

Table of Contents

Help page of the Site Status Board.....	1
Information for the end user.....	1
Basic concepts.....	1
Navigation tools.....	2
Information for the metric providers.....	5
Adding new metrics:.....	5
Modifying existing metrics.....	7
Creating virtual metric.....	8
Creating views.....	8
Notification when a site changes color in particular view.....	13
Information for the administrators.....	14
Usage of the new dashboard.authorise module in SSB.....	15
Items management.....	15
Integration steps.....	16
Before first step.....	16
First step.....	19
Second step.....	19
Metric data modification from UI.....	21
How to understand which plugin updates given metric.....	22
Profiles.....	24
General help page on Dashboard.....	24

Help page of the Site Status Board

Information for the end user

Basic concepts

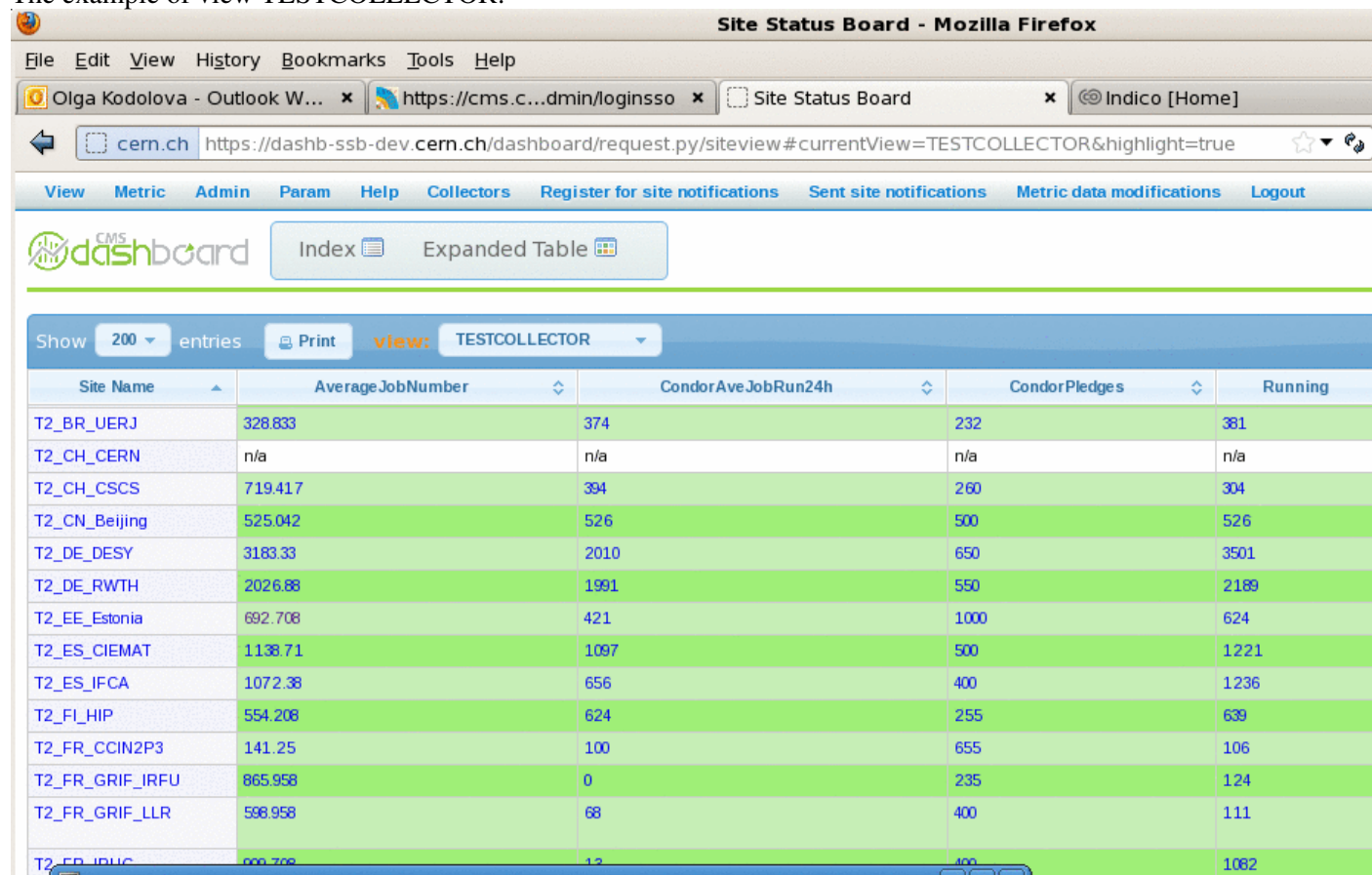
The Site Status Board is a monitoring tool that describes the status of Sites. The status is based on **metrics**. A **metric** is anything that can be measured: number of jobs, number of successful transfers, status of services, declaration of downtimes, etc.. The metrics are fetched regularly, and they give values to all the sites that are being monitored.

A metric can define quite a lot of information. Only the first two attributes are required:

- **Metric:** the metric that it is being gathered
- **Site:** name of the site
- **Validity:** Defined by 'start time' and 'end time'.
- **Color:** (red there are problems, yellow for warning, green for everything ok)
- **Label:** Free style text
- **Value:** Number for th
- **URL :** pointer to more information

Metrics can be combined into **views**. A view is a set of metrics that apply to the same sites. For instance, there could be a view called 'Job Monitoring' that has metrics like the 'Number of running jobs', 'Number of successful jobs', 'Status of the CE', etc. Usually, the SSB administrators define the views, and the users just see the information.

The example of view TESTCOLLECTOR:



The screenshot shows the Site Status Board interface in Mozilla Firefox. The browser address bar displays the URL: <https://dashb-ssb-dev.cern.ch/dashboard/request.py/siteview#currentView=TESTCOLLECTOR&highlight=true>. The interface includes a navigation bar with links: View, Metric, Admin, Param, Help, Collectors, Register for site notifications, Sent site notifications, Metric data modifications, and Logout. Below the navigation bar, there is a 'dashb' logo and a toggle for 'Index' and 'Expanded Table'. The main content area shows a table with 5 columns: Site Name, AverageJobNumber, CondorAveJobRun24h, CondorPledges, and Running. The table displays data for various sites, with the first row highlighted in green. The 'view' dropdown is set to 'TESTCOLLECTOR'.

Site Name	AverageJobNumber	CondorAveJobRun24h	CondorPledges	Running
T2_BR_UERJ	328.833	374	232	381
T2_CH_CERN	n/a	n/a	n/a	n/a
T2_CH_CSCS	719.417	394	260	304
T2_CN_Beijing	525.042	526	500	526
T2_DE_DESY	3183.33	2010	650	3501
T2_DE_RWTH	2026.88	1991	550	2189
T2_EE_Estonia	692.708	421	1000	624
T2_ES_CIEMAT	1138.71	1097	500	1221
T2_ES_IFCA	1072.38	656	400	1236
T2_FI_HIP	554.208	624	255	639
T2_FR_CCIN2P3	141.25	100	655	106
T2_FR_GRIF_IRFU	865.958	0	235	124
T2_FR_GRIF_LLR	598.958	68	400	111
T2_FR_IPHC	699.708	12	400	1082

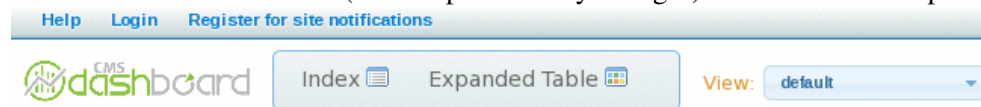
Fig.1 The example of the view

Metrixx are: AverageJobNumber, CondorAveJobRun24h, CondorPledges, Running, TopologyMaintenance

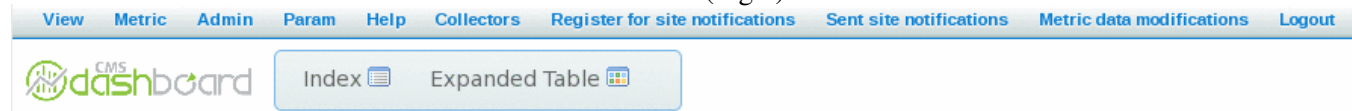
Navigation tools

There is a menu on the top where you can find links to the help, bug report system and to how to authenticate to the application. You will need to authenticate if you want to change some of the values.

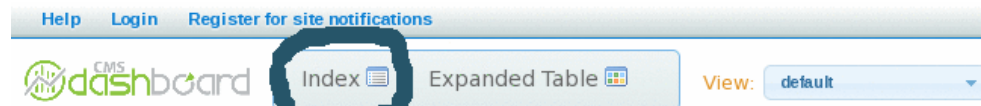
For the non-authorized user (can not perform any changes) the menu on the top of the view (Fig.2):



**Fig. 2 Menu for the non-authorized users.**

After authorization the additional items are added to menu (Fig.3):

**Fig. 3 Menu for the authorized users (administrator of view).**

From any page, if you click in the 'Index' (Fig.4), you will be redirected to the home page (Fig.4.1), and the 'Expanded table' (Fig.5) shows more details about View.. In these two pages, you can select the view with the dropdown menu (Fig.6).

**Fig. 4 Click to index to return to the home View page**

The home page (Fig.4.1) presents all the sites, with an icon that represents the status of the site on that particular view. To calculate the status of the site, the SSB takes all the critical metrics of the view (how to set the critical metrics for a view is explained in Fig. 4.2). A green tick next to a site name means the site is working fine, a yellow tick means at least one of the critical metrics is in warning, a red dot means at least one of the critical metrics is failing, an  symbol means at least one of the critical metrics reports maintenance color (brown), and a  symbol means the site is under investigation (i.e. a ticket has been submitted in the Savannah ticketing system to follow up the problem)

Help Login Register for site notifications

dashboard CMS

Index Expanded Table View: default

T0 +T1 T2

Status	Site Name	Status	Site Name	Status	Site Name
✓	T0_CH_CERN	✓	T2_AT_Vienna	✓	T2_FR_IPHC
✓	T1_CH_CERN	✓	T2_BE_IHHE	⚠	T2_HU_Budapest
⚠	T1_DE_KIT	⚠	T2_BE_UCL	⚠	T2_IN_TIFR
✓	T1_ES_PIC	✓	T2_BR_SPRACE	✓	T2_IT_Bari

Fig. 4.1 Home page

In the home page, if you click on:

- A site: you will get all the metrics for the site
- An icon next to the site: you will get the history of site for the critical metrics
- The magnifying glass (next to some sites): you will see the ticket associated to the problem
 - ◆ a ticket (acutally any URL) can be associated to the problem, meaning it is under invesdtigation, the lens will be automatically removed when the site becomes green again
 - ◆ `HowToSetTheLensInSSB`

The first screenshot shows the CMS dashboard with the 'Metric' menu open and 'List' selected. The second screenshot shows the 'View' dropdown menu with 'TESTCOLLECTOR' selected. The third screenshot shows the 'Defined columns' table with 'Critical' and 'Visible' toggle buttons.

Column Id	Column Name	Critical	Visible
11	# Links		
10078	1h Alarm		
10079	24h Alarm		
33	Active T1 links from T0		
35	Active T1 links from/to T1s		
34	Active T1 links to T2s		

Column Id	Column Name	Critical	Visible
10069	AverageJobNumber	OFF	ON
10071	CondorAve.JobRun24h	OFF	ON
10072	CondorPledges	OFF	ON
9	Running	OFF	ON
121	TopologyMaintenances	OFF	ON

Fig. 4.2 Defining critical metrics for a view

- From the menu for the authorized users select Metric -> List
- You go to a page with the defined metrics , from the view select list, select the view for which you want to defined critical metrics
- You go to the list of metrics defined for the selected view, now you can set which metrics are critical for this view

The screenshot shows the CMS dashboard with the 'Expanded Table' button highlighted.

Fig. 5 Click to Expanded table to get more details of the view

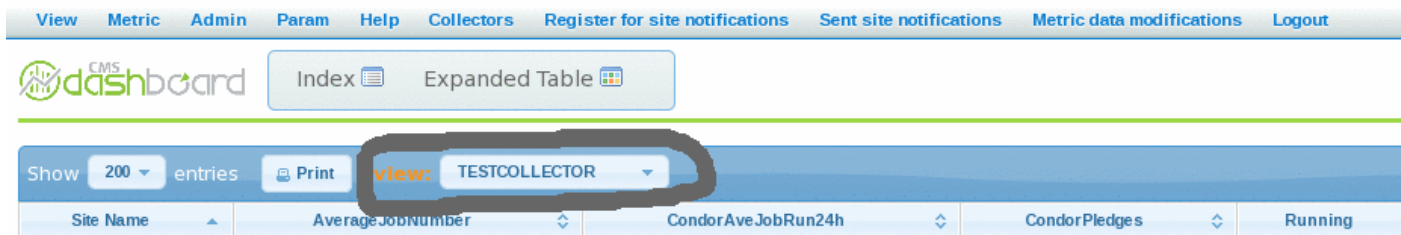


Fig. 6 Choose the view

The Expanded table shows all the metrics and the sites for a particular view. You can filter, sort, and save the information. Clicking on the header of any of the rows will lead to a page with more details about that particular metric: plots and tables with the evolution of the metric over, quality measurements plots, etc.

Information for the metric providers

Adding new metrics:

The SSB offers the possibility to define new metrics. The procedure to do this is quite straightforward.

- First of all, you will need write access to the SSB. If you click on 'Help -> List of administrators', you can see the people who are authorized to define the new metrics. If you are not there, you will have to ask one of the existing administrators to add you. You should send the DN subject (find it in your ~/.globus/usercert.pem, i.e. /C=XX/O=XXXX/....)

When you are registered and enter <https://dashb-ssb-dev.cern.ch> or <http://dashb-ssb.cern.ch> url or one of the other 2 instances (for alice and lhcb), you should see menu as in Fig. 3. If you still have Fig. 2 version something is wrong with your authentication. Please, inform administrators.

- Next, create a text file that will contain the values that you want to insert in the SSB. Each line of the file should be a row, with tab separated entries. The format is the following:

```
< timestamp > < sitename > <value > < color> < URL>
```

Lines starting with a # will be treated as comments and ignored. For instance:

```
[root@dashboard06 /tmp]# cat siteviewtext.13
#R.Santinelli: Dashboard feeder
#Space token share pledged
2009-12-14 13:30:45 LCG.CERN.ch 400 green http://sls.cern.ch/sls/history.php?id=CERN-LHCb_
2009-12-14 13:30:45 LCG.CNAF.it 30 green http://sls.cern.ch/sls/history.php?id=CNAF-LHCb_M
2009-12-14 13:30:45 LCG.GRIDKA.de 40 green http://sls.cern.ch/sls/history.php?id=GRIDKA-LH
2009-12-14 13:30:45 LCG.NIKHEF.nl 65 green http://sls.cern.ch/sls/history.php?id=SARA-LHCb
2009-12-14 13:30:45 LCG.PIC.es 15 green http://sls.cern.ch/sls/history.php?id=PIC-LHCb_MC_
2009-12-14 13:30:45 LCG.IN2P3.fr 65 green http://sls.cern.ch/sls/history.php?id=IN2P3-LHCb
2009-12-14 13:30:45 LCG.RAL.uk 40 green http://sls.cern.ch/sls/history.php?id=RAL-LHCb_MC_
```

This file should be accessible through a web page (let's assume <http://myhost/mymetricsource.txt>).

Be careful that you have all column in the proper order. If you do not see the update of the column (Metric) that you added to view, please, check the format of the mymetricsource.txt.

- Define the metric in the SSB. Login to the SSB. On the top left of the page, there should be a menu. Click on 'Metric' (Fig.7), and then on 'new...' (Fig.8). If you are one of the authorized contributors for that SSB, you will go to a form where you can describe the new metric (Fig.9).



Fig.7 Click on metric

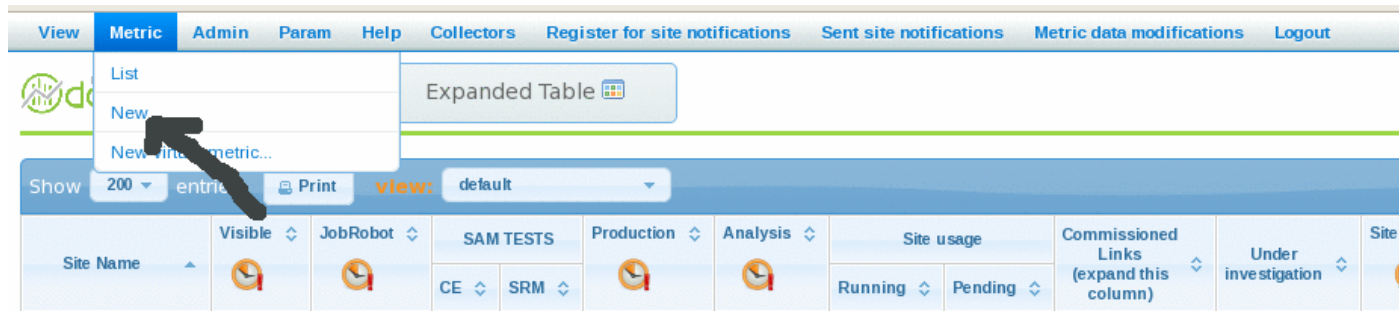


Fig.8 Click on new

 This screenshot shows the 'Metric data modifications' form. It is divided into two main sections: 'Column Information' and 'Column Source'.

 In the 'Column Information' section, there are input fields for:

- Column Name: MyMetric
- Column Description: Testing new metric
- Column Description Long: Testing new metric for sites
- Column Responsible: Rick.Smith@cern.ch
- Column URL: http://myhost/mymetricsource.txt
- Column Group: (empty)

 There are also checkboxes for 'Column Critical' (unchecked) and 'Visible' (checked).

 In the 'Column Source' section, there is a text area for 'Data source' containing 'http://myhost/mymetricsource.txt'. Below it, there is a checkbox for 'Uses Collector' (unchecked). If 'Uses Collector' is checked, there would be fields for 'Collector name', 'Column frequency' (set to 600), and 'Column validity' (set to 1200).

 At the bottom of the form, there is a blue button labeled 'Add column' with a black arrow pointing to it.

Fig.9 Describe new metric. Do not forget to click to Add column

The most important fields are the name of the column, the source (which should point to the file that you created in the previous step, <http://myhost/mymetricsource.txt>). If you use txt file then frequency and validity can be corrected while metric modification (next chapter). If you choose the collector then the frequency and validity can be set while metric creating.

* Note: Frequency 600 means update of column each 10 minutes. Validity value should be twice higher, i.e. 1200.*

Adding new metrics:

Modifying existing metrics

If you want to change the information of any metric (like the name of the metric, source, refresh rate, etc.):

Click to Metric and then to the list (Fig. 10).

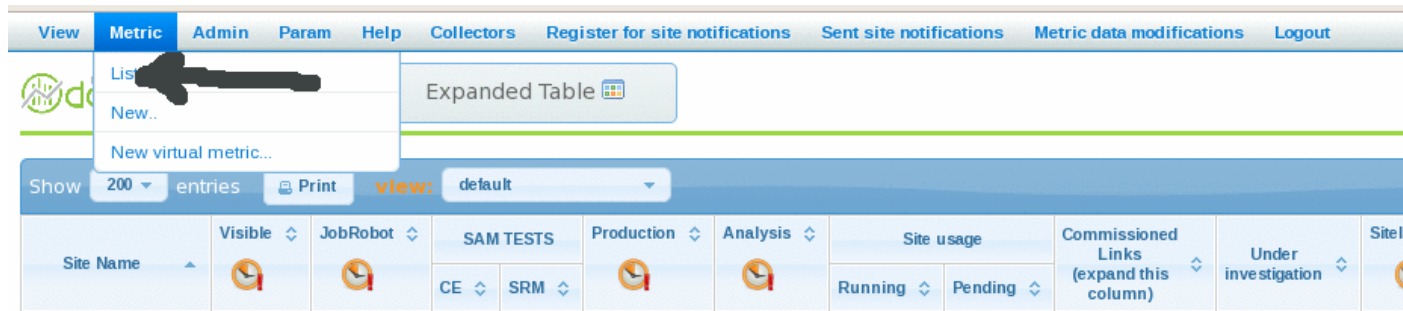


Fig.10 Click on list

Browse over the list and choose the needed metric. Click to Modify (Fig.11).

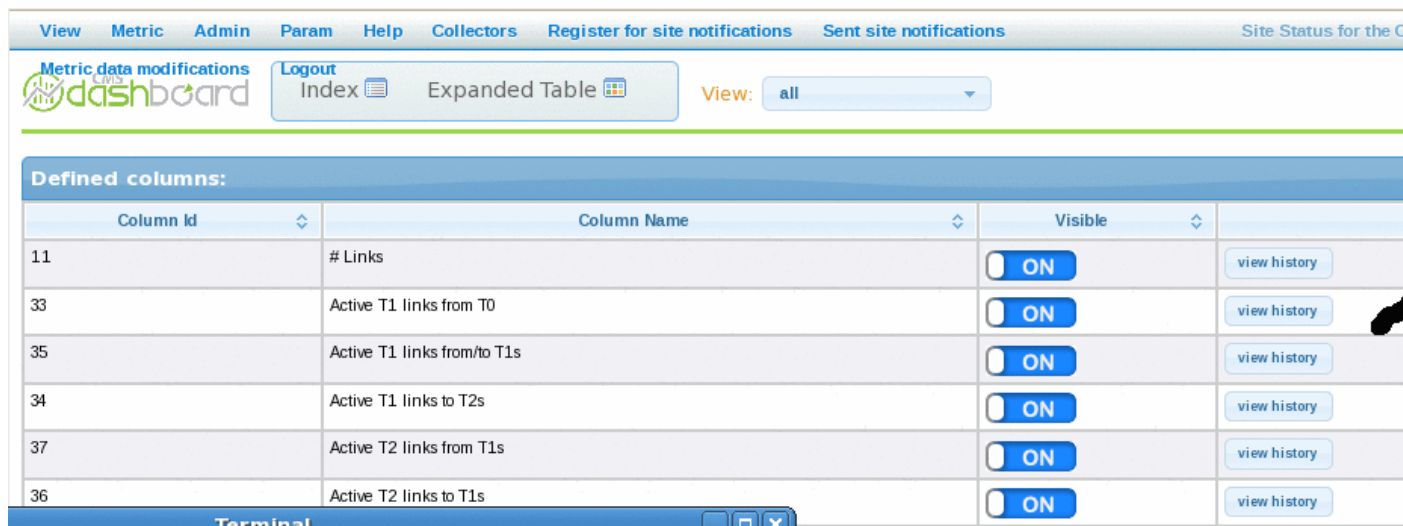



Fig.10 Click on Modify

After modification the data, click to **save** (Fig.11).

View Metric Admin Param Help Collectors Register for site notifications Sent site notifications Site Status for the C

Metric data modifications Logout Index Expanded Table

Modify Column

 Go to the plot

Fill the fields that you want to modify

Name	Value		
Column Name	AverageJobNumber	?	Save
Column Description	Averaged number of jobs	?	Save
Column Description Long	Averaged number of jobs per unit time for 24 hours interval	?	Save
Column Responsible	Olga.Kodolova@cern.ch	?	Save
Column URL	http://cmsdoc.cern.ch/~kodolova/SSBSEPT2012/mysites.txt	?	Save
Column Source	http://cmsdoc.cern.ch/~kodolova/SSBSEPT2012/mysites.txt	?	Save
Column Group	null	?	Save
Column Combined	null	?	Save
Column Combined Links	null	?	Save
Column Frequency	600	?	Save
Column Validity	7200	?	Save

Fig.11 Click on save

Creating virtual metric

A virtual metric is a combination of two or more metrics. The SSB will fetch the source metrics, and it will store the combined status of all the metrics. To define a new 'virtual metric', you have to login to the application. Then, click on 'Metric -> New virtual metric', and fill up the form.

Creating views

Creating a new view is also pretty easy. Again, you will need write access to the application. Once you have it, follow the next steps:

- Login to the application (<http://dashb-ssb.cern.ch> url or one of the other 2 instances (for alice and lhcb))
- Click on 'View -> List'


View Metric Admin Param Help Collectors Register for site notifications Sent site notifications Metric data modifications Logout

List Index Expanded Table

Fig.12 Click to view list

- Fill up the form at the bottom with the name of the view.

View Metric Admin Param Help Collectors Register for site notifications Sent site notifications Metric data modifications Logout

 CMS [Index](#) [Expanded Table](#)

Existing views

View Id	View	Columns
3	commission	SiteComm JR, TopologyMaintenances, HC, JR (Dashboard), Site Nagios availability, IsSiteInSiteDB?, Good links combined, SiteReadiness Status, Comm
4	expert	Pledged, TB, SiteSize, Pledged slots
6	maint	Maint Info, TopologyMaintenances, BDII Topology, All maintenances with topology, Topology, Maintenance in SAM
7	transfers	Transfer_rate_phedex, T0_T3_phedex, T0_T2_phedex, T0_T1_phedex, T0_T0_phedex, Out_rate_phedex, In_rate_phedex
8	jobrobot_debug	JobRobot_1h, JobRobot_24h
9	storage	Custodial (TB), Installed tape (TB), Installed Disk (TB), Total tape (TB), Total disk (TB), Free tape (TB), Used tape (TB), Tape Pledge (TB), Free disk (TB), Incoming (TB), Non-custodial (TB)
10	test	Commissioned T2
11	default	Visible, TopologyMaintenances, Savannah CMS, Out_rate_phedex, In_rate_phedex, SiteIssues, Under investigation, Commissioned Links, Pending, Running, Analysis, Production, SRM, CE, JobRobot
13	TESTCOLLECTOR	Running, CondorPledges, CondorAveJobRun24h, AverageJobNumber, TopologyMaintenances
12	blabla	null

Add new view

Add new view:

Name: ☒ Visible [Create View](#)



 Type name and then push Create view

Fig.13 Fill the form and click to Create view

You can also uncheck the 'visible' checkbox, to make sure that, for the time being, you are the only one who can see the view.

- After that, you should see the view in the table above, and you can click on 'modify' to change its configuration

[View](#)
[Metric](#)
[Admin](#)
[Param](#)
[Help](#)
[Collectors](#)
[Register for site notifications](#)
[Sent site notifications](#)
[Metric data modifications](#)
[Logout](#)



[Index](#)
[Expanded Table](#)

Existing views

View Id	View	Columns
3	commission	SiteComm JR,TopologyMaintenances,HC,JR (Dashboard),Site Nagios availability,IsSiteInSiteDB?,Good links combined,SiteReadiness Status,Commissioned L
4	expert	Pledged, TB,SiteSize,Pledged slots
6	maint	Maint Info,TopologyMaintenances,BDII Topology,All maintenances with topology,Topology,Maintenance in SAM
7	transfers	Transfer_rate_phedex,Tb_T3_phedex,Tb_T2_phedex,Tb_T1_phedex,Tb_T0_phedex,Out_rate_phedex,In_rate_phedex
8	jobrobot_debug	JobRobot_1h,JobRobot_24h
9	storage	Custodial (TB),Installed tape (TB),Installed Disk (TB),Total tape (TB),Total disk (TB),Free tape (TB),Used tape (TB),Tape Pledge (TB),Free disk (TB),Used disk (TB),Incoming (TB),Non-custodial (TB)
10	test	Commisioned T2
11	default	Visible,TopologyMaintenances,Savannah CMS,Out_rate_phedex,In_rate_phedex,SiteIssues,Under investigation,Commissioned Links,Pending,Running,Analysis,Production,SRM,CE,JobRobot
13	TESTCOLLECTOR	Running,CondorPledges,CondorAveJobRun24h,AverageJobNumber,TopologyMaintenances
12	blabla	null
14	MyView	null

Add new view


Add new view:

Name: ☒ Visible [Create View](#)

Fig.14 Find your newly created view in the list click to modify

- Each correctable value has a box where you can type value and the button save attached to each row.
- In the middle of the page there is a box separated into two boxes (Fig.15). The right box shows the list of existing metric. Choose metric and push "+". The name of metric will be moved to the left box (Fig.16). Choose as many metrics as you need. Do not forget to push save in the bottom of the page (Fig.17).

View Metric Admin Param Help Collectors Register for site notifications Sent site notifications Metric data modifications Logout

 Index Expanded Table

Modify View

View Name: MyView

☒ Visible

Text comments in the SSB pages:




View responsible:

This are the columns visible in the the view :


0 items selected	Remove all		Add all
		# Links	+
		Active T1 links from T0	+
		Active T1 links from/to T1s	+
		Active T1 links to T2s	+
		Active T2 links from T1s	+
		Active T2 links to T1s	+
		All maintenances	+
		All maintenances color	+
		All maintenances with topology	+
		Analysis	+
		Available BDII space	+
		AverageJobNumber	+
		BDII Topology	+



Fig.15 Find the needed metric in the list and click to "+"


View Metric Admin Param Help Collectors Register for site notifications Sent site notifications Metric data modifications Logout


 Index  Expanded Table 

Modify View















View Name: MyView  Save

 Visible  Save

Text comments in the SSB pages:  Save

View responsible:  Save

This are the columns visible in the the view :

1 items selected	Remove all		Add all
 AverageJobNumber	-	# Links	
		Active T1 links from T0	
		Active T1 links from/to T1s	
		Active T1 links to T2s	
		Active T2 links from T1s	
		Active T2 links to T1s	
		All maintenances	
		All maintenances color	
		All maintenances with topology	
		Analysis	
		Available BDII space	
		BDII Topology	
		BDII used space	


 Save

Fig.16 Metric is moved to the left column. If you want to remove it - push to "-"

View Metric Admin Param Help Collectors Register for site notifications Sent site notifications Metric data modifications Logout

ddashboard CMS

Index Expanded Table

Modify View

View Name: MyView [Save](#)

☒ Visible [Save](#)

Text comments in the SSB pages: [Save](#)

View responsible: [Save](#)

This are the columns visible in the the view :

1 items selected	Remove all		Add all
↕ AverageJobNumber	—	# Links	+
		Active T1 links from T0	+
		Active T1 links from/to T1s	+
		Active T1 links to T2s	+
		Active T2 links from T1s	+
		Active T2 links to T1s	+
		All maintenances	+
		All maintenances color	+
		All maintenances with topology	+
		Analysis	+
		Available BDII space	+
		BDII Topology	+
		BDII used space	+

[Save](#)

Fig.17 Push save at the bottom of the page

Notification when a site changes color in particular view

Authorized users can register for notifications when a site changes color (Fig 18)

View Metric Admin Param Help Collectors **Register for site notifications** Sent site notifications Metric data modifications Logout

ddashboard CMS

Index Expanded Table

The following people will be notified when a change in the color of a site occurs (per particular view)

Name	Email	Cert	VONamePattern
Andrea	Andrea.Sciaba@cern.ch	/DC=ch/DC=cern/OU=Organic Units/OU=Users/CN=asciaba/CN=430796/CN=Andrea Sciaba	T
Andrea	Andrea.Sciaba@cern.ch	/DC=ch/DC=cern/OU=Organic Units/OU=Users/CN=samcms/CN=430796/CN=Andrea Sciaba	T
Ivan	idzhunov@cern.ch	/DC=ch/DC=cern/OU=Organic Units/OU=Users/CN=idzhunov/CN=696069/CN=Ivan Antoniev Dzhunov	T

Fig.18 Register for site notifications

- From the Register site notifications link you go to the page with people registered for notifications. You can add/edit/delete only your own entries (you are logged in with your certificate). You can define for which view and for which sites (SQL like search operation) to be notified. Once a change

in color for a site and view occurs, you will receive an email with link to the site history for this view (Fig. 19)

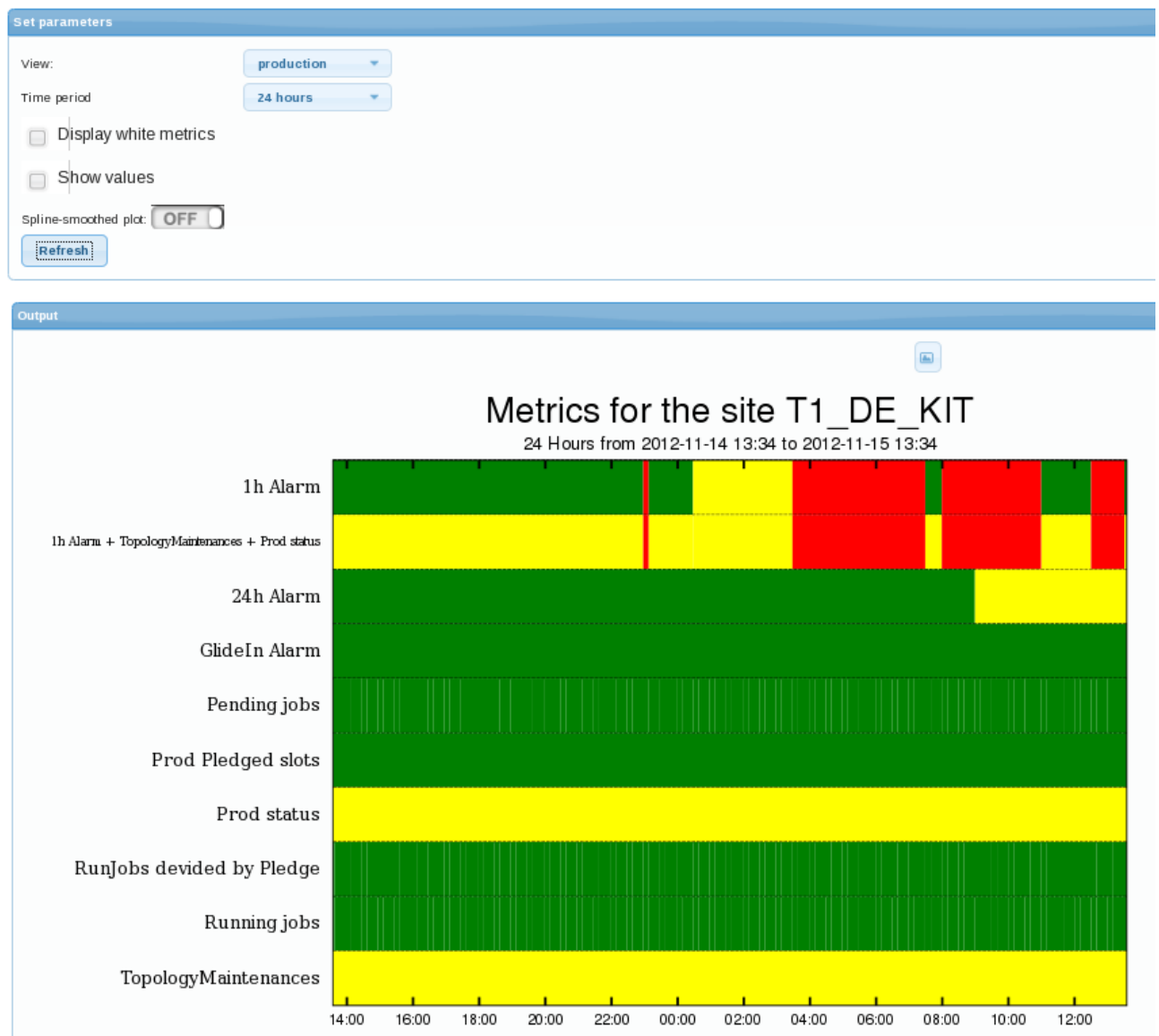


Fig.19 Site history for a view

For Site readines calculations 'loop of information' approach is used. That means that SSB admin can insert data into SSB for a set of metrics and after that with algorithm for analyzing and combining metrics' data into single metric to feed a new metric into SSB. Then a SSB admin can register to receive notifications when a site changes color for a view in which the newly created metric is a critical one.

Information for the administrators

At the moment, there are four instances of the SSB installed:

- dashb-ssb.cern.ch. Connecting to the cms_dashboard database. There are two collectors on that machine, siteviewcms and siteviewcmstext. The first one creates the metrics that we generate ourselves (like SAM, maintenance, job efficiency, etc.), and the second one collects the URL published by the metric providers. To start/stop these collectors:

```
ssh -l root dashb-ssb /opt/dashboard/bin/dashb-agent-start siteviewcms
ssh -l root dashb-ssb /opt/dashboard/bin/dashb-agent-start siteviewcmstext
```

There is a cronjob that checks every hour if the collectors are running.

- dashb-ssb-devel.cern.ch. At the moment, no collectors are running on this machine. This machine should be configured to connect to the development database of cms
- dashb-alice-ssb (aka dashboard06). ALICE only needs one collector:

```
ssh -l root dashb-alice-ssb sudo -u dashbop /opt/dashboard/bin/dashb-agent-start siteviewg
```

- dashb-lhcb-ssb (aka dashboard06). LHCb has two collectors:

```
ssh -l root dashb-alice-ssb sudo -u dashbop /opt/dashboard/bin/dashb-agent-start siteviewt
ssh -l root dashb-alice-ssb sudo -u dashbop /opt/dashboard/bin/dashb-agent-start siteviewg
```

Usage of the new dashboard.authorise module in SSB

The new module dashboard.authorise was made by Frédéric Saam, part of his job was to integrate it in the SSB application. Here are the step he followed to integrate them, those are important information for the administrator and developers of the website.

Please, keep in mind this is a part of his report, it is not written like a tutorial.

Here is the documentation of the module.

Items management

In this application, the different items we have to take care of are:

- Metrics
- Views
- Admins
- Parameters
- Profiles

The Admins part will be removed, since with my access control system we have a new way of managing the users. For the other items, what I found playing with the application is that:

- Metrics

You need to be logged in and have the modifyMetric Boolean set too true if you want to add/modify/remove them. Using the authorise module, I can now split the permissions with more granularity, which means that if someone wants to add a metric, he will need to have the metric.add permission and to modify, he will need metric.modify .

Even more, now the permissions will be set by metric, which means that the person with the metric.modify.% will have the permission to modify every metrics of the database, but the user who has the metric.modify.123 will only be able to modify the metric with the ID 123.

For this type of item, one thing I have to do is that if someone adds a metric in the application, I have to give him the permission to modify/remove it in case he doesn't have the generic permissions on the items.

- Views

Those items are working the same way that metrics are so I will use the same type of permission values using view instead of metric .

- Admins

Like I said before, this item will be removed from the application. However, what I saw is that normal users are able to see the list of administrators and the permissions that they have. To make it work the same way, I will create the `users` group and add the permission `user.view` in order to give normal users access to the list of my module.

- Parameters

At the moment, you need to be logged in and to be present in the administrator table (even without Boolean set to true) to modify parameters in this application. What I will do is to set the same type of permission than the one we use for the other items.

- Profiles

In the actual application, someone who has the `updateView` Boolean set to true is also able to add/modify/remove profiles. What I will do is to separate it from the views with my module.

Integration steps

The integration of my module in the SSB application will be done in two different steps. First, I will modify all of the current access control parts of the code to use my module. I will next add some logic in the application (for instance giving all the permission on a metric for the person who created it).

For the first step, what I have to do is:

1. Make the DAO inherit from the `OracleAuthoriseDAO`
2. Modify the `authenticate` method
3. Find the files containing the method that I have to replace
 1. Method : `authenticateWithUpdateViewMetricAdminAccess`
 2. Replace the actual code
 3. Remove the old access control system
 1. Also remove the old admin view page

For the second step, what I have to do is to define the logic parts that I want to add and implement them.

Before first step

After playing a bit with the SSB application, I noticed that for each view there are multiple endpoints that are called. In order to make the integration easier, I will make a list of the endpoints that are called for each view, but only on the endpoint where we actually need an access control system. While making the list, I will add the permission value that I will set for each.

I know that everything in this list should be in lower case, but I just copy/pasted the name of the endpoints without modifying them.

Views related endpoints

- /viewsform (list of views)
 - ◆ /meta
 - ◆ /getviewsandcols
 - ◆ /removeview (when pressing the remove button)
 - ◇ Permission : `view.remove.<viewId>`
 - ◆ /addview (when submitting the form)
 - ◇ Permission : `view.add`

- ◆ /changeOneField (when modifying the visible value)
 - ◇ Permission : view.update.<viewId>
 - ◇ /modifyviewform?viewid (update a view)
 - /meta
 - /getViewInfo
 - Permission : view.update.<viewId>
 - This endpoint is only used to create the modify view form.
- ◆ /modifyview (when submitting the form)
 - ◇ Permission : view.update.<viewId>
- ◆ /changeOneField (when updating for one field)
 - ◇ Permission : view.update.<viewId>
- ◆ /modifyviewlegend (when saving a new legend)
 - ◇ Permission : view.update.<viewId>

Metrics related endpoints

- /columnsform (list of metrics)
 - ◆ /meta
 - ◆ /getviews
 - ◇ This endpoint is used to give the view list in the metric section. You don't need a special permission you just need to be logged in.
 - ◆ /getviewsandcols
 - ◆ /modifycolumnform (update column information)
 - ◇ /meta
 - ◇ /getColumnInfo
 - Permission : metric.update.<metricId>
 - This endpoint is only used to create the update metric form.
 - ◆ /changeOneField
 - ◇ Permission : metric.update.<metricId>
 - ◇ /addcolumnform (add a metric)
 - /meta
 - /addcolumn (when submitting the form)
 - Permission : metric.add
 - /addvirtualform
 - ◆ /meta
 - ◆ /getviewsandcols
 - ◆ /addvirtual
 - ◇ Permission : metric.add
 - ◇ This endpoints call the same action than the /addcolumn endpoint

Metric history related endpoints

Those endpoints are different views on a metric history. I have to check the same permission value in all of them which is metric.view .

- /siteviewhistory (history of a metric)
 - ◆ /meta
 - ◆ /getchangedsitesforcolumn
 - ◆ /getcolumninfo
 - ◆ /getplotdata
 - ◆ /siteviewhistorywithstatistics (history of a metric with statistics)
 - ◇ /meta
 - ◇ /getchangedsitesforcolumn

- ◊ /getcolumninfo
- ◊ /getplotdatawithstatistics
- ◊ /sitereadinessrank (site state)
 - /meta
 - /getchangedsitesforcolumn
 - /getcolumninfo
 - /getsitereadinessrankdata
 - /sitereadinesssites
 - /meta
 - /getchangedsitesforcolumn
 - /getcolumninfo

Virtual metrics related endpoints

- /addvirtualform (adding a metric which keep tracks of multiple metrics)
 - ◆ /meta
 - ◆ /getviewsandcols
 - ◆ /addvirtual (when submitting the form)
 - ◊ Permission : metric.add

Parameters related endpoints

- /parametersform (list of parameters)
 - ◆ /meta
 - ◆ /getparams
 - ◊ Permission : param.view
 - ◆ /setparams (to add/update/remove a parameter)
 - ◊ Permission : param.manage
 - ◊ Since there is only one action file for the parameter management, it's more accurate to put manage instead of add/update/remove

Profiles related endpoints

- /profilesform (list of profiles)
 - ◆ /meta
 - ◆ /getprofiles
 - ◊ Permission : profile.view
 - ◆ /removeprofile (when clicking the remove button)
 - ◊ Permission : profile.remove.<profileId>
 - ◊ /addprofileform (add a profile)
 - /meta
 - /getmetricsforprofilecreation
 - Permission : profile.view
 - ◆ /addprofile
 - ◊ Permission : profile.add
 - ◊ /listprofilemodifications (history of profiles)
 - /meta
 - /getprofilemodifications
 - Permission : profile.view
 - /modifyprofileform (update profile values)
 - ◆ /meta
 - ◆ /getmetricsforprofilecreation
 - ◊ Permission : profile.view
 - ◆ /getProfileInfo

- ◊ Permission : profile.update.<profileId>
- ◆ /updateprofile (when clicking on the update button)
 - ◊ Permission : profile.update.<profileId>

Endpoints not used in the graphical interface of the SSB

- /getAggregationInfo
 - ◆ An aggregation is something used in virtual metrics.
 - ◆ Permission : metric.view
 - ◆ /modifyAggregation
 - ◊ Permission : metric.update.<metricId>
 - ◊ /getplotdatacombined
 - Retrieves metric related values
 - Permission : metric.view

In this list there are three particular endpoints to notice which are `meta` , `getviewsandcols` and `changeOneField` . The `meta` endpoint is used to create the menu that the user can see and the other ones `changeOneField` and `getviewsandcols` are used by different objects (metrics and views).

The `getviewsandcols` endpoint retrieves the views and the columns (as the name is saying). The problem is that this is two different types of object, so the best solution is to return nothing if the user hasn't both `metric.view` and `view.view` permissions.

The `changeOneField` endpoint is also used by multiple types of object. However one of the parameters that you have to give contains the table that you want to modify. The best solution is to make an access control by type of object.

The `meta` endpoint is used to create the top bar which contains the menu. What I will do is to check if the user is logged in and if he exists in the database. If so, I'll show the admin menu, if not I'll show the standard menu (this is the actual behaviour).

First step

Now that I have the list of endpoints and the permission that I want them to check, I'll have to implement it. After the implementation is finished, I'll check if there are still some files with the `authenticateWithUpdateViewMetricAdminAccess` method in it.

The `authenticate` method checks if the current user is in the database. If he is, nothing particular is done, if he isn't an exception is raised. I will replace the actual check to use my DAO instead.

Each time I modify something in the backend, I have to check that it works (obviously) and I also have to check that the view is still correct. I also took the opportunity to correct some typos and some other errors like a form that was not updated accordingly to the backend.

Second step

Now that every action file in the SSB uses my module instead of the old access control system, it is necessary to add some logic to it.

The first thing to do is whenever a person adds a metric/view/profile, I have to add all permission on this object to the creator. Concerned action files are:

- Metric : AddColumn.py
- View : AddView.py
- Profile : AddProfile.py

The parameters are not concerned by this logic step, because the endpoint never uses IDs to check if the user is authorised. Instead we just use `param.manage` and the person is able to modify all of them (higher granularity).

To do so, I need to add some function in the `SSBDAO.py` (`ProfilesDAO.py` for the profiles) file in order to be able to retrieve the last inserted ID, since I need it to create the new permission. After this I will add the creation of the permission in the action files.

The second thing is that each time one of those object is removed from the database, we have to remove the permissions related to it in order to keep the integrity of the database. To do so I have a special function in my DAO called `removeObjectPermission` which will be applied to:

- View : `RemoveView.py`
- Profile : `RemoveProfile.py`

I would do it for the metrics also, but at the moment there is no way in the SSB application to remove a metric. There is no endpoint, which means that if someone wants to remove a metric he needs to do it in the database directly.

In order to add/remove permission for one person, I need to create a group for him and to check if he already has a global permission on this object in one of the other groups. For instance, if the person already has `object.update.%` and `object.remove.%`, I don't need to create the new permission. This works the same way for the person who has the permission `%` or `object.%`.

This do this part, I will create a function in the `SSBAction.py` file called `addPersonalPermissionIfNeeded` which verify if the person already possess the necessary permission on the object, and if not it will either create/get the personal group and add the permission to it (and link the user to the group).

Here is how the function is used in the `add` actions:

As we can see, the function itself takes three parameters which are:

- request

This is the request object as you receive it in the `habitual perform` method in action files. This is used to retrieve the user.

- objectString

This is just the name of the object for which you want to add the permission to the user.

- objectId

This is the ID of the object.

In the future what would be great is to add a way for the user to choose the group for which he wants to add the permission to. This would require a bit of work in the interface to add a group list in each add form. Unfortunately I don't have the time to do it myself.

Metric data modification from UI

Collected metric data can be modified from the UI. There are some requirements for this:

- The metric should allow metric data modifications , go to modify metric form and set it either to 'with time update' or 'without time update'

Column Name	Overall slots
Column Description	Number of expected slots at a given site
Column Description Long	null
Column Responsible	gutsche@fnal.gov
Column URL	null
Column Source	null
Column Group	null
Column Combined	null
Column Combined Links	null
Column Frequency	0
Column Validity	0
Column Update Hour	-1
Column Value Graph Title	null
Column History	<input type="checkbox"/> OFF
Column Visibility	<input checked="" type="checkbox"/> ON
Modify metric's history data	Without time update
Show values on statistics plot tooltip	<input type="checkbox"/> OFF
Combined default plot	<input type="checkbox"/> OFF
First Column Title	SITE Name
Topology Type	site
Obsolete	<input type="checkbox"/> OFF

- You should be SSB admin with Modify Metrics privileges - if you are not, ask dashboard-support.cern.ch

- You should be logged in SSB

When all 3 criterias are met, you can go to metric default plot in a couple of ways:

- From Expanded Table , click on a Column header
- Log in SSB, Go to Login -> Metric -> List and click "View history" button for the desired metric

Once you are at metric default plot page, you can right click on a cell from the plot and a popup window will show up from where you can modify the data

The screenshot shows the CMS dashboard interface. The browser address bar displays `https://dashb-ssb.cern.ch/dashboard/request.py/siteviewhistory?columnid=145`. The dashboard has a top navigation bar with links: View, Metric, Admin, Param, Help, Collectors, Register for site notifications, Sent site notifications, Metric data modifications, and Logout. Below this is a sub-navigation bar with 'Index' and 'Expanded Table' tabs. The main content area has a 'Default plot' tab selected, with other tabs being 'Table with statistics', 'Ranking plot', and 'Number of sites'.

On the left, the 'Plotting parameters' section includes:

- Time period: 24 hours
- Sites: all sites
- Show values: ☐
- Spline-smoothed plot: OFF
- Debug layer: OFF
- Retfresh button

On the right, the 'Column Info' section has a 'View Column Info' button.

A 'Modify cell data' popup window is open, containing:

- ☐ Remove cell data from database
- Value: 1200
- Status: 1200
- URL: (empty field)
- ☐ Apply changes to all sites
- Ok and Cancel buttons

At the bottom left, the 'Output' section shows a list of site identifiers:

- T1_DE_KIT
- T1_ES_PIC
- T1_FR_CCIN2P3
- T1_IT_CNAF
- T1_TW_AS GC
- T1_UK_RAL
- T1_US_FNAL
- T2_AT_Vienna
- T2_BE_IIHE
- T2_BE_UCL
- T2_BR_SPRACE
- T2_BR_UERJ

How to understand which plugin updates given metric

This is what I do

`cd /opt/dashboard/var/log/siteviewgeneric_input/done` on the server and then

`ll *.metricId` for examble `ll *.35` will give you

Sep 21 16:13 1316614405.ATLASGENERIC_Virtual.35

Sep 22 11:17 1316683051.ATLASGENERIC_Virtual.35

and you see that Virtual plugin is responsible for metric 35

Profiles

- How to create and update a profile
- How to request a correction
- How to apply corrections

General help page on Dashboard

- CMSDashboardHelp

-- PabloSaiz - 12 Nov 2008

This topic: ArdaGrid > SiteStatusBoard

Topic revision: r22 - 2018-11-07 - AlbertoAimar



Copyright &© 2008-2024 by the contributing authors. All material on this collaboration platform is the property of the contributing authors.
or Ideas, requests, problems regarding TWiki? use Discourse or Send feedback