

The SAMGrid Test Harness

Matthew Leslie

CHEP 2004



Fermilab

Overview

- Testing Methodologies
- Features of the Test Harness
- Operational testing example
- Performance tuning example
- Other testing software
- Conclusions

Testing Methodologies



- Unit Tests
 - Tests an individual piece of code meets its specification
- Stress Tests
 - Tests code under a controlled load
 - Can reveal race conditions in multithreaded code
- Performance Tests
 - Getting optimal performance from complex systems often requires tuning many parameters.



Design Goals

- Eliminate heterogeneous testing mechanisms
- Allow all test types within a single framework
- Simple configuration
- Readable Output
- Easy to use
- Easy to add new tests
- Well documented

Features for Unit Testing



- Test Options
 - Required or Optional
- Test dependencies
 - Run test B only if test A passed
- Run a command line program
 - Check return code
 - Check stdout or stderr for a string



Features for Stress Testing

- Concurrent Tests
 - Any test can be run in its own thread simply by setting the fork parameter in the configuration file
 - Allows a unit test to become a stress test
- Looping over Tests
 - You may run the tests in the suite several times
- Tests automatically timed
 - Check performance is still OK

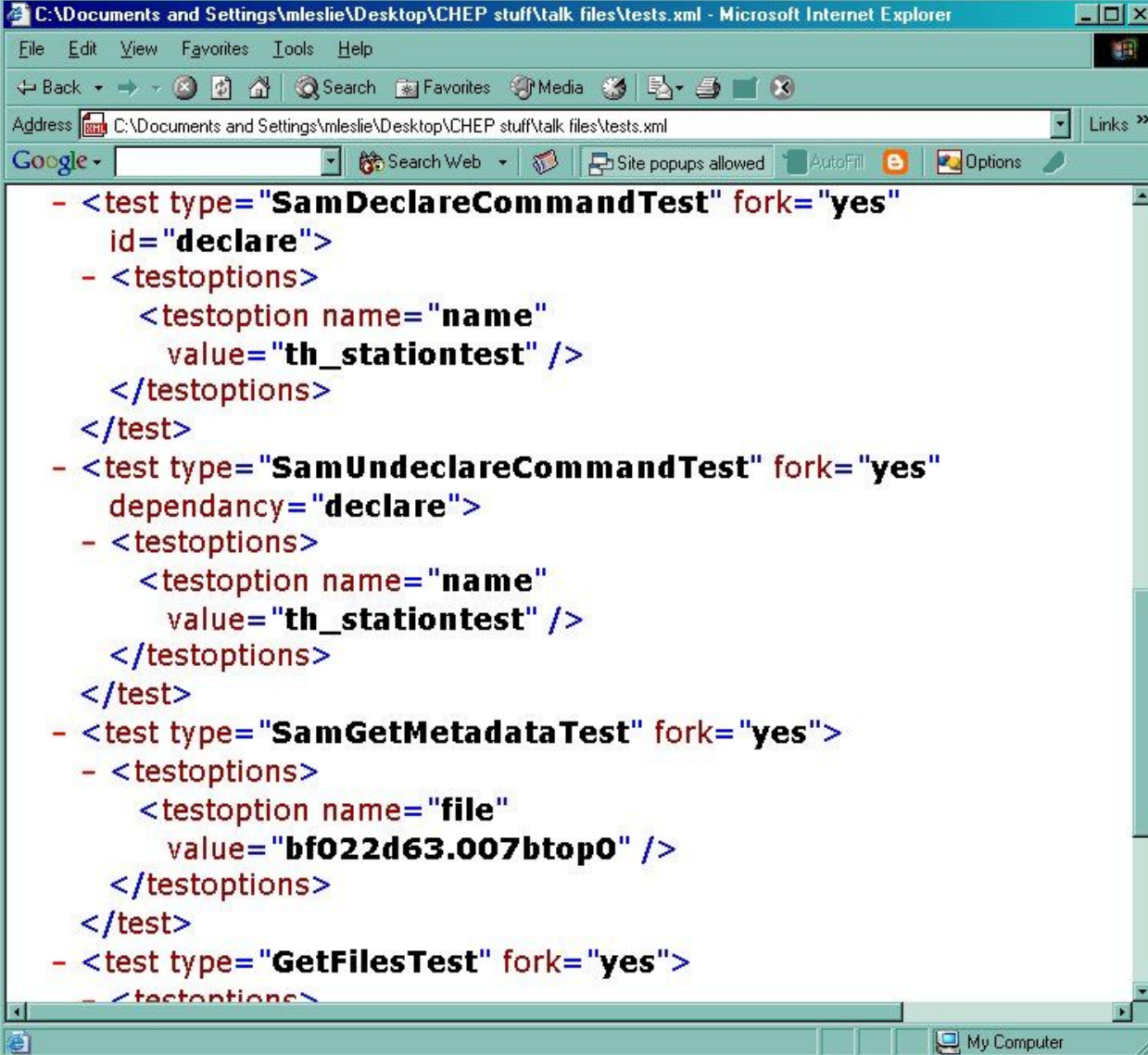
Features for Performance Tuning

- Simulate real world load
 - Randomized delays between tests
- Monitoring
 - Register a 'performance monitor' object and have its output logged at regular intervals
 - Performance information written with test report
 - e.g. Plot load average while test is running

Simplifying Configuration

- Single XML configuration file with self explanatory syntax
- Details a suite of tests
- Each test configured by test option sub tags
- Harness checks that all required test options are set

Example Configuration



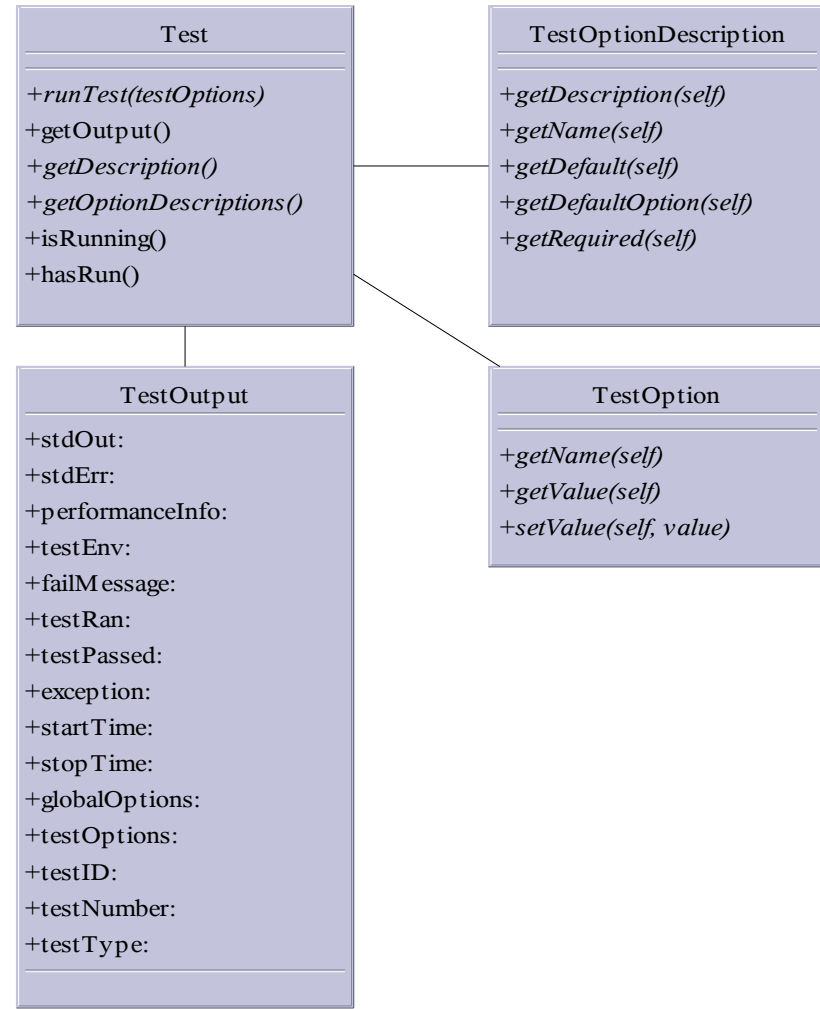
The image shows a screenshot of a Microsoft Internet Explorer browser window. The title bar reads "C:\Documents and Settings\mleslie\Desktop\CHEP stuff\talk files\tests.xml - Microsoft Internet Explorer". The address bar shows the file path "C:\Documents and Settings\mleslie\Desktop\CHEP stuff\talk files\tests.xml". The main content area displays XML code for test configurations. The code is as follows:

```
- <test type="SamDeclareCommandTest" fork="yes"
  id="declare">
  - <testoptions>
    <testoption name="name"
      value="th_stationtest" />
    </testoptions>
  </test>
- <test type="SamUndeclareCommandTest" fork="yes"
  dependancy="declare">
  - <testoptions>
    <testoption name="name"
      value="th_stationtest" />
    </testoptions>
  </test>
- <test type="SamGetMetadataTest" fork="yes">
  - <testoptions>
    <testoption name="file"
      value="bf022d63.007btop0" />
    </testoptions>
  </test>
- <test type="GetFilesTest" fork="yes">
  - <testoptions>
```

The browser's taskbar at the bottom shows the "My Computer" icon.

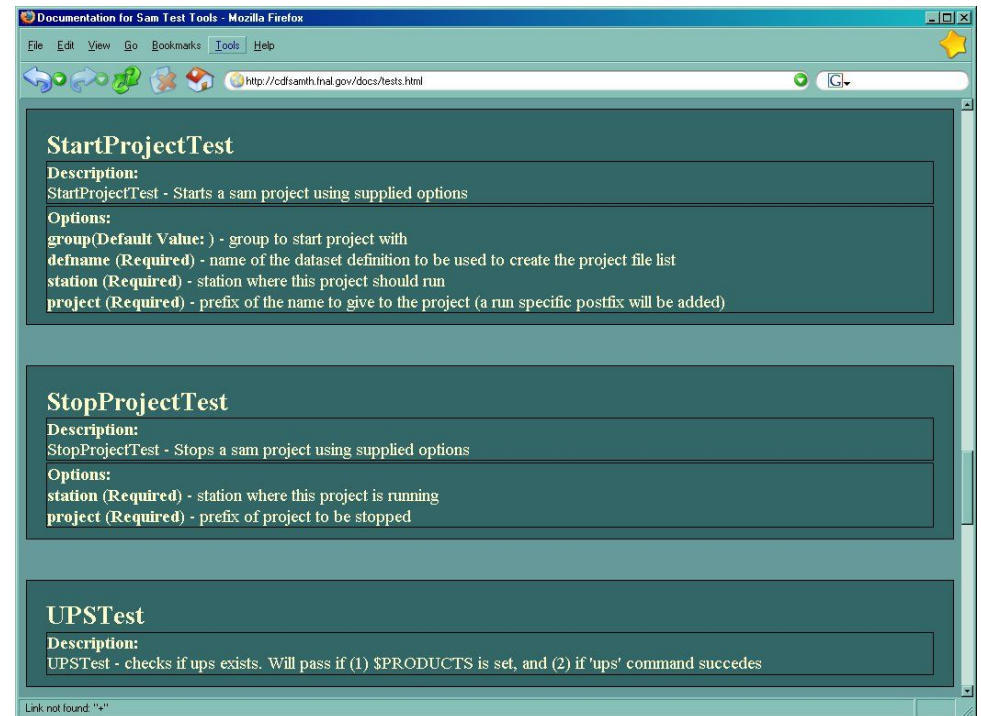
Easily Extended

- Inherit from test object.
- Provide runTest method
- Provide option descriptions
- ~10 lines of code



Documentation

- Automatic!
 - Introspective methods
 - Check Configuration
 - Generate Documentation



Readable Output

- Real time output
 - Output from parallel processes Interleaved
- Color coded HTML report
 - Separates each processes output
 - Tests color coded depending on success
- XML output
 - Integrate with other tools or build scripts

Example Output

The screenshot shows a Mozilla Firefox browser window titled "Sam Test Harness - Mozilla Firefox". The address bar contains the URL "http://cdfsamth.fnal.gov/archive/health/testout/WedJun091754.html". The main content area displays two test results:

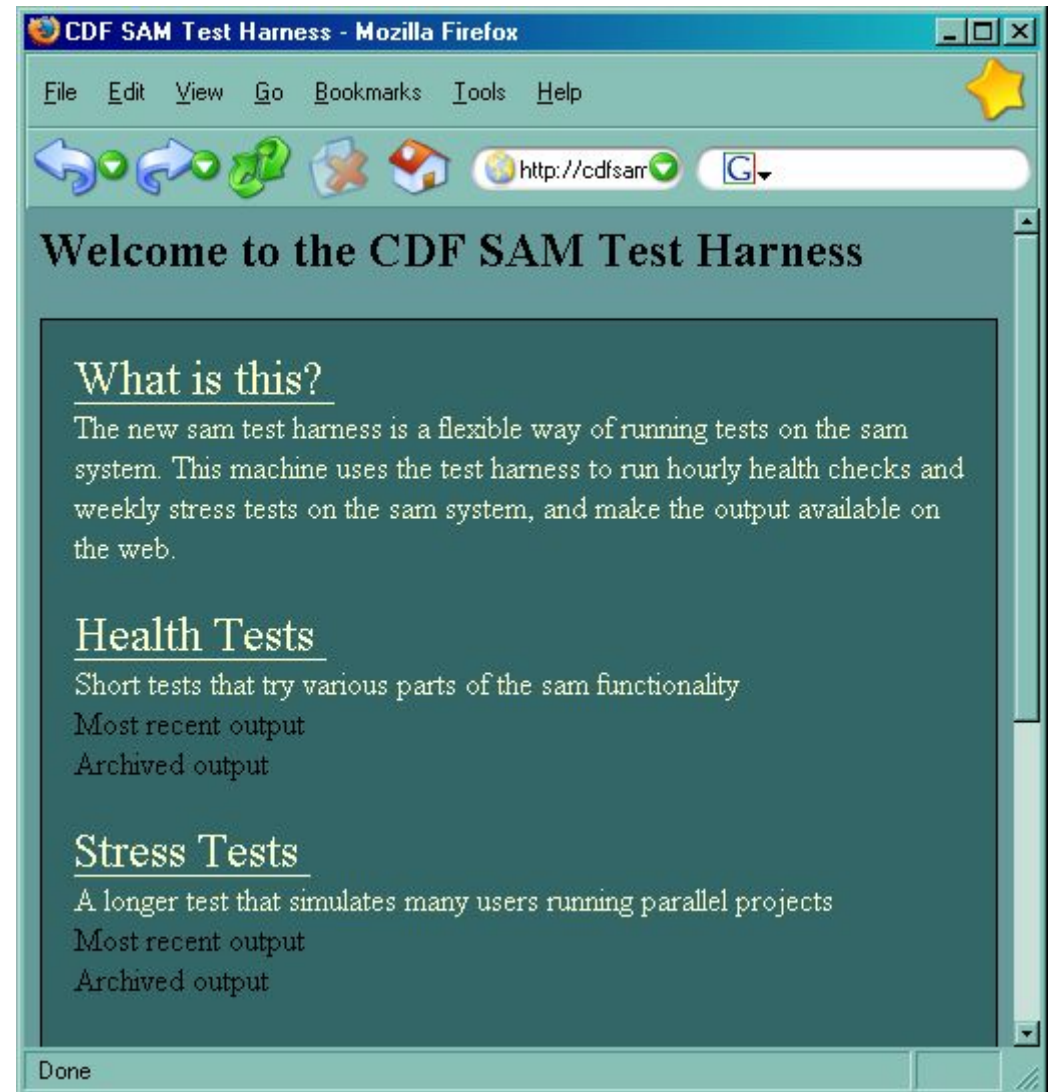
SamLocateTest:
Passed
Test Number:2 Start Time:Wed Jun 9 17:54:30 2004 End Time:Wed Jun 9 17:54:31 2004
Test Options:
file: bf022d63.007btop0

StartProjectTest:
Failed
Test Number:3 ID:start Start Time:Wed Jun 9 17:54:31 2004 End Time:Wed Jun 9 17:54:34 2004
Test Options:
group:
defname: st-onesmall
station: cdf-harness
project: th_stationTest
Standard Error:
Invalid Command:
Missing RequiredOption(s): --group=

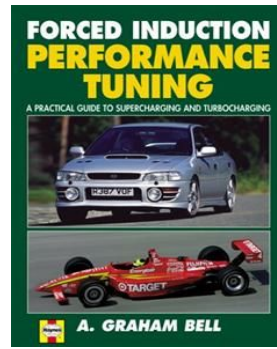
The status bar at the bottom of the browser window shows "Done".

Operational testing

- Hourly health checks
- Output on web
- Help diagnose user problems



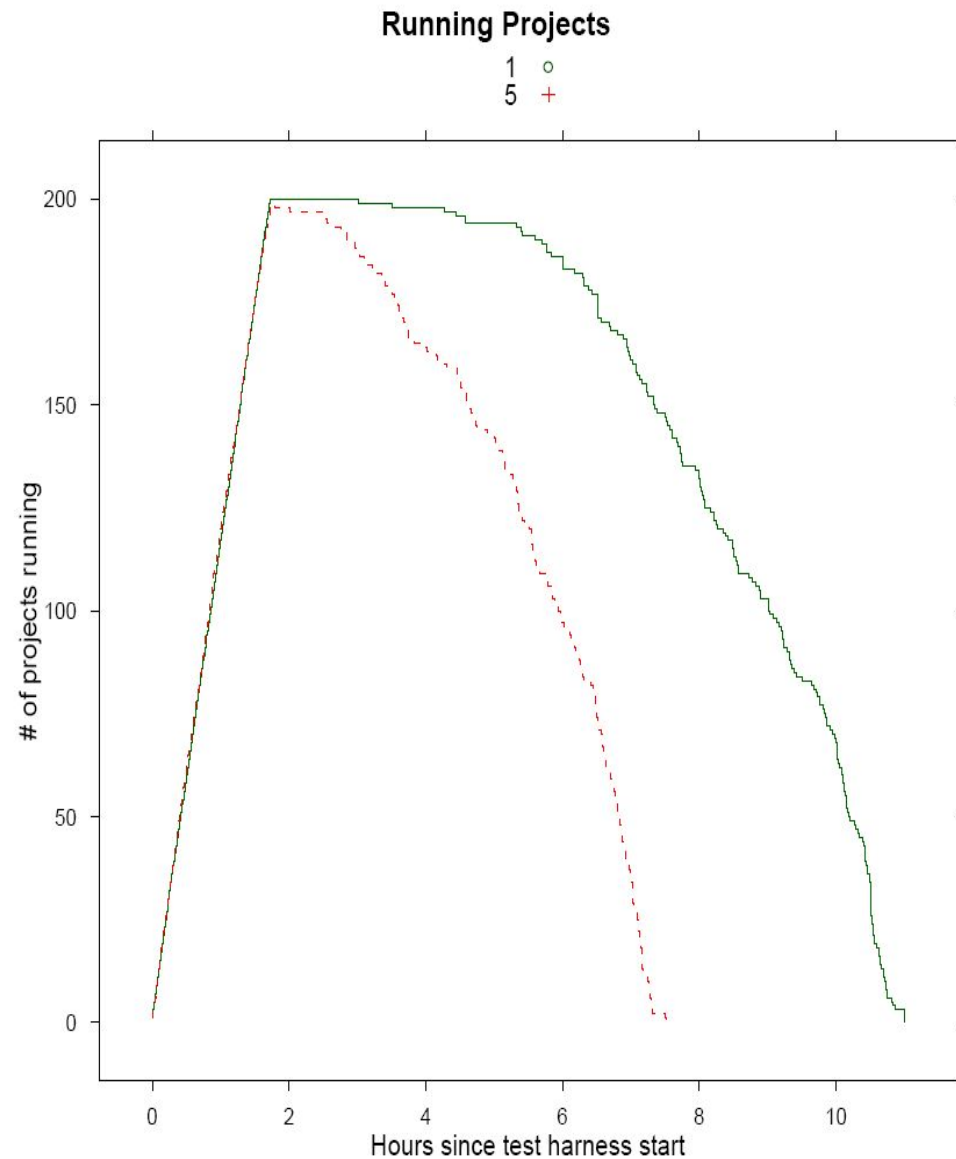
Performance tuning



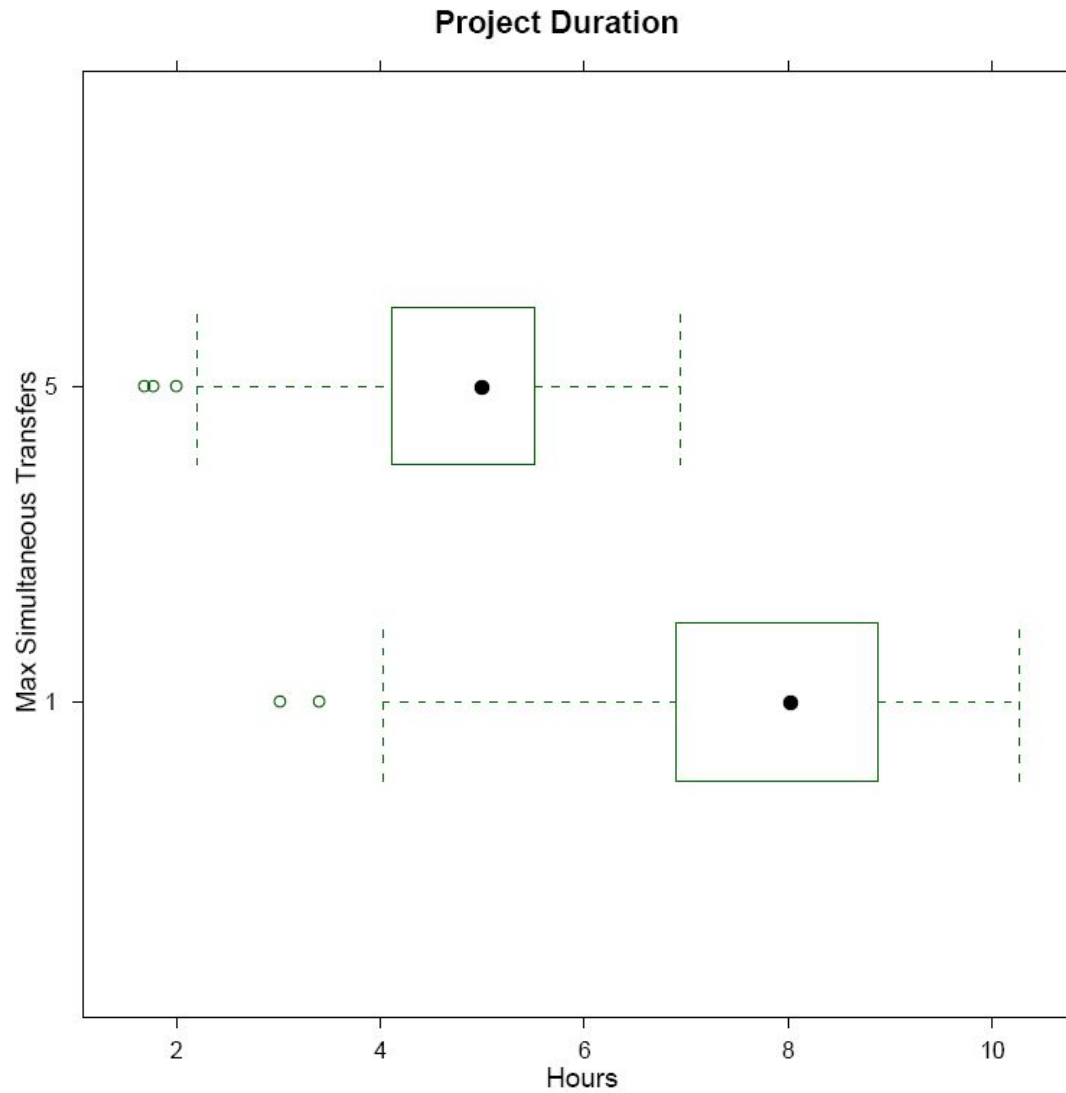
- Optimize a 128 CPU compute farm
- SAM caches files on disk, many files stored on tape
- Minimize wait times for tape transfers
- Optimize 'max-transfers' parameter
 - Controls how many files a node will transfer at once
- Try two extreme cases, 5 and 1 transfer.
- Harness starts 200 Projects

The number of projects running

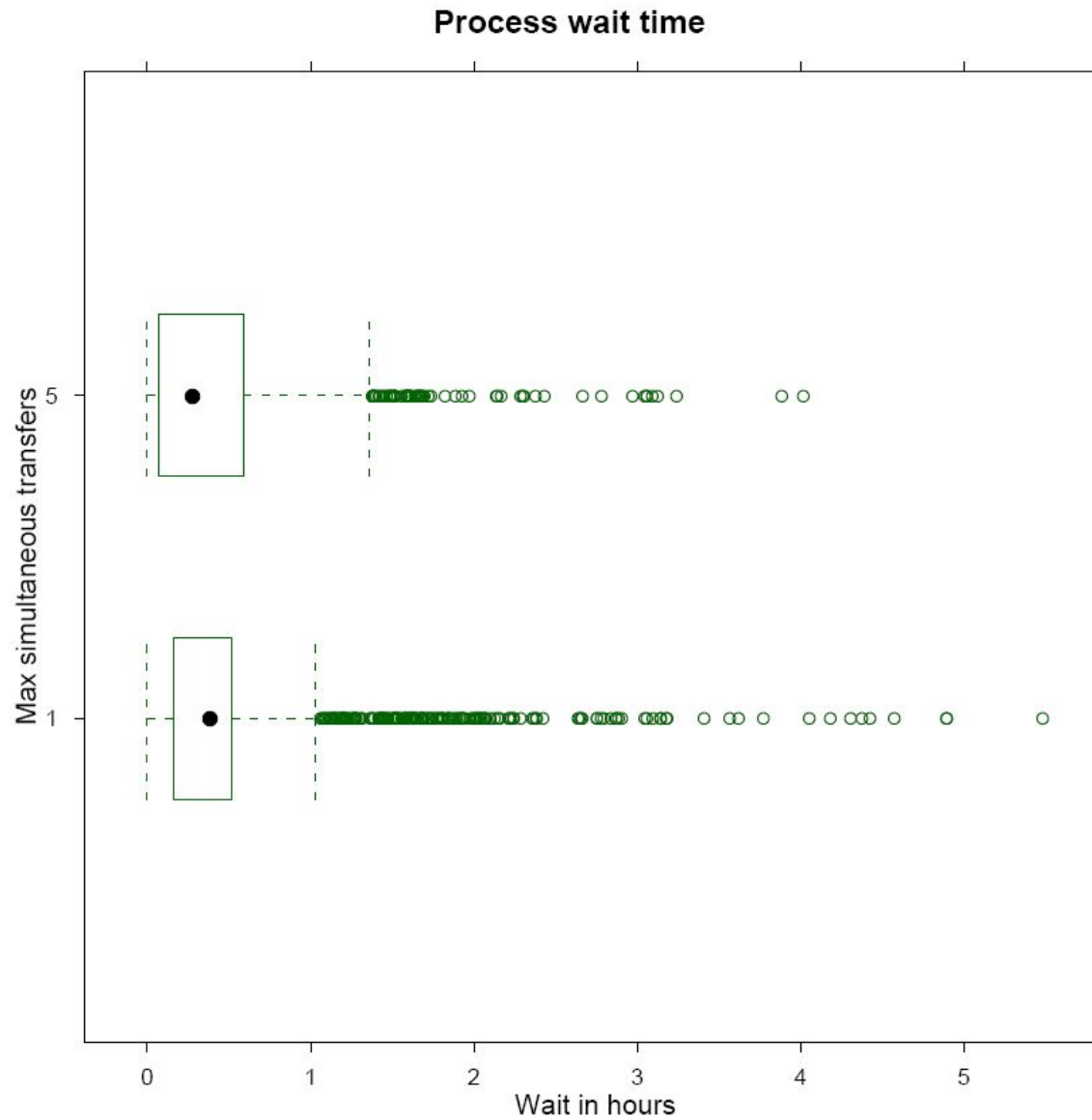
- Track number running throughout test
- With 5 Transfers, 8 hours mean
- With 1 Transfer, 10 hours mean



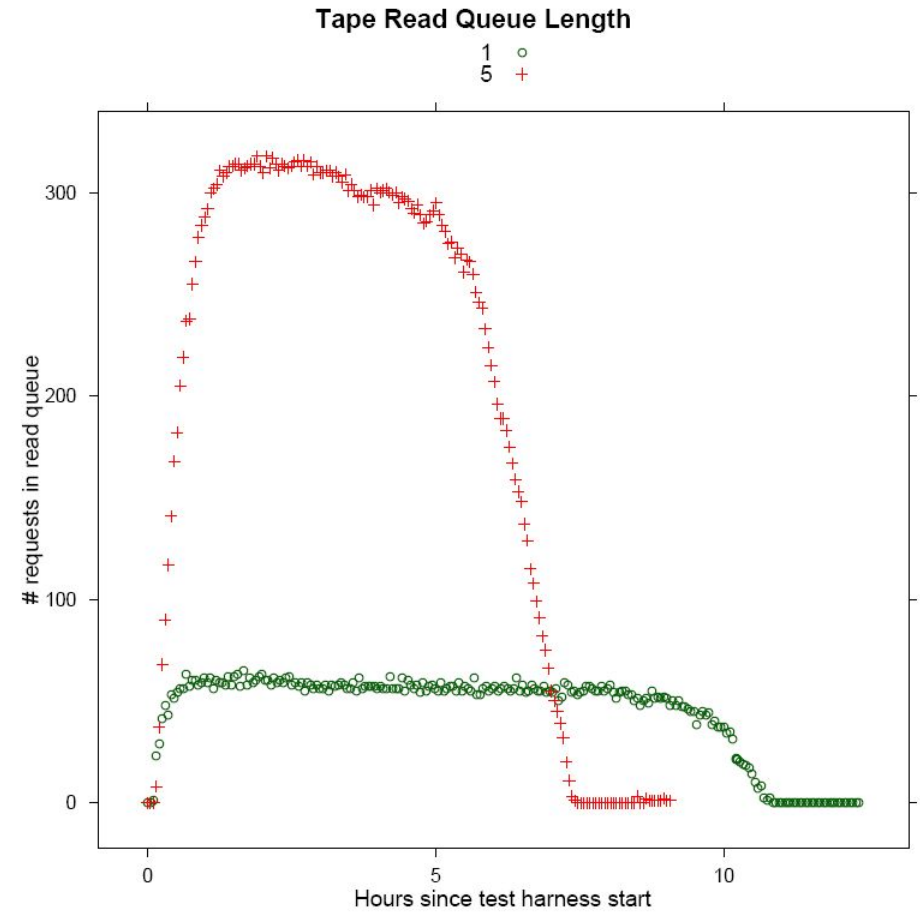
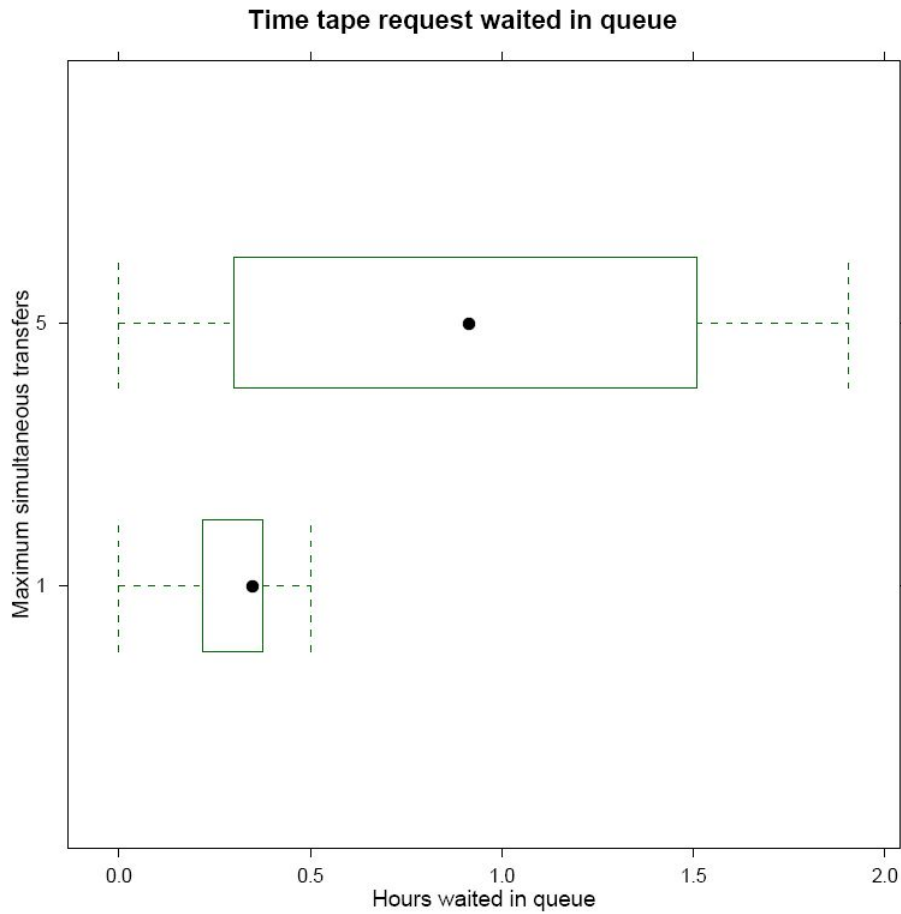
Project Duration



Process Wait Time



What is going on?



Conclusions

- Harness allows for Unit and Stress Testing
- In use at Fermilab
- Has proved useful for performance tuning
- Helpful to day to day operation of a Grid