store the history of the Flags and associated metadata, e.g. alerts sent;
- allow the user to visualize these flags and their history via a web user interface (WUI);



The information is displayed as a multicell table where each cell is named **flag** and it denotes the status of a specific item; the state of the flags reveal information relating to either a given channel or a combination of channels and related computations and expressions.

1.2 Flag states

Flags must be associated to one of the following states:

Color	Color definition	State value	State definition
	Green	1	Standard working condition (OK);
	Yellow	2	Non-standard working condition (Warning);
	Grey	3	Data unavailable;
	Dark-grey	4	Corrupted data
	Red	5	Non-standard working condition (Error).

1.3 Diagram block

The easiest and most common way to reduce too sensitive red flags or false alarm notifications (which may lead to the point where the operators ignore it, and thus fails to bring attention to a hazardous situation) is an appropriate set/configuration of the alarm criteria; some other techniques have been implemented to reduce the alarm activation as displayed in the following block diagram.



1.4 Channels data are written to downstream JSON-formatted files.

In this step, data are written to downstream JSON-formatted files. These data arrive via two main types of providers: they are built from the DAQ data by Moni processes connected to a common shared memory; and from so-called other providers.

1.4.1 DMS data providers connected to a shared memory

1.4.1.1 Moni processes

The Moni processes are based on the Moni library package [VIR-074A-08]. This library allows to define a set of flags whose values are determined by subflags (called "**Condition flags**") which are built by comparing to thresholds the result of computation made on a given input data channel taken from the DAQ stream. All the computed flag values are sent back to DAQ stream but also used to build the information (previously XML files, now JSON formatted files) taken by the DMS. The processing of the input channels, the thresholds, the flags dependence, etc... are defined in one configuration file per Moni process.

For more details see also Moni processes, How-to for the Moni processes.

1.4.2 DMS data providers not connected to a shared memory

1.4.2.1 ComputingDMS

This tool is used to get the status of the equipment connected on the network and build a JSON file to be read by the DMS.

The flags generated by this server are the ones displayed on the second row of the section DAQ-Computing on the DMS page.

There are two kinds of information updated every 120sec:

- ping: performed by the server by a system command;
- cpu use, memory use, etc..: elaborated by the server making a query to Ganglia;
- For more details see also <u>ComputingDMS</u>, <u>How to for the ComputingDMS process</u>.

1.5 Data JSON files

Each process providing data to the DMS system must do so using a standardized format. The datapayload should be provided using JavaScript Object Notation (JSON). For more details see also <u>Data JSON files details</u>.

1.6 The DMS server

The DMS server is written in Python and its start/stop is incorporated into the Virgo Process Monitoring (VPM) application. For more details see also <u>The DMSserver in detail</u>, <u>How-to for DMSserver</u>.

1.7 Alarm suppression

1.7.1 Shelving Flags

It is possible to shelve Flags for determinate lengths of time. Shelving means to hide the Flag entirely from the rest of the application. The history of the shelving of a Flag is recorded in order to available to users via the WUI. It is easily understandable to users when a Flag is in a shelved state.

For more details see also The Shelving section, How to shelve-unshleve a condition flag.

1.8 Alarm filtering for notification

1.8.1 Persistence

The DMS is able to manage checks on the persistence of an alert. In the event of an alert exceeding a pre-defined persistence time period, only at that the alert notification is sent. For more details see also <u>How to configure a flag in the DMSserver with email notification</u>, <u>How to configure a flag in the DMSserver with sms notification</u>.

1.8.2 Muting Flags

It is possible to mute alerts for a pre-determined length of time. When an alert is muted, its Flag continues to appear in the DMS WUI and its state continues to be displayed, but any associated alert notification is not sent. The history of the muting of an alert is stored and available to the user via the WUI. It is easily understandable to users when an alert is in a muted state. For more details see also <u>The Muting section</u>, <u>How to mute-unmute a condition</u> flag.

1.9 Alarm filtering for displaying

1.9.1 Visualization delay

In order to avoid a constantly blinking WUI, DMS is able to delay the visualization of certain pre-defined Flags; the Condition Flags of which may fluctuate constantly around a given threshold. In the case of these Flags, from the moment in which one of its Condition Flags enters into a red state, the length of continuous time it passes in that state must be recorded and the Flag must only be set to red state in the event of a pre-determined time-length threshold being exceeded.

For more details see also <u>How to configure a flag in the DMSserver with</u> <u>"delay before visualization"</u>.

1.9.2 Grouping Flags

It is possible to group specific flags into a single overall group; the state of which takes on the state, i.e. the color, of the lowest common denominator flag in the group. For example, in a group called 'flag_group', made up of 'flag_1', 'flag_2' and 'flag_3', when 'flag_1' and 'flag_3' are green, but 'flag_2' is red, the overall 'flag_group' group must be red. For more details see also How to configure a flag in the DMSserver with "group".

1.10 Alert notifications

The DMS is able to alert pre-defined users in the event that a certain Flag is in a certain state for a pre-determined length of time. It is possible for the DMS to send alerts in the following formats:

- Sound emitted via the WUI (when a Sound alert is emitted, the WUI must also display the flag so that it blinks for a pre-defined amount of time);
- Email sent to pre-defined users;
- SMS sent to pre-defined users;

The DMS is be able to send alerts when a Condition Flag remains red for a pre-determined length of time.

For more details see also <u>The Alerts section</u>, <u>The DMS event monitor WUI</u>, <u>How to configure a</u> <u>flag in the DMSserver with email notification</u>, <u>How to configure a flag in the DMSserver with</u> <u>sms notification</u>.

1.11 Snapshot JSON file

The DMS server creates a Snapshot JSON file every 10 seconds.

The Snapshot JSON files are used by the WUI to display the status of the DMS at any given point in its past.

Two types of Snaphot JSON file are stored:

- one file called **online.json**, which stores the current state values, thresholds and comments. This file is over-written each time the thread handles the Snapshot JSON loop;
- files called {GPS_TIME}.json, which display the values, thresholds and comments at a specific GPS time.

For more details see also <u>Snapshot JSON payload</u>.

1.12 Visualization of Flags and associated information via a WUI

It is possible for users to access the Flag information via a WUI. This section details the various pages of the WUI that must be available.

The Web User Interfaces (WUI) are written in HTML5, CSS7 and JavaScript8. Interaction with the under-lying DMS system is be done via PHP and AJAX.

By default, on large and medium-sized screens, the WUI automatically refresh every 10 seconds; although it is possible to pause this refresh via the WUI.

The WUI, is responsive, i.e. it is automatically adapted to the dimensions of the screen being used – smartphone, tablet, desktop, laptop – and display the information in an easy to understand and accessible manner. This clearly has implications in terms of the ways in which the application displays information in the different sections, particularly in relation to the homepage and sub-system-details pages.

Below the list of the available WUI:

- the "Homepage";
- The Flag-shelving WUI;
- The Flag-muting WUI;
- The Dashboard;
- Associated Condition-flag plots;

- The "DMS archive";
- The "DMS event log"
- The "DMS playback".
- The "currently shelved condition-flags"

For more details see also <u>The Web User Interface</u>, <u>How-to for the homepage</u>, <u>How-to for DMS</u> <u>event monitor</u>, <u>How-to for DMS playback</u>, <u>How-to for DMS archive</u>, <u>How to for the DMS</u> <u>currently shelved condition-flags</u>.

1.13 The DMS publisher

The DMS publisher is written in Python and its start/stop can be easily incorporated into the Virgo Process Monitoring (VPM) application.

For more details see The DMSpublisher in detail.

1.13.1 Archive snapshot

In this phase the online snapshot is compressed into a zip file building the archive snapshots. This make possible to reconstruct the status of the DMS at a given moment when a request is made via the DMS playback WUI

For more details see also Compression of Snapshot JSON files, The DMS playback WUI.

1.13.2 Archive trend

In this phase the publisher reads the data in the Raw tables of the Online database every 30 minutes – this time is configurable – copies all data and then deletes any data that is more than 30 minutes old; again this time is configurable. It then converts them into JSON payloads, which are ultimately stored as Trend JSON files, which are, in turn, compressed into ZIP archives. The Trend JSON files are used when a request is made via the WUI to reconstruct the history of a given Conditional Flag.

For more details see also <u>Reading raw data and writing it to Trend</u>, <u>Compression of Trend JSON</u> <u>files</u>, <u>DMS archive</u>.

2 Moni processes

2.1 Moni process configuration file

Its parsing is done using the Cfg library developed in Virgo. The main keywords used by the Moni process are:

- QC_JSON : followed by the path/name of the JSON formatted file which contains the Moni process results and the refresh period (in seconds) of this file.
- QC_NAME: followed by the Moni_name that is used in the prefix of the name of the output flags.
- QC_FLAG: followed by the name of the flag and a comment string to be used when the flag is on.
- QC_MONITOR: followed by a "lock_status" string, the name of the flag it is associated to, the condition to be tested and a comment string to be used when the condition is not fulfilled. In addition, a second condition and a second comment string can be added to define a warning (yellow color in DMS) instead of an alarm (red color in DMS). Each configuration line beginning with the keyword defines a "Condition flag". All the Condition flags associated to a given flag are used to set the value of the flag.

For more details see also <u>How-to for the Moni processes</u>, <u>How to configure a generic channel</u> in the Moni process, <u>How to configure a channel with thresholds depending by the value of</u> <u>ITF_LOCK_STATE</u>.



Figure 3 – Configuration file of TCSMoni.

2.2 Moni processes mathematical functions

The signal filtering is performed using some predefined mathematical functions that can be stated in the configuration file. The functions are the following:

- mean(ChName,T): it computes the average value of the channel ChName over an interval of T seconds. For more details see also <u>How to configure a channel computed</u> with the mean() mathematical function.
- **rms(ChName,T)**: it computes the rms value of the channel ChName over an interval of T seconds. For more details see also <u>How to configure a channel computed with the rms() mathematical function</u>.
- **brms(ChName,T, Navg, f_min, f_max)**: it computes the rms value of the channel ChName over an interval of T seconds in the frequency band f_min f_max, averaging over Navg fft. For more details see also <u>How to configure a channel computed with the brms() mathematical function</u>.
- dynamic(ChName, T): computes the difference max-min where max and min are computed over the duration indicated as second parameter (to monitor thatthe signal is moving different from zero). For more details see also <u>How to configure a channel</u> <u>computed with the brms() mathematical function</u>.
- delta(ChName,T):computes the max of the absolute difference between two consecutive samples. The samples taken into account are those contained into a buffer having a duration of T seconds. For more details see also <u>How to configure a channel</u> <u>computed with the delta() mathematical function</u>.
- exist: green flag if the channel is present (even if containing only zero) and red flag if absent. For more details see also <u>How to configure a channel computed with the exist()</u> <u>mathematical function</u>.

3 ComputingDMS

3.1 ComputingDMS configuration file

Its parsing is done using the Cfg library developed in Virgo. The main keywords used by the ComputingDMS process are:

- JSON : followed by the path/name of the JSON formatted file which contains the Moni process results and the refresh period (in seconds) of this file.
- RR: refresh rate of the process
- CF: it is a line to be considered as valid input; then every field separated by a blank space is:
 - Flag name in the DMS;
 - DNS name or IP address;
 - ganglia group name;
 - info to monitor;
 - o arithmetical function over the values of the last hour;
 - low threshold for yellow flag (the flag is yellow if the value is higher);
 - low threshold for red flag (the flag is red if value is higher);

For more details see also How to for the ComputingDMS process.

CFG_PRIO 1										
CFG_DEBUG 0										
FD_NO_CONFIG_CHECK # allow custom keywords										
<pre># No commit into file CFG_NOFILESAVE</pre>										
CFG_CMDOMAIN Cascina										
<pre># Current logfile path <path>/<cm CFG_PWD /virgoLog/VirgoOnline</cm </path></pre>	Name>									
# Name and path for the output js	on file	NADIC Seen								
JSON /opt/Monitoringweb/butter_dm	is/json_qcmon1/Computi	ngums.json								
<pre># Refresh rate in [s] of the serv RR 120</pre>	er									
<pre># FLAG olserver38 CF ping(olserver38) CF cpu_user(olserver38)</pre>	olserver38 olserver38	olserver38.virgo.infn.it olserver38.virgo.infn.it	None New_Olservers	ping cpu_user	None max	0 70	0 90			
# FLAG olserver53										
CF ping(olserver53) CF cpu_user(olserver53)	olserver53 olserver53	olserver53.virgo.infn.it olserver53.virgo.infn.it	None On-LineNodes2	ping cpu_user	None max	0 70	0 90			
# FLAG olserver112	al convontita	olconyon112 yingo infn it	Nono	ning	Nono	0				
CF cpu_user(olserver112)	olserver112	olserver112.virgo.infn.it	New_Olservers	cpu_user	max	70	90			
# FLAG olserver113										
CF ping(olserver113) CF cpu_user(olserver113)	olserver113 olserver113	olserver113.virgo.infn.it olserver113.virgo.infn.it	None New_Olservers	ping cpu_user	max	0 70	90			
# # FLAG olserver117										
CF ping(olserver117)	olserver117	olserver117.virgo.infn.it	None New Olsenvers	ping	None	0	0			
#	01561.061.117	orserver 117. virgo. 1010.11	New_Orservers	cpu_user.	max	/0				
# FLAG olserver118	olserver118	olserver118.virgo.infn.it	None	ping	None	0	0			
CF ping(ofserverits)	olserver118	olserver118.virgo.infn.it	New_Olservers	cpu_user	max	70	90			
CF cpu_user(olserver118)										
CF cpu_user(olserver118) #										

3.2 Available information

The information about are the following:

- PING; for more details see also <u>How to configure a channel computed with the ping()function</u>.
- LOAD: load_1-min [number] , load_cpus [number], load_procs [number]; for more details see also <u>How to configure a channel computed with the load()function</u>
- MEMORY: mem_use [Bytes], mem_share [Bytes], mem_cache [Bytes], mem_buffer [Bytes], mem_free [Bytes], mem_swap [Bytes], mem_total [Bytes]; for more details see also <u>How to configure a channel computed with the mem_use ()function</u>, <u>How to configure a channel computed with the mem_swap()function</u>
- CPU: cpu_user, cpu_nice [%], cpu_system [%], cpu_wait [%], cpu_steal [%], cpu_sintr [%], cpu_idle [%]; for more details see also <u>How to configure a channel</u> <u>computed with the cpu_user()function</u>.
- NETWORK: net_in [Bytes], net_out [Bytes]; for more details see also

3.3 ComputingDMS mathematical functions

The signal filtering is performed using some predefined mathematical functions that can be stated in the configuration file. The functions are the following:

- none: no operations; mean: compute the mean in the last hour of samples;
- max: compute the max in the last hour of samples; min: compute the max in the last hour of samples;

4 Data JSON files details

Each process providing data to the DMS system must do so using a standardized format. The data payload should be provided using JavaScript Object Notation (JSON). These data JSON files are defined in this section.

4.1 Payload



Figure 5 – QcTCSData.json, JSON file generated by the Moni process TCSMoni.

The branches of the above payload are explained here

Name	Value	Details	Description
frame_latency	number	float, 2 decimal places	The frame latency value received from the DAQ.
frame_gps	number	integer	The frame GPS time received from theDAQ.
frame_number	number	integer	The frame number received from the DAQ.
frame_time	string	UTC	The UTC date/time received from

			theDAQ.
provider	string		The name of the provider - Moni script orother - that has written the data file.
flags	object	dictionary	Contains a dictionary of flag data.

Flag names are used as the root index of the flags dictionary. To each flag-name index is associated another dictionary containing the following leaves:

Name	Value	Details	Description
flag_state	number	integer	The DMS flag state
flag_dq_value	number	integer	The DMS flag value for use in data-quality processing
condition_flags	object	dictionary	Contains a dictionary of data related to a condition applied to a flag.

The index of the condition_flags dictionary is constituted of the related flag name, an underscore and an incrementing integer value, beginning at zero. For example:

{flag_name}_0

{flag_name}_1 {flag_name}_2

{IIay_IIame}_2

Each of these indices is associated to an associated dictionary, which contains the following elements:

Name	Value	Details	Description
channel_name	string	-	The name of the DAQ channel used in the condition.
condition_flag_thresholds	string	-	The thresholds against which the channel data has been applied in order to arrive at the condition- flag state.
condition_flag_comment	string	-	An underscore-delimited comment for use in describing why a condition-flag is in a non-standard state.
condition_flag_computed_value	number	float	The computed value that is compared to thresholds to set the condition-flag.
condition_flag_state	number	integer	The state of the condition-flag in DMS terms.
condition_flag_dq_value	number	integer	The value of the condition-flag for use in data-quality processing

5 The DMSserver in detail

Once started through the VPM, the DMS Server undertakes the following actions:

- reads the latest Data JSON files;
- checks whether the Provider is providing data;
- elaborates the data and writes it to the Online Database;
- removes old data from the Online table;
- handles flag state;
- handles alerts;
- and, when required, creates Snapshot JSON files.

The handling of flag states and alerts is not necessarily done in the order defined above, but is inter-changeable dependent upon the data provided.

Each of these actions are detailed in this section:

• Reading latest Data JSON files

At the start of the process, following the configuration phase, the DMS Server launches threads for

each Provider. Each of these, concurrently, reads the contents of each of the Data JSON files and

undertakes the following actions:

 uses Python native JSON library to directly read the contents of each file into a Python dictionary.

• Checking the provision of Provider data

Following the reading and initial actions, each DMS-Server thread checks whether the Provider is correctly providing data. To do this, the thread gets the GPS time provided in the Data JSON files and compares it with the actual GPS time. If the difference between the two times is greater than that specified by the

delay_before_unavailable_s variable, the Provider is considered to be inactive, the subsystem dedicated to the Provider is displayed in the Data Unavailable state and the alert-handling process is triggered. Otherwise, the Provider is considered to be active and the elaboration of the data continues.

• Elaborating and writing the online data

Once certain that the Provider is not unavailable and up-to-date data is being provided, each DMS-Server thread undertakes the following actions:

- checks whether each provider exists already in the database; where it does not exist, it inserts it, where it does, it updates the last-seen time;
- checks whether each flag exists already in the database; where it does not exist, it inserts it, where it does, it updates the last-seen time;
- checks whether each conditional flag exists already in the database; where it does not exist, it inserts it, where it does, it updates the last-seen time;
- o inserts the new data to the Online table and Raw table;

• Removing the non-online data

• The next step undertaken by each DMS-Server thread is to delete data associated to non-current GPS times from the Online table.

• Handling flag state

Each thread determines the state of each flag, taking into consideration any state_time_delayassociated to a specific flag in the tb_time_delay table; applying the following: 0 = delay has been reset; 1 = error state in progress; 2 = completed successfully

• Handling alerts

When handling alerts, each thread first undertakes the following actions:

- checks if the state of a provider or condition flag is in error;
- checks if an alert has been configured for this provider or condition flag;
- checks that the flag has not been shelved;
- checks that the flag has not been muted.

If all of these checks are not met, the alert-handling process for the provider or condition flag is abandoned. Instead, if they are all met, the following actions are undertaken:

- checks the persistence the alert_recipient_mail_persistance,
 - alert_recipient_sound and alert_recipient_sms_persistance values, defined within the configuration, for each individual provider or conditional flag;
 - $\circ~$ if the provider or conditional flag remains in error state beyond the persistence times, an associated alert is generated.

Creating Snapshot JSON files

The Snapshot JSON files are used by the WUI to display the status of the DMS at any given point in its past.

For more details see also <u>How-to for DMSserver</u>, <u>How to stop and start the DMSserver</u>.

5.1 DMSserver configuration file

The DMS Server is started via the VPM. It provides it with a configuration file, which is read by the VPM at the moment in which the Server is activated.

The header

Certain variables are required by the VPM in a header by default.

The body

The remainder of the configuration file is dedicated to the definition of the manner in which flags are displayed and alerts are disseminated.

The information that must be supplied in this part of the configuration file is divided into three main areas:

- **PROVIDER** the Moni or other processes supplying data to the DMS Server;
- ALERT_RECIPIENT address of recipient of a specific alert;
- **FLAG** the name of the flag.

The PROVIDER group includes the following fields:

Name	Туре	Description
file_path	string	The location of the providing file.
provider_name	string	Alias of the provider, e.g. SuspMoni.
delay_before_unavailable_s	integer	Period of time, in seconds, before the Provider is recognized as being Unavailable.
alert_recipient_mail	list	Names of the ALERT_RECIPIENT groups that should receive an email alert.
alert_recipient_mail_persistance	integer	The length of time, in seconds, required to have passed with the flag in Error state before email alerts are sent.
alert_recipient_sms	list	Names of the ALERT_RECIPIENT groups that should receive an SMS alert.
alert_recipient_sms_persistance	integer	The length of time, in seconds, required to have passed with the flag in Error state before SMS alerts are sent
alert_recipient_sound_persistance	integer	An integer defining the number of seconds that need to pass before a default sound is played and the length of time for which the flag blinks in the WUI

For more details see also <u>How to configure a provider</u>, <u>How to configure an alert recipient</u>, <u>How to test an alert recipient</u>.



Figure 7 – DMSserver configuration file, PROVIDER part.

The ALERT_RECIPIENT group includes the following fields:

alert_recipient_group	string	An alias given to the group to distinguish it from the others.
alert_recipient_addresses	list	List of addresses (email, telephone)associated to the group.

For more details see also How to configure an alert recipient, How to test an alert recipient.



The FLAG group includes the following fields:

Name	Туре	Description
flag_name	string	An alias given to the flag to distinguish it from the others.
subsystem	string	The name of the sub-system to which the flag is associated.
display_row	integer	The row on which the flag appears in the associated sub-system area of the WUI.
group_flag	string	Alias of the group flag to which the flag belongs, in the event of it forming a part of a wider group.
delay_before_error_s	integer	Length of time, in seconds, that must pass before the flag is considered to be in error state.
alert_recipient_mail	list	Names of the ALERT_RECIPIENT groups that should receive an email alert.
alert_recipient_mail_persistence	integer	The length of time, in seconds, required to have passed with the flag in Error state before email alerts are sent.
alert_recipient_sms	list	Names of the ALERT_RECIPIENT groups that should receive an SMS alert.
alert_recipient_sms_persistence	integer	The length of time, in seconds, required to have passed with the flag in Error state before SMS alerts are sent.
alert_recipient_sound_persistence	integer	An integer defining the number of seconds that need to pass before a default sound is played and the length of time for which the flag blinks in the WUI.

For more details see also <u>How to configure a flag in the DMSserver</u>, <u>How to configure an alert</u> recipient, <u>How to test an alert recipient</u>.

# 1 ? flag_name	(string										
2 ? subsystem	(upper-case string										
# 3 ? display_ro	w (integer										
<pre>4 ? group_flag</pre>	(string										
<pre>\$ 5 ? delay_befo</pre>	re_error_s (integer										
<pre># 6 ? flag_alert</pre>	_mail_recipient (con	mma-separated	list,	or 1	None						
7 ? flag_alert	_mail_persistance (:	integer									
8 ? flag_alert	_sms_recipient (com	na-separated	list, (or Ne	one						
9 ? flag_alert	_sms_persistance (1)	nteger									
f 10 ? flag_alert	_sound_persistence	(integer									
TNIECTION											
FLAC STR1 TD	Injection	1 None	2	0	None	0		None			
FLAG STR1 BENCH	Injection	1 None	3	0	None	0		None			
FLAG STR1 BR	Injection	1 None	3	0	None	0		None			
FLAC STR1 Ver+	Injection	1 None	2	0	None	0		None			2
FLAG STRI TE	Injection	1 None	3	0	None	0		None		1	2
FLAG SIB1 Guard	Injection	1 SIB1 Gua	rd	30	None	0	0	None		0	
FLAG STB1 Guard T	rigger Injection	1 STB1 Gua	rd	30	OPERATION MALL SUSPENSIONS MAL	T. 18	00	SUSPENSIONS	SMS 1	000	14
FLAG SIB1 Electr	Injection	1 None	31	0	None			None 0 0	II		
and orbi_sietti	injection	1 Hone	3	-		0					
FLAG MC TP	Injection	2 None	30	N	one	0	Ne	ne	0		0
FLAG MC PAY	Injection	2 None	30	N	one	0	No	ne	0		õ
FLAG MC BR	Injection	2 None	30	N	one	0	No	one	0		õ
FLAG MC Vert	Injection	2 None	30	N	one	0	N	ne	0		0
FLAG MC TE	Injection	2 None	30	N	one	0	No	one	0		õ
FLAG MC Guard	Injection	2 MC Guard	30	N	one	0	No	one	ō		0
FLAG MC Guard Tri	gger Injection	2 MC Guard	30	0	PERATION MAIL SUSPENSIONS MATL	1800	ST	SPENSIONS SMS	1800	18	00
FLAG MC Electr	Injection	2 None	30	N	one	0	No	one	0		0
1000											
FLAG MasterLaser	Injection	3 Laser	30	0	PERATION_MAIL, OPTICS_MAIL	300	No	one	0	3	60
FLAG SlaveLaser	Injection	3 Laser	30	0	PERATION_MAIL, OPTICS_MAIL	300	No	one	0	9	60
FLAG PMC	Injection	3 Laser	30	0	PERATION_MAIL, OPTICS_MAIL	600	No	one	0		60
FLAG LaserAmpliPo	wer Injection	3 LaserAmpli	30	0	PERATION_MAIL, OPTICS_MAIL	600	No	one	0		0
FLAG LaserChiller	Injection	3 None	200	0	PERATION_MAIL, OPTICS_MAIL	600	No	one	0		0
FLAG SL_TempContr	oller Injection	3 None	200	0	PERATION_MAIL, OPTICS_MAIL	300	No	one	0		0
FLAG RFC	Injection	3 None	120	N	one	0	No	one	0		0
FLAG LNFS	Injection	3 None	120	N	one	0	No	one	0		0
FLAG PC	Injection	3 None	120	N	one	0	No	one	0		0
	-										
FLAG MC_Power	Injection	4 None	120	N	one	0	No	one	0		0
FLAG PSTAB	Injection	4 None	120	N	one	0	No	one	0		0
LAG IMC AA	Injection	4 None	120	N	one	0	No	one	0		0
FLAG IMC AA GALVO	Injection	4 None	120	N	one	0	No	one	0		0
FLAG MC_FO_z	Injection	4 None	30	N	one	0	No	one	0		0
LAG BPC	Injection	4 None	120	N	one	0	No	one	0		0
FLAG BPC_Electr	Injection	4 None	30	N	one	0	No	one	0		0
DETECTION		4			N						
FLAG PD	Detection	1 None		30	None		0	None		0	
ETVC ODD B3	Detection	1 None		30	None		0	None		0	
LAG QPD B2	Detection	1 None		30	None		0	None		0	

6 Snapshot JSON payload

The Snapshot JSON files are used by the WUI to display the status of the DMS at any given point in its past.

	₽{	
	Ė.	"metatron_info":,
14	ė.	"subsystems":,
31	é	"active_comments": {},
32	Ē	"active_muting":,
100	Ē	"group_flags":,
164	Ē.	"dms_info":,
179	Ē.	"active_shelving":,
220	Ē	"providers":,
291	Ē.	"active_alerts":,
315	ė.	"configuration_info":,
320	Ė.	"flags":,
8790	ė.	"cond_flags_conflicts":
.8834	}	

Figure 10 – OnlineSnapshot.json, JSON file generated by DMSserver.

The branches of the above payload are explained here

Name	Value	Details	Description
metatron_info	object	dictionary	Contains a dictionary of Metatron data
subsystem	object	dictionary	Contains a dictionary of subsystem data
active_comments	object	dictionary	Contains a dictionary of active comments data
active_muting	object	dictionary	Contains a dictionary of active muting data
group_flags	object	dictionary	Contains a dictionary of group flags data
dms_info	object	dictionary	Contains a dictionary of dms info data

active_shelving	object	dictionary	Contains a dictionary of active shelving data
providers	object	dictionary	Contains a dictionary of providers data
active_alerts	object	dictionary	Contains a dictionary of active alerts data
configuration_info	object	dictionary	Contains a dictionary of configuration info data
flags	object	dictionary	Contains a dictionary of flags data
cond_flags_conflicts	object	dictionary	Contains a dictionary of condition flags conflicts data

6.1 metatron_info

2 🛱	<pre>"metatron_info": {</pre>
3 🛱	"state_info": {
4	"state": "DOWN",
5	"mode_length_s": 5176,
6	"state_length_s": 373,
7	"mode": "Maintenance"
8	},
9 🛱	"metadata": {
10	"last_mode_change_time": 1237621194,
11	"last_state_change_time": 1237625997
12	}
13	},

Figure 11 – OnlineSnapshot.json, metatron_info branch.

metatron_info is used as the root index of a dictionary having the following keywords:

Name Valu	e Details	Description
state_info obje	ct dictionar	y Contains a dictionary of Metatron info data
metadata obje	ct dictionar	y Contains a dictionary of metadata

6.1.1 state_info

state_into is a dictionary containing the following elements:

Name	Value	Details	Description
state	string		Current ITF state
mode_lenght_s	number	integer	Segment of the current ITF mode [S]
state_lenght_s	number	integer	Segment of the current ITF state [S]
mode	string		Current ITF mode

6.1.2 metadata

metadata is a dictionary containing the following elements:

Name	Value	Details	Description
last_mode_change_time	Number	Integer	UNIXTIMESTAMP of the last ITF mode change
last_state_change_time	Number	Integer	UNIXTIMESTAMP of the last ITF state change

6.2 subsystem



Figure 12 – OnlineSnapshot.json, subsystem branch.

The index of the subsystem dictionary is constituted of the related subsystem name and the codified value of the subsystem. For example:

"Calib_Hrec": 5,

"Environment": 3,

6.3 active_comments

active_comments is used as the root index of a dictionary that at moment is always empty because the functionality to add comment is not implemented.

6.4 active_muting



Figure 13 – OnlineSnapshot.json, active_muting branch.

active_muting is used as the root index of a dictionary; the indexes of this dictionary are the flag-name of the flag currently muted. For example:

mean_HVAC__CEB_HOT_CORR_60

mean_HVAC__EER_COLD_TE_ERR_60

To those flag-name index is associated another dictionary containing the following leaves:

Name	Value	Details	Description
id_muting	number	integer	Index of the related entry in the database

6.5 group_flags

100 🛱	"group_flags": {
101	"PR_Guard": 1,
102	"TcsAl": 1,
103	"ACS_WE": 1,
104	"Storage": 1,
105	"ACS_WAB": 1,
106	"DET_Area": 5,

Figure 14 – OnlineSnapshot.json, group_flags branch.

The index of the group_flags dictionary is constituted of the related group name and the codified value of the group. For example: "PR_Guard": 1,

"TcsAl": 1, "ACS_WE": 1,

6.6 dms_info

164	dms_info": {
165	<pre>"cond_flags_conflicts": 6,</pre>
166	"tot_providers": "29",
167	"active_providers": "23",
168	"active_flags": "350",
169	"tot_flags": "549",
170	"file_write_unixtimestamp": 1553591145,
171	"active_channels": 3112,
172	"active_cond_flags": "2739",
173	"tot_channels:": 0,
174	"tot_cond_flags": "5826",
175	"file_write_gps": 1237626363,
176	"tot_channels": 6404,
177	"file_write_utc": "2019-03-26 09:05:45"
178	},



dms_info is a dictionary containing the following elements:

Name	Value	Details	Description
cond_flags_conflicts	number	integer	Number of the current condition flags conflicts
tot_providers	number	integer	Number of the providers inserted in the database
active_providers	number	integer	Number of the providers read by the DMSserver
active_flags	number	integer	Number of the flags read by the DMSserver
tot_flags	number	integer	Number of the flags inserted in the database
file_write_unixtimestamp	number	integer	UNIXTIMESTAMP of the writing of the snapshot
active_channels	number	integer	Number of the channels (providers + flags + condition flags) read by the DMSserver
active_cond_flags	number	integer	Number of the condition flags read by the DMSserver
tot_channels	number	integer	Number of the channels (providers + flags + condition flags) inserted in the database
tot_cond_flags	number	integer	Number of the condition flags inserted in the database
file_write_gps	number	integer	GPS of the writing of the snapshot
file_write_utc	string		UTC of the writing of the snapshot

6.7 active_shelving



Figure 16 – OnlineSnapshot.json, active_shelving branch.

active_shelving is used as the root index of a dictionary; the indexes of this dictionary are the flag-name of the flag currently shelved. For example:

mean_HVAC__MCB_2_COLD_CORR_60

mean_HVAC__MCB_1_COLD_CORR_60

To those flag-name index is associated another dictionary containing the following leaves:

Name	Value	Details	Description
id_shelving	number	integer	Index of the related entry in the database

6.8 providers

220	Ē	"providers": {
221	ģ	"InjMoni": {
222		"state": 1
223		},
224	Ė.	"InfraMoni": {
225		"state": 1
226		},
227	ė	"SuspMoni": {
228		"state": 1
229		},

Figure 17 – OnlineSnapshot.json, providers branch.

providers is used as the root index of a dictionary; the indexes of this dictionary are the provider-name. For example:

InjMoni,

InfraMoni

To those provider-name index is associated another dictionary containing the following leaves:

Name	Value	Details	Description
state	number	integer	State (color) of the DMS subsystem

6.9 active_alerts



Figure 18 – OnlineSnapshot.json, active_alerts branch.

Active_alert is a dictionary containing the following elements:

Name	Value	Details	Description
flags	object	list	Contains a list of flags-names that has at least one condition flag in alarm
DMS	object	dictionary	Contains a dictionary of DMS alert data
condition_flags	object	dictionary	Contains a dictionary of condition flags alert data
group_flags	object	list	Contains a list of the current group-names that has at least one condition flag in alarm
providers	object	dictionary	Contains a dictionary of provider alert data
601 flags			

6.9.1 flags

the elements of the list are the flag-names that has at least one condition flag in alarm.

6.9.2 DMS

DMS is a dictionary containing the following elements:

Name	Value	Details	Description
channels	string		Number of the channel above the allowed threshold
condition_flags	string		Number of the condition flags above the allowed threshold
condition_flags_conflicts	string		Number of the condition flags conflicts above the allowed threshold
flags	string		Number of the flags above the allowed threshold
providers	string		Number of the providers above the allowed threshold

6.9.3 condition_flags

condition_flags is used as the root index of a dictionary; the indexes of this dictionary are the conditions flag-name who have an active alert. For example:

mean_ENV_CEB__DUST_0P3UM_30

mean_INF__WEB_HEATER_TE_IN_60

To those conditions flag-name index is associated another dictionary containing the following leaves:

Name	Value	Details	Description
alert_sms_state	number	integer	Current state of sms notification
alert_email_state	number	integer	Current state of email notification
alert_sound_state	number	integer	Current state of sound notification
alert_notification_id	number	integer	Index of the related entry in the database of the current notification

6.9.4 group_flags

the elements of the list are the group-names that has at least one condition flag in alarm.

6.9.5 providers

providers is used as the root index of a dictionary; the indexes of this dictionary are the providers-name who have an active alert. For example:

CaliMoni

EnvMoni

To those providers-name index is associated another dictionary containing the following leaves:

Name	Value	Details	Description
alert_sms_state	number	integer	Current state of sms notification
alert_email_state	number	integer	Current state of email notification
alert_sound_state	number	integer	Current state of sound notification
alert_notification_id	number	integer	Index of the related entry in the database of the current notification

6.10 configuration_info

315 🛱	"configuration_info": {
316	"id_configuration_file": 723,
317	"archive_path_json_configuration_file": "/data/archive_dms/json_configuration_files/
318	"archive_path_configuration_file": "/data/archive_dms/configuration_files/723.json"
319	},

Figure 19 – OnlineSnapshot.json, configuration_info branch.

Configuration_info is a dictionary containing the following elements:

Name	Value	Details	Description
id_configuration_file	number	integer	
archive_path_json_configuration_file	string		Path of the JSON configuration file
archive_path_configuration_file	string		Path of the configuration file

6.11 flags

	_	
320	닏	"†lags": {
321	ė.	"BACnet_devices": {
322		"state_delay_before_error": 0,
323		"flag_state": 1,
324	ė.	"cond_flags": {
325	ģ	"ping_det_device": {
326		"condition_flag_computed_value": 1.0,
327		"condition_flag_comment": "",
328		"condition_thresholds": "ping(det_device)",
329		"condition_flag_state": 1
330		},
331	ė	"ping_inj_device": {
332		"condition_flag_computed_value": 1.0,
333		"condition_flag_comment": "",
334		"condition_thresholds": "ping(inj_device)",
335		"condition_flag_state": 1
336		}
337		}
338		},
339	þ	"SWEB_SBE": {

Figure 20 – OnlineSnapshot.json, flags branch.

Flag names are used as the root index of the flags dictionary. To each flag-name index is associated another dictionary containing the following leaves:

		J	
Name	Value	Details	Description
state_delay_before_error	number	integer	
flag_state	number	integer	
cond_flags	object	dictionary	Contains a dictionary of data related to a condition applied to a flag.

The index of the condition_flags dictionary is constituted of the related condition flag name. For example:

mean_SPRB_DBOX_SBE_ps_temp_60

mean_SPRB_DBOX_SBE_slot3_temp_60

Each of these indices is associated to an associated dictionary, which contains the following elements:

Name	Value	Details	Description
condition_flag_computed_value	number	integer	
condition_flag_comment	string		
condition_thresholds	string		
condition_flag_state	number	integer	

6.12 cond_flags_conflicts

18790	ė	<pre>"cond_flags_conflicts": {</pre>
18791	ė.	<pre>"mean_TFMoni_WEPCAL_PD1_PD2_60_Phase_10": {</pre>
18792	ė	"10162": {
18793		"unix_cr": 1538156425,
18794		"unix_up": 1547398868,
18795		"flag_name": "NCAL",
18796		"provider": "CaliMoni"
18797		},
18798	ģ	"10166": {
18799		"unix_cr": 1538156425,
18800		"unix_up": 1553561714,
18801		"flag_name": "PCalWE",
18802		"provider": "CaliMoni"
18803		}
18804		},

Figure 21 – OnlineSnapshot.json, cond_flags_conflict branch.

cond_flags_conflicts is used as the root index of a dictionary; the indexes of this dictionary are the conditions flag-name that have name conflicts. For example:

mean_TFMoni_WEPCAL_PD1_PD2_60_Phase_10

mean_TFMoni_WEPCAL_PD1_PD2_60_Phase_10

The name conflict arises when the same channel is configured with the same criteria in two different places; to those conditions flag-name conflict are contains other dictionary having the root index of the condition flag-id of the database. For example:

10162 10166

Each of these indices is associated to a dictionary, which contains the following elements:

Name	Value	Details	Description
unix_cr	number	integer	UNIXTIMESTAMP of the first insertion in the database
unix_up	number	integer	UNIXTIMESTAMP of the last update in the database
flag_name	string		Name of the flag
provider	strina		

6.13 Notes on the Snapshot JSON payload

While much of the Snapshot JSON payload is easily understandable, a few points should be taken into consideration when using it:

• the Snapshot JSON files do not contain the GPS start and stop times for alerts, muting or shelving. Instead they contain the values inserted to the database in relation whether they are

active or not;

- the alert_email_state, alert_sound_state and alert_sms_state default to a -1 value meaning that no alert of that type has been triggered. This value exists only in the Snapshot JSON files and not in the Online Database. This is because, where no alert is present, related alert information is obviously not stored in the database. However, the Snapshot JSON file, in order to provide a coherent and reproducible payload, must provide each of the alert branches regardless of whether an alert of that type is associated to it or not;
- the active alerts, shelving, muting and comments dictionaries are populated only by those conditional flags that are actually considered to be in any of those states at that moment.

7 The DMSpublisher in detail

The Publisher undertakes the following specific actions:

- compression of Snapshot JSON files;
- reading of raw data from the Online Database and writing it to trend;
- removing raw data from the Online Database, once it has been written to trend;
- compression of Trend JSON files.
- This section describes each action in detail.

The Publisher runs every 10 seconds compresses the Snapshot JSON files. Every 30 minutes, it also converts the raw data to trend and compresses the results.

These parameters are configurable within the Publisher code. Furthermore, the Publisher is designed so that it can be paused as and when required, yet, on restart, recover the complete archive process, i.e. if paused, it simply begins to archive all those Snapshot files that have been produced during the period of time in which it has been offline.

The Publisher is written in Python.

7.1 Compression of Snapshot JSON files

When compressing the Snapshot JSON files, the Publisher undertakes the following actions:

- creates a JSON file that serves as an index in order to make finding the correct information again later, as easy as possible;
- checks if an open archive file in ZIP format is available, if not, it creates one. If an open file is available, it uses it. The new archive file uses the naming convention: {GPS_TIME_OF_FIRST_FILE_IN_ARCHIVE}.zip;
- gets any available Snapshot JSON files and puts them into the currently-open archive;
- updates the index file with the new information;
- deletes those original Snasphot JSON files that have been archived.

7.2 Reading raw data and writing it to Trend

The publisher reads the data in the Raw tables of the Online database every 30 minutes - this time is configurable - copies all data and then deletes any data that is more than 30 minutes old; again this time is configurable. It then converts them into JSON payloads, which are ultimately stored as Trend JSON files, which are, in turn, compressed into ZIP archives. The Trend JSON files are used when a request is made via the WUI to reconstruct the history of a given Conditional Flag.

7.3 Compression of Trend JSON files

Periodically, the Publisher takes any data older than 30 minutes within the buffer files and transfers it directly to files that exist within the DMS Server memory and transfers them directly to a dedicated ZIP archive within the dedicated archive sub-directory. This leads to the production of archives

8 The Web User Interface

8.1 The homepage

The main-screen area display the sub-system and flag information. Each sub-system and flag have a dedicated container, the background color of which depends upon the associated state. he homepage of the WUI should provide the following functionality:

• Flags, grouped into Subsystems, displaying the name of the flag, within a container taking the color of the Flag state;

 each Subsystem must have a container that takes the color of the state of the lowestcommon denominator Flag within the Subsystem; exact details are described in the subsequent sections

of this paper;

- clicking on a Flag should provide details on the state of the Subsystem, with the focus being on the called Flag;
- clicking on a Subsystem should provide details on the state of the Subsystem;
- general information, including:
 - o current UTC time, in the following format, e.g.: 'Fri Nov 18 11:09:58 2016';
 - current GPS time, in seconds, e.g.: 1163502615;
- information on the status of the interferometer:
 - ITF mode, e.g.: COMMISSIONING;
 - ITF State, e.g.: LOW_NOISE_3

The WUI is divided into the following main containers:

- a top bar which sits in a fixed position at the top of the screen;
- a main-screen area the whole main screen area;
- a group of buttons this is available to the right-hand-side of the screen. These buttons include links to display the following:
 - a dashboard;
 - the all-flag list;
 - a log of all alerts;
 - the shelving interface;
 - the muting interface;
- a fixed-width left-sided container, which slides-in from the left-hand side of the screen when called and is used to display the following information:
 - a history of all alerts;
 - the shelving interface;
 - \circ the muting interface.
- a dashboard a responsive left-sided container, which fits to 95% of the width of the screen and displays a general DMS overview
- an individual-flag-details container, which fits to 90% of the width of the screen and is used to display individual flag-details, when called;
- a covering modal a modal container that slides-in from the top of the screen and is used principally to provide users with generic information, such as warnings

For more details see also <u>How-to for the homepage</u> and <u>How to open the DMS homepage</u>.



Figure 22 – DMS homepage

8.1.1 Subsystem states

The exact color a Subsystem container takes depends upon the lowest-common denominator Flag within that Subsystem and follows the logic described here:

- Subsystem Green All Flags Green;
- Subsystem Yellow At least one Flag Yellow and all other Flags Green;
- Subsystem Grey At least one Flag Grey and all other Flags are not Red and are not Dark Grey;
- Subsystem Dark Grey At least one Flag Dark Grey and no Flag is Red;
- Subsystem Red At least one Flag Red

8.1.2 The individual-flag-information section

This section displays in the dedicated container and is automatically refreshed every 10s. The following information is displayed in relation to the specific called flag:

- the called flag;
- if the flag is a group flag, the flags that constitute the group flag;
- the condition flags from which the flag (and any group-composition flags) is composed.
- the history state, value, alert-notification, shelving and muting of the specific flag;
- the possibility to shelve and/or mute the flag;
- a configurable plot showing, by default the last 30 minutes, of the life of the flag;
- the possibility to show plots of the associated condition flag(s).

In addition, it is possible in this page to select up to three more flags, against which the first flag can be compared

For more details see also How to open the individual-flag-information section



Figure 23 – DMS individual-flag-information section

8.1.2.1 Detail of the individual-flag-information section



Figure 24 – detail of the individual-flag-information section

- Flag: it is name shown in the homepage, it could be also a group.
- Sub-sys: it is subsystem to which the flag belongs.
- Alerts: if present it indicates that the flag has been configured to send alert. Three kind
 of icons denotes the type of notification:

Mathematical Sector Sector

□→ sms notification

 \rightarrow sound notification

By passing the mouse over the icon you can get info about the alert configuration:



• Delay: it indicates the status of the visualization delay

This is a group

It shows that this is a group.

• Sub- and condition flags: starting from here there is the list of flags and related condition flags;

The info for each condition flag are the following

- ID: is the ID associated to each condition flag;
- Condition flag criteria and thresholds;
- Condition flag current value;
- Condition flag comment associated to yellow or red flag;
- *1* link to the shelving interface;
- Iink to the muting interface;
- Mullink to the associated plot;

- **I**link to the DMS event monitor;
- Blink to the DMS archive;
- a plot of the last 30 minutes that can be open/closed by clicking on the icon ${igsar}$

8.1.3 The Dashboard

The dashboard provides information describing the current state of the DMS. It shows information including:

- User information;
- Sound alert acctivation;
- 100 most recent events: alert-notification, recently-shelved flags, recently-muted flags;
- links to open the "DMS event monitor WUI" and the "DMS currently shelved flags WUI"
- a list of all flags currently in conflict;
- For more details see also <u>How to open the dashboard</u>.

🚯 Dashboard						×)8:41::	
User information:	You can sign-in here					ctr		
Sound alerts:	ා 🗘 Activate ම 🖉 Deactivate	🗘 Activate 🖲 💢 Deactivate						
						- 2	ectr	
100 most-recent events (Last UTC update: 2019-03	-27 08:33:53) % Go to Event Monitor	Go to shelved-flag che	cklist					
Local Date/Time	Provider		Conditi	on-Flag	Details		ectr	
	AlertGraceDB		GraceDf	3_alert	Alert 🔊	î	ectr	
2019-03-27 09:28:41	LockMoni		META_ITF_LC	DCKindex	Alert 🔊		ectr	
2019-03-27 09:08:14	LockMoni		META_ITF_LC	DCKindex	Alert 🔊		lectr	
2019-03-27 09:04:16	VPM		lvalert_	_virgo	Alert 🖂		lectr lectr	
2019-03-27 08:57:38	LockMoni		META_ITF_LC	DCKindex	Alert 🔊		Mon	
						~	AB	
Condition-flag conflicts							_WAB COB	
Condition flag		Provider(s)	Flags	Created (UTC)	Last updated (UTC)		B_LC	
ToVac		VPM	DataAccess	2017-05-03 06:48:08	2019-03-25 15:30:27		TE	
		VPM	Vacuum	2019-03-25 15:30:43	2019-03-27 08:33:41		RVOS	
mean_TFMoni_WEPCAL_PD1_PD2_60_Phase_1		CaliMoni	NCAL	2018-09-27 08:48:45	2019-01-14 14:22:40		um	
		CaliMoni	PCalWE	2018-09-27 09:30:02	2019-03-27 08:33:49			
mean_TFMoni_WEPCAL_PD1_PD2_60_Phase_10		CaliMoni	NCAL	2018-09-28 17:40:25	2019-01-13 17:01:08		atures	
		CaliMoni	PCalWE	2018-09-28 17:40:25	2019-03-27 08:28:28		Iniecti	

Figure 25 – DMS dashboard

8.1.4 The Alerts section

In this section, the 500 most recent alerts are displayed. For each alert, the following information is displayed:

- the UTC time at which the notification was sent;
- the name of the Provider, Flag, Condition-Flag for which the notification has been sent;
- the message string sent with the notification;
- the recipients that received the notification;

For more details see also How to open the alerts section

● Alerts ×	^		ITF Mo	de: Commission	ning (0d 2h 41m 1s)	ITF State: LOC	KING_OMC1_B	L s2_DC (0d 0h 0m 47		019-03-27 08:42:18
Alerts	SIB1_IP	SIB1_B	INCH	SIB1_BR	SIB1_Ve	ert	SIB1_TE	SIB1_Guar	d d	SIB1_Electr 🚽
Currently displaying the 500 most-	MC_IP	MC_P	AY	MC_BR	MC_Ve	rt	MC_TE	MC_Guard		MC_Electr
recent alerts:	Laser	LaserA	mpli L	aserChiller	SL_TempCor	ntroller	RFC	LNFS		PC 🗖
	MC_Power	PSTA	B	IMC_AA	IMC_AA_G	ALVO	MC_F0_z	BPC		BPC_Electr
Date/Time: 2019-03-27 09:34:21	PD	QPD_E	31p	QPD_B2	QPD_B	5	OMC	PicoDisabl	e	Shutter
Provider: LockMoni	SDB1_IP	SDB1_		SDB1_BR	SDB1_V	ert	SDB1_TE	SDB1_Gua	rd	SDB1_Electr
C. flag: META_ITF_LOCKindex	B2 8MHz DPHI	B4_56MH:	z_DPHI I	DARM_UGF	UNLOC	к	SSFS_UGF	FmodErr		GIPC
Date/Time: 2019-03-27 09:30:06	B1p_DC	B4_	112MHz_MAG	B7_[DC	B8_DC		LSC_rms		ASC_rms
Alert type: እ	BS_IP	BS_F7	BS_PA	Y E	S_BR	BS_Vert	BS_TE	BS_C	Guard	BS_Electr
Provider: AlertGraceDB C. flag: GraceDB_alert	NI_IP	NI_F7	NI_PA	r I	NI_BR	NI_Vert	NI_TE	NI_C	Guard	NI_Electr
	NE_IP	NE_F7	NE_PA	Y N	IE_BR	NE_Vert	NE_TE	NE_C	Guard	NE_Electr
Date/Time: 2019-03-27 09:28:41	PR_IP	PR_F7	PR_PA	Y F	PR_BR	PR_Vert	PR_TE	PR_C	Guard	PR_Electr
Provider: LockMoni	SR_IP	SR_F7	SR_PA	Y S	SR_BR	SR_Vert	SR_TE	SR_0	Guard	SR_Electr
C. flag: META_ITF_LOCKindex	WI_IP	WI_F7	WI_PA	Y V	VI_BR	WI_Vert	WI_TE	WI_Guard		WI_Electr
Date/Time: 2019-03-27 09:08:14	WE_IP	WE_F7	WE_PA	Y V	WE_BR		WE_TE	WE_Guard		WE_Electr
Alert type: እ	CB_Hall	MC_Hall	TCS_zor	ies N	E_Hall	WE_Hall	WindActivi	ty Seis	mon	BRMSMon
C. flag: META_ITF_LOCKindex	INJ_Area	DET_Area	EE_Room	DAQ_Room	Externa	al DeadC	hannel L	ights S	eaActivity	WAB
	CS_CB_Hall	ACS_TB	ACS_DAQ_Room	ACS_EE_Roor	n ACS_M	C ACS	_INJ AC	S_DET	ACS_NE	ACS_WAB
Date/Time: 2019-03-27 09:04:16	UPS_TB	UPS_MC	UPS_NE	UPS_WE	FlatChan	nel ExistCl	hannel AC	S_WE ACS_CB_CR		ACS_COB
Provider: VPM	EIB_SBE	SDB2_SBE	SDB2_LC	SNEB_SBE	SNEB_L	SNEB_LC SWEB_		/EB_LC S	PRB_SBE	SPRB_LC
C. flag: lvalert_virgo	N	II_CO2_Laser		WI_CO2_Laser				Chillers		
Message sent: flag: DetChar condition flag: lvalert_virgo	PLL	Squee	zer	SQZ_AA	SQZ_Shu	tter	Cohe_CTRL	SQZ_Inj		Rack_TE
thresholds: lvalert_virgo value:0 comment:	LargeValves	Clean_Air	TubeStati	ons Tub	ePumps	MiniTowers	TurboLink	s RemD	ryPMP	VAC_SERVOS
N.C.	Pressure	Compres	sedAir To	owerServers	TowerPur	nps	CryoTrap	O2_Senso	rs	Tank
Recipients:	ctorSEnvironment	ControlRoom	Minitowe	ers	ISC	Injection	TCS	Susp	ension	Vacuum
berenarginate	DetectorMonitoring	g Da	ataCollection	Stora	ige	DataAccess	S	Automation		DetChar
Date/Time: 2019-03-27 08:57:38	Latency		Disk		Timing)	ADCs_Temp	erature	Daq_Box	es_Temperatures
Alert type: M Provider: LockMoni	MS_machines	DetOp_machin	es olserve	rs	rtpcs	CoilSwitchBoxes	INF_device	es ENV_c	levices	VAC_devices
C. flag: META_ITF_LOCKindex	CalNE	CalWE	CalINJ	CalBS	CalPR	PCalNE	PCalWE	HOFT	NCAL	NoiseInjection
Date (Time: 2010.02.27.08/52/26	SoftwareAl		Temperature	sAl	Injectior	nAl	UpsA	1		TcsAl
Alert type: 3	v	Horizon			flag_AlertGra	ceDB 🌲		STA	TE_VECTOR	8
		F	iaure 20	6 -DMS	alert s	section				

8.1.5 The Shelving section

In this section, condition-flags currently shelved displayed. For each shelving, the following information is displayed:

- the UTC time at which the shelving was inserted;
- the name of condition-flag for which the shelving has been done;
- the name of the user that made the shelving;
- the UTC time at which the shelving starts;
- the UTC time at which the shelving ends;
- the reason of the shelving

For more details see also <u>How to open the shelving section</u>, <u>How to shelve-unshleve a</u> <u>condition flag</u>.

Currently shelved: SIB1_1 Currently shelved: Laser 27/11/18 15:46 - Chillers MC_Pow Point MC_Pow Shelved by: Chiller_Alarm_10 PO Shelved by: Gattalo SDB1_1 On: 27/11/18 15:46 B2_8MHz	P SII	31_BENCH MC_PAY aserAmpli PSTAB PD_B1p	La:	SIB1_BR MC_BR serChiller IMC_AA	SIB1_V MC_Ve SL_TempCo	ert ert	SIB1_TE MC_TE		SIB1_Guard MC_Guard		SIB1_Electr
Currently shelved: MC_1f 27/11/18 15:46 - Chillers MC_Pow mean_TCSRW1_Chiler_Alam_10 PD Shelved by: datbio On: 27/11/18 15:46 SDB1_1 SDB1_1 Extra 7211/18 10:40 B2_8MHz	er C	MC_PAY aserAmpli PSTAB 2PD_B1p	La:	MC_BR serChiller [MC_AA	MC_Ve SL_TempCo	ntroller	MC_TE		MC_Guard		MC Flashs
2/11/18 15:46 - Chillers Laser mean_TCSRW1_Chiler_Alam_10 PD Shelved by: datalo SDB1_1 Or:: 2711/18 15:46 SDB1_2 Shelved by: datalo SDB1_3	er C P S	PSTAB PD_B1p	La	serChiller IMC_AA	SL_TempCo	ntroller					
27/11/18 15:46 - Chillers MC_Pow mean_TCSR_W1_Chiller_Alarm_10 PD Shelved by: datilio SDB1_1 On: 27/11/18 15:46 SDB1_1 B2_SMHz_ B2_SMHz_	er C P S	PSTAB PD_B1p	I	IMC_AA			RFC		LNFS		PC 👧
mean_TCSR_WI_Chiller_Alarm_10 PD Shelved by: datilio SDB1_1 On: 27/11/8 15:46 SDB1_2 Shelt 27/11/8 15:46 B2_8MHz_		PD_B1p	(IMC_AA_C	ALVO	MC_F0_z		BPC		BPC_Electr
Shelved by: dattilo SDB1_1 On: 27/11/18 15:46 B2_8MHz_ Start: 37/11/18 14:00 B2_8MHz_		DB1 LC		QPD_B2	QPD_I	35	OMC		PicoDisable		Shutter 🔒
B2_8MHz_				DB1_BR	SDB1_\	/ert	SDB1_TE		SDB1_Guard		SDB1_Electr 👩
Start: 27/11/19 14:00	JPHI 64_5	6MHz_DPHI	DA	ARM_UGF	UNLO	к	SSFS_UG		FmodErr		GIPC
Stop: 01/05/20 14:10 B1p.	DC	B4_112MHz_N	1AG	B7_C	bC	B8	DC	LSC_	rms		ASC_rms
Comment: Chappel not available with the BS IP	BS	F7	BS PAY	B	IS_BR	BS Ver	rt	BS_TE	BS Gua	rd	BS Electr
AdV chiller (availale only with teh old type). NI IP	NI		NI PAY		I BR	NI Ver	rt	NI TE	NI Gua	rd	NI Electr
Today the new chiller has been reconnected (#43773) NE_IP	NE	F7	NE_PAY	N	IE_BR	NE Ver	rt	NE_TE	NE Gua	rd	NE Electr
PR IP	PR		PR PAY	P	RBR	PR Ver	rt	PR TE	PR Gua	rd	PR Electr
SR IP	SR	F7	SR PAY		RBR	SR Ver	rt	SR TE	SR Gua	rd	SR Electr
26/03/19 14:01 - ACS_WE_ALARMED WI IP	WI	F7	WI PAY		VI BR	WI Ver	rt	WI TE	WI Gua	rd	WI Electr
mean_INFWEB_PRES_OUT_60 WE IP	WE	F7	WE PAY	E_PAY WE_BR		WE Ve	rt	WE TE	WE_Guard		WE Electr
Shelved by: soldani CB Hall	MC H	lall	TCS zone	nes NE Hall		WE Ha	all WindActivity		y Seismon		BRMSMon
INJ Area	DET_Area	EE R	loom	DAQ Room	Exterr	al [DeadChannel	Lights	Sea/	Activity	WAB
Start: 26/03/19 14:00 Stop: 27/03/19 14:00 CS_CB_Ha	ACS_TB	ACS_DA	Q_Room	ACS_EE_Roon	n ACS_N	1C	ACS_INJ	ACS_DET	AC	S_NE	ACS_WAB
Comment: false alarm due to strong wind UPS_TB	UPS_MC	UPS	NE	NE UPS WE		FlatChannel Exist		tChannel ACS_WE		WE ACS_CB_CR	
EIB_SBE	SDB2_SB	SDB2	2_LC	SNEB_SBE	SNEB_	LC	SWEB_SBE	SWEB_LC	SPR	B_SBE	SPRB_LC
P- Vet hay	NI_CO2_Las	er			WI_CO2_	Laser			Chillers		
26/03/19 12:21 - O2_NE PLL	5	queezer	S	SQZ_AA	SQZ_Sh	utter	Cohe_CTR	L	SQZ_Inj		Rack_TE
mean_VACNE_02_TUNNELDOOR_UP_5	s Clean	_Air	TubeStation	ns Tub	ePumps	MiniTowe	ers Ti	irboLinks	RemDry	РМР	VAC_SERVOS
26/03/19 12:21 - Pressure Pressure	e Con	npressedAir	Tow	verServers	TowerPu	mps	CryoTrap		O2_Sensors		Tank
VAC_LINKSR_PR1 ctorSEnviro	<mark>nment</mark> Control	Room	Minitowers		ISC	Injectio	on	TCS	Suspens	ion	Vacuum
26/03/19 12:21 - TurboLinks DetectorM	onitoring	DataCollectio	on	Stora	ge	Data	Access	Autom	nation		DetChar
VAC_LINKSRSCU1600TEMP	tency	1	Disk				ADC	s_Temperature		Daq_Box	es_Temperatures
25/02/19 11:49 ACS DET ALARMED MS_machi	ies DetOp_m	achines	olservers		rtpcs	CoilSwitch	Boxes IN	F_devices	ENV_dev	ices	VAC_devices
mean_HVACDET_FLUX_OUT_10 CalNE	CalWE	CalINJ	0	CalBS	CalPR	PCalNE	PCalWE	НО	FT	NCAL	NoiseInjection
Sof	wareAl	Tem	peraturesA	Al I	Injectio	nAl		UpsAl			TcsAl
25/03/19 11:19 - ACS_DET_ALARMED mean_HVAC_DET_FLUX_IN_10	Horizon				flag_AlertG	raceDB			STATE	VECTOR	

Figure 27 – DMS shelving section

8.1.6 The Muting section

In this section, condition-flags currently muted are displayed.

- For each shelving, the following information is displayed:
 - the UTC time at which the muting was inserted;
 - the name of condition-flag for which the muting has been done;
 - the name of the user that made the muting;
 - the UTC time at which the muting starts;
 - the UTC time at which the muting ends;
 - the reason of the muting

For more details see also <u>How to open the muting section</u>, <u>How to mute-unmute a condition</u> <u>flag</u>.

▲ Muting	^		ITF	Mode: Com	missionin	1g (0d 2h 41m 54s	ITF Stat	te: LOCKING_	OMC1_B1s2_	DC (0d 0h 1m 40s)		019-03-27 08:43	
, maxing	SIB1_IP	SIB1_B	ENCH	SIB1_B	R	SIB1_	/ert	SIB1_1	TE I	SIB1_Guard	1	SIB1_Electr	•)
Currently muted:	MC_IP	MC_F		MC_B		MC_Vert						MC_Electr	0
	Laser	LaserA	LaserAmpli Las		serChiller SL_TempContr		ontroller	RFC		LNFS		PC	B
18/05/18 23:56 -	MC_Power	PST/	PSTAB IMC_AA		A	IMC_AA_	GALVO	MC_F0		BPC		BPC_Electr	
ACS_CB_HALL_ALARMED mean HVAC CEB HOT CORR 60	PD	QPD_I	31p	QPD_B		QPD_	B5	OMC		PicoDisable		Shutter	
muted by: berni	SDB1_IP	SDB1		SDB1_E	3R	SDB1_	Vert	SDB1_	TE	SDB1_Guar		SDB1_Electr	Ø
On: 18/05/18 23:56	B2_8MHz_DPHI	B4_56MH	z_DPHI	DARM_U	GF	UNLO	СК	SSFS_U	GF	FmodErr		GIPC	
Start: 18/05/18 08:00	B1p_DC	B4_	112MHz_MAG		B7_D0		В	8_DC	LS	iC_rms		ASC_rms	
Stop: 18/05/19 08:00	BS_IP	BS_F7	BS	_PAY	BS	S_BR	BS_Ve	ert	BS_TE	BS_G	uard	BS_Electr	
Comment: Not requested by Davide	NI_IP	NI_F7		_PAY	NI	I_BR	NI_Ve	ert	NI_TE	NI_G	uard	NI_Electr	
M View flag	NE_IP	NE_F7	NE	_PAY	NE	_BR	NE_Ve	ert	NE_TE	NE_G	uard	NE_Electr	
	PR_IP	PR_F7	PR	_PAY	PR	L_BR	PR_Ve	ert	PR_TE	PR_G	uard	PR_Electr	
18/05/18 23:56 - ACS CB HALL ALARMED	SR_IP	SR_F7	SR	_PAY	SR	R_BR SR_Ve		ert	SR_TE	SR_G	uard	SR_Electr	
mean_HVACCEB_COLD_CORR_60	WI_IP	WI_F7	WI	_PAY	WI	I_BR	WI_Ve	ert	WI_TE	WI_G	uard	WI_Electr	
muted by: berni	WE_IP	WE_F7	WE	_PAY	WE_BR		WE_V	WE_Vert WE_TE		WE_Guard		WE_Electr	
On: 18/05/18 23:56	CB_Hall	MC_Hall	TCS	zones	NE	_Hall	WE_H	iali 🛛 🔪	VindActivity	Seisr	non	BRMSMon	
Start: 18/05/18 08:00	INJ_Area	DET_Area	EE_Room	DA	Q_Room	Exter	nal	DeadChannel	Lights	; Se	aActivity	WAB	
Stop: 18/05/19 08:00	CS_CB_Hall	ACS_TB	ACS_DAQ_Ro	om ACS	_EE_Room	ACS_	МС	ACS_INJ	ACS_DI	ET /	CS_NE	ACS_WAB	
Comment: Not requested by Davide	UPS_TB	UPS_MC	UPS_NE	U	PS_WE	FlatCha	nnel	ExistChannel	ACS_W	E AC	S_CB_CR	ACS_COB	
M View flag	EIB_SBE	SDB2_SBE	SDB2_LC	SN	IEB_SBE	SNEB	LC	SWEB_SBE	SWEB_	LC SF	RB_SBE	SPRB_LC	
12/06/17 11:51 -		VI_CO2_Laser				WI_CO2	Laser			(Chillers		
DetectorSEnvironmentMonitoring	PLL	Squee	zer	SQZ_A	A	SQZ_SH	utter	Cohe_C	rrl 🛛	SQZ_Inj		Rack_TE	
ParticleCounterDer	LargeValves	Clean_Air	Tubes	Stations	Tube	Pumps	MiniTov	vers	TurboLinks	RemDr	yPMP	VAC_SERVOS	\$
On: 12/06/17 11:51	Pressure	Compres	sedAir	TowerSer	vers	TowerP	imps	СгуоТга	ар	O2_Sensor		Tank	
Start: 18/04/15 00:00	ctorSEnvironment	ControlRoom	Mini	towers		SC	Injecti	ion	TCS	Suspe	nsion	Vacuum	
Stop: 03/05/19 00:00	DetectorMonitorin	g Da	ataCollection		Storag	e	Dat	aAccess	Aut	omation		DetChar	
Comment: Alert notification not needed	Latency		Dis	k		Timi		AI	OCs_Temperat	ure	Daq_Bo>	es_Temperatures	
P View flag	MS_machines	DetOp_machin	es olse	ervers	rt	pcs	CoilSwitch	1Boxes I	NF_devices	ENV_d	evices	VAC_devices	
	CalNE	CalWE	CalINJ	CalBS		CalPR	PCalNE	PCall	VE I	HOFT	NCAL	NoiseInject	ion
12/06/17 11:51 - DetectorSEnvironmentMonitoring	SoftwareA		Tempera	turesAl		Injecti	onAl		UpsAl			TcsAl	
ParticleCounterINJ	v 1	Horizon				flag_AlertG	aceDB 🌲			STAT	E_VECTOR		

Figure 28 – DMS muting section

8.1.7 Associated Condition Flag plots

When selected this provides recent information on the state of the called Condition Flag. This takes the following form:

- by default the "up-sampled" data are displayed, with the option to choose the "downsampled" data. "up-sampled" data are the last 30 minutes of data sampled every 10s while the "down-sampled" data are the last 3 weeks of data sampled every 60s.
- In the "up-sampled" data mode the plot is automatically refreshed every 10 seconds;
- In the "down-sampled" data mode the plot is paused;
- the data are zoom-able;
- it is possible to pan across previous and successive time periods;
- the background provides a shaded representation of the alert state associated to the Condition Flag during the selected time period;
- there is the possibility to pause the refresh of the plot;
- there is the possibility to download the plot as a png image.

For more details see also <u>How-to for Associated Condition Flag plots</u>.



8.1.8 Responsiveness

The sub-systems display differently dependent upon the size of the screen being used. On small screens, the DMS main-screen area display as follows on small-screens:

H DMS UTC: 2019-03-27 10:48:53	
1TF State: DOWN (0d 0h 0m 22s)	
ITF Mode: Commissioning (0d 4h 47 m 33s)	
Injection	
Detection	
ISC	
Suspensions	
Environment	
Infrastructures	
тся	
sqz	
Vacuum	
VPM	
DAQ-Computing	
Calib_Hrec	
ITFOnCall	
DetChar	

Figure 30 – DMS homepage on small screen

Note that, as long as performance allows, clicking on a flag should not open a new page in the browser, instead it should simply display the information that by default is hidden within the page, via a JavaScript onclick event.

Clicking on the name of the sub-system should re-close the sub-system, i.e. re-hide the associated flags.

Clicking on an individual flag shows its detailed information in the left nav-bar.

8.2 The DMS event monitor WUI

The DMS event log WUI shows the following DMS events:

- shelving events: new shelving, shelving expired, manual unshelve;
- muting events: new muting, muting expired, manual unmute;

- SMS alert notification;
- EMAIL alert notification;
- SOUND alert notification

Events are associated to one of the following color:

Color	Event type
	shelving
	muting
	SMS
	EMAIL
	SOUND

Through the section on the left it is possible to set 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.

The possible searchable areas are:

- Event type
- Condition flag
- Time window

The events are listed on the right part in ordered way; from the most recent to the less recent. For more details see also <u>How-to for DMS event monitor</u>.

Filter toolbar	e			Events	
Select event type					
	Filtering				
Shelving	Selected quant hungu shelidi	a mutica eme mail e	ound -		
✓ « muting	Selected condition flag: All Selected time window (yy-n	nm-dd, LT): from 2019-02-:	25 00:00:00 to 2019-03-27 24:00:00		
🗹 🛛 sms alert	Found 1506 events				
✓	⊞90 shelving events				
	⊞ 14 muting events ⊞ 164 sms events				
Sound alert	⊞ 532 mail events ⊞ 736 sevent events				
	E 700 sound events				
Select condition flag	Date (LT)	Event	Condition-Flag	Flag	Provider
Vent VetoCollect	2019-03-27 11:59:40	sound ھ	META_ITF_LOCK_index	UNLOCK	LockMoni
VetoMergerCollect VetoThr	2019-03-27 11:55:17	sound 🕅	META ITE LOCK Index		
VetoThrCollect vic_channel_reader waam_torry	2019-03-27 11:48:54	a sound	META ITE LOCK Juden		
WEB_DBOX_TCS_PCAL_timing_error WEB_DBOX_TCS_PCAL_timing_error	2019-03-27 11:13:32		meta_in_cock_index	UNLOCK	LockMoni
WEB_PCal WEB_PCal_Fast WF Db lat	2010 02 27 10 29 52	W Sound	MEIA_IIIP_LOCKindex		Leibhen
WE_Dh_lon _HLStubeN_meanHLSvalve_meanDh_lon_0+offset_lon	2019-03-21 10-20-32	a sound	META_ITF_LOCKindex		
HLStubeN_meanHLStubeW_meanDh_lat_0+offset_lat	2019-03-27 10:26:52	sound 🕅	META_ITF_LOCKindex	UNLOCK	LockMoni
Select all O Deselect all	2019-03-27 09:34:41	 shelving 	mean_INFNEB_PRES_OUT_60	ACS_NE_ALARMED	InfraAlarmedMoni
Search and select condition flags	2019-03-27 09:34:21	sound 🕷	META_ITF_LOCKindex		
	2019-03-27 09:30:06	ŵ sound	GraceDB alert	flag_AlertGraceD8	
Select time window (yy-mm-dd, LT)	2019-03-27 09:28:41	a sound	META_ITF_LOCKindex		
From: 2019-02-25 00:00:00	2019-03-27 09:08:14	a sound	META_ITF_LOCKindex		
To: 2010 02 23 04 00 00	2019-03-27 09:04:16	🖂 mail	kalert viroo	DetChar	VPM
2019-03-27 24:00100	2019-03-27 08:57:38		META_ITF_LOCKindex		
	2019-03-27 08:53:26	a sound	META ITE LOCK index		
SUBMIT	2019-03-27 08:46:19	a sound	META ITE LOCK index		
	2019 03 27 08:42:08				LockMoni

8.3 The DMS playback WUI

ThePlay-back section allows users to display the DMS in the exact configuration and state in which it was at a given GPS time or between two different GPS times.

In the event of a single GPS time being selected, the main-content area ceases to display the currentDMS and instead shows the image at the requested time.

If both GPS-start and GPS-stop times are provided, then main-content area again ceases to display the current DMS information and instead shows the state between the two times. By default, the WUI loops through the snapshots within the GPS-time range, changing them with a customable frequency. However, the user is be able to pause/un-pause the play-back as well as change the play-back speed; both in terms of speeding it up and slowing it down.

A DMS Playback													
Filter toolbar	A DMS		1	TF Mode:	Science (od	2h 15m 29s)	ITF St	ate: LOW_	NOISE_3_S	6QZ (Od 2h	16m 44s)	UTC: 2019	-03-31 12:00:35
		SIB1_IP	SIB1_	BENCH	SIB1	BR	SIB1	_Vert	SIB1_	re 🛛		_Guard	SIB1_Electr
UTC start		MC_IP	MC	PAY	MC_	BR	MC	Vert	MC_T	E	MC_	Guard	MC_Electr
2010 02 21 12:00:00	Injection	Laser	Lase	rAmpli	LaserC	hiller	SL_Temp	Controller	RFC			NFS	PC
2019-03-31 12:00:00		MC_Power	PS	TAB	IMC_	AA	IMC_A/	GALVO	MC_F0	z	E	BPC	BPC_Electr
Snapshot rate		PD	QPD	_B1p	QPD	_B2	QPI	D_B5	OMC		Picol	Disable	Shutter
1	Detection	SDB1_IP	SDB		SDB1	_BR	SDB:	L_Vert	SDB1_	TE	SDB1	L_Guard	SDB1_Electr
-		B2_8MHz_DPH	I B4_56M	Hz_DPHI	DARM	UGF	UNI	.OCK	SSFS_U	IGF	Fm	odErr	GIPC
Snapshot pause [s]	ISC	B1p_DC	B4_	112MHz_I	MAG	B7_C	C	E	8_DC		LSC_rm	s	ASC_rms
1		BS_IP	BS_F7		BS_PAY	BS	S_BR	BS_\	/ert	BS_TE		BS_Guard	BS_Electr
· ·		NI_IP	NI_F7		NI_PAY	N	I_BR	NI_V	/ert	NI_TE		NI_Guard	NI_Electr
		NE_IP	NE_F7		NE_PAY	N	E_BR	NE_\	/ert	NE_TE		NE_Guard	NE_Electr
	Suspensions	PR_IP	PR_F7		PR_PAY	PF	R_BR PR_		Vert PR_TE			PR_Guard	PR_Electr
Json File		SR_IP	SR_F7		SR_PAY	SI	R_BR	SR_\	/ert	SR_TE		SR_Guard	SR_Electr
		WI_IP	WI_F7		WI_PAY	W	I_BR	WI_V	/ert	WI_TE		WI_Guard	WI_Electr
📩 Upload		WE_IP	WE_F7		WE_PAY	_PAY WE		_BR WE_Vert		WE_TE		WE_Guard	WE_Electr
Slogia Nessun file selezionato.	hanne and the second	CB_Hall	MC_Ha	ΙТ	CS_zones	NE	_Hall	WE_	Hall W	WindActivity		Seismon	BRMSMon
	Environment	INJ_Area	DET_Area	EE_R	oom D	AQ_Room	Ext	ernal I	DeadChannel	Lig	hts	SeaActivit	/ WAB
🚣 Download		ACS_CB_Hall	ACS_TB	ACS_DA	Q_Roon ACS	EE_Roo	m ACS	_MC	ACS_INJ	ACS	CS_DET ACS_NE		ACS_WAB
1238068853.json	Infrastructures	UPS_TB	UPS_MC	UPS_	NE	UPS_WE	FlatC	hannel	ExistChannel	ACS	_WE	ACS_CB_C	R ACS_COB
	SBE	EIB_SBE	SDB2_SBE	SDB2	LC S	NEB_SBE	SNE	B_LC	SWEB_SBE	SWE	B_LC	SPRB_SBB	SPRB_LC
Display	TCS	1	VI_CO2_Lase	r			WI_CO	2_Laser				Chillers	
Construction	sqz	PLL	Squ	eezer	SQZ	_AA	SQZ_	Shutter	Cohe_C	TRL	SQ	Z_Inj	Rack_TE
Shapshot		LargeValves	Clean_A	ir Tu	beStations	Tube	Pumps	MiniTo	wers T	urboLinks	s F	RemDryPMP	VAC_SERVOS
Info and debug	Vacuum	Pressure	Compre	essedAir	TowerS	ervers		Pumps	CryoTr	ар	02_9	Sensors	Tank
	1001	DetectorSEnviron	ControlRo	om M	linitowers	1	ISC	Injec	tion	TCS	5	Suspension	Vacuum
Log	VPM	DetectorMonito	ring D	ataCollecti	on	Stora	ge	Dat	aAccess	A	utomatio	on	DetChar
	DAG Committing	Latency		Disk		Timir	ng	Fa	st_DAC	ADCs	_Tempe	rature Daq_	Boxes_Temperatur
- gps start: 1238068800	DAQ-computing	DMS_machines	DetOp_mac	hines	olservers	r	tpcs	CoilSwite	hBoxes IN	IF_device	es E	NV_devices	VAC_devices
- archive: 1238030424.zip	Calib_Hrec	CalNE	CalWE	CalINJ	CalB	S	CalPR	PCalN	E PCall	VE	HOFT	NCAL	NoiseInjection
entry in 3654 (tot. 5116)	TTEOnCall	Software	<u>۸۱</u>	Tomn	oraturocAl		Thior	tionAl		[]neAl			TreAl
	Elauro	22 10	MC ml	avika	AL 11		naha	1" d	icplay	,			



DMS Playback	
Filter toolbar	SNAPSHOT INFO
UTC start	Shapshot, time 2019-03-01 (2000-030 tor), (22000-030 tor), stavid Stav
o re start	Configuration Inter / data/archive_cmms/son_compguration_Intes/ / 27.00 Configuration Ison: / data/archive_dme/configuration_Intes/ / 27.00
2019-03-31 12:00:00	compared on the campacent part of the campac
	N. subsystems: 15
Snapshot rate	N. gorup Itagis: 61 N. Brow 249
	N. condition flags: 2704
1	
	ITF INFO
Snapsnot pause [s]	ITF state: LOW_NOISE_3_SQZ since 1238060653 GPS, 2019-03-31 09:43:55 UTC; length: 0d 2h 16m 44s, 8204 seconds
1	ITF mode: Science since 1238060728 GPS, 2019-03-31 09:45:10 UTC; length: 0d 2h 15m 29s, 8129 seconds
1	
	AlertGradeUdS green (1, vo alarm) RBMSNor green (1 No alarm)
	Californi green (1, No alarm)
lson File	ComputingDMS: green (1, No alarm)
556111110	DAQMoni: green (1, No alarm)
the local	DOMONI: green (1, No alarm)
- Opload	Environi green (1, No alarm)
Sfogla Nessun file selezionato.	ITFOnCallMoni: green (1, No alarm)
	ITFOperationDMS: green (1, No alarm)
Download	InfraAlarmedMoni: green (1, No alarm)
	InfraMonia green (1, No alarm)
1238068853.json	Initionity green (1. No alarm)
	LockMoni: green (1, No alarm)
Display	ServersMoni: green (1, No alarm)
and the R	SczMonie green (1, No alarm)
Snapshot	Suspection upreen (1, No adam)
	SuspShottMoni: green (1, No alarm)
A Info and debug	TCSMoni: green (1, No alarm)
• Into and debug	VPM: green (1, No alarm)
	vacuunikuni: green (1, vuo alainti)
Log	MILTED CONDITION FLAG (No alarm patification sent)
	Detton (d muting: 14)
- gps start: 1238068800	InfraMoni (id_muting: 15)
- archive: 1238030424.zip	LockMoni (id_muting: 16)
- entry n. 3654 (tot. 5118)	ParticleCourterDIG (m. muting: 9) ParticleCourterDIG (m. muting: 10)
	Figure 33 – DMS playback, "info and debug" display

There is the possibility to select the desired display:

- Snapshot: is the standard visualization of the DMS;
- Info and debug: is a different visualization of the DMS in which all the info contained in the JSON payload are shown in a more readable way.

The JSON payload can be downloaded and uploaded.

A dedicated python script, called GetPlaybackLinks, can be launched to get directly the link to a desired snapshot. The script needs as argument the GPS start and the GPS stop, it extracts from the archives all the snapshot between the two GPS and provides the URLs.

For more details see also <u>How-to for DMS playback</u>.

8.4 The DMS archive WUI

The DMS archive section allows user to retrieve one year of information on the state of the called Condition Flag. The information is organized in twelve boxes; one box for one month. Each box behaves as the plot associated to a condition flag. The user can select:

- Time window:
 - Last 12 months:
 - Year 2017, 2018, etc ...;
 - The sampling rate of the plot:
 - 60s
 - o 600s
 - o 1800s
 - o 3600s
 - The y scale:

0

- o autoscale;
 - manual scale:
 - y min;
 - y max;
- the plot resolution
 - x width [px]
 - y width [px]

For more details see also How-to for DMS archive.



8.5 The DMS currently shelved condition-flags WUI

The DMS currently shelved condition-flags shows the detailed information about the condition flags currently shelved. The information can be ordered by "ASC" or "DESC" by clicking on the specific column. After authentication the user can add additional note to the shelving, build a report to store an HTML file and also to un-shelve the condition flag.

For more details see also How to for the DMS currently shelved condition-flags.

List of the reports	×							
2020-04-10_08:11:30_berni.html		Condition flag ¥	UTC start	UTC stop	Comment	Checked	Additional note	Unshelve button
2020-04-09_12:11:52_berni.html		VAC_TUBE1800WV21ST	25/02/20 12:00	28/04/20 09:00	Offline , under monintoring.			Un chelvo
2020-03-11_20:02:15_Gherardini.html							lione	Un-snerve
2020-03-04_21:14:10_amagazzu.html		VAC_CRYOWIVCRYOOPENRELAYST	14/04/20 09:30	28/04/20 09:30	stdby		none	Un-shelve
2020-02-12_19:18:00_amagazzu.html			14/04/20 00:20	20/04/20 00:20	atabu			
2020-02-05_20:04:06_menzione.html		VAC_CKTOWE_VCKTOCLOSEDRELATST	14/04/20 09:20	28/04/20 09:30	stuby		none	Un-shelve
2020-01-29_17:39:28_berni.html		VAC_CRYONIVCRYOOPENRELAYST	14/04/20 09:30	28/04/20 09:30	stdby		none	Un-shelve
2020-01-22_20:38:38_Gnerardini.ntml								on share
2020-01-15_20:55:45_menzione.num	_	VAC_CRYONEVCRYOOPENRELAYST	14/04/20 09:30	28/04/20 09:30	stdby		none	Un-shelve
2020-01-01_20:00:29_0herarum.html	av	TCS CO2 REL7	06/04/20 08:00	31/12/20 08:00	ITE securing from March			
2019-12-23_20.10.34_magazzo.html 2019-12-11 17:10:17 herni.html	.,	100_001_100		51/12/20 00:00	31st onward for O3		none	Un-shelve
2019-12-04 21:50:47 amagazzu.html					suspension			_
2019-11-27 19:48:40 menzione.html	ау	TCS_CO2REL3	06/04/20 08:00	31/12/20 08:00	31st onward for O3		none	Un-shelve
2019-11-20 18:34:35 berni.html					suspension			
2019-11-13_19:57:13_Gherardini.html		SWEB_LC_Y_err	31/03/20 08:00	31/12/20 08:00	ITF securing from March 31st onward for Q3		none	Un-shelve
2019-11-06_18:53:35_menzione.html					suspension			
2019-10-16_20:14:13_amagazzu.html		SWEB_LC_TZ_err	31/03/20 08:00	31/12/20 08:00	ITF securing from March		none	Un-shelve
2019-09-25_14:14:42_magazzu.html					31st onward for O3 suspension			
2019-09-18_16:02:19_menzione.html		SWEB_LC_TY_lowNoise_disable_inv	31/03/20 08:00	31/12/20 08:00	ITF securing from March			Up abelor
2019-09-11_13:24:33_berni.html					31st onward for O3 suspension		none	On-sherve
2019-09-04_14:11:35_Gherardini.html		CHIER LC TOUT with	21/02/20 00:00	21/12/20 00:00	TTP annulus from Marsh			_
2019-08-30_07:04:07_berni.html		SWED_LC_TATZ_ellbr	31/03/20 08:00	51/12/20 08/00	31st onward for O3		none	Un-shelve
2019-08-30_05:48:04_test.html					suspension			
2019-08-21_14:04:17_nencl.html		SWEB_LC_TX_err	31/03/20 08:00	31/12/20 08:00	ITF securing from March 31st onward for O3		none	Un-shelve
2019-08-14_14:01:55_gherardini.html					suspension			
2019-08-08_13:23:32_menzione.html		SWEB_B8_Cam1_FitPosY	06/04/20 08:00	31/12/20 08:00	ITF securing from March		none	Un-shelve
2019-08-07_16:40:52_Magazzu.html					suspension			

9 DMS workflow

This section describes with same example how a flag can be created from scratch: from the configuration of a Moni process, passing to the configuration of the DMS and then the visualization on the WUI.

9.1 Moni process configuration



Figure 36 - InfraMoni.cfg, detail of the configuration for ACS_DET_ALARMED

9.2 Moni process output: JSON payload

```
"ACS_DET_ALARMED": {
    "flag_state": 5,
    "flag_dq_value":1,
    "condition_flags" : {
    ...
    "ACS_DET_ALARMED_04": {
        "channel_name": "HVAC..DET_FLUX_IN",
        "condition_flag_thresholds": "mean(HVAC..DET_FLUX_IN,10) < 1000",
        "condition_flag_comment": "Possible_AHU_failure",
        "condition_flag_computed_value": 10402.7,
        "condition_flag_state": 5,
        "condition_flag_dq_value": 1
      },</pre>
```

Figure 37 – QcInfraMoni.json, flag ACS_DET_ALARMED branch

9.3 DMSserver configuration

FLAG ACS_DET_ALARMEDInfrastructures1ACS_DET600CONDITIONING_MAIL,OPERATION_MAIL600CONDITIONING_SMS6000

Figure 38 – DMSserver.cfg, detail of the configuration for ACS_DET_ALARMED

9.4 DMSserver output: snapshot JSON payload

```
"Infrastructures": 3,
'group flags": {
    "ACS DET": 1,
    "mean_HVAC__DET_FLUX_IN_10": {
      "id_shelving": 916
"providers": {
    "DetMoni": {
      "state": 1
"flags": {
     "ACS DET_ALARMED": {
      "state_delay_before_error": 0,
"flag_state": 1,
      "cond_flags": {
           "mean_HVAC__DET_FLUX_IN_10": {
              "condition_flag_computed_value": 10402.7,
              "condition_flag_comment": "Possible_AHU_failure",
              "condition thresholds": "mean(HVAC..DET FLUX IN,10) < 1000",
              "condition flag state": 1
           Figure 39 - online_snapshot.json, flag ACS_DET_ALARMED branch
```

9.5 Homepage

i nis is a group nag.	•	Eile Modifie	ca <u>V</u> isualizza <u>C</u>	ronologia S <u>e</u> gnalit	bri Strumenti	i <u>A</u> iuto			-		ITE S	tate: LOW	NOTSE 3	SO7 (od 16	25m E4a)	LITC: 201	9-04-02 15-49-2	5
Sub- and condition flags:	CIDI	DMS - Data	Viewer	× +								CID4			25111 545)	010.201		2
-	MC	$\leftrightarrow \rightarrow$	C ŵ	0 A https://	dms. virao-a	w.eu/d:	🛛 🖞	Q Cerca	III)				-	MC	_Guard		MC Electr	+0
ACS_DET_ALARMED	Las	+	D David	bod								RFC			NES		PC	Ų
D: mean ENV DIAB TE 10	MC P		- Cus	JET_ALAR	RMED -	mean_H	IVAC	DET_FLU	X_IN_10			MC FO			RPC		BPC Electr	629
21.5 > mean(ENVDLAB_TE,10) > 19.5	PI	11000						2019/ Value	/03/19 23:39:0 10402.7	2:		OMC		Pico	Disable		Shutter	•
(value = 20.3146)	SDB1				•		- <u>r</u>					SDB1		SDB1	Guard		SDB1 Electr	-
IDI mean HVAC DET DER TE 10	B2 8MH	10000										SSFS L	JGF	Fm	odErr		GIPC	49
24 > mean(HVACDET_DER_TE,10) > 18	B1	9000									B8 C)C		LSC rms			ASC rms	
	BS_I										/ert		BS_TE		BS_Guar	rd d	BS_Electr	
ID: mean HVAC DET DLAB TE 10	NI_I	8000									/ert		NI_TE		NI_Guar	ď	NI_Electr	
21.3 > mean(HVACDET_DLAB_TE,10) >	NE_I										/ert		NE_TE		NE_Gua	rd 🛛	NE_Electr	
(Value = 20.9122)	PR_I	7000									/ert		PR_TE		PR_Guar	ď	PR_Electr	
∽ ◀ 🗠 🖸 🖺	SR_I	6000									/ert		SR_TE		SR_Guar		SR_Electr	
ID: mean HVAC DET FLUX_IN_10	IWII	0000									/ert		WI_TE		WI_Guar		WI_Electr	
mean(HVACDET_FLUX_IN,10) < 1000	WE_I	5000									/ert		WE_TE		WE_Gua	rd	WE_Electr	
(Value = 10402.7)	CB_Ha										Hall		WindActivi		Seismor		BRMSMon	
	INJ_Are	4000									Dea	adChannel	L	ights	SeaA	ctivity	WAB	
1100	CS_CB_I	2000									A	ACS_INJ	AC	S_DET	ACS	NE	ACS_WAB	
	UPS_TB	3000									nel	ExistCh	annel	ACS_WE	AC	S_CB_CR	ACS_COB	
1000	EIB_SB	2000					_					VEB_SBE	SW	EB_LC	SPRE	5_5BE	SPKB_LC	
	DI											Caba C		Laser	7 101		Deek TE	
	PL Largo\/a	1000									Word	Cone_C	Turbol ink			мр		
ID: mean_HVAC_DET_FLUX_OUT_10	Pros	0									Wers	CryoTr	an	02 9	Sensore		Tank	
mean(HVACDET_FLUX_OUT,10) < 200 (Value = 2054.32)	ctorSEnv		14 Mar 16	Mar 18 Mar	20 Mar	22 Mar	24 Mar	26 Mar 21	8 Mar 30 Mar	01 Apr	tion		TCS		Suspensi	on	Vacuum	
Ø 🖌 🗠 🖸 🖺	Detector	rMonitor	ina	Data	Collection	n	1	Storag	e		DataAc	cess		Automatio			DetChar	
ID: mean_HVACDET_FREQ_IN_10	Latend	cy		Disk		Timing		Timir	ng rtpc	Tim	ng dsp		Fast DAC		ADCs T	E	Dag Boxes TE	
40 > mean(HVACDET_FREQ_IN,10) > 13 (Value = 19.8272)	MS_mac	thines	DetOp	_machines		olserver		rt	tpcs	CoilSv	itchBo	(es	INF_device	s	ENV_devi	ces	VAC_devices	
Ø> 🖌 🗹 🚺	CalNE		CalWE		alINJ		CalBS		CalPR	PCa	NE	PCal	WE	HOFT		NCAL	NoiseInjectio	n
ID: mean_HVAC_DET_FREQ_OUT_10	Soft	wareAl		Tempe	eratures	AI		Injectio	nAl		Ups/	Al		GeneratorA	d I		TcsAl	
65 > mean(HVACDET_FREQ_OUT,10) > 30 (Value = 37.5361)	,	ŀ	Irec_RANC	SE_BNS					flag_Aler	tGraceDB					STATE_	VECTOR		

Figure 40 – DMS homepage, individual-flag-information section for ACS_DET_ALARMED, associated condition flag plots for ACS_DET_ALARMED

9.6 DMSpublisher: reading raw data and writing it to Trend

ctrl6[/data/archive_dms/trend/conditional_flags/archive]: ll drwyrwyr-y 2553 yirgorun yirgo 98304 Dec 15 2017 2017
drwxrwxr-x. 4485 virgorun virgo 172032 Dec 23 19:33 2018
drwxrwxr-x. 3366 virgorun virgo 135168 Apr 2 17:36 2019
ctrl6[/data/archive_dms/trend/conditional_flags/archive/2018]:
abs_Dh_lat abs_NE_Dh_lat
 drwxrwxr-x. 2 virgorun virgo 44 Dec 1 00:28 mean_HVACDET_FLUX_IN_10
 WE_Dh_lat
WE_Dh_lon
ctrl6[/data/archive_dms/trend/conditional_flags/archive/2018/mean_HVACDET_FLUX_IN_10]:
02.zip
 11.zip
12.zip
Figure 41 – content of the folder

9.7 DMSpublisher: compression JSON snapshot

ctrl6[/data/archive_dms/archive_snapshots]: ll total 257763584 -rwxrwxrwx. 1 virgorun virgo 544746804 Sep 12 2017 1189074163.zip -rwxrwxrwx. 1 virgorun virgo 535867367 Sep 13 2017 1189214934.zip

-rwxrwxrwx. 1 virgorun virgo	535011050 Sep 15 2017 1189353246.zip
-rw-rw-r 1 virgorun virgo	652417981 Mar 31 03:21 1237925177.zip
-rw-rw-r 1 virgorun virgo	651368959 Apr 1 08:40 1238030424.zip
-rw-rw-r 1 virgorun virgo	658070236 Apr 2 14:25 1238135996.zip
-rw-rw-r 1 virgorun virgo	78787826 Apr 2 17:50 1238243059.zip
	Figure 42 – content of the folder

9.8 DMS event monitor WUI

Filter toolbar	•			Events	
Select event type					
🖬 👁 shelving	Filtering				
✓ 4 muting	Selected event type: shelv Selected condition flag: me	ring - muting - sms - n an_HVACDET_FLU mm_dd_UT): from 201	nail - sound - K_IN_10 - 0.02.02.00:00:00 to 2010.04.02.24:00:00		
🖾 🛛 sms alert	Found 4 events	nin-dd, Er). noin 201	-05-05 00.00.00 to 2019-04-02 24.00.00		
🗹 🖂 mail alert		VUI			
🗹 ត្ថ sound alert	I shelving expired				
	events listed by condition (flag		events listed by recipients	
Select condition flag	1 events for mean_HVAC_DE	ET_FLUX_IN_10		1 events for CONDITIONING_SMS	
mean HVAC_DADB_COLD_TE_50_conflict2	events listed by condition	flag		events listed by recipients	
mean_iNVAC_DUAR_COLD_TIT_ERA_10 mean_INVAC_DUTR_TIL_0_conflict mean_INVAC_DUTDIR_TIL_0_conflict mean_INVAC_DUTDIR_TIL_0_conflict mean_INVAC_DUTDIR_TIL_0_conflict mean_INVAC_DUTD_LUAR_TIL_0_0_10 mean_INVAC_DUTD_LUAR_TIL_0_conflict mean_INVAC_DUTD_LUAR_TIL_0_conflict	Date (LT) 2019-03-25 11:19:32 id event: 32472 id shekving: 916	Event shelving	Condition-Flag mean_HVAC_DET_FLUX_IN_10	Flag ACS_DET_ALARMED	Provider InfraAlarmedMoni
meso-livAc_OIT_DUAL_TLERe_10 meso-livAc_OIT_DUAL_TLERe_10 Select all O Select all Search and select condition flags	new shelving from WUI start: 2019-03-25 11:15:00 stop: 2019-04-08 09:00:00 user: soldani comment: waiting for resolut	tive intervention by Kieba	ck & Peter		
	2019-03-25 09:10:17	🖂 mail	mean_HVACDET_FLUX_IN_10	ACS_DET_ALARMED	InfraAlarmedMoni
Select time window (yy-mm-dd, LT)	id event: 32470 Message sent: flag: ACS_DET_ Recipient: CONDITIONING_N	_ALARMED condition flag MAIL OPERATION_MAIL	; mean_HVACDET_FLUX_IN_10 thresholds: mean(H)	VAC_DET_FLUX_IN,10) < 1000 value:10402.7 comment: Possible_AHU_failure	
From: 2019-03-03 00:00:00	2019-03-25 09:10:17	0 sms	mean HVAC DET FLUX IN 10	ACS_DET_ALARMED	
To: 2019-04-02 24:00:00	id event: 32469 Message sent: flag: ACS_DET, Recipient: CONDITIONING_SI	_ALARMED; condition fla MS			

9.9 DMS playback WUI

Filter toolbar	Elag	dotaile ×	Â	ITF I	Mode: Calib	ration (od :	2h 11m 56s)	ITF S	tate: LOV	N_NOISE	_3_SQZ (0	d 4h 6m 7s)	UTC: 2019	-04-01 00:00
		uetans	1_IP	SIB1_B	ENCH	SIB1_B	R	SIB1	_Vert	SIB	1_TE	SIB1	_Guard	SIB1_Elect
UTC start	film	400 DET	IPIP	MC_P	PAY	MC_BR	र	MC_	Vert		MC_TE		Guard	MC_Elect
2019-04-01 00:00:00	riag	ACS_DEI	ser	LaserA	mpli	LaserChil	ller S	L_Temp	Controller		FC	L	NFS	PC
	Sub-sys	Infrastructures	Power	PSTAB		IMC_A/	A	IMC_AA	_GALVO	MC_	_F0_z		BPC	BPC_Elec
Snapshot rate	Alerts		PD	QPD_B1p		QPD_B	2	QPD	_B5	0	МС	Pico	Disable	Shutter
1 2	Th	is is a group flag.	31_IP	SDB1_	_LC	SDB1_B	BR	SDB1	_Vert	SDB	1_TE	SDB:	L_Guard	SDB1_Ele
	Sub- and cond	ition flags:	Hz_DPHI	B4_56MH:	z_DPHI	DARM_U	GF	UNL	оск	SSFS	S_UGF	Frr	nodErr	GIPC
Snapshot pause [s]	Sub- and cond	nion nags.	1p_DC	B4_1	.12MHz_MA	3	B7_DC		E	38_DC		LSC_rm	S	ASC_rms
1	ACS_DET_A	LARMED	IP	BS_F7	BS	_PAY	BS_	BR	BS_	Vert	BS_TE		BS_Guard	BS_Ele
	ID: mear	_HVACDET_FREQ_IN_10	IP	NI_F7	NI	_PAY	NI_	BR	NI_\	/ert	NI_TE		NI_Guard	NI_Ele
• •	40 > mean(H	VACDET_FREQ_IN,10) > 13 /alue = 19.9328)	19	NE_F7	NE	_PAY	NE_	BR	NE_	vert	NE_TE		NE_Guard	NE_Ele
and the second		1	1P TD	PK_F7	PR		PR_	BR	PK_	vert	PK_IE		PR_Guard	PR_EI
Json File	ID: mean	HVAC DET FLUX_OUT_10	19	SR_F7	SR MI		SR_ W/T	DR DD	SR_	Vert	SK_IE		SK_Guard	WT EL
Unload	mean(HVAC	Shelved DET_FLUX_OUT.10) < 200	TD	WE E7	\\/E		WE	RD	WE	Vert	WE TE		WE Guard	WE EN
	0	/alue = 1811.22)	Hall	MC Hall	TCS	zones	NE I	Hall	WE	Hall	WindActiv	itv	Seismon	BRMS
Noga. Nesson me selezionato.		à	rea	DET Area	EE Roon	DAO	Room	Exte	rnal	DeadChan	nel Li	ahts	SeaActivity	WA
Download	ID: m 21.5 > mea	ean_ENV_DLA8_TE_10 n(ENVDLA8_TE,10) > 19.5	Hall	ACS TB	ACS DAO I	Roon ACS E	EE Room	ACS	MC	ACS IN	ACS	DET	ACS NE	ACS
38112030.json	(\	/alue = 20.3465)	ТВ	UPS MC	UPS NE	UP	S WE	FlatCh	annel	ExistChan	nel AC	S WE	ACS CB C	R ACS
			BE	SDB2 SBE	SDB2 LO	SNE	B SBE	SNE	3 LC	SWEB SE	E SW	EB LC	SPRB SBE	SPRB
Display	ID: mean	HVAC DET FLUX_IN_10 Ø Shelved	NE_RH		WE_	RH		NI_CO2	 2Laser		WI_CO2_	Laser		Chillers
	mean(HVAC	DET_FLUX_IN,10) < 1000	LL	Squee	zer	SQZ_A	A	SQZ_S	hutter	Cohe	_CTRL	SQ	Z_Inj	Rack_T
Snapshot		alde = 10402.7)	alves	Clean_Air	Tube	Stations	TubeP	umps	MiniTo	owers	TurboLinl	ks I	RemDryPMP	VAC_SE
Info and debug	ID: mean	HVAC_DET_FREQ_OUT_10	ssure	Compres	sedAir	TowerServ	vers	Tower	Pumps	Cryo	oTrap	02_	Sensors	Tank
	65 > mean(HV	ACDET_FREQ_OUT,10) > 30	Environ	ControlRoor	m Mini	towers	IS	с	Injec	tion	TCS		Suspension	Vacuu
Log		ر solution (orMonitor	ring Dat	aCollection		Storage	9	Dat	taAccess		Automati	on	DetChar
	ID: mean	HVAC DET DLAB_TE_10	atency		Disk		Timing		Fa	st_DAC	ADC	ADCs_Tempe		Boxes_Tem
start: 1238112000	21.2 > moon	Shelved (HVAC, DET, DLAB, TE 10) >	achines	DetOp_machi	nes olse	ervers	rtp	CS	CoilSwite	chBoxes	INF_devic	es E	NV_devices	VAC_de
y n. 7716 (tot. 10000)	21.2 > mean	20.8		CalWE	CalINJ	CalBS	С	alPR	PCalN	IE PC	CalWE	HOFT	NCAL	NoiseIr
	0	(alue = 20.9105)	Coffigero	A1	Tempera	turgeAl		Inject	lionAl		line/	M.		TeeAl

9.10 DMS archive WUI



10How-to

10.1 How-to for the Moni processes

The operation described below can be performed only by expert users. In case of doubt please contact the operation team.

10.1.1 How to configure a generic channel in the Moni process

- Check that the channel is available in the DAQ
 - To do this you can use Datadisplay and connect to online provider to search for the desired channel; if present in the list it can be defined.
- Edit the Moni configuration file
 - From VPM open the configuration file of the Moni process
 - Set the FDIN_TAG in the header section; you have to specific the prefix of the channel: V1_ENV, V1_HVAC, ...
 - \circ $\;$ Define the channel; the generic syntax of the row is the following:

keyword	ITF_LOCK_state	Flag	mathematic	comment	mathematic	comment
		name	al function	associated	al function	associated
			of the	for the red	of the	for the
			channel and	flag	channel and	yellow flag
			thresholds		thresholds	, ,
			for red flag		for vellow	
			, and the set of the s		flag	
					inda	

QC_MONITOR * TubePumpsNorth "0<VAC_TUBE600N..PR1<0.3" "600 N Turbo could have a problem" "*<0.1" "TBD"

QC_MONITOR * TubePumpsWest "VAC_TUBE600W..PR2<1e-7" "600 W Turbo could have a problem" "*<2e-8" "TBD"

• Stop and restart the process

10.1.2 How to configure a channel computed with the mean() mathematical function

QC_MONITOR * TCS_zone "18<mean(ENV..TCS_CHILROOM_TE,30)<28.0" "Temperature in TCS Chiller Room is too High/Low" "*<26.0" "Temperature in TCS Chiller Room is quite High"

10.1.3 How to configure a channel computed with the rms() mathematical function

QC_MONITOR * DetectionLab "rms(ENV_EDB_MIC,10)<0.1" "Microphone on External Detection Bench is too high"

10.1.4 How to configure a channel computed with the brms() mathematical function

QC_MONITOR * MC_Hall "brms(ENV_MC_ACC_Z,1,10,3,500)<0.004" "Accelerometer at MC tower is too High, brms(0.1Hz to 1000Hz)"

10.1.5 How to configure a channel computed with the delta() mathematical function

QC MONITOR * EnvServers "delta(ENV..EAB1 TE,1800)>1e-08" "EnvServer frozen"

10.1.6 How to configure a channel computed with the exist() mathematical function

QC MONITOR * ExistChannel "exist(HVAC..CEB HOT CORR)>0" "channel not aquired"

10.1.7 How to configure a channel with thresholds depending by the value of ITF_LOCK_STATE

QC_MONITOR *to124.5 B2_8MHz_DPHI "-100<mean(LSC_B2_8MHz_DPHI,60)<100" "nn" QC_MONITOR 125to* B2_8MHz_DPHI "-0.4<mean(LSC_B2_8MHz_DPHI,60)<0.4""B2_8MHz phase mistuned" "-0.2<*<0.2" "B2_8MHz phase mistuned"

QC_MONITOR *to149 B1p_DC_rms"rms(LSC_B1p_DC,30)>0""B1p_DC is fluctuating too much" QC_MONITOR 149.5to* B1p_DC_rms "rms(LSC_B1p_DC,30)<0.003" "B1p_DC is fluctuating too much "rms(LSC_B1p_DC,30)<0.0015" "B1p_DC is fluctuating"

QC_MONITOR *to40.5 ExtPLL "SQZ_PLL_Ext_Status>-1" "ExtPLL unlocked!" QC_MONITOR 41to* ExtPLL "SQZ_PLL_Ext_Status>1.5" "ExtPLL unlocked!"

10.2 How to for the ComputingDMS process

The operation described below can be performed only by expert users. In case of doubt please contact the operation team.

To add a channel in this process you must edit the configuration file and then stop and restart the process from VPM.

10.2.1 How to configure a channel computed with the ping()function

CF ping(olserver38) olserver38 olserver38.virgo.infn.it None ping None 0 0

10.2.2 How to configure a channel computed with the cpu_user()function

CF cpu user(olserver38) olserver38 olserver38.virgo.infn.it New Olservers cpu user max 70 90

10.2.3 How to configure a channel computed with the load()function

CF load(olserver137) olserver137 olserver137.virgo.infn.it DMS load 1-min max 7 10

10.2.4 How to configure a channel computed with the mem_use ()function

CF mem use(olserver137) olserver137 olserver137.virgo.infn.it DMS mem use max 10e9 11e9

10.2.5 How to configure a channel computed with the mem_swap()function

CF mem_swap(olserver137) olserver137 olserver137.virgo.infn.it DMS mem_swapmax1e9 2e9

10.3 How-to for DMSserver

The operation described below can be performed only by expert users. In case of doubt please contact the operation team.

10.3.1 How to stop and start the DMSserver



The DMSserver can be stopped and restarted from the VPM; the application name is: DMSserver.

10.3.2 How to configure a provider

- Check that the provider you are interested in is writing a JSON file;
- Get the path of the JSON file;
- Edit the DMS configuration file;

PROVIDER /opt/MonitoringWeb/buffer_dms/json_qcmoni/QcInfrastructuresAlarmedData.jsonInfraAlarmedMoni 0None 0 None00

Stop and restart the server

10.3.2.1 How to configure a provider with "delay_before_unavailable_s"

PROVIDER /opt/MonitoringWeb/buffer_dms/json_qcmoni/QcInfrastructuresAlarmedData.jsonInfraAlarmedMoni 60None 0 None00

10.3.2.2 How to configure a provider with email notification

PROVIDER /opt/MonitoringWeb/buffer_dms/json_qcmoni/QcInfrastructuresAlarmedData.jsonInfraAlarmedMoni 600PERATION MAIL600 None00

ROVIDER /opt/MonitoringWeb/buffer_dms/json_qcmoni/QcSuspensionsData.jsonSuspMoni120 OPERATION MAIL,SUSP MAIL 600 None 0 00

10.3.2.3 How to configure a provider with SMS notification

PROVIDER /opt/MonitoringWeb/buffer_dms/json_qcmoni/QcInfrastructuresAlarmedData.jsonInfraAlarmedMoni 600PERATION MAIL600 OPERATION SMS6000

PROVIDER /opt/MonitoringWeb/buffer_dms/json_qcmoni/QcSuspensionsData.json 120 OPERATION MAIL,SUSP MAIL 600 OPERATION SMS,SUSP SMS 600 0 SuspMoni

10.3.2.4 How to configure a provider with SOUND notification

PROVIDER /opt/MonitoringWeb/buffer_dms/json_qcmoni/QcInfrastructuresAlarmedData.jsonInfraAlarmedMoni 600PERATION MAIL600 OPERATION SMS60060

PROVIDER /opt/MonitoringWeb/buffer_dms/json_qcmoni/QcSuspensionsData.json 120 OPERATION MAIL,SUSP MAIL 600 OPERATION SMS,SUSP SMS 600 0

SuspMoni

10.3.3 How to configure an alert recipient

The operation described below can be performed only by expert users. In case of doubt please contact the operation team.

10.3.3.1 How to configure an alert recipient for email notification

ALERT_RECIPIENT OPERATION_MAILuser1@ego-gw.it

ALERT_RECIPIENT SUSP_MAILuser2@ego-gw.it, user3@ego-gw.it

10.3.3.2 How to configure an alert recipient for sms notification

ALERT RECIPIENT SUSPENSIONS SMS 393381212345,338585987

ALERT RECIPIENT OPERATION SMS 393383535265

10.3.4 How to test an alert recipient

The test can be done by clicking on a dedicated buttons on the VPM after authentication. The buttons are in the DMSServer section.

DMSserver	GPS: 1238169771	.16 - provders: 23/30 [25] - flags: 355/557 [400]	C
Appli	cation: /virgoApp/DMSserve	r/v1r8/src/DMSserver.py	
	Host: olserver137.virgo.in	fn.it Pid: 51278 Process uptime: 1h15m11s	
Configu	ration: /virgoData/VirgoOnli	ne/DMSserver.cfg	
1	C Log: /virgoLog/VirgoOnlin	e/DMSserver/DMSserver_2019-04-01-14h47m22-UTC.log	
DMS TDS docum	nent https://tds.virgo-gw.e	eu/ql/?c=12364	
	Test recipient mail	Test recipient SMS	

Figure 47 – DMSserver application in VPM, testing alert recipient

10.3.4.1 How to test an email recipient

- Click on the button "Test recipient mail";
- In the modal window write the value of the recipient you are interested in to test in the input text;
- Click the button "Send Command and Close Window";

	Virgo Online	Search	QF Edit + Tools + Help + User berni - Logout
Sqzilloni			ate=V1:META_ITF_LOCKindex=170.00 ()
SuspMoni	0 1238169974-1 latency 3.59 nAdc=2278 nSms=8; (15.5 kB to_FbmQc outTime: 0.003	0) inTime 0.0891 outTime 0.0172 - Oc Suspension	s=1_itfState=V1:META_ITF_LOCKindex=170.00 (red flags) SATSR_satserversr1_State
SuspEBMont	0 1238169974-1 latency 3.45 nAdc=3598 nSms=26; (8.8 kB to_FbmQc outT DMS	server / Test recipient mail	<pre>X TA_ITF_LOCKindex=170.80 ()</pre>
SuspShortMoni	0 1238169974-1 latency 3.36 nAdc=392 nSms=4; (9.5 kB to_FbmQc outTime		ETA_ITF_LOCKindex=170.00 ()
TC SMoni	1238169974-1 latency 3.33 nAdc=45 nSms=6; (7.1 kB to_FbmQc outTime:	Recipient OPERATION_MAIL	dex=170.00 ()
VacuumMoni	0 1238169973-1 latency 3.27 nSms=58; (12.8 kB to_FbmQc outTime: 0.00		ex=-9999.00 (red flags) VAC1500N1_Temp1 VAC1500N1_Temp1 VAC1500N1_Temp2 VAC. 🔳
DM collector -			
FbmQc	0 1238169967-1 latency 10.33 nAdc=10963 nSms=791 nProc=5; All 18 fr Cloperation	se Window Send Command Send Command	and Close Window B inTime 0.0643 outTime 0.0132
QcToSt	▲ 1238169967-1 latency 10.44 nAdc=32 nSms=44 nProc=5; (130.7 kB to_Pomot outra	HE: 0.0004) INTINE 0.0005 OUTLINE 0.0005	
DMS -			
DIMSserver			- gps_snapshot: 1238169961 - conflicts: 4 [30] - snapshot e.t.: 4 - snapshot s.t.: 6
Annalo	Figure 48 – DMSserver	application in V	PM, testing alert recipient.

10.3.4.2 How to test an SMS recipient

- Click on the button "Test recipient SMS";
- In the modal window write the value of the recipient you are interested in to test in the input text;
- Click the button "Send Command and Close Window";



Figure 49 – DMSserver application in VPM, testing alert recipient.

10.3.5 How to configure a flag in the DMSserver

The operation described below can be performed only by expert users. In case of doubt please contact the operation team.

- Check that the flag and its condition flags are available in the JSON payload generated by one DMS provider; this operation can also done by looking at the status of the related Moni server in the VPM checking the "lost" signals.
- Check that the DMS provider is properly configured in the DMS configuration file;
- Edit the DMS configuration file;

FLAG NI Guard Trigger Suspensions 2 None 0 None 0 None 0 0

Stop and restart the server

10.3.5.1 How to configure a flag in the DMSserver with "group"

FLAG NI_Guard_Trigger Suspensions 2 NI_Guard 0 None 0 None 0 0

10.3.5.2 How to configure a flag in the DMSserver with "delay_before_visualization"

FLAG NI_Guard_Trigger Suspensions 2 NI_Guard30 None 0 None 0 0

10.3.5.3 How to configure a flag in the DMSserver with email notification

FLAG NI Guard Trigger Suspensions 2 NI Guard30 OPERATION MAIL, SUSP MAIL600 None 0 0

FLAG NI IP Suspensions 2 None 0SUSP MAIL500 None 0

10.3.5.4 How to configure a flag in the DMSserver with sms notification

FLAG NI_Guard_Trigger Suspensions 2 NI_Guard30 OPERATION_MAIL,SUSP_MAIL600 OPERATION_SMS 900 0

FLAG NI_IP Suspensions 2 None 0None0SUSP_SMS600 (

FLAG NI_F7 Suspensions 2 None 0None0None01800

10.4 How-to for the homepage

10.4.1 How to open the DMS homepage

The link to open the DMS homepage is the following:

https://dms.virgo-gw.eu/

it can be open only from the internal network (or after firewall authentication).

10.4.2 How to open the individual-flag-information section

To open the information of the individual flag you have to click on the flag you are interested in at the level of the homepage.

10.4.3 How-to for Associated Condition Flag plots

10.4.3.1 How to open the associated condition flag plot

- Open the individual-flag-information section for the desired flag
- Click on the icon

If you want to open a dedicated window containing the plot you can click inside box containing the plot just open.



10.4.3.2 How to switch from up-sampled/down-sampled data

• Open the dedicated window containing the plot.

When you open the associated condition flag plot the default visualization is "up-sampled data" (30 minutes of data sampled every 10s).

To switch from "up-sampled data" to "down-sampled data" click on the icon \heartsuit To switch from "down-sampled data" to "up-sampled data" click on the icon \circlearrowright



Figure 51 – how to switch from up-sampled/down-sampled data

10.4.3.3 How to zoom, un-zoom and pam a plot

The plot can be zoomed in both axis: X, Y.

• Open the dedicated window containing the plot.

To zoom the plot:

- click the left button of the mouse
- drag the mouse
- release the left button of the mouse

To un-zoom the plot:

• double-click of the left button of the mouse.



Figure 52 – how to zoom, un-zoom and pam a plot

To start pamming the plot:

- keep pressed shift+ctrl+left mouse button
- move the mouse

10.4.3.4 How to pause-play a plot

• Open the dedicated window containing the plot.

To pause a plot:

- Click on the icon
 - Click on the icon



10.4.3.5 How to download a plot as png

• Open the dedicated window containing the plot.

• Click on the icon 🛓

10.4.4 How to log into-out the system

To log into the system:

- Click on the icon not the top right side of the homepage to open a modal window to insert the user detail
- Enter your user details
- Click on the button Sign-in

# DMS				ITF Mode: :	Science (1d 14h 17m 2)	() ITF State	B: LOW_NOIS	E_3_SQZ (18 14	h 21m 36s) UTC: 20	19-05-03 08:32:09		
	SIB1_IP	SIB1_B	ENCH	SIB1_BR								
Trainetion	MC_IP									MC F		
Injection	Laser			asenChiller						-C 0		
	MC_Power	DMS sign	-in				×		BPC	BPC_Electr		
A CONTRACTOR OF	PD	Divis sign								Shutter		
Detection	SDB1_Tr	Enter your up	v dotaile:					SDP _uard		SDB1_Electr		
100	B2_8MHz_FPHI	Linter your use	a details.					Fit		GIPC		
150	Blp	Usemame						LSC_rms		ASC_rms		
	BS_IP							TE		BS_Electr		
	NI_IP	Concerned and						TE		NI_Electr		
	NE_IP	Password						TE		NE_Electr		
Suspensions	PR_IP							TE		PR_Electr		
	SR_IP							TE		SR_Electr		
	WI_IP	Sign-in						TE		WI_Electr		
	WE_IF	WE_F7	WE_PA	W/	VE_BR	WE_Vert	WE	TE		WE_Electr		
Environment	CB_Hall								Seismon	BRMSMon		
Environment	INJ_Area								SeaActivity	WAB		
Infractructures						ACS	LINI]	ACS_DET		ACS_WAB		
Innastructures	UPS_TB	UPS_CB	UPS_MC	UPS_NE	UPS_WE Fli	atChannel				ACS_COB		
	EIB_SBE	SDB2_SBE		SNEB_SBE	SNEB_LC	SWE	B_SBE		SPRB_SBE	SPRB_LC		
								02_Laser		Chillers		
sqz										Rack_TE		
										VAC_SERVOS		
vacuum										HLS		
1000	DetectorSEnvironme		Minitowers	ISC						Metatron		
VPM	DetectorMonitor	ing D							n	DetChar		
										Daq_Boxes_TE		
DAQ-computing										VAC_devices		
	CalNE			CalBS	CalPR	PCaINE	PCalWE	HOFT		NoiseInjection		
ITFOnCall	SoftwareAl	Tei	mperaturesAl	Injecti	onAl			Generator		TcsAl		
DetChar		frec_RANGE_BNS			flag_AlertGrac	eDB		STATE_VECTOR				
			1 hou		into ou	+ + h -						

Figure 54 - how to log into-out the system

To know if the user is logged:

• Pass the mouse over the icons ➡ , ➡ on the top right side of the screen To log out the system:

• Click the icon 🗭 on the top right side of the screen

10.4.5 How to open the dashboard

• Click on the icon 🕮 on the right side of the homepage;

10.4.6 How to open the alerts section

• Click on the icon 🕕 on the right side of the homepage

10.4.7 How to open the shelving section

10.4.8 How to open the muting section

• Click on the icon **I**on the right side of the homepage

10.4.9 How to shelve-unshleve a condition flag

• Log into the system

To shelve a condition flag:

- Open the individual-flag-information section for the desired flag;
- Click on the icon be to open the shelving section;
- Fill the form;
- Click on the button Shelve



To un-shelve a condition flag:

- Click on the icon ${\mathscr D}$ on the right side of the homepage to open the shelving section;
- Search the condition flag you want to un-shelve;
- Click on the button Un-shelve.

& Shelving ×	^			1	TF Mode: Ad	ijusting (od of	n 6m 49s)	ITF State:	LOW_NOIS	1 33m 26s) UTC	: 2019-05-03 11:20	
12 Sherving	SIB1_IP	SIB1_I	BENCH	SIB1_	BR	SIB1_V	Vert	-	SIB1_TE	SIB1	_Guard	SIB1_Electr
Currently shelved:	MC_IP	MC_	PAY	MC_E	3R.	MC_V	ert		MC_TE	MC	Guard	MC_Electr
,	Laser	Laser	Ampli	LaserCh	hiller	SL_TempCo	ontroller		RFC		NFS	PC
27/11/18 15:46 - Chillers	MC_Power	PST	TAB	IMC_	AA	IMC_AA	GALVO	N	4C_F0_z		ВРС	BPC_Electr
mean_TCSRWI_Chiller_Alarm_10	PD	QPD	B1p	QPD	B2	QPD_	B5		OMC	Pico	Disable	Shutter
	SDR1 IP	SDR	LIC .	SDR1	RP	SDR1	Vort	-	DR1 TE	SDB	Guard	SDR1 Flectr
26/03/19 12:21 - O2_NE		B4_56MI		DARM	UGF	UNLO	CK .	5	SFS_UGF	En Fu	noaerr	GIPC
Un-shelve	B1p_DC	B4	_112MHz_MAG	6	B7_DC			B8_DC		LSC_rms		ASC_rms
U	BS_IP	BS_F7	E	S_PAY	BS	BR	BS	Vert	BS_	TE I	BS_Guard	BS_Electr
24/04/17 2 12 - WI_CO2_Laser TCS_CO2EL2_Un-shelve	NI_IP	NI_F7	1	NI_PAY	NI	BR	NI_	Vert	NI_T	re	NI_Guard	NI_Electr
	NE_IP	NE_F7		IE_PAY	NE	_BR	NE_	Vert	NE_		NE_Guard	NE_Electr
18/06/18 19:50 - NE_TE	PR_IP	PR_F7	F	R_PAY	PR	_BR	PR_	Vert	PR_1	re	PR_Guard	PR_Electr
mean_envne_r2_re1_ou onsneve	SR_IP	SR_F7	SR_F7 S		SR	_BR	SR	Vert	SR_TE		SR_Guard	SR_Electr
09/04/19 16:45 - ACS_INJ_ALARMED	WI_IP	WI_F7	V	WI_PAY W		_BR	WI_	WI_Vert		TE	WI_Guard	WI_Electr
ean_HVACINJ_FLUX_OUT_10 Un-shelve	WE_IP	WE_F7	v	VE_PAY	WE	_BR	WE,	Vert	WE_	TE	WE_Guard	WE_Electr
	CB_Hall	MC_Hall	TC	S_zones	NE	Hall	WE,	Hall	WindAc	tivity	Seismon	BRMSMon
09/04/19 16:25 - ACS_INJ_SAS	INJ_Area	DET_Area	EE_Roor	Room DAQ_R		Extern	nal	DeadCha	annel	Lights	SeaActivit	y WAB
Un-shelve	CS_CB_Hall	ACS_TB	ACS_DAQ_F	Room ACS	S_EE_Room	ACS_I	МС	ACS_I	LNJ	ACS_DET	ACS_NE	ACS_WA
	UPS_TB	UPS_CB	UPS_MC	UPS_N	E U	PS_WE	FlatCha	nnel E	xistChannel	ACS_WE	ACS_CE	_CR ACS_CO
09/04/19 16:05 - ACS_WE_ALARMED mean INF WEB CHILLER TE OUT 60	EIB_SBE	SDB2_SBE	SDB2_L	c s	NEB_SBE	SNEB		SWEB_S	SBE	SWEB_LC	SPRB_SB	SPRB_LC
Un-shelve	NE_RH		WE	_RH		NI_CO2_	Laser		WI_CC	02_Laser		Chillers
09/02/19 22:49 - DET Lab Dust	PLL	Sque	ezer	SQZ_	AA	SQZ_Sh	utter	Co	ohe_CTRL	SC	2_Inj	Rack_TE
mean_ENV_DETRDUST_0P3UM_30	LargeValves	Clean_Air	Tub	eStations	Tube	Pumps	MiniT	owers	TurboL	inks	RemDryPMP	VAC_SERVO
Un-shelve	Pressure	Compressed	Air Tow	erServers	Tower	Pumps	Cryo	oTrap	O2_Ser	nsors	Tank	HLS
09/03/19 22:49 - DET_Lab_Dust	ctorSEnvironme	ControlRoom	Minitowe	rs	ISC	Injecti	ion	TCS		Suspension	Vacuum	Metatron
mean_ENV_DETRDUST_0P5UM_30	DetectorMonitor	ing C	DataCollection		Storage		D	ataAccess		Automatio	n	DetChar
OTESTICAVE	Latency	Disk		Timing	Timin	g_rtpc	Timin	ig_dsp	Fast_C	DAC	ADCs_TE	Daq_Boxes_
09/03/19 22:49 - DET_Lab_Dust	MS_machines	DetOp_machi	nes ol	servers	rt;	pcs	CoilSwit	tchBoxes	INF_de	vices	ENV_devices	VAC_device
mean_ENV_DETRDUST_0P7UM_30	CalNE	CalWE	CalINJ	CalBS		CalPR	PCall	IE I	PCalWE	HOFT	NCA	L NoiseInjec
	SoftwareAl	T	emperaturesAl		Injection	A		UpsAl		Generator	Al I	TcsAl
09/03/19 22:49 - DET_Lab_Dust		Hrec RANGE BNS			flag AlertGraceDB						STATE VECT	OR SOL

Figure 56 – how to shelve-unshleve a condition flag

Another dedicated way to unshelve a condition-flag is to use the "DMS currently shelved condition flags", see <u>How to for the DMS currently shelved condition-flags</u>.

10.4.10 How to mute-unmute a condition flag

• Log into the system

To mute a condition flag:

• Open the individual-flag-information section for the desired flag;

To be noted that only alarmed flag can be muted.

• Click on the icon d to open the mute section;

- Fill the form;
- Click on the button Mute



Figure 57 – how to mute-unmute a condition flag

To un-mute a condition flag:

- Click on the icon some the right side of the homepage to open the muting section;
- Search the condition flag you want to un-mute;
- Click on the button Un-mute.

Muting ×	<u>^</u>				ITF Mode: S	Science (od a	h 56m 1s)	ITF State:	LOW_NOIS	E_3_SQZ (0d			19-05-04 14:15			
		STR1	BENCH	SIB1	BR	SIB1_	Vert		SIB1_TE	SI	B1_Guard		SIB1_Electr	R		
Currently muted:	MC_IP			-		MC_V	/ert		MC_TE	M	C_Guard		MC_Electr			
contendy mateur	Laser	Laser	Ampli	LaserC	niller				RFC		LNFS		PC			
18/05/18 23:56 -	MC_Power	PST	AB	IMC_	AA	IMC_AA	GALVO				200		BPC_Electr			
ACS_CB_HALL_ALARMED	PD	OPD	B1p	OPD B		QPD	B5		OMC	Pi	coDisable		and the second second	7		
Un-mute	SDB1_IP	SDB		SDB1	BR	SDB1	Vert		SDB1_TE	SD	B1_Guard		SDB1_Electr			
	B2 8MHz DPHI	B4 56M	HZ DPHI	DARM	UGF	UNLO	ОСК		SFS UGF		modErr		GIPC			
8/05/18 23:56 -	B1p DC	B4	112MHz MAG		B7 DC			B8 DC		LSC rm	าร		ASC rms			
nean_HVACCEB_COLD_CORR_60	BS IP	BS F7	BS	PAY	BS	BR	BS	Vert	/ert BS TE		BS Gu	ard	BS Electr			
Un-mute	NI IP	NI F7	NI	PAY	NI	BR	NI	Vert	NI		NI Gu	ard	NI Electr			
2/06/17 11:51 -	NE IP	NE_F7	NE	PAY		BR	NE	Vert	NE	TE	NE Gu	ard	NE Electr			
etectorSEnvironmentMonitoring	PR IP	PR F7	PR	PR PAY		BR	PR	Vert	PR	TE	PR Gu	ard	PR Electr			
and ecountering of more	SR IP	SR F7		PAY		BR		Vert		TE	SR Gu	ard	SR Electr			
2/06/17 11:51 -	WI IP	WI F7	WI	PAY		WI BR		Vert		WI TE		WI Guard				
atticleCounterWE Un-mute	WE IP	WE F7	WE	PAY	WE	BR	WE	Vert	WE	те	WE Gu	ard	WE Electr			
	CB Hall	MC Hall	TCS	zones	NE	Hall	WE	Hall	WindAc	tivity	Seism	on	BRMSMon			
2/06/17 11:51 - DetectorMonitoring	INJ Area	DET Area	EE Room	C	AQ Room	Exte	rnal	DeadCha	annel	Lights	Sea	Activity	WAB			
	CS CB Hall	ACS TB	ACS DAQ Ro	om AC	S EE Room	ACS	MC	ACS I	CN:	ACS DET	AC	CS NE	ACS WAR			
2/06/17 11:51 - DetectorMonitoring	UPS TB	UPS CB	UPS MC	UPS N	IE UI	PS WE	FlatCha	nnel E	xistChannel	ACS W	E A	CS CB CR	ACS CO	3		
fraMoni Un-mute	EIB SBE	SDB2 SBE	SDB2 LC		NEB SBE	SNEB	LC	SWEB	SBE	SWEB LC	SPF	RB SBE	SPRB LC			
2/06/17 11:51 - DetectorMonitoring	NE RH		WE RH		NI CO		O2 Laser		WI CO2		2 Laser		Chillers			
ockMoni Un-mute	PLL	Saue	ezer	507 44		SOZ Shutter		Cohe CTRL		L SOZ			Rack TE			
2/06/17 11:51 - DetectorMonitoring	LargeValves	Clean Air	TubeS	tations	TubeF	Pumps	Mini	Towers	TurboL	inks	RemDry	PMP	VAC SERVO			
erversMoni Un-mute	Pressure	Compressed	Air Tower	Servers	Tower	Pumps	Crv	oTrap	O2 Ser	nsors	Tank		HLS			
2/06/17 11:51 - DetectorMonitoring	ctorSEnvironme	ControlRoom	Minitowers		ISC	Iniec	tion	TCS		Suspension	Va	acuum	Metatron			
uspEBMoni Un-mute	DetectorMonito	ring	DataCollection		Storage		C	ataAccess		Automat	ion		DetChar			
	Latency	Disk	Tin	nina	Timin	a rtpc	Timir	na dsp	Fast (DAC	ADCs	TE	Dag Boxes T	Е		
2/11/18 21:41 - ACS_COB ean_HVACCOB_COLD_TE_60	MS machines	DetOp machi	nes olse	rvers	rte	ocs	CoilSwi	tchBoxes	INF de	vices	ENV de	vices	VAC devices			
Jn-mute	CalNE	CalWE	CalINJ	CalBS	5	CalPR	PCall	NE	PCalWE	HOFT		NCAL	NoiseIniec	io		
2/11/18 21-28 - Sez 1500W	SoftwareAl	Т	emperaturesAl		Injection	Al		UpsAl		Generato	orAl		TcsAl			
qz_ENVSQZ_BENCH_TE1 Un-mute		Hrec RANGE BNS					GraceDB				STATE	VECTOR	ASC_rms BS_Electr NI_Electr NE_Electr PR_Electr WI_Electr WI_Electr BRNSMon vity WAB iE ACS_WAB GG_CR ACS_COB BRNSMon OG_CR ACS_COB RASK_TE VAC_SERVOS HLS Metatron DetChar Daq_Boxes_TE VAC_devices CAL NoiseInjection TcsAl			
	*													-		

Figure 58 – how to mute-unmute a condition flag

10.5 How-to for DMS event monitor

10.5.1 How to open the DMS event monitor

The DMS event monitor can be open in two ways:

- By clicking on the link Go to Event Monitor from the dashboard;
- By clicking on the icon \square of the condition flag
- By clicking on Condition-Flag name in the table of the Events in the DMS Event monitor page.

The first option opens the homepage of the DMS event monitor with no filter criteria preconfigured.

The second and third options open the DMS event monitor with the events associated to the condition flag from which the application has been called.

10.5.2 How to know the last events in a specified time window

- Open the DMS event monitor from the dashboard
- Check all the events in the "Select event type" box inside the "Filter toolbar" section
- Check the option "Select all" in the "Select condition flag" box inside the "Filter toolbar" section
- Set the time window in the "Select time window" box inside the "Filter toolbar" section
- Click on the button Submit

	# DMS Event Monitor									
	Filter toolbar	e Events								
	Select event type									
	shelving	Filtering								
1	muting	Selected event type: shelvin Selected condition flag: All	g - muting - sms - mail -	sound -						
_	sms alert	Selected time window (yy-mm-dd, LT): from 2019-04-04 00:00:00 to 2019-05-04 24:00:00								
	The mail start	B 36 shelving events								
		⊕ 1 muting events ⊕ 121 sms events								
	a sound alert	⊕ 202 mail events ⊕ 174 sound events								
	Select condition flag	Date (LT)	Event	Condition-F	Flag	Provider				
	VetiCollect VetiMerper VetiMerper	2019-05-03 00:00:12	• moting	Particle	DetectorSEnvironmentMonitoring	VPM				
	VetoThr VetoThrCellect vic channel reader	2019-05-02 09:20:43	sbelving	ATER, TE_IN_60	ACS_WE_ALARMED	InfraAlarmedMoni				
	WARM_LOCK WEB_DBOX_TCS_PCAL_bining_error WEB_DBOX_TCS_PCAL_bining_error	2019-05-02 09:10:17	🖬 nul	WEB_HEATER_TE_IN_60	ACS_WE_ALARMED	InfraAlarmedMoni				
	WEB_PCAL_Fast WEB_PCAL_Fast WEB_PCAL_FAST_DAC_ssfs_flags	2019-05-02 09:10:17		INF_WEB_HEATER_TE_IN_60	ACS_WE_ALARMED	InfraAlarmedMoni				
1	WE_Dh_lat WE_Dh_lon _HLStuBeN_meanHLSvalve_meanDh_lon_0+offset_lon	2019-05-02 09:00:43		mean_INF_WEB_HEATER_TE_IN_60	ACS_WE_ALARMED	InfraAlarmedMoni				
	Spect all Deselect all	2019-05-02 06:51:03		mean_INFCOB_FARM_HOTAIR1_TE2_60	ACS_COB	InfraAlarmedMooi				
	Search and select condition flags	2019-05-01 19:54:06	A	mean_INFCOB_FARM_HOTAIR1_TE2_60	UNLOCK	LockMoni				
		2019-05-01 19:25	A word	META_ITF_LOCK_index	UNLOCK	LockMoni				
-	Select time winds	2019-05-	N sound	META ITE LOCKINDex		LockMoni				
2	From: 2019-04-04 00:00:00	7	A sound	META_ITF_LOCKindex		LockMoni				
-	To: 2019-05-04 24:00:00	x31:28	🔤 mul	mean_INFCOB_FARM_CDZ1_60	ACS_COB	InfraAlarmedMoni				
		J5-01 15:31:28		mean_INFCOB_FARM_CDZ1_60		InfraAlarmedMoni				
	SURMIT	2019-05-01 12:21:29	🔤 mul	mean_INFCOB_FARM_CDZ1_60	ACS_COB	InfraAlarmedMoni				
4	SUBMIT	2019-05-01 12:21:29		mean_INFCOB_FARM_CDZ1_60		InfraAlarmedMoni				
		2019-05-01 09:49:48	shelving	mean_INFWEB_HEATER_TE_IN_60	ACS_WE_ALARMED	InfraAlarmedMoni				
-		2019-05-01 09:10:16	🔤 mail	mean_INFWEB_HEATER_TE_IN_60	ACS_WE_ALARMED	InfraAlarmedMoni				

Figure 59 – how to know the last events in a specified time window

10.5.3 How to know a specific events for specific condition flags a specified time window

- Open the DMS event monitor from the dashboard
- Check only the events you are interested in the "Select event type" box inside the "Filter toolbar" section

- Select only the condition flags you are interested in the "Select condition flag" box inside the "Filter toolbar" section
- Set the time window in the "Select time window" box inside the "Filter toolbar" section
- Click on the button Submit

10.5.4 How to know details of the event

• Click on the icon of the event in column Events of the search result.

10.6 How-to for DMS playback

10.6.1 How to open the DMS playback

- By clicking on the icon \square on the right side of the DMS homepage.
- Coping the URLs provided by other API.

10.6.2 How to start-pause-stop the playback

Open the DMS playback from DMS homepage

To start the playback:

- Set "UTC start" on the "Filter toolbar" section
- Set "Snapshot rate" on the "Filter toolbar" section
- Set "Snapshot pause" on the "Filter toolbar" section
- Click on the icon

To pause the playback:

Click on the icon

- To restart the playback:
 - Click on the icon

To stop the playback:

Click on the icon

Filter toolbar	# DMS			ITF Mode: 1	Science	(0d 7h 36m 58s)	ITF St	ate: LOW_	NOISE_3_	SQZ (0d 7	h 51m 38s)	UTC: 2019	-05-01 00:00:1
		SIB1_IP	SIB1	BENCH	SI	B1_BR	SIB1	_Vert	SIB1	TE	SIB1	Guard	SIB1_Electr
UTC start		MC_IP	MC	PAY	M	IC_BR	MC	Vert	MC_	TE	MC_0	Guard	MC_Electr
2010 05 01 00 00 00	Injection	Laser	Lase	rAmpli	Lase	erChiller	SL_Temp	Controller	RFG		LN	IFS	PG
		MC_Power	PS	ТАВ	IM	1C_AA	IMC_AA	GALVO	MC_F)_z	B	PC	BPC_Electr
Snapshot rate	P. Hardberg	PD	QPE	D_B1p	QI	PD_B2	QPD	B5	OM	6	PicoD	isable	Shutter
	Detection	SDB1_IP	SDI	B1_LC	SD	B1_BR	SDB:	L_Vert	SDB1	TE	SDB1	Guard	SDB1_Electr
	700	B2_8MHz_DP	HI B4_56N	Hz_DPHI	DAF	RM_UGF	UNI	OCK	SSFS_	UGF	Fmo	dErr	GIPC
Snapshot pause [s]	ISC	B1p_DC	B4	_112MHz_I	MAG	B7_D	C	B	B_DC		LSC_rms		ASC_rms
		BS_IP	BS_F7	Z	BS_PAY	BS	S_BR	BS_V	ert	BS_TE		S_Guard	BS_Electr
		NI_IP	NI_F7		NI_PAY	NI	BR	NI_V	ert	NI_TE	1	NI_Guard	NI_Electr
-		NE_IP	NE_F7	2	NE_PAY	NE	E_BR	NE_V	ert	NE_TE	N	E_Guard	NE_Electr
-		PR_IP	PR F7		PR_PAY	PR	R_BR	PR_V	ert	PR_TE	F	R_Guard	PR_Electr
J on File			SR_F7	7	SR_PAY	SR	R_BR	SR_V	ert	SR_TE	9	R_Guard	SR_Electr
4		WI_IP	WI_F7	2	WI_PAY	W	I_BR	WI_V	ert	WI_TE	٧	VI_Guard	WI_Elect
Upload		WE_IP	WE_F	7	WE_PAY	W	E_BR	WE_V	'ert	WE_TE	V	/E_Guard	WE_Elect
fogla Nessun file selezionato.	A CONTRACTOR OF A	CB_Hall	MC_Ha	JI T	CS_zone	IS NE	Hall	WE_H	fall V	/indActivi	ity	Seismon	BRMSMor
l aurorecene	Environment	INJ_Area	DET_Area	EE_R	oom	DAQ_Room	Ext	ernal C	eadChanne	l Lig	ghts	SeaActivity	WAB
Download	Televative	ACS_CB_Hall	ACS_TB	ACS_DA	Q_Roon A	ACS_EE_Rool	m ACS	_MC	ACS_INJ	ACS	DET	ACS_NE	ACS_WA
40704035.json	Infrastructures	UPS_TB	UPS_CB	UPS_MC	UP	S_NE U	PS_WE	FlatChan	nel ExistCl	nannel	ACS_WE	ACS_CB_	CR ACS_CO
C BARANA A	SBE	EIB_SBE	SDB2_SBE	SDB2	LC	SNEB_SBE	SNE	B_LC	SWEB_SBE	SWE	B_LC	SPRB_SBE	SPRB_L
Display	TCS	NE_R	н	. V	/E_RH		NI_CO	2_Laser		NI_CO2_	Laser		Chillers
Constant	SQZ	PLL	Squ	ieezer	SC	QZ_AA	SQZ	Shutter	Cohe_0	TRL	SQZ	_Inj	Rack_TE
Snapsnot	Contraction of the Contraction o	LargeValves	Clean_/	Air Tu	beStatio	ns Tube	Pumps	MiniTov	wers	TurboLink	s Re	mDryPMP	VAC_SERVO
Info and debug	Vacuum	Pressure	Compr	essedAir	Towe	erServers	Tower	Pumps	CryoT	rap	02_S	ensors	Tank
		DetectorSEnviro	ControlRoon	n Minito	wers	ISC	Inje	ction	TCS	Susp	ension	Vacuum	Metatro
Log	VPM	DetectorMonit	oring E	oataCollecti	on	Stora	ge	Data	Access	A	utomatio	n	DetChar
	Data Computing	Latency	Disk		Timing	Timir	ng_rtpc	Timing	_dsp	Fast_DAG		ADCs_TE	Daq_Boxes_
start) 1240704000	DAQ-Computing	DMS_machines	DetOp_mag	machines olservers rt		tpcs CollSwitchBox		hBoxes I	xes INF_devices		IV_devices	VAC_device	
Ive: 1240674782.zip	Calib_Hrec	CalNE	CalWE	CalINJ	Ca	alBS	CalPR	PCalNE	PCa	WE	HOFT	NCAL	NoiseInjeg

Figure 60 – how to start-pause-stop the playback

10.6.3 How to download the snapshot JSON payload

- Stop or pause the playback
- Click the icon 📩 in the "Json File" box on the left section

10.6.4 How to upload the snapshot JSON payload

- Open the DMS playback from DMS homepage
- Click the icon Lin the "Json File" box on the left section

10.6.5 How to switch the display

- To switch from "Snapshot" to "info and debug" click the icon fin the "Display" box on the left section
- To switch from "info and debug" to "Snapshot" click the icon Oin the "Display" box on the left section

10.7 How-to for DMS archive

10.7.1 How to open the DMS archive

- Open the individual-flag-information section for the desired condition-flag;
- Click on the icon to open the DSM archive for desired condition-flag

10.7.2 How to edit summary plots

- Open DSM archive for desired condition-flag
- Click on the icon 🥸 on the top left side of the page
- Set the "Period"
- Set the "Y scale"
- Set the "Sampling"
- Click on the button SUBMIT



Figure 61 – how to edit summary plots

10.7.3 How to edit custom plots

- Open DSM archive for desired condition-flag
- Click on the icon State on the bottom left side of the page
- Set the "Plot title"
- Set the "Y scale"
- Set the "Sampling"

- Set the "time window"
- Set the "plot dimension"
- Click on the button SUBMIT



Figure 62 – how to edit custom plots

10.8 How to for the DMS currently shelved condition-flags.

10.8.1 How to open the DMS currently shelved conditionflags.

The DMS event monitor can be open by clicking on the link Go to shelved flags checklist from the dashboard.

10.8.2 How to order the results

The results can be order by:

- "Shelved by" ASC / DESC;
- "Date shelving" ASC / DESC;
- "Flag" ASC / DESC;
- "Condition-flag" ASC / DESC;
- "UTC start" ASC / DESC;
- "UTC stop" ASC / DESC;

This can be achieved by clicking on the related field on top of the table.

DMS Currently Shelved Flags User Interface											
Shelved by	Date shelving ¥	Flag	Condition flag	UTC start	UTC stop	Comment	Checked	Additional note	Unshelve butto		
pasqualetti	14/04/20 15:16	LargeValves	VAC_CRYOWEVCRYOCLOSEDRELAYST	14/04/20 09:20	28/04/20 09:30	stdby	•	none	Un-shelve		
pasqualetti	14/04/20 15:16	LargeValves	VAC_CRYONEVCRYOOPENRELAYST	14/04/20 09:30	28/04/20 09:30	stdby		none	Un-shelve		
pasqualetti	14/04/20 15:16	LargeValves	VAC_CRYONIVCRYOOPENRELAYST	14/04/20 09:30	28/04/20 09:30	stdby	•	none	Un-shelve		
pasqualetti	14/04/20 15:16	LargeValves	VAC_CRYOWI_VCRYOOPENRELAYST	14/04/20 09:30	28/04/20 09:30	stdby	•	none	Un-shelve		
masserot	11/04/20 06:01	Latency	DAQ_FFLRAW_LATENCY	01/04/20 00:00	31/05/20 23:55	АМ		none	Un-shelve		
berni	10/04/20 10:09	WE_Vert	mean_Sa_WE_F7_LVDT_V_50Hz_10	10/04/20 08:00	31/12/20 08:00	ITF securing from March 31st onward for O3 suspension		none	Un-shelve		
berni	10/04/20 10:09	WI_Vert	mean_Sa_WI_F7_LVDT_V_50Hz_10	10/04/20 08:00	31/12/20 08:00	ITF securing from March 31st onward for O3 suspension	•	none	Un-shelve		

10.8.3 How to log into the system

To log into the system:

- Click on the icon + Ion the top right side of the homepage to open a modal window to insert the user detail
- Enter your user details
- Click on the button Sign-in
- To know if the user is logged:
- Pass the mouse over the icons \Rightarrow , \Rightarrow on the top right side of the screen To log out the system:
 - Click the icon 🕩 on the top right side of the screen

10.8.4 How to view the list of the reports

Click on the icon on the right side of the homepage

10.8.5 How to build the report

- Log into the system
- Click on the icon 🗹 on the right side of the homepage

10.8.6 How to un-shelve a flag

• Log into the system

Un-shelve

Click on the icon

on the related condition flag

11Documentation

- DMS Snapshot of the system
- <u>DMS Presentation at Detchar meeting</u>
- <u>New Detector Monitoring System (DMS) Software Project</u>
- <u>New Detector Monitor System (DMS) User Requirements</u>
- Meeting DAQ/Automation/DMS