RelMon: A General Approach to QA, Validation and Physics Analysis through Comparison of Histograms Sets

Validation of the Physics Performance of a HEP Experiment

Several types of changes are regularly introduced in the code and in the tools that lead to the delivery of HEP software. For example **algorithmic improvements** in object reconstruction, simulation or trigger, code performance optimizations, upgrade of the compiler or the operating system. The effect of these variations on the features of the physics objects used for data analysis must be continuously assessed:

a crucial aspect of the operations of a modern HEP experiment is the validation of the physics performance of its software

This goal can be achieved with a data quality monitoring infrastructure allowing to collect the relevant properties of each dataset in a collection of histograms and a tool to compare sets of histograms and estimate their level of compatibility.

RelMon

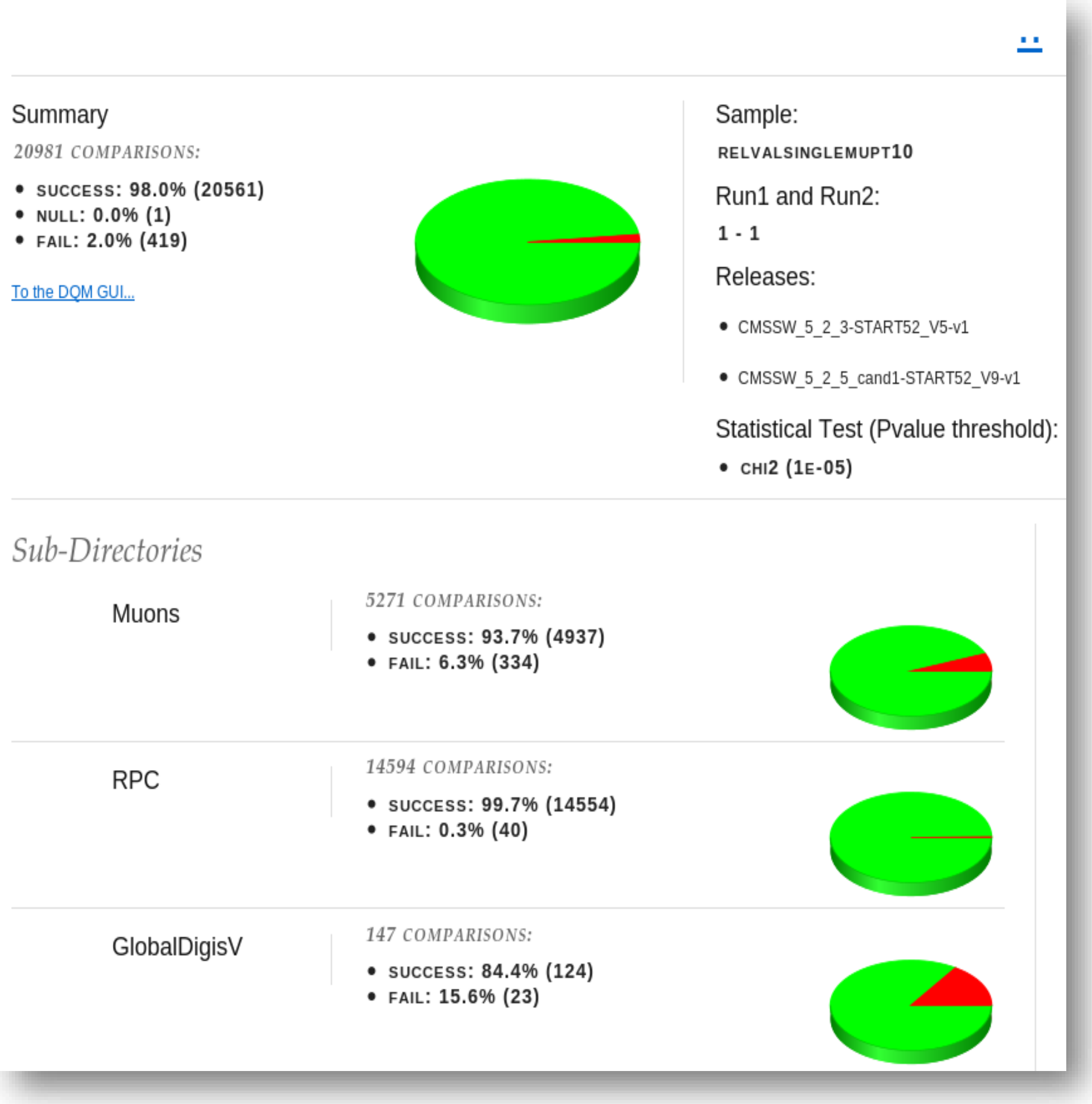
RelMon is a general tool to **compare large sets of histograms pairs according to a statistical test defined by the user.** It is implemented in about 4000 lines of Python and depends only on ROOT. A simple command-line interface is offered to operate the tool. The analysed histograms can be provided stored in ROOT files, also organised in a directory structure.

The default tests provided by RelMon to check the compatibility of two histograms are:

- Chi-squared
- Kolmogorov-Smirnov
- Bin-to-bin

The first two are well known statistical procedures and the tool relies on their ROOT implementation. The **bin-to-bin test is useful in case the identity** of two sets of histograms is to be checked. The content of corresponding bins of each of the two histograms is checked to be identical and the outcome of the test is the fraction of bins which are different.

All the information about the amount of succeeding, failing or null comparisons is aggregated per directory.



Summary

20981 COMPARISONS:

- SUCCESS: 98.0% (20561)
- NULL: 0.0% (1)
- FAIL: 2.0% (419)

To the DQM GUI...

Sample:

RELVALSINGLEMUPT10

Run1 and Run2:

1 - 1

Releases:

- CMSSW_5_2_3-START52_V5-v1
- CMSSW_5_2_5_cand1-START52_V9-v1

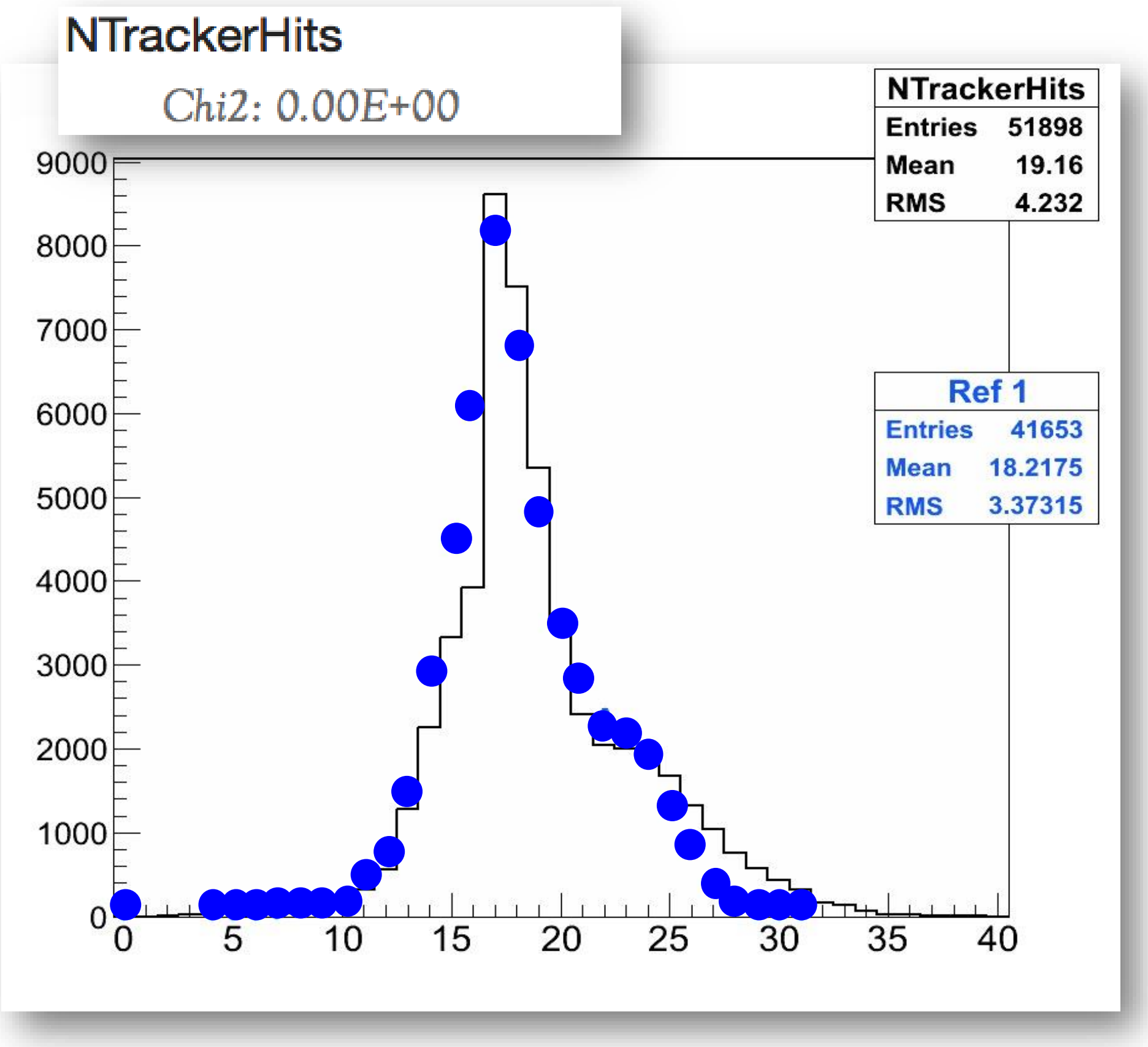
Statistical Test (Pvalue threshold):

- CHI2 (1E-05)

Sub-Directories

Muons	5271 COMPARISONS: • SUCCESS: 93.7% (4937) • FAIL: 6.3% (334)
RPC	14594 COMPARISONS: • SUCCESS: 99.7% (14554) • FAIL: 0.3% (40)
GlobalDigisV	147 COMPARISONS: • SUCCESS: 84.4% (124) • FAIL: 15.6% (23)

The result of the comparison of two sets of histograms can be displayed in two ways: either with a plain ASCII report printed on screen or with an **elegant web report where all the plots of the histograms overlays are displayed.**

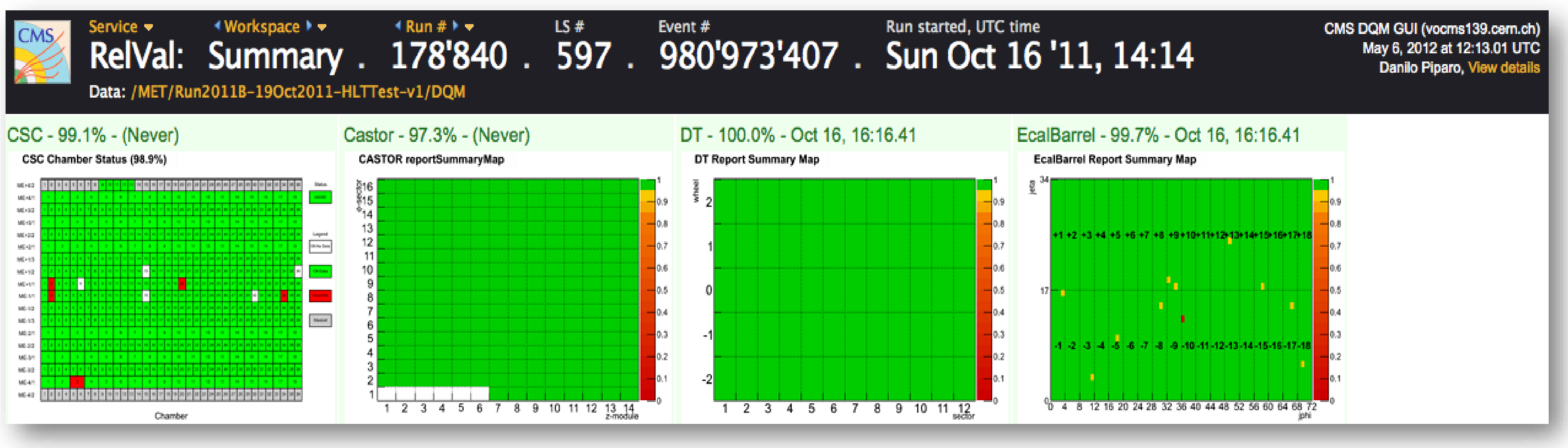


See also at CHEP

1) CMS reconstruction improvements for the tracking in large pile-up events - **Contrib. ID 240**
2) Development and Evaluation of Vectorised and Multi-Core Event Reconstruction Algorithms within the CMS Software Framework - **Contrib. ID 210**

CMS Data Quality Monitoring

One of the ideas behind the CMS DQM infrastructure is to **execute monitoring modules after the algorithms used for event reconstruction in the same process** in order to collect in histograms the relevant quantities for data certification. The CMS framework takes care of **writing such histograms in dedicated ROOT files.** These files are automatically uploaded to the DQM servers which further process their content and store it in an internal database for performance reasons. A web interface allows to browse the histograms in a given DQM server. **The plots are created on the fly with a special renderer. RelMon can rely on this advanced feature for the web reports** or create png images of histograms overlays.


Service
Workspace
Run #
LS #
Event #
Run started, UTC time
RelVal: Summary . 178'840 . 597 . 980'973'407 . Sun Oct 16 '11, 14:14
Data: /MET/Run2011B-19Oct2011-HLTTest-v1/DQM
CMS DQM GUI (vocms139.cern.ch)
May 6, 2012 at 12:13.01 UTC
Danilo Piparo, View details
CSC - 99.1% - (Never)
CSC Chamber Status (98.9%)
Chamber
Castor - 97.3% - (Never)
CASTOR reportSummaryMap
z-module
DT - 100.0% - Oct 16, 16:16.41
DT Report Summary Map
wheel
sector
EcalBarrel - 99.7% - Oct 16, 16:16.41
EcalBarrel Report Summary Map
ieta
iphi

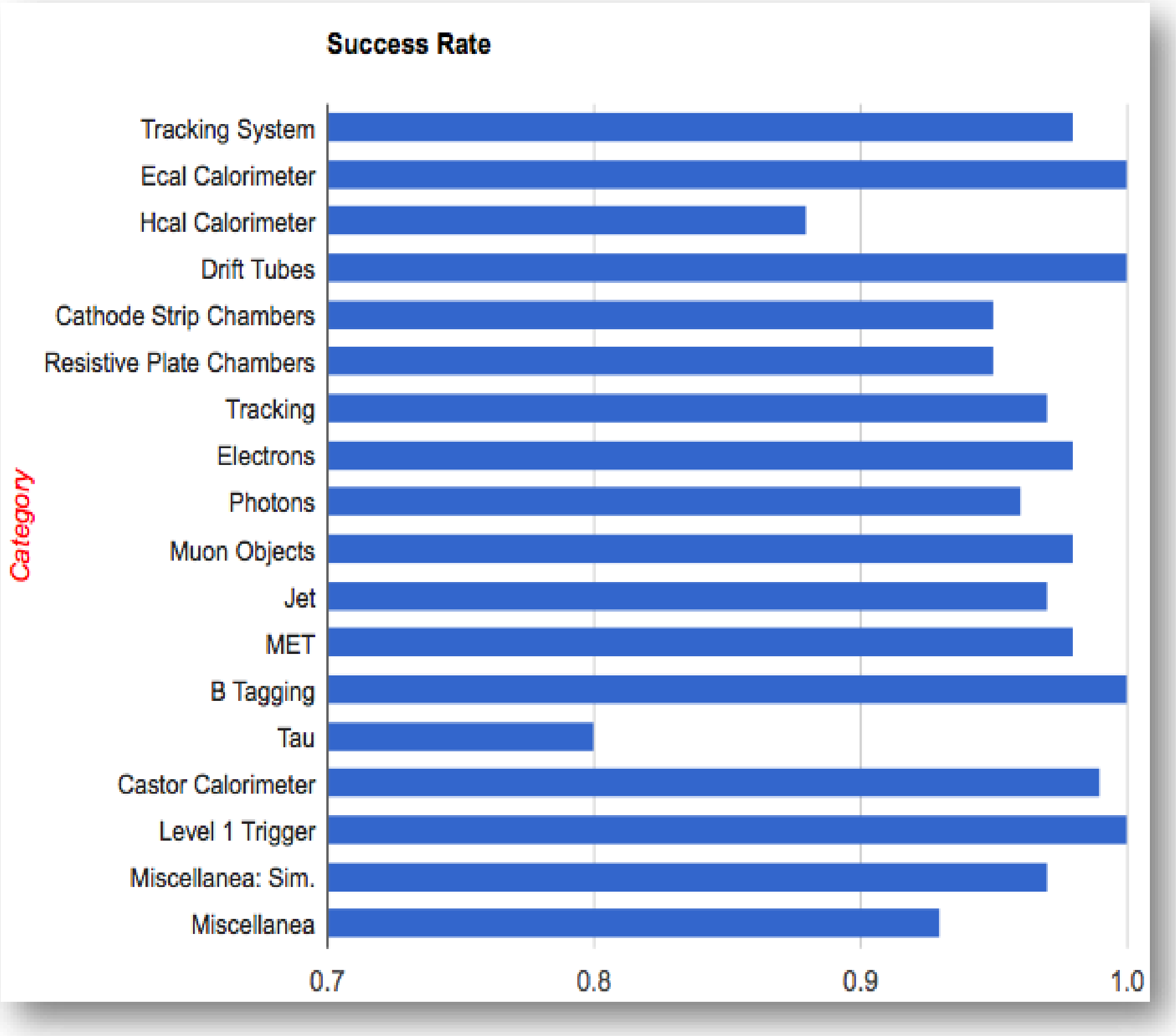
A Rich Set of Use Cases in CMS Already!

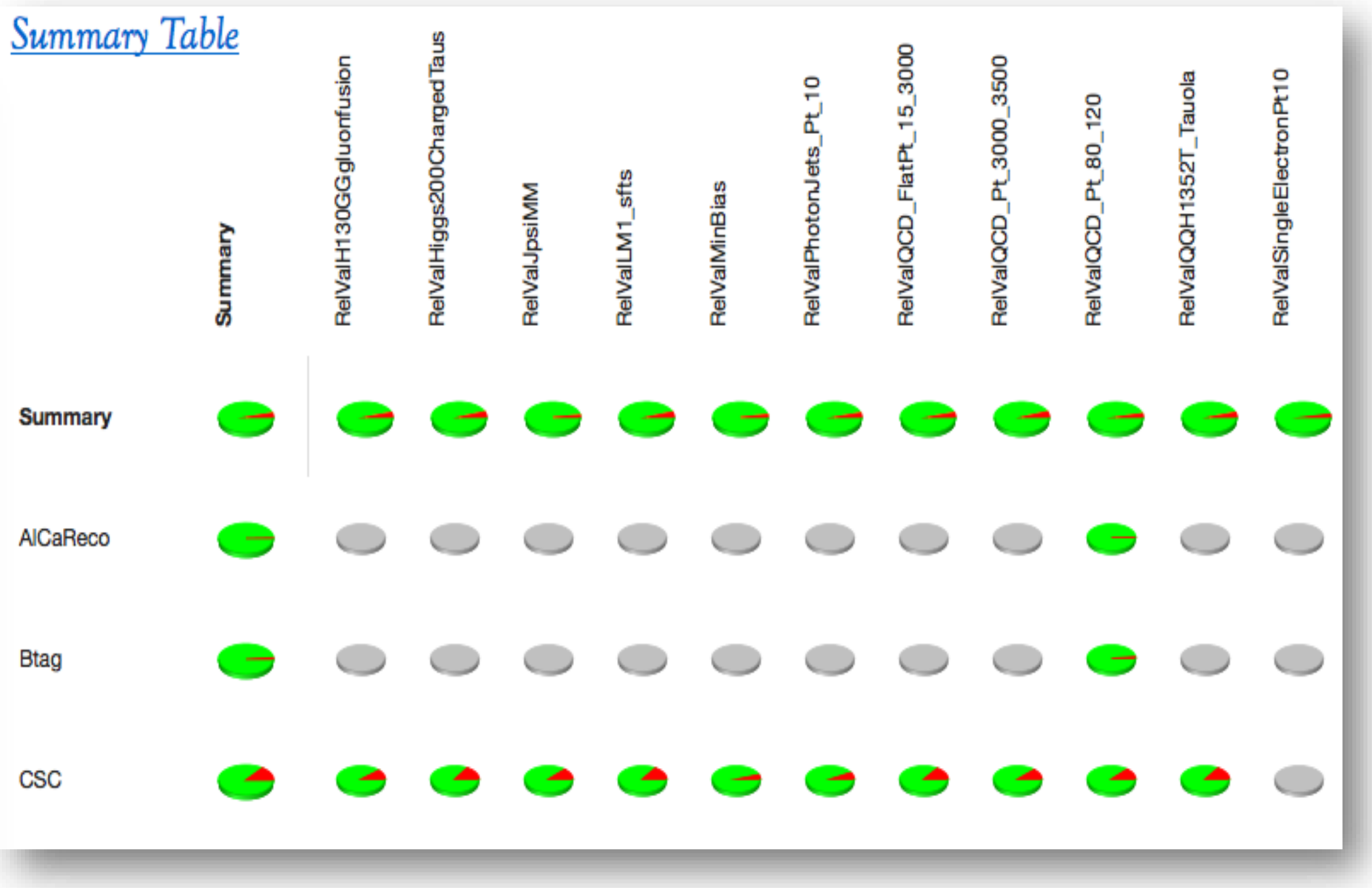
CMSSW Integration Builds

Twice a day, the code candidate to be part of an official CMSSW release is compiled on different platforms. This allows to perform important sanity checks but also to **process small datasets**. RelMon is executed to check the compatibility of the histograms obtained with a known reference. This procedure represents a **first line of defense against unexpected behaviors** of CMSSW or bugs introduced by the developers.

Reconstruction, Simulation, Calibration Constants and Trigger

CMSSW pre-releases are provided once a week in order to consolidate the state of the code, test interdependencies among software components while releases are cut approximately once per month to close a development cycle. **For every (pre)release, a complete set of release validation datasets of Monte Carlo simulated and re-processed real collisions events is provided.**





Experts of all CMS sub-detectors, physics objects and analyses scrutinize these datasets against the ones of the preceding release to obtain an **incremental validation** of CMSSW. **The main tool used to perform these regressions is RelMon. For each (pre)release regression is centrally provided,** with more than one million histograms compared. With this strategy, anomalies can be immediately pin-pointed.

Software Performance Optimizations

The CMS collaboration puts an enormous effort in the **continuous improvement of the CPU performance of its software without affecting the quality of the detector's measurements.** This process implies a precise scrutiny of the results obtained with the improved software. **RelMon can help in this process**, with the bin-to-bin test. CMS took advantage of this feature in several occasions, for example during the transition from the ***GCC*** compiler from the 4.3 to the 4.6 version or when the ***jemalloc*** memory allocator was adopted.

User Analysis

One weakness of data analysis code is that its robustness is not always thoroughly checked, mainly because of the tight schedules involved in discovery analyses. An analysis group can take advantage of RelMon in this context, checking the compatibility of the results produced on a known dataset before and after the new developments. In addition, RelMon makes the evaluation of the Monte Carlo description of data straightforward.

Danilo Piparo CERN • May 24th 2012 • Computing in High Energy and Nuclear Physics 2012 • New York • US • Contrib. ID 211