**EGI-InSPIRE**

UMD Compute Capabilities

Quality Criteria

v1.1

|  |  |
| --- | --- |
| Document identifier: | EGI-COMPUTE-QC-V1.1.docx |
| Date: | 03/02/2011 |
| Document Link: | https://documents.egi.eu/document/240 |

|  |
| --- |
| Abstract  This document describes the Quality Criteria for the Compute Capabilities identified in the UMD Roadmap. |

Copyright notice

Copyright © Members of the EGI-InSPIRE Collaboration, 2010. See www.egi.eu for details of the EGI-InSPIRE project and the collaboration. EGI-InSPIRE (“European Grid Initiative: Integrated Sustainable Pan-European Infrastructure for Researchers in Europe”) is a project co-funded by the European Commission as an Integrated Infrastructure Initiative within the 7th Framework Programme. EGI-InSPIRE began in May 2010 and will run for 4 years. This work is licensed under the Creative Commons Attribution-Noncommercial 3.0 License. To view a copy of this license, visit http://creativecommons.org/licenses/by-nc/3.0/ or send a letter to Creative Commons, 171 Second Street, Suite 300, San Francisco, California, 94105, and USA. The work must be attributed by attaching the following reference to the copied elements: “Copyright © Members of the EGI-InSPIRE Collaboration, 2010. See www.egi.eu for details of the EGI-InSPIRE project and the collaboration”. Using this document in a way and/or for purposes not foreseen in the license, requires the prior written permission of the copyright holders. The information contained in this document represents the views of the copyright holders as of the date such views are published.

Document Log

|  |  |  |  |
| --- | --- | --- | --- |
| **Issue** | **Date** | **Comment** | **Author/Partner** |
| 1.0 | 19/01/2011 | Reorganisation of criteria according to UMD Roadmap v2 | Enol Fernández |
| 1.1 | 02/02/2011 | Review and added Job Scheduling | Enol Fernández, Álvaro Fernández |

**Table of contents**

1 Job Execution 5

1.1 CREAM Interface 5

JOBEXEC\_CREAM\_API\_1 5

1.1.1 Job Submission tests 6

JOBEXEC\_CREAM\_JOB\_1 6

JOBEXEC\_CREAM\_JOB\_2 7

JOBEXEC\_CREAM\_JOB\_3 8

1.2 ARC Interface 9

JOBEXEC\_ARC\_API\_1 9

1.2.1 Job Submission tests 10

JOBEXEC\_ARC\_JOB\_1 10

JOBEXEC\_ARC\_JOB\_2 11

JOBEXEC\_ARC\_JOB\_3 12

1.3 BES Interface 13

JOBEXEC\_BES\_API\_1 13

1.3.1 Job Submission tests 14

JOBEXEC\_BES\_JOB\_1 14

JOBEXEC\_BES\_JOB\_2 15

JOBEXEC\_BES\_JOB\_3 16

1.4 Execution Manager Support 17

JOBEXEC\_EXECMNGR\_1 17

JOBEXEC\_EXECMNGR\_2 18

JOBEXEC\_EXECMNGR\_5 19

JOBEXEC\_EXECMNGR\_6 20

1.5 Availability/Scalability 21

JOBEXEC\_AVAIL\_1 21

JOBEXEC\_AVAIL\_2 22

JOBEXEC\_AVAIL\_3 23

2 Parallel Job 24

2.1 Submission of parallel jobs 24

PARALLEL\_JOB\_1 24

PARALLEL\_JOB\_2 25

PARALLEL\_JOB\_3 26

PARALLEL\_JOB\_4 27

PARALLEL\_JOB\_5 28

2.2 MPI support 29

PARALLEL\_MPI\_1 29

PARALLEL\_MPI\_2 30

2.3 OpenMP support 31

PARALLEL\_OMP\_1 31

PARALLEL\_OMP\_1 32

3 Job Scheduling 33

3.1 gLite WMS 33

JOBSCH\_WMS\_API\_1 33

JOBSCH\_WMS\_API\_2 34

JOBSCH\_WMS\_API\_3 35

3.1.1 End-to-end job tests 36

JOBSCH\_WMS\_JOB\_1 36

JOBSCH\_WMS\_JOB\_2 37

JOBSCH\_WMS\_JOB\_3 38

JOBSCH\_WMS\_JOB\_4 39

JOBSCH\_WMS\_JOB\_5 40

JOBSCH\_WMS\_JOB\_6 41

JOBSCH\_WMS\_JOB\_7 42

JOBSCH\_WMS\_JOB\_8 43

JOBSCH\_WMS\_JOB\_9 44

JOBSCH\_WMS\_JOB\_10 45

3.2 Job Execution Capability Support 46

JOBSCH\_EXEC\_1 46

3.3 Service availability, monitoring and error handling. 47

JOBSCH\_SERVICE\_1 47

JOBSCH\_SERVICE\_2 48

JOBSCH\_SERVICE\_3 49

JOBSCH\_SERVICE\_4 50

JOBSCH\_SERVICE\_5 51

4 References 52

# Job Execution

The Compute Capability relates to the ability to describe, submit, manage and monitor a work item on a specific site submitted for either queued batch or interactive execution. Currently, there are 3 different interfaces considered for the Compute Capability. The implementations must support, at least, one of them.

## CREAM Interface

These QC refer to implementation of the Compute capability using the CREAM interface [R 5]. May be deprecated in future versions of the QC.

|  |  |  |
| --- | --- | --- |
| Cream API TestSuite | | |
| ID | | JOBEXEC\_CREAM\_API\_1 |
| Mandatory | | |
| Applicability | | Implementations of CREAM interfaces |
| Related Requirements | | None |
|  | | |
| Description  Test the all the functions of the CREAM interface. | | |
| Input from TP  Complete Test suite for the CREAM API. It must include tests for all the documented functions in the CREAM WSDL.  For all functions, check both correct and invalid input. Invalid output should throw an exception as documented. Test also with valid and invalid credentials. Invalid credentials should throw security related exceptions. | | |
| Test Suite Description | | |
| **Pre-condition** | Valid user credentials. | |
| **Test** | Test all CREAM functionality, with correct/incorrect input and with valid and invalid credentials. | |
| **Expected Outcome** | Log of all the operations performed. All the documented functions work as documented. | |
| Pass/Fail Criteria  Pass if the testsuite is provided and passes. | | |
| Related Information | | |
| History | | |

### Job Submission tests

The following job submission tests use the gLite JDL format for the specification of jobs.

|  |  |  |
| --- | --- | --- |
| Simple Job Submission | | |
| ID | | JOBEXEC\_CREAM\_JOB\_1 |
| Mandatory | | |
| Applicability | | Implementations of CREAM interfaces |
| Related Requirements | | None |
|  | | |
| Description  Test the job submission for simple job. | | |
| Input from TP  Test for the submission of a job. | | |
| Test 1 | | |
| **Pre-condition** | Valid User proxy and valid delegation in the service. | |
| **Test** | Job submission of simple job:  Executable = /bin/sleep;  Arguments = "120"; | |
| **Expected Outcome** | Job finishes correctly. Unique Identifier for the submitted jobs, status log of the job. | |
| Pass/Fail Criteria  Pass if the test is provided and passes. | | |
| Related Information | | |
| History | | |

|  |  |  |
| --- | --- | --- |
| Simple Job Submission with files | | |
| ID | | JOBEXEC\_CREAM\_JOB\_2 |
| Mandatory | | |
| Applicability | | Implementations of CREAM interfaces |
| Related Requirements | | None |
|  | | |
| Description  Test the job submission for simple job with input and output files. | | |
| Input from TP  Test for the submission of a job with input and output files. | | |
| Test 1 | | |
| **Pre-condition** | Valid User proxy and valid delegation in the service.  Non-empty file “myfile” | |
| **Test** | Job submission for job with input and output files:  Executable = "/bin/ls";  Arguments = "-l";  StdOutput = "std.out";  StdError = "std.err";  InputSandbox = {"myfile"};  OutputSandbox = {"std.out", "std.err"};Job Submission | |
| **Expected Outcome** | Job finishes correctly, output contains the listing of the directory including the input file with correct size. Unique Identifier for the submitted jobs, status log of the job. | |
| Pass/Fail Criteria  Pass if the test is provided and passes. | | |
| Related Information | | |
| History | | |

|  |  |  |
| --- | --- | --- |
| Job Cancel | | |
| ID | | JOBEXEC\_CREAM\_JOB\_3 |
| Mandatory | | |
| Applicability | | Implementations of CREAM interfaces |
| Related Requirements | | None |
|  | | |
| Description  Test the job cancellation for a job. | | |
| Input from TP  Test for the cancellation of a job. | | |
| Test 1 | | |
| **Pre-condition** | Valid User proxy and valid delegation in the service. | |
| **Test** | Job Submission and then cancellation.  Possible description for job:  Executable = "/bin/sleep";  Arguments = "20m"; | |
| **Expected Outcome** | Job is submitted and then cancelled correctly. Unique Identifier for the submitted jobs, status log of the job. | |
| Pass/Fail Criteria  Pass if the test is provided and passes. | | |
| Related Information | | |
| History | | |

## ARC Interface

These QC refer to implementation of the Compute capability using the ARC-CE interface [R 6]. May be deprecated in future versions of the QC.

|  |  |  |
| --- | --- | --- |
| ARC-CE API TestSuite | | |
| ID | | JOBEXEC\_ARC\_API\_1 |
| Mandatory | | |
| Applicability | | Implementations of ARC-CE interface. |
| Related Requirements | | None |
|  | | |
| Description  Test the all the functions of the ARC-CE interface. | | |
| Input from TP  Complete Test suite for the ARC-CE API. It must include tests for all the documented functions of the API.  For all functions, check both correct and invalid input. Invalid output should throw an exception as documented. Test also with valid and invalid credentials. Invalid credentials should throw security related exceptions. | | |
| Test Suite Description | | |
| **Pre-condition** | Valid user credentials. | |
| **Test** | Test all ARC-CE functionality, with correct/incorrect input and with valid and invalid credentials. | |
| **Expected Outcome** | Log of all the operations performed. All the documented functions work as documented. | |
| Pass/Fail Criteria  Pass if the testsuite is provided and passes. | | |
| Related Information | | |
| History | | |

### Job Submission tests

The following job submission tests use the ARC xRSL format for the specification of jobs.

|  |  |  |
| --- | --- | --- |
| Simple Job Submission | | |
| ID | | JOBEXEC\_ARC\_JOB\_1 |
| Mandatory | | |
| Applicability | | Implementations of ARC-CE interface. |
| Related Requirements | | None |
|  | | |
| Description  Test the job submission for simple job. | | |
| Input from TP  Test for the submission of a job. | | |
| Test 1 | | |
| **Pre-condition** | Valid User proxy. | |
| **Test** | Job submission for simple job:  &(executable="/bin/sleep")(arguments="120") | |
| **Expected Outcome** | Job finishes correctly. Unique Identifier for the submitted jobs, status log of the job. | |
| Pass/Fail Criteria  Pass if the test is provided and passes. | | |
| Related Information | | |
| History | | |

|  |  |  |
| --- | --- | --- |
| Simple Job Submission with files | | |
| ID | | JOBEXEC\_ARC\_JOB\_2 |
| Mandatory | | |
| Applicability | | Implementations of ARC-CE interface. |
| Related Requirements | | None |
|  | | |
| Description  Test the job submission for simple job with input and output files. | | |
| Input from TP  Test for the submission of a job with input and output files. | | |
| Test 1 | | |
| **Pre-condition** | Valid User proxy and valid delegation in the service.  Non-empty file “myfile” | |
| **Test** | Job Submission of job with input and output files:  &  (executable="/bin/ls")  (arguments="-l")  (stdout="std.out ")  (stderr="std.err ")  (inputFiles=("myfile" "")) | |
| **Expected Outcome** | Job finishes correctly, output contains the listing of the directory including the input file with correct size. Unique Identifier for the submitted jobs, status log of the job. | |
| Pass/Fail Criteria  Pass if the test is provided and passes. | | |
| Related Information | | |
| History | | |

|  |  |  |
| --- | --- | --- |
| Job Cancel | | |
| ID | | JOBEXEC\_ARC\_JOB\_3 |
| Mandatory | | |
| Applicability | | Implementations of ARC-CE interface. |
| Related Requirements | | None |
|  | | |
| Description  Test the job cancellation for a job. | | |
| Input from TP  Test for the cancellation of a job. | | |
| Test 1 | | |
| **Pre-condition** | Valid User proxy and valid delegation in the service. | |
| **Test** | Job Submission and then cancellation.  Possible job description for job:  &(executable="/bin/sleep")(arguments="20m") | |
| **Expected Outcome** | Job is submitted and then cancelled correctly. Unique Identifier for the submitted jobs, status log of the job. | |
| Pass/Fail Criteria  Pass if the test is provided and passes. | | |
| Related Information | | |
| History | | |

## BES Interface

These QC refer to implementation of the Compute capability using the OGSA BES interface [R 7]. May be deprecated in future versions of the QC.

|  |  |  |
| --- | --- | --- |
| BES API TestSuite | | |
| ID | | JOBEXEC\_BES\_API\_1 |
| Mandatory | | |
| Applicability | | Implementations of BES interface. |
| Related Requirements | | None |
|  | | |
| Description  Test the all the functions of the BES interface. | | |
| Input from TP  Complete Test suite for the BES API. It must include tests for all the mandatory documented functions of the specification.  For all functions, check both correct and invalid input. Invalid output should throw an exception as documented. Test also with valid and invalid credentials. Invalid credentials should throw security related exceptions. | | |
| Test Suite Description | | |
| **Pre-condition** | Valid user credentials. | |
| **Test** | Test all OGSA BES functionality, with correct/incorrect input and with valid and invalid credentials. | |
| **Expected Outcome** | Log of all the operations performed. All the documented functions work as documented. | |
| Pass/Fail Criteria  Pass if the testsuite is provided and passes. | | |
| Related Information | | |
| History | | |

### Job Submission tests

The following job submission tests use the UNICORE UCC JSON format for the specification of jobs.

|  |  |  |
| --- | --- | --- |
| Simple Job Submission | | |
| ID | | JOBEXEC\_BES\_JOB\_1 |
| Mandatory | | |
| Applicability | | Implementations of BES interface. |
| Related Requirements | | None |
|  | | |
| Description  Test the job submission for simple job. | | |
| Input from TP  Test for the submission of a job. | | |
| Test 1 | | |
| **Pre-condition** | Valid User credentials. | |
| **Test** | Job Submission for simple job:  {  Executable: "/bin/sleep",  Arguments: ["20"],  } | |
| **Expected Outcome** | Job finishes correctly. Unique Identifier for the submitted jobs, status log of the job. | |
| Pass/Fail Criteria  Pass if the test is provided and passes. | | |
| Related Information | | |
| History | | |

|  |  |  |
| --- | --- | --- |
| Simple Job Submission with files | | |
| ID | | JOBEXEC\_BES\_JOB\_2 |
| Mandatory | | |
| Applicability | | Implementations of BES interface. |
| Related Requirements | | None |
|  | | |
| Description  Test the job submission for simple job with input and output files. | | |
| Input from TP  Test for the submission of a job with input and output files. | | |
| Test 1 | | |
| **Pre-condition** | Valid User proxy and valid delegation in the service.  Non-empty file “myfile” | |
| **Test** | Job Submission of job with input/output files:  {  Executable: "/bin/ls",  Arguments: ["-l"],  Stdout: std.out,  Stderr: std.err,  Imports: [  { From: "myfile", To: "myfile" },  ]  Exports: [  { From: "std.out", To: "std.out" },  { From: "std.err", To: "std.err" },  ]  } | |
| **Expected Outcome** | Job finishes correctly, output contains the listing of the directory including the input file with correct size. Unique Identifier for the submitted jobs, status log of the job. | |
| Pass/Fail Criteria  Pass if the test is provided and passes. | | |
| Related Information | | |
| History | | |

|  |  |  |
| --- | --- | --- |
| Job Cancel | | |
| ID | | JOBEXEC\_BES\_JOB\_3 |
| Mandatory | | |
| Applicability | | Implementations of BES interface. |
| Related Requirements | | None |
|  | | |
| Description  Test the job cancellation for a job. | | |
| Input from TP  Test for the cancellation of a job. | | |
| Test 1 | | |
| **Pre-condition** | Valid User proxy and valid delegation in the service. | |
| **Test** | Job Submission and then cancellation.  Possible Job description for job:  {  Executable: "/bin/sleep",  Arguments: ["20m"],  } | |
| **Expected Outcome** | Job is submitted and then cancelled correctly. Unique Identifier for the submitted jobs, status log of the job. | |
| Pass/Fail Criteria  Pass if the test is provided and passes. | | |
| Related Information | | |
| History | | |

## Execution Manager Support

These QC refer to the interaction of the Job Execution Capability with the underlying execution manager (usually a LRMS) for the work items submitted. Job Execution Capabilities are expected to support the most common LRMS used in the current EGI infrastructure:

* Torque/PBS
* LSF
* SGE/OGE

|  |  |
| --- | --- |
| Not Invasive Deployment | |
| ID | JOBEXEC\_EXECMNGR\_1 |
| Mandatory | |
| Applicability | Implementations of Job Execution Capability |
| Related Requirements | None |
|  | |
| Description  The Job Execution Capability should not introduce any modifications to the underlying execution manager or to the operations of the resources. | |
| Input from TP  Description of all the needed modifications on the local resources in order to deploy the Job Execution Capability implementation. | |
| Pass/Fail Criteria  Pass if the modifications are not invasive, namely:   * Installation of additional software at the WN is permitted as long as no extra services are run permanently at the WN. * Do not require the deployment of extra shared filesystems * Do not modify the local submission mechanism of jobs (e.g. do not require the modification of prologue/epilogue scripts of the batch system) * Do not require the creation of extra user accounts or add special privileges to a specific account. | |
| Related Information | |
| History | |

|  |  |  |
| --- | --- | --- |
| Job Submission | | |
| ID | | JOBEXEC\_EXECMNGR\_2 |
| Mandatory | | |
| Applicability | | Implementations of Job Execution Capability |
| Related Requirements | | None |
|  | | |
| Description  The Job Execution Capability must be able to submit, manage and monitor jobs to the underlying execution manager. | | |
| Input from TP  Test for job submission. | | |
| Test 1 | | |
| **Pre-condition** | None | |
| **Test** | Job submission to batch system | |
| **Expected Outcome** | Job is submitted to batch system, a valid Job ID is returned. | |
| Test 2 | | |
| **Pre-condition** | Already submitted job. | |
| **Test** | Query job status in the execution manager. | |
| **Expected Outcome** | Job status can be fetched, show a message with it. | |
| Test 3 | | |
| **Pre-condition** | Already submitted job. | |
| **Test** | Cancel job in the execution manager. | |
| **Expected Outcome** | Job is successfully cancelled. | |
| Pass/Fail Criteria  Pass if the test is provided and passes for each of the supported execution managers. | | |
| Related Information | | |
| History | | |

|  |  |  |
| --- | --- | --- |
| Information Retrieval | | |
| ID | | JOBEXEC\_EXECMNGR\_5 |
| Mandatory | | |
| Applicability | | Implementations of Job Execution Capability |
| Related Requirements | | None |
|  | | |
| Description  The Job Execution Capability must be able to fetch information from the execution manager and publish following the GlueSchema v1.3 | | |
| Input from TP  Test for information retrieval from execution manager. | | |
| Test 1 | | |
| **Pre-condition** | Configured system. | |
| **Test** | Information retrieval from execution manager. | |
| **Expected Outcome** | Computing Element related entities of GlueSchema v1.3 using the **actual** information from the execution manager. | |
| Pass/Fail Criteria  Pass if the test is provided and passes. | | |
| Related Information | | |
| History | | |

|  |  |  |
| --- | --- | --- |
| Information Retrieval | | |
| ID | | JOBEXEC\_EXECMNGR\_6 |
| Not Mandatory | | |
| Applicability | | Implementations of Job Execution Capability |
| Related Requirements | | None |
|  | | |
| Description  The Job Execution Capability must be able to fetch information from the execution manager and publish following the GlueSchema v2 | | |
| Input from TP  Test for information retrieval from execution manager. | | |
| Test 1 | | |
| **Pre-condition** | Configured system. | |
| **Test** | Information retrieval from execution manager. | |
| **Expected Outcome** | Computing Element related entities of GlueSchema v2 using the **actual** information from the execution manager. | |
| Pass/Fail Criteria  Pass if the test is provided and passes. | | |
| Related Information | | |
| History | | |

## Availability/Scalability

|  |  |  |
| --- | --- | --- |
| Stress Test | | |
| ID | | JOBEXEC\_AVAIL\_1 |
| Mandatory | | |
| Applicability | | Implementations of Job Execution Capability |
| Related Requirements | | None |
|  | | |
| Description  The Job Execution Capability should be available under realistic conditions. | | |
| Input from TP  Stress test for the service that calculates the maximum throughput of the service. | | |
| Test 1 | | |
| **Pre-condition** | Correctly configured service. | |
| **Test** | Stress test the service until is not available. | |
| **Expected Outcome** | Throughput of the service. | |
| Pass/Fail Criteria  Pass if the throughput is enough to handle realistic load. TBD: reasonable throughput limit. | | |
| Related Information | | |
| History | | |

|  |  |  |
| --- | --- | --- |
| Service Redundancy | | |
| ID | | JOBEXEC\_AVAIL\_2 |
| Mandatory | | |
| Applicability | | Implementations of Job Execution Capability |
| Related Requirements | | None |
|  | | |
| Description  More than one Job Execution Capability implementation should be able to access a single local batch system concurrently. | | |
| Input from TP  Test for the concurrent access of more than one implementation to a single batch system. | | |
| Test 1 | | |
| **Pre-condition** | More than one Compute Capability configured to use the same execution manager | |
| **Test** | Submission of jobs to all the Compute Capabilities | |
| **Expected Outcome** | All jobs are executed correctly; they are not mixed up in any situation. | |
| Pass/Fail Criteria  Pass if the test is provided and passes. | | |
| Related Information | | |
| History | | |

|  |  |  |
| --- | --- | --- |
| Self-disabling Mechanism | | |
| ID | | JOBEXEC\_AVAIL\_3 |
| Not Mandatory | | |
| Applicability | | Implementations of Job Execution Capability |
| Related Requirements | | None |
|  | | |
| Description  The Job Execution Capability should detect high load conditions and self-disable the job submission in order to maintain the quality of the service. | | |
| Input from TP  Stress test for the service that triggers a self-disabling mechanism. | | |
| Test 1 | | |
| **Pre-condition** | Correctly configured service in high load system. | |
| **Test** | Submission of job. | |
| **Expected Outcome** | Service self-disables submission, a message to the client is sent when the submission is tried. | |
| Pass/Fail Criteria  Pass if the test is provided and passes. | | |
| Related Information | | |
| History | | |

# Parallel Job

## Submission of parallel jobs

|  |  |  |
| --- | --- | --- |
| CREAM Simple Parallel Job Submission | | |
| ID | | PARALLEL\_JOB\_1 |
| Mandatory | | |
| Applicability | | Implementations of CREAM interfaces with parallel job support. |
| Related Requirements | | None |
|  | | |
| Description  Test the submission of a parallel job. | | |
| Input from TP  Test for the submission of parallel jobs. | | |
| Test 1 | | |
| **Pre-condition** | Valid User proxy and valid delegation in the service. | |
| **Test** | Submission of a job requesting more than one execution slot:  Executable = "/bin/sleep";  CPUNumber = 2;  Arguments = "20"; | |
| **Expected Outcome** | Job is submitted and the requested slots are allocated. Unique Identifier for the submitted jobs, status log of the job. | |
| Pass/Fail Criteria  Pass if the test is provided and passes. | | |
| Related Information | | |
| History | | |

|  |  |  |
| --- | --- | --- |
| CREAM Fine grained Parallel Job Submission | | |
| ID | | PARALLEL\_JOB\_2 |
| Not Mandatory | | |
| Applicability | | Implementations of CREAM interface with parallel job support. |
| Related Requirements | | None |
|  | | |
| Description  Test the submission of a parallel job with fine grained specification for the layout of processes. | | |
| Input from TP  Test for the submission of parallel jobs with fine grained specification for the layout of processes: use complete node, specify number of cores per node. | | |
| Test 1 | | |
| **Pre-condition** | Valid User proxy and valid delegation in the service. | |
| **Test** | Submission of job requesting a complete node:  Executable = "/bin/sleep";  NodeNumber = 2;  SMPGranularity = 2;  Arguments = "20"; | |
| **Expected Outcome** | Job is submitted and the requested slots were allocated. Unique Identifier for the submitted jobs, status log of the job. | |
| Test 2 | | |
| **Pre-condition** | Valid User proxy and valid delegation in the service. | |
| **Test** | Submission of job requesting a complete node:  Executable = "/bin/sleep";  NodeNumber = 1;  SMPGranularity = 4;  WholeNode = True;  Arguments = "20"; | |
| **Expected Outcome** | Job is submitted and the requested slots are allocated. Unique Identifier for the submitted jobs, status log of the job. | |
| Pass/Fail Criteria  Pass if the test is provided and passes. | | |
| Related Information | | |
| History | | |

|  |  |  |
| --- | --- | --- |
| ARC Simple Parallel Job Submission | | |
| ID | | PARALLEL\_JOB\_3 |
| Mandatory | | |
| Applicability | | Implementations of ARC-CE interface with parallel job support. |
| Related Requirements | | None |
|  | | |
| Description  Test the submission of a parallel job. | | |
| Input from TP  Test for the submission of parallel jobs. | | |
| Test 1 | | |
| **Pre-condition** | Valid User proxy and valid delegation in the service. | |
| **Test** | Submission of job requesting more than one execution slot:  &(executable="/bin/sleep")  (count="2")  (arguments="20") | |
| **Expected Outcome** | Job is submitted and the requested slots are allocated. Unique Identifier for the submitted jobs, status log of the job. | |
| Pass/Fail Criteria  Pass if the test is provided and passes. | | |
| Related Information | | |
| History | | |

|  |  |  |
| --- | --- | --- |
| BES Simple Parallel Job Submission | | |
| ID | | PARALLEL\_JOB\_4 |
| Mandatory | | |
| Applicability | | Implementations of BES interface with parallel job support. |
| Related Requirements | | None |
|  | | |
| Description  Test the submission of a parallel job. | | |
| Input from TP  Test for the submission of parallel jobs. | | |
| Test 1 | | |
| **Pre-condition** | Valid User proxy and valid delegation in the service. | |
| **Test** | Submission of job requesting more than one execution slot:  {  Executable: "/bin/sleep",  Arguments: ["20"],  Resources: { CPUs: 2, },  } | |
| **Expected Outcome** | Job is submitted and the requested slots are allocated. Unique Identifier for the submitted jobs, status log of the job. | |
| Pass/Fail Criteria  Pass if the test is provided and passes. | | |
| Related Information | | |
| History | | |

|  |  |  |
| --- | --- | --- |
| BES Fine grained Parallel Job Submission | | |
| ID | | PARALLEL\_JOB\_5 |
| Not Mandatory | | |
| Applicability | | Implementations of BES interface with parallel job support. |
| Related Requirements | | None |
|  | | |
| Description  Test the submission of a parallel job with fine grained specification for the layout of processes. | | |
| Input from TP  Test for the submission of parallel jobs with fine grained specification for the layout of processes: use complete node, specify number of cores per node. | | |
| Test 1 | | |
| **Pre-condition** | Valid User proxy and valid delegation in the service. | |
| **Test** | Submission of job requesting a complete node:  {  Executable: "/bin/sleep",  Arguments: ["20"],  Resources:{  CPUsPerNode: 2,  Nodes: 2,  }  } | |
| **Expected Outcome** | Job is submitted and the requested slots were allocated. Unique Identifier for the submitted jobs, status log of the job. | |
| Test 2 | | |
| **Pre-condition** | Valid User proxy and valid delegation in the service. | |
| **Test** | Submission of job requesting a particular process distribution:  {  Executable: "/bin/sleep",  Arguments: ["20"],  Resources: { CPUsPerNode: 4, Nodes: 1 },  ExclusiveExecution: True,  } | |
| **Expected Outcome** | Job is submitted and the requested slots are allocated. Unique Identifier for the submitted jobs, status log of the job. | |
| Pass/Fail Criteria  Pass if the test is provided and passes. | | |
| Related Information | | |
| History | | |

## MPI support

|  |  |  |
| --- | --- | --- |
| Precompiled MPI Job Submission | | |
| ID | | PARALLEL\_MPI\_1 |
| Mandatory | | |
| Applicability | | Implementations of Job Execution interfaces with parallel job support. |
| Related Requirements | | None |
|  | | |
| Description  Test the submission of a precompiled MPI job. | | |
| Input from TP  Test for the submission of a MPI job that is already compiled. | | |
| Test 1 | | |
| **Pre-condition** | Valid User proxy and valid delegation in the service. MPI Binary | |
| **Test** | Submission of a MPI job requesting more than one execution slot with MPI Binary included in input sandbox of job (description of job depending on Job Execution interface) | |
| **Expected Outcome** | Job is submitted and executed without errors; the requested slots are allocated. Unique Identifier for the submitted jobs, status log of the job. | |
| Pass/Fail Criteria  Pass if the test is provided and passes for all the MPI implementations supported. | | |
| Related Information | | |
| History | | |

|  |  |  |
| --- | --- | --- |
| MPI Job Submission with compilation | | |
| ID | | PARALLEL\_MPI\_2 |
| Not Mandatory | | |
| Applicability | | Implementations of Job Execution interfaces with parallel job support. |
| Related Requirements | | None |
|  | | |
| Description  Test the submission of a MPI job that is compiled at the remote site. | | |
| Input from TP  Test for the submission of a MPI job that gets compiled at the remote site. | | |
| Test 1 | | |
| **Pre-condition** | Valid User proxy and valid delegation in the service. Source code for MPI application. | |
| **Test** | Submission of a MPI job requesting more than one execution slot with MPI source code included in input sandbox of job (description of job depending on Job Execution interface). Prior to the execution of the application, the source must be compiled with the available compiler at the site. | |
| **Expected Outcome** | Job is submitted, compiled and executed without errors; the requested slots are allocated. Unique Identifier for the submitted jobs, status log of the job. | |
| Pass/Fail Criteria  Pass if the test is provided and passes for all the MPI implementations supported. | | |
| Related Information | | |
| History | | |

## OpenMP support

|  |  |  |
| --- | --- | --- |
| Precompiled OpenMP Job Submission | | |
| ID | | PARALLEL\_OMP\_1 |
| Mandatory | | |
| Applicability | | Implementations of Job Execution interfaces with parallel job support. |
| Related Requirements | | None |
|  | | |
| Description  Test the submission of a precompiled OpenMP job. | | |
| Input from TP  Test for the submission of a OpenMP job that is already compiled. | | |
| Test 1 | | |
| **Pre-condition** | Valid User proxy and valid delegation in the service. OpenMP Binary | |
| **Test** | Submission of a OpenMP job requesting more than one execution slot with OpenMP Binary included in input sandbox of job (description of job depending on Job Execution interface) | |
| **Expected Outcome** | Job is submitted and executed without errors; the requested slots are allocated. Unique Identifier for the submitted jobs, status log of the job. | |
| Pass/Fail Criteria  Pass if the test is provided and passes for all the OpenMP implementations supported. | | |
| Related Information | | |
| History | | |

|  |  |  |
| --- | --- | --- |
| OpenMP Job Submission with compilation | | |
| ID | | PARALLEL\_OMP\_1 |
| Not Mandatory | | |
| Applicability | | Implementations of Job Execution interfaces with parallel job support. |
| Related Requirements | | None |
|  | | |
| Description  Test the submission of an OpenMP job that is compiled at the remote site. | | |
| Input from TP  Test for the submission of an OpenMP job that gets compiled at the remote site. | | |
| Test 1 | | |
| **Pre-condition** | Valid User proxy and valid delegation in the service. Source code for OpenMP application. | |
| **Test** | Submission of an OpenMP job requesting more than one execution slot with OpenMP source code included in input sandbox of job (description of job depending on Job Execution interface). Prior to the execution of the application, the source must be compiled with the available compiler at the site. | |
| **Expected Outcome** | Job is submitted, compiled and executed without errors; the requested slots are allocated. Unique Identifier for the submitted jobs, status log of the job. | |
| Pass/Fail Criteria  Pass if the test is provided and passes for all the OpenMP implementations supported. | | |
| Related Information | | |
| History | | |

# Job Scheduling

Compute Job Scheduling capability refers to the ‘end-to-end’ service that can be delivered to a user in response to their request for a job to be run. This includes managing the selection of the most appropriate resource that meets the user’s requirements, the transfer of any files required as input or produced as output between their source or destination storage location and the selected computational resource, and the management of any data transfer or execution failures within the infrastructure.

## gLite WMS

These QC refer to implementation of the Compute Job Scheduling capability using the WMProxy interface [R 8]. May be deprecated in future versions of the QC.

|  |  |  |
| --- | --- | --- |
| WMProxy API TestSuite | | |
| ID | | JOBSCH\_WMS\_API\_1 |
| Mandatory | | |
| Applicability | | Implementations of WMProxy interface. |
| Related Requirements | | None |
|  | | |
| Description  Test the all the functions of the WMProxy interface. | | |
| Input from TP  Complete Test suite for the WMPRoxt API. It must include tests for all the documented functions in the WSDL.  For all functions, check both correct and invalid input. Invalid output should throw an exception as documented. Test also with valid and invalid credentials. Invalid credentials should throw security related exceptions. | | |
| Test Suite Description | | |
| **Pre-condition** | Valid user credentials. | |
| **Test** | Test all WMProxy documented functions, with correct/incorrect input and with valid and invalid credentials. | |
| **Expected Outcome** | Log of all the operations performed. All the documented functions work as documented. | |
| Pass/Fail Criteria  Pass if the testsuite is provided and passes. | | |
| Related Information | | |
| History | | |

|  |  |  |
| --- | --- | --- |
| JSDL Submission | | |
| ID | | JOBSCH\_WMS\_API\_2 |
| Non Mandatory | | |
| Applicability | | Implementation of gLite WMS |
| Related Requirements | | None |
|  | | |
| Description  Job Scheduling services should allow submission of jobs described with JSDL language | | |
| Input from TP  Testsuite for submission of JSDL jobs covering different kinds of jobs and with several parameters, as much complete as possible. Test will check job submission and correct completion. A third party test can be provided to check it is JSDL compliant. | | |
| Test 1 | | |
| **Pre-condition** | Valid user credentials and delegation in the service. | |
| **Test** | Submission of a provided JSDL compliant job, and follow the job until it reaches final state. | |
| **Expected Outcome** | A complete job output submission, until reaching final state (success/failure) | |
| Pass/Fail Criteria  Will pass if every JSDL is analized and completed. In case of JSDL syntax error, it must be reported as well. Will fail if a JSDL compliant job is not accepted by the job scheduling services, or if it does not behave as expected by the definition of the job | | |
| Related Information | | |
| History | | |

|  |  |  |
| --- | --- | --- |
| Proxy Renewal | | |
| ID | | JOBSCH\_WMS\_API\_3 |
| Mandatory | | |
| Applicability | | Implementations of gLite WMS. |
| Related Requirements | |  |
|  | | |
| Description  The WMS must manage the user credentials and renew them if necessary. | | |
| Input from TP  Test of the proxy renewal functionality that checks what happens when the user credential expires and the job is still running. | | |
| Test 1 | | |
| **Pre-condition** | Valid user credentials (short duration) and delegation in the service. Credentials Renewal service available. | |
| **Test** | Submit job that takes longer to complete that the credential lifetime. | |
| **Expected Outcome** | Job executes successfully. The scheduling services should perform a proxy renewal and state it in the log messages (if there is an error, log it also). Output of the job, and status messages stating the renewal or not of the user credentials. | |
| Pass/Fail Criteria  Will Pass if the proxy renewal is done, or if there is an error logged stating the problem. Will fail if there is no clear information about the process. | | |
| Related Information | | |
| History | | |

### End-to-end job tests

|  |  |  |
| --- | --- | --- |
| Simple Job Submission | | |
| ID | | JOBSCH\_WMS\_JOB\_1 |
| Mandatory | | |
| Applicability | | Implementations of gLite WMS |
| Related Requirements | | None |
|  | | |
| Description  Test the job submission for simple job. | | |
| Input from TP  Test for the submission of a job. | | |
| Test 1 | | |
| **Pre-condition** | Valid user credentials and delegation in the service. | |
| **Test** | Job submission for simple job:  Executable = /bin/sleep;  Arguments = "120"; | |
| **Expected Outcome** | Job is submitted and finishes its execution correctly, all states of the job must be logged correctly. | |
| Pass/Fail Criteria  Pass if the test is provided and passes. | | |
| Related Information | | |
| History | | |

|  |  |  |
| --- | --- | --- |
| Simple Job Submission with files | | |
| ID | | JOBSCH\_WMS\_JOB\_2 |
| Mandatory | | |
| Applicability | | Implementations of gLite WMS |
| Related Requirements | | None |
|  | | |
| Description  Test the job submission for simple job with input and output files. | | |
| Input from TP  Test for the submission of a job with input and output files. | | |
| Test 1 | | |
| **Pre-condition** | Valid user credentials and delegation in the service.  Non-empty file “myfile” | |
| **Test** | Job submission for simple job with input and output files:  Executable = "/bin/ls";  Arguments = "-l";  StdOutput = "std.out";  StdError = "std.err";  InputSandbox = {"myfile"};  OutputSandbox = {"std.out", "std.err"}; | |
| **Expected Outcome** | Job is submitted and finishes its execution correctly and the output of the job contains the listing of the directory including the input file with correct size, all states of the job must be logged correctly. | |
| Pass/Fail Criteria  Pass if the test is provided and passes. | | |
| Related Information | | |
| History | | |

|  |  |  |
| --- | --- | --- |
| Job Cancel | | |
| ID | | JOBSCH\_WMS\_JOB\_3 |
| Mandatory | | |
| Applicability | | Implementations of gLite WMS |
| Related Requirements | | None |
|  | | |
| Description  Test the job cancellation for a job. | | |
| Input from TP  Test for the cancellation of a job. | | |
| Test 1 | | |
| **Pre-condition** | Valid user credentials and delegation in the service. | |
| **Test** | Job submission and then cancellation for simple job:  Executable = "/bin/sleep";  Arguments = "20m";. | |
| **Expected Outcome** | Job is submitted and cancelled correctly at the Computing Capability. | |
| Pass/Fail Criteria  Pass if the test is provided and passes. | | |
| Related Information | | |
| History | | |

|  |  |  |
| --- | --- | --- |
| Parallel Job Submission | | |
| ID | | JOBSCH\_WMS\_JOB\_4 |
| Mandatory | | |
| Applicability | | Implementations of gLite WMS |
| Related Requirements | | None |
|  | | |
| Description  Test the submission of a parallel job. | | |
| Input from TP  Test for the submission of parallel jobs. | | |
| Test 1 | | |
| **Pre-condition** | Valid user credentials and delegation in the service. | |
| **Test** | Job submission for simple job:  Executable = "/bin/sleep";  CPUNumber = 2;  Arguments = "20"; | |
| **Expected Outcome** | Job is submitted and executed correctly. The requested slots where allocated. | |
| Pass/Fail Criteria  Pass if the test is provided and passes. | | |
| Related Information | | |
| History | | |

|  |  |  |
| --- | --- | --- |
| Job List Match and Rank | | |
| ID | | JOBSCH\_WMS\_JOB\_5 |
| Mandatory | | |
| Applicability | | Implementations of gLite WMS |
| Related Requirements | | None |
|  | | |
| Description  Test the list match for jobs. | | |
| Input from TP  Test for the list match functionality. | | |
| Test 1 | | |
| **Pre-condition** | Valid user credentials and delegation in the service. | |
| **Test** | Job list match for job with requirements and rank expressions:  Executable = "/bin/sleep";  Requirements = other.GlueCEStateStatus = "Production";  Rank = -other.GlueCEStateEstimatedResponseTime; | |
| **Expected Outcome** | List of resources with correct rank is returned. | |
| Pass/Fail Criteria  Pass if the test is provided and passes. | | |
| Related Information | | |
| History | | |

|  |  |  |
| --- | --- | --- |
| Parametric Job Submission | | |
| ID | | JOBSCH\_WMS\_JOB\_6 |
| Mandatory | | |
| Applicability | | Implementations of gLite WMS |
| Related Requirements | | None |
|  | | |
| Description  Test the submission of a parametric job. | | |
| Input from TP  Test for the submission of parametric jobs. | | |
| Test 1 | | |
| **Pre-condition** | Valid user credentials and delegation in the service. | |
| **Test** | Job submission of job with numeric parameters (Parameters = 10000;ParameterStart = 1000; ParameterStep = 10;). | |
| **Expected Outcome** | Job is executed correctly. List of JobIds for the parametric jobs and each of the subjobs is obtained; all states of the jobs must be logged correctly. | |
| Test 2 | | |
| **Pre-condition** | Valid user credentials and delegation in the service. | |
| **Test** | Job submission of job with a list of parameters (Parameters={A, B, C,…}). | |
| **Expected Outcome** | Job is executed correctly. List of JobIds for the parametric jobs and each of the subjobs is obtained; all states of the jobs must be logged correctly. | |
| Pass/Fail Criteria  Pass if the test is provided and passes. | | |
| Related Information | | |
| History | | |

|  |  |  |
| --- | --- | --- |
| Job Collection Submission | | |
| ID | | JOBSCH\_WMS\_JOB\_7 |
| Mandatory | | |
| Applicability | | Implementations of gLite WMS |
| Related Requirements | | None |
|  | | |
| Description  Test the submission of a job collection. | | |
| Input from TP  Test for the submission of job collections. | | |
| Test 1 | | |
| **Pre-condition** | Valid user credentials and delegation in the service. | |
| **Test** | Job submission for job collection. | |
| **Expected Outcome** | Job is executed correctly. List of JobIds for the job collections and each of the subjobs is obtained; all states of the jobs must be logged correctly. | |
| Pass/Fail Criteria  Pass if the test is provided and passes. | | |
| Related Information | | |
| History | | |

|  |  |  |
| --- | --- | --- |
| DAG Submission | | |
| ID | | JOBSCH\_WMS\_JOB\_8 |
| Mandatory | | |
| Applicability | | Implementations of gLite WMS |
| Related Requirements | | None |
|  | | |
| Description  Test the submission of a DAG. | | |
| Input from TP  Test for the submission DAGs. | | |
| Test 1 | | |
| **Pre-condition** | Valid user credentials and delegation in the service. | |
| **Test** | Job submission for DAG. | |
| **Expected Outcome** | Job is executed correctly. List of JobIds for DAG and each of the subjobs is obtained; all states of the jobs must be logged correctly. | |
| Pass/Fail Criteria  Pass if the test is provided and passes. | | |
| Related Information | | |
| History | | |

|  |  |  |
| --- | --- | --- |
| Job Resubmission | | |
| ID | | JOBSCH\_WMS\_JOB\_9 |
| Mandatory | | |
| Applicability | | Implementations of gLite WMS |
| Related Requirements | | Requirements gathered in MS305 related to resubmission of jobs, and information provided in error messages. |
|  | | |
| Description  Job failures due to resource malfunctioning and not to the job itself must be resubmitted to other resources, with a configurable amount of repetitions. In the case of job failures due to the job itself must be resubmitted with a configurable amount of repetitions. In botch situations, status must reflect clearly what is the cause of resubmission, new resource selected and attempt number | | |
| Input from TP  Test and for checking resubmission mechanisms | | |
| Test 1 | | |
| **Pre-condition** | Valid user credentials and delegation in the service. | |
| **Test** | Job submission that fails due to simulated remote resource malfunctioning. | |
| **Expected Outcome** | Job is resubmitted to other resource. Log of all failures and a complete trace of the job. | |
| Test 2 | | |
| **Pre-condition** | Valid user credentials and delegation in the service. | |
| **Test** | Job submission for job that always fails (e.g. exit code 1) | |
| **Expected Outcome** | Job is resubmitted until resubmission attempts reach the configured limit. Log of all failures and a complete trace of the job. | |
| Pass/Fail Criteria  Pass if the test is provided and produces a complete trace of the job failures due to remote causes or the job itself | | |
| Related Information | | |
| History | | |

|  |  |  |
| --- | --- | --- |
| JDL Acceptance limits | | |
| ID | | JOBSCH\_WMS\_JOB\_10 |
| Non Mandatory | | |
| Applicability | | Implementations of gLite WMS |
| Related Requirements | |  |
|  | | |
| Description  The service should accept JDLs without size restrictions. | | |
| Input from TP  A test to submit a job and check if it is accepted or rejected, specially for big JDLs. | | |
| Test 1 | | |
| **Pre-condition** | Valid user credentials and delegation in the service. | |
| **Test** | Submission of job descriptions (specially large) | |
| **Expected Outcome** | Normal job submission if everything is correct; an error message if any problem arises. | |
| Pass/Fail Criteria  Will Pass if JDL is correct, and submits the job or if there is a report on a known sysntax error in the jdl. Will Fail if a wrong Jdl is accepted or if it crashes | | |
| Related Information | | |
| History | | |

## Job Execution Capability Support

These QC refer to the interaction of the Job Scheduling Capability with the underlying Job Execution Capability implementations for the work items submitted. Job Scheduling Capabilities are expected to support the most common Job Execution Capability Implementations used in the current EGI infrastructure: CREAM, ARC and UNICORE

|  |  |  |
| --- | --- | --- |
| Job Submission | | |
| ID | | JOBSCH\_EXEC\_1 |
| Mandatory | | |
| Applicability | | Implementations of Job Scheduling Capability |
| Related Requirements | | None |
|  | | |
| Description  The Job Scheduling Capability must be able to submit, manage and monitor jobs to the underlying Job Execution Capability. | | |
| Input from TP  Test for job submission. | | |
| Test 1 | | |
| **Pre-condition** | None | |
| **Test** | Job submission to Job Execution Capability | |
| **Expected Outcome** | Job is submitted to Job Execution Capability, a valid Job ID is returned. | |
| Test 2 | | |
| **Pre-condition** | Already submitted job. | |
| **Test** | Query job status in the Job Execution Capability. | |
| **Expected Outcome** | Job status can be fetched, show a message with it. | |
| Test 3 | | |
| **Pre-condition** | Already submitted job. | |
| **Test** | Cancel job in the Job Execution Capability. | |
| **Expected Outcome** | Job is successfully cancelled. | |
| Pass/Fail Criteria  Pass if the test is provided and passes for each of the supported Job Execution capability Implementations. | | |
| Related Information  Job Execution Capability Criteria (see Section 2 of this document) | | |
| History | | |

## Service availability, monitoring and error handling.

|  |  |  |
| --- | --- | --- |
| Service ping | | |
| ID | | JOBSCH\_SERVICE\_1 |
| Mandatory | | |
| Applicability | | Implementations of Job Scheduling Capability |
| Related Requirements | |  |
|  | | |
| Description  Check if all implied services respond to a ping. | | |
| Input from TP  A test to check all required services are running from a remote machine. This is not a script to check daemons are running, but the services are reachable from a remote machine (usually a User Interface) | | |
| Test 1 | | |
| **Pre-condition** | Valid user credentials, information discovery service available. | |
| **Test** | Contact remote machine providing job scheduling and perform a ping test. | |
| **Expected Outcome** | Name of remote service and its reachable status (Ok /Fail) | |
| Pass/Fail Criteria  Test will effectively contact remote services, and provide their status. | | |
| Related Information | | |
| History | | |

|  |  |
| --- | --- |
| Error Messages | |
| ID | JOBSCH\_SERVICE\_2 |
| Non Mandatory | |
| Applicability | Implementations of Job Scheduling Capability. Applicable for every service, and specially for the command line interface |
| Related Requirements |  |
|  | |
| Description  The error messages provided by the service should be clear and facilitate the solution of those errors. | |
| Input from TP  For every service a list of possible errors that can appear must be provided. In case of command line interface, this list has to be exhaustive to all the messages that a user can obtain from its usage.  The list of messages have to contain the following fields:   * Error code (if applicable) * Error message * Error source (internal module or remote resource (specify it explicitly)) * Cause of error (syntax error, module malfunctioning, configuration problem, network error, other (specify it explicit)) * Type (critical, informative) | |
| Pass/Fail Criteria  A complete list of errors per service is provided | |
| Related Information | |
| History | |

|  |  |  |
| --- | --- | --- |
| Service Information | | |
| ID | | JOBSCH\_SERVICE\_3 |
| Mandatory | | |
| Applicability | | Implementations of Job Scheduling Capability. |
| Related Requirements | |  |
|  | | |
| Description  Job Scheduling services should publish information about themselves. | | |
| Input from TP  Test for information generation about the service status. | | |
| Test 1 | | |
| **Pre-condition** | Configured system, Information Discovery Capability available. | |
| **Test** | Generate service information and publish to Information Discovery Capability | |
| **Expected Outcome** | Information is produced and can be accessed through the Information Discovery Capability. | |
| Pass/Fail Criteria  Pass if the test is provided and passes. | | |
| Related Information | | |
| History | | |

|  |  |  |
| --- | --- | --- |
| Stress Test | | |
| ID | | JOBSCH\_SERVICE\_4 |
| Mandatory | | |
| Applicability | | Implementations of Job Scheduling Capability |
| Related Requirements | | None |
|  | | |
| Description  The Job Scheduling Capability should be available under realistic conditions. | | |
| Input from TP  Stress test for the service that calculates the maximum throughput of the service. | | |
| Test 1 | | |
| **Pre-condition** | Correctly configured service. | |
| **Test** | Stress test the service until is not available. | |
| **Expected Outcome** | Maximum throughput of the service. | |
| Pass/Fail Criteria  Pass if the maximum throughput is enough for realistic use of the service. TBD: reasonable throughput limit | | |
| Related Information | | |
| History | | |

|  |  |  |
| --- | --- | --- |
| Self-disabling Mechanism | | |
| ID | | JOBSCH\_SERVICE\_5 |
| Not Mandatory | | |
| Applicability | | Implementations of Job Scheduling Capability |
| Related Requirements | | None |
|  | | |
| Description  The Job Scheduling Capability should detect high load conditions and self-disable the job submission in order to maintain the quality of the service. | | |
| Input from TP  Stress test for the service that triggers a self-disabling mechanism. | | |
| Test 1 | | |
| **Pre-condition** | Correctly configured service in high load system. | |
| **Test** | Submission of job. | |
| **Expected Outcome** | Service self-disables submission, a message to the client is sent when the submission is tried. | |
| Pass/Fail Criteria  Pass if the test is provided and passes. | | |
| Related Information | | |
| History | | |

# References

|  |  |
| --- | --- |
|  | UMD roadmap: <https://documents.egi.eu/public/ShowDocument?docid=100> |
|  | Generic UMD Quality Criteria |
|  | Security Capabilities Quality Criteria |
|  | Operational Capabilities Quality Criteria |
|  | CREAM: <http://grid.pd.infn.it/cream/> |
|  | A. Konstantinov, ARC Computational Job Management Component – A-REX, NORDUGRID-TECH-14 |
|  | OGSA Basic Execution Service v1.0: <http://www.ogf.org/documents/GFD.108.pdf> |
|  | gLite WMS: <http://web.infn.it/gLiteWMS/> |