The CMS Phase 2 Outer Tracker Analyzer of Test Outputs - POTATO!

Sweta Baradia[a,b], J. Dickinson[c], A. Klupshas[d], L. Lee[e], P. Merkel[c], R. Ogaz[e], C . Riggal[e], M. Sizemore[e], L. Uplegger[c], I. Zoi[c]

a. Saha Institute of Nuclear Physics, 1/AF Bidhan Nagar, Kolkata 700064, India
b. Homi Bhabha National Institute, Training School Complex, Anushaktinagar, Mumbai 400094, India
c. Fermi National Accelerator Laboratory, Kirk and, Pine St, Batavia, IL 60510
d. University of Tennessee, Knoxville, TN 37996
e. Purdue University
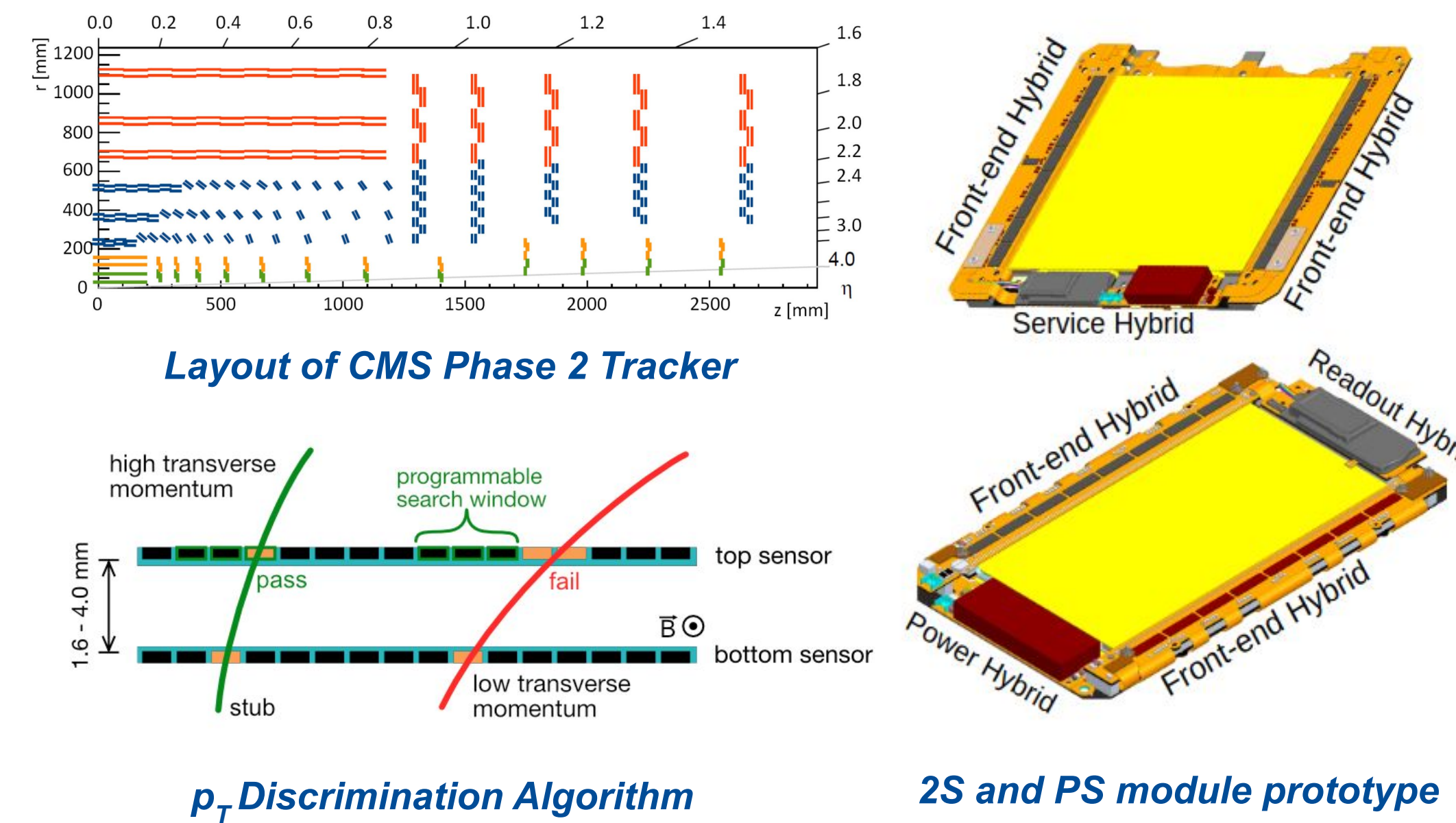
57th Annual Users Meeting (2024)

CMS and Phase 2 Upgrade

- Compact Muon Solenoid (CMS) is a gigantic particle detector at the Large Hadron Collider (LHC)

- Tracker is the innermost sub-detector
  - Made of silicon sensors - Pixels and Microstrips
- High Luminosity LHC (HL-LHC) is expected to operate at peak instantaneous luminosity of $5\text{-}7.5 \times 10^{34}$ cm^{-2}s^{-1} !
  - Deliver a total integrated luminosity of at least 3000 fb^{-1}
  - ~200 pileup per event - requires full detector upgrade (Phase 2 Upgrade)
  - A completely new tracking system needed for HL-LHC to sustain the challenging environment - Inner Tracker (IT) and Outer Tracker (OT)
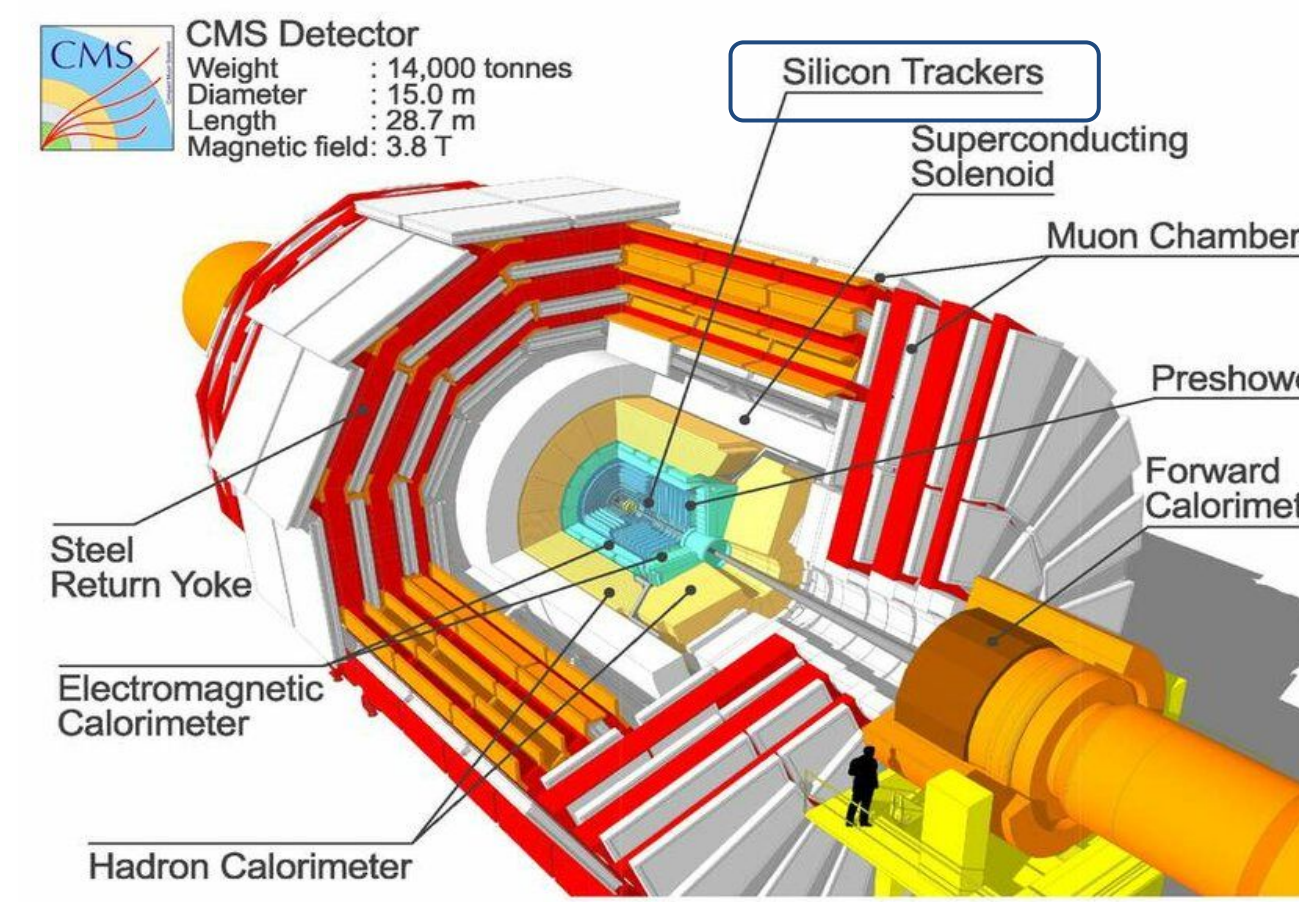
Why do we need a new Outer Tracker ?

- For the very first time, the OT will send tracking information to the CMS Level 1 trigger to control the data rates
  - OT will be made up of modules with two closely spaced sensors (p$_T$ modules)
    - Strip-Strip (2S) and Pixel-Strip (PS)
  - Accept track with p$_T$ beyond a set threshold of 2 GeV

Layout of CMS Phase 2 Tracker

p$_T$ Discrimination Algorithm

2S and PS module prototype

Module Testing using Ph2_ACF

- Module is passed through a measurement routine - **Phase II Acquisition and Control Framework (Ph2_ACF)**
- Ph2_ACF is a framework written in C++
  - Communicates with the module via Data, Trigger and Control (DTC) equipped with FPGA
  - Implements all calibrations and optimization procedures and dumps the results in a ROOT file
- Output contain quantities which are important to quantify the quality status of the module (Noise, Pedestal, Occupancy etc.)

Assembled 2S (left) and PS (right) module at Fermilab
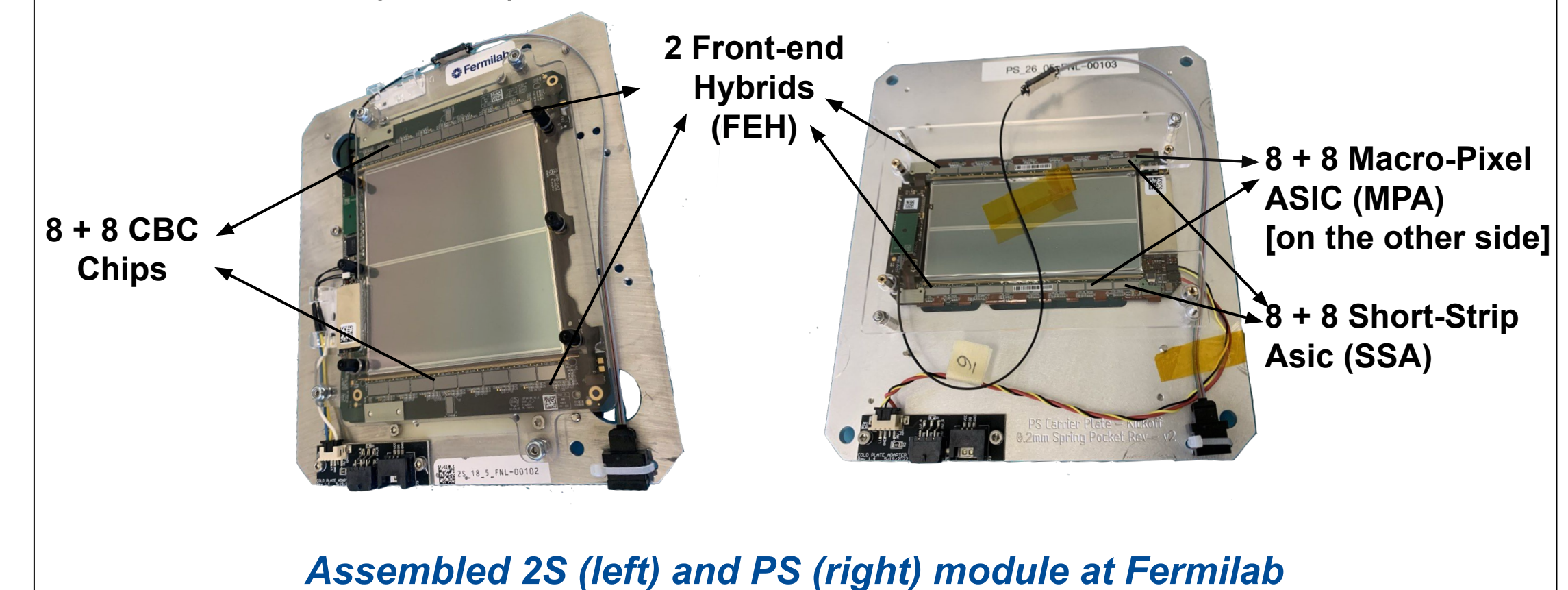
Phase 2 Outer Tracker Analyzer of Test Outputs (POTATO!)

Module Test Setup

Noise per Channel in a FEH (left) , Noise per Channel (middle) and SCuve in a FEH CBC 0 (right)

Phase 2 Outer Tracker Analyzer of Test Outputs

Motivation

- Over 13,000 modules (2S and PS), will replace the current strip tracker!
- Module production is distributed across centers worldwide
  - Necessitates coordinated efforts and standardized procedures
  - Preparations for production, testing and quality assessment ongoing
- We are working on a tool called POTATO, that will **analyze, grade, upload** and manage the large quantity of files to be stored in the Database (DB)
  - Uses C++, QT and ROOT enabling real-time analysis of test outputs, providing immediate feedback to address issues quickly
  - **Root Files from the module tests are taken as input**

POTATO GUI Utility and Workflow

Analyzer and Grader   Root File Upload to DB   Analyzer and Grader Results Upload to DB

Module test files selector

Module name, sensor spacing and serial number

- Designed to simplify **analyzing, grading (A,B,C…), and uploading** of the test files

Analysis and Grading in POTATO

- Analysis done on significant quantities present in input file such as noise, pedestal etc. for chips, hybrids and module as a whole
- Grading criteria on mean noise, number of unusable channels etc. are established and applied on the analysis results
- Significant parameters such as test temperature, versions also uploaded to DB along with analysis and grading results

Noise Distribution in a FEH
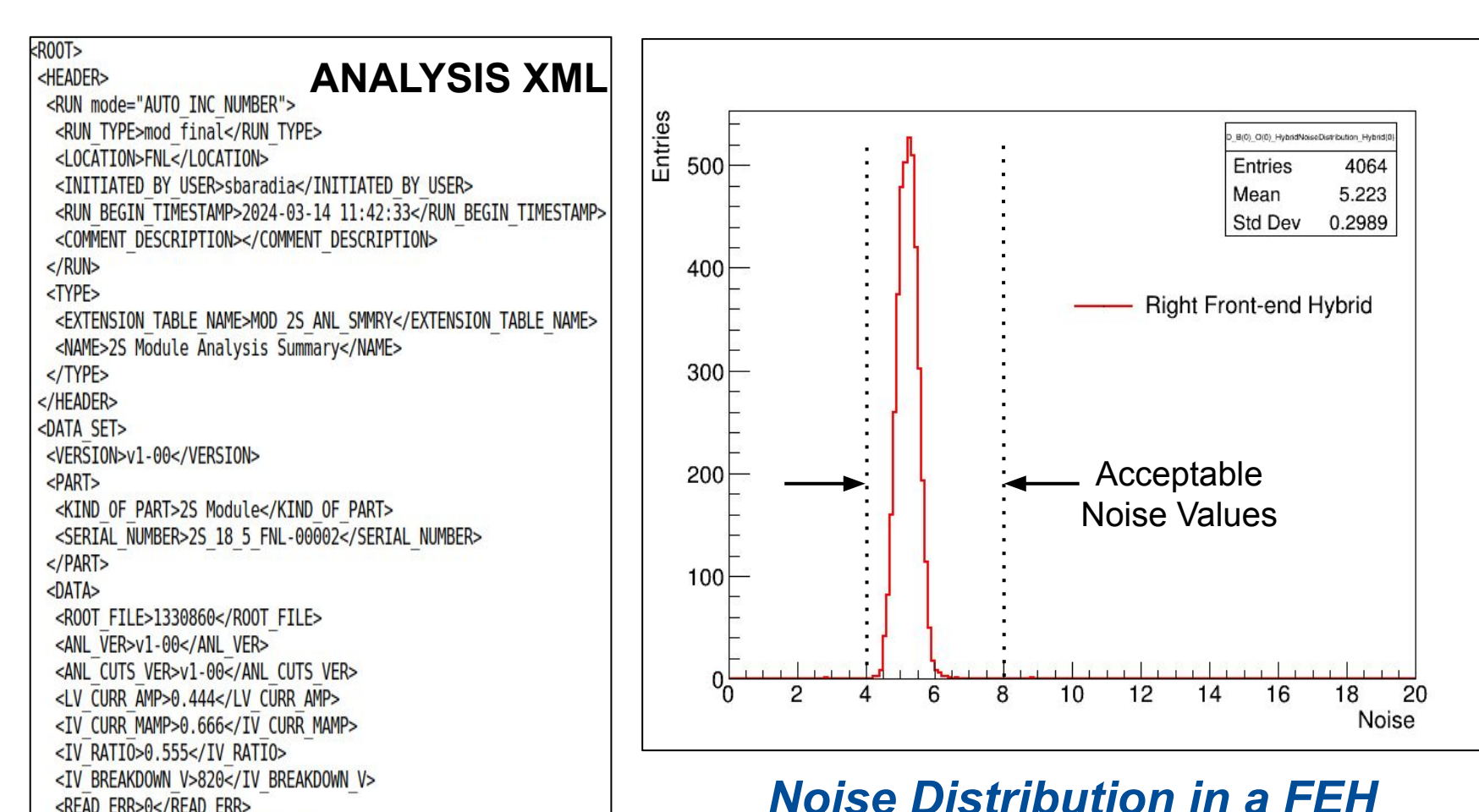
Plugins and Other Salient Features

- POTATO offers the possibility of writing plugin scripts which can be shared among the community
  - Also helps for cross-checking the analysis and grading when required
- Plugins developed for visualizing quantities for single and/or multiple files

- Can read-off quantities corresponding to a given uploaded file from the DB using SQL queries
- GUI-less test file processing can also be done with this tool - POTATO Express
 - Just compile and run!
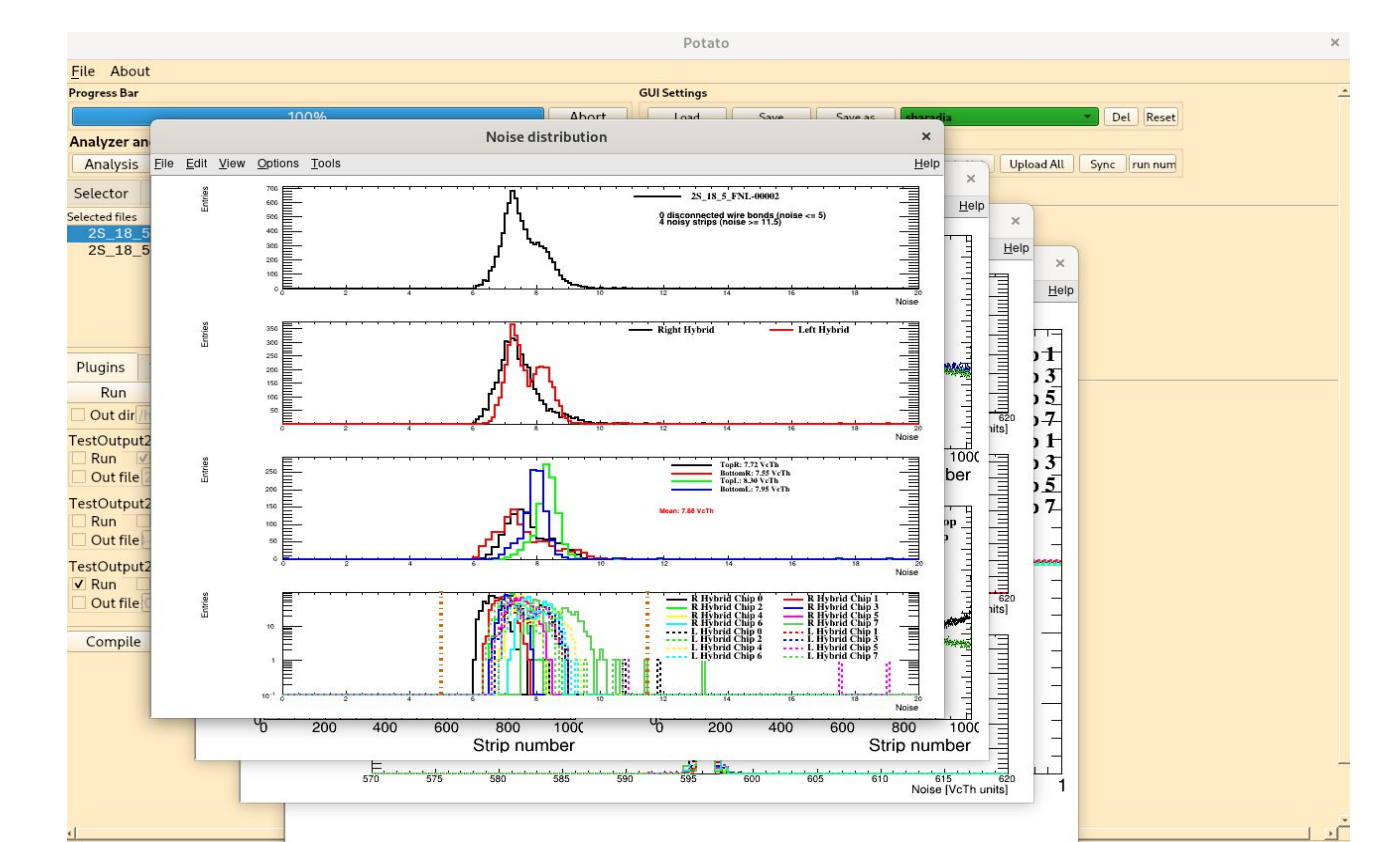
Summary

- POTATO is a GUI enabled tool which allows users to interact with the Database
- It is crucial for handling the large number of module test outputs
- It allows occasional reprocessing of files
- It is easily adaptable to new calibrations and grading schemes
- It will provide consistency in analysis results and quality controls during module production
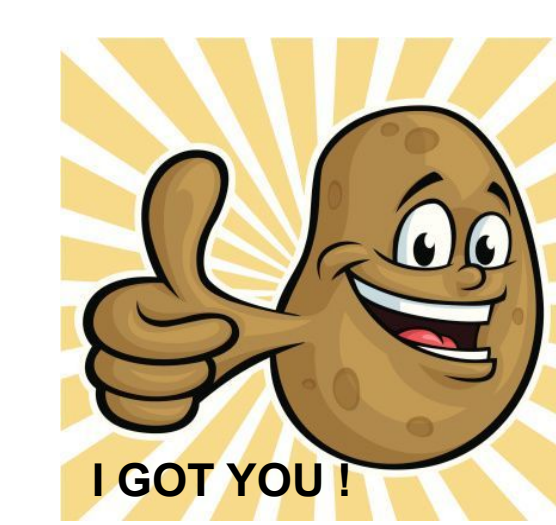
References

- The CMS Experiment at the CERN LHC10.1088/1748-0221/3/08/S08004
- CMS Collaboration, "The Phase-2 Upgrade of the CMS Tracker", CMS-TDR-014
- CMS Ph2 ACF (Acquisition & Control Framework)

Contact: sweta.baradia@cern.ch

I GOT YOU !