COMPAT
Computing Patterns for High Performance Multiscale Computing

# D6.1 Report on the Assessment of Operational Procedures and Definition of the ComPat Operational Model

| | |
|---|---|
| **Due Date** | Month 6 |
| **Delivery** | Month 9 |
| **Lead Partner** | BADW-LRZ |
| **Dissemination Level** | Public |
| **Status** | Submitted |
| **Approved** | Executive Board yes |
| **Version** | V1.02 |

This project has received funding from the *European Union's Horizon 2020 research and innovation programme* under grant agreement No 671564.

# DOCUMENT INFO

| Date and version number | Author | Comments |
|---|---|---|
| 28.04.2016 v0.1 | Siew Hoon Leong | First version |
| 03.05.2016 v0.2 | Helmut Heller | Corrections |
| 04.05.2016 v0.3 | Siew Hoon Leong | Additional sections |
| 06.05.2016 v0.4 | Siew Hoon Leong | Added functional areas section |
| 09.05.2016 v0.5 | Helmut Heller | Corrections |
| 06.05.2016 v0.6 | Siew Hoon Leong | Added various subsections |
| 07.05.2016 v0.7 | Siew Hoon Leong | Changes according to comments of HH and PSNC |
| 07.05.2016 v0.8 | Stephan Hachinger | Re-working of some sections |
| 07.05.2016 v0.9 | Stephan Hachinger | Re-added comments of SHL (v0.7) |
| 17.05.2016 v0.91 | Stephan Hachinger | Added comments of WP4 (DS) |
| 20.05.2016 v0.92 | Stephan Hachinger | Hartree Centre updates. |
| 06.06.2016 v0.93-v0.95 | Helmut Heller,<br>Stephan Hachinger | KPI refinements and other updates |
| 22.06.2016 v0.96 | Stephan Hachinger | Changes following internal refereeing comments [MP] |
| 24.06.2016 v0.97 | Stephan Hachinger | Changes following internal refereeing comments [TP] |
| 27.06.2016 v0.98 | Stephan Hachinger | Internal consolidation revision |
| 27.06.2016 v0.99-v1.0 | Stephan Hachinger, H. Heller | Corrections |
| 30.06.2016 v1.01-v1.02 | Stephan Hachinger | Final corrections / work on announced downtime process |

# CONTRIBUTORS

The contributors to this deliverable are:

| Contributor | Role |
|---|---|
| Siew Hoon Leong | WP6 leader and main author of this deliverable |
| Helmut Heller | Member of steering committee |
| Stephan Hachinger | WP6 contributor |
| Dirk Schubert | WP4 contributor |

## TABLE OF CONTENTS



## LIST OF FIGURES

## LIST OF TABLES

# 1 Executive Summary

The ComPat Work Package 6 (WP6) provides an Experimental Execution Environment (EEE) for executing multiscale simulation codes and evaluating their performance in terms of speed and power consumption (energy-to-solution). The codes to be executed, profiled and enhanced are instantiations of ComPat computing patterns and serve concrete scientific use. In order to build the EEE, a set of distributed, heterogeneous computing resources from the ComPat consortium is integrated using specifically adapted middleware tools.

In this document, an operational model for ComPat (or, more specifically, for the EEE) is defined. We first discuss the functional areas of the EEE, i.e., the scope in which the operational model will be applied (Section 2). In Section 3, we discuss our actual operational model, describing its two processes – integration and operation – and the respective procedures. These procedures are already being used to select and operate the first components within the EEE. In Section 4, we define the mechanisms for quality assessment, and in Section 5, we draw conclusions from our work up to now.

Already in the starting phase of the project, the operational model defined has proven valuable for the integration of tools. In order to collect and share the information needed in the integration and operation processes, a data collection point in Google Drive[1] has been set up, which also serves documentation purposes. We have successfully deployed the middleware QCG[2] on PSNC and LRZ resources, despite vast differences in the usage policies and the production environments. The monitoring software Nagios[3] has been installed on a dedicated VM. QCG, having advanced job and resource-management capabilities, provides a unified interface to users of distributed supercomputers, and thus had already been selected as the grid middleware to be used before the commencement of the project. Nagios, a widely-adopted monitoring software, is used as the central monitoring system; integration with QCG is fast and reliable, as Nagios modules (probes) for testing QCG already exist. Nagios helps us to oversee the reliability and availability of the EEE, and thus to flexibly adapt operational procedures if needed. For making quantitative assessments of the EEE performance, we define KPIs to be calculated from the Nagios monitoring database. First actual KPI values will be reported in the next WP6 deliverable.

[1] https://www.google.com/drive
[2] http://www.qoscosgrid.org
[3] http://www.nagios.org

# 2 Functional Areas

Six functional areas of WP6 have been defined in the ComPat proposal. We now discuss our work in each of these areas in some further detail. In this way, we make clear where the operational model (defined below – Section 3) applies.

## 2.1 Authentication, Authorisation, and Certificate Management

We have opted for the use of X.509 certificates and proxy certificates[4] as a basis for user authentication. The use of certificate-based authentication and authorisation methods has become a widely adopted standard to access European e-Infrastructures, e.g. PRACE[5] and EGI[6]. The EEE of ComPat partly leverages on such European resources (e.g. PLGrid[7] as part of EGI) and thus aims at following their standards. In the same spirit, QCG naturally supports certificate-based authentication.

The registration of a user and his certificate at a ComPat computing resource is done via a request in the form of an e-mail, with a standardised account request form (available on our Google Drive) attached.

## 2.2 Virtual Organisation (VO) Management

VO Management is a useful concept to make users from geographically distributed institutions appear as members of a single entity (the VO) in the context of a project. VO membership usually implies a common access to computing resources, without having to grant rights to individual users.

Currently, however, the usage of the EEE is limited to ComPat users. Hence, the number of user and service accounts is small (some 10-30), making VO management an unnecessary complication. Thus, up to possible further notice, we will neither manage nor use VOs. If the EEE gains a significant amount of additional users from different science domains, one or more ComPat VOs may be set up to manage users and also to distinguish users from different science areas.

## 2.3 Data Management and Privacy

Each resource within the EEE uses their own data-management system; for efficient data transfer from one to another resource, GridFTP (as usual in European e-Infrastructures) will be utilised.

With regards to research data, but also for data collected for EEE management (e.g. monitoring data), privacy is a critical concern – for both resource providers and users. WP6 ensures that local security and data-privacy policies of resource providers are supported and respected by the EEE tools. As an example, QCG logs at LRZ will be deleted after 7 days to comply with the German TMG/TKG

[4] https://en.wikipedia.org/wiki/X.509; http://toolkit.globus.org/toolkit/docs/6.0/gsic/key/index.html
[5] http://www.prace-ri.eu
[6] http://www.egi.eu
[7] http://www.plgrid.pl

laws and with the respective BGH jurisdiction (Az. III ZR 391/13). When integrating and operating critical components, WP6 makes sure that the EEE remains compatible with security and privacy regulations. Furthermore, WP6 informs EEE users of the respective regulations in place at the various EEE computing resources.

## 2.4 Infrastructure Monitoring

Infrastructure monitoring is a crucial function of WP6 within the operation process (refer to Section 3.1.2). In order to enable the successful operation and assessment of the EEE, it is necessary to have an overview of the infrastructure status. To this end, we have set up Nagios monitoring as an important prerequisite for quality control within the operational model (Section 4).

## 2.5 Energy Consumption Monitoring and Profiling

As the cost of energy becomes a dominant factor in the operation cost of computing centres, the ability to evaluate energy consumption of a simulation, and to subsequently optimise energy usage, is an important design goal for the EEE. Tools/software for this purpose will be selected and integrated; as first such tool, we will use “Allinea Energy Pack” for generating energy profiles of ComPat applications.

## 2.6 Installation and Configuration

As a result of WP6 developing the functional areas 2.1-2.5, Software and Tools have to be integrated within the EEE and operated. New computing resources have to be added to the EEE, which means that we apply the EEE standards to them, and install missing tools and software. To all these purposes, the ComPat Operational Model, which we describe in Section 3, provides the necessary integration and operation procedures.

To facilitate the development of the EEE, a set of common configuration policies are established at LRZ, STFC and PSNC, the providers initially contributing resources to the EEE:

(i) **Common directory** – On each EEE computing resource, a directory shared among all users has to be made available for the installation and configuration of required software and tools. This common directory must be accessible from login and compute nodes, and must have enough capacity to hold shared project software. The detailed information for each resource is shown in Table 1.

| **Resource** | **Disk quota** | **Location** |
|---|---|---|
| SuperMUC (LRZ) | 1 TB | $WORK/di25pul |
| STFC machines (NCD, BGAS, BlueWonder) | tbd. | tbd. |
| Inula (PSNC) | No quota (43TB disk) | $PLG_GROUPS_STORAGE/plggcompat |

| | | |
|---|---|---|
| Eagle (PSNC) | No quota (121TB disk) | $PLG_GROUPS_STORAGE/plggcompat |

Table 1: Common directories of EEE computing resources.

(ii) **Granted computing quotas (core-hours and individual storage)** – Each computing site has its own policies in granting CPU hours and disk space to the project. The ComPat WP6 members at each site support users with the necessary formalities. In urgent cases, a user can be granted an account for limited software testing without further formal procedures at least on some EEE sites (e.g. LRZ).

The runs of multiscale applications within ComPat will first take place on the site best suited for the respective code; however, with time, applications are expected to show a decent level of portability. As codes mature, and so does the EEE, WP6 will thus come up with an administrative/technical concept for distributing applications among the different resources.

(iii) **Accounting** – Accounting information will be extracted from the local accounting systems of each resource provider. Table 2 shows the methods to access this information on each EEE computing resource.

| **Resource** | **Access to accounting information** |
|---|---|
| SuperMUC (LRZ) | Summary given upon interactive login; further information can be requested from administrators. |
| NCD (STFC) | Request to STFC Helpdesk. |
| BGAS (STFC) | Request to STFC Helpdesk. |
| BlueWonder (STFC) | Request to STFC Helpdesk. |
| Inula (PSNC) | Available via PLGrid accounting portal (http://accounting.plgrid.pl). |
| Eagle (PSNC) | Available via PLGrid accounting portal (http://accounting.plgrid.pl). |

Table 2: Accounting information of EEE computing resources.

(iv) **Required "scheduler functionalities"** – The functionalities of "advance reservation" and "frequency scaling" (i.e. setting the CPU frequency), typically provided in the scope of scheduling systems, are important for some runs of experimental software within the EEE. The availability of these functionalities on different systems is summarised in Table 3, which serves as a guideline on where to run software depending on these functions.

| **Resource** | **Advance reservation** | **Frequency scaling** |
|---|---|---|
| SuperMUC | Yes, but requires notification six months in advance. | CPU frequencies are set by the on-demand governor; the maximum frequency (for cores under load) can be set at submission time within reasonable limits. |
| NCD (STFC) | Yes, but requires notification five days in advance. | Implementation possibilities are discussed. |

| | | |
|---|---|---|
| BGAS (STFC) | Yes, but requires notification five days in advance. | No. |
| BlueWonder (STFC) | Yes, but requires notification five days in advance. | Yes, via LSF batch Scheduler. |
| Inula (PSNC) | Yes, via QCG; may be limited to specific partitions. | For now dynamic changing of the frequency is not supported. In the future it can be considered, after operational / security audits of the proposed solution. |
| Eagle (PSNC) | Yes, via QCG; may be limited to specific partitions. | For now dynamic changing of the frequency is not supported. In the future it can be considered, after operational / security audits of the proposed solution. |

Table 3: Scheduler functionalities on EEE computing resources.

# 3 Operational Model

We define an operational model for ComPat, allowing the project partners to manage the EEE according to the ISO 20000 Plan-Do-Check-Act scheme[8] (PDCA cycle).

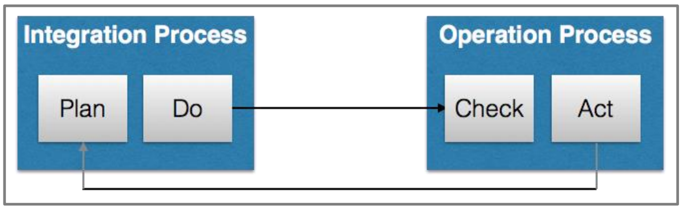



Figure 1: EEE operational model: basic PDCA scheme.

The basic bricks of the EEE Operational model are the two processes of integration (activities “Plan” and “Do”) and operation (activities “Check” and “Act”) displayed in Figure 1. In Section 3.1, we will lay out what this means with respect to software (including software providing services, e.g. QCG) being added to the EEE and being operated. In Section 3.2, we describe how the PDCA cycle is implemented when adding new EEE computing resources. Possible integration and operation tasks for EEE components not falling in one of these two categories can be carried out analogously.

## 3.1 Software Integration / Operation

When new software packages are to be installed in the EEE, we first assess their suitability and then install and monitor them. As a last step, possible problems are corrected. Below, we describe how we formalise these activities in our PDCA-based integration/operation scheme.

[8] https://www.iso.org/obp/ui/#iso:std:iso-iec:20000:-1:ed-2:v1:en

### 3.1.1 Integration Process

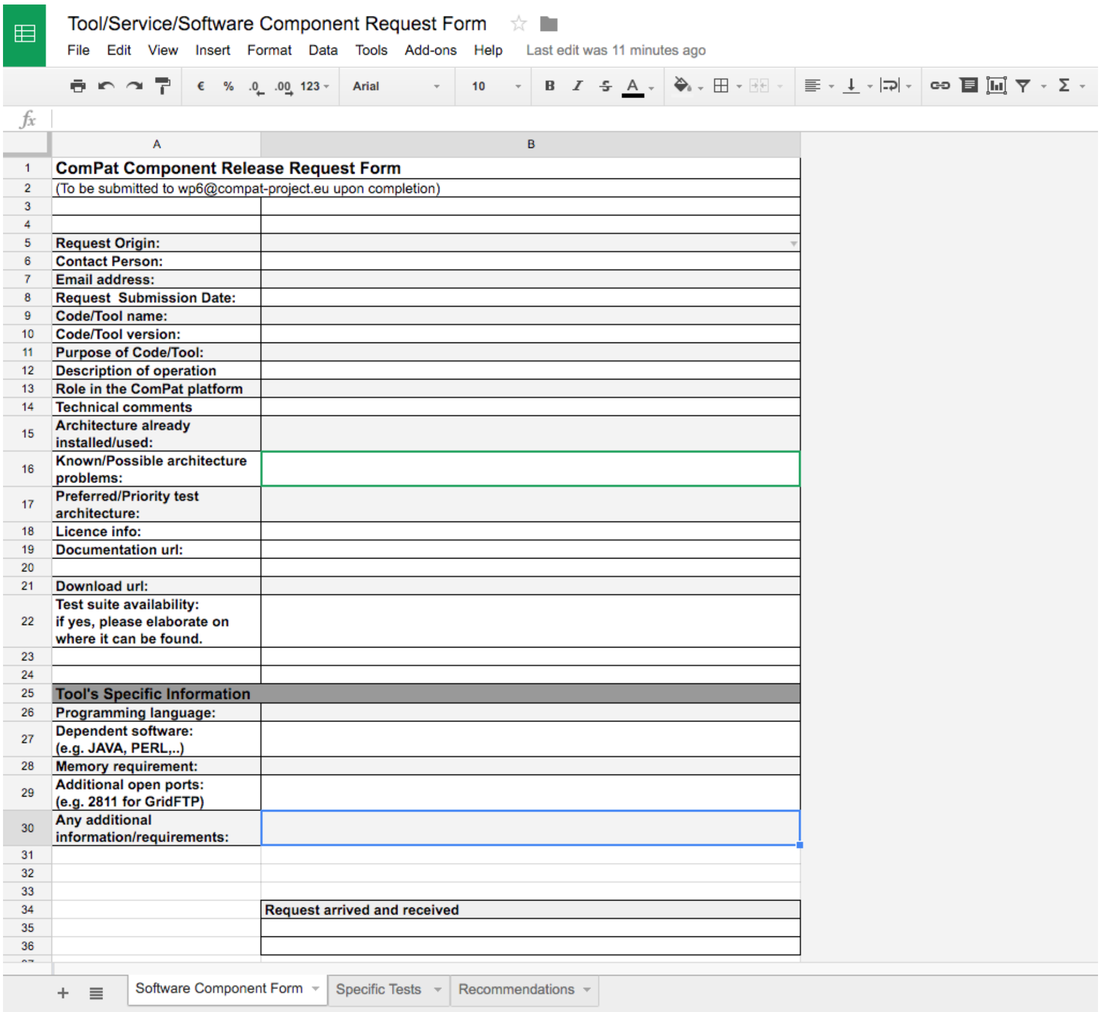
The integration process, as summarized in Figure 2 below, makes new software available within the EEE, ensuring the consistency, reliability, and sustainability of the environment. The process consists of a three-step procedure: recommendation, evaluation, and actual integration. The recommendation is received and an evaluation of the suitability of the software is performed (corresponding to the "Plan" step of the PDCA cycle, beginning of Section 3). Once the software is accepted, the actual integration into the EEE takes place (PDCA "Do" step).


Figure 2: Integration process.

Below, we discuss all three steps of the integration procedure in some more detail.

#### 3.1.1.1 Recommendation

A recommendation for software to be installed ("component request") is expected to originate from WP2, 3, 4 or 5. A component request form as shown in Figure 3 is provided on Google Drive. Using this form, critical information on the component is shared with WP6.


Tool/Service/Software Component Request Form

| | A | B |
|---|---|---|
| 1 | ComPat Component Release Request Form | |
| 2 | (To be submitted to wp6@compat-project.eu upon completion) | |
| 3 | | |
| 4 | | |
| 5 | Request Origin: | |
| 6 | Contact Person: | |
| 7 | Email address: | |
| 8 | Request Submission Date: | |
| 9 | Code/Tool name: | |
| 10 | Code/Tool version: | |
| 11 | Purpose of Code/Tool: | |
| 12 | Description of operation | |
| 13 | Role in the ComPat platform | |
| 14 | Technical comments | |
| 15 | Architecture already installed/used: | |
| 16 | Known/Possible architecture problems: | |
| 17 | Preferred/Priority test architecture: | |
| 18 | Licence info: | |
| 19 | Documentation url: | |
| 20 | | |
| 21 | Download url: | |
| 22 | Test suite availability: if yes, please elaborate on where it can be found. | |
| 23 | | |
| 24 | | |
| 25 | Tool's Specific Information | |
| 26 | Programming language: | |
| 27 | Dependent software: (e.g. JAVA, PERL,..) | |
| 28 | Memory requirement: | |
| 29 | Additional open ports: (e.g. 2811 for GridFTP) | |
| 30 | Any additional information/requirements: | |
| 31 | | |
| 32 | | |
| 33 | | |
| 34 | | Request arrived and received |
| 35 | | |
| 36 | | |

Software Component Form | Specific Tests | Recommendations

Figure 3: Component request form.

#### 3.1.1.2 Evaluation

A crucial part in the procedure shown in Figure 2 is the evaluation – i.e. the rating of the suitability of recommended software based on the following three criteria:

1. **Functions and usability** of the recommended software have to fulfil the users' requirements.
2. **Ease of integration** of the software with the EEE computing resources: tools to be integrated shall, for example, not violate any usage policies of resource providers.
3. **Support** from the developers of the software shall absolutely be available at least for the run time of the ComPat project. Thus, we ensure maintainability, sustainability and reliability.

Furthermore, specific tests of each component to be integrated[9] will be defined and conducted on the resources of the EEE. The tests can be recommended by the developers and/or provided by experts from other ComPat WPs. The persons responsible for an EEE computing resource can require additional security tests or audits before a component is deployed on their resource; WP6 encourages such requests

[9] Software already audited and certified by external organizations or projects will be excluded from tests, with the exception of new functionality implemented in scope of the ComPat project.

to be made at the earliest possible point of time. Within the “component request” spreadsheet/document (Figure 3), a form “Specific Tests” is provided for documentation, as shown in Figure 4 below:

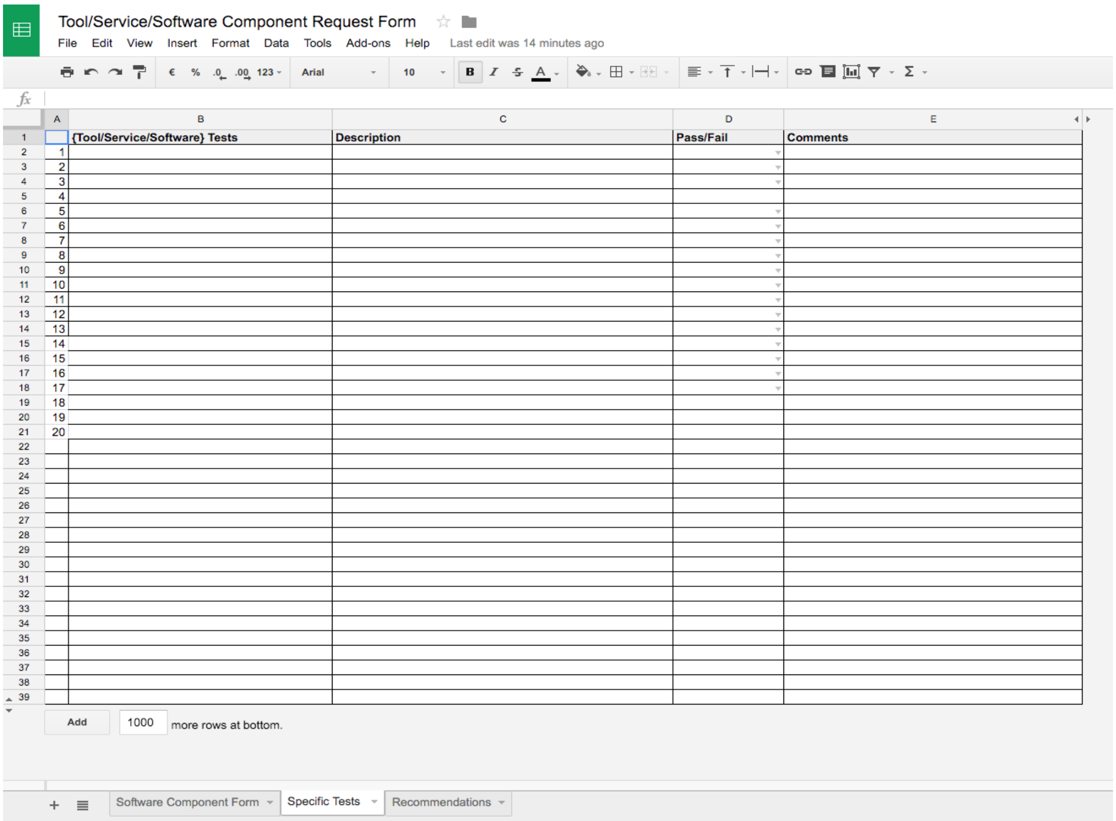


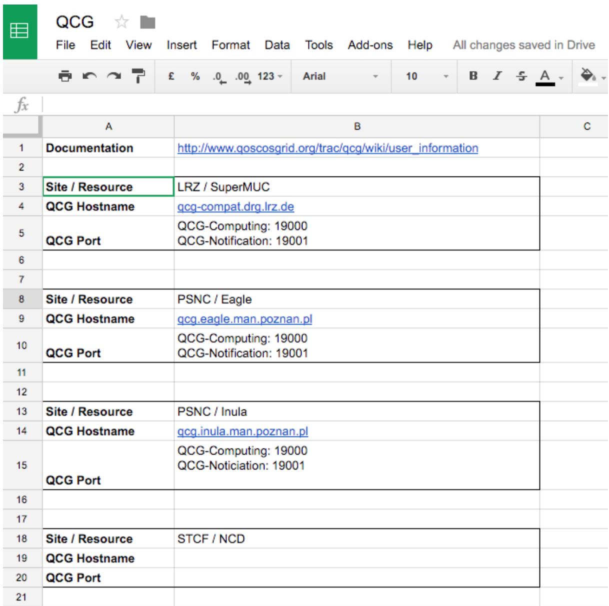

Figure 4: Specific tests form.

If problems are found with the evaluated component, recommendations for improvement are provided to the respective WPs that requested the evaluation. To this purpose, the Component Request spreadsheet/document contains also a “Recommendations” form. The requesting WPs are additionally informed via emails.

The decision whether a component shall be accepted for integration will be made by WP6 depending on the suitability rating (discussed at the beginning of this Section) and the results of the specific tests (with possible improvements based on WP6’s recommendations).

#### 3.1.1.3 Actual integration

Upon the successful evaluation of the component (or the fulfilment of WP6’s recommendations to improve the component), the software will be integrated into the EEE. This includes the monitoring of the component and the provision of a set of documents on how to effectively use this component. The documents must be provided by WP4 and WP5; a very short example (for QCG) is shown in Figure 5. Components without proper documentation will not be integrated in the EEE.



| | A | B | C |
|---|---|---|---|
| 1 | Documentation | http://www.qoscosgrid.org/trac/qcg/wiki/user_information | |
| 2 | | | |
| 3 | Site / Resource | LRZ / SuperMUC | |
| 4 | QCG Hostname | qcg-compat.drg.lrz.de | |
| 5 | QCG Port | QCG-Computing: 19000<br>QCG-Notification: 19001 | |
| 6 | | | |
| 7 | | | |
| 8 | Site / Resource | PSNC / Eagle | |
| 9 | QCG Hostname | qcg.eagle.man.poznan.pl | |
| 10 | QCG Port | QCG-Computing: 19000<br>QCG-Notification: 19001 | |
| 11 | | | |
| 12 | | | |
| 13 | Site / Resource | PSNC / Inula | |
| 14 | QCG Hostname | qcg.inula.man.poznan.pl | |
| 15 | QCG Port | QCG-Computing: 19000<br>QCG-Noticiation: 19001 | |
| 16 | | | |
| 17 | | | |
| 18 | Site / Resource | STCF / NCD | |
| 19 | QCG Hostname | | |
| 20 | QCG Port | | |
| 21 | | | |

Figure 5: Deployed software information.

### 3.1.2 Operation Process

The operation process ensures that the EEE operates reliably for its users. It is summarised in Figure 6:

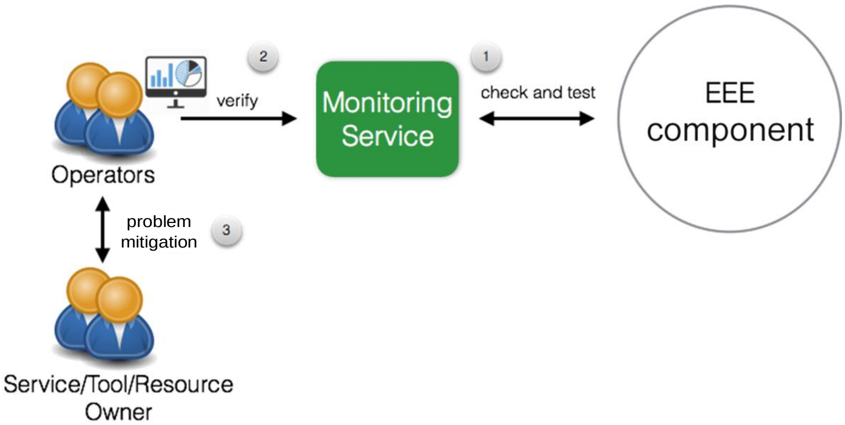



Figure 6: Operation process.

The process consists of a three-step procedure:

(1) periodic checks and tests of the software components (monitoring, see also Section 4.1),

(2) verification and assessment of results (quality control and problem detection), and

(3) commencement of the problem mitigation process in case a component fails to fulfil the quality requirements, with possible removal of the component.

Here, steps (1) and (2) correspond to the "Check" step of the PDCA cycle (beginning of Sec. 3), and the initiation of the problem mitigation process (3) and its consequences correspond to the "Act" step. The problem mitigation process is defined as follows:

(i) Component requesters ("owners") are notified via email.

(ii) The component owners are required to diagnose the problem. They initially have to give an estimate of the time to resolution within three working days.

(iii) If required, a service down time will be announced to all ComPat users.

(iv) Once the problem is solved, the component owners notify WP6, and the issue is closed. If, however, the estimated time to resolution has passed and the issue is still open, steps (i)-(iv) will be repeated.

If the problem mitigation process has been repeated two times without success for a certain component, the problem will be brought to the attention of the Technical Manager of ComPat, who may decide to reject the component from the EEE or to repeat the problem mitigation process another fixed number of times and decide again on how to proceed.

The PDCA cycle ends with continuous monitoring and bug-fixing or possible removal of the component, and repeats separately for each new software component (or for a defined set of components treated as an entity).

Only components which show an "O.K." status in the monitoring are considered fully accepted in the EEE and basic information (including pointers to further documentation) on them is included in the EEE user documentation.

## 3.2 Computing Resource Integration / Operation

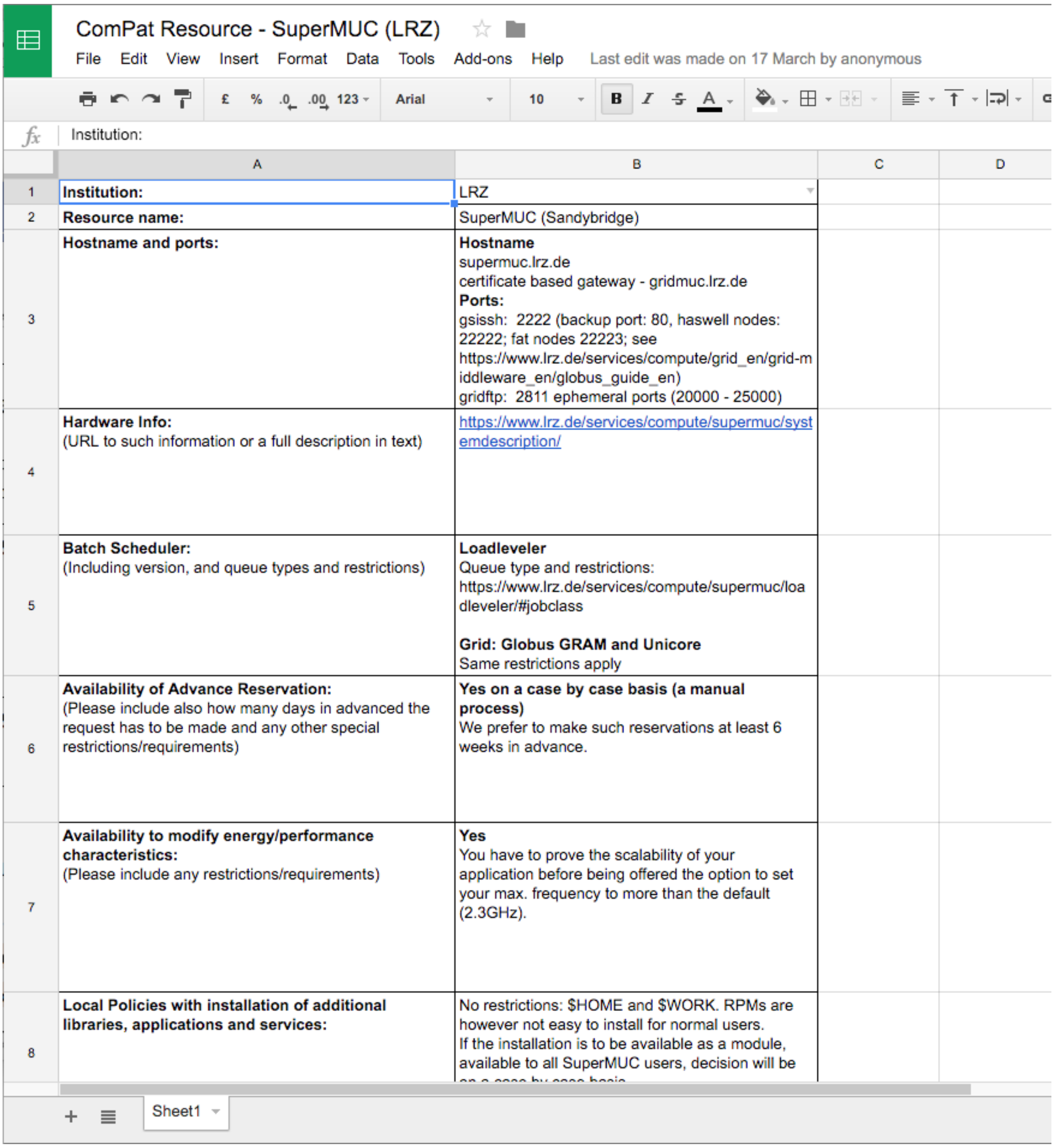
When new EEE computing resources shall be integrated, this merely means that the software accepted within the EEE to date must be installed there in a fast-forward process. Afterwards, for operation, the computing resource will be included in the EEE monitoring system (i.e. ping checks and monitoring of the software components will be performed – see also Section 4.1), and possible problems will be resolved. Besides these procedures ("normal operation"), the operation process for Computing Resources includes another procedure ("suspended operation") for treating downtimes announced by the system management of EEE computing resources.

### 3.2.1 Integration Process

In analogy to the "recommendation" step in the software integration process (Section 3.1.1), integration begins with the resource provider giving WP6 details about his resource, for which he uses a dedicated form shown below (Figure 7). WP6 then evaluates which parts of the EEE software stack can be installed on the Resource in order to complete the PDCA "Plan" activity. A list of software components to be installed is made. If fundamental EEE components cannot be installed because of system characteristics, the integration of the computing resource into the EEE may be rejected.

The PDCA "Do" activity then consists in installing all software components and in deploying authentication information such that ComPat users can log onto the system.

As the number of EEE computing resources to be integrated is expected to be moderate, documentation of the integration process[10] will have the form of a short summary in the respective EEE progress report (D6.2, D6.3, D6.4) following the integration attempt.




| | A | B | C | D |
|---|---|---|---|---|
| 1 | **Institution:** | LRZ | | |
| 2 | **Resource name:** | SuperMUC (Sandybridge) | | |
| 3 | **Hostname and ports:** | **Hostname**<br>supermuc.lrz.de<br>certificate based gateway - gridmuc.lrz.de<br>**Ports:**<br>gsissh: 2222 (backup port: 80, haswell nodes: 22222; fat nodes 22223; see https://www.lrz.de/services/compute/grid_en/grid-middleware_en/globus_guide_en)<br>gridftp: 2811 ephemeral ports (20000 - 25000) | | |
| 4 | **Hardware Info:**<br>(URL to such information or a full description in text) | https://www.lrz.de/services/compute/supermuc/systemdescription/ | | |
| 5 | **Batch Scheduler:**<br>(Including version, and queue types and restrictions) | **Loadleveler**<br>Queue type and restrictions:<br>https://www.lrz.de/services/compute/supermuc/loadleveler/#jobclass<br><br>**Grid: Globus GRAM and Unicore**<br>Same restrictions apply | | |
| 6 | **Availability of Advance Reservation:**<br>(Please include also how many days in advanced the request has to be made and any other special restrictions/requirements) | **Yes on a case by case basis (a manual process)**<br>We prefer to make such reservations at least 6 weeks in advance. | | |
| 7 | **Availability to modify energy/performance characteristics:**<br>(Please include any restrictions/requirements) | **Yes**<br>You have to prove the scalability of your application before being offered the option to set your max. frequency to more than the default (2.3GHz). | | |
| 8 | **Local Policies with installation of additional libraries, applications and services:** | No restrictions: $HOME and $WORK. RPMs are however not easy to install for normal users.<br>If the installation is to be available as a module, available to all SuperMUC users, decision will be | | |

Figure 7: EEE computing resource information.

[10] This only applies to EEE Computing Resources other the initial ones already listed in the present deliverable.

### 3.2.2 Operation Process

#### 3.2.2.1 Normal Operation: Monitoring / Problem Mitigation

After integration, the new EEE computing resource is registered in the Nagios monitoring system of the EEE (see Section 4.1), and is thus subject to ping checks and software monitoring (PDCA “Check”). As soon as all fundamental EEE components run on the resource, it is considered accepted into the EEE and listed in the EEE documentation. If checks fail, the problem mitigation process described in Section 3.2.1 is analogously applied to the resource as an EEE component (instead of a software component), and its provider has to resolve the problems (PDCA “Act”). In case of continuous failure, the resource can be expelled from the EEE following the rules defined in Section 3.2.1.

#### 3.2.2.2 Suspended Operation: Announced Downtimes

In contrast to the operation process for software management, the computing resource operation process needs to take into account downtimes announced by the respective system managers. These occur due to system maintenances (including emergency maintenances). If (and only if) a system downtime is announced by the system managers at least one week in advance, we will call it a “scheduled downtime” below.

WP6 will continuously retrieve and collect downtime announcements by EEE contributors. Users need then to be notified of the downtimes and QCG job submission to the respective resources has to be disabled. In addition, during scheduled downtimes, Nagios monitoring needs to be suspended (cf. Section 4). The exact procedure for all this is as follows:

(i) Downtimes are collected in an EEE downtime database accessible via the internet. This database is viewable via an http interface (in the course of implementation).

(ii) Notifications of scheduled downtimes are sent to the mailing list maintenance@compat-project.eu (containing all relevant users and EEE system managers) one week before the scheduled downtime and, in addition, one day before. Non-scheduled downtimes are announced to maintenance@compat-project.eu at first notice.

(iii) QCG, which continuously evaluates the EEE downtime database, stops deploying jobs to the respective resource. In case of a scheduled downtime, Nagios, querying the downtime database as well, suspends monitoring of the resource.

(iv) After the ordinary end of the scheduled maintenance, Nagios automatically recommences to monitor the resource.

(v) After the real end of the maintenance – i.e. only after a possible unforeseen prolongation (as announced by the administrators of the respective resource), an announcement is sent out to maintenance@compat-project.eu and QCG job deployment re-commences.

For each announced downtime, a short log of the actions (i)-(v) is given in the following WP6 deliverable (D6.2, D6.3, D6.4). Severe problems with the procedure and mitigation actions are reported.

# 4 Assessment of Operational Procedures

Based on experience with similar projects, the operational procedures outlined above have been designed to optimally suit the needs of the ComPat project. In order to enable us to continuously assess the availability of the EEE, and to improve not only EEE components, but also operational procedures for ComPat where necessary, we have identified a set of Key Performance Indicators (KPIs).

On the basis of monitoring data (Section 4.1), we will calculate the KPIs as discussed in Section 4.2. Entities monitored and subject to KPI evaluation are called "services" below. These include EEE software components for common usage, and EEE computing resources (for which a ping monitoring and tests of the scheduling system are implemented). We will continuously assess and report the quality of our services (Sections 4.3, 4.4) on the basis of the KPIs, and make improvements where needed.

## 4.1 Nagios Monitoring

To assess the status of EEE components, a central Nagios monitoring system has been set up (https://nagios-compat.drg.lrz.de). Figure 8 shows the monitored status of EEE on June 24th, 2016, with QCG Computing at LRZ undergoing some further configuration, and STFC resources still to be added to the monitoring system.

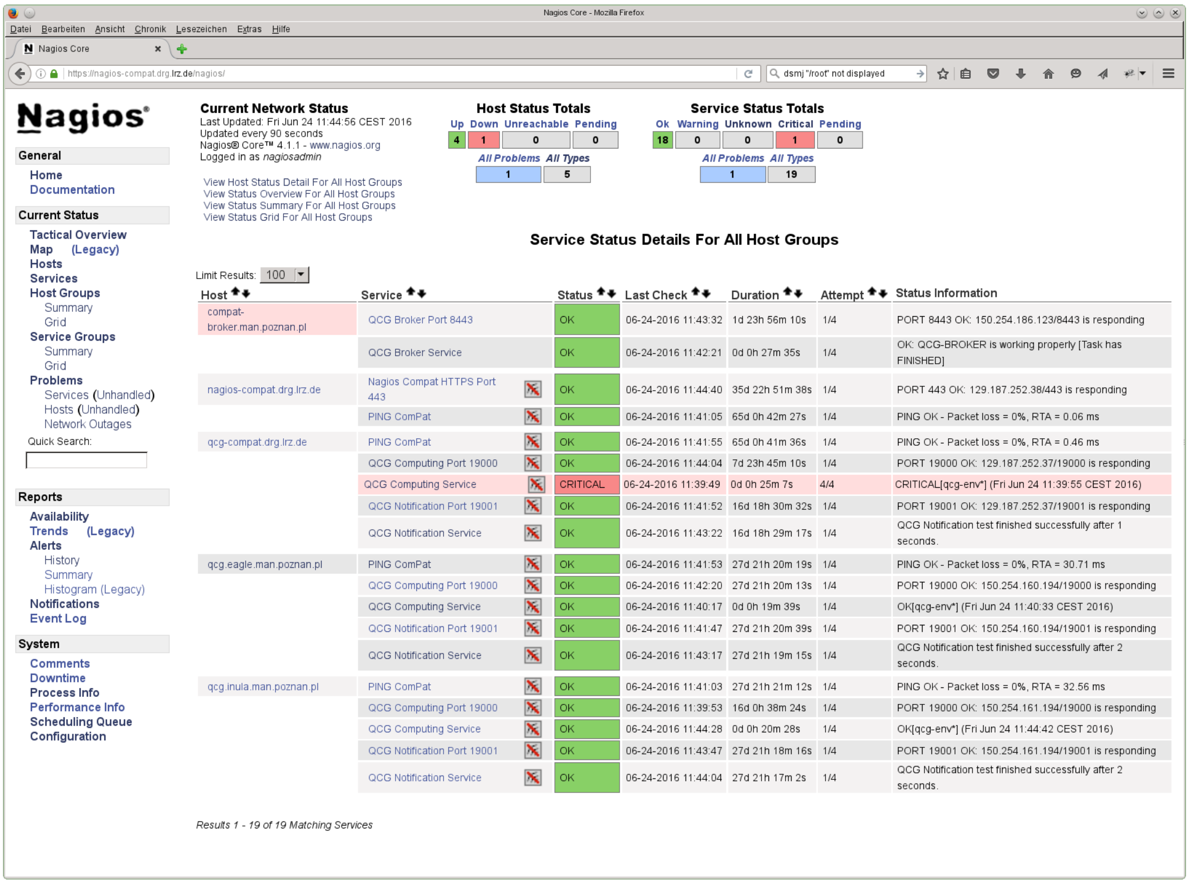



Figure 8: Status of EEE on June 24th, 2016.

## 4.2 Key Performance Indicators (KPIs)

From the Nagios database, we can extract the status *S(t)* of any EEE service (1: available, 0: not available) at any given point of time *t*. From this data, we calculate "availability of services" KPIs, which are complemented by a "security" KPI.

### 4.2.1 Availability of Services

Availability of EEE services can be measured using the metrics defined in this section – our KPIs, which will provide an insight into how reliable the EEE services are.

<table>
<tr><td>Percentage of service availability (PSA)</td><td>This is a status-based metric that is reported in the percentage of time the service is running during an evaluation period. For each single EEE computing resource and service, this metric will be calculated separately as:<br>$$PSA = 100 \times \frac{\int_{t_{begin}}^{t_{end}} [1 - S(t)]\ \mathrm{d}\,t}{\int_{t_{begin}}^{t_{end}} \mathrm{d}\,t} \approx 100 \times \frac{\sum_i [1 - S(t_i)]\Delta t_i}{\sum_i \Delta t_i}$$<br>Here $S(t)$ is the status at the time $t$ (1: available, 0: not available). The second formula is used in practice, summing over discrete logging intervals $\Delta t_i$ (set as needed for accurate results, without producing an excessive load by monitoring – i.e. 10 minutes or longer) from the beginning ($t_{begin}$) to the end ($t_{end}$), is used in practice. Planned maintenance intervals – which will be few in any case – will be excluded from the summing process and reported separately.<br><br>Besides the PSA for each computing resource and service, we will report PSA values averaged over all EEE computing resources.</td></tr>
<tr><td>Duration of service interruptions (DSI)</td><td>This metric is the average duration of the interruptions of a service within a given time frame. Again, for each service we calculate this value by EEE computing resource and as an average over all resources. In mathematical terms, the DSI is computed as:<br>$$DSI = \frac{1}{NSI} \times \int_{t_{begin}}^{t_{end}} [1 - S(t)]\ \mathrm{d}\,t \approx \frac{1}{NSI} \times \sum_i S(t_i)\Delta t_i$$<br>The symbols used here have been explained above (PSA formula), except for NSI, which is the number of service interruptions (i.e. status changes from "available" / S($t$) = 1 to unavailable / S($t$) = 0) within the time frame considered. Again, the time intervals $\Delta t_i$ are reasonably chosen as necessary for an accurate measurement (10 minutes or longer), and scheduled maintenances are excluded from the calculation.</td></tr>
</table>

| | |
|---|---|
| **Monitoring coverage** | This is the percentage of EEE services and computing resources that are monitored for availability. The percentage is relative to the total number of services and computing resources accepted as parts of the EEE. |

Table 4. “Availability of services” KPIs.

### 4.2.2 Security

In order to evaluate the security of services provided by EEE platform, security incidents related to software components developed within the scope of the EEE will be assessed.

| | |
|---|---|
| **Number of major (EEE-related) security incidents** | This is the total number of major security incidents caused by bugs or misconfiguration of components developed within the ComPat EEE. Major security incidents are incidents that require an immediate suspension of operation of a deployed component across all resources, and subsequent patching, to prevent a compromise in security. |

Table 5. “Security” KPIs.

## 4.3 Quality Assessment

EEE services and tools are split into two categories, external and internal. Services and tools that are used internally by WP6 for operational purposes, e.g. Nagios, do not need to be subject to quality measurements. Only services and tools that are directly exposed to ComPat users, i.e. external, will be assessed.

With the ComPat EEE, we aim at practically matching professional production standards, even if the environment and the applications run on it are experimental.

### 4.3.1 Availability of Services

The proposed associated quality classification for “availability of services” KPIs is shown in Table 6. All external services offered by ComPat should have at least a “good” level of quality for it to be considered acceptable. Resource providers with services having a “mediocre” or “poor” level of quality will be informed and requested to initiate mitigation actions (cf. Section 3.1.2).

| Excellent | Very Good | Good | Mediocre | Poor |
|---|---|---|---|---|
| Above 99% | 99%-95% | 95-85% | 85-70% | Below 70% |

Table 6. Quality measurement classification for “availability of services” KPIs.

### 4.3.2 Security

Here, instead of percentages, absolute numbers will be evaluated. We aim at zero security incidents caused by software under control of ComPat (i.e. developed in its context); if this cannot be fulfilled, the respective incidents will be discussed in detail in the reports on EEE operation (D6.2, D6.3, D6.4).

## 4.4 Results of Quality Measurements

As the initial setup phase of the EEE is just being concluded, there are no relevant measurement results yet. In the next report, first results will be shown.

# 5 Conclusions and Outlook

The functional areas of the EEE, as well as an operational model consisting of two processes – integration and operation – have been defined. This includes detailed monitoring and problem mitigation procedures, and a procedure to handle resource downtimes. To facilitate the setup and usage of the EEE, information on software, on its resources and on authentication/authorisation is collected and shared on a Google Drive. Forms corresponding to the processes in the operational model have been set up on this platform in order to homogenise and standardise the information flow.

The EEE is monitored using Nagios, which measures the availability of the deployed services. A set of key performance indicators (KPIs) has been defined to help in the quantitative assessment of the measurements and of security incidents / patching events. Performance criteria – i.e. aims – have been defined for each KPI; first assessments are deferred to later WP6 deliverables (D6.2, D6.3, D6.4) as the initial setup phase of the EEE has just finished. In these deliverables, we will also report on possible changes in the operational model, which will continuously be optimised as a response to quality assessments and user feedback.

Within the next few months, the QCG middleware as a central EEE component is expected to be available on all EEE computing resources. We will then start a dialogue with WP2-5 in order to identify further usage requirements, and take action to fulfil these requirements (e.g. installation of additional tools). The EEE service quality will simultaneously be assessed based on the criteria mentioned above. With time, we will project and implement new core services of the EEE (e.g. in the area of energy monitoring). In addition, a systematic scheme for the distribution of the application runs among EEE computing resources will be defined.