tkLayout

A Design Tool for Innovative Detectors

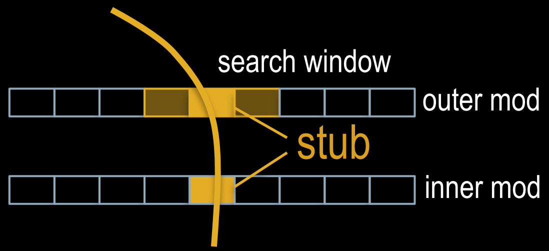
G. Bianchi (CERN) – On behalf of the CMS Collaboration – IPRD13 Siena

Outline

- Towards HL-LHC
- The tkLayout tool
  - Architecture
  - Geometry model
  - Material model
- Performance metrics
  - Tracking performance analysis
  - Stub frequency analysis
- Case Studies
  - 7 disks vs. 5 disks Endcap geometries
  - Trigger Tower architecture
- Going beyond CMS?
- Conclusions

CMS Tracker for HL-LHC

- New design to cope with HL-LHC
- Instrumented with Level-1 triggering capabilities
- Dual sensor modules
  - Measure pT locally
    (tracks bend in CMS magnetic field)
  - Weed out hits from low pT tracks
- Array of trigger processors (Trigger Towers) to process trigger data from different parts of the tracker – Divide et Impera



tkLayout

- Standalone tool to model CMS tracker layouts
- 5 years in the development (originally by S. Mersi)
  - Started off as simple calculation tool
  - Many features added along the way
- Used to help in the study of the enormous phase space of Phase 2 Tracker design
  - Many possible architectures tested and compared
  - Candidate layout being optimized iteratively
- Recently rewritten major portion of code (now C++11)
  - Improved performance
  - Improved flexibility
  - Improved usability

tkLayout

- 3d model of the tracker from simple configuration files
- Material model
- Runs analyses:

Tracking performance

Power dissipation after irradiation

Stub rate

Multiple trigger processors

…

- Exports geometry to “CMSSW” Simulation framework for full simulations
- Outputs figures and plots as a mini-website, for ease of consultation

tkLayout

3xPS_3x2S_5disks_longer

layouts

geometry | geometry (pixel) | bandwidth | trigger cpus | power | material (outer) | material (pixel) | weights (outer) | resolution | resolution (trigger) | trigger info | log page

layers and disks

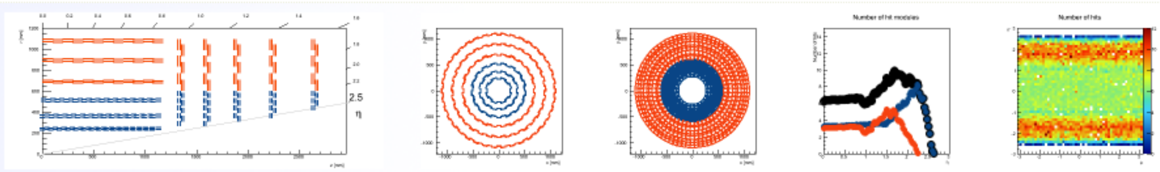
Layer	L1	L2	L3	L1	L2	L3	Total
r	230	357	508	686	888	1080	
# mod	1008	1320	1836	1152	1488	1824	8628
# rods	16	24	34	48	62	76	

Disk	D1	D2	D3	D4	D5	Total
z	1349	1597	1891	2239	2650	
# mod	688	688	664	640	580	6520

Ring	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
r_{min}	263	310	340	387	414	462	484	533	552	602	669	774	835	941	994
r_{max}	309	356	386	434	460	508	531	579	599	702	770	874	936	1041	1095

modules

plots



tkLayout vs. full simulation

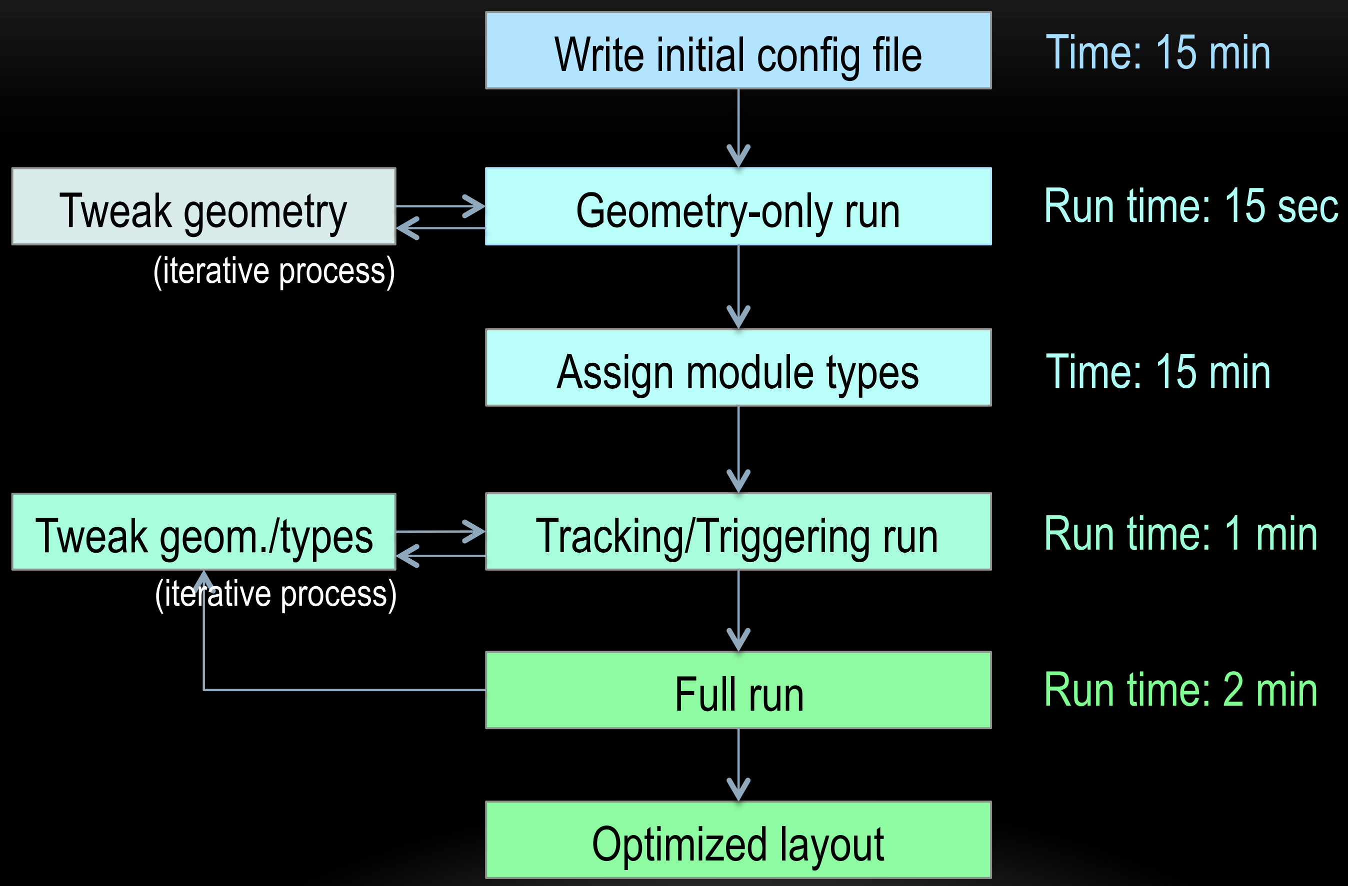
tkLayout

- Does calculations too long and tedious to do by hand
- Automatically defines the geometry features (i.e. supports), by making assumptions on the tracker structure
- Uses distributions of track properties
- Is set up in 30 min
- Runs in ≤ 2 min (on a pc)
- Is ideal for experimenting with different layouts and to make iterative optimizations on a candidate layout
- Is a design tool

Full simulation

- Does a full-scale Monte Carlo simulation
- Forces the user to explicitly define all the geometry features
- Does an event-by-event simulation
- Needs <u>weeks</u> to set up
- Runs in <u>hours</u> (on <u>grid</u>)
- Is best used for analysis once a good candidate layout has been chosen and optimized
- Is a physics analysis tool

tkLayout workflow



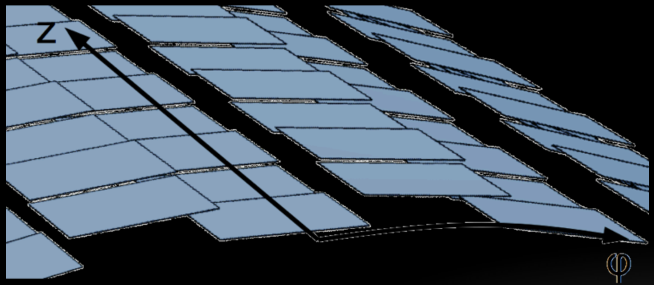
Configuration File Example

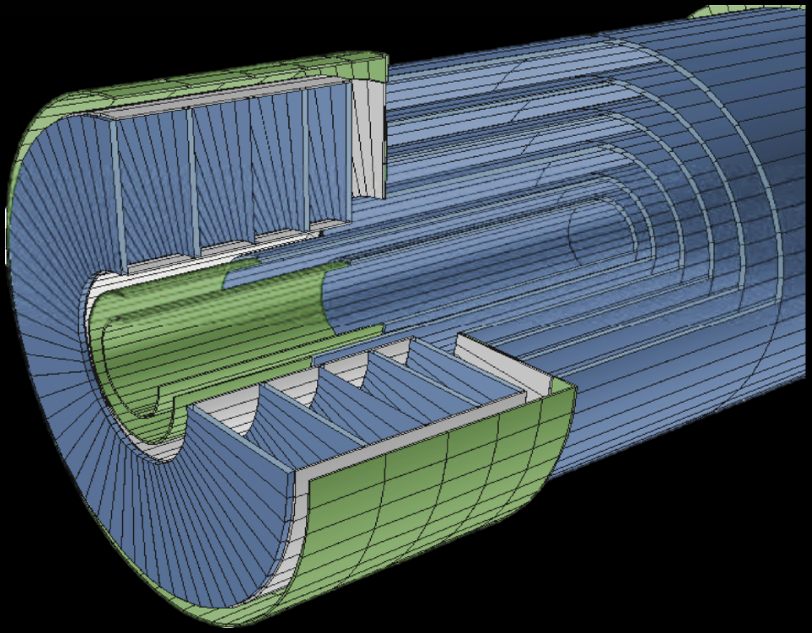
```
Tracker Example {
    smallDelta 10
    bigDelta 16
    Barrel bar {
        numLayers 6
        maxZ 1100
        innerRadius 200
        outerRadius 1600
        Layer 1-3 { moduleType ptMixed }
        Layer 4-6 { moduleType ptOut }
    }
    Endcap end {
        numDisks 5
        numRings 15
        outerRadius 1600
        barrelGap 100
        Ring 1-5 { moduleType ptMixed }
        Ring 6-15 { moduleType ptOut }
    }
}
```

- Simple (not complete) example of a config file
- Very quick and intuitive to write
- Time to write from scratch = 10-15 min

Geometry

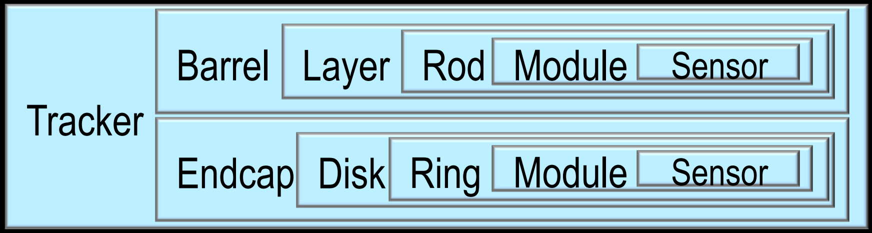
- tkLayout models:
  - Active elements (modules)
  - Supports
  - Services
- Modules automatically placed so that the tracker is hermetic





Geometry

- Geometry is a hierarchy of objects



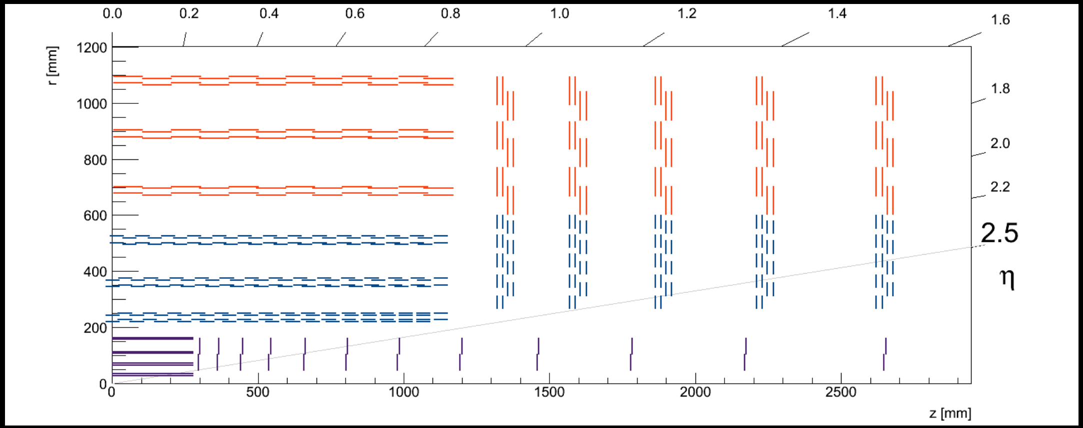
- Easily extensible or modifiable
  - Each object doesn’t need to know anything about its parent
  - Each object has properties, which automatically parsed from the config. files:

```
Property<double, Default> smallDelta("smallDelta", parsedOnly(), 10);
```

- Modules are the only objects with an internal 3d representation

Geometry

- tkLayout supports the definition of many types of geometries, among which the following have been considered and performance-evaluated with the tool

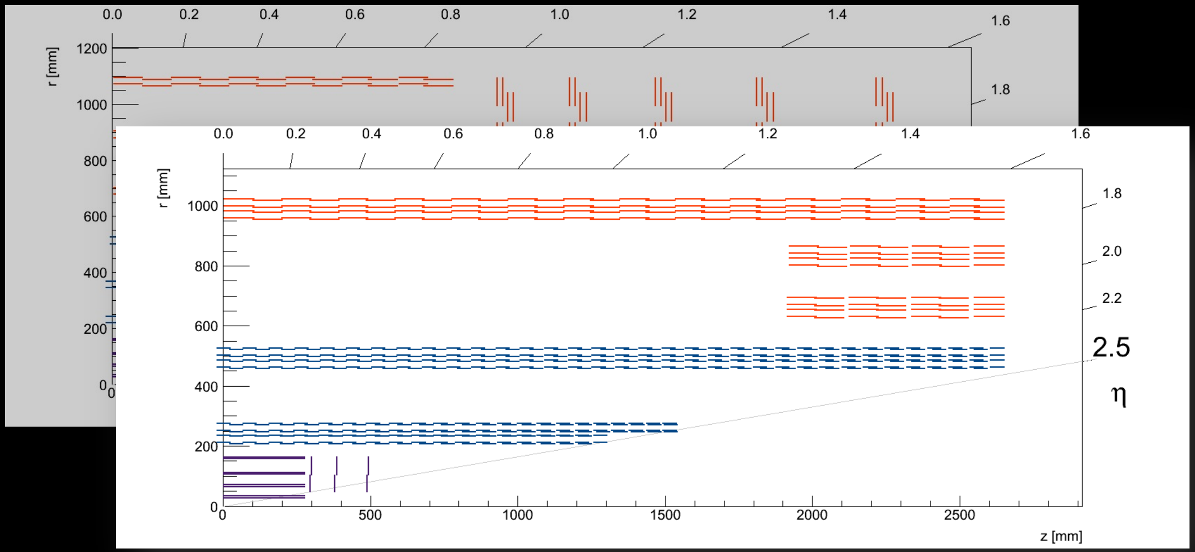


Barrel Endcap

- tkLayout has been instrumental in choosing the candidate geometry for the tracker upgrade

Geometry

- tkLayout supports the definition of many types of geometries, among which the following have been considered and performance-evaluated with the tool

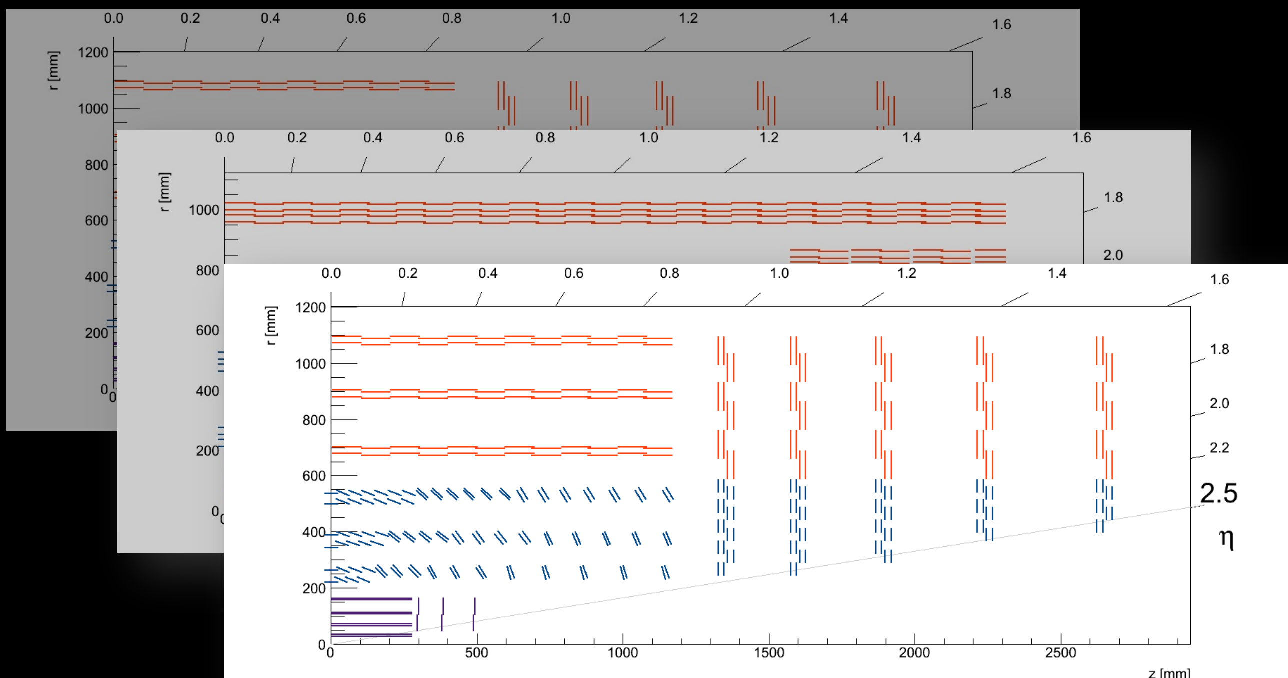


Long Barrel

- tkLayout has been instrumental in choosing the candidate geometry for the tracker upgrade

Geometry

- tkLayout supports the definition of many types of geometries, among which the following have been considered and performance evaluated with the tool



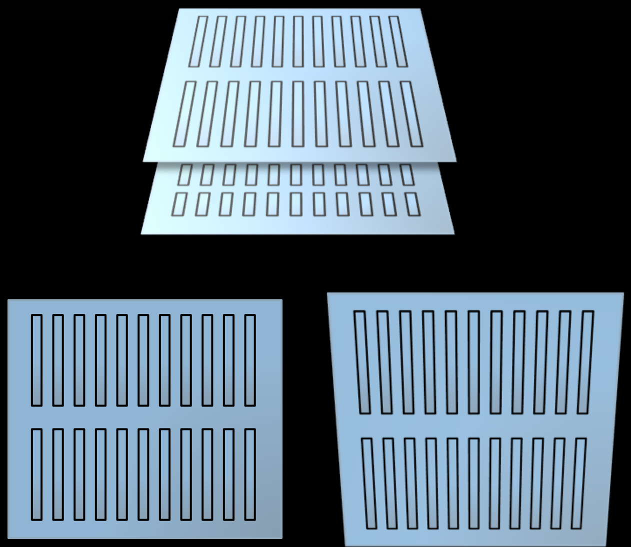
Tilted Barrel Endcap

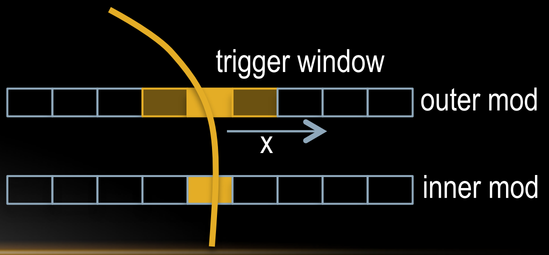
Experimental, still under study

- tkLayout has been instrumental in choosing the candidate geometry for the tracker upgrade

Modules & Sensors

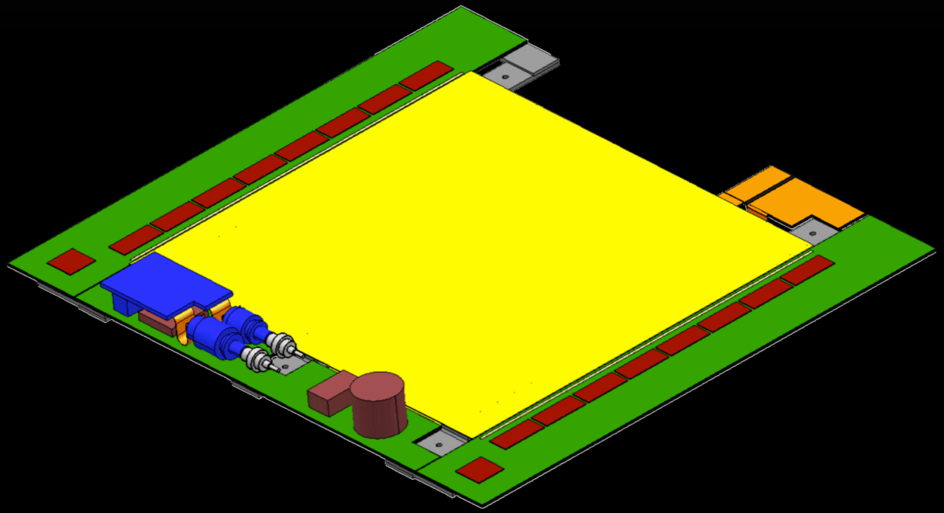
- Can have one, two (or even more) sensors
- Each sensor can have a different topology
  - Num. sensing elements along the beam dir.
  - Num. sensing elements across
  - Pitch between sensing elements
- Modules can be
  - Rectangular
  - Wedge-shaped
- pT-aware modules correlate hits on both sensors to obtain a measurement of the track Pt
  - Hits within a maximum *x* distance (trigger win.) form a “stub”
  - Stubs are sent off-detector for trigger decision





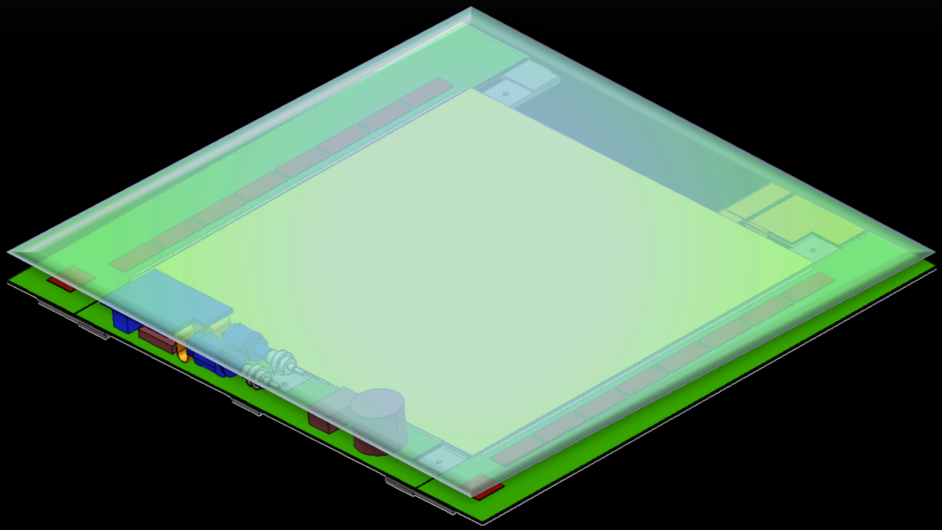
Material

- All the material is assigned to modules
- Can represent:
  - module components
  - services exiting the module
  - support structures



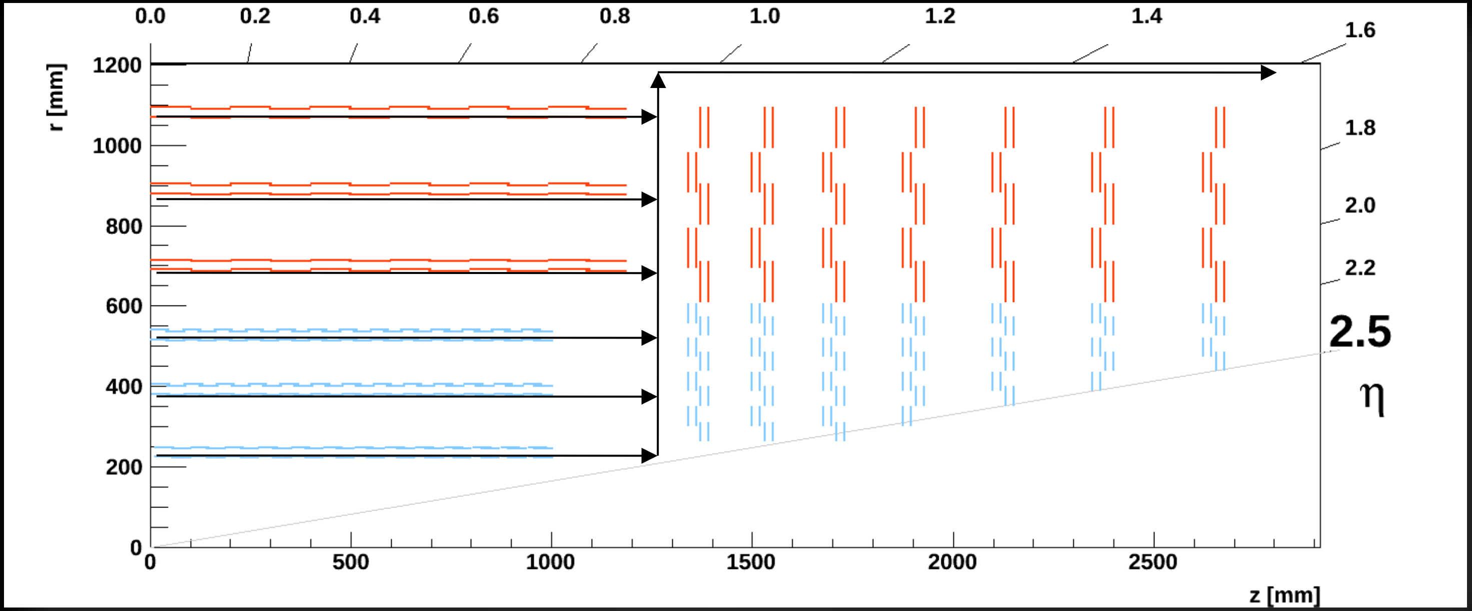
Material

- All the material is assigned to modules
- Can represent:
  - module components
  - services exiting the module
  - support structures
- Material is spread out over the surface of the module



Material

- Material for services is automatically routed



Outline

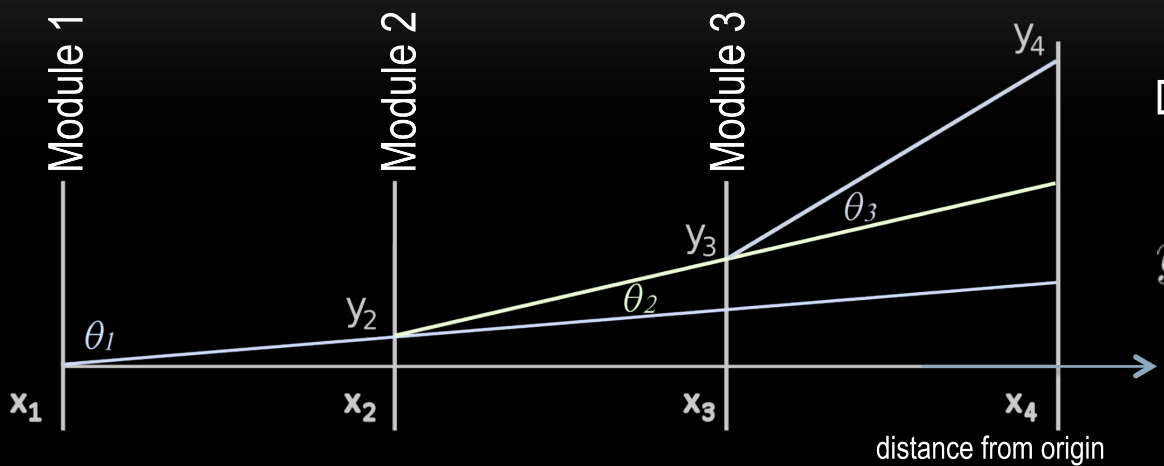
- Towards HL-LHC
- The tkLayout tool
  - Architecture
  - Geometry model
  - Material model
- Performance metrics
  - Tracking performance analysis
  - Stub rate analysis
- Case Studies
  - 7 disks vs. 5 disks Endcap geometries
  - Trigger Tower architecture
- Going beyond CMS?
- Conclusions

Tracking Performance Analysis

- A-priori error estimate:
  - No Monte Carlo
  - No fitting

- Ingredients:
  - Error propagation
  - Sensor resolution (measurement error)
  - Multiple scattering (treated as a correlated measurement error)

Tracking Performance Analysis



Deviation due to scattering:

$$y_n = \sum_{i=1}^{n-1} (x_n - x_i)\,\theta_i$$

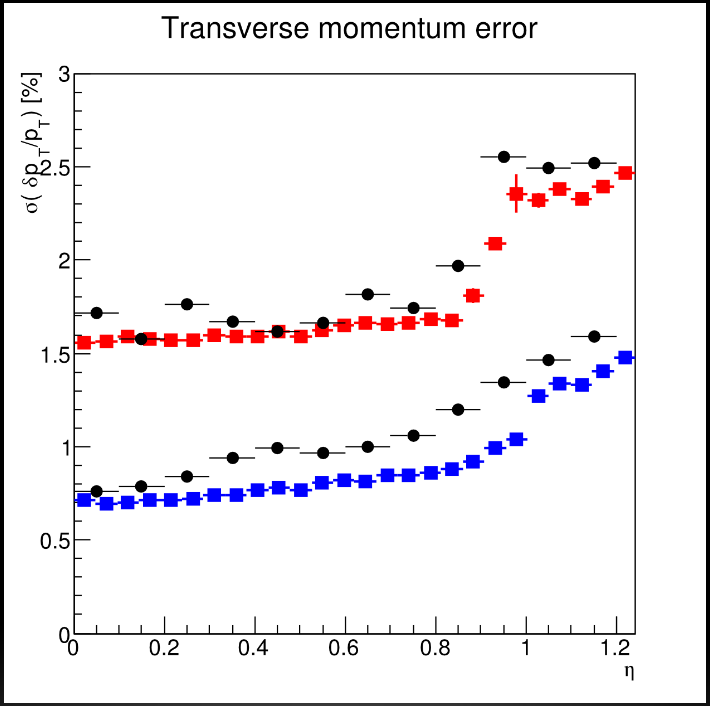
Covariance matrix of the measured hit coordinates:

$$\sigma_n^2 = \frac{p^2}{12}$$

$$\sigma_{n,m} = \langle y_n y_m \rangle = \sum_{i=1}^{n-1} (x_m - x_i)(x_n - x_i)\langle \theta_i^2 \rangle$$

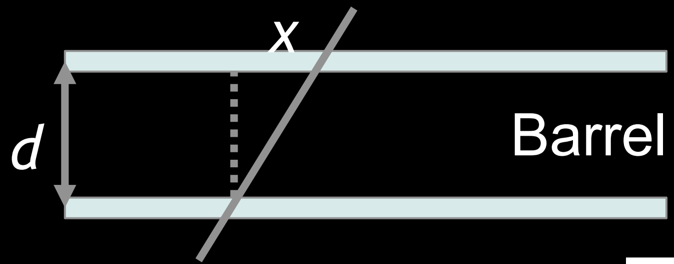
Tracking Performance Analysis

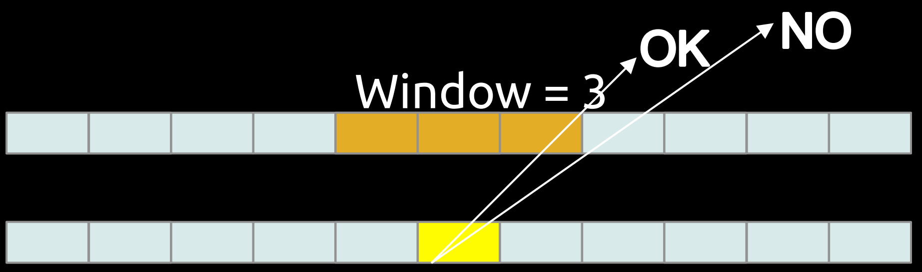
- Method validated by modelling the current CMS tracker and comparing the results with CMSSW Full Sim



Stub Rate Analysis

- For the tuning of sensor spacing and trigger windows
- Given spacing and window it is possible to compute the probability of a stub being found as a function of pT



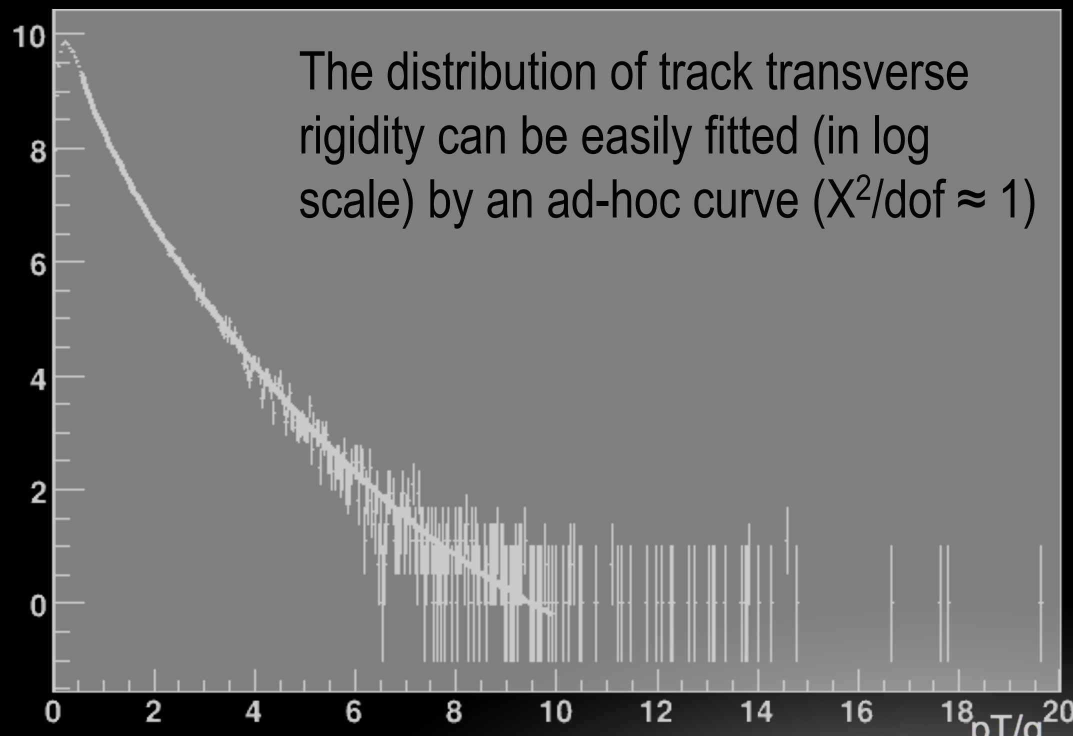


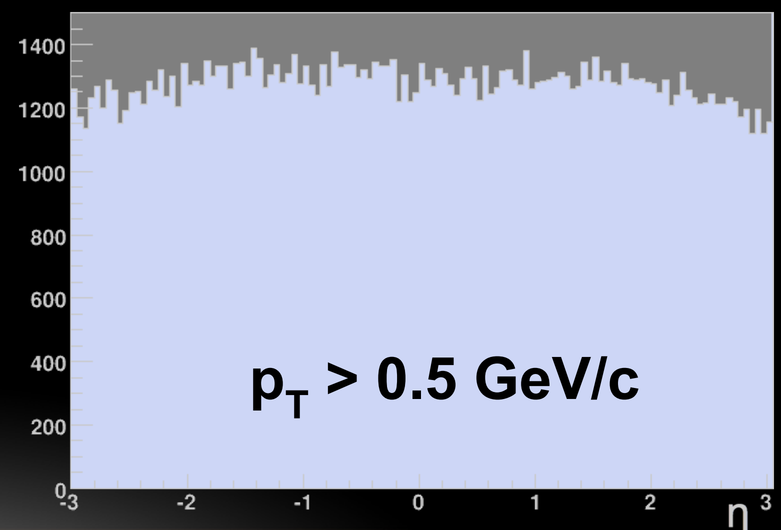
$$x \cong k \cdot \frac{d \cdot r}{p_T}$$

- Hypothesis
  - “High” pT particles (pT>0.5 GeV) are mainly primaries (as extracted from a MB Monte-Carlo Generator)
  - Low pT particles are much more abundant and can be extracted from the predicted occupancy. Low pT stubs derive from a pure combinatorial of hits

Stub Rate Analysis

- For stubs from high-pT particles, tkLayout uses the following fitted distribution of primaries (from MC)





Stub Rate Analysis

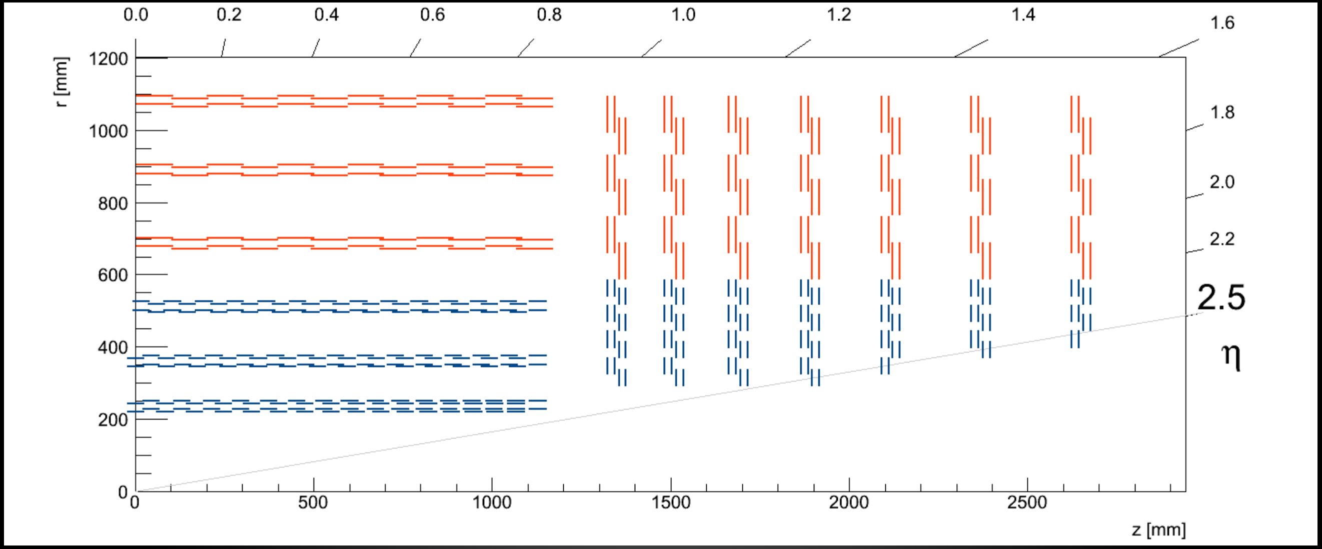
- Figures of merit provided:
  - Trigger efficiency
  - High pT particle frequency
    - Theoretical no. of stubs from high pT hits (per event)
  - Trigger frequency true
    - High pT particle frequency * trigger efficiency
  - Trigger frequency fake
    - no. of stubs from low pT hits making the cut (per event)
  - Trigger purity
    - Stubs from high-Pt / total stubs formed

Outline

- Towards HL-LHC
- The tkLayout tool
  - Architecture
  - Geometry model
  - Material model
- Performance metrics
  - Tracking performance analysis
  - Stub rate analysis
- Case Studies
  - 7 disks vs. 5 disks Endcap geometries
  - Trigger Tower architecture
- Going beyond CMS?
- Conclusions

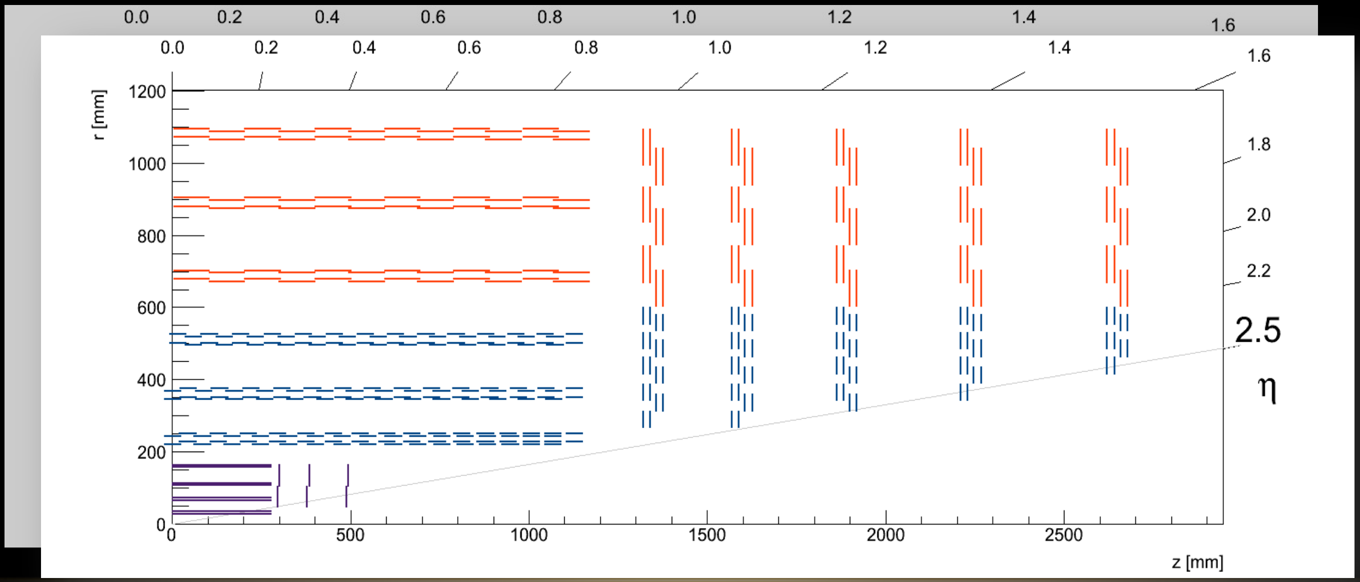
Case study: 7 vs. 5 disks Endcap

- Example of a study done with tkLayout
- Compare two variations of the Barrel Endcap layout
  - Endcap with 7 disks



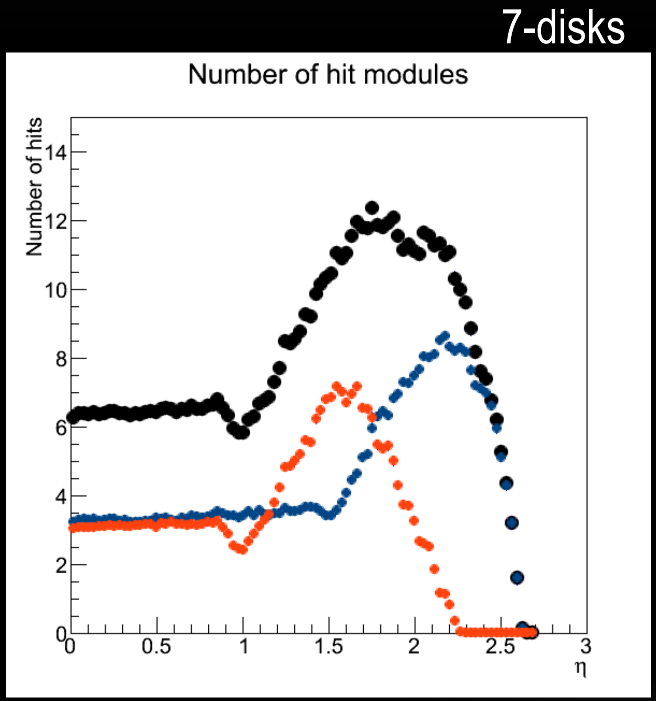
Case study: 7 vs. 5 disks Endcap

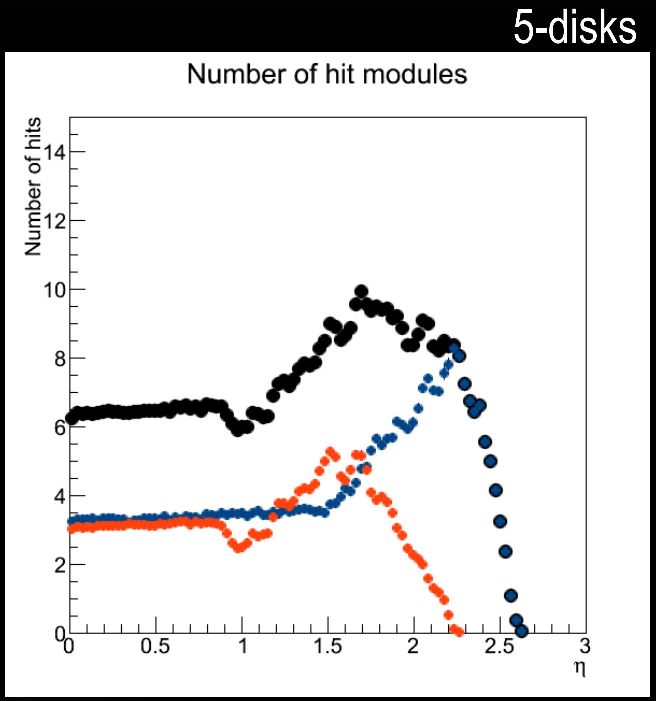
- Example of a study done with tkLayout
- Compare two variations of the Barrel Endcap layout
  - Endcap with 7 disks
  - Endcap with 5 disks



Case study: 7 vs. 5 disks Endcap

- Number of hit modules across eta (per module type)

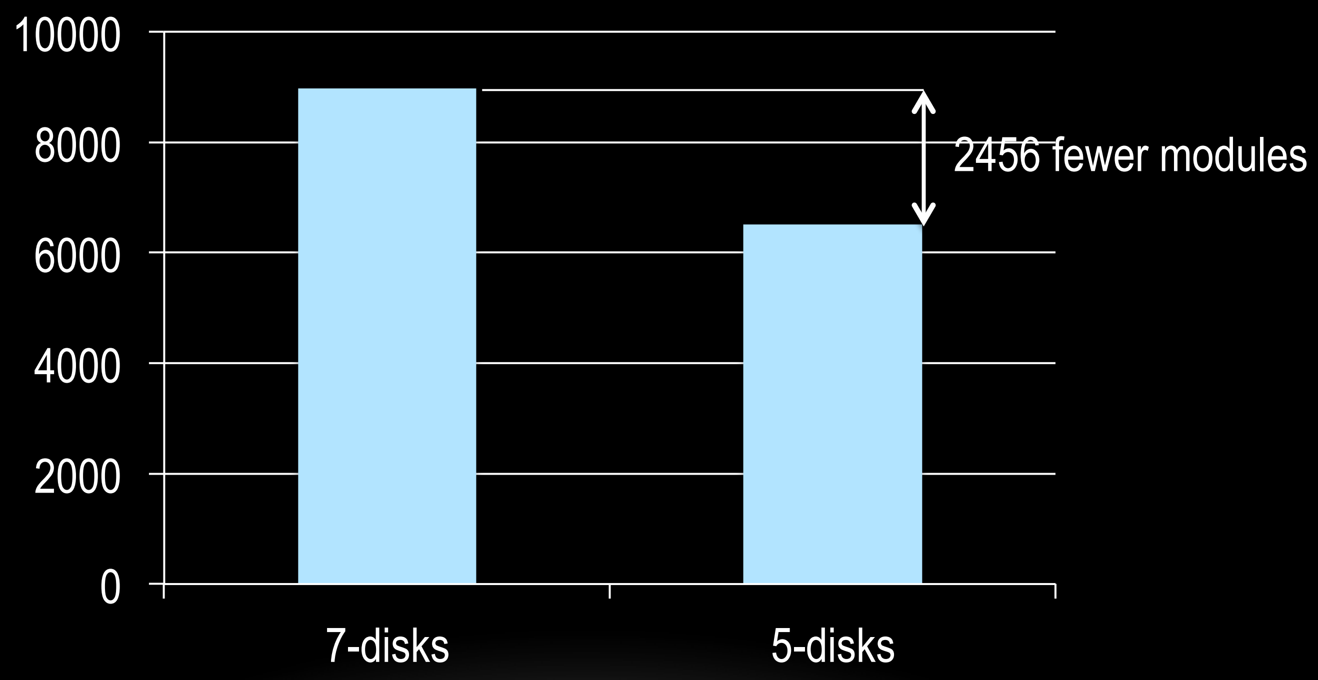




- 7 disks had redundant coverage, 5 disks reduces that

Case study: 7 vs. 5 disks Endcap

- Number of modules in the Endcap

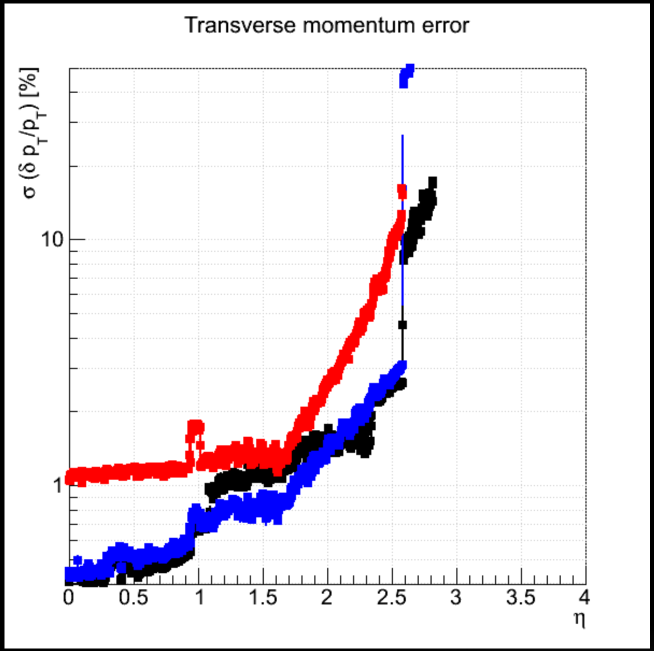


- Fewer modules means lower cost

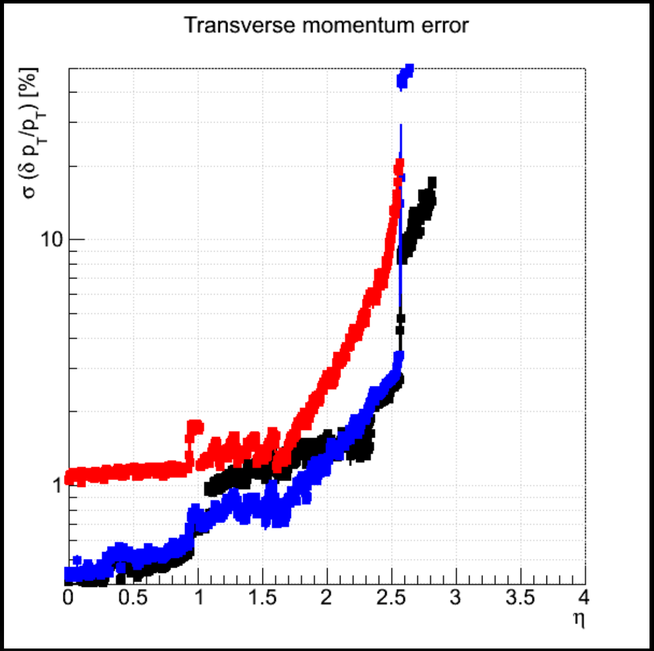
Case study: 7 vs. 5 disks Endcap

- Tracking performance

7-disks



5-disks

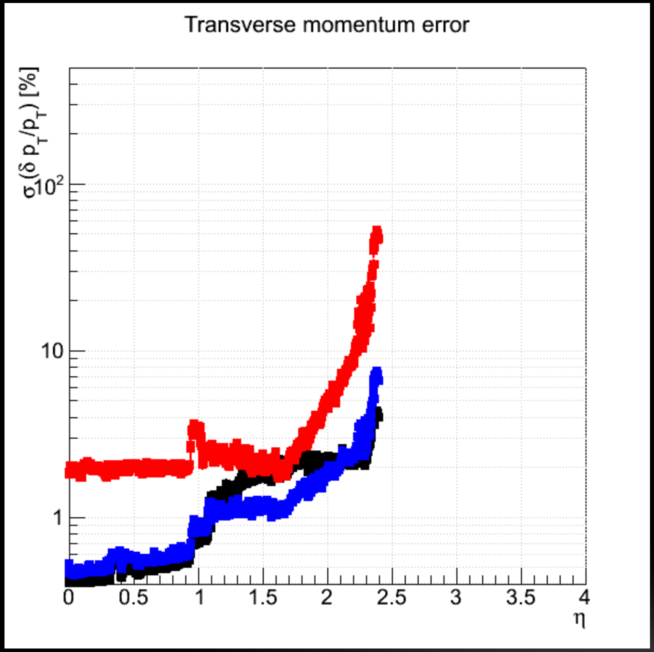


- Performance is just about the same

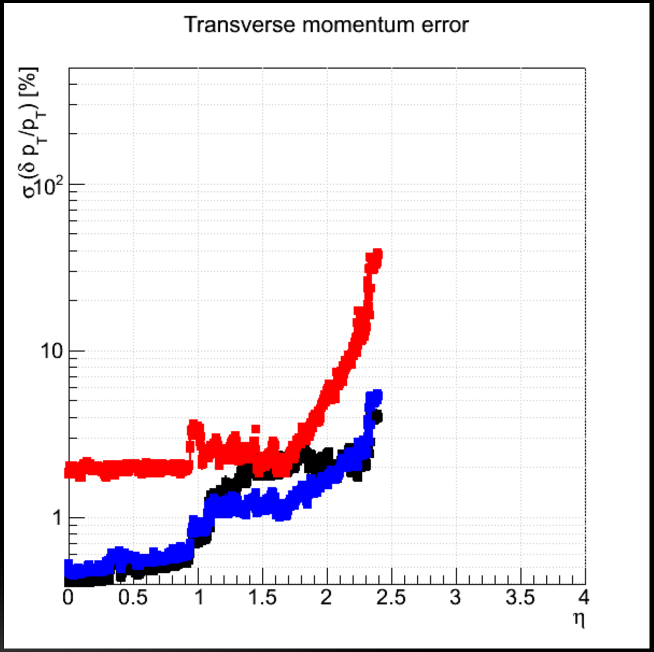
Case study: 7 vs. 5 disks Endcap

- Triggering performance
  - Only pT-aware modules – No help from Pixel detector
  - Up to eta = 2.2

7-disks



5-disks



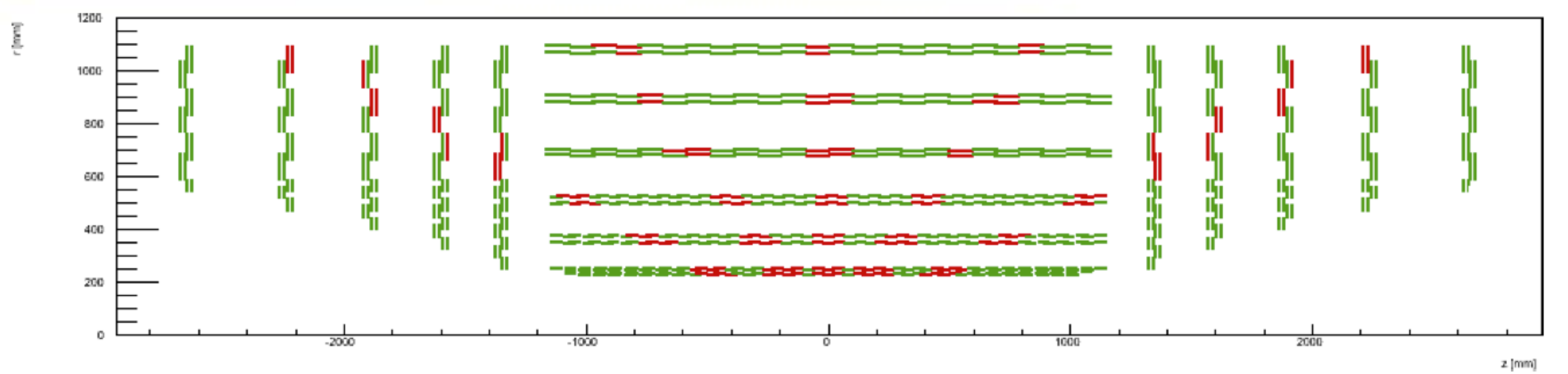
Case study: 7 vs. 5 disks Endcap

- Conclusions
  - No fundamental difference in tracking and triggering performance
  - Substantially fewer modules in the 5-disks configuration
    - Lower cost
    - Lower material interaction
  - 5-disks is the better option

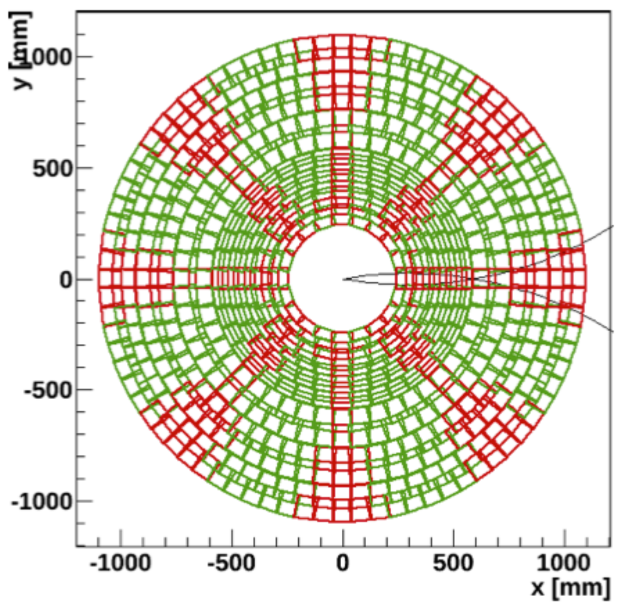
Case study: Trigger Tower Architecture

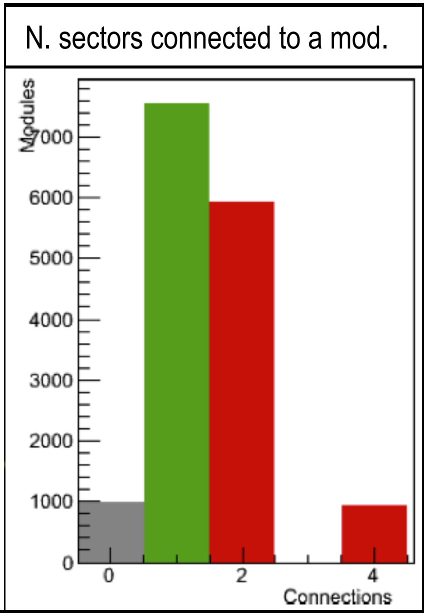
- Adding triggering to the tracker means shipping stub data off-detector
- tkLayout estimates 10s Tbps of trigger data (!!!)
- Tracker divided into sectors, each one belonging to a different Trigger Tower
- Trigger Towers:
  - Are based on Associative Memories or other technologies for track reconstruction
  - Receive all the stubs they need for tracks in their sector
- New feature of tkLayout:
  - Estimate the number of stubs each TT receives
  - Lay out a sector topology map
  - Define the interconnections between TT's

Case study: Trigger Tower Architecture



- Red areas = Overlap areas
  - Modules belonging to more than one TT
- Overlap due to:
  - Luminous region ΔZ
  - Minimum pT cut



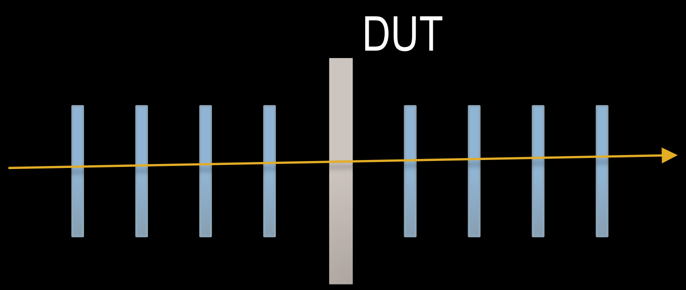


Outline

- Towards HL-LHC
- The tkLayout tool
  - Architecture
  - Geometry model
  - Material model
- Performance metrics
  - Tracking performance analysis
  - Stub rate analysis
- Case Studies
  - 7 disks vs. 5 disks Endcap geometries
  - Trigger Tower architecture
- Going beyond CMS?
- Conclusions

Going beyond CMS?

- tkLayout can be quickly adapted to model other types of detector than the CMS tracker
  - TAC-PF (Turkish Accelerator Center - Particle Factory)
    - Already shown interest
    - Support can be easily added
  - Beam Test
    - Will require some extension



Conclusions

- tkLayout is versatile
  - Can model many types of CMS tracker geometries
  - Can be extended to model in principle any kind of detector (TAC-PF, beam test, …)
  - Can be extended with additional analyses and parameterizations
- A host of information at a glance
  - Easy to make comparisons between different layouts during the early design phase
- Very quick to setup and run
  - Can be used to make quick, iterative optimizations to a layout
  - Non-simulative approach proves to be a winning choice for this kind of tasks
- For all these reasons tkLayout has been a key tool in the design of the CMS Tracker for Phase 2 Upgrade
  - And who knows, perhaps other detectors one day?

Thanks

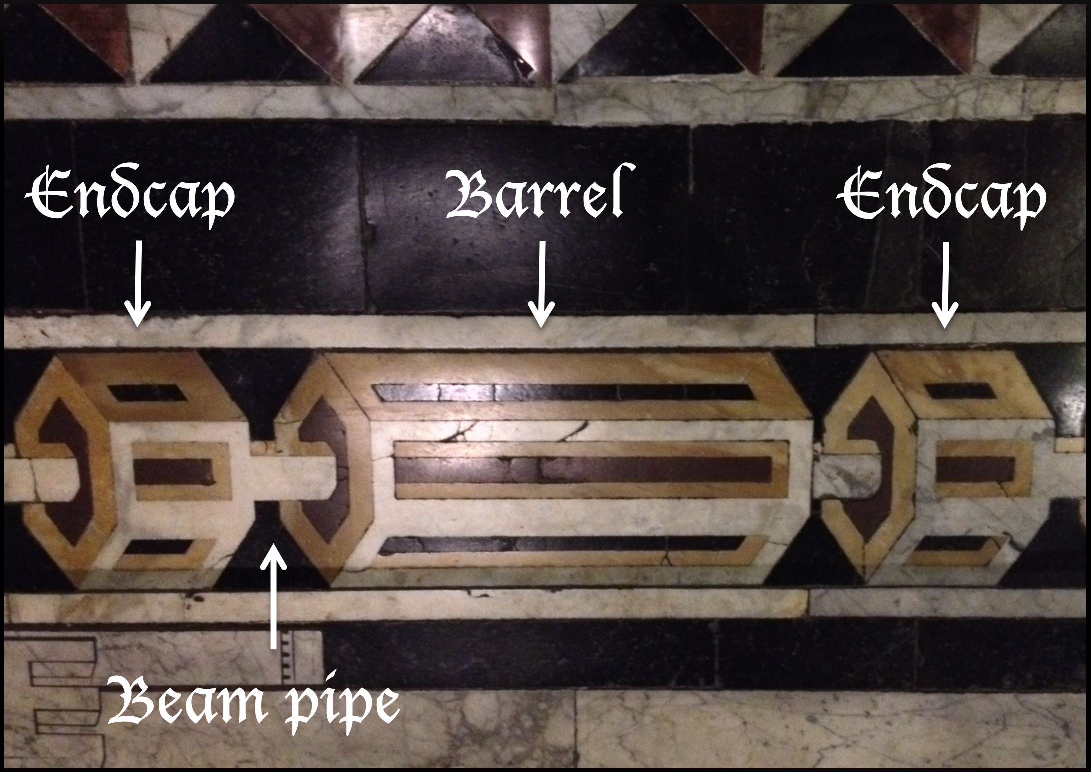
Bonus slides

Breakthrough discovery

- Went around the city a bit
- Beautiful city!
- Went to the Siena Cathedral with some fellow conference participants

- And we discovered….

Particle Detector ca. AD 1300



- Found on a floor tile in the Dome of Siena
- It is **clearly** a particle detector!!!1
- Close to other tiles depicting low orbit satellites with eerie accuracy
- Enough for Dan Brown to write another book involving the Vatican, the Illuminati and the hypersonic CERN jet plane???
- Is the Higgs Boson actually the GOD'S PARTICLE???

Thanks again ☺

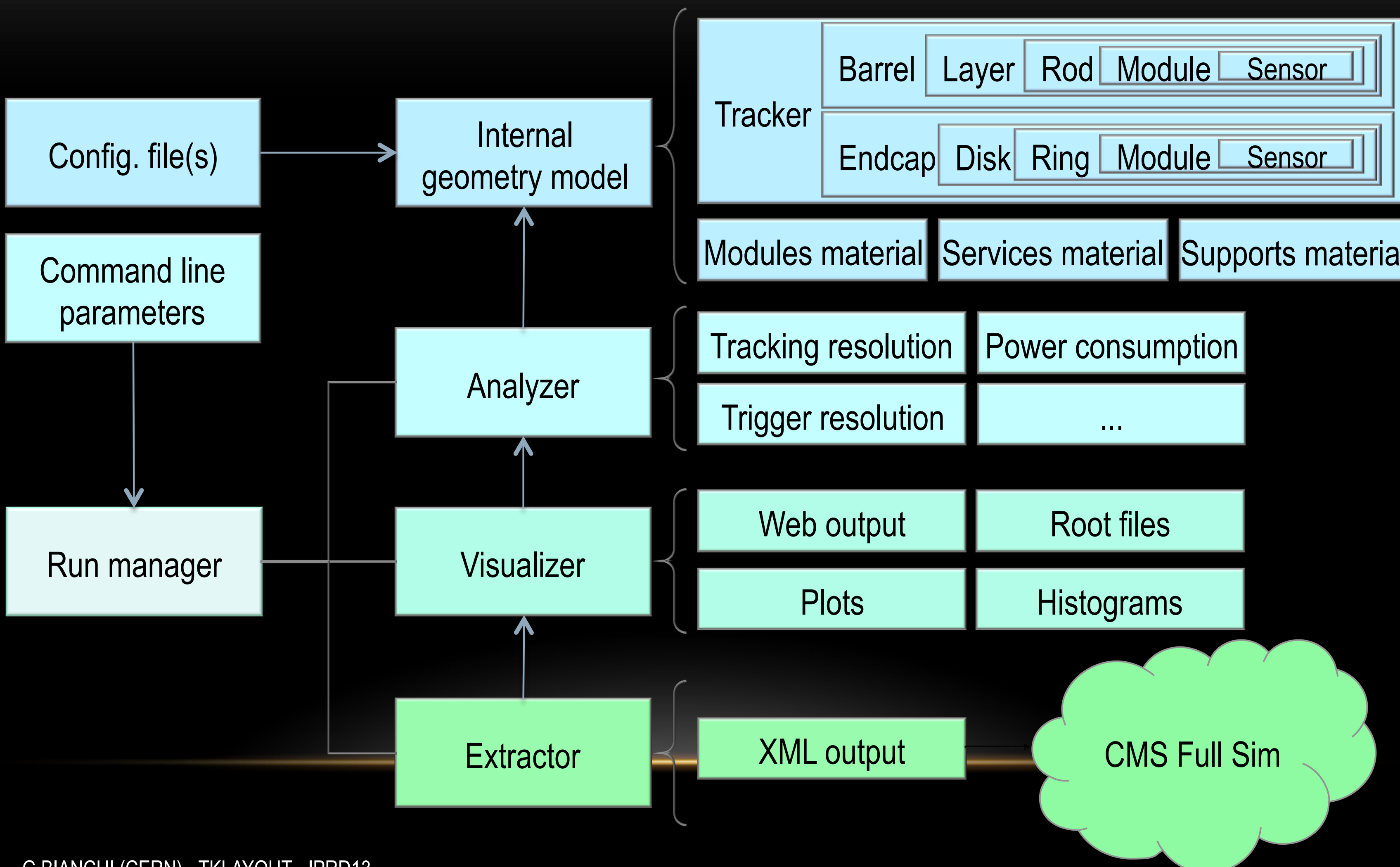
Backup

Towards HL-LHC

	≤ 2012	LS1	2015-17	LS2	2019-21	LS3	2023 HL-LHC
E_{CM} [TeV]	8		14		14		14
L [$cm^{-2}s^{-1}$]	8×10^{33}		$\leq 2 \times 10^{34}$		$\leq 2.5 \times 10^{34}$		$\sim 5 \times 10^{34}$
$\int L$ [fb^{-1}]	≤ 30		≤ 50		≤ 50		≤ 3000
$\langle\mu\rangle$	≤ 50		≤ 120		≤ 120		≤ 140

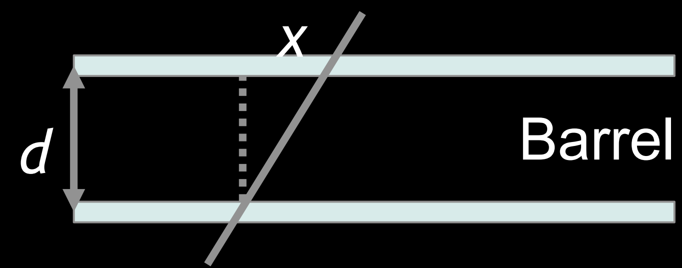
- HL-LHC will entail much harsher irradiation conditions
- Much more data to readout

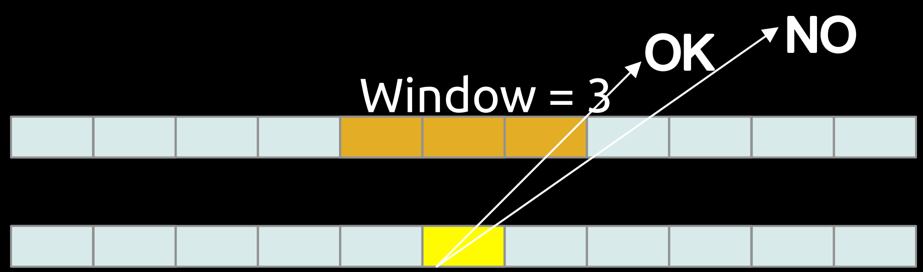
tkLayout architecture



Stub Rate Analysis

- For the tuning of sensor spacing and trigger windows
- Given spacing and window it is possible to compute the probability of a stub being found as a function of pT





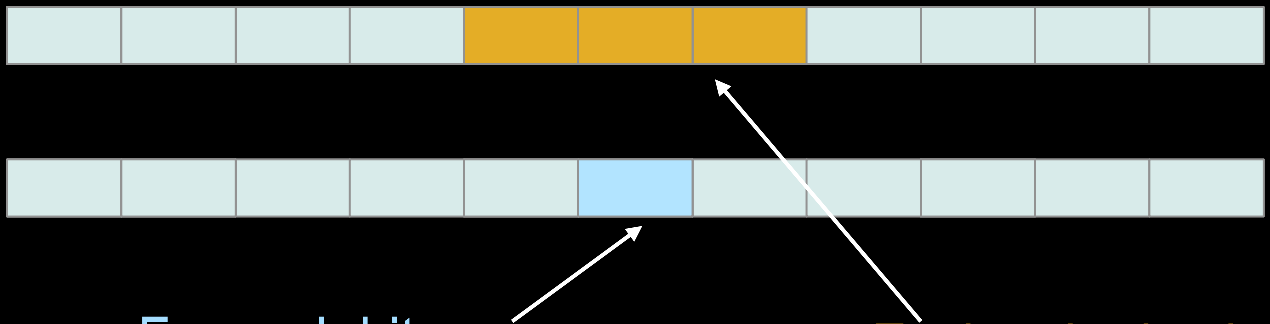
$$x = \frac{d}{\sqrt{a-1}} \qquad a = \left(\frac{2p}{0.3 \cdot B \cdot r \, \frac{GeV}{T \cdot m}} \right)^2$$

- Hypothesis
  - “High” pT particles (pT>0.5 GeV) are mainly primaries (as extracted from a MB Monte-Carlo Generator)
  - Low pT particles are much more abundant and can be extracted from the predicted occupancy

Stub Rate Analysis

- Stubs from low-pT particles are derived by pure combinatorial of hits

Window = 3



For each hit on one sensor

Each paired strip in window prob = occupancy

$$N \times occ \quad \times \quad W \times occ$$

Web Output

3xPS_3x2S_5disks_longer

layouts

geometry | geometry (pixel) | bandwidth | trigger cpus | power | material (outer) | material (pixel) | weights (outer) | resolution | resolution (trigger) | trigger info | log page

layers and disks

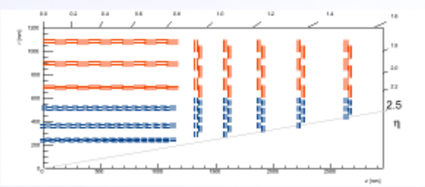
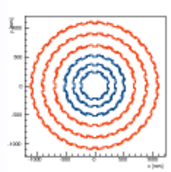
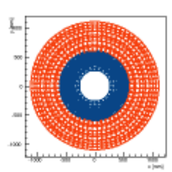
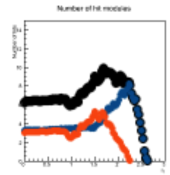
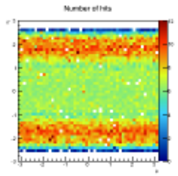
Layer	L1	L2	L3	L1	L2	L3	Total
r	230	357	508	686	888	1080	
# mod	1008	1320	1836	1152	1488	1824	8628
# rods	16	24	34	48	62	76	

Disk	D1	D2	D3	D4	D5	Total
z	1349	1597	1891	2239	2650	
# mod	688	688	664	640	580	6520

Ring	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
r_{min}	263	310	340	387	414	462	484	533	552	602	669	774	835	941	994
r_{max}	309	356	386	434	460	508	531	579	599	702	770	874	936	1041	1095

modules

plots

Web Output

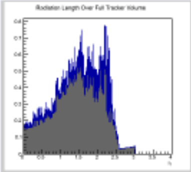
3xPS_3x2S_5disks_longer

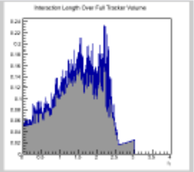
layouts

geometry | geometry (pixel) | bandwidth | trigger cpus | power | **material (outer)** | material (pixel) | weights (outer) | resolution | resolution (trigger) | trigger | info | log page

1d overview

Average radiation length in full volume (eta = [0, 4.0]) 0.23172
Average interaction length in full volume (eta = [0, 4.0]) 0.07371



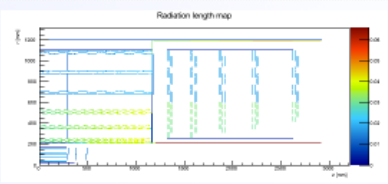


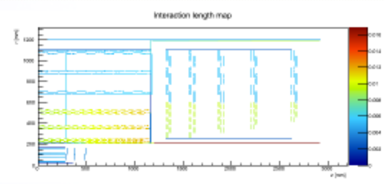
tracking volume

detailed

module components detail

material distribution





nuclear interactions