

Table of Contents

WMS Service tests.....	1
Available Functionality tests.....	1
WMS-delegation-test.py.....	1
WMS-job-cancel.py.....	1
WMS-job-cycle.py.....	1
WMS-job-parametric.py.....	2
WMS-job-resubmit.py.....	2
WMS-listmatch-test.py.....	3
WMS-prologue-epilogue.py.....	3
WMS-proxy-renewal.py.....	3

WMS Service tests

Package WMS-service contains tests for automating functional testing of the WMS service. The main goal of available tests is to check the behavior of all the critical operations in the WMS service. Generally, each test script contains a number of sub-tests for a specific WMS operation. Tests at this package give user the possibility to run only some specific sub-tests through a new command line option (-t).

The syntax of -t option is flexible as users can specify either the number of specific sub-tests to run (-t 1,3,7) or a range of sub-tests (-t 2-4) or use a combination (-t 1,3-5). Also, there is another new command line option (-s) which lists the descriptions and numbers of all the available sub-tests.

Available Functionality tests

WMS-delegation-test.py

Test delegation operation.

- Try to delegate using option --autm-delegation
- Try to delegate using option --delegationid
- Try to delegate with a short proxy and check the validity

success means that delegation is stored in the WMS and that we are able to submit a job.

The following tests are possible only in interactive mode (i.e. use option -i) :

- Try to delegate with an expired proxy
- Try to submit with an expired delegation

success if operation fails with the expected reason

WMS-job-cancel.py

Test cancel operation

- Try to cancel a normal job
- Try to cancel a DAG job
- Try to cancel one node of a DAG job
- Try to cancel a collection job
- Try to cancel one node of a collection job
- Try to cancel a parametric job
- Try to cancel one node of a parametric job
- Try to cancel a MPI job

Success means that the final state of the job is "Canceled"

- Try to cancel a finished job.

Success if command fails.

WMS-job-cycle.py

Test a complete job cycle: from submission to get job output

- submission of a normal job to an LCG-CE
- submission of a normal job to a CREAM CE
- submission of a normal job (without specific target)
- submission of a bulk of jobs to an LCG-CE
- submission of a bulk of jobs to a CREAM CE
- submission of a bulk of jobs (without specific target)
- submission of a parametric job to an LCG-CE
- submission of a parametric job to a CREAM CE
- submission of a parametric job (without specific target)
- submission of a DAG job
- submission of a MPI job to an LCG-CE
- submission of a MPI job to a CREAM CE
- submission of a MPI job (without specific target)
- submission of a perusal job

success if job terminated correctly and the retrieved output files are the expected files.

You can try submission to a specific CE using command's option -c

WMS-job-parametric.py

Test parametric job.

- Test integer version of Parameters
- Test integer version of Parameters changing 'start' and 'step' values
- Test list version of Parameters

For these three tests job is only register and the produced jdl is checked.

- Test submission and check the output
- Test submission using a list as parameters

Success means that job finished with Done Success, that the required produced output files are correctly retrieved and that they are as expected.

Known issues:

- bug #79027 ParameterStart must be equal to 0
- bug #79144 NodesCollocation is not support
- bug #79165 Parameters list must be composed by strings
- bug #79308 parameters should not be negative (SegFault)

WMS-job-resubmit.py

Test the two types of possible resubmissions.

- shallow resubmission
- deep resubmission

Success if job fails with the expected reason

WMS-listmatch-test.py

Test the list-match operation.

- try a simple matching
- test delegation proxy option `-delegationid` (delegate before the matchmaking)

success if command returns 0

- test a restricted matching (match only CREAM Ces), success if none LCG CE matches
- try a failure matching (`Requirements == false`), success if no matchmaking is found
- try a simple matching with data requirements, success if listed Ces are "close" to the used S

WMS-prologue-epilogue.py

Test jdl's attributes Prologue (PrologueArguments) and Epilogue (EpilogueArguments).

- Test prologue attribute
- Test epilogue attribute
- Test prologue and epilogue attributes

Success means that job has been correctly submitted, and that at the end output files are correctly retrieved. Then some checks are done on the output files written by the prologue, the executable and the epilogue scripts.

WMS-proxy-renewal.py

- Test proxy renewal operation. All tests are possible only in interactive mode (i.e. use option `-i`)
- Test proxy renewal with LCG CE
- Test proxy renewal with CREAM CE

success means that job finished successfully

- Test proxy renewal with LCG CE (without setting `MYPROXYSERVER`)
- Test proxy renewal with CREAM CE (without setting `MYPROXYSERVER`)

success if jobs fails because of proxy expiration

This topic: WMS > WmsServiceTestSuite

Topic revision: r1 - 2011-07-13 - AlessioGianelle



Copyright © 2008-2021 by the contributing authors. All material on this collaboration platform is the property of the contributing authors.

Ideas, requests, problems regarding TWiki? Send feedback