ICT Call 4 - ICT for Energy Efficiency

ICT services and software tools enhanced with energy features

FIT4Green

Federated IT for a sustainable environment impact

Project Nº 249020

D6.3 - Report of 2nd Pilot Evaluation

Pilot Evaluation of Energy Control Plug-In inside single and federated Data Centres

Responsible:	Marco Di Girolamo (HPIS)
Contributors:	Giovanni Giuliani (HPIS), Juan-Carlos Lopez Egea, Alfonso Calpena, Fernando Almarza (GFI), Andrè Giesler (FZJ), Domenico Sannelli (ENI), Vasiliki Georgiadou (ALM), Andrey Somov (CN), Robert Basmadjian (UNI-PA), Thomas Schulze (UNI-MA)

Document Reference:	D6.3
Dissemination Level:	Public
Version:	6.0
Date:	03/02/2012

CONTRIBUTORS TABLE

DOCUMENT SECTION	AUTHOR(S)	REVIEWER(S)
Executive summary	HPIS	(ALL THE PARTNERS)
Introduction	HPIS	ALM
Traditional data centre testbed	ENI,GFI	UNI-MA
Supercomputing data centre testbed	FZJ	CN
Cloud computing data centre testbed	HPIS	UNI-PA
General conclusions	HPIS,ENI,GFI,FZJ	(ALL THE PARTNERS)

EXECUTIVE SUMMARY

FIT4Green aims at developing and deploying a set of holistic energy optimization policies for data centres and data centre federations. These policies must be applicable to and verifiable for different computing styles (traditional, super-computing and cloud). Within the project, WP6 assesses the effectiveness of those policies in saving energy. This work package does so by testing and evaluating the components implementing the global energy optimization policies in three different testbeds, each one representative of a specific computing style, for single or federated data centres, respectively.

In this deliverable WP6 reports on the second pilot cycle, from the setup of the testbeds to the chosen testing methodologies. The deliverable provides evidence of the claimed energy saving capabilities of the energy control plug-in for single and federated data centres having adopted either one among traditional, super-computing or cloud computing style. For each of the testbeds, the deliverable presents its environment and configuration, the applied testing methodology, the type of test workload, the achieved energy saving, concluding with usability evaluation and feedback results

The deliverable ends with a recap of the pilot cycle's results evaluation, highlighting the lessons learned, and outlining the feedbacks retrofitted to technology work packages, aimed at enhancing and improving *FIT4Green* policies effectiveness in the forthcoming third (and final) pilot cycle.

TABLE OF CONTENTS

TABLE OF FIGURES

TABLE OF TABLES

I. INTRODUCTION

FIT4Green aims at developing and deploying a set of global energy optimization policies for data centres, applicable to and verifiable for different computing styles (traditional, super-computing and cloud). Within the project, WP6 assesses the effectiveness of those policies in saving energy. This WP does so by testing and evaluating the components implementing the global energy optimization policies in three different testbeds, each one representative of a specific computing style, for single or federated data centres, respectively.

The present deliverable comes after two previous ones produced by this workpackage, D6.1 and D6.2:

D6.1	D6.1 had provided snapshots of all testbeds at FIT4Green's start-up time, outlining their characteristics and standard operational models, describing their available hardware and software equipment, and assessing the data centre automation environments which FIT4Green's plug-in components have to interact with.
D6.2	D6.2 was the final report of pilot cycle 1, executed in the timeframe between project months 11 and 15, and including single site data centre configuration testing and evaluation. D6.2: • detailed the actual single data centres configurations that, in tight cooperation with WP5, after the first release of *FIT4Green*'s energy control plug-in, were eventually picked to host the implemented components. • presented the testing methodologies elaborated for each of the testbeds, explaining how the test workloads had been generated, and how the testing strategies guaranteed the best coverage of energy optimization performance and usability for *FIT4Green*'s plug-in.

The present D6.3 deliverable will provide a full report of the second piloting cycle, which completes the plug-in's evaluation lifecycle for single site configurations, and brings in the federated site configurations for the three testbeds. As planned, the network testbed is not kept in the second pilot, being it meant as an instrument to improve the network layer modelling and behaviour, and therefore enhance the results obtained by the three actual testbeds in the federated cases.

In particular, the deliverable details the actual single and federated data centres configurations that, in tight cooperation with WP5, after the second release of FIT4Green's energy control plug-in, were eventually picked to host the implemented components. It presents the testing methodologies elaborated for each of the testbeds, highlighting differences and/or enhancements with respects to the pilot1, explaining how the test workloads have been generated, and how the testing strategies guarantee the best coverage of the evaluation of the energy optimization performance and usability of the FIT4Green's plug-in.

The present deliverable is organized as follows:

- Sections II-IV deliver a full account of the data centre testbed results; in detail:
  - Subsections II-IV.1-3 provide a full exposition of the pilot cycle preliminary setup, in terms of the data centre testbed environment and configuration, chosen testing methodology and test workload;

  - Subsections II-IV.4 provide for the different data centre testbeds the actual numerical results achieved throughout the testing campaign of the energy control plug-in;
  - In subsections II-IV.5 for the different data centre testbeds, conclusions are drawn about the observed plug-in performance, recommendations for improvements of the energy control plug-in, and acceptance/usability feedbacks from data centre operators where applicable.
- The closing section V consolidates the evaluation results yielded by the different testbeds into a single global picture.

The gained knowledge base will provide input to all the work packages for the work to do in the third development cycle:

- WP2 will use the inputs to update the target business scenarios for FIT4Green, and elaborate an amended proposition about energy-aware SLAs;
- WP3 will refine the energy consumption models of single and federated data centre components;
- WP4 will refine the algorithms and policies used by the energy optimization engine;
- WP5 will finally implement the changes asserted by WP3 and WP4, and will fix any implementation flaw noted during the evaluation.

The conclusions of this deliverable will be the starting point for the third pilot cycle, the final one scheduled inside the project.

II. TRADITIONAL DATA CENTRE TESTBED

II.1. Testbed environment and configuration

Eni's test trial environment is based on the Microsoft Unified Communications ISIM (MS Office Exchange 2010 integrated with MS Lync 2010 and other services) that offers mailing and instant messaging services to 4.000 users. This farm is an equivalent and small scaled architecture of a production environment (that hosts 45.000 users).

Users can access mailing services using their Smartphone, iPad, Blackberry, and via any MS Outlook 2010 client (see Figure 1).

MS Unified Communications Suite is hosted on a VMWare vSphere 4.1 infrastructure, running on 8 blade servers each configured as follows:

Configuration Item	Description
CPU type	Intel Xeon X5650 @2,66 GHz
N° of CPUs	2
N° of Cores	12
RAM Qty (GB)	24

Table 1 - Server configuration

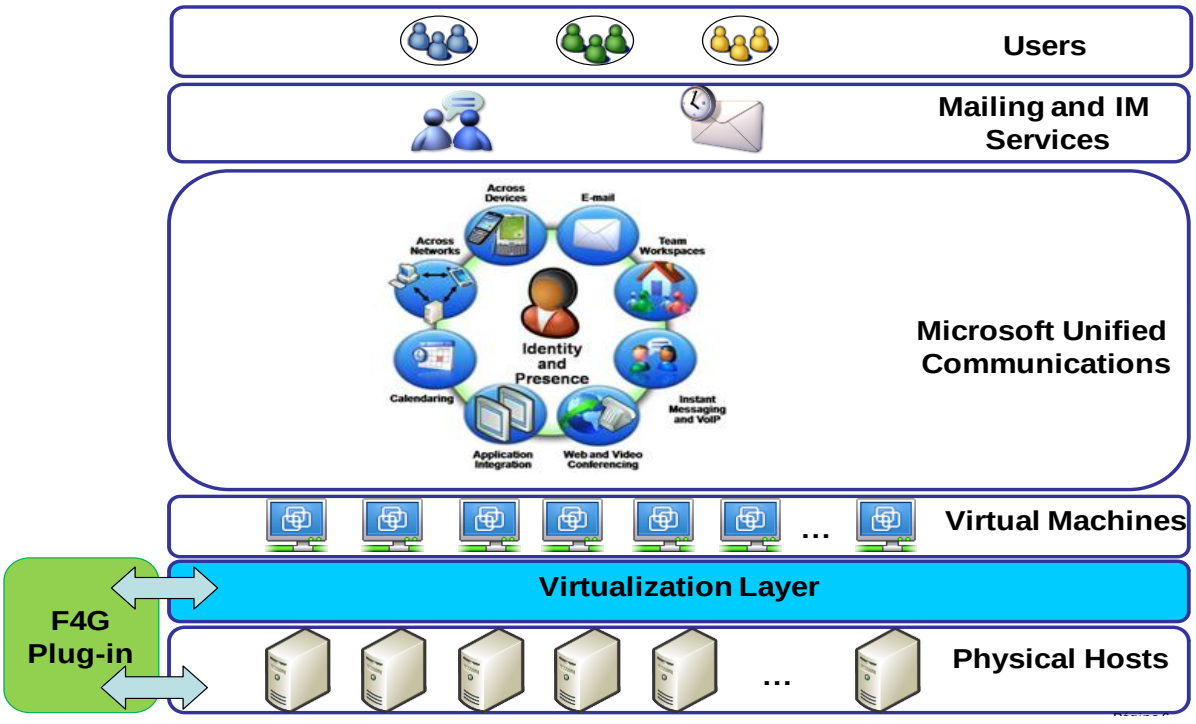


Figure 1 - ENI's virtual farm logical architecture

II.2. Testing methodology

Synthetic workload has been generated using Microsoft LoadSim tool, installed on a Windows 7 64 bit desktop, that allows simulating 4.000 users activity (sending emails, using instant messaging client, logon activity, sending "heavy" attachment via email, etc.). The image below is a screen-shot of the LoadSim user interface that allows controlling each "virtual user" and "MS Exchange 2010 virtual servers":

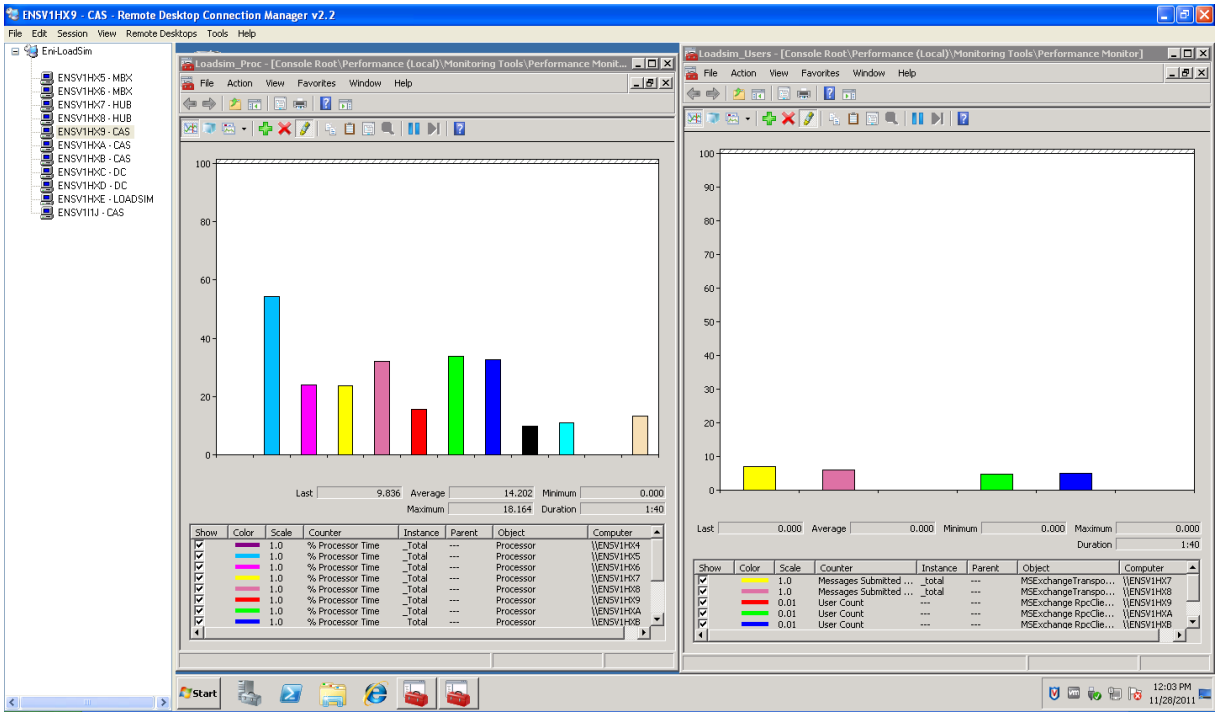


Figure 2 - The user interface of the Microsoft LoadSim tool

The *FIT4Green* plug-in has been installed on a Linux virtual server on a dedicated VMWare lab cluster. It communicates with physical servers through HP ILO (HP Integrated Lights-Out) and Virtual Center (as described in D6.2 it is responsible for all operational activities on physical hosts), which is the native management interface providing both the detection of energy metrics and the possibility of managing servers for basic activities, as e.g. power ON/OFF or installing an operating system. In addition the ILO management interface allows to store all energy metrics for physical servers that are part of VMWare Clusters. Those in term can be automatically downloaded via ftp using SSH scripts. Therefore it is possible to get energy data from two different sources.

In correspondence with the provided data the Fit4green plug-in analyzes the workload of the virtual farm and, whenever an opportunity for saving energy arises, it will perform consolidation of virtual servers on fewer hosts and then turn off unused physical servers. At the moment of unanticipated demand for additional computational resources, the needed hosts will be restarted.

II.3. Test workload

Test workloads were generated synthetically using MS LoadSim tools and it was similar to ENI's Exchange Production Systems (see the paragraph below).

Users do several heterogeneous tasks (instant messaging, sending outlook calendar meetings, read/send emails, etc.) during the business hours. We have condensed the 9 business hours activities (9-12, 1 hour lunch break, 13-18) to a shorter time interval of 5 hours. In this way we have more time to repeat tests and to do a "fine tuning" of the synthetic workload during test environment setup, to make it more similar to real case.

In our observations of the real production systems we have investigated that sending/receiving instant messages and email are by far the most frequent executed tasks (see table below). As a result, 42.285 mails were sent and 239.762 instant messages were processed within a single 5 hour test. The table below show all details about the executed and completed tasks by synthetic users during each test session:

Task Name	Count
BrowseCalendarTask	15.218
BrowseContactsTask	12.781
BrowseTasksTask	3.891
CreateContactTask	1.365
CreateFolderTask	132
CreateTaskTask	1.367
DeleteMailTask	543
DownloadOabTask	642
EditSmartFoldersTask	672
LogoffTask	2.069
MakeAppointmentTask	2.654
PostFreeBusyTask	2.502
ReadAndProcessMessagesTask	239.762
RequestMeetingTask	6.059
SendMailTask	42.285

Table 2 - Executed and completed tasks by synthetic users during each test session in Traditional DC case

The image below shows realized workload (note: MS Exchange 2010 applications are "RAM intensive", because these kind of applications need to pre-allocate a quantity of RAM to run).

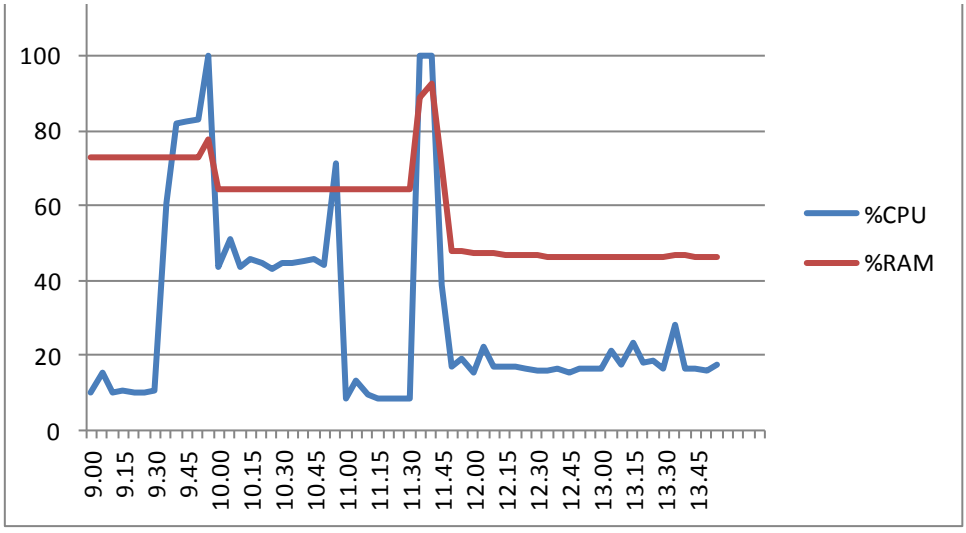


Figure 3 - Realized workload at ENI

It is characterized by an initial peak due to "boot-storm logon" and from a resource usage decrease during lunch hour and end of business day.

II.4. Numerical results

II.4.1. Workload Test

In a first step we have captured the energy consumption signature of the previously described test workload. Figure 4 - Power consumption of ENI's Servers depicts the measurements for a configuration of two clusters:

- MOEX that contains enbdc104, enbdc105, enbdh109, enbdh10a, enbdh10b and enbdh10c.
- MOEX2 that contains enbdc106 and enbdc107.

These measurements were taken for a full length execution of 8 hours, with a 1 hour break simulating lunch break of real users.

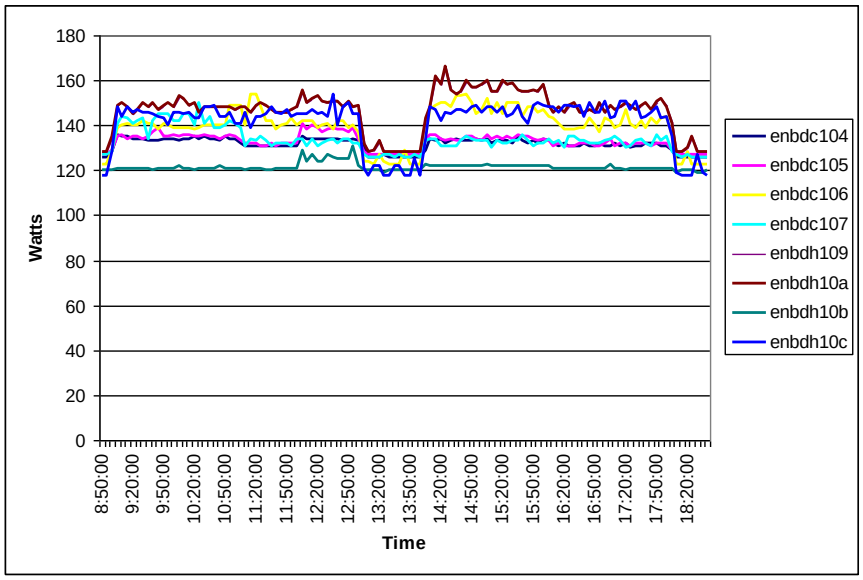


Figure 4 - Power consumption of ENI's Servers

Further tests are executed with compact version of the workload, running a 5 hours test with a lunch break of 1 hour

II.4.2. Single cluster case

Within the single cluster case, the servers are contained only inside the MOEX cluster (MOEX contains enbdc104, enbdc105, enbdc106, enbdc107, enbdh109, enbdh10a, enbdh10b and enbdh10c hosts).

II.4.2.a. Measurements without *FIT4Green* plug-in

In order to get a benchmark for the FIT4Green plug-in we have executed the single cluster case without the plug-in, in a first step Figure 5 shows the consumption of the servers with the execution of the workload. We can see a variation on consumption for every server, but the minimum for the server is around 118 Watts and the maximum peak is at 138 Watts.

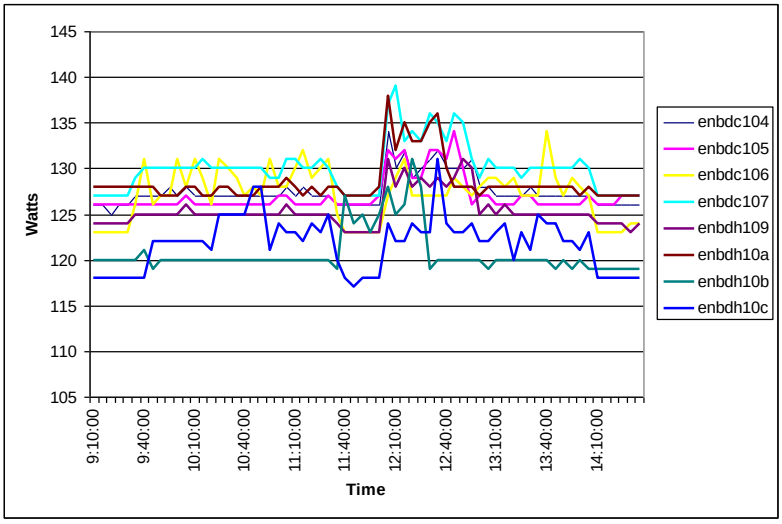


Figure 5 - Average power consumption per server without FIT4Green

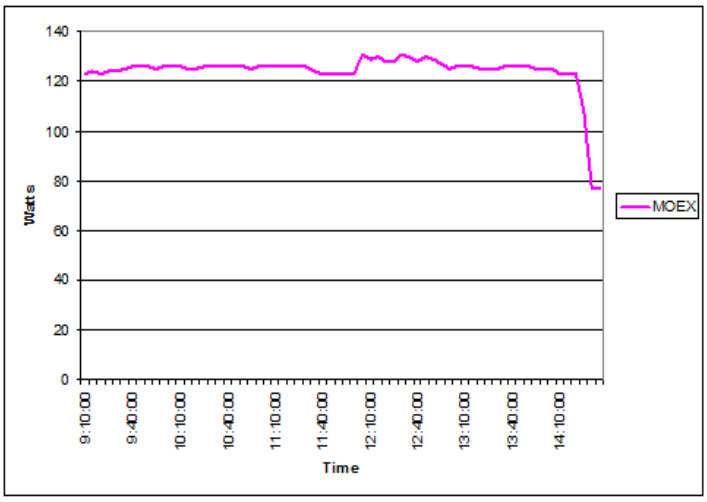


Figure 6 - Power consumption of ENI's Servers without

The average consumption of the servers is around 120 Watts with a little decreasing at lunch break and an increase after lunch (however never above 135 Watts).

II.4.2.b. Measurements with *FIT4Green* plug-in

The execution of the FIT4Green plug-in results in two optimizations:

- The first is at 9:50, the reallocation of the VMs from the servers' results in powering off enbdc106 and enbdh10c.
- The second is at 10:20 resulting in the shutting down of enbdc107.

The remaining servers raise their consumptions to around 130 Watts and 140 Watts. This is most obvious for endbh10a where the consolidation of VMs is the highest. However, this uprising on the consumption per server is the better choice when we look at the average consumption and the total energy on the cluster, as we will see in the next paragraphs.

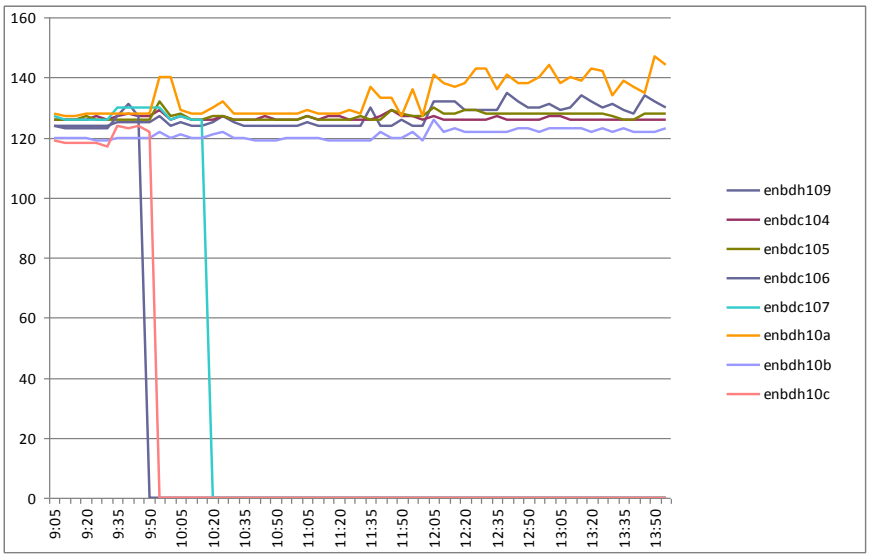


Figure 7 - Average power consumption of ENI's Servers with FIT4Green

At the beginning, the average consumption of all servers is around 120-130 Watts. By the time of the first optimization the consumption decreases to around 90 Watts. When the second optimization is finished the consumption is around 80 Watts.

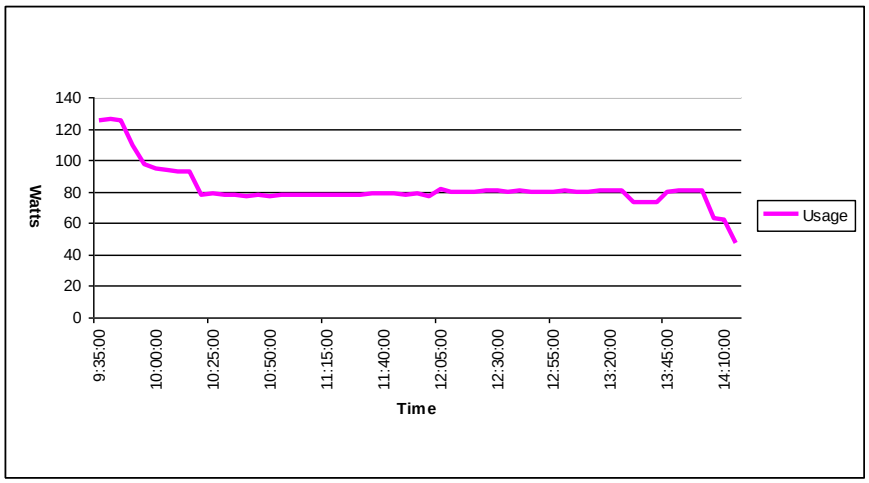


Figure 8 - Average power consumption per server with FIT4Green

II.4.2.c. Results for the single cluster tests

Figure 9 shows the energy usage of the test bed prior to the optimizations, the usage is around 20200 Watts, with a peak of usage between 20500 - 21000

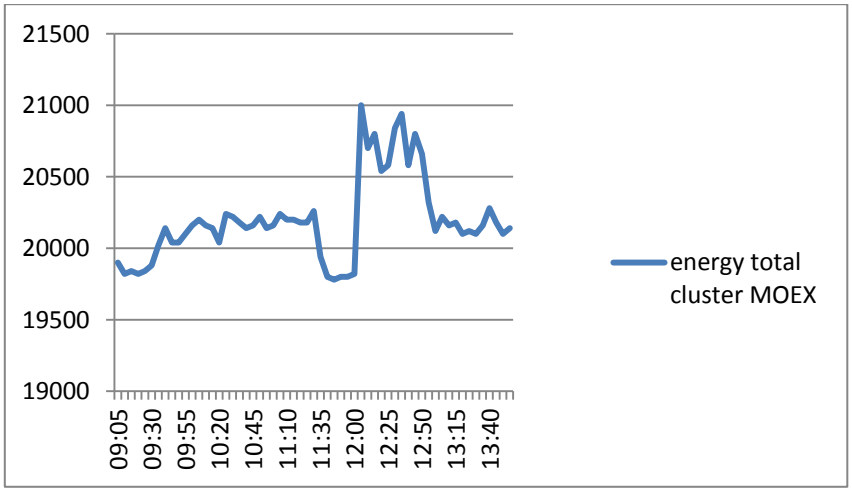


Figure 9 - Energy usage of the testbed without FIT4Green

If we look at Figure 10 we can observe the behaviour of the optimizations applied to the MOEX cluster. We start with a consumption around 20000, that is in line with the consumption obtained without the FIT4Green plug-in. When the first optimization is performed around 9:50 2 servers are powered off. Here, we can see a decrease in energy consumption to around 15000 Watts. Thirty minutes later when the second optimization is finished around 10:20 the energy consumption again decreases to around 12500 and maintained over time. Therefore we obtain a total a savings of around 7500 Watts. In terms of percentage this saving is around 30% that is totally in line with the results obtained in phase 1.

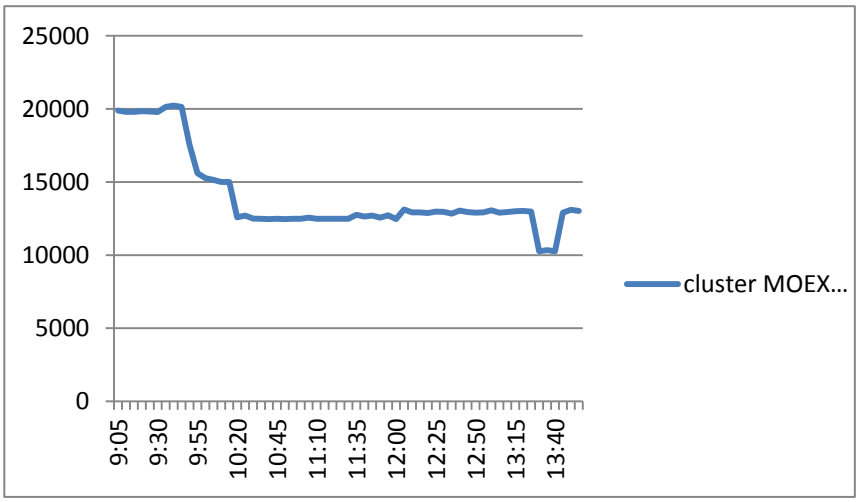


Figure 10 - Energy usage of the testbed with FIT4Green

One interesting remark is that the workload without the Fit4Green plug-in isn't working with only 5 servers even in his lower peak. However with the Fit4Green plug-in the workload is working with only 5 servers which is achieve by moving the VMs within the execution of the workload performed by the Fit4Green plug-in.

II.4.3. Federated clusters case Workload definition

In this case we have a MS Unified Communications Architecture Shared and Load Balanced between two different DCs. For Capacity reasons we have shared only part of a complete exchange system (servers that manage mailboxes service components) and all other services (e.g. domain controllers) are running only on one of them.

In these test we generate a synthetic workload that simulate eni's users activities during a business day, but a sw load balancer will distributes users requests across four different

mailbox servers (vmware virtual machines) running on two different VMWare Cluster hosted on two different DCs.

Datacenter Operator could manually turn-off one (or more) mailbox server and manually re-direct users request on the other mailbox servers; this don't cause downtime errors.

The image below shows a logical view of the overall architecture:

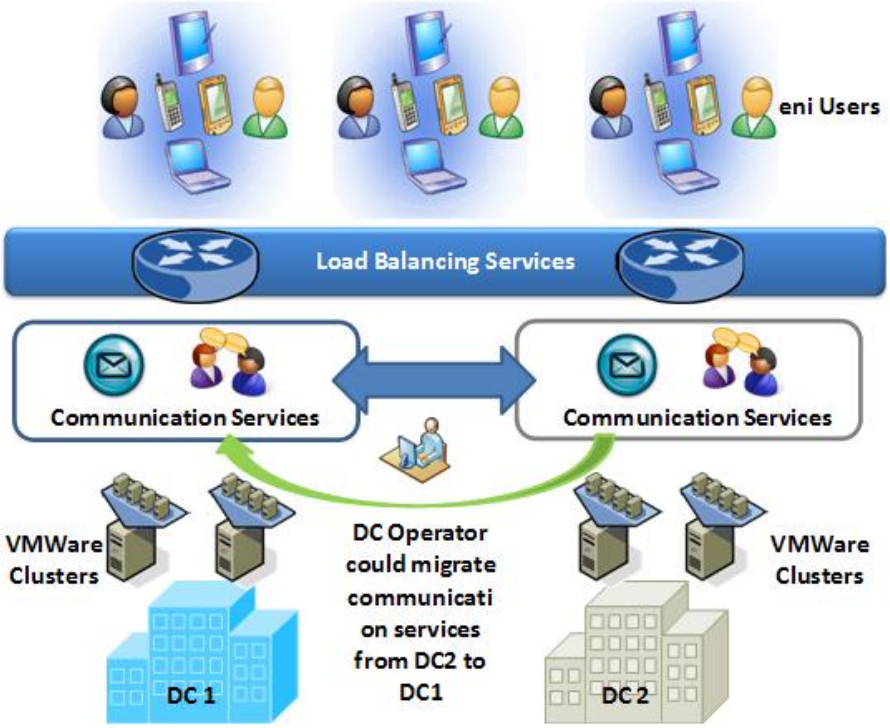


Figure 11 - Federated clusters case logical architecture

II.4.4. Federated clusters case static Workload

Federated cluster case, contains 2 clusters:

- Cluster MOEX contains 8 servers, enbdc104, enbdc105, enbdc106, enbdc107, enbdh109, enbdh10a, enbdh10b and enbdh10c.
- Cluster MOEX2 contains 2 servers, enbdh10e, enbdh10f.

The workload is executed on the two data centres MOEX and MOEX2 and it's not moved or redistributed across the data centres, so it is called static workload. In the chapter II.4.5 we will move the workload from MOEX2 to MOEX during the execution and we will analyze the results from the execution of this dynamically moved workload.

II.4.4.a. Measurements without *FIT4Green* plug-in

Again to have a benchmark for the energy signature for our test workload we have executed tests without the FIT4Green plug-in. Fig 11. shows the consumption of the servers inside the MOEX Cluster with the execution of the static workload. We can see a variation on consumption for every server, but the minimum for the server is around 118 Watts and the maximum peak is at 138 Watts. Furthermore, we can clearly see the influence of the lunch break on enbdh10a where most of the VMs are located.

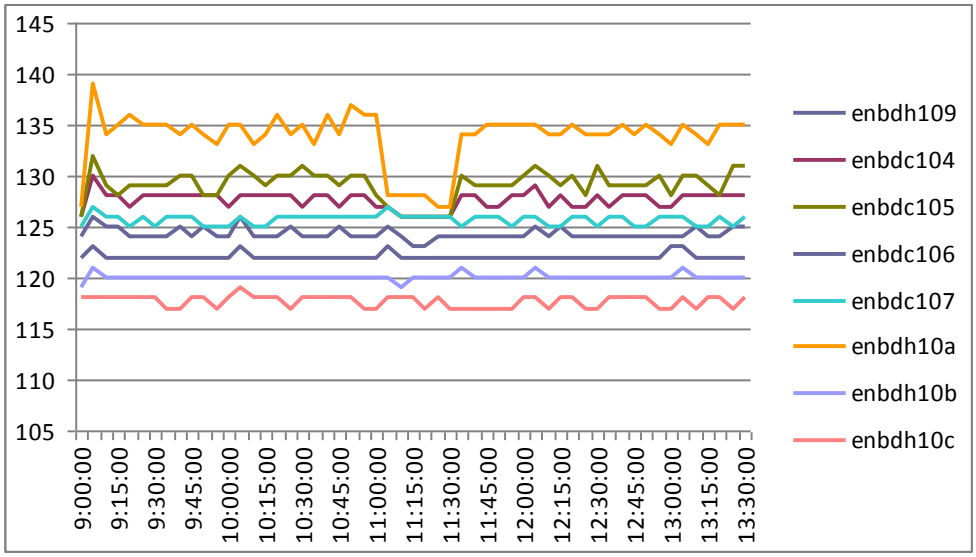


Figure 12 - Power consumption of Cluster MOEX using static workload without fit4green

The average consumption of the servers in MOEX Cluster with static workload is around 125 Watts as is depicted in Figure 13.

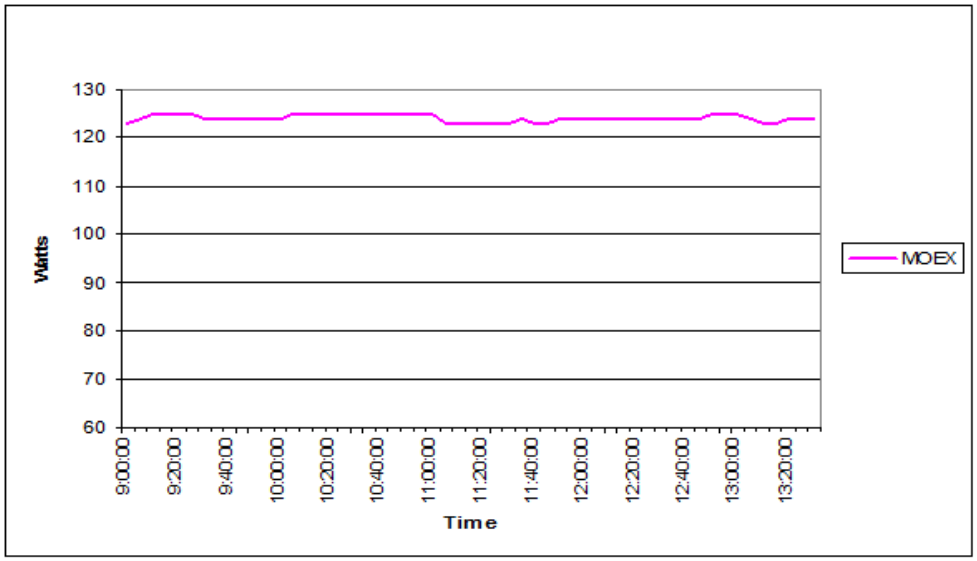


Figure 13 - Average power consumption per server on cluster MOEX using static workload without FIT4Green

Figure 14 shows the consumption of the servers inside MOEX2 Cluster with the execution of the static workload, we can see that the minimum for the servers is around 126 Watts at lunch break and the maximum peak is at 133 Watts

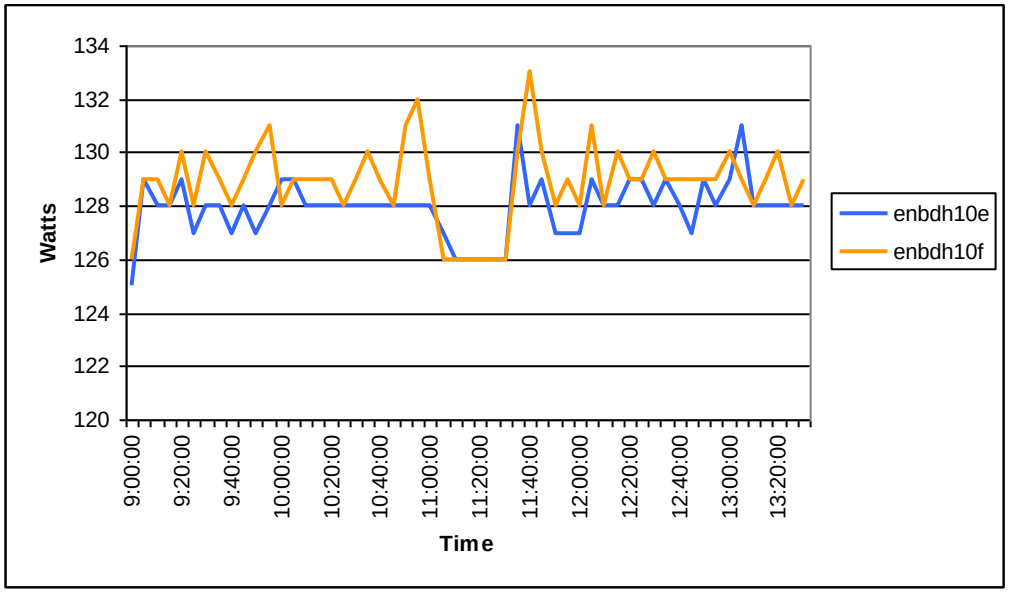


Figure 14 – Power consumption of Cluster MOEX2 using static workload without fit4green

The average consumption within the MOEX2 Cluster is around 128Watts

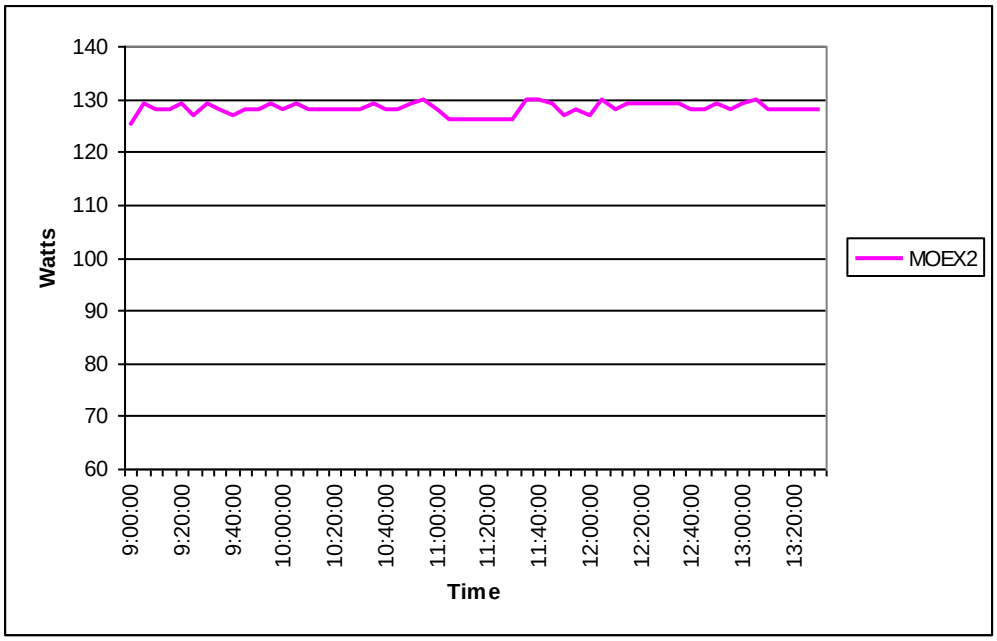


Figure 15 - Average power consumption per server on cluster MOEX using static workload without FIT4Green

II.4.4.b. Measurements with *FIT4Green* plug-in

The execution of FIT4Green plug-in with the static workload results in two optimizations:

- The first optimization is at 14:20 which results in the reallocation of the VMs from the servers enbdh109, enbdh10a and enbdh10b and a powering off action.
- The second is at 14:40 resulting in the shutdown of enbdc106.

The consumptions of the remaining 4 servers raise to around 130 Watts.

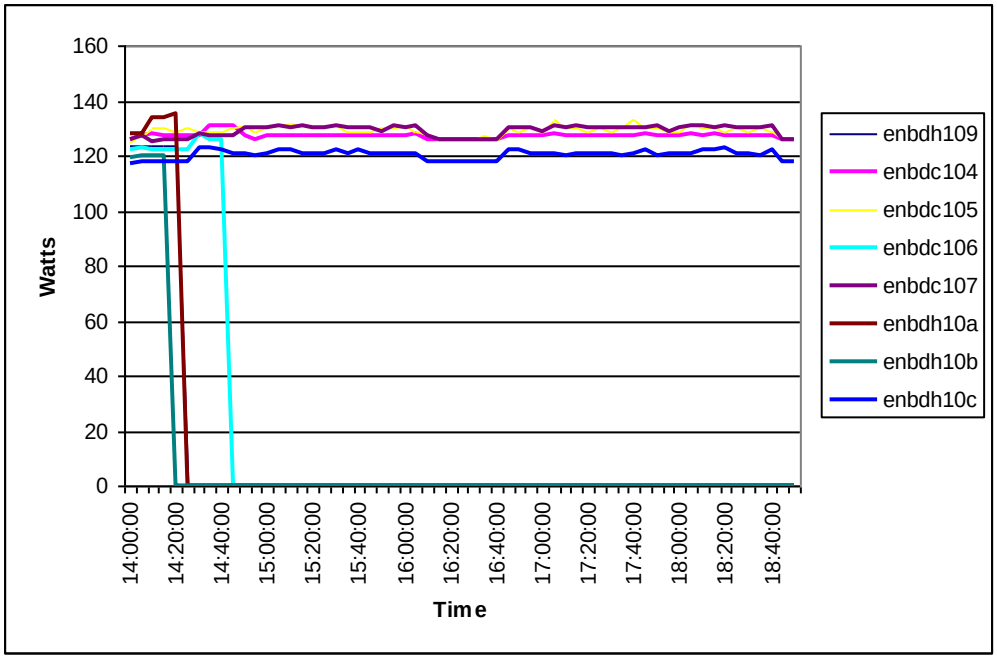


Figure 16 - Power consumption of Cluster MOEX using static workload with fit4green

At the beginning, the average consumption of the servers is around 120-130 Watts then when the first optimization is performed the consumption decreases up to 70 Watts. When the second optimization is finished the average consumption drops to around 60 Watts.

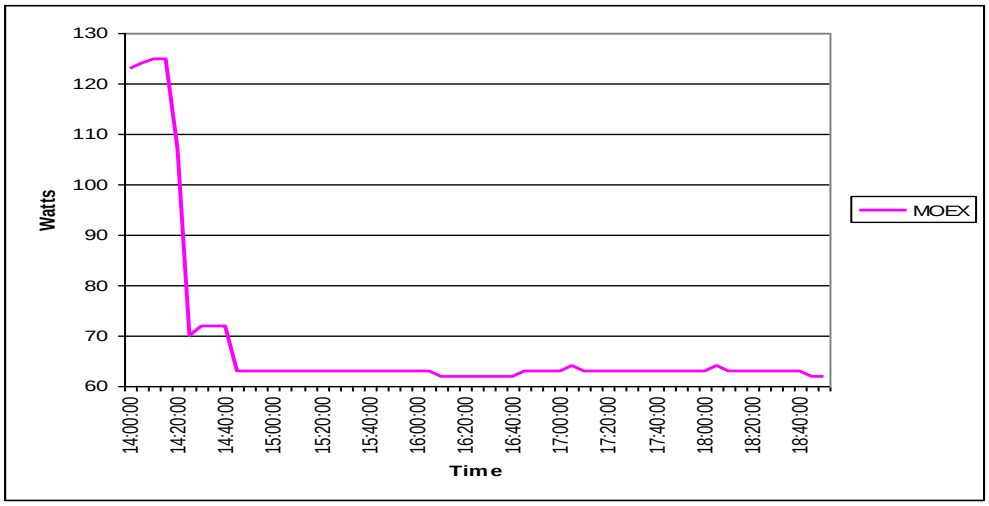


Figure 17 - Average power consumption per server on cluster MOEX using static workload with FIT4Green

MOEX2 is composed of 2 servers, the execution of FIT4Green plug-in with the static workload results in powering off 1 of the servers by the time of the first optimization at 14:20

The consumptions of the remaining server raise a little bit to around 130 Watts with a peak of 140 Watts.

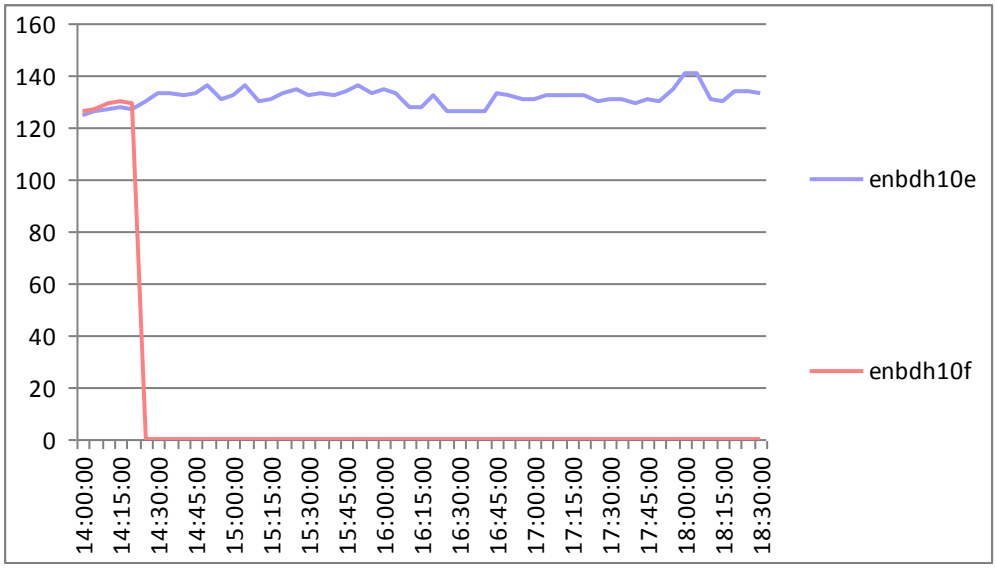


Figure 18 - Power consumption of Cluster MOEX2 using static workload with fit4green

The average of the consumption with the first optimization drops to around 60 – 70 Watts caused by the powering off the half of the servers.

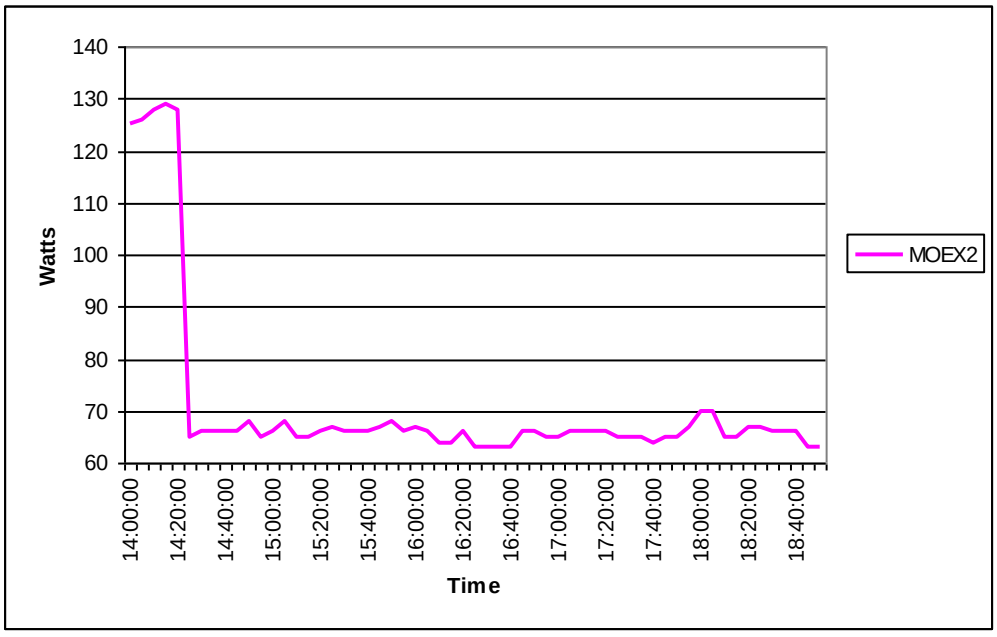


Figure 19 - Average power consumption per server on cluster MOEX2 using static workload with FIT4Green

II.4.4.c. Results of the federated tests with static workload

Figure 20 shows the energy usage of the test bed prior to the optimizations, the usage is around 25200 Watts, with a low peak of usage of 24800 Watts and a high peak of 25450 Watts.

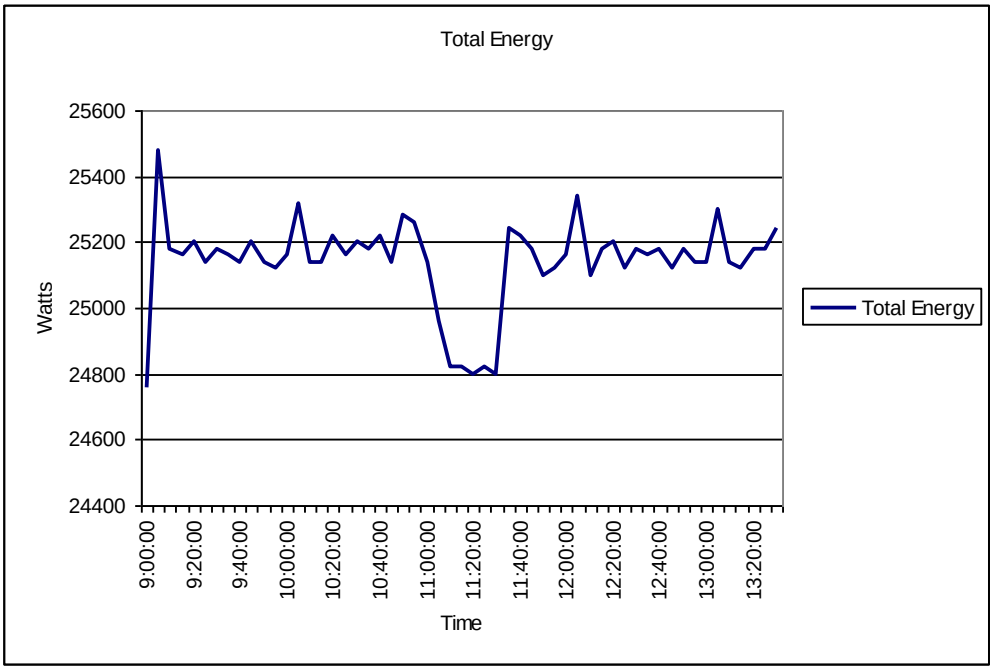


Figure 20 - Energy usage of the test bed without fit4green using static workload

Figure 21 depicts the energy consumption of the MOEX and MOEX2 clusters and the total consumption of the test bed

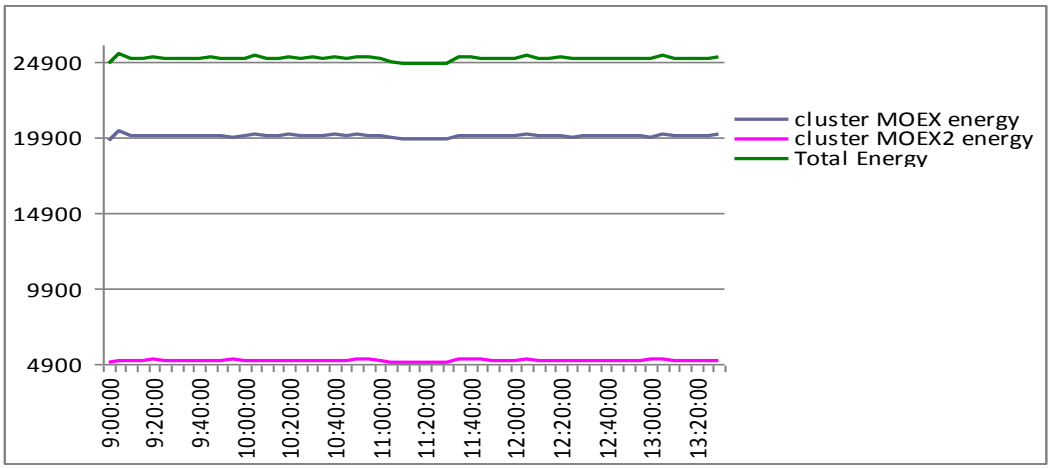


Figure 21 - Energy usage of the Clusters and the test bed without fit4green using static workload

We can observe the behaviour of the optimizations applied to the test bed with MOEX and MOEX2 clusters. We start with a consumption of around 25000. When the first optimization is performed at 14:20 the result is powering off 3 servers on MOEX and 1 server on MOEX2. With this optimizations the energy consumption drops to 15000 Watts, 20 minutes later when the second optimization is finished the energy consumption is again decreased to 13000 and maintained over time. Therefore we obtain a total a savings of around 12000 Watts. In terms of percentage this saving is around 45%.

This is obtained because of the distribution of the workload across the clusters allowing to powering off 4 servers on MOEX and 1 server on MOEX2.

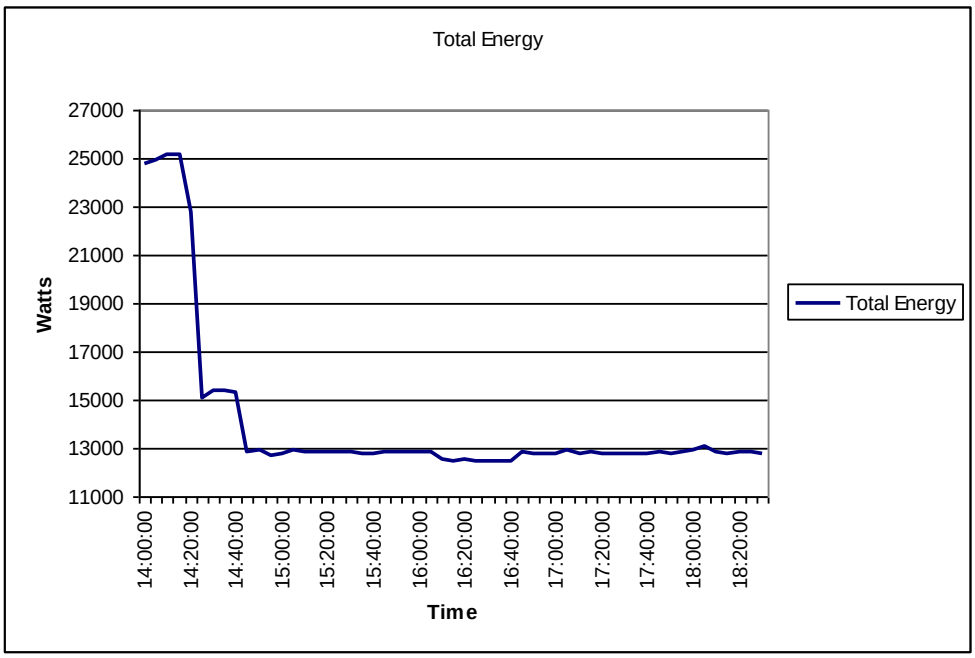


Figure 22 - Energy usage of the test bed with fit4green using static workload

Figure 23 depicts the energy consumption of the MOEX and MOEX2 clusters and the total consumption of the test bed.

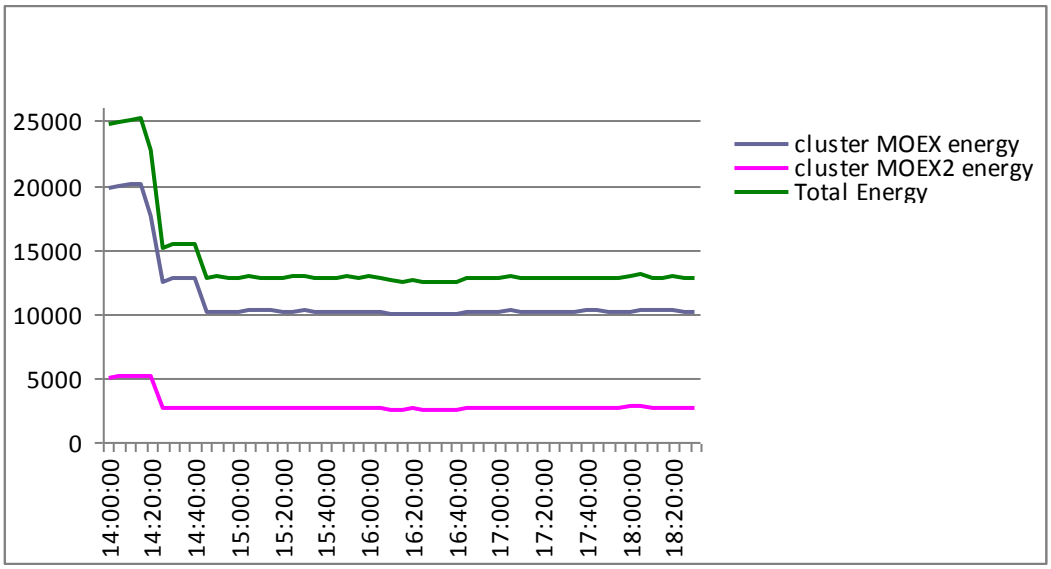


Figure 23 - Energy usage of the Clusters and the test bed with fit4green using static workload

II.4.5. Federated cluster case dynamic Workload

The configuration of the federation is the same as the one used for the execution of the static workload with 2 clusters, MOEX containing 8 servers and MOEX2 containing 2 servers.

In this section will show and analyze the results of the execution of the workload, including a dynamic workload that is moved around 1 hour after the beginning of the execution from the data centre MOEX2 to the data centre MOEX.

The next table shows the configurations of the two data centres to obtain the CUE values used for the execution of the tests.

Data Centres	Energy sources			PUE	CUE
MOEX	Coal 80% (0.910)	Oil 20% (0.610)		2.0	1.7
MOEX2	Oil 20% (0.610)	Hydro 40% (0.0)	Nuclear 40% (0.0161)	2.0	0.25688

Table 3 - Energy parameters of data centres

II.4.5.a. Measurements without *FIT4Green* plug-in

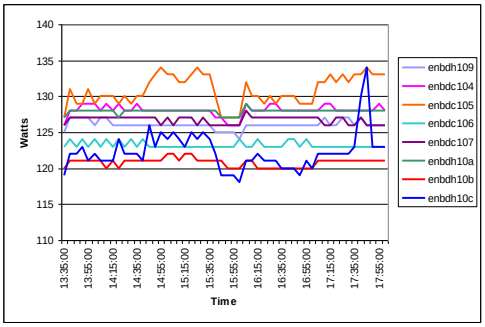
The benchmark is again being taken without the FIT4Green plug-in. Figure 24 shows the consumption of the servers inside the MOEX and MOEX2 clusters with the execution of the dynamic workload. We can see the variation on consumption for every server.

On MOEX the minimum consumption for the servers before moving the workload is around 120 Watts and the maximum peak is at 131 Watts. When the first hour of the execution of the workload is completed the workload is moved to MOEX data centre raising the consumption on MOEX to 134 Watts and lowering the consumption on MOEX. After the lunch break, the workload is executed on both data centres MOEX and MOEX2 and finally one hour after the lunch break the workload is moved again to MOEX data centre, setting consumption's peak at 134 Watts again.

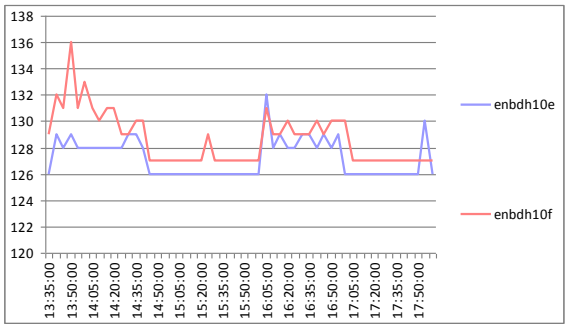
On MOEX2 the minimum consumption for the servers before moving the workload is around 129 Watts and maximum peak is at 136 Watts, after moving the workload to MOEX the consumption decrease to a minimum of 126 Watts and maximum of 129 Watts.

For the average consumption of the MOEX data centre the increasing of the consumption is noticeable when the workload is moved at the first hour of execution. Also the lunch break is visible and the increasing after the movement of the workload after the lunch break.

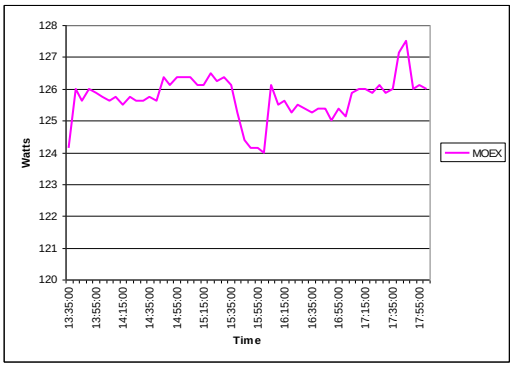
The average consumption on MOEX2 data centre shows also the decreasing of the consumption when the workload is moved. Furthermore it is clearly visible that the energy consumption after the lunch break is increasing when the workload is started again on MOEX2 and is in the following decreasing when the workload is moved to MOEX.



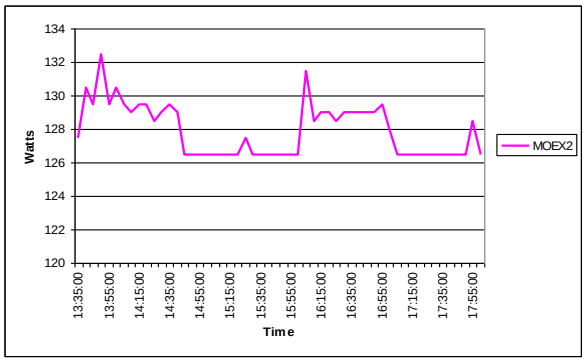
Power consumption of MOEX without fit4green



Power consumption of MOEX2 without fit4green



Average Power consumption of MOEX without fit4green



Average Power consumption of MOEX without fit4green

Figure 24 – Measurements of MOEX, MOEX2 without fit4green

II.4.5.b. Measurements with FIT4Green plug-in

The SLA constraint values are configured to get a maximum optimization result within the limits of the policies of ENI. As a note we would like to mention that depending on the time the optimizer executes and the exact location of the virtual machines across the servers, the result of the optimization might vary. That means in a concrete context that sometimes one more server gets powered off in our test trails leading to a total of four servers within the MOEX data centre.

On the MOEX cluster the execution of the FIT4Green plug-in results in a movement of the VMs from three of the servers on MOEX to the rest of the servers and a powering off of these three servers.

When the execution of the workload begins the consumption of the servers are around 120 and 130 Watts, and when the optimizations finish the consumption raises to around 130 and 140 Watts for the remaining servers. When the powering off actions ends, the FIT4Green plug-in do several moving of the VMs between the left powered on servers that maintains the Data Centre in a stable state and distributing the consumption between the remaining servers.

On MOEX2 the execution of the FIT4Green plug-in results in movement of VMs from endbh10f to endbh10e and a powering off of endbh10f.

This doesn't affect the consumption of the remaining server because the VMs on MOEX2 don't stress the physical server to its limits as it is done on MOEX.

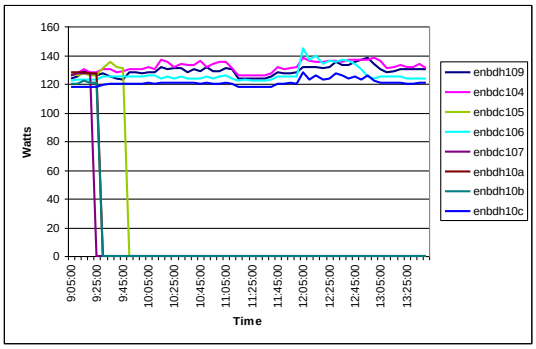
Looking at the average consumption on MOEX, it can be seen that the initial consumption, before the optimization takes place, is around 125 Watts, decreasing to 80 Watts when as a result of the optimization the three servers are powered off and further below 60 Watts in the cases when the fourth server is powered off.

The average consumption on MOEX2 is around 130 Watts before the optimizations and decreases to around 65 Watts after the optimization.

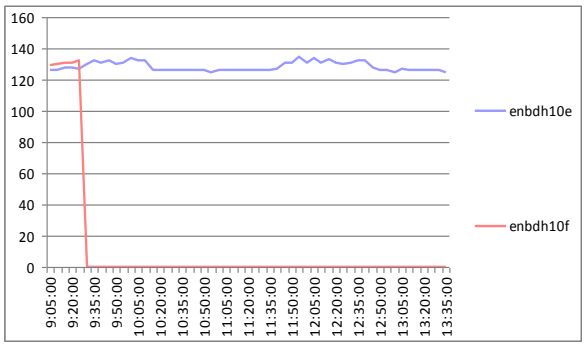
These optimization actions are the same for all the cases that we have tested in our Traditional test bed. This is because moving virtual machines inter data centre is not allowed, so even with different PUE values for each data centre we cannot move the actual virtual machines where the workload is executed to the best PUE data centre and the workload doesn't create new virtual machines, that would be allocated in to the best data centre, during the execution of the workload resulting in the fact that the differences when we change PUE or CUE values or when we seek to optimize for power or for emissions are minimal or nonexistent.

After these round of tests described above. In the next section we can found the values for one execution of the workload including allocation of new virtual machines.

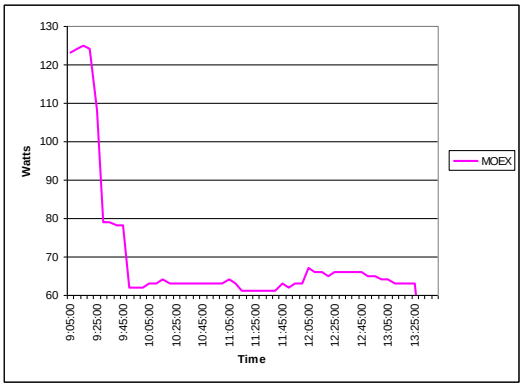
Here both the case, where 4 servers and the case where 3 servers are powered off is presented.



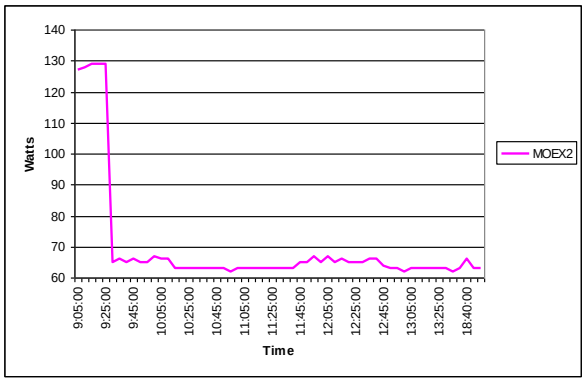
Power consumption of Cluster MOEX with fit4green using dynamic workload PUE 2.0 - 2.0 CUE 1.7 – 1.7



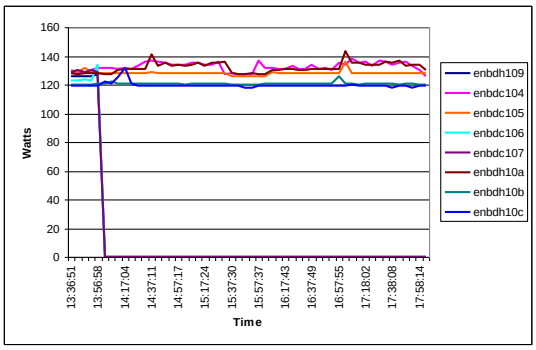
Power consumption of Cluster MOEX2 with fit4green using dynamic workload PUE 2.0 -2.0 CUE 1.7 – 1.7



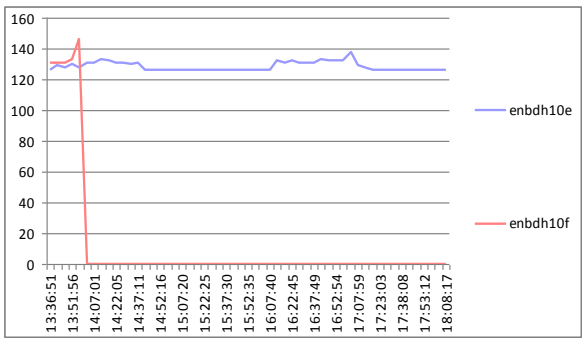
Average Power consumption per server of Cluster MOEX with fit4green using dynamic workload PUE 2.0 - 2.0 CUE 1.7 – 1.7



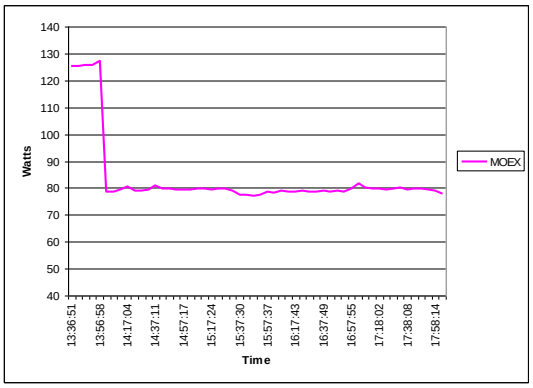
Average Power consumption per server of Cluster MOEX2 with fit4green using dynamic workload PUE 2.0 - 2.0 CUE 1.7 – 1.7



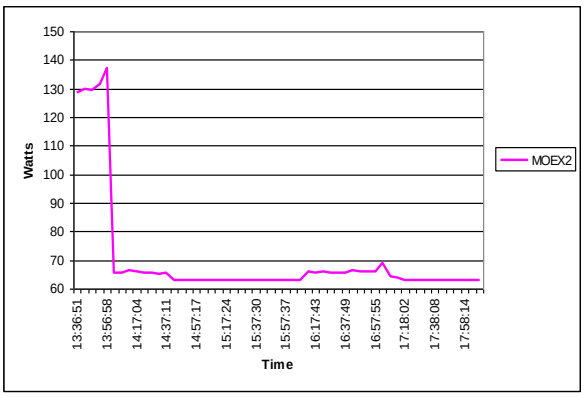
Power consumption of Cluster MOEX with fit4green using dynamic workload PUE 2.0 - 2.0 CUE 1.7 – 1.7



Power consumption of Cluster MOEX2 with fit4green using dynamic workload PUE 2.0 -2.0 CUE 1.7 – 1.7



Average Power consumption per server of Cluster MOEX with fit4green using dynamic workload PUE 2.0 - 2.0 CUE 1.7 – 1.7



Average Power consumption per server of Cluster MOEX2 with fit4green using dynamic workload PUE 2.0 - 2.0 CUE 1.7 – 1.7

Figure 25 – Exemplary Results for execution of dynamic workload in a federated case

II.4.5.c. Dynamic workload including allocation of virtual machines

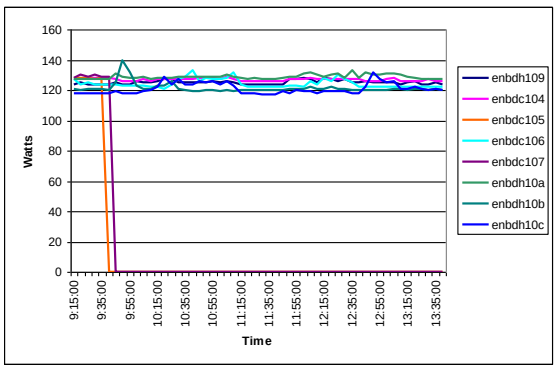
The next results are obtained including allocation of new virtual machines driven by the FIT4Green plug-in and a more restrictive SLA configuration.

With this configuration the result of the optimizer is to power off only two servers on MOEX and not powering on actions on MOEX2.

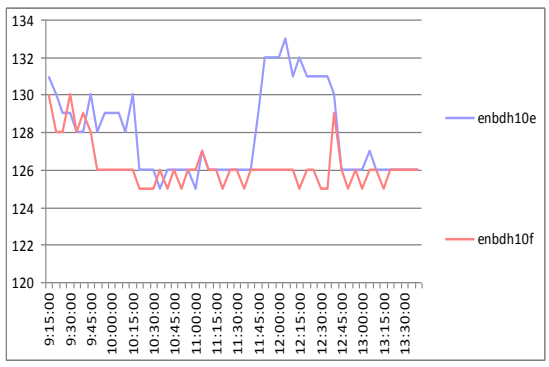
This time the consumption of the servers is maintained thought time because the virtual machines executing on the powered off servers is evenly distributed among the rest of the servers.

When we look at the average consumption we can see that the average consumption before the optimizations start is around 125 Watts and after the powering off actions ends the consumption is lowered to around 95 Watts.

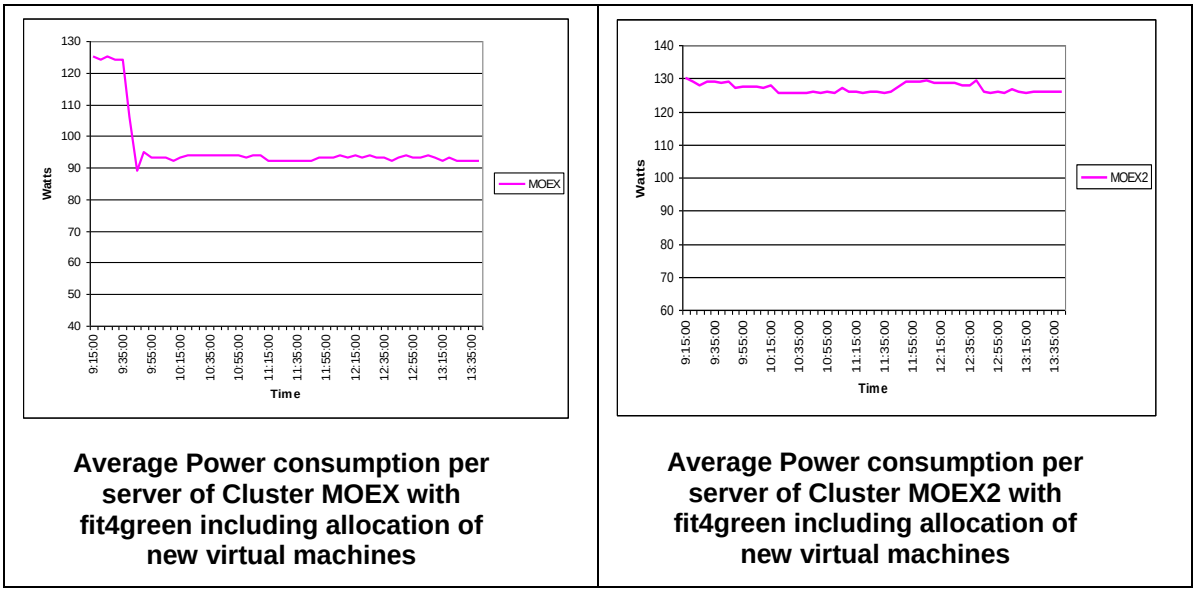
The consumption on MOEX2 doesn't change because no powering off actions are performed, so the consumption is between 125 and 130 Watts.



Power consumption of Cluster MOEX with fit4green including allocation of virtual machines



Power consumption of Cluster MOEX2 with fit4green including allocation of virtual machines



Average Power consumption per server of Cluster MOEX with fit4green including allocation of new virtual machines

Average Power consumption per server of Cluster MOEX2 with fit4green including allocation of new virtual machines

Figure 26 - Results for execution of the workload with PUE 1.5 – 2.5 CUE 1.7 – 1.7 optimizing power including allocations

II.4.5.d. Results

These results have been obtained with a configuration of the values for the SLA Cluster Constraints that allows a maximum optimization result. This concludes to a saving of 50% at its peak. This high value is achieved not only because of this SLA Constraints configuration values but in addition are due to the Workload used which doesn't contains allocation of new VMs.

The last result presented in this section was obtained with a configuration of the values for the SLA Cluster Constraints that are stricter and more in line with ENI's SLA policies and allocations of new VMs added to the Workload. This scenario presents more restrained results of 30% at peak.

Figure 27 shows the energy usage of the test bed prior to the optimizations. The usage is around 25300 Watts, with a low peak of usage 24900 Watts and a high peak of 25450 Watts

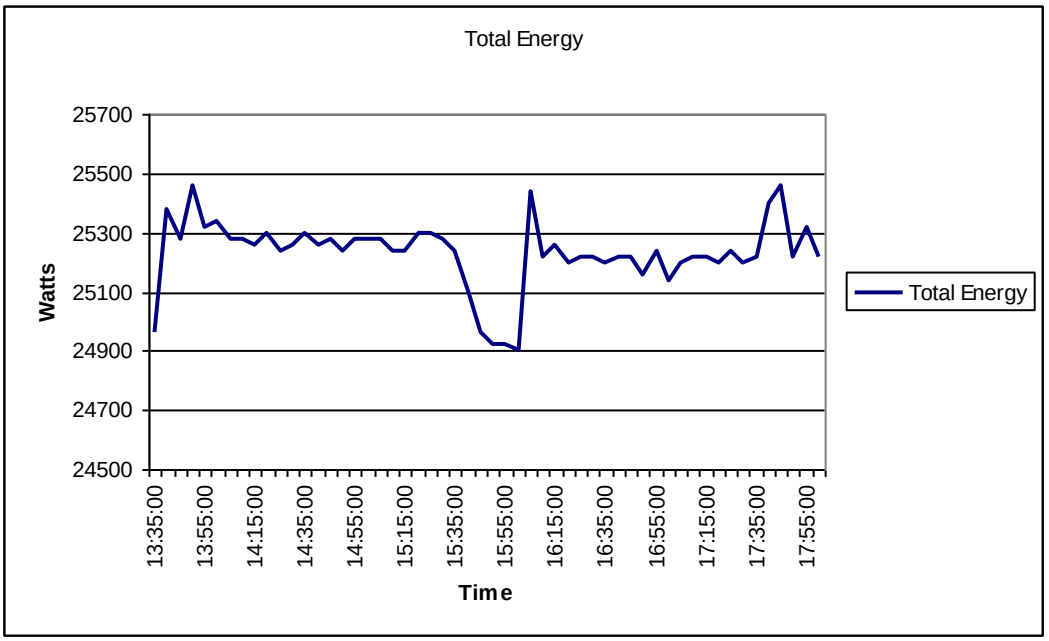


Figure 27 – Total energy consumption without fit4green

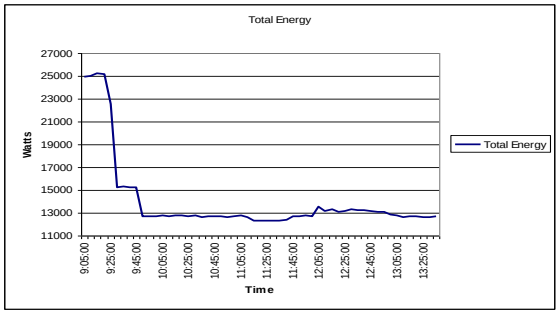
On the next figures we will see the results for the execution of the workload with different variations of PUE and CUE for both data centres optimizing by power, depicted on the left figures, and emissions, depicted on the right figures.

As stated before in the previous section the results between the different configurations of PUE and CUE are quite similar because of the lack of allocations of new virtual machines.

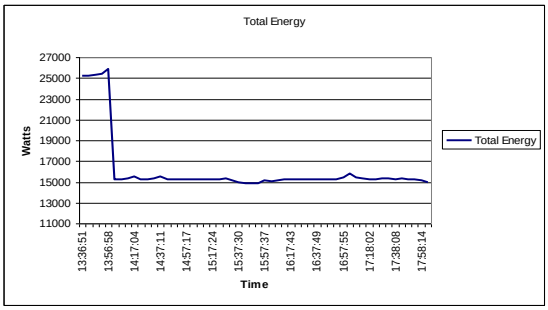
Before the optimizations take place the energy consumption is around 25000 Watts, when the optimization is performed, the result is powering off three or four servers on MOEX depending on the situation as explained before and one server on MOEX2. With this optimizations the energy consumption drops to between 15000 and 13000 Watts and is maintained over time due to the several optimizations by the FIT4Green plug-in moving virtual machines across the servers. Therefore we obtain total savings of between 10000 12000 Watts, that in terms of percentage is a saving of around 45%.

The same applies when optimizing for both power and emissions as explained before.

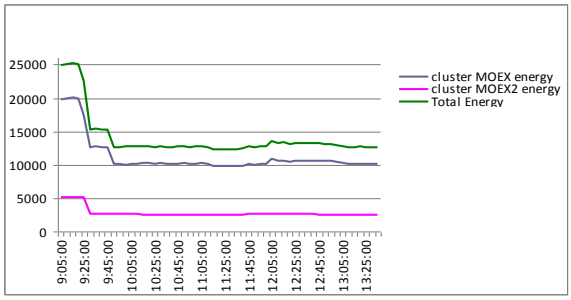
Next all the figures for the all combinations for PUE-CUE and optimizations for power and emissions are presented.



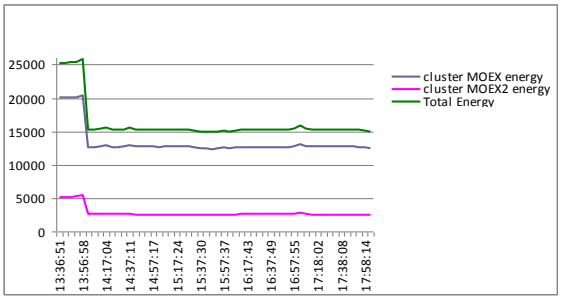
Energy usage of the test bed with fit4green using dynamic workload with PUE 2.0 - 2.0 CUE 1.7 - 1.7 optimizing power



Energy usage of the test bed with fit4green using dynamic workload with PUE 2.0 - 2.0 CUE 1.7 - 1.7 optimizing emissions



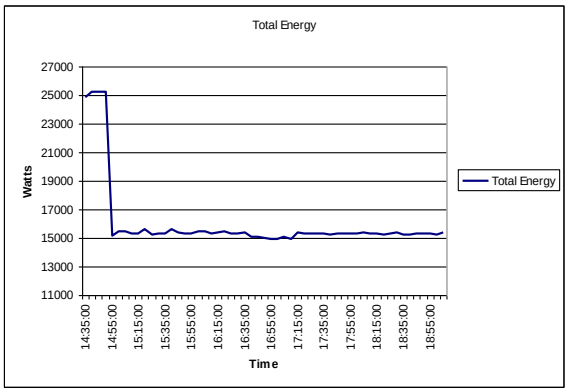
Energy usage of the test bed using dynamic workload with PUE 2.0 -2.0 CUE 1.7 – 1.7 optimizing power



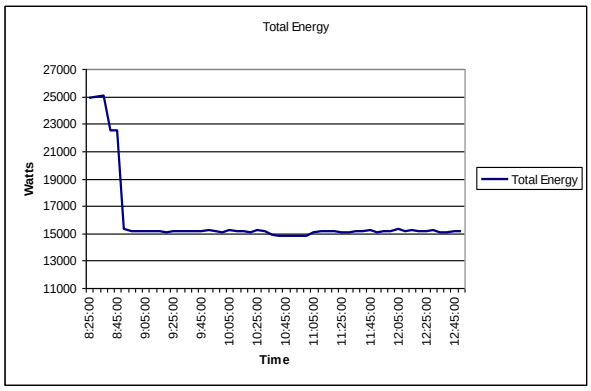
Energy usage of the test bed using dynamic workload with PUE 2.0 - 2.0 CUE 1.7 – 1.7 optimizing power

Figure 28 – Results using PUE 2.0 – 2.0 CUE 1.7 – 1.7

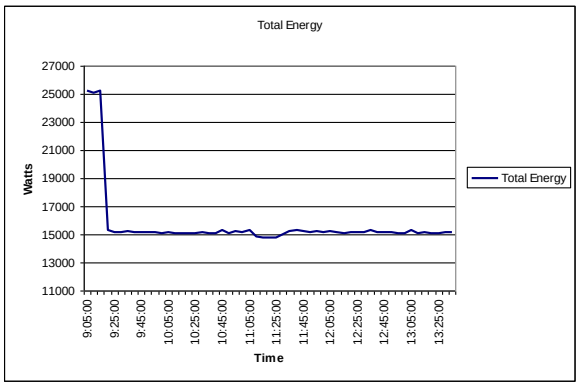
The next table presents the results for optimizations for emissions with PUE 1.5 – 2.5 and PUE 2.5 – 1.5 with CUE 1.7 – 1.7 on MOEX and MOEX2.



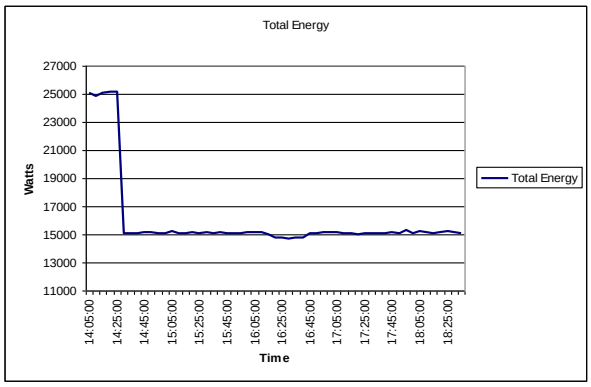
Energy usage of the test bed with fit4green using dynamic workload with PUE 1.5 - 2.5 optimizing power



Energy usage of the test bed with fit4green using dynamic workload with PUE 1.5 - 2.5 optimizing emissions



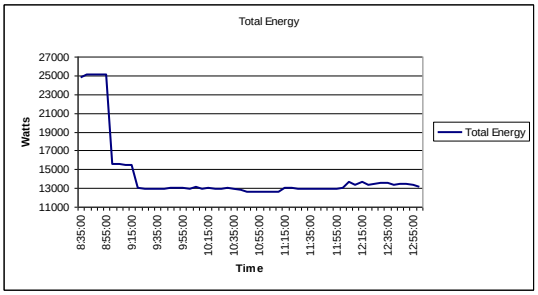
Energy usage of the test bed with fit4green using dynamic workload with PUE 2.5 - 1.5 optimizing power



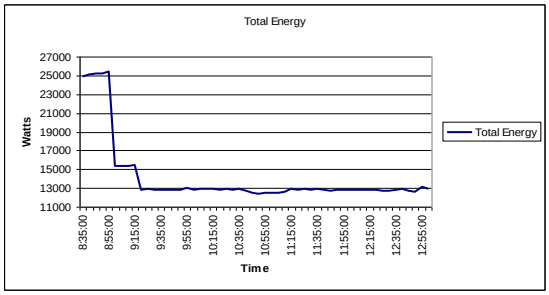
Energy usage of the test bed with fit4green using dynamic workload with PUE 2.5 - 1.5 optimizing emissions

Figure 29 – Detail of the total energy usage of the test bed using different PUEs

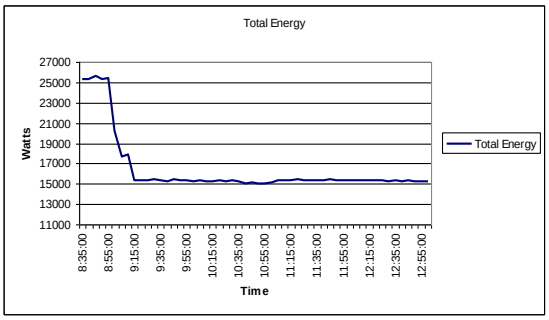
The next table presents the results for optimizations for emissions with CUE 0.25688 – 1.7 and CUE 1.7 – 0.25688 with PUE 2.0 – 2.0 on MOEX and MOEX2.



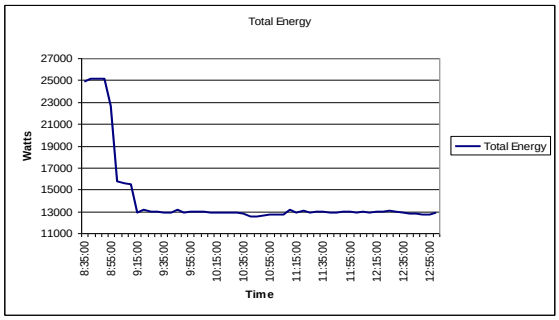
Energy usage of the test bed with fit4green using dynamic workload CUE 0.25688 – 1.7 optimizing power



Energy usage of the test bed with fit4green using dynamic workload CUE 0.25688 – 1.7 optimizing emissions



Energy usage of the test bed with fit4green using dynamic workload CUE 1.7 - 0.25688 optimizing power



Energy usage of the test bed with fit4green using dynamic workload CUE 1.7 - 0.25688 optimizing emissions

Figure 30 - Detail of the total energy usage of the test bed using different CUEs

The next figures show the results regarding Carbon emissions. When the two data centres are using the same CUE of 1.7 the emissions before the optimisation actions are around 43000 Kg and after the optimization actions are performed the emission drops to 26000 Kg. Thus the savings are 17000 Kg (i.e. 39.5%).

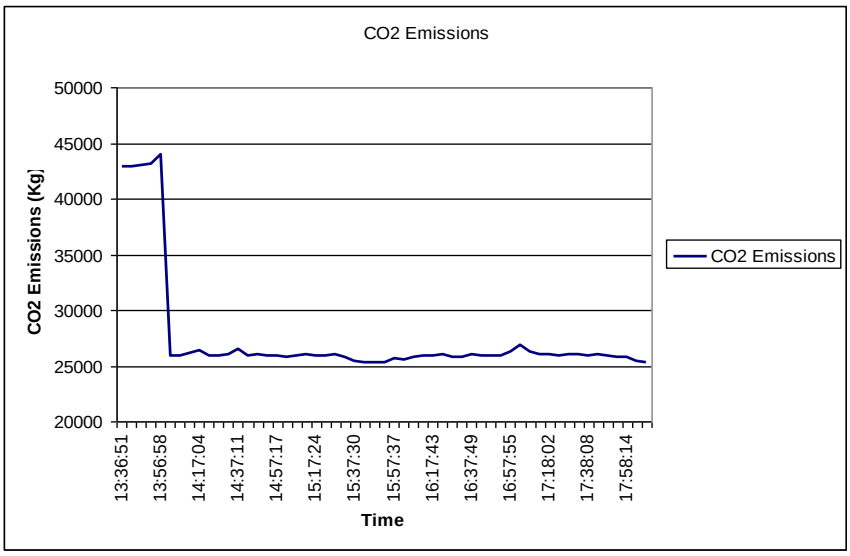


Figure 31 - Carbon emissions of the test bed with fit4green using dynamic workload CUE 0.1.7 – 1.7 optimizing emissions

If the MOEX data centre use a CUE of 0.25688 and MOEX2 use a CUE of 1.7 the emissions before any optimization actions are around 14000 Kg whereas the emissions drop to 7000 Kg after the optimizations. The savings are therefore around 7000 Kg (i.e. 50%).

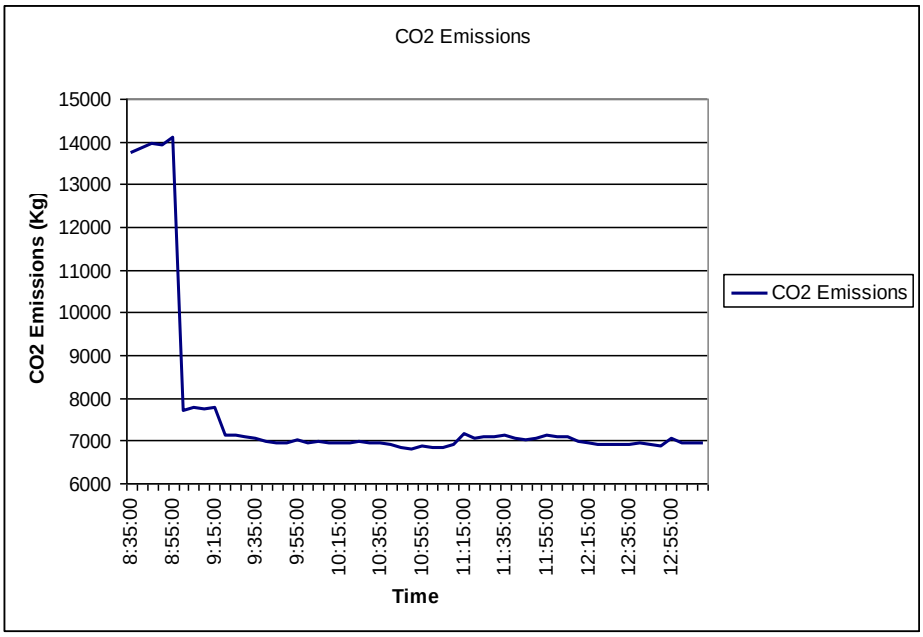


Figure 32 - Carbon emissions of the test bed with fit4green using dynamic CUE 0.25688 – 1.7 optimizing emissions

If the MOEX data centre uses a CUE of 1.7 and MOEX2 use a CUE of 0.25688 the emissions before optimisation are around 35.000 Kg and after the optimizations are performed the emissions drop to 18000 Kg This leads to savings of 17.000 Kg that in terms of percentage is 48% (in his high peak).

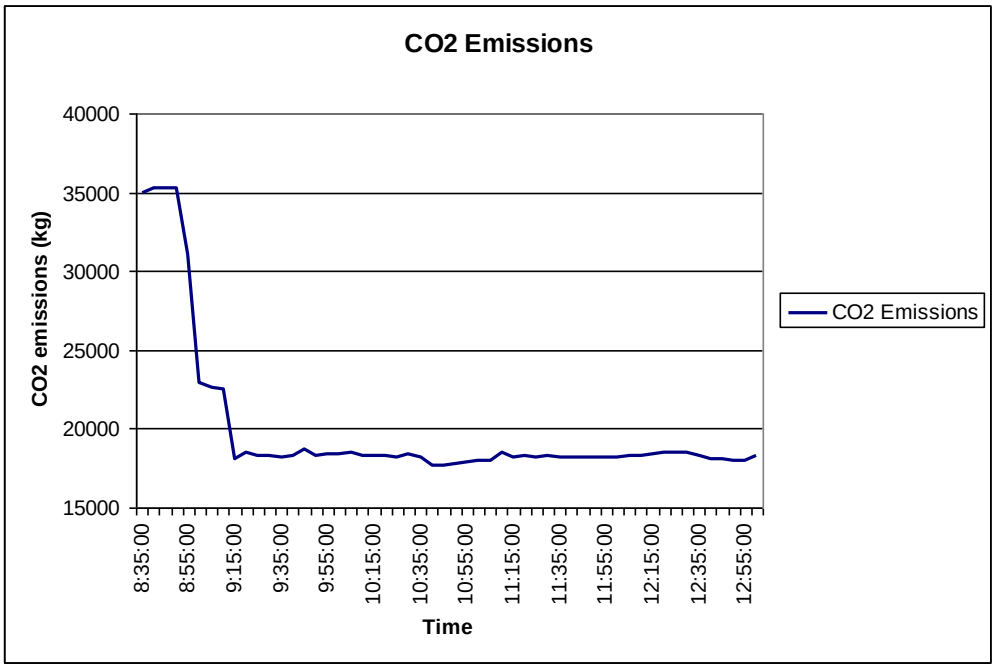


Figure 33 - Carbon emissions of the test bed with fit4green using dynamic CUE 1.7 - 0.25688 optimizing emissions

If we look at the carbon emission we can conclude that when the CUE value for MOEX is greater the total carbon emissions are greater, too. This is because the MOEX data centre is greater than MOEX2. If we look only at percentage savings, we found that the percentage saved in all cases is between 40% and 50%. This is due to the absence of the

creation of new VMs during the workload and the strong optimizations performed by FIT4Green plug-in.

The next figure represents the total energy saved with a configuration more compliant with ENI's policies, no more than two virtual machines per server and no more than 90% of memory allocation per server.

The consumption starts at a level of around 25000 Watts. When the optimization is performed with these policies, the result is powering off two servers on MOEX decreasing the consumption to 18000 Watts, with a total saving of 7000 Watts, that in term of percentage is around 28%.

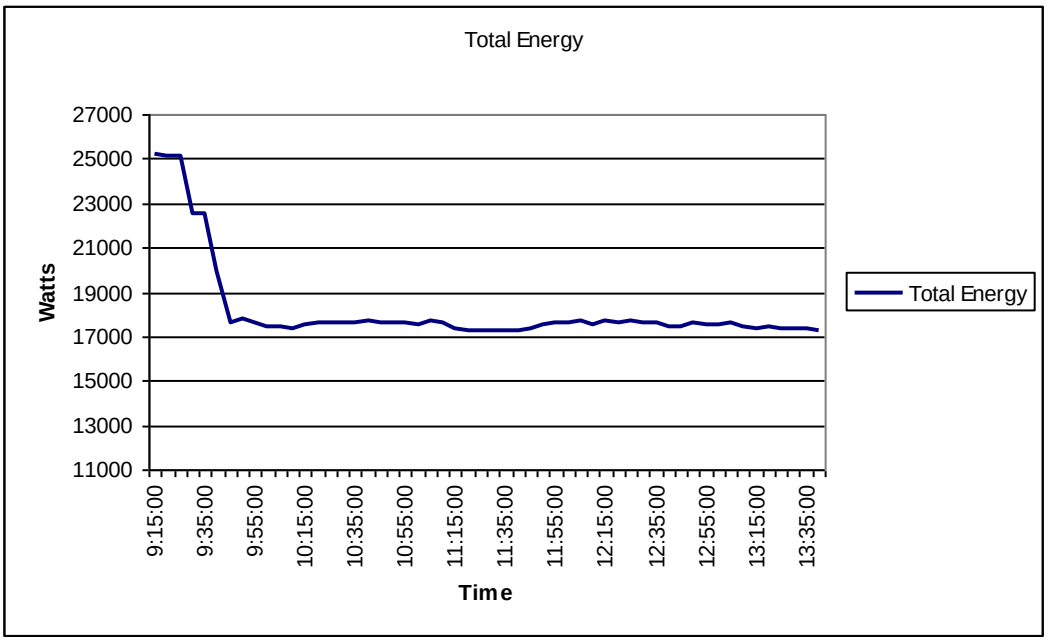


Figure 34 – Total energy consumption with allocating VMs and stricter SLA

II.5. Evaluation and feedback to next phases

II.5.1. Technical evaluation

The Workload used for the tests is based on Microsoft Exchange applications that extensively use the memory of the hosts and very little of the CPU. During the tests the fit4green plug-in made a heavy consolidation that didn't take the physical RAM usage in account. VMWare best practices suggest to not do a RAM overbooking because if there is a high CPU usage the effects could be a "performance issue" (critical or not depends from how CPU consolidation and CPU usage are high), but if we allocated all physical RAM to VMs there won't be enough resources that the Hypervisor could use. The VMWare Hypervisor is responsible for allocating the needed hardware resources to the VMs Operating Systems and to schedule their tasks on the physical host. As a result of that, the VMWare Hypervisor doesn't have enough resources to work with VMs and can "freeze".

For verifying that there wasn't a resource overcapacity, an additional task was added to the test: new VM creation task. It creates a Virtual Machine with a pre-installed application that has a custom workload (graphical elaborations like "Autocad"). When the fit4green plug-in started his optimization activity it doesn't care about "how is the free quantity of RAM on the cluster" and this generate first a performance decrease on MS Communication System and then a critical performance issue that generate an critical error that doesn't allow to the workload to be run.

We decided to get a maximum of 2 VMs per Host creating a new configuration for the SLA values (conservative values, so we could have a “low consolidation” but we was sure that there were enough resources to let workload run) but, during the test, we saw that the fit4green plug-in consolidates a high number of VMs per server (until 4 VMs per hosts) because of the little CPU used. This could be explained because at the moment the configuration parameters for SLAs are on “development phase”.

Finally we could say that for a “efficient and effective consolidations” only “VMs per server” is not enough to take into account, but we have to analyze how many physical resources are free and how many resources a consolidation task could need (for example, if I power off 2 server I have to be sure that the total amount of CPU and RAM on the cluster is capacity enough for VMs requirements).

The conclusion of this is that it is needed to have an explicit SLA constraint value for memory on the SLA cluster constraint configuration file in the FIT4Green plug-in, along with the previously requested number of virtual machines per server. With the addition of new values on the SLA constraint configuration file to fine tuning the behaviour of the FIT4Green plug-in opens our hope that we will have very good chance to achieve our goal to have a really good savings in compliance with the SLA policies in our Traditional test bed for the third phase.

II.5.2. Usability evaluation

From the data centre operator viewpoint, the FIT4Green User Interface has been improved.

The most important feature is displaying the actions suggested by the FIT4Green plug-in and especially allowing the approval of the list of actions; this feature was critical for the Traditional test bed and has been successfully implemented.

The new statistics pane has been a great addition and it has proven very useful to record historical data about the execution of tests, and the actions suggested by the FIT4Green plug-in.

The XML file editor is still the weakest point of FIT4Green User Interface. Improving the editor or implementing some helper to automatically generate the FIT4Green meta-model file would be very beneficial.

III. SUPERCOMPUTING DATA CENTRE TESTBED

III.1. Testbed environment and configuration

The FIT4Green Supercomputing testbed at Forschungszentrum Jülich (FZJ) as described in D6.2 has been extended for facilitating the use cases of the second pilot phase. In contrast to the first pilot phase, there are now two clusters, referred to as 'Juggle' and 'Jufit', available in the testbed allowing particularly the usage of the FIT4Green resource allocation requests for the federated scenario. Both clusters are located in the same data centre facility at the Jülich Supercomputing Centre (JSC) of the FZJ. Test times on both systems have been reserved for the FIT4Green benchmarks. Supercomputing resources from other facilities outside of Jülich were not available at the time of benchmark tests of the second pilot phase.

Juggle

The Juggle cluster was already used for the benchmark tests in the first pilot phase. However, for the second phase, the equipment of 4 worker nodes has been enlarged; so that the cluster comprises now altogether 12 nodes (see Table 2

). This allows executing more applications in parallel as well as stressing the system with larger and more typical High Performance Computing (HPC) applications. Besides that extension no other hardware changes were done on the cluster since the 1st pilot phase.

Concerning software updates, Intelligent Platform Management Interface (IPMI) services and monitoring tools if not already provided by the operating system have been installed on each node of the system to enable the monitoring of dynamic system parameters as core voltage and frequency, memory load, fan RPM, and disk read/write-rates. In addition, a Unicore target system interface has been installed on the head node of the Juggle cluster to allow job submissions with Unicore clients which was required for the federated test scenario.

Processor type	Dual AMD Opteron F2216 2.4GHz
Number of nodes	1 head node, 12 worker nodes
Cores per node	4
Overall number of cores	48
Main memory	8 GB per node
Network	InfiniPath(QLOGIC), Gigabit Ethernet
2 file servers	disk capacity: 6 TB
Power supply efficiency	~ 83%
Operating system	SLES 10, Scientific Linux 5.2
RMS	Torque (PBS Scheduler)
Node power consumption standby/idle/maximum	117W / 162.5W / 230W

Table 4 - Operating numbers of the Juggle cluster

Jufit

Compared to the Juggle cluster Jufit is a more modern system providing a more modern generation of processors. It consists of 2 worker nodes with 12 cores each (see Table 3**Error! Reference source not found.**). Compared to Juggle, Jufit is in general more energy efficient in terms of CPU power consumption and power supply efficiency. The Jufit cluster has been used in the measurements for the federated scenario.

Processor Type	Quad-core Intel Xeon X5660 (Westmere), 2.6 GHz
Hyperthreading	SMT (Simultaneous Multithreading)
Number of Nodes	1 head node, 2 worker nodes
Cores per node	12
Overall number of cores	24
Main Memory	24 GB per node
Network	InfiniPath(QLOGIC), Gigabit Ethernet
2 File Servers	disk capacity: 6 TB
Power Supply Efficiency	~ 91%
Operating System	OpenSuSE 11.3
RMS	Torque (Maui Scheduler)
Node power consumption standby/idle/maximum	142W / 175W / 232W

Table 5 - Operating numbers of the Jufit cluster

Software configuration

The Resource Management System (RMS) Torque is installed on both clusters for managing nodes and the scheduling and monitoring of Jobs. For that reason the Proxy connector which monitors the system and executes actions for FIT4Green is adapted to that batch system. Furthermore, Target System Interface (TSI) modules of the Unicore middleware are installed on the head nodes of the clusters to allow submitting Unicore jobs which is needed in the case for the federated FIT4Green scenario.

Power Measurement

Both clusters are connected to Raritan Power Distribution Units (PDUs) which measure the power consumption of each single head and worker node. The results are requested conveniently by clients through the Simple Network Management Protocol (SNMP). The measurements were updated every 3 seconds during the tests, so that the values could be provided relatively precisely.

Power Management and Energy Saving Modes

The motherboards of the Juggle and Jufit compute nodes support the ACPI[1] state S1, which is also known as 'standby' state. As soon as the FIT4Green software is generating a standby action due to a global optimization request, the appropriate compute node will be set to standby mode. Measurements showed that energy savings between 15% (Jufit) and 28 % (Juggle) can be expected in standby mode compared to the energy consumption of a respective normal idle compute node (ACPI state S0). While on Juggle 'wake-on-lan' is used to bring the machine back from standby to normal state, the same result on Jufit is achieved by an IPMI wake up command.

III.2. Testing methodology

The goal of the test measurements is to compare the energy consumption of FIT4Green adapted supercomputing environments with systems using default non-FIT4Green solutions. The results of the tests should analyse the energy saving capabilities of the FIT4Green software in two different areas:

- Savings on a single cluster by using the FIT4Green global optimization
- Savings in a federated cluster scenario by using the FIT4Green resource allocation

For each of these investigation areas specific test approaches have been carried out regarding the used benchmark workloads, kind job submission, and energy metering. All measurements depended on the available hard- and software as described in chapter III.1.

The workloads stressing the FIT4Green testbed consist of jobs which make use of the available HPC applications described in III.3. The jobs can be either submitted directly from the test user's home directory on the head node of the cluster or alternatively from the Unicore client, so that the user doesn't need to be logged in to the cluster. Both submission types are quite common in supercomputing scenarios.

Single Site scenario

The single site scenario tests have been performed on the Juggle system which provides a suitable number of nodes for testing the energy saving potential by using the FIT4Green global optimization. Firstly, this scheduling mechanism schedules jobs in the queue of the cluster to the particular nodes/cores of the cluster, and, secondly, it sets nodes to standby if they are completely idle. Thus, when having equal workloads and comparing FIT4Green global optimization with a non-FIT4Green strategy, the possible energy consumption depends on how efficiently the jobs can be scheduled without loss of time, so that as many nodes as possible can be set to standby.

In the supercomputing testbed at FZJ the energy measurements have been analysed by comparing the default PBS scheduler with the FIT4Green global optimization strategy. While the PBS scheduler is based on an enhanced FIFO (first in first out) algorithm, the FIT4Green one used the backfill first fit approach (see D4.1). The particular measurements were performed with workloads generating different system utilizations on the Juggle cluster (0%, 50%, 66%, and 90%) to map potential loads of real supercomputing machines.

The total energy consumption generated by a single test workload has been calculated in Joule as a product of the measured average power of all cluster nodes and the elapsed time which was needed to run all jobs of the workload. The elapsed time involves the total time elapsed from the submission of the test user's first job until the output files of the last executed job has been stored where requested. So, this period includes the time for

[1] http://acpi.sourceforge.net/documentation/sleep.html

transferring input files, the waiting time in the RMS queue, the actual execution time, and the time to stage the output files to the requested locations. Each measurement has been stopped immediately after all jobs of the test workload were finished. This approach is mandatory to measure the time that different scheduling strategies need to process the workload. So, the energy of one measurement was calculated as follows:

$$total\_energy_{SingleSite} = elapsed\_time_{Workload} * avg\_power_{Cluster}$$

Federated scenario

When performing tests in the single site scenario the jobs of the workload have been submitted locally from the cluster's head node by using provided shell commands of the RMS on the dedicated cluster. This approach is common practice in supercomputing environments when the dedicated target system of the job is already known. In the federated scenario another job submission procedure has come into operation since it is not known in advance on which cluster the job should be executed. So, the jobs are at first submitted to a Unicore server which acts as a centralized entry point for all incoming jobs. This Unicore instance is connected to the installed Unicore TSIs on both testbed clusters (see III.1.), so that the server is able to submit incoming jobs to the RMS of an appropriate machine. Before that, the Unicore Service Orchestrator (USO) service of the Unicore server initiates a resource allocation request to ask FIT4Green for a suitable target machine for the job. Figure 35 illustrates the difference in terms of job submission between the single and federated scenario.

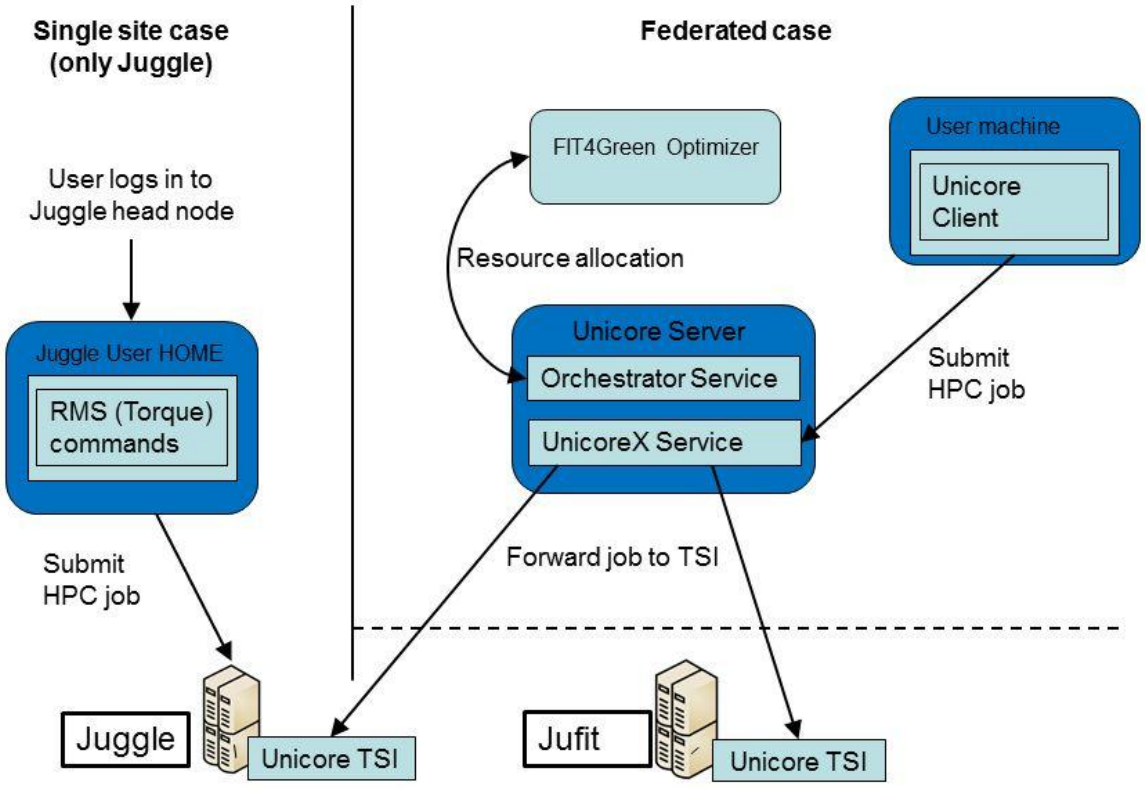


Figure 35: Difference of job submission in single and federated benchmark tests

In general, the workloads have been compared in each test case by using FIT4Green scheduling strategies as well as default mechanisms for getting results of how efficiently the FIT4Green software is able to save energy. Without FIT4Green software means that

existing software and state of the art mechanisms have been used to process the jobs on the test clusters.

The approach for measuring the energy consumption in the federated scenario is to consider only the elapsed time which is needed on each testbed cluster to run the assigned jobs of the benchmark workload. This incorporates that each involved cluster produces in one benchmark a different elapsed time and different average power consumption. So, the total energy consumption in one measurement is the product of the elapsed time and the average power of each cluster:

$$total\_energy_{Federated} = time_{Cluster1} * avg\_power_{Cluster1} + time_{ClusterN} * avg\_power_{ClusterN}$$

III.3. Test workload

The benchmark measurements on the testbed were aimed at stressing the testbed as close as possible as clusters in a real supercomputing environments. For that purpose different typical HPC applications were installed on the test clusters. LINPACK, a collection of FORTRAN subroutines to solve linear systems, was already used in the first pilot phase. Additionally, PEPC[2] (Pretty Efficient Parallel Coulomb Solver) has been installed, which is used to run astrophysical N-body simulations.

Single Site scenario

For the single site scenario the test workloads have been created by a configurable Perl script which can parameterise the jobs in terms of the used HPC application, the level of computation intensity, the number of used nodes and cores, as well as the planned walltime (also known as wall clock time) which is the time elapsed until a job should have been finished. In this way workloads were created stressing the system with different system utilization in order to analyse the energy savings under those different loads. Also real world clusters working in production show often varying system utilization between entirely idle and working to capacity. The utilization factor is defined here as the percentage of time when cluster resources are stressed with jobs relative to the total elapsed time of the workload. For instance, a system load of 90% means that only on an average of 10% of the elapsed time cluster nodes are able to be set to the energy saving standby mode because they are otherwise busy with running jobs.

Federated scenario

In case of the federated scenario the workloads of the single site scenario have been adapted as a template to create workloads using the same HPC applications within the graphical Unicore Rich Client (URC). This workload is embedded in a workflow from where the single jobs are submitted in parallel to the Unicore server which in turn initiates resource allocation requests to FIT4Green and forwards subsequently the jobs to the chosen cluster. Figure 36 shows the operation of the URC in the federated scenario tests.

[2] *www2.fz-juelich.de/zam/pepc*

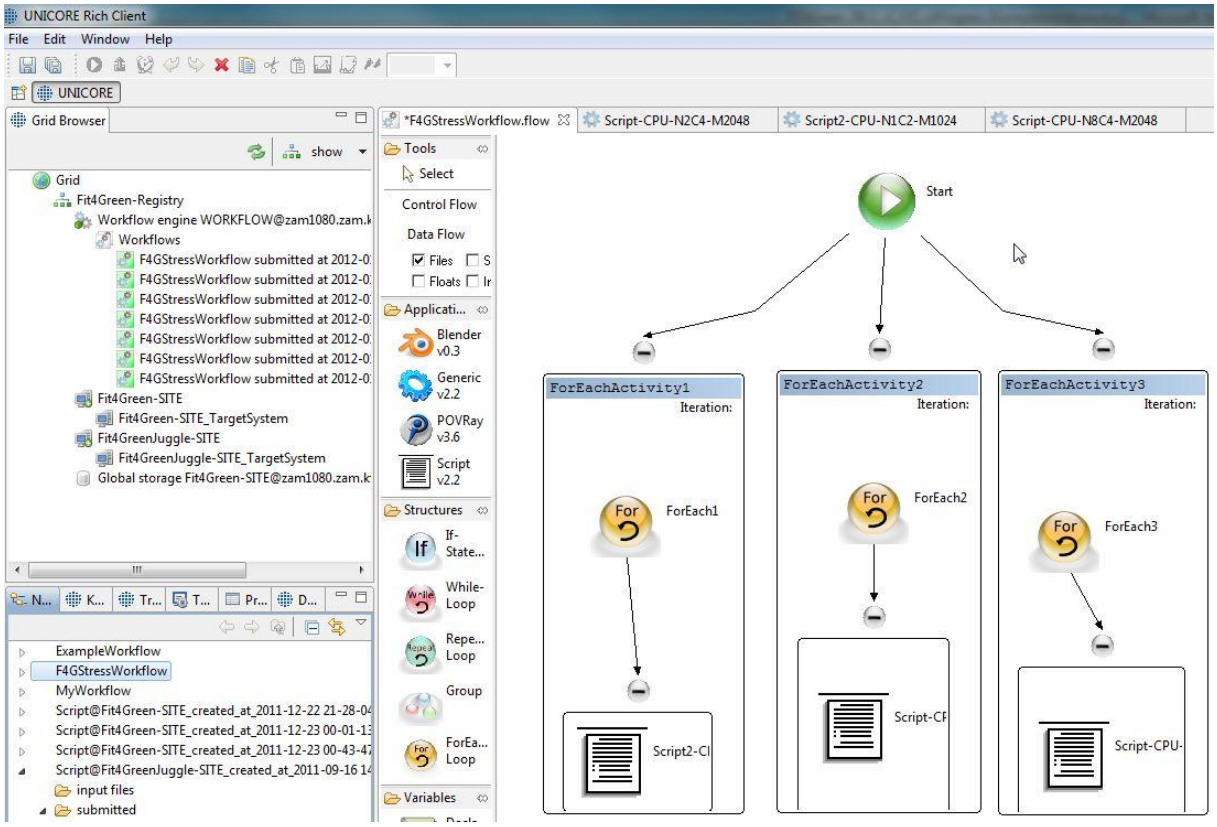


Figure 36 – Creating and submitting workloads with the Unicore client

Round about 90% of the workload jobs can run on both clusters, while the requirements of always 5% of the jobs can only be met on one of the clusters. This distribution created a base load on the clusters to map better real world environments where jobs are usually not able to run on every available target machine.

III.4. Numerical results

III.4.1. Single Site scenario

The single site scenario tests have been performed on the Juggle system which provides a suitable number of nodes for testing the energy saving potential by setting nodes to standby status. In general the energy measurements have been analysed by comparing the PBS default scheduler with the usage of the FIT4Green global optimization request for scheduling the jobs on the compute nodes. These measurements were performed with workloads generating a different total load on the cluster (0%, 50%, 66%, and 90%). Each workload measurement has been repeated with five iterations. The results from Table 4**Error! Reference source not found.** show the average values of those tests.

The energy consumption of a single workload has been calculated in Joule as a product of the measured average power of all cluster nodes and the elapsed time which was needed by the appropriate workload. The elapsed times of each workload depend on the composition of the workload to achieve certain system utilization, so there is no correlation between the elapsed time values of different system loads. In contrast, the average power consumption increases with more intensive workloads, since less idle nodes can be set to an energy saving status.

When using FIT4Green we could calculate energy savings in each test case compared to the usage of the default RMS software. The highest possible energy saving on the cluster is

27.3% which happens when the system is completely idle. That value is limited by the matter of fact that a compute node which was set to ACPI state standby cannot save more than 28% energy (see chapter III.1. **Error! Reference source not found.**).

Considering the workloads from 50% to 90% system load, it can be detected that the energy saving when using FIT4Green decreases from 10% to 3.8% as more as intensive the workload is. This correlation is logical consequence, since as higher the system utilization is as less compute nodes can be set to an energy saving status.

When comparing the values of non-FIT4Green measurements with FIT4Green enabled tests it is apparent that the elapsed time is most time slightly higher with the FIT4Green strategy which is caused by the overhead of the global optimization process. In particular, cluster information as node and job statuses must be read and analysed at the remote FIT4Green server, and generated actions must be sent back to the RMS of the cluster. However, this process has been optimized in terms of performance during the implementation phase, so that the time impact has been minimized. Taken into account similar elapsed time results in the measurements, the main factor for saving energy is clearly the measured average power of the tested cluster which is proportional in the FIT4Green measurements to the decreasing system load. While the default RMS scheduler cannot make advantage of idle compute nodes, FIT4Green sets them in an energy saving standby mode which reduces noticeably the average power of the system.

System utilization [%]	**0%**	**50%**	**66%**	**90%**
With FIT4Green [kJ]	***1594***	***4822***	***9274***	***8860***
Elapsed Time [s]	*1000*	*2397*	*4372*	*3721*
Average Power [W]	*1594*	*2012*	*2122*	*2381*
No FIT4Green [kJ]	***2193***	***5366***	***9909***	***9208***
Elapsed Time [s]	*1000*	*2326*	*4252*	*3704*
Average Power [W]	*2193*	*2308*	*2331*	*2486*
Energy saving by using FIT4Green [%]	**27.3**	**10.0**	**6.4**	**3.8**

Table 6 - Numerical results of single site measurements

III.4.2. Federated scenario

In the supercomputing federated scenario we wanted to measure the emission and energy saving capabilities of the FIT4Green resource brokering strategies. Jobs should be assigned to suitable cluster resources in the most energy efficient way. The FIT4Green HPC Optimizer implements two different strategies to achieve that resource allocation:

- Fastest possible scheduling
- CO_2 (CUE)/ energy aware scheduling

FIT4Green D4.2 gives already a detailed description of the both strategies. In short, the fastest possible algorithm calculates the wait time of a job on all potential suitable clusters and submits the job to the system which provides the most minimal estimated queue time.

In contrast, the CO_2 aware algorithm estimates at first the CO_2 emission which would be produced by a job on a particular cluster. The emission is calculated by considering the CUEs of the clusters as well as estimating the energy consumption of particular jobs on a cluster. In the supercomputing testbed at FZJ PUE and CUE indexes are equal for both

testbed clusters, since both machines are located in the same datacentre environment. So, the crucial factor in terms of energy efficiency is the power consumption of the testbed clusters over a dedicated time. Additionally, the energy aware algorithm checks if the user defined 'latest job finishing time' can be satisfied. This means it is checked if the job can be executed and finished on a cluster within a user defined limit. If not the job cannot be scheduled in the best energy efficient way and the next cluster is chosen where the estimated wait time is smaller than the user defined value. By this mechanism the user can set a threshold value from where jobs must not be scheduled energy efficiently anymore.

Because of the equal CUEs in both clusters the CO_2 emissions are proportional to the measured energy consumption. So, in the following context only the energy consumption is considered.

We compared the results of the brokering strategies developed in FIT4Green with algorithms that are already used in distributed scenarios and didn't make use of FIT4Green resource allocation algorithms. Nevertheless, it has to be mentioned that resource brokering in heterogeneous environments is still an on-going research area in HPC computing. Brokers are usually not yet deployed in real production environments but rather in smaller research and test environments. One of the main barriers is that HPC job requirements are often strongly system-related so that they can only run in a dedicated hard- and software environment. There is also a lack of dynamic resource information on the broker level about node and queue statuses of the bounded supercomputers.

Since Unicore is the default Grid middleware at FZJ and was also used as an entry point for jobs in the federated FIT4Green scenario, we investigated the default brokering mechanism of that software stack, which is embedded in the Unicore Service Orchestrator (USO). At the time of performing this investigation, the USO didn't provide any dynamic system information about jobs in queue or the statuses of nodes, so it was restricted to use a simple round-robin strategy for brokering the jobs to the available clusters.

Furthermore, we took into account that in Grid Computing scenarios brokering software is often not available. When there are various clusters available users are used to submit their jobs randomly to a suitable resource fulfilling the job requirements. Such a user controlled job submission can be considered as a kind of poor distribution of jobs to the clusters in terms of energy efficiency, since users are generally not aware about the energy consumption of the particular clusters. So, we considered the following non-FIT4Green strategies:

- Unicore Service Orchestrator round-robin
- Randomly user controlled job simulation

The measurements showed that the numerical results of the stress tests using the same brokering strategy are often differentiating in terms of the elapsed time the workload needs on a cluster and respectively the average power consumption that was generated on the resource. Thus, measurement series were performed for each strategy and the mean value was calculated. In the federated tests the FIT4Green global optimization for the single sites was activated regardless if FIT4Green enabled brokering strategies or default mechanisms were analysed. So, equal conditions on the single sites in terms of local scheduling were established. For analysing the impact of short and long term system utilization two different workloads that differ in size have been generated. In both workloads the 'latest job finishing time' parameter has been set to 3600 seconds.

Short system utilization

The numerical results of the short system utilization are listed in Table 5. The FIT4Green enabled brokering strategies showed clearly a gain in terms of the energy consumption. The best results were achieved by brokering the jobs with the FIT4Green energy aware algorithm. Compared to the default round-robin strategy an energy saving of 45.2% could

be measured. When comparing to the user controlled submission an even higher conservation of 51.7% has been measured. The reason for this clearly is that the FIT4Green algorithms make much more use of the more efficient Jufit cluster than the other strategies does. In contrast, the round-robin strategy and the user controlled submission distributes the jobs almost equally to each of the clusters.

Different elapsed times, 51 jobs in workload	**FIT4Green fastest possible**	**FIT4Green energy aware**	**USO Round-Robin**	**User controlled submission**
Juggle [kJ]	***1080***	***1025***	***2363***	***2731***
Jobs	*13*	*5*	*24*	*28*
Elapsed Time [s]	*541*	*529*	*1148*	*1446*
Average Power [W]	*1997*	*1941*	*2057*	*1889*
Jufit [kJ]	***400***	***419***	***271***	***259***
Jobs	*38*	*46*	*27*	*23*
Elapsed Time [s]	*1242*	*1278*	*849*	*792*
Average Power [W]	*322*	*328*	*320*	*326*
Total cluster [kJ]	**1480**	**1444**	**2634**	**2990**
Energy saving to USO round-robin	**43,8%**	**45,2%**	**-**	**-**
Energy saving to user controlled submission	**50,5%**	**51,7%**	**-**	**-**

Table 7 - Federated results considering different elapsed times (51 Jobs)

Although less total cores are available on Jufit, its hardware is more modern and is able to perform a faster computation of the jobs. So, when for instance the USO controlled measurement is completed the elapsed time of the Juggle cluster is higher than on Jufit. That is as much more significant, since the more inefficient Juggle system generates substantial higher average power consumptions than Jufit. A Juggle compute node with 4 cores consumes approximately the same power than a Jufit node with 12 cores. An even worse job distribution is shown by the user submission simulation. The scheduling is similar to the equal distribution with round-robin. However, in that simulation the most CPU intensive jobs were submitted to the slower Juggle cluster, where they need more compute time which results in high energy consumptions. Certainly, user controlled submission can be more efficient when users are aware of the capabilities of the particular clusters.

The fast as possible algorithm produces numerical results which exhibit the fastest total elapsed time of all strategies, since it tries to save energy by calculating the shortest wait times for the jobs of the workload. However, the algorithm doesn't consider the energy directly so that the average power on both systems is slightly higher than with the energy aware strategy. So, with submitted workload the FIT4Green energy aware scheduling has been identified as the best algorithm in terms of energy efficiency. It submits jobs each time to the most efficient cluster as long as the user's latest job finishing time can be satisfied. This condition is very often satisfied, since the defined latest job finishing time is almost always higher than the calculated wait time of the jobs when having such short system load.

Long system utilization

Table 6 shows the results of a measurement with the same methodology but with a fivefold amount of jobs. In that scenario much more jobs have to be processed and submitted to the queues of the clusters. As a consequence the calculated wait times of the workload's jobs rise significantly. The FIT4Green fastest possible algorithm considers the wait times and distributes the jobs to the clusters in such a way that the elapsed times of both systems reach comparable levels. Juggle requires there a slightly longer time which is caused by the older CPU generation compared to Jufit. This results in energy savings of 30% related to the USO round-robin strategy.

The FIT4Green energy aware algorithm allocates again the majority of the jobs to the more energy efficient Jufit cluster which results in more different elapsed times on the clusters than with the fastest possible strategy. However, compared to the smaller workload, the ratio of jobs on Juggle and Jufit is more balanced. This is caused by the latest job finishing time parameter. It forces the energy aware algorithm to submit more jobs to the Juggle cluster, since the job wait times on Jufit would become too high otherwise. So, in case of an increasing system utilization the energy aware algorithm behaves more and more like the FIT4Green fastest possible algorithm. This behaviour depends however on the latest job finishing time. As higher as the user sets this limit as more jobs would be submitted to the more efficient cluster. If the parameter would be set to zero, the energy aware strategy would act identically as the fastest possible algorithm.

Nevertheless, the strategy is again more successful in saving energy related to the energy which is consumed by the USO round robin. In these measurements 33.4% energy could be saved with the FIT4Green energy aware algorithm. The lower saving compared to the achieved saving in the small workload is caused by the more balanced ratio of jobs on the clusters. Since supercomputing resources show in general relatively high system utilization the results of that long system utilization can be considered as more reasonable.

Different elapsed times, 255 jobs in workload	**FIT4Green fastest possible**	**FIT4Green energy aware**	**USO Round-Robin**	**User controlled submission**
Juggle [kJ]	**8042**	**7505**	**11985**	**14121**
Jobs	136	110	123	135
Elapsed Time [s]	3584	3476	5740	6987
Average Power [W]	2244	2159	2088	2021
Jufit [kJ]	**1142**	**1288**	**1216**	**1116**
Jobs	119	145	132	120
Elapsed Time [s]	3300	3702	3730	3522
Average Power [W]	346	348	326	317
Total cluster [kJ]	**9184**	**8793**	**13201**	**15237**
Energy saving to USO round-robin	**30%**	**33,4%**	**-**	**-**
Energy saving to user controlled submission	40%	42,3%	-	-

Table 8 - Federated results considering different elapsed times (255 Jobs)

III.5. Evaluation and feedback to next phases

III.5.1. Technical evaluation

Single Site scenario

An important requirement for an efficient energy aware supercomputing is the performance of the used software mechanisms. It is mandatory that jobs in a queue of an RMS can be scheduled approximately in the same speed as without an energy aware scheduling. To achieve this we have worked intensively on the performance of the FIT4Green software to enhance tasks in the optimization process. However, during the WP6 2nd pilot phase, we identify some issues which might be optimised further. These concerns in particular the process of how FIT4Green generated actions are broadcasted to the target system. We have to analyse if there are capabilities to save even more time in the range of seconds or milliseconds in the communication between FIT4Green and the RMS. The time that is needed to perform the optimization process and to send the appropriate actions to the RMS has an important impact on the whole energy consumption of the FIT4Green enabled system.

The energy saving capabilities of the testbed clusters is limited by the maximum possible ACPI state of the clusters. The hardware of both machines supports only the ACPI state S1 'standby'. However, there are ACPI states like S3 'suspend-to-RAM' and S4 'suspend-to-disk' which would enable savings of more than 90%. Especially, the S3 state would be very promising for evaluations since the current state of the system would be stored in the fast RAM which allows fast recover times when waking up the system[3]. However, an evaluation in this regard is not very likely since the hardware situation at least regarding the FZJ machines will probably not change in the remaining project period.

Federated scenario

The benchmark tests of the federated scenario show relatively high energy savings when comparing FIT4Green brokering strategies with other procedures. The measurements resulted in savings between 30 and 51%. However, these numbers have to be relativized by bringing them in the right context. The following conditions have to be kept in mind:

- The measurements and appropriate results can only be related to the available soft- and especially hardware environment of the testbed at FZJ.
- There were no more intelligent non-FIT4Green brokering software solutions available which could provide a bigger challenge for the FIT4Green algorithms.
- The measurements have been performed on a testbed cluster with an artificially engineered workload.

So, the achieved energy saving cannot be considered as a general rule for supercomputing environments. Nevertheless, for the next project phase thoughts should be given how the benchmarks could better map real environments. A first approach could be to reserve more testing time on the available clusters for enabling larger and more complex workloads.

Concerning the federated scenario it would be interesting to have testbed machines in different geographical locations providing also different PUE/CUE values. The both test clusters at the JSC are placed in the same datacentre and showed therefore the same PUE parameters. In the next phase there could be additional clusters, possibly at VTT or University of Mannheim, which should provide different CUEs then.

[3] http://www.efficient-server.eu/fileadmin/docs/reports/E-Server-Report_PartII.pdf

III.5.2. Usability evaluation

In terms of usability no particular observations could be detected. The FIT4Green software appears as very stable and easily to configure. The User Interface, in this testbed, is actually not used.

IV. CLOUD COMPUTING DATA CENTRE TESTBED

IV.1. Testbed environment and configuration

The testbed realized at HP Italy Innovation Centre Cloud Lab's premises has been generally described in the deliverable D6.2, at Chapter 4.1. As explained in that document, this testbed provides a computational environment implementing a cloud computing platform for the IaaS and PaaS layers, based upon a Lab-grade infrastructure fully resembling (in smaller scale) both the configuration and functional capabilities of actual production-grade IaaS implementations, either private or public.

As anticipated in D6.2, the testbed is split into two different data centres, feature to leverage in the current pilot phase 2 for evaluating the federated model in its different configurations. For convenience, we replicate here the picture with the physical location of the two data centres in HP Milan campus (Figure 1).

The single site test cases in pilot 2 have been run on the "Innovation Centre Lab" data centre, the same one employed for all the testing performed in the pilot 1. The federated configuration test cases have spanned also the second data centre (the one dubbed "Technology Showroom"). All the test cases have run over a cloud IaaS type platform, already presented and explained in D6.2.

Even in this pilot 2 phase, regardless the single site vs. federated configurations, this testbed keeps its own peculiarities: first and foremost a very low predictability of the instantaneous load, and potential sharp steps (up or down) in resources' utilization rate.

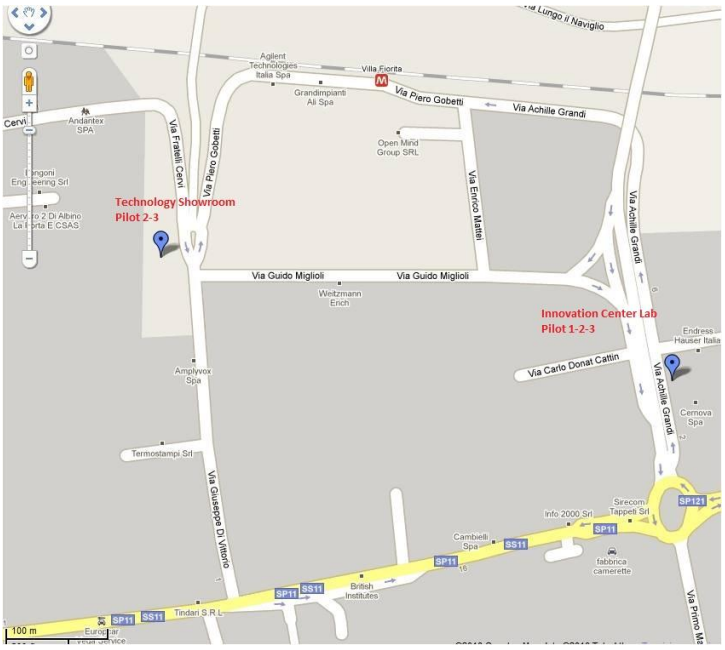


Figure 37 - Cloud testbed

Hardware configuration

The hardware equipment used in pilot phase 2 is basically the same where pilot 1 was run, but almost replicated twice in both sites taking part to this cycle. For convenience, we replicate here the description given in D6.2.

In each site, the equipment consists of two sets of equivalent ISS (Industry Standard Server) Blade servers, each set hosted inside a HP *Blade System C7000*[4] enclosure. With the setup used for this pilot, the first site (*Innovation Centre Lab*, from now on dubbed as *DC1*) bears in total 7 physical blade servers, the second one (*Technology Showroom*, from now on dubbed as *DC2*) hosts a total of 5 servers.

The two sites are interconnected through a LAN, and use a SAN device to store all data, including the virtual machine images. Inside the enclosures, the servers are interconnected through *HP Virtual Connect* modules, offering a fast internal 1GB/sec network.

The servers belong to the *HP ProLiant BL460c G6*[5] series. These servers are half-height blades, configured as in the table below:

CPU	• DC1: Dual CPU, quad-core, Intel® Xeon® E5520 2.27 GHz • DC2: Dual CPU, quad-core, Intel® Xeon® E5540 2.53 GHz8 MB L3 cache
Memory	24 GB (6 x 4 GB DIMMs)
Hard disk	Two hot plug hard drives 2 x 300 GB
Network	• Dual-port 10 gigabit Ethernet adapter NC532m • Fiber Channel bus (for Storage Area Network)
SAN	• HP StorageWorks 2024FC G2 Modular Smart Array

Table 9 - Server configuration

Power supply is bundled with the enclosure, through 6 high efficiency (90%) 1200W *HP Common-Slot* Power Supply units. Cooling is provided by 6 *HP Active Cool 100* fan units, also directly installed in the enclosures.

From an energy supply standpoint, the sites have no intrinsic special features to take into account: they have a traditional energy supply contract, neither tied to specified energy source types, nor including special clauses related to energy trade-back or similar. For the sake of getting significant results from the tests, in the final phase of measurements two different PUE and CUE values have been considered, so that the effectiveness of FIT4Green plug-in can be duly evaluated in each of the testing configurations. To make the test results as slim as possible to real cases, we decided to assign these values as if the two sites were powered by different energy providers (the two largest ones serving Milan's metropolitan area), and used the actual PUE/CUE values declared by those providers. These values are summed up in the following table:

Table 10 - Energy providers' indexes

Energy Provider	Average emission	PUE	CUE
A2A	368 g/kWh[6]	2.1	0.7728 g/Wh
ENEL	443 g/kWh[7]	1.8	0.797 g/Wh

[4] http://h18000.www1.hp.com/products/blades/components/enclosures/c-class/c7000/?jumpid=reg_R1002_USEN

[5] http://h10010.www1.hp.com/wwpc/us/en/sm/WF25a/3709945-3709945-3328410-241641-3328419-3884098.html

[6] Derived by data from http://bilanciosostenibilita.a2a.eu/it/ambiente/protezione-dell%E2%80%99ambiente/emissioni-gas-effetto-serra

Energy measurement is again performed by the HP hardware component named *iLO*[8](Integrated Lights-Out), already referenced in the traditional data centre testbed (II.1.

We don't repeat here the considerations expressed in D6.2 about the advantages of running the tests on hardware tightly fitting the typology of the testbed. Those remarks keep nonetheless fully applicable even in pilot phase 2.

Software configuration

As said, the pilot 2, as well as pilot 1, is run over a cloud IaaS platform configuration, where the cloud controller software plays the role of existing data centre automation framework. However, with respect to pilot 1, the cloud controller software has been upgraded, even though the general configuration of the testbed is very similar to the previous one ().

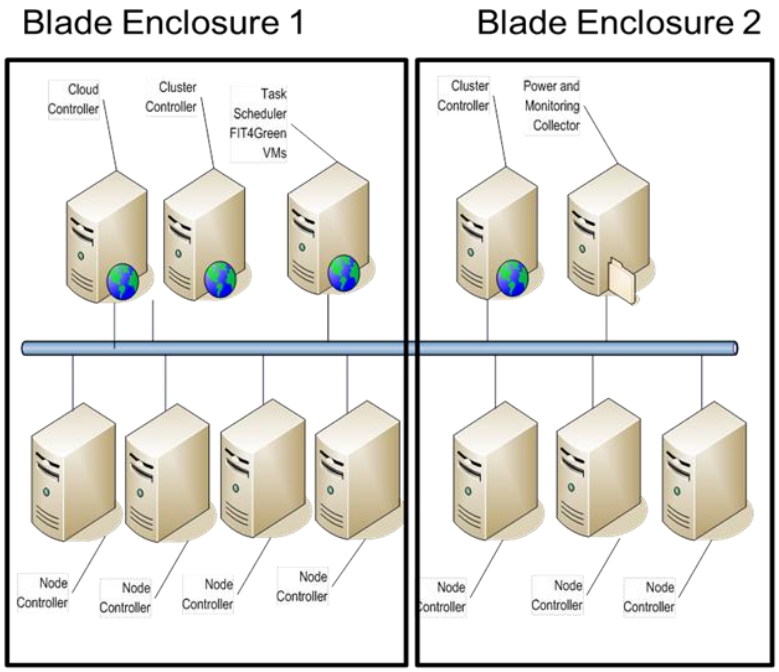


Figure 38 - Cloud testbed configuration

The new cloud management system is based on the HP proprietary software *Matrix OE*[9] (formerly known as Insight Dynamics), partially described also in the deliverable D6.1, Chapter IV.4.1, where it still bore the older naming. HP Italy Innovation Center decided to upgrade its Cloud Computing Lab infrastructure from Eucalyptus (utilized in the pilot 1) to Matrix OE, in order to have a platform more scalable, better tuneable to satisfy the needs of Lab grade tests and experiments, and offering a greater potential of integration with the whole HP Software portfolio of data centre automation products and solutions. This latter factor, in particular, is aimed at increasing the exploitation opportunities for the FIT4Green technology.

The Matrix OE platform exposes the same cloud IaaS service types available with the pilot 1 Eucalyptus based framework, first and foremost creation and termination of virtual machine instances based on pre-defined images. The testbed environment reproduces overall a typical private cloud IaaS environment. The implemented user services include

[7] Derived by data from http://www.enel.it/it-IT/azienda/ambiente/enel_ambiente/zero_emissioni

[8] http://h18013.www1.hp.com/products/servers/management/ilo_table.html?jumpid=reg_R1002_USEN

[9] http://h18006.www1.hp.com/products/solutions/insightdynamics/info-library.html?jumpid=reg_r1002_usen

delivery of computational resources, and even of storage services. Matrix OE offers its own API to access these services. Nevertheless the Hybrid Cloud Portal is fully transparent to it, and presents to its end user the same interface as it was in the pilot 1. A snapshot of the service GUI is reported in Figure 39.

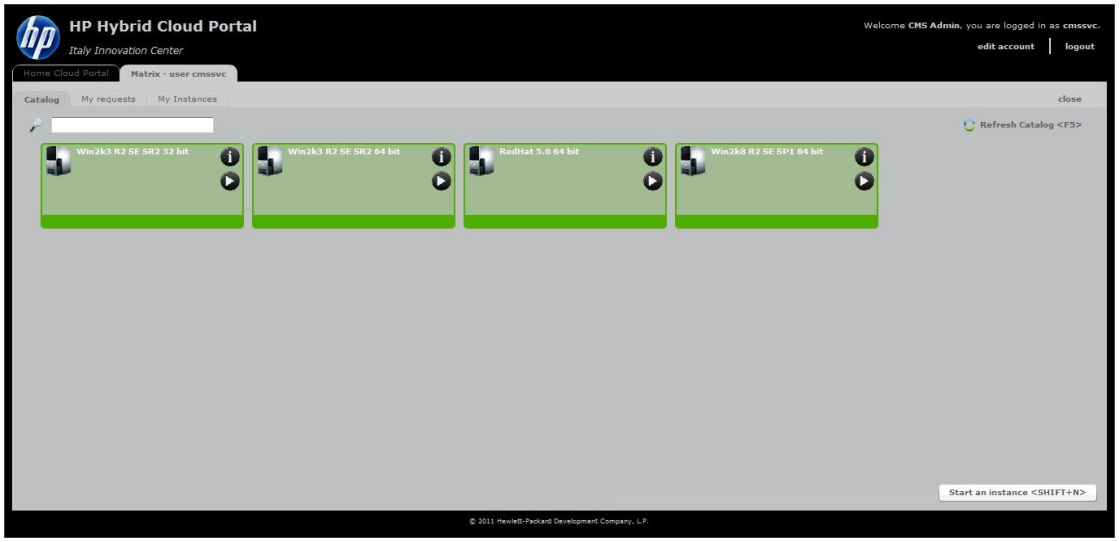


Figure 39 - HP Cloud Portal

For the sake of *FIT4Green* pilot, the HP Italy Innovation Center team integrated the Matrix OE based services with a revised COM component to allow a transparent integration of the *FIT4Green* plug-in inside the testbed itself (demonstrating once again the portability of *FIT4Green* plug-in on multiple platforms).The main updates done to the COM component are described in the deliverable D5.2 (Energy Control Plug-in for Single and Federated Site Data Centres), at chapter V.3.3.b. Just as a quick recap, the most important enhancements concern the set of information provided to the Monitor, status information handling, support to MOE platform and to live migration functionality.

The testbed software architecture consists of the same logical components as in pilot 1; of course their implementation has changed due to the platform update. For a quick recap, the components are:

1. The Cloud Controller (CLC), even dubbed Front End (FE), running Matrix OE, and implementing the single access point to the whole cloud;
2. The Cluster Controllers (CC), one for each site, managing the set of Node Controllers of that specific site, also part of the Matrix OE software suite;
3. The Node Controllers (NC), running a VMware ESX v4.0 native hypervisor;
4. The FIT4Green plug-in, and the scheduler of the workload tasks (VM creation and load generation), both deployed as virtual machines themselves on a dedicated VMware node;
5. The Power and Monitoring Collector, deployed as a VM on a separate VMWare node.

The functional description of the architecture doesn't present big changes with respect to pilot 1, so for the core description we can still refer to D6.2. The main changes concern the used hypervisor: in pilot 2, we use a VMware ESX 4.0 hypervisor, and manage Linux images (Ubuntu, RedHat).

The power and monitoring collector is still implemented through *collectd*, but the iLO functions (power on and off) are invoked through the interface available by Matrix OE.

IV.2. Testing methodology

As explained in D6.2, a cloud computing IaaS environment is by definition fairly unpredictable, and undergoes very wide variances of the instantaneous computational load swinging between zero and maximum available physical capacity. To describe the type of workload used across the tests, the same methodology of pilot 1 has been adopted, leveraging a workload profile derived by detailed usage observation and measurement of the cloud resources in a real customer site during a Proof of Concept. Then, with similar modalities, the workload simulator tool was duly setup, and the scheduler was accordingly configured.

The same isolation strategy for the testbed as in pilot 1 was used, to guarantee that test results were not corrupted by unwanted computational tasks improperly changing the workload.

IV.3. Test workload

Again, recalling what was done in pilot 1, we created a user activity profile shaped up to fully reproduce the actual cloud IaaS usage logged in a real environment, and suitable to be replayed in a controlled mode (via the scheduler), to run the full set of tests in a way granting test sequencing and repeatability according to the pilot plan's needs.

The type of user profile has no major differences with respect to the one used for pilot 1, but this time - to make the scenario even more challenging – a second workload type was generated. This second workload class came off the observation that:

- Weekend days (Saturday and Sunday) were substantially low in utilization with respect to weekdays, hence they had a favourable impact on the average power measurement over the whole week;
- Weekdays had a substantially very similar profile over the 24-hour period, in the average.

From the considerations above, it was clear that focusing the measurement on a single weekday time window, increasing the measurement resolution, was a more challenging way to test the performance of FIT4Green's plug-in, while keeping unstained the value and accuracy of the estimation. So, aside the legacy "average day for week" workload profile, we set out a second "single weekday" profile.

In D6.2 (chapter IV.3) we had described the application of a compression factor to whole measurement timescale, in order to squeeze down the overall test duration, ensure the full execution of the test plan during the weekends according to the set strategy. Running the tests in the weekend assured that the testbed was entirely to the tests, and minimized the risk of "spurious" workload staining the faithful test cases and altering the correctness of measures. This same strategy was kept phase 2, but the time compression rate was reduced from 7:1 to 2:1. This way, a 24-real time scale was squeezed to a 12-hour measurement trail, reproducing even accurately the time dependencies detected in the real user profile, and taking account the effect of quick load spikes. Time

Figure 40 shows the detected weekly load pattern, where Y axis values represent the number of concurrently active virtual machines; the red box identifies the single work day considered for the second pilot.

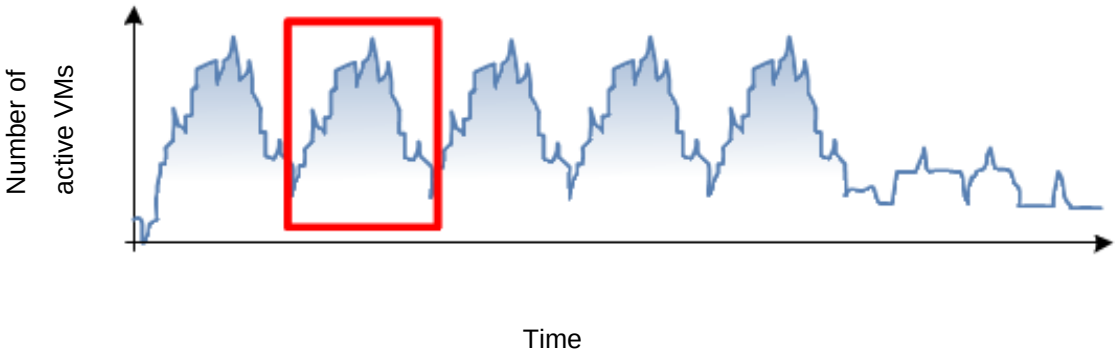


Figure 40 – Weekly load pattern

The time scale compression methodology had substantially proved its reliability during the pilot 1, thus the preliminary accuracy and fine tuning measurements didn't need to be repeated a second time.

IV.4. Numerical results

IV.4.1. Power Calculator module tuning

As described in D6.2, prior to measurement execution, a fine tuning of the Power Calculator model was repeated, to ensure a reliable and realistic prediction of the energy consumption associated to a given target distribution of the workload. The refinement took into account also the feedbacks extracted by the pilot phase 1, and resumed in D6.2 at chapter VI.3.

The extensive updated results of this pre-assessment phase can be found in the deliverable D3.2 at chapter II.2.2, hence they are not replicated here.

IV.4.2. Energy optimization tests

IV.4.2.a. Single Site Trial

The trial for a Single Site scenario has been performed using only the first site (DC1), and both types of workload (full week and weekdays). The Task Scheduler allocates virtual machines (through Cloud Controller and Cluster Controller primitives) only to the nodes in DC1; data collection for power and system monitoring, however, runs on a blade server in DC2.

Table 11 shows the results in terms of overall energy consumed by the node controllers. Due to the lab-grade configuration, with a limited amount of working nodes available, the total number of management nodes vs. actual working nodes is far too high compared to a real cloud environment. Therefore, management nodes have been omitted from the computation, to allow a clearer interpretation of the results. Three types of scenario have been taken into account and measured:

1. The FIT4Green plug-in is off (not running, no acting optimization);
2. The FIT4Green plug-in is on, the optimizer can trigger switch-on and switch-off of nodes, but not inter-node migration of virtual machines;
3. The FIT4Green plug-in is on, the optimizer can trigger switch-on and switch-off of nodes, and also inter-node migration of virtual machines (more optimization options available).

All the three scenarios are evaluated with the two types of workload. The results show as:

- with the “lighter” workload profile, the results of pilot 1 are more or less reconfirmed without migration, and improved with the migration;
- with the “tougher” workload profile, if we can exploit the migration function, results of pilot 1 are equally reached, meaning that the enhanced capabilities of the optimizer balance the reinforcement of workload conditions.

Scenario	Average Day for Week workload	Single Weekday workload
Without FIT4Green	6029 Wh	6621 Wh
With FIT4Green – no migration	4867 Wh Saving 19.2%	5938 Wh Saving 10.3%
With FIT4Green – using migration	4592 Wh Saving 23.8%	5444 Wh Saving 17.7%

Table 11 - Single Site optimization results

IV.4.2.b. Federated Sites Trial

In this case, the workload for the first data centre (DC1) replicated the single weekday case of the single site trial described above. The workload for the second data centre, instead, was scaled down by a factor ¾, and had its peak time-shifted of approximately 1/24 of the time scale (1 hour off the 24 hours scenario). The objective was again to find the best possible combination of test execution time and measure's fine grain accuracy.

The results were collected in a number of different configurations:

- Without the FIT4Green plug-in, with independent allocation of the workload on the two data centres; in other words, each chunk of workload was statically pre-assigned to one of the available sites;
- With the FIT4Green plug-in on, still with independent allocation of the workload on the two data centres;
- With the FIT4Green plug-in on, and dynamic allocation of the workload on the two data centres; when a chunk of workload needs to be started, the plug-in is queried to decide on which cluster to allocate and run it;
- With the FIT4Green plug-in on, dynamic allocation of the workload on the two data centres, and best-optimized policies; in this case, the "buffer" of free slots of each data centre cluster was decreased, capitalizing on the availability of additional resources in the other cluster.

Table 12 presents, for the different configurations, the numerical results in term of global energy consumed by the whole of each data centre's node controllers (cluster nodes), and the total for the whole federation in the rightmost column.

The ability to exploit the federation as a unique pool of resources at allocation time allows achieved saving to grow from 16.7% to 18.5%; the additional fine tuning of policies, reducing at cluster level the amount of resources to be kept free for coping with load peaks, allows saving to further grow up until 21.7%.

Configuration	**Energy consumption DC1**	**Energy consumption DC2**	**Energy consumption Whole Federation**
Without FIT4Green	6350 Wh	4701 Wh	11051 Wh
With FIT4Green and Static Allocation	5190 Wh	4009 Wh	9199 Wh Saving **16.7%**
With FIT4Green and Dynamic Allocation	5068 Wh	3933 Wh	9001 Wh Saving **18.5%**
With FIT4Green, Dynamic Allocation and Optimized Policies	4860 Wh	3785 Wh	8645 Wh Saving **21.7%**

Table 12- Federated Sites with ICT energy optimization

Figure 41 provide a graphical view of the numbers presented in Table 12.

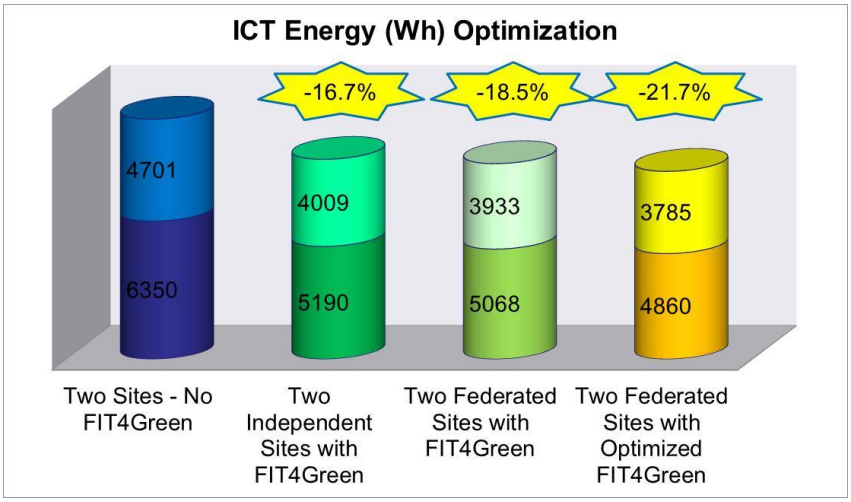


Figure 41 - ICT Energy optimization

IV.4.2.c. Federated Sites Trial – Energy vs. Emissions based optimization

In the first chunk of federated tests, the two data centres were assumed to have exactly the same characteristics in terms of energy and emissions efficiency (as in the reality, since they're indeed co-hosted in the same site).

In order to simulate the scenario with data centres having different energy and emissions characterization, the final test round has been run in two additional modes, by modifying the meta-model PUE and CUE attributes of the data centre configuration with the rationale exposed in IV.1. :

- Data Centre 1 with PUE=2.1 and Data Centre 2 with PUE=1.8 (lower PUE - more efficient); the optimization targets the total energy consumption of the federation;
- Data Centre 1 with CUE=0.772 and Data Centre 2 with CUE=0.797 (lower CUE - more efficient); the optimization targets the total emission level of the federation.

Table 13 reports the final test results for the different configurations; the goal is to evaluate the effectiveness of the optimizer when dealing with a federation of data centres with different characteristics in terms of both energy and emissions efficiency.

Configuration	ICT Energy DC1	ICT Energy DC2	Total Energy	Total Emissions
Without FIT4Green	19050 Wh	14103 Wh	65390 Wh	25.94 g CO2
With FIT4Green • optimize ICT Energy • ignore PUE and CUE	15486 Wh	11663 Wh	53514 Wh Saving 18.16%	21.25 g CO2 Saving 18.10%
With FIT4Green • optimize Total Energy • considering PUE	14188 Wh	12953 Wh	53110 Wh Saving 18.78%	21.27 g CO2 Saving 17.99%
With FIT4Green • optimize Emissions • considering CUE	17381 Wh	9624 Wh	53823 Wh Saving 17.68%	21.08 g CO2 Saving 18.72%

Table 13– Federated Sites, total energy and emissions optimization

It's worth to notice that, when FIT4Green optimizes the total energy, the saving ratio is 0.6% greater than what we got with ICT energy optimization. This shows how the plug-in can capitalize on the energy efficiency difference between the two data centres, by unbalancing the load as much as possible towards the most efficient data centre (DC2, that has a better PUE value). When optimizing the emissions, on the contrary, the load is biased towards DC1, since it has better emissions efficiency (lower value of CUE), and the total improvement is also 0.6% better than the ICT energy optimization case.

Figure 42 and Figure 43 provide a graphical view of the numbers presented in Table 13.

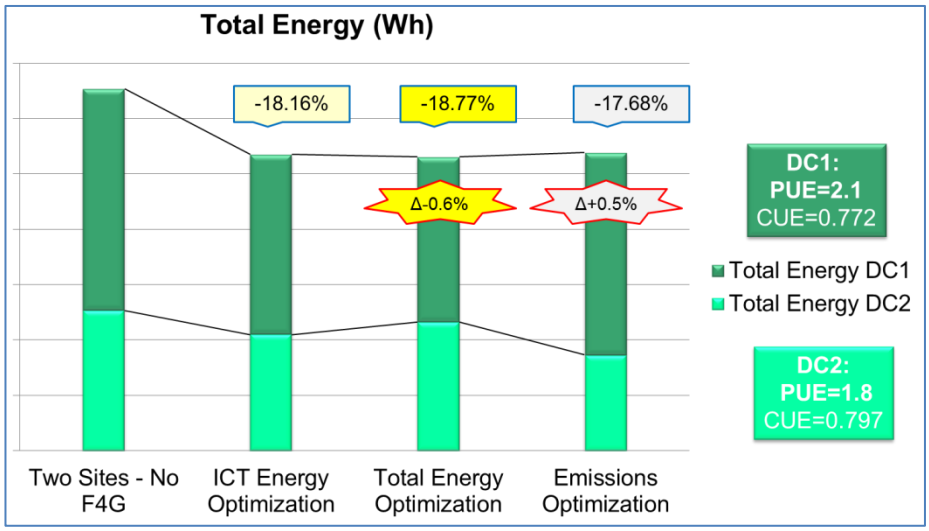


Figure 42 – Total energy and emissions optimization

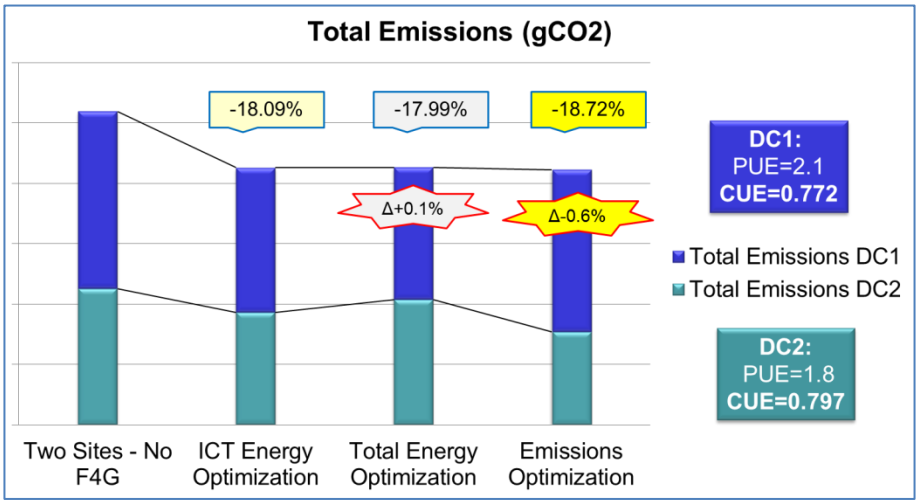


Figure 43 - Total energy and emissions optimization

IV.4.3. Plug-in's own consumption

To duly complete the outline of achieved results, as we did in the pilot 1 and presented in the deliverable D6.2, we need to ensure that the *FIT4Green* plug-in, when activated, doesn't induce itself an additional energy consumption whose value can somehow be significant with respect to the amount of energy saved by its application. So, even in pilot 2, we conducted a dedicated test on the physical node where the plug-in is running, to measure its own power consumption. The result is summarized in Table 14, where it's compared versus the same value as measured in the pilot 1 (see D6.2, chapter IV.4). The data taken into account is the one of the federated case.

FIT4Green Plug-in Power Consumption	**Test Phase 1**	**Test Phase 2**
Node hosting the FIT4Green VM (one single server)	+ 6 W	**+24.9 W**
Total ICT Energy saving (all servers)	185 W (1 data centre)	**512 W** **(2 federated data centres)**

Table 14 - Plug-in energy cost

It can easily be observed that the power consumption induced by the plug-in has increased by a factor 4, but at the same time the energy saving obtained thanks to the plug-in has increased of a factor 3 (approximately). Considering that, in absolute terms, there were two orders of magnitude between the compared values, we can conclude that:

- The energy consumption accountable to the plug-in is still negligible with respect to the amounts of energy savings obtained by means of its action; and
- even if the testbed is strongly scaled up, it's absolutely unlikely that the plug-in can ever have a negative side effect bale to balance the impact of its core function

IV.5. Evaluation and feedback to next phases

IV.5.1. Technical evaluation

Workload simulation and testing methodology

The pilot phase 2 reaffirmed the effectiveness of the approach used in terms of workload generation, test management and supporting tools. The methodology was applied again, with significant changes in the parameters used, and even this time it proved reliable, flexible and repeatable. In this second pilot, the measurement period was squeezed to one day only, with a double effect:

- Weekend days were fully taken out of the measure; in those days, the actual load was quite low, hence, when computing average or accumulated values, it had an overall downgrading effect on the final numbers;
- Squeezing from one-week to one-day windows, whilst not losing anything in terms of accuracy of the real profile replication, allowed to better unveil spottier peaks and shorter-term events conditioning the measure.

Another improvement done to the measurement context was the removal from measures of the nodes hosting cloud management and control components, to let in only nodes hosting actual load (virtual machines). This change made the measure environment closer to a real one, due to the following reason: in a real environment, the ratio between active nodes and management nodes is normally very high (>100), consequently the weight of management nodes on the overall power consumption is very low. In our lab environment, instead, due to the lower scale of the whole system, this ratio is not negligible, and the bias of management nodes on the overall measured values was too high. Thus, excluding management nodes from the measure allowed to obtain more significant values, which can be considered equally valid even if the system is scaled up towards a real production configuration.

General evaluation of numerical results

In pilot cycle 2, as presented in IV.4.2. above, different types of configurations were evaluated:

- Single site, to evaluate the performance of the improved version of FIT4Green plug-in in a similar configuration to pilot 1;
- Federated sites, that was brand new to pilot 2, in two different flavours:
    - Targeted optimized index is ICT energy consumption, right as in the single site case;
    - Targeted optimized index is total federation energy consumption or emission level, to evaluate the effect of the plug-in when the federated sites have different energy characteristics, expressed by the two metrics PUE and CUE.

We separately look at the obtained results for the three different cases.

Single site

As already explained, the numerical results we took into account were derived by applying a tougher workload profile, excluding the weekend days. Nonetheless, to have one set of measures fully comparable to the ones obtained in pilot 1, a measure round was executed with full-week load profile, and the same optimization capabilities available in pilot 1, i.e. without the ability to migrate virtual machines (only switching nodes on/off). We can see that, in totally comparable conditions, the plug-in basically reaffirms the optimization result got in pilot 1 (a slight improvement from 17.98% to 19.2%). It's worth underlining as this improvement is also coming from the refinement done to the Power Calculator component, as described in deliverable D3.2,

If, keeping all the conditions unchanged, we move to the one-weekday load profile, the optimization goes down to 10.3%, as we could expect given the elimination of energy-mean weekend days.

Once we add the ability to use virtual machine migration, the optimization ramps up to 17.7% with the one-weekday load profile, reaching the same values of pilot 1, and gets up to 23.8% with the same full-week profile of pilot 1.

So, to wrap-up the results for single site configuration, the upgraded plug-in confirms a satisfactory average optimization rate slim to 18%, even when measured in less favourably defined boundary conditions, thanks to its improved capabilities, especially the exploitation of virtual machine move. With test profiles alike the ones of pilot 1, the effectiveness of the plug-in overmatches what measured in pilot 1, and is beyond the 20% general objective set for the FIT4Green project.

Homogeneous federated sites– ICT energy results

In this scenario, we measure the ICT energy consumption saved when the optimization exploits the presence of more than one data centre site, two in the case of our tests. The first test group in this scenario was performed considering the two sites homogeneous in terms of energy efficiency, i.e., assigning them equal PUE and CUE values; therefore these parameters are not taken into account by the optimizer's policies, and the only measured indicator is the ICT energy.

The measured results stay in a range of about five percentage points, spanning different cases where at each step we augmented either the optimizer's smartness, or the tightness of the coupling between the federated sites. The least efficient case, which we took as a low-end reference, occurs when the two sites, more than a real federation, are actually two operationally separated sites, and FIT4Green plug-in runs on each of them in single-site mode. This way, the plug-in doesn't influence the choice of the site where a given chunk of workload is allocated, but only decides on which node inside a given site the workload is going to be placed. As we could expect, this kind of sum of single-site configurations offers an optimization rate not particularly high, and we don't see a breakthrough with respects to a pure single-site scenario.

A first step forward is done when we really federate the two sites. In this scenario, FIT4Green plug-in can look at the federation as a whole, and, when deciding where a certain load should be allocated, can choose both the site and the node. This allows improving the optimization, since we can balance the load on the two sites, besides on the different nodes within each site. This way, we can get up to an average of 18.5%, which is already a bit better than what we got in single sites.

A second improvement consists in playing with the policies employed by the optimizer, to further exploit the availability of multiple sites. As described in the deliverable D4.2, the optimizer keeps a "buffer" of available resources in each data centre, which are not taken into account when deciding where a new load is allocated. In the cases exposed so far, the policies set these resources buffers' size independently on each of the data centres. Nonetheless, if we look at the federation as a whole, the approach can be optimized in a more dynamical sense: instead of "statically" setting the buffer size on each site, we can set

a unique, total buffering level, to be kept on the federation as a whole. In other words, we can make the best possible usage of the available capacity, by allocating also the buffer resources on the site where it's more convenient from an energy standpoint. This approach allows a further improvement, bringing the optimization rate to 21.7%. Related to the 17.7% of single site case, this result shows a 4% increase, that's the additional value we can gain from the availability of two data centres, handled by the FIT4Green plug-in with a holistic approach.

Heterogeneous federated sites – Total energy optimization

In the previous configuration, FIT4Green plug-in could exploit the multiplicity of data centre sites to achieve extra savings with respects to the single site case. From now on, we introduce an additional element that the policies can leverage to increase the saving, i.e., the differentiation of energy (PUE) and emission (CUE) efficiency between the federated sites.

With these additional control parameters, we end up with three possible configurations for the optimizer, since its policies can be tuned to optimize the ICT energy again, the total energy, or the emissions. For each of these three cases, we can then measure the achieved improvements on the same three parameters (as illustrated by Figure 42 and Figure 43), with a total of nine possible cases. However, measuring the ICT energy makes sense only when the optimized parameter is ICT energy itself: it's a logical subcase of total energy and emissions, which are actual final indicators. Hence, the ICT measurement is fully encompassed by what we presented in the previous sub-paragraph.

Let's first focus on total energy improvement measured in the three optimization tunings. We can observe that the achieved saving doesn't change so much dependently upon the type of optimization chosen. There's only a 1.09% gap between the best measured case (total energy optimization, equal to 18.77%) and the worst one (emission optimization, equal to 17.68%). Thus, even if we tell the optimizer to try unbalancing the allocation towards the site with better PUE, we gain an additional 0.6% with respects to the ICT energy optimization, and a 1.1% compared to an optimization biased the other way around (the site with better PUE is the one with worse CUE). In the pilot 3, it could be worth digging deeper into these numbers, to understand if this gap may be widened somehow, by acting on the policies or by changing any boundary conditions, starting from a larger difference between PUE/CUE coefficients of different sites.

Heterogeneous federated sites – Emission optimization

This is the specular case of the previous one. We again run the three types of optimization, but this time measure the result in terms of emissions rather than total energy saving. In this case, the result span is even narrower, just 0.73%, so the very same remarks are fully valid for this case too.

Summary

As a general evaluation, we can say that pilot phase 2 provided good results, in line with the overall FIT4Green's objectives. The new, improved version of FIT4Green's plug-in confirmed the energy saving achieved in pilot 1 for single site configuration, even though the testing conditions were less favourable and by themselves caused a squeeze of the results. By introducing federated sites, so giving to the FIT4gGreen plug-in additional elements to leverage, we improved the values up to 4 percentage points (in relative terms, 22.5% better than single site), and basically hit the general goal set at the beginning of the project. Last but not least, notwithstanding the increased footprint, the additional energy consumption induced by running the plug-in is still absolutely negligible compared to the amount of energy it can save in the data centres.

As possible improvements to pursue in the third and final pilot, we can try to increase the marginal saving achieved when the optimization is tuned on a specific metric or the other. The results, though good in absolute terms, look a bit "flat" across the different optimization patterns. We can investigate if and how we may take better advantage of specific

optimization patterns, or if it's just a matter of efficiency value spread for the available data centres.

IV.5.2. Usability evaluation

On this specific point, there is no major feedback compared to pilot 1. As explained in D6.2, in the cloud testbed the FIT4Green UI is more marginal than in other testbeds, and the operator intervention is more limited. A positive acknowledgement goes to the new data logging UI, in line with the evaluation of the other testbeds. The setup and installation process of the plug-in also proved to be better and smoother than in pilot 1.

V. GENERAL CONCLUSIONS

V.1. Technical conclusions

Generally speaking, the numerical results of the second pilot cycle represent a step forward towards the achievement of the general objectives set out by FIT4Green. As we had expected, the introduction of the federated case, hence the opportunity to capitalize the availability of multiple data centre sites, brought about an overall increase of the achieved energy saving. The 20% saving milestone was caught up in all three testbeds, in test configurations which can be considered a reliable replication of real operating conditions for each testbed type.

To reinforce this evaluation, the testing conditions have been intentionally made harsher, since the initial results often were too good, and each testbed put an effort to understand where the boundary conditions had been settled in a too favourable way. With different modalities, this happened in all the three testbeds, showing that a 20% saving can be reasonably considered an average target, and in especially favourable conditions it can even be overmatched.

The result dispersion among the testbeds has narrowed with respects to pilot phase 1, especially if taking into account the most finely tuned configurations and the most significant final results. This was one of the goals of pilot cycle 1, and it's evident as now there is a better alignment among the different computing styles. The most significant numerical improvement came from the HPC testbed, which had shown lower results compared to the traditional and cloud data centre cases.

As an additional general observation, in pilot phase 2, and thanks to the experience acquired in pilot 1, the testing methodology, including usage of tools for synthetic workload generation and test execution handling, once more proved to be a valuable asset in support of the whole test cycle execution.

V.1.1. Single site configurations

The tests confirmed somewhat the results got in pilot 1 for the three testbeds. This is not surprising, since the improvements implemented to the FIT4Green plug-in were substantially focused on addressing the federated case. Nevertheless, it can be considered positive that adding all the federation related features did not strain at all the single-site performance. Actually, in the case of cloud testbed, the single-site main results were obtained with conditions more challenging to the plug-in than those during pilot 1; when normalized to the same conditions of pilot 1, the test gave better results.

V.1.2. Federated site configurations

In this second pilot, while introducing the federation concept, we wanted to evaluate the plug-in's behaviour in configurations not faithfully replicating each other in terms of compared characteristics of the federated sites. The sites can have characteristics tightly similar to each other; in this case, the federation is a homogeneous assemblage, where the saving increment comes from the availability of more resources, a factor in itself enabling a more energy optimized distribution of the total workload. On the other hand, the sites can be different in some of their characteristics, and this breed can offer additional opportunities to take advantage of when deciding the placement for a workload piece.

Accordingly, for our second pilot, we identified a couple of possible differentiation factors among federated sites, to get a better insight of how each specific factors could by itself influence the effectiveness of FIT4Green's plug-in. The two main differentiation factors we figured out for data centres were energy efficiency and computational power. In traditional and cloud computing testbeds, we focused on data centres where computational power was very similar, differentiated by their PUE and CUE values. This way, we could

investigate FIT4Green's plug-in effectiveness in leveraging the major efficiency of certain sites to save energy. On the contrary, in the HPC testbed, we exploited the presence of two clusters fairly unbalanced in terms of computational resources, removing any difference in own energy efficiency, to capture how this performance spread influenced the plug-in's behaviour and obtained results. This strategy allowed to collect interesting data on the two types of scenario, as the individual testbed sections explained in more detail.

The tests in traditional data centre testbed pointed out some imperfect behaviour, eventually accountable to the structure of the SLA, which in this testbed caused side effects (see II.5.1.). This observation generated a feedback to WP2 (see below), for its research work targeting green SLAs. In general, the results from this testbed suggest, as a future research extension, that we should delve deeper into the relationship between SLA parameters and plug-in performance.

The HPC testbed, as stated above, provided interesting indications about the plug-in's behaviour in presence of a significant performance unbalance between the two federated sites. Moreover, in this testbed, the system utilization level (amount of running workload) has a direct and prominent impact on the plug-in outcomes: the achieved energy saving is sharply decreasing with the increase of system utilization. Such an effect, better explained in III.4.2. above, is peculiar of the HPC style (dependence on time related parameters, like the latest job finishing time). It was better underlined by running two separate sets of measures, with a 5:1 ratio in the generated system utilization, where the achieved saving showed a 35% decrease.

The third pilot phase will try to better drill down these HPC aspects, for instance by measuring on a longer time window with more complex workload. Also, if time and resource constraints allow it, a heterogeneous federation test for the HPC scenario will be taken into account, to check the impact of energy differentiation on this testbed too.

The cloud computing testbed showed a steady progress with respects to pilot 1, leveraging the enhanced functionality of the automation environment (virtual machine migration), and the improvements of the plug-in. Looking towards pilot phase 3, there could be room for further progress, since, at a first glance, the single/federated result fork in this testbed appears less wide than in the others. This testbed took again the task of (successfully) ensuring that the energy cost of running the FIT4Green's plug-in is negligible when compared to the amount of achieved savings.

V.2. Usability conclusions

Plug-in's user interface

The user interface showed some improvement, appreciated especially in the traditional data centre testbed. For that testbed, a prominent new feature was displaying the suggested action list, and providing support for its active acknowledgement. This feature was important in a testbed where the operator's manual control has a stronger role than in others.

Also well working and appreciated was the new statistical pane, requested by WP6 partners after pilot phase 1. It proved well working and usable, and had no perceivable impact on the plug-in performance or on its resource availability.

In terms of plug-in's deployment and configuration, the setup and configuration procedure (other feedback from pilot 1) made progress, becoming smoother and easier to use. The ISO image creation worked out, and proved to be better than in pilot 1. A less strong point, according to users, is still the meta-model authoring interface, not considered enough user-friendly yet.

V.3. Main feedbacks to next phase

WP2

- (interesting to WP4 as well) SLAs must be explicit and not implicit; every parameter must be visible and duly tuneable as such, the plug-in should not embed any hidden constraint/condition;
- We should introduce an additional requirement for the optimizer, to avoid cases where the answer to a workload allocation request is “no answer” (that is, no suitable hosting node was detected). In such situations, there must be a secondary path, which can be for instance either handing over to a manual operator decision, or bypassing the standard reasoning and accepting a less optimized allocation.

WP3

- (possibly interesting to WP4 as well) Node start-up and shutdown latencies should be revised, to avoid “overreaction” effects. Interesting if they could be tuneable;

WP4

- The tuneable optimization policies could be refined in some aspects. For instance, moves of virtual machines are decided even when the related saved energy would not really be worth the move. Setting a threshold criterion could help to avoid potential switchback effects;
- Sometimes, the virtual machine moves suggested by the plug-in seem more targeted at fine tuning the existing balance than at turning nodes off, which should be the first priority in basically any case;
- Comparing energy-based versus emission-based optimizations, the result fork looks a bit narrow: the effect of choosing a certain optimization pattern instead of another would supposedly be more evident in the achieved results; a similar remark can be done regarding the energy efficiency (PUE/CUE) inter-site delta: when this delta increases, the corresponding increase of saving seems a bit scarce. We should investigate if tuning the optimizer can exploit more these site differencies;
- Optimization could benefit from having federation-wide policies; one example can be the resource usage limit set by the policies, currently enacted for each data centre concurrently; if we had a unique, global federation limit, treating the federation’s resources as a whole, the plug-in could pick some actions which to date are out of its constraints;

WP5

- Confirming a feedback coming from pilot phase 1, an improvement area for pilot 3 can be the XML Editor, either through editor upgrades, or by adding a kind of automated/guided meta-model writer.
- In the HPC testbed case, the PROXY/COM components should possibly be faster in passing on the actions to the automation framework (the queue scheduler in that case).


FIT 4 GREEN