PMES Framework

PMES Dashboard

Version 2.3.3

User Manual

BSC
Barcelona Supercomputing Center
Centro Nacional de Supercomputación

Review Table

Date	Author	Changelog
29/04/2015	Cristián Ramón-Cortés	Layout and sections. General format
30/04/2015	Cristián Ramón-Cortés	Sections 1 and 2
04/05/2015	Cristián Ramón-Cortés	Screenshots and section 3 **(no 3.4, 3.5.1)**
05/05/2015	Cristián Ramón-Cortés	Section 3.4
06/05/2015	Cristián Ramón-Cortés	Section 3.5.1
14/05/2015	Rosa M Badia	Review of document
15/05/2015	Cristián Ramón-Cortés	Review of document
17/05/2015	Rosa M Badia	Second review
29/07/2015	Cristián Ramón-Cortés	Correct section 1. Add section 4.

Index

1. PMES Framework

1.1. Overview

The *PMES Framework* allows users to execute jobs in the cloud. Jobs are user defined applications that have been particularized (by defining its parameters, input files and output files) and that run inside one or more virtual machines (that have been automatically instantiated from a pre-selected image). The *PMES Framework* also provides a portal to interface with cloud infrastructures, allowing users to define applications and to execute and monitor jobs in the cloud.

For clarity purposes, the following modules are distinguished inside the *PMES Framework*:

- PMES-Service

  Consists of the *PMES* Runtime that performs all the PMES features

- PMES-Dashboard

  Provides a GUI to manage the *PMES-Service* via a web-client interface. Its current features include storage management, application management, job management, user management and some monitoring tools. For further information please take a look at section 2. *PMES Dashboard* in this document.

- PMES-WS

  Provides a Web-Service API. For further information please refer to *PMES WS user guide* and *the PMES WS JavaDoc* files that are accessible through the COMPSs BSC[1] webpage under the *PMES Framework* section[2].

1.2. Keywords definition

This subsection defines some terminology related to the *PMES Framework* in order to clarify aspects described in the next sections:

[1] http://compss.bsc.es

[2] http://www.bsc.es/computer-sciences/grid-computing/pmes-framework/downloads-and-documentation

- Storage

  Provides an accessible location to input and output files. A storage is defined as a location where users can access to upload or download the files needed or generated by a job. Currently, the *PMES Framework* supports FTP, GCube and CDMI protocols.

- Application

  Piece of software that can be run in the underlying infrastructure. It can be defined as a *COMPSs* application or as a simple application.

- Job

  A job is defined as a combination of an application and a set of files to be deployed at execution time. The results that users can obtain from a job execution are delivered in different files containing the application output and the PMES logs generated during the execution.

2. PMES Dashboard

The *PMES Dashboard* provides a GUI to manage the *PMES Service*. Principally, it enables the management of storages, applications, jobs and users.

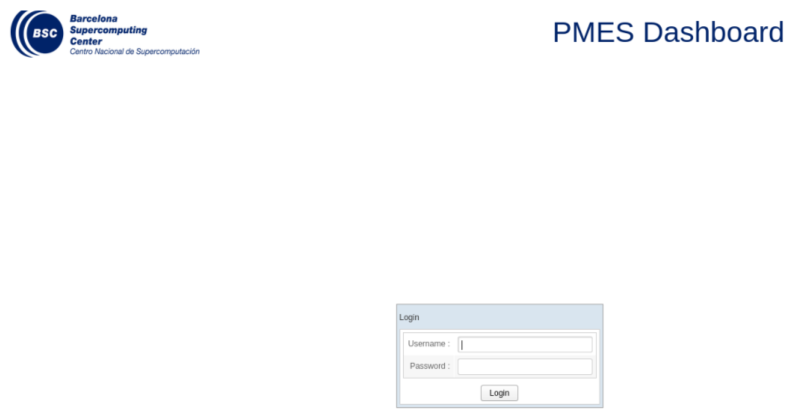
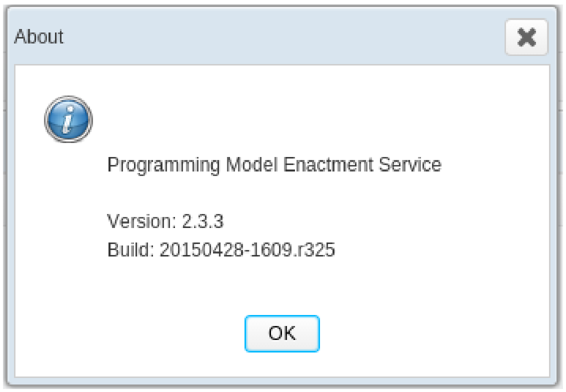

BSC
Barcelona Supercomputing Center
Centro Nacional de Supercomputación
PMES Dashboard
Login
Username :
Password :
Login

Figure 1: PMES Dashboard login webpage

As shown in *Figure 1*, it can be accessed via webpage and it is equipped with a credential system. Moreover, an *About* menu is provided to clarify which version of the *PMES-service* is used by the currently deployed *PMES Dashboard* (see *Figure* 2).



Programming Model Enactment Service

Version: 2.3.3
Build: 20150428-1609.r325

Figure 2: About menu

2.1. Typical Dashboard usage

The typical usage workflow of the dashboard consists of the following steps:

- Add the needed storages (endpoints and credentials)
- Create the needed application (either simple or COMPSs)
- Create and submit a new job
- Monitor the job state
- Retrieve the output and log files

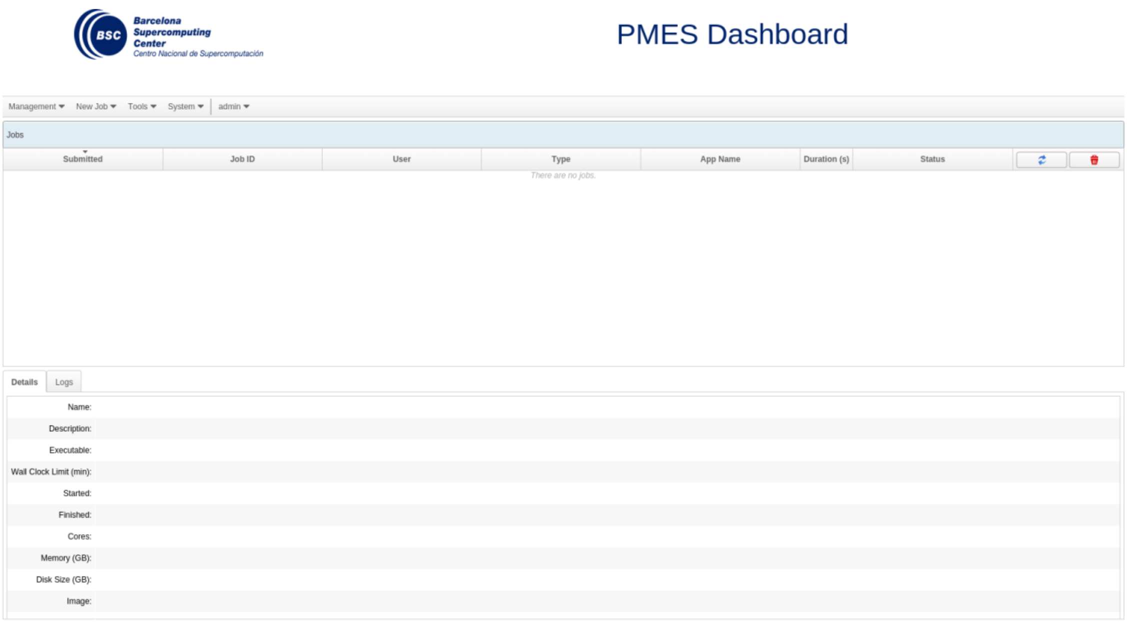
The next sections are meant to provide more in depth information for each of these steps.

3. Specific GUI Features

3.1. Main Page

After login into the *PMES Dashboard* the main page is displayed (see *Figure 3*). This window is designed to allow the users to access all the different features provided. The page is divided in three parts:

- A global menu with different sections for management, jobs creation, tools visualization, system information and user information
- A table providing information about user jobs
- Two tabs providing specific information of each job


Barcelona Supercomputing Center
Centro Nacional de Supercomputación
PMES Dashboard
Management
New Job
Tools
System
admin
Jobs
Details
Logs

Submitted	Job ID	User	Type	App Name	Duration (s)	Status

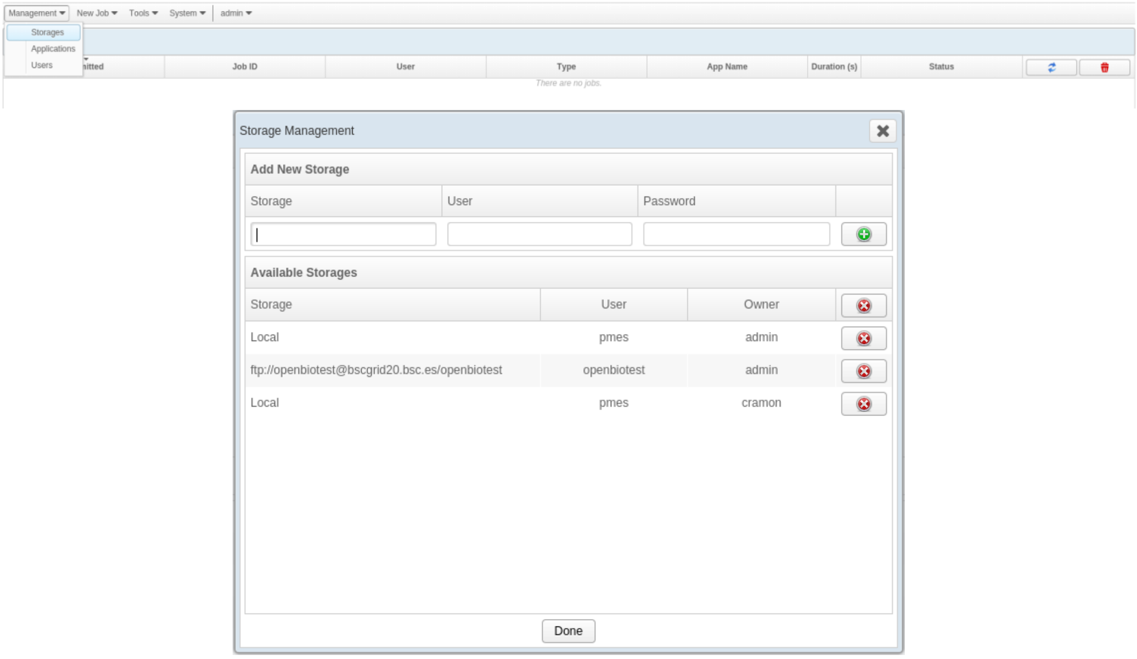
There are no jobs.

Name:
Description:
Executable:
Wall Clock Limit (min):
Started:
Finished:
Cores:
Memory (GB):
Disk Size (GB):
Image:

Figure 3: Main page

3.2. Storage Management

The *Storage Management* menu allows users to add a new remote location (e.g. FTP Server) and to remove existing ones. Storages are meant to enable jobs to retrieve input data and upload output data. As shown in *Figure 4* the menu is accessible though *Management > Storages*.



Add New Storage

Storage	User	Password
|		

Available Storages

Storage	User	Owner
Local	pmes	admin
ftp://openbiotest@bscgrid20.bsc.es/openbiotest	openbiotest	admin
Local	pmes	cramon

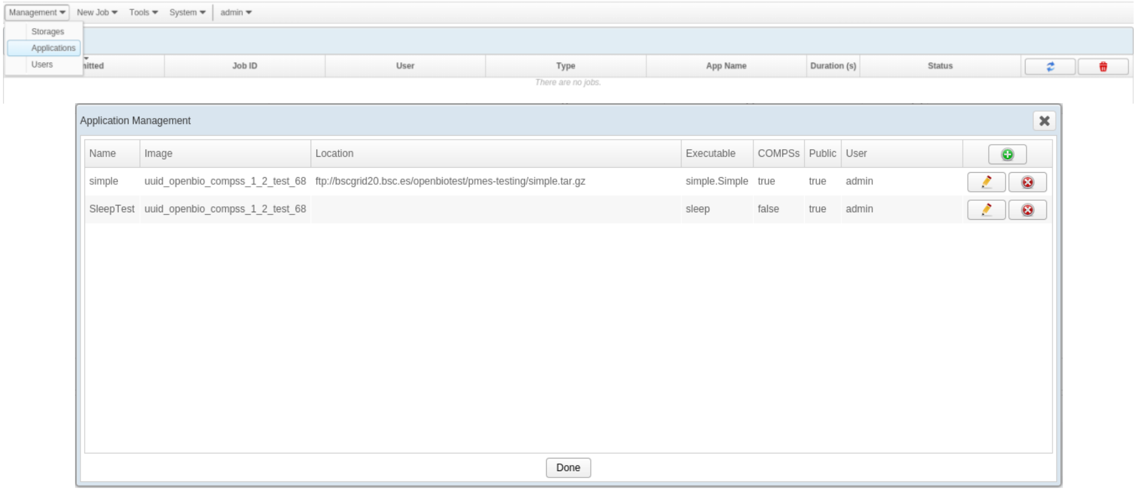
Figure 4: Storage Management

To add a new storage users must provide an endpoint, a login user and, optionally, a password in the upper box and hit the green button at its right (⊕).

To remove an existing storage from the list, users must hit the red button at its right (⊗).

3.3. Application Management

The Application Management menu allows users to create, modify and delete applications. As shown in *Figure 5* it is accessible through *Management > Applications*.

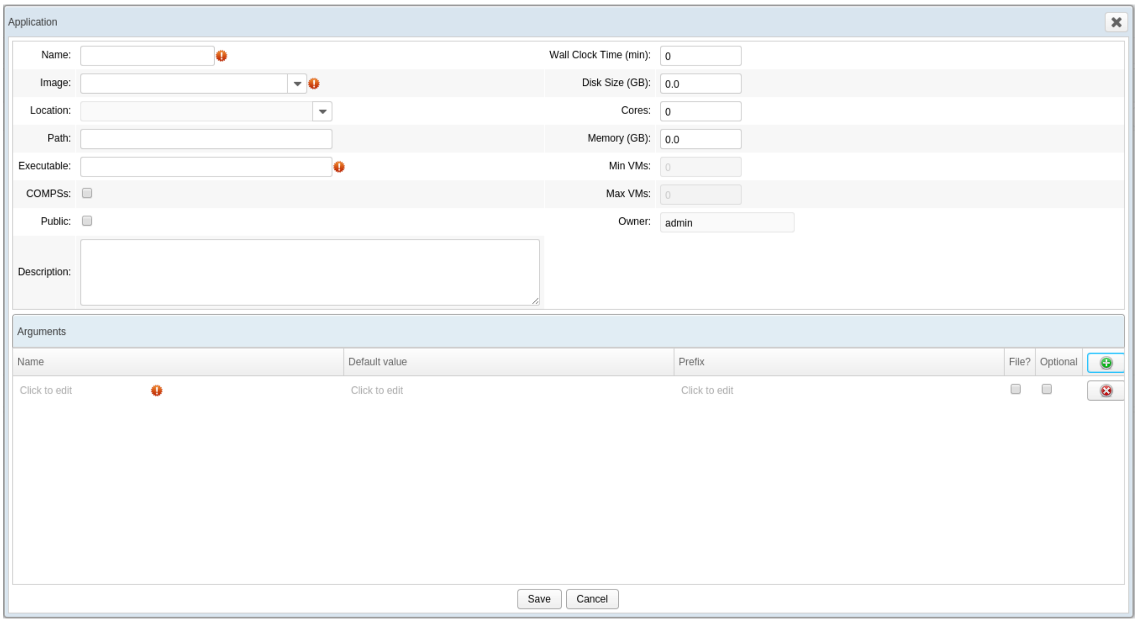


Name	Image	Location	Executable	COMPSs	Public	User
simple	uuid_openbio_compss_1_2_test_68	ftp://bscgrid20.bsc.es/openbiotest/pmes-testing/simple.tar.gz	simple.Simple	true	true	admin
SleepTest	uuid_openbio_compss_1_2_test_68		sleep	false	true	admin

Figure 5: Application Management

Users can create a new application by hitting the top right green button (⊕), modify an existing application by hitting the pencil button at its right (✎) or erase an existing application by hitting the red button at its right (⊗).

When creating a new application or modifying an existing application the form shown in *Figure 6* is displayed.



Name:		Wall Clock Time (min):	0
Image:		Disk Size (GB):	0.0
Location:		Cores:	0
Path:		Memory (GB):	0.0
Executable:		Min VMs:	0
COMPSs:	□	Max VMs:	0
Public:	□	Owner:	admin
Description:			

Name	Default value	Prefix	File?	Optional
Click to edit	Click to edit	Click to edit	□	□

Figure 6: Application Creation

The first thing that users must do is to specify if it is a COMPSs application or a single application by hitting the checkbox COMPSs (this will enable or disable some of the form fields, e.g. max VMs and min VMs).

Next, they must provide the mandatory fields (❗) and the optional fields. Below we provide detailed information about each field:

- **Name:** The name of the application
- **Image:** The name of the image that has to be used to create virtual machines when executing the application. If the infrastructure provider does not inform about the available images this information will have to be entered manually.
- **Location and Path (Optional):** Only if the application consists of a package that has to be deployed in the virtual machines before its execution (useful for COMPSs applications). The location specifies the storage where the package resides and the Path specifies the relative path to the package inside that storage. For example:

  Location = `ftp://bscgrid20.bsc.es`

  Path = `/app_repository/simple.tar.gz`
- **Executable:** Name of the executable to run. In COMPSs applications it refers to the name of the main class (e.g. `simple.Simple`) and, in single applications, it refers to the name of the command (e.g. `/usr/bin/maker`).
- **COMPSs:** Checkbox to indicate whether it is a COMPSs application or not.
- **Public:** Checkbox to indicate whether the application can be executed by other users or not.
- **Description (Optional):** A brief description of the application and its arguments.
- **Wall Clock Time (Optional):** The default limit time for the application execution.
- **Disk Size (Optional):** The default disk size of the virtual machines where the application must be run.

- **Cores (Optional):** The default number of cores of the virtual machines where the application must be run.
- **Memory (Optional):** The default memory size of the virtual machines where the application must be run.
- **Max VMs (mandatory if COMPSs):** The maximum number of virtual machines to run the application. Only used for COMPSs.
- **Min VMs (mandatory if COMPSs):** The minimum number of virtual machines to run the application. Only used for COMPSs.
- **Owner:** Specifies the user that has created this application. Not modifiable.

Finally, users can add as many arguments as their application needs by hitting the bottom right green button (⊕) and filling the following information for each parameter:

- **Name:** Name of the parameter.
- **Default value (Optional):** Default value for the parameter.
- **Prefix (Optional):** Prefix chain inserted before the parameter value.
- **Is file?:** Checkbox to indicate if the parameter is a file or not.
- **Optional:** Checkbox to indicate whether is an optional parameter or not.

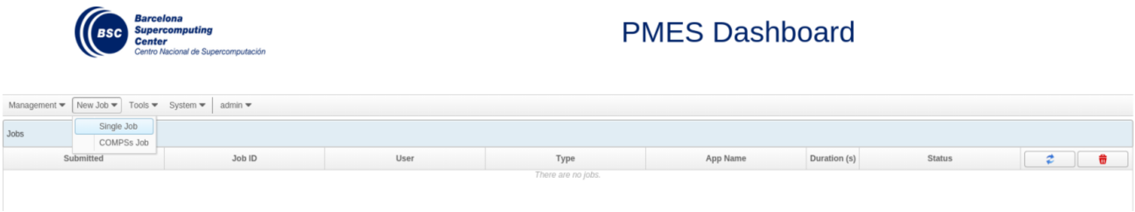
If needed, each argument can be erased by hitting the red button (⊗) at its right. Once all the information is provided, hitting the *Save* button will save the information into the *PMES Framework* database.

3.4. Jobs Management

Once an application has been created and the needed storages have been defined, it is possible to run jobs in the infrastructure.

3.4.1. Single job creation

As shown in *Figure 7*, to create a job for single applications the users need to access the *New Job > Single Job* menu.

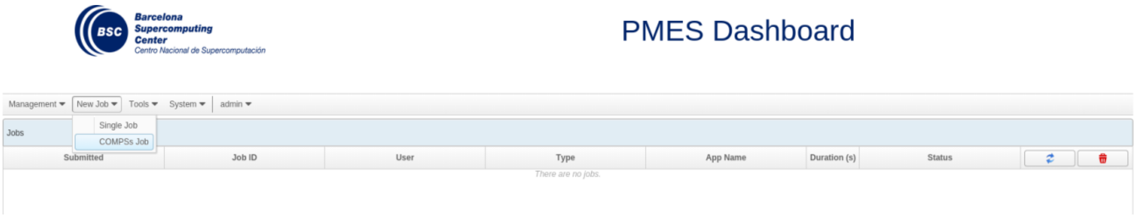


There are no jobs.

Figure 7: Single Job creation

3.4.2. COMPSs jobs creation

As shown in *Figure 8*, to create a job for COMPSs applications users need to access the *New Job* > *COMPSs* Job menu.



There are no jobs.

Figure 8 :COMPSs Job creation

3.4.3. Common features: job transfers and arguments

For any type of job, the users must fulfill the information asked by the submission dialog depicted in *Figure 9*. First of all, users must provide the name of the application to be run. If the application has not been created before, the application management menu can be accessed by hitting the green button (⊕) located below the application pull-down menu.

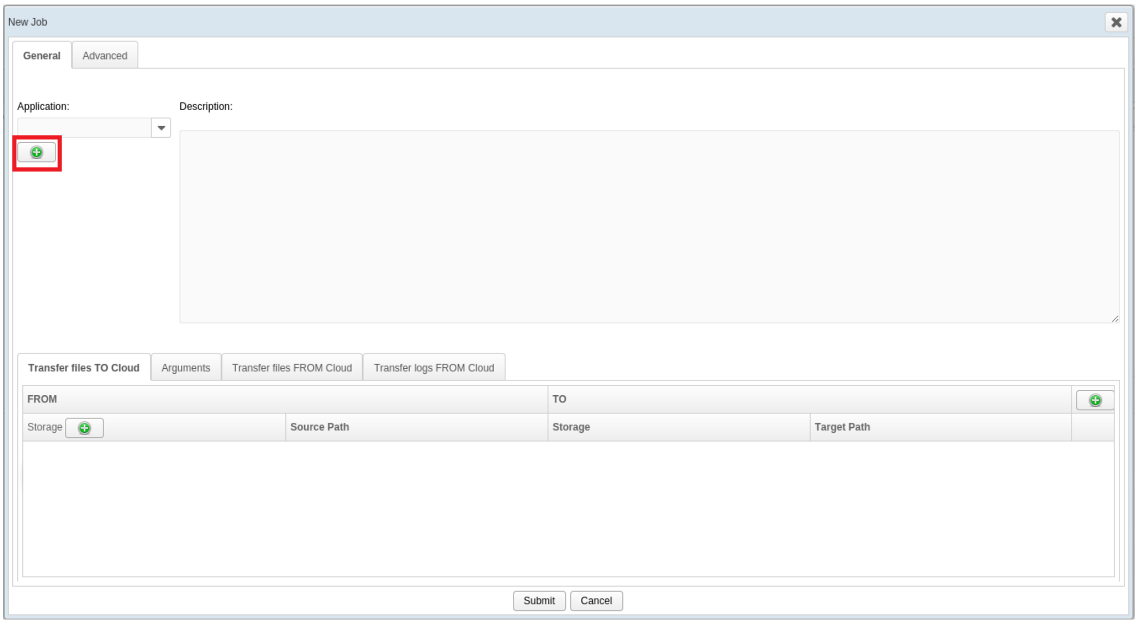


Figure 9: Job submission dialog

Next, in the *Transfer files TO Cloud* tab, they can specify a file transfer. Transfers are needed if any file located in a storage is needed by the job during the execution. If it is the case, users can hit the green button (⊕) located at the top-right corner of the tab and a new file transfer dialog box will appear (see *Figure 10*). For each file transfer users need to specify the source storage and the relative path of the file inside the storage (folders can also be transferred by ending the source path with "/"). If the *Local* storage is selected the file browser is activated and users can deploy local files directly to the shared disk. In any case, users must also indicate whether to copy the file directly to the virtual machine disk or to the shared disk[3] and the relative path inside it. If the target relative path is empty the file will be copied directly to the application working directory with the same source name.

Any transfer can be deleted by hitting the red button (⊗) at its right.

[3] Attention: shared disk is only available if *PMES Framework* administrator has activated this feature. If you are not seeing the shared disk option please contact your administrator.

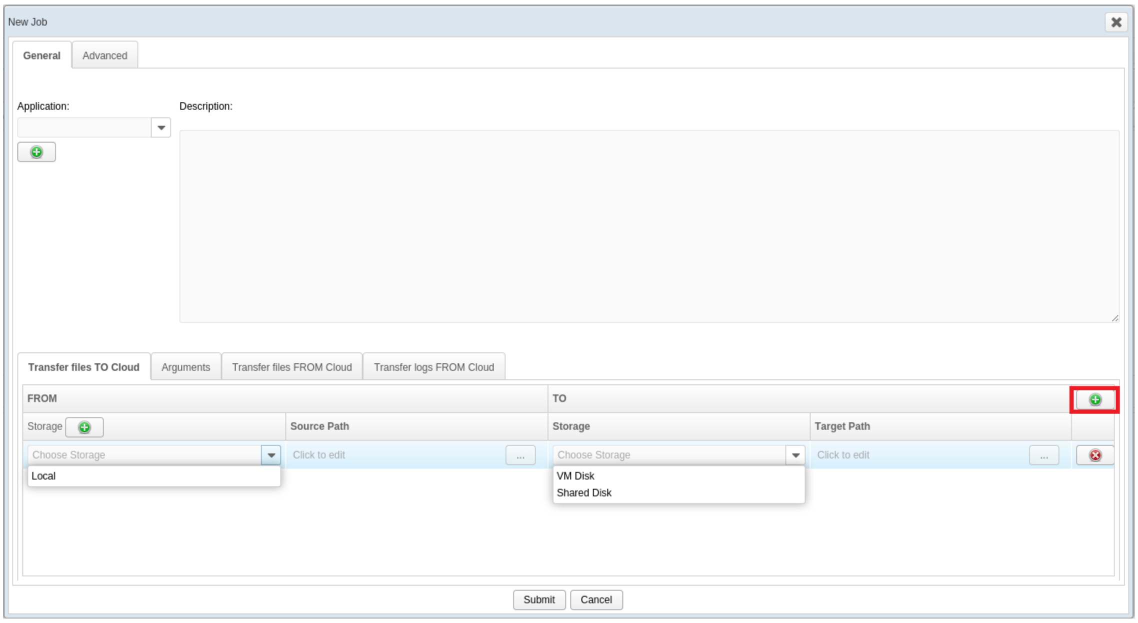
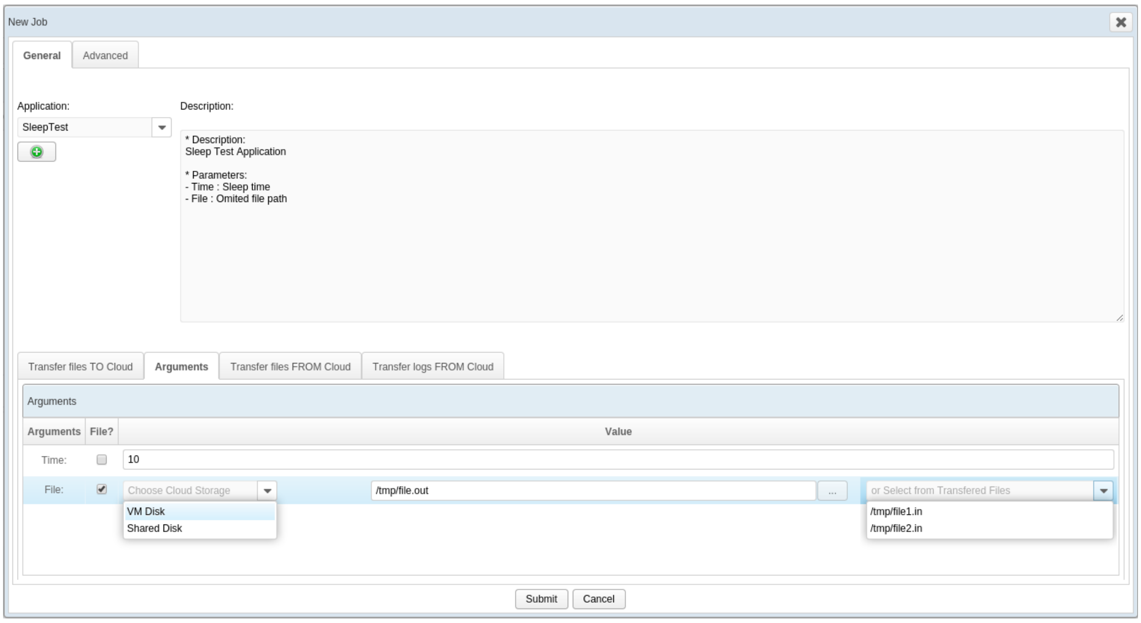


Figure 10: Job submission dialog – Transfer files to cloud

Next, in the *Arguments* tab, users can specify the values of the application arguments. The default values defined during the application creation will be automatically loaded. As shown in *Figure 11*, if the argument is a file, users must indicate whether it is located in the virtual machine disk or in the shared disk and its relative path. For the sake of simplicity, if the specified file is being created during the *Transfer TO cloud* stage, users can select this information by hitting the list provided at right of each argument. If the argument is not a file, users must provide its value (the prefix defined in the application menu is automatically added).



Application:
SleepTest

Description:

* Description:
Sleep Test Application

* Parameters:
- Time : Sleep time
- File : Omited file path

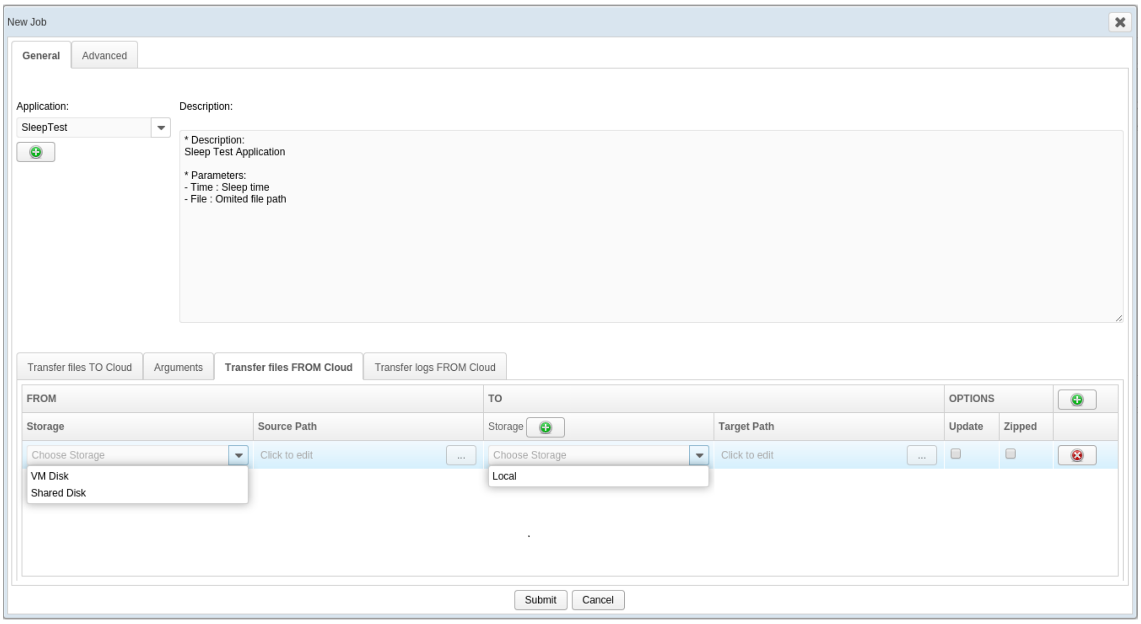
Arguments

Arguments	File?	Value
Time:	□	10
File:	☑	Choose Cloud Storage /tmp/file.out ... or Select from Transfered Files

Figure 11: Job submission dialog – Arguments

Next, in the *Transfer files FROM Cloud* tab, users can specify transfers to be done at the end of the application execution. Transfers are needed if users need to recover any file at the end of the execution and save it into a storage. Thus, they can add transfers by hitting the green button (⊕) at the top-right corner of the tab (see *Figure 12*). For each file transfer they need to specify the source location (by indicating whether it is located in the virtual machine disk or in the shared disk) and its relative path. Users must also specify the target storage and its relative path. If the target path is left empty, the file will be transferred with the same name into the root base directory of the storage. Moreover, they can indicate if the file must be updated during the execution and if it has to be zipped before performing the transfer by checking the checkboxes provided at right of each transfer.

Any transfer can be deleted by hitting the red button (⊗) at its right.



Application: SleepTest

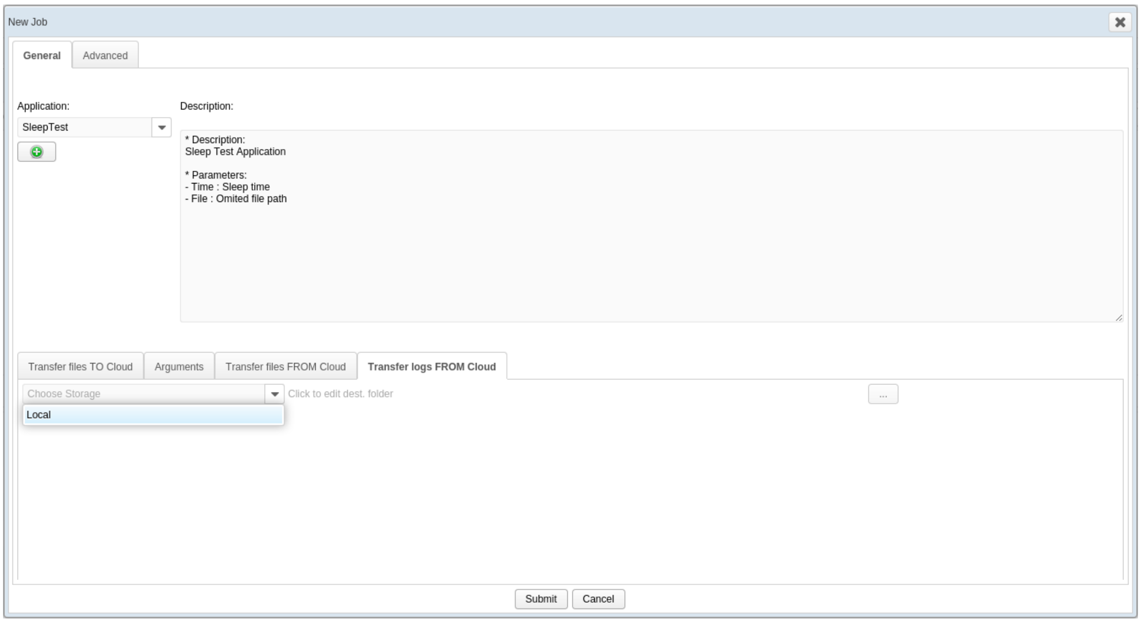
Description:

* Description:
Sleep Test Application

* Parameters:
- Time : Sleep time
- File : Omited file path

Figure 12: Job submission dialog – Transfer files FROM cloud

Next, the *Transfer logs FROM Cloud* tab allows users to indicate a storage to save all the job log files generated by the *PMES Framework*. If provided, these files will be transferred at the end of the execution. To configure this feature they must provide a target storage and the relative path inside this storage of the **directory** where the logs must be saved (see F*igure 13*).



Application: SleepTest

Description:

* Description:
Sleep Test Application

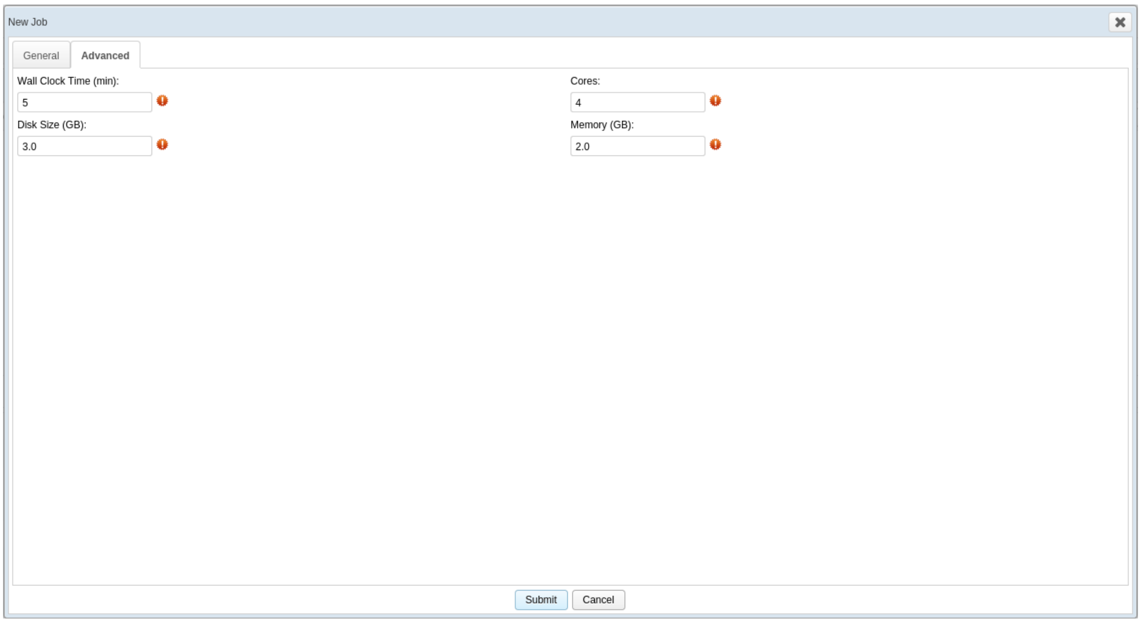
* Parameters:
- Time : Sleep time
- File : Omited file path

Figure 13: Job submission dialog – Transfer logs FROM cloud

Finally, in the top-left corner users can select the *Advanced* tab (see *Figure 14*). This tab allows the user to infrastructure options such as:

- Wall Clock Time
- Disk Size
- Number of cores
- Memory size

Despite the fact that all of them are mandatory, if default values have been provided during the application creation these fields will be automatically filled.



Wall Clock Time (min): 5

Disk Size (GB): 3.0

Cores: 4

Memory (GB): 2.0

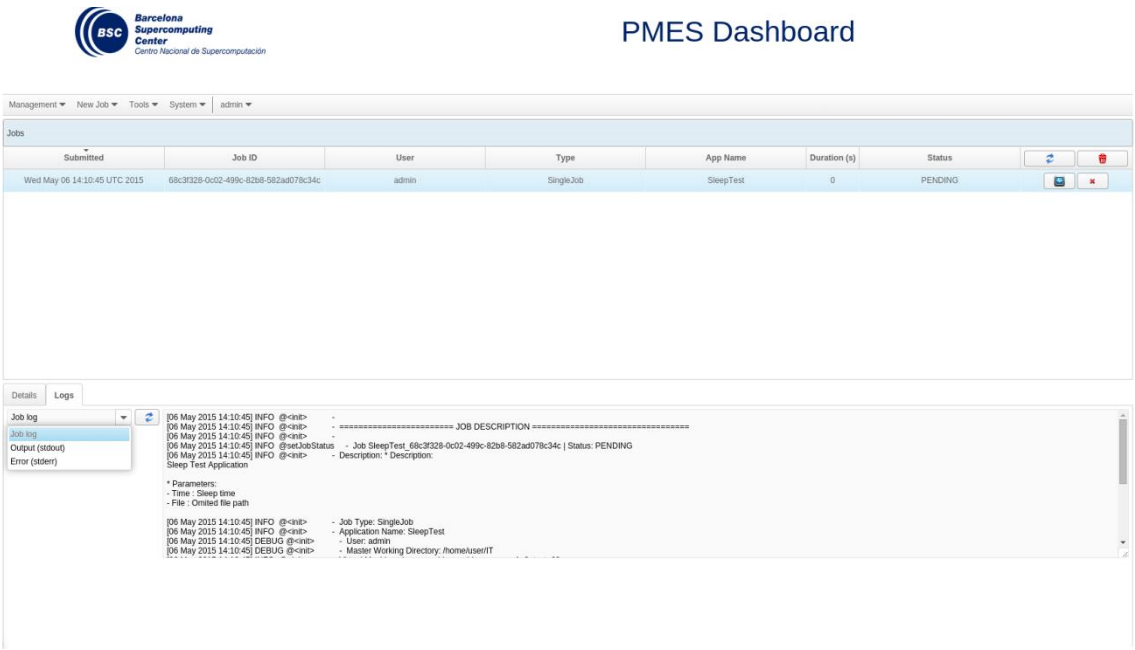
Figure 14: Job submission dialog – Advanced options

Once all the needed fields are filled, hitting the bottom *Submit* button will create the job.

3.5. Tools

3.5.1. Jobs monitoring

Once a job has been submitted the *PMES Framework* provides different information about its status. First of all a new row will appear in the main window showing the basic information about the job: submission time, job ID, user, job type, application name, duration, status and actions. By hitting the row, users can obtain more detailed information that will be shown in the tabs below (see *Figure 15*). The *Logs* tab allows users to check the job status by providing the job logs (generated by *PMES Framework*) and the standard output and error of the executed application.



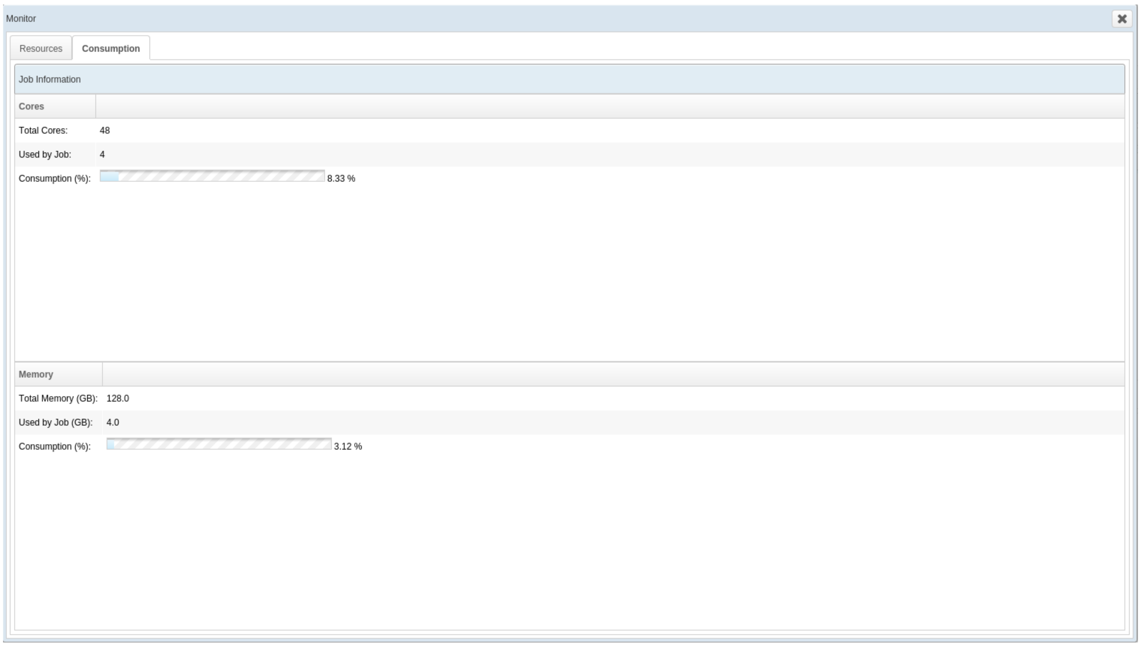
Submitted	Job ID	User	Type	App Name	Duration (s)	Status	
Wed May 06 14:10:45 UTC 2015	68c3f328-0c02-499c-82b8-582ad078c34c	admin	SingleJob	SleepTest	0	PENDING	

```
[06 May 2015 14:10:45] INFO  @<init>        -
[06 May 2015 14:10:45] INFO  @<init>        - ======================== JOB DESCRIPTION ================================
[06 May 2015 14:10:45] INFO  @<init>        -
[06 May 2015 14:10:45] INFO  @setJobStatus    - Job SleepTest_68c3f328-0c02-499c-82b8-582ad078c34c | Status: PENDING
[06 May 2015 14:10:45] INFO  @<init>        - Description: * Description:
Sleep Test Application

* Parameters:
- Time : Sleep time
- File : Omited file path

[06 May 2015 14:10:45] INFO  @<init>        - Job Type: SingleJob
[06 May 2015 14:10:45] INFO  @<init>        - Application Name: SleepTest
[06 May 2015 14:10:45] DEBUG @<init>         - User: admin
[06 May 2015 14:10:45] DEBUG @<init>         - Master Working Directory: /home/user/IT
```

Figure 15: Main window - Logs

Regarding the actions that users can execute on a job, the *PMES Dashboard* enables dynamically different actions depending on the job type and on the job status. When accessible, all of them are listed in the actions cell at right of each job description. Next, specific information about the actions is listed:

- **Monitor ():** *Only available on PENDING or RUNNING status*. Enables the job monitor window (see *Figure 16*) which provides information of the current resources consumed by the job.



Job Information

Cores

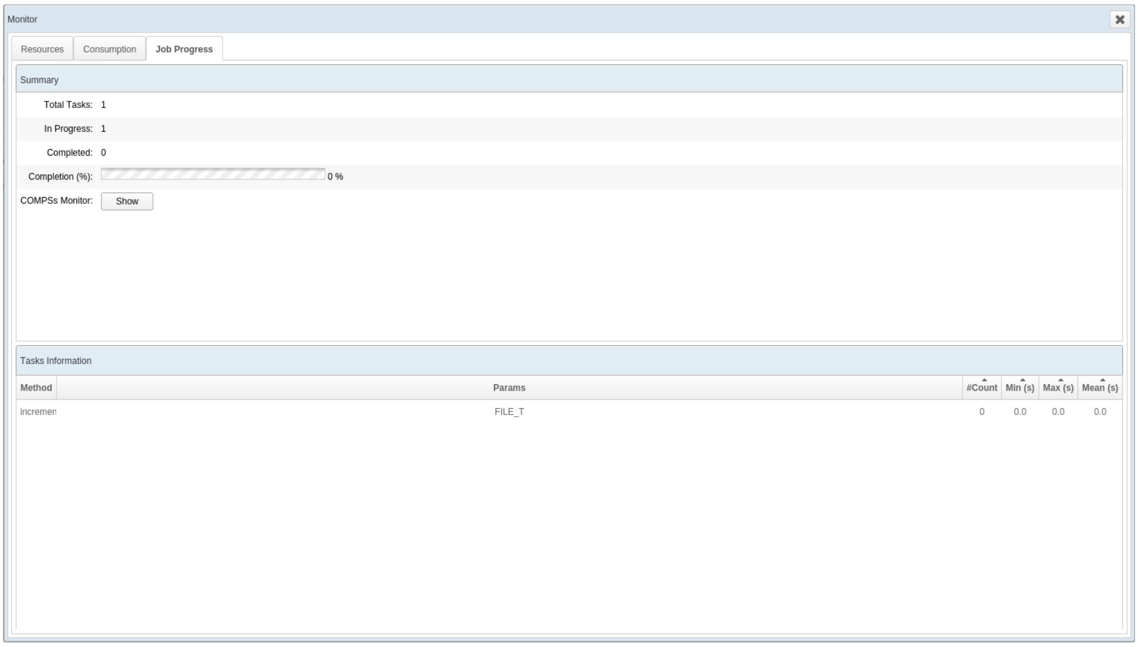
Total Cores:	48
Used by Job:	4
Consumption (%):	8.33 %

Memory

Total Memory (GB):	128.0
Used by Job (GB):	4.0
Consumption (%):	3.12 %

Figure 16: Monitoring Window – Consumption

For COMPSs Jobs a J*ob progress* tab is added providing specific information about the COMPSs tasks and a link to the COMPSs monitor (see *Figure 17*)



Summary

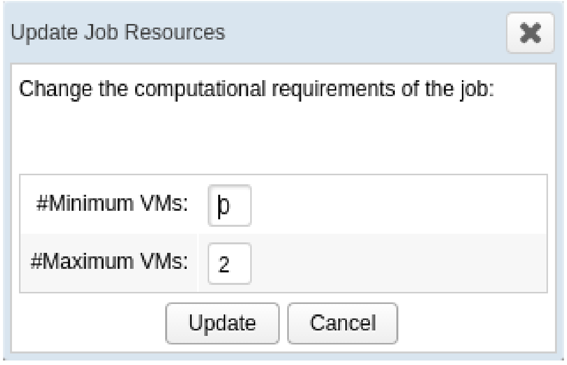
Total Tasks:	1
In Progress:	1
Completed:	0
Completion (%):	0 %
COMPSs Monitor:	Show

Tasks Information

Method	Params	#Count	Min (s)	Max (s)	Mean (s)
incremen	FILE_T	0	0.0	0.0	0.0

Figure 17: Monitoring Window – Job progress

- **Cancel (✖):** *Only available on PENDING or RUNNING status.* Cancels de job execution stopping the virtual machine and retrieving the log files.
- **Scale (⤢):** *Only available on PENDING or RUNNING status of COMPSs jobs.* As shown in *Figure 18*, allows users to modify the number of maximum and minimum resources used by COMPSs.



Update Job Resources ✖

Change the computational requirements of the job:

#Minimum VMs: 0

#Maximum VMs: 2

Figure 18: Monitoring Window – Consumption

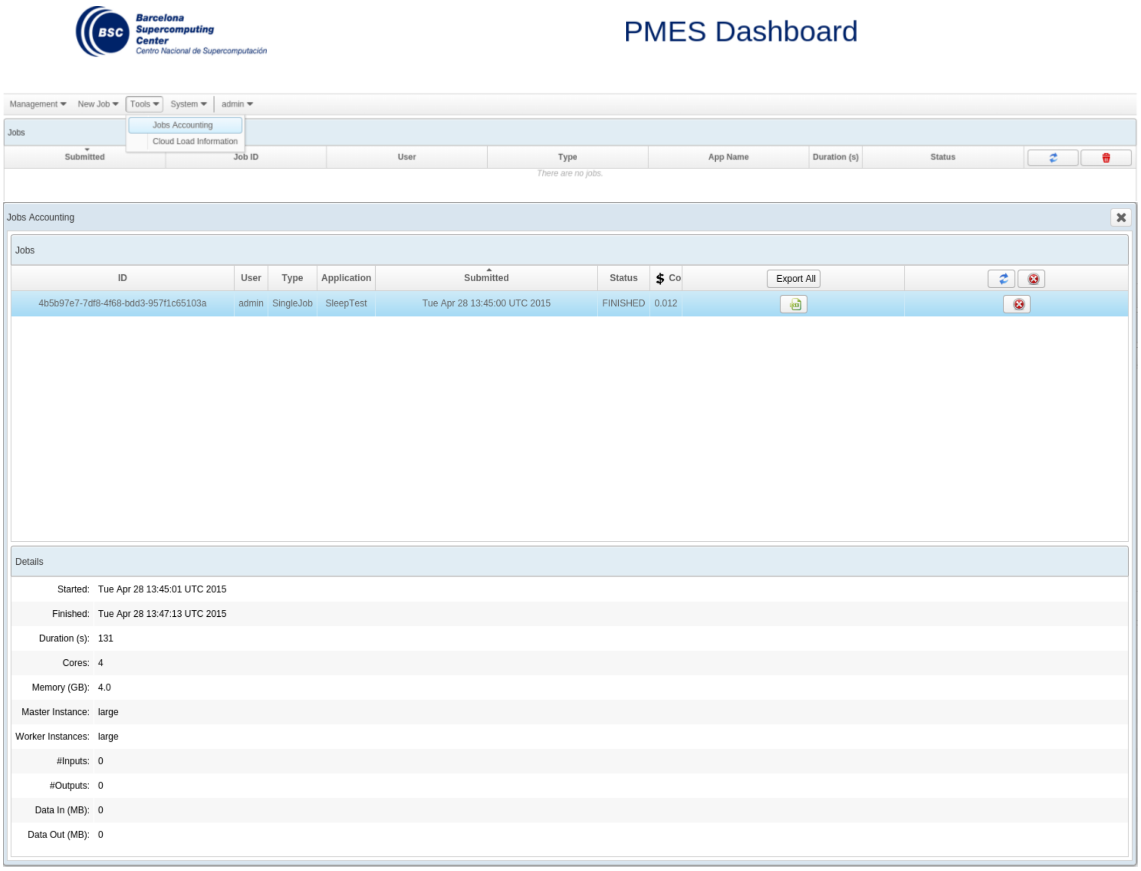
- **Edit (✎):** *Only available on FINISHED or FAILED status.* Once a job has finished, by hitting this button the job submission menu will appear and users will be able to resubmit the job with the same parameters or with modifications.
- **Delete (☒):** *Only available on FINISHED or FAILED status.* Erases the job from the view and the database. **Attention: This action cannot be reverted.**
- **Reload (⟳):** The job view reloads automatically (the ratio is fixed by the *PMES Framework* administrator) but users can manually force the reload by hitting this button.
- **Delete All (🗑):** Erases **all** FINISHED or FAILED jobs. **Attention: This action cannot be reverted.**

3.5.2. Jobs accounting

The jobs accounting table is accessible through *Tools > Jobs Accounting* (see *Figure 19*) and allows users to retrieve data from all their previously executed jobs (even if they failed).

Data can be exported in csv format by hitting the single export button (csv) at right of each job or hitting the all export button (Export All) at top of the table. Any job of the table can be erased by hitting the red button (⊗) at its right and the full table can be wiped by hitting the top right red button (⊗).

Administrators can configure whether to activate this feature or not through the *PMESConfig.xml* file.



Submitted	Job ID	User	Type	App Name	Duration (s)	Status

There are no jobs.

Jobs Accounting

Jobs

ID	User	Type	Application	Submitted	Status	$ Co
4b5b97e7-7df8-4f68-bdd3-957f1c65103a	admin	SingleJob	SleepTest	Tue Apr 28 13:45:00 UTC 2015	FINISHED	0.012

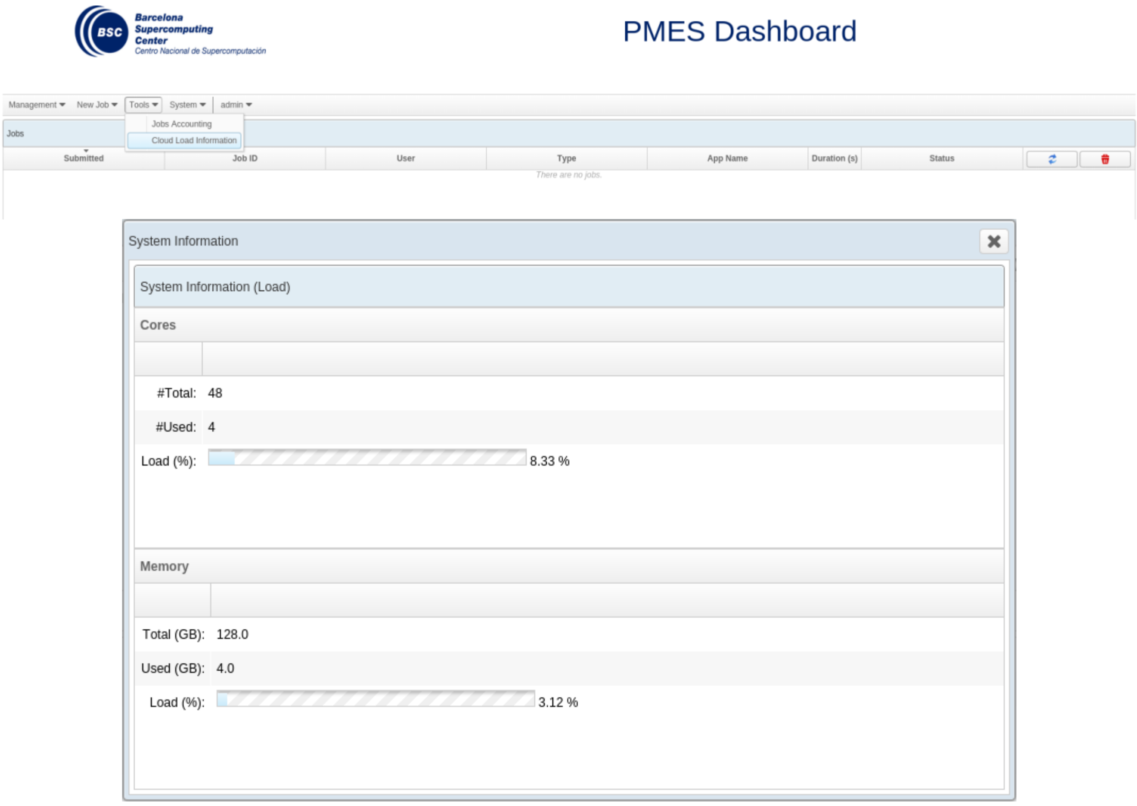
Details

Started: Tue Apr 28 13:45:01 UTC 2015
Finished: Tue Apr 28 13:47:13 UTC 2015
Duration (s): 131
Cores: 4
Memory (GB): 4.0
Master Instance: large
Worker Instances: large
#Inputs: 0
#Outputs: 0
Data In (MB): 0
Data Out (MB): 0

Figure 19: Jobs Accounting

3.5.3. Cloud load status

The cloud load status tool allows users to know the current load of the main cloud provider in terms of core consumption and memory consumption (see *Figure 20*). The cloud provider load status includes information about all the resources reserved by **ANY** user and it is meant to provide enough information for users to correctly size their jobs[4].



System Information

System Information (Load)

Cores

#Total: 48

#Used: 4

Load (%): 8.33 %

Memory

Total (GB): 128.0

Used (GB): 4.0

Load (%): 3.12 %

Figure 20: Cloud load monitor

3.6. Users and groups management

Administrator users can access to the users and groups management menu by clicking *Management* > *Users* (see *Figure 21*).

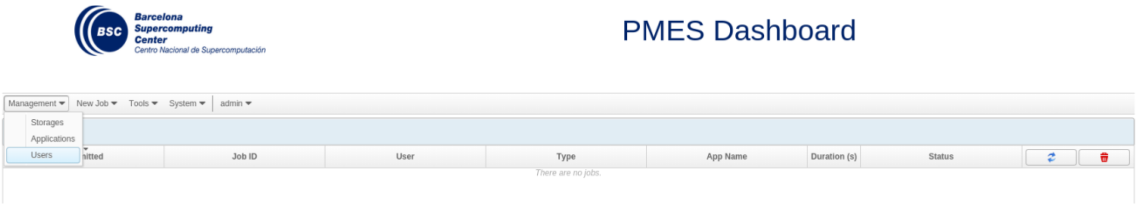
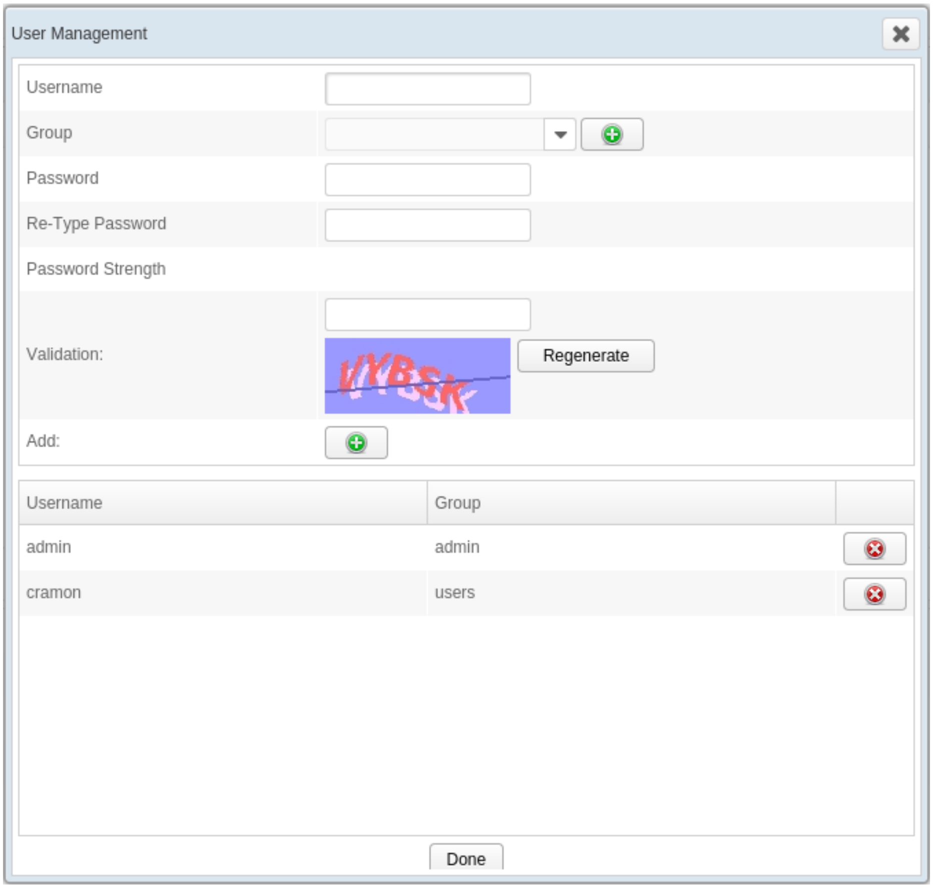


Figure 21: Users Management

[4] Sometimes a job remains in *Pending* state because there is not enough space in the main cloud provider to deploy it

Administrators can add users by fulfilling the upper form button (see *Figure 22*) and hitting the green button (⊕). The following information is required for a user creation:

- **Username:** Login name for the user.
- **Group:** Group for the user.
- **Password:** Default password for the user (we recommend new users to modify as soon as possible after being granted access to *PMES Dashboard*).
- **Captcha:** Captcha validation to avoid automatic user creation.



User Management ✖

Username	
Group	▾ ⊕
Password	
Re-Type Password	
Password Strength	
Validation:	Regenerate
Add:	⊕

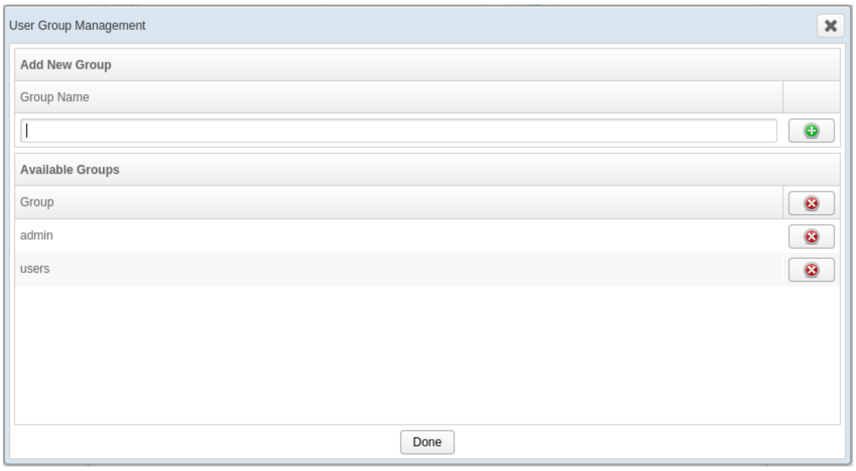
Username	Group	
admin	admin	⊗
cramon	users	⊗

Done

Figure 22: Users Management

Administrators can also eliminate users by hitting the red button (⊗) at right of each existing user (attention: admin user cannot be erased).

If needed, a group management window can be accessed by hitting the green button (⊕) at right of the group selection (see *Figure 23*). This window allows administrators to add new groups by providing a group name and hitting the green button (⊕), and to remove empty groups by hitting the red button (⊗) at right of each group (attention: admin group cannot be deleted).



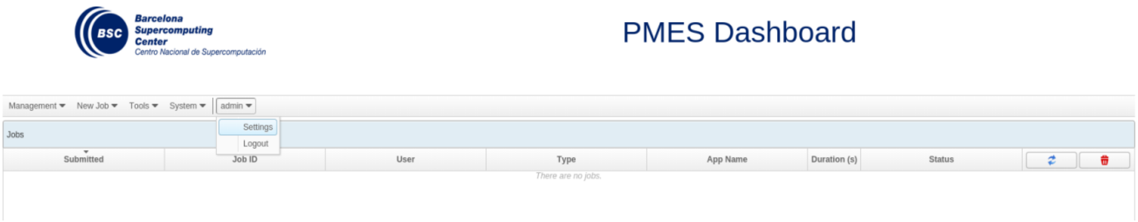
Group	⊗
admin	⊗
users	⊗

Figure 23: Groups Management

3.6. User settings

Every user can access and modify its personal information by accessing *USERNAME > Settings*.

Currently, users can only modify their password by providing their old password, fulfilling two times their new password and hitting the *Change Password* button (see *Figures 24 and 25*).



Submitted	Job ID	User	Type	App Name	Duration (s)	Status
			There are no jobs.			

Figure 24: Access to user settings

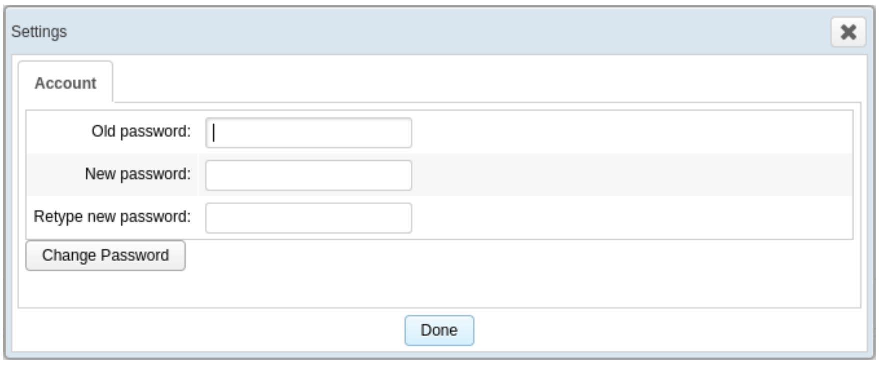


Figure 25: User settings

4. Information and support

Please find more details on the PMES Framework at:

http://www.bsc.es/computer-sciences/grid-computing/pmes-framework

For any further information please contact:

support-compss@bsc.es