

# New Beamtest Software

L.Baldini - J. Bregeon

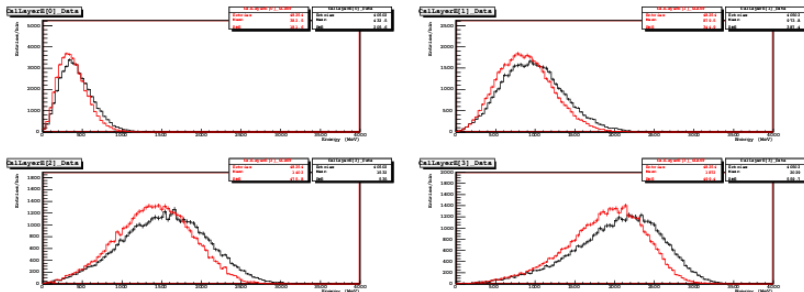
8th August 2007

- CUTower
  - Committed in beamtest06/src/CUTower
  - Standalone GEANT4 simulation with one LAT tower geometry
  - Works with GEANT4 v8 and v9
  - Geometry,cuts and physics lists can be changed from the macro
  - refences people : Johan, Francesco, Carmelo
- BtDataBrowser
  - Committed in beamtest06/python/BtDataBrowser
  - python kind of mini-database for beamtest runs
  - scripts to select runs that satisfy a wide range of criteria
  - scripts to generate standard set of jobOptions to be used in the pipeline for the selected runs
  - reference people: Luca B., Johan
- BtSysTest
  - Committed in beamtest06/python/BtSysTest
  - see following slides...

# BtSysTest - What is it meant for?

- Produce standard plots each time a new mc dataset is generated
  - can be run automatically on a (potentially long) list of runs
  - the list of canvas/plots to be produced can be easily configured
  - the histogram parameters (i.e. ranges) are automatically set, based on the run configuration information
- Compare data with mc (or data with data, or mc with mc, or data with several different mcs...)
  - you essentially pass to the program a run number and one or more labels identifying the data sets
  - paths to the data files are hard-coded in the program, but...
  - ...it runs @SLAC—you can run it on Noric
  - ...it can be easily hacked to run locally, if needed
- Improvements and new features can be added depending on how much people is interested into

# BtSysTest - How does it look like?



- Three different outputs (ROOT, eps, text)
- Histograms of residuals can be automatically produced
- And you get other nice features for free:
  - colours are automatically set
  - stat boxes are automatically moved (wow!)
- All available reports are on [/nfs/farm/g/glast/u37/MC-tasks/BtSysTest](https://nfs/farm/g/glast/u37/MC-tasks/BtSysTest)

# BtSysTest - How do I run it?

- Check it out from cvs...
  - it lives in beamtest06/python
  - there's a single executable called `BtSysTest.py`
  - type  
`BtSysTest.py -h`  
and you'll get all the supported command line options
- ... or drop us (Luca B., Johan, Carmelo) an e-mail!
  - feel free to ask directions or new features