

Sharing Common Code in ctest based testing

A. Dotti for the Physics Validation Task Force
Parallel Session 1B – Physics Validation Tools

Note

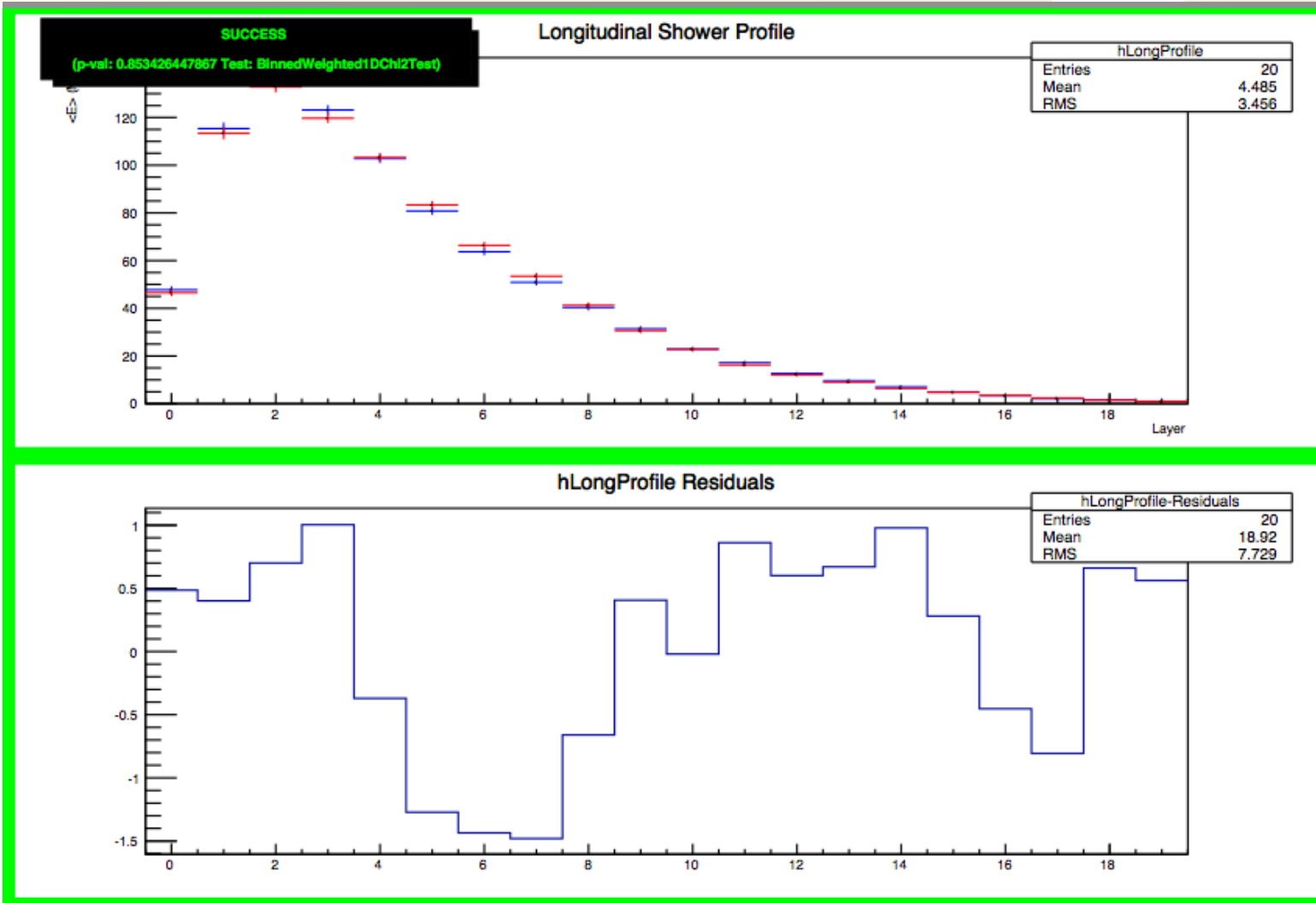
- This presentation contains similar concepts as the one in Parallel Session 2B – Hadronic Testing later in the afternoon
- If you all plan to participate to session 2B, tell me, I will go fast in these slides

- This package provide basic histogramming and ntuple functionalities
- It is ported to MT and handles merging of histograms for you (!)
- Provides output to ROOT format transparently (see next slide why this is useful)
- We should all make an effort to develop tests based on this package

Comparison to reference

- **StatTest** package is Geant4 add-on that uses ROOT to compare the output file (in ROOT format) of a test and its output to a reference file
- Needs a simple text file for configuration:
 - Which histograms / tree variables to check
 - Which statistical test run (several provided), typically use p-value, limits for “OK”, “BAD”, “NotSure”
- Produces graphical output

Example



- Big improvement in testing system since we introduced CTest based nightly system
- In 2012 has been extended to have a PhysicsChecks group (only 12 tests up to now, small physics coverage)
- There are few issues (access to plots, handling of references), but I think we can solve these with some new developments (see Session 2B)

Add testing

- The use of StatAccep test is general:
 - An additional ctest is added AFTER the MC simulation as a separate step
 - To add a new test, it is a matter of producing the test configuration file (very easy in simple cases, believe me!)
 - Package comes with an example, can be downloaded from SVN:
g4tests/verification/StatTest
 - Requires ROOT and Python
- True strength is that is integrated with CTest/CDash
 - A single line in CMakeList.txt

FIND_PACKAGE(StatTest)

```
IF(STATTEST_FOUND)
```

```
    STATTEST_ADD_TEST( aName
```

```
        G4TEST testProducingOutput
```

```
        CONFIG configurationFileName
```

```
        INPUT fileToCheck
```

```
        REFERENCE referenceFile
```

```
        IMG fileNameOfGraphicReport
```

```
    )
```

```
ENDIF()
```


Issues

- I believe this tool is not used widely for two reasons:
 1. Access of output files in ctest/cdash
 2. Reference handling
- See session 2B on possible solution to these problems

MT Physics Checking

- With MT we need to start thinking of physics validation of MT
- Current strategy:
 - Compare MT results with equivalent SEQUENTIAL code
- Tests done so far:
 - Strong Reproducibility
 - Test w.r.t. sequential
 - Test w.r.t. 1 thread

MT Testing: strong reproducibility

- Currently done by hand
- Start MT job for SimplifiedCalorimeter, save RNG status at beginning and end of events
- Start SEQ job for same application, re-seed each event check RNG status at the end of event

- Can this be automatized somehow?

MT Testing: physics output

- Run MT SimplifiedCalorimeter and use StatTest to compare statistically with SEQ version
- Again, can we automatize this?
- Finally, we need another independent application that tests also low-E models