ALICE MUON Software for run 3

Sean Murray
Physics
University of Cape Town

July 6, 2016

Outline

ALICE

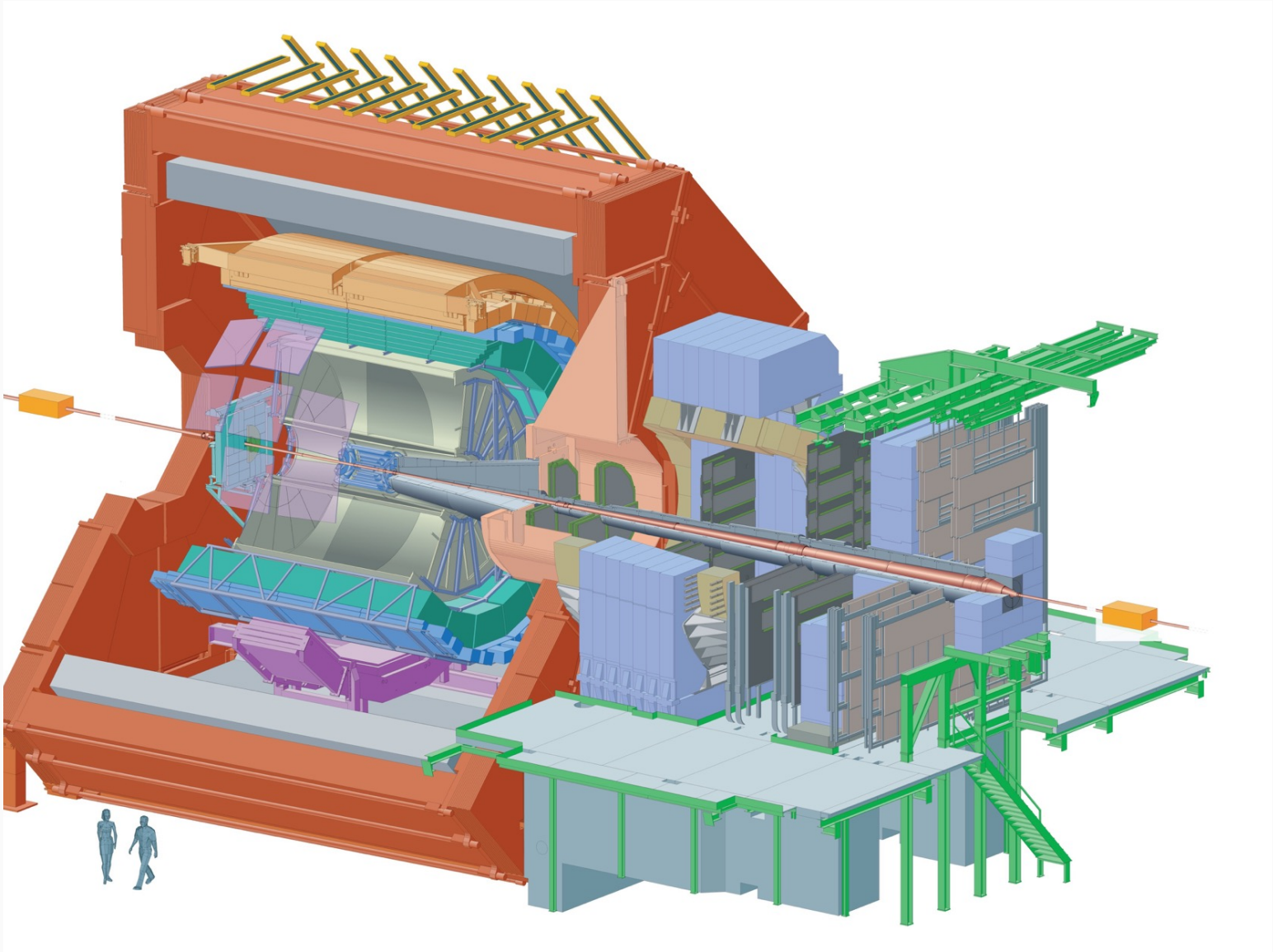
O2 and Run3

ALICE Muons

Current MUON Software

Cluster Finder

ALICE



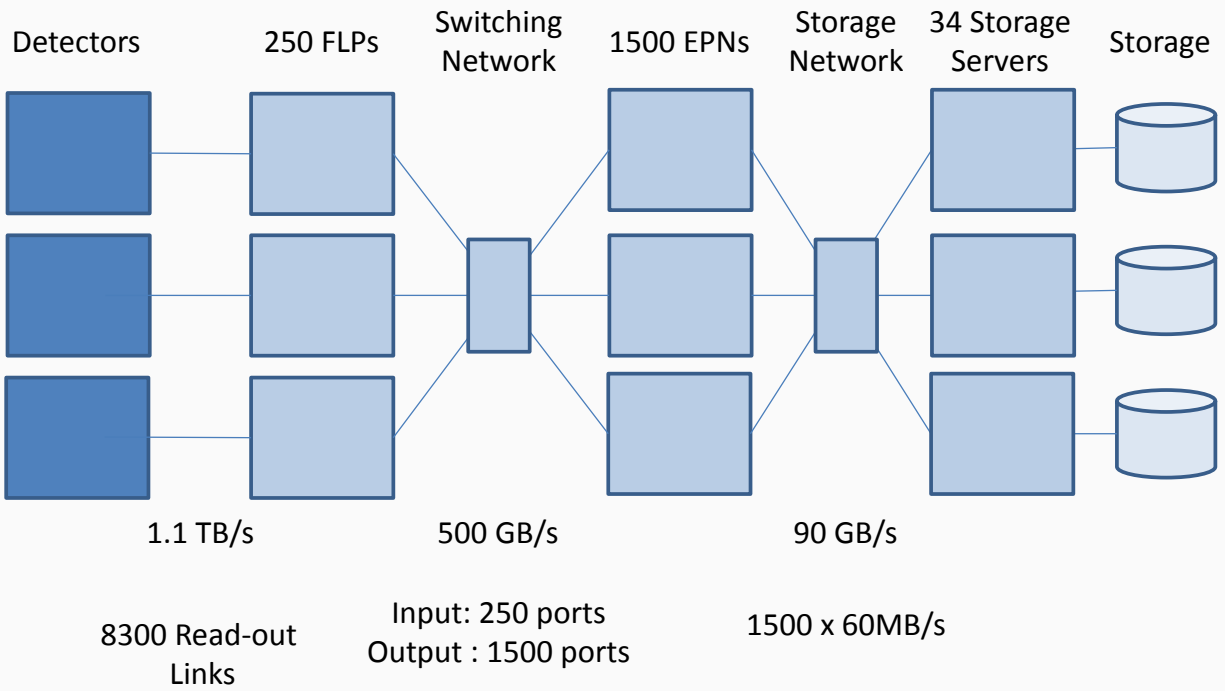
Outline

O2 Online/Offline

We will not go into depth, this is a whole other presentation.
Important numbers to take away :

- 50kHz event rate.
- 1.1 TB/s aggregate data rate.
- continuous read out for most detectors, TPC and MUON joint chip design.
- Completely new framework.
- merge HLT, data acquisition and offline into 1 code base and platform.
- calibration and reconstruction online.
- Collaboration with FAIR for new framework.

O2 Farm


Detectors
250 FLPs
Switching Network
1500 EPNs
Storage Network
34 Storage Servers
Storage
1.1 TB/s
500 GB/s
90 GB/s
8300 Read-out Links
Input: 250 ports
Output : 1500 ports
1500 x 60MB/s

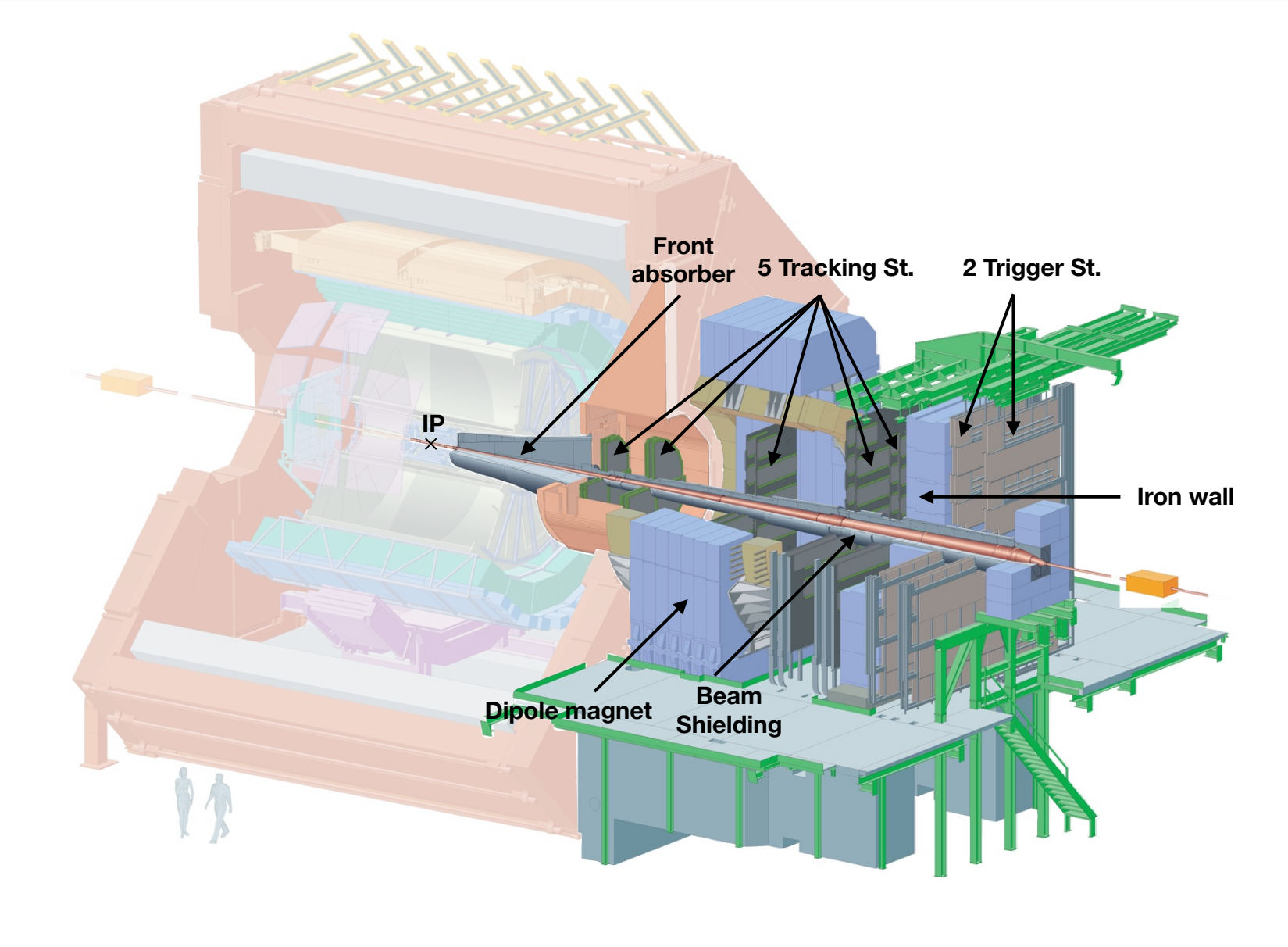
MUON Links

Detector	Link type	Number of links DDL1	DDL2	GBT	Read-out board type	Number of boards C-RORC	CRU
ACO	DDL1	1			C-RORC	1	
CPV	DDL1	6			C-RORC	1	
CTP	GBT			14	CRU		1
EMC	DDL2		20		C-RORC	4	
FIT	DDL2		2		C-RORC	1	
HMP	DDL1	14			C-RORC	4	
ITS	GBT			495	CRU		23
MCH	GBT			550	CRU		25
MFT	GBT			304	CRU		14
MID	GBT			32	CRU		2
PHS	DDL2			16	C-RORC	4	
TOF	GBT			72	CRU		3
TPC	GBT			5832	CRU		324
TRD	Custom			1044	CRU		54
ZDC	GBT			1	CRU		1
Total		21	38	8344		15	447

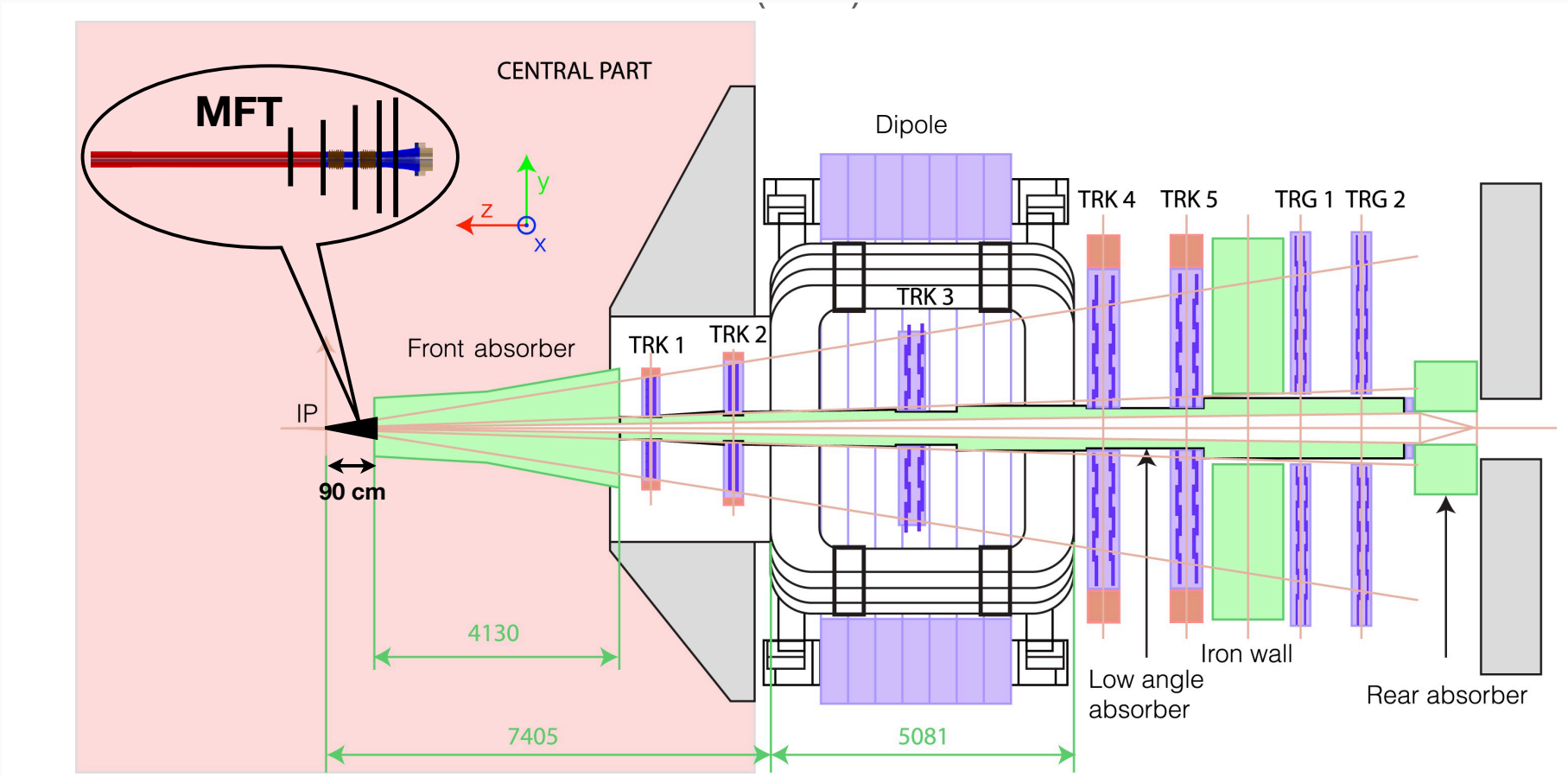
Muon therefore get 6 FLPs

Outline

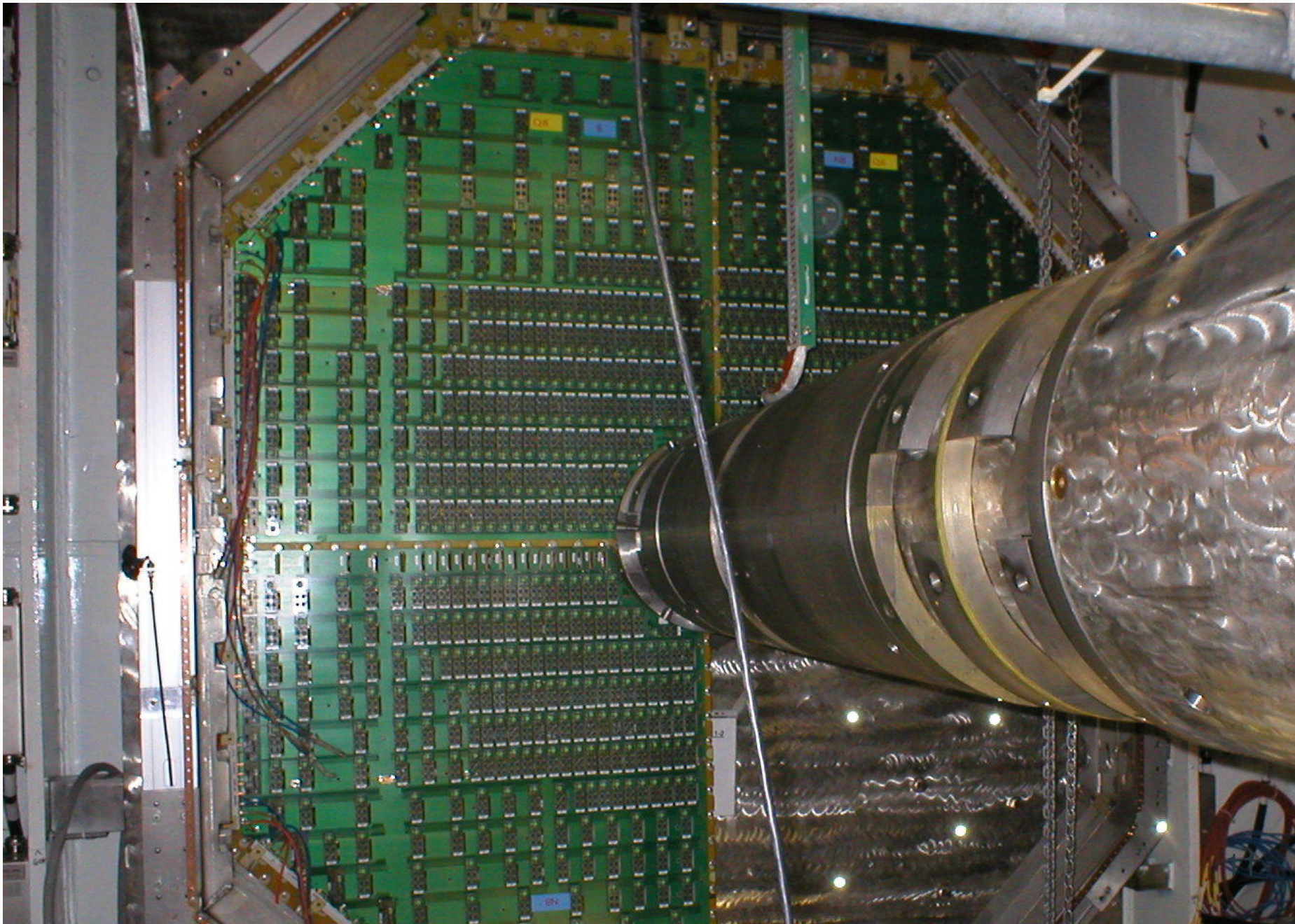
ALICE MUON Arm



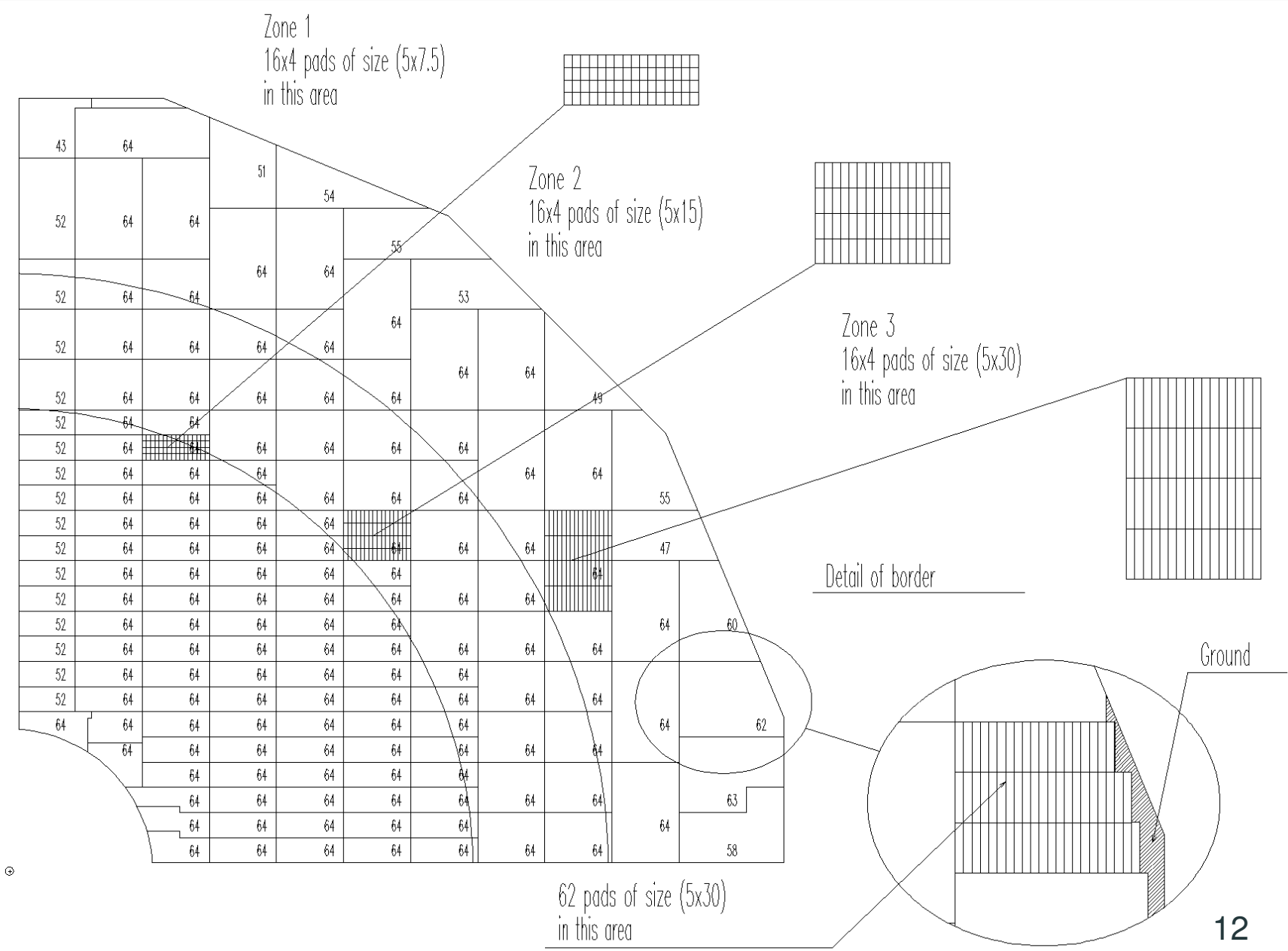
ALICE MUON diagram



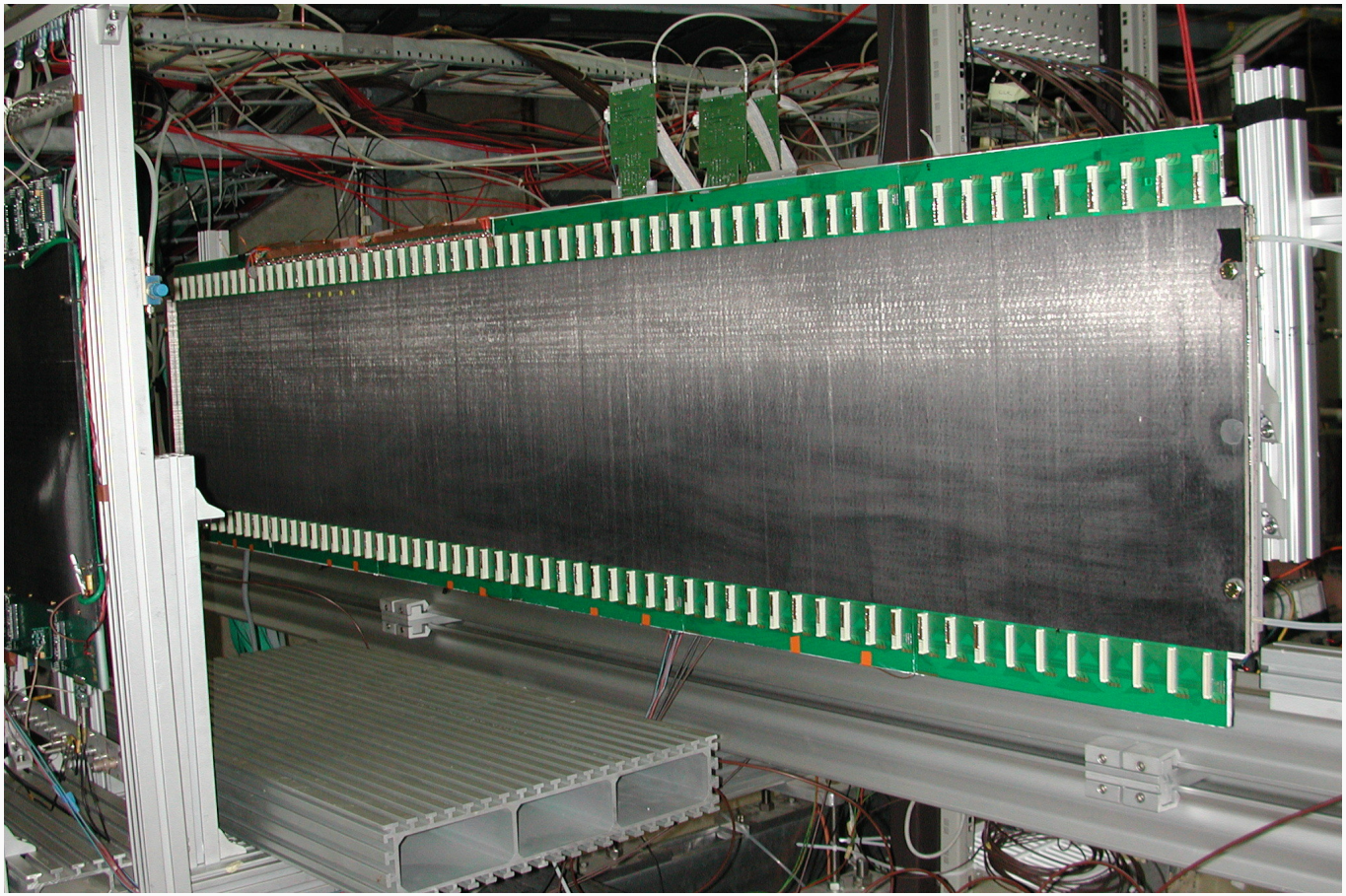
Detector Structure Station1



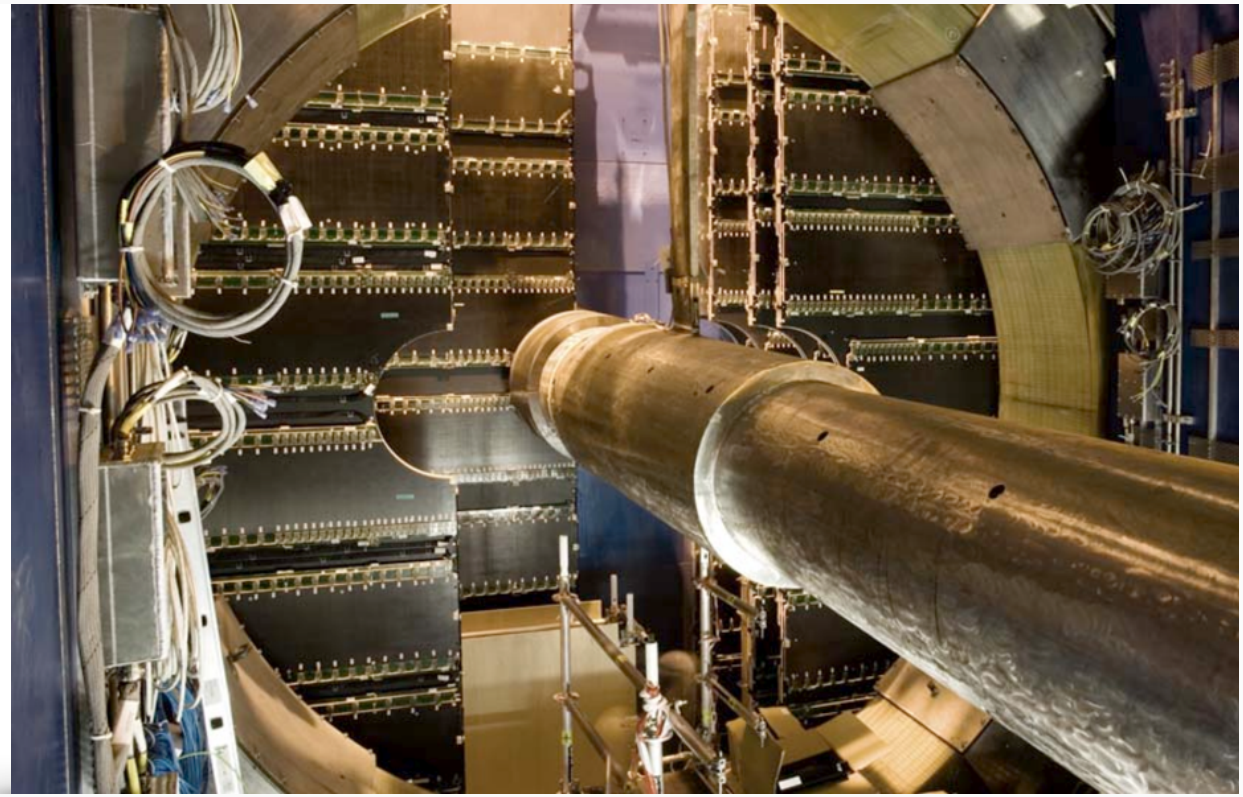
Detector Structure Station1 quadrant



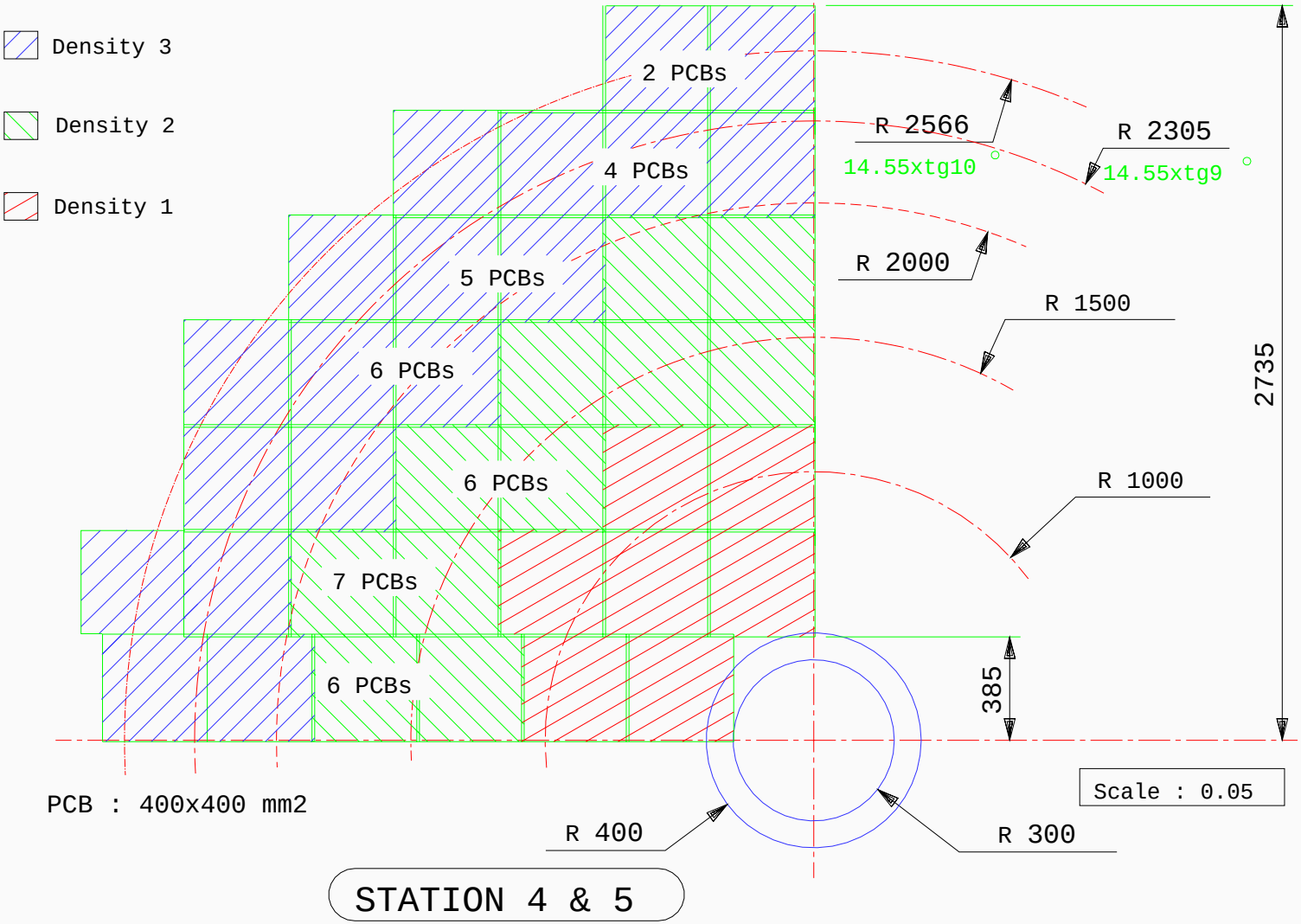
Detector Slats



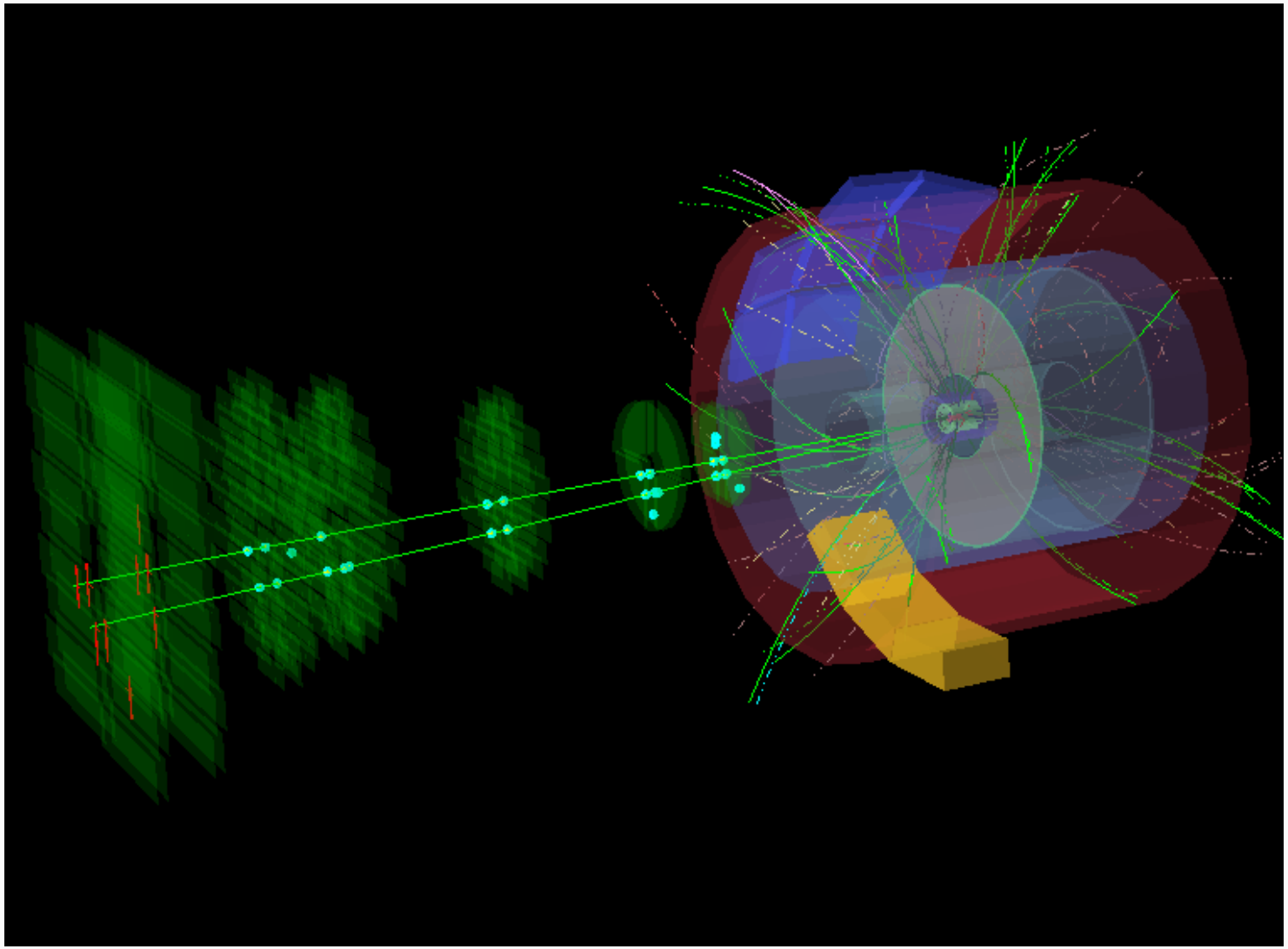
MUON Slats through the Dipole



Detector Slats schematic



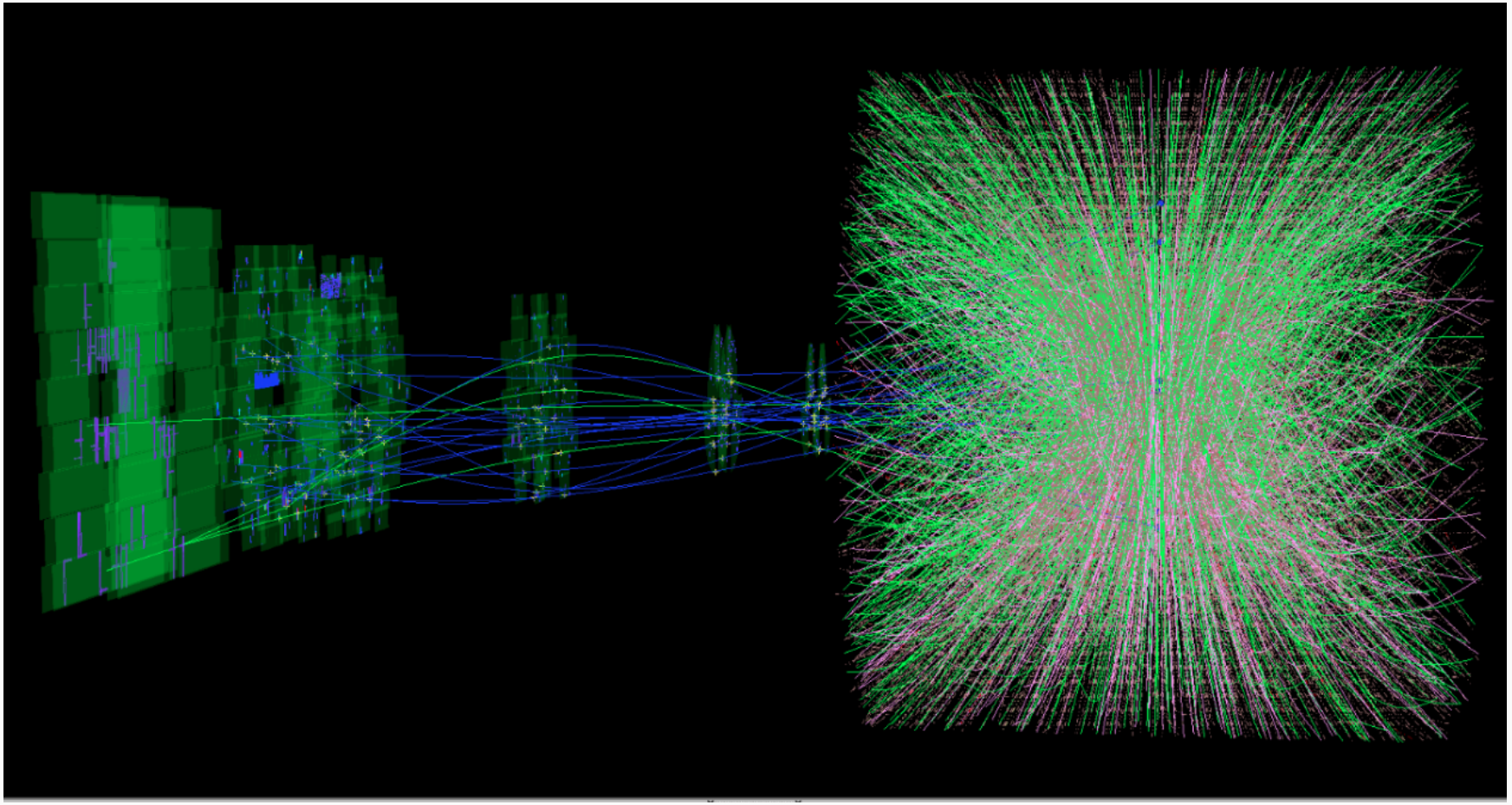
Nominal pp event



pp at $\sqrt{s}$=7TeV, 2010

Nucl.Phys. A862-863 (2011) 223-230

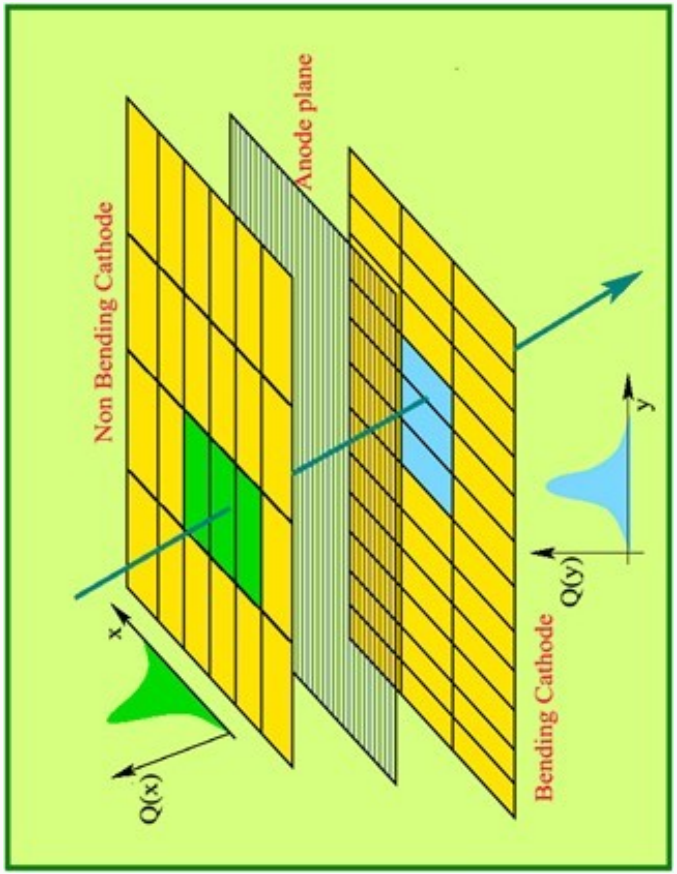
Nominal PbPb event



PbPb at $\sqrt{s}$=7 TeV, 2015

Diagram of detection

- 100m^2 total area
- 1.4 million channels
- Wire diameter = 20μm
- Wire Pitch : 2*mm* St1
  2.5*mm* St 2,3,4,5
- Pad sizes
  1. 5x7.5*mm*
  2. 5x15*mm*
  3. 5x30*mm*



Outline

Current MUON reconstruction Software

- Raw Data Decoding
- Raw Data Filtering
- Pre Clustering
- Clusterise, locate cluster interaction point.
- Tracking (MCH)
- Tracking (MID)
- Track Matching MCH-MID

MUON software upgrade in lieu of Run3

- O(500) times faster at least
- Of all computing muon are 15% of the time spent.
- 1.1-1.5s to reconstruct depending on beam.
- Aim to have everything online, AOD's out.
- Test in current HLT.

where to start

> *There is no doubt that the grail of efficiency leads to abuse. Programmers waste enormous amounts of time thinking about, or worrying about, the speed of non-critical parts of their programs, and these attempts at efficiency actually have a strong negative impact when debugging and maintenance are considered. We should forget about small efficiencies, say about 97% of the time: premature optimization is the root of all evil. Yet we should not pass up our opportunities in that critical 3%. A good programmer will not be lulled into complacency by such reasoning, he will be wise to look carefully at the critical code; but only after that code has been identified.*

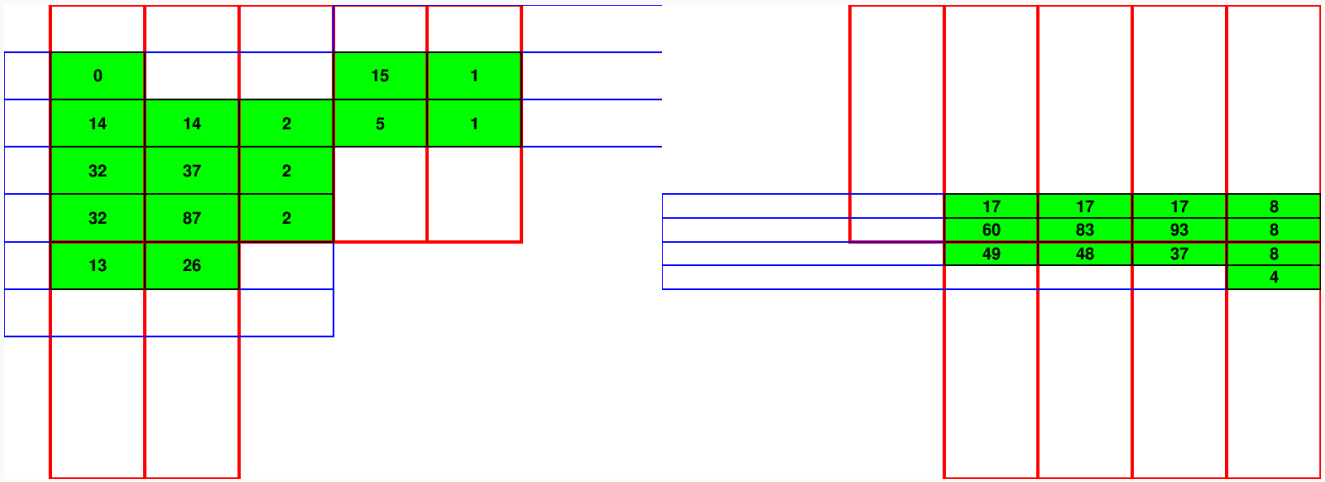
Donald Knuth, ACM Computing Surveys, Vol 6, No. 4, Dec. 1974 (see p.268)

Time Spent

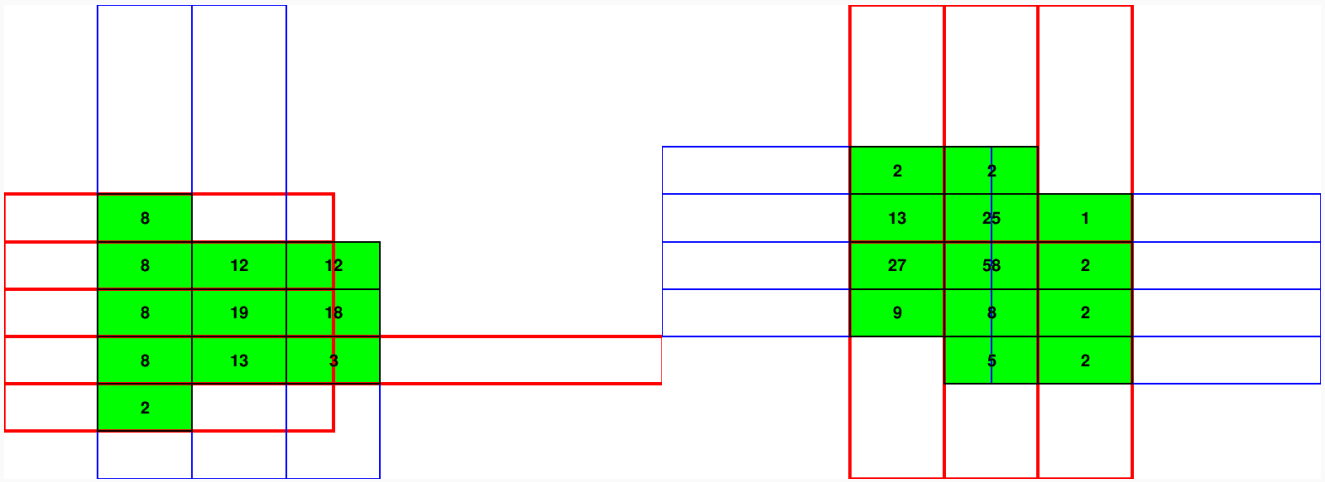
Function	Time in %		
	pp 16	PbPb 11	PbPb 15
Raw Data Decoding	4	3	1
Raw Data Filtering	2	2	1
Pre Clustering	10	10	10
Clustering	63	68	72
Tracking MCH	7	6	5
Tracking MID	6	6	6
Track matching	8	5	5

Outline

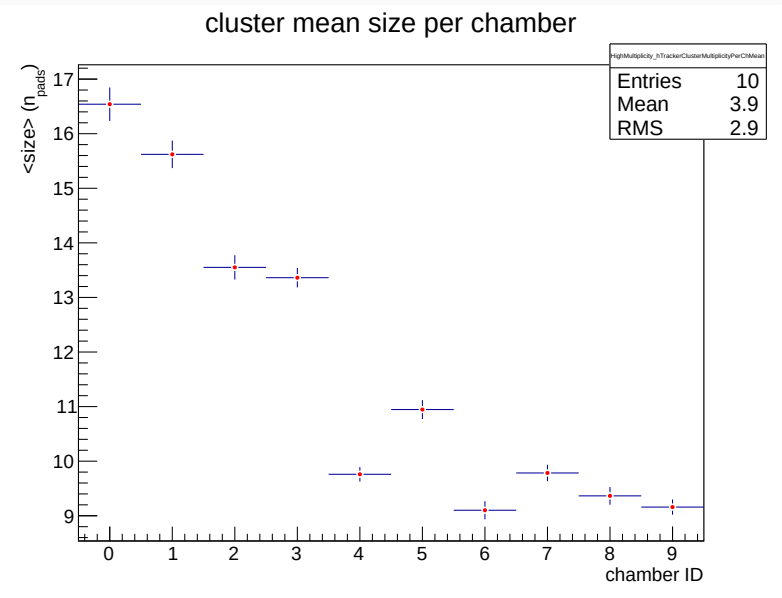
What is a cluster finder

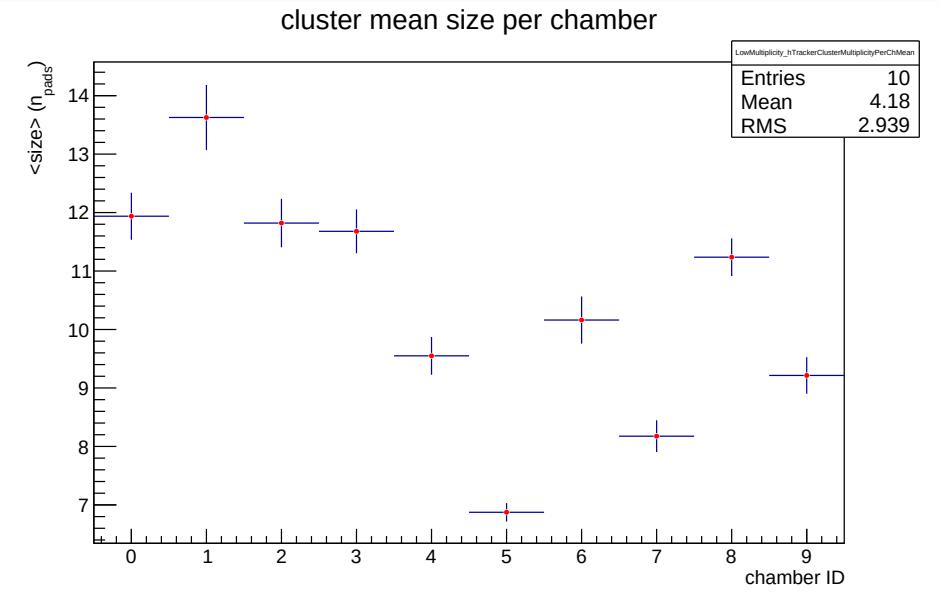


What is a cluster finder



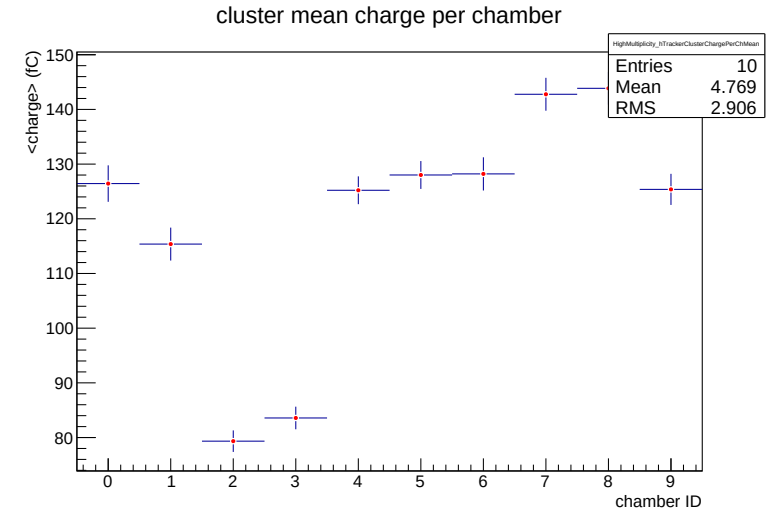
Cluster Mean Size per Chamber PbPb

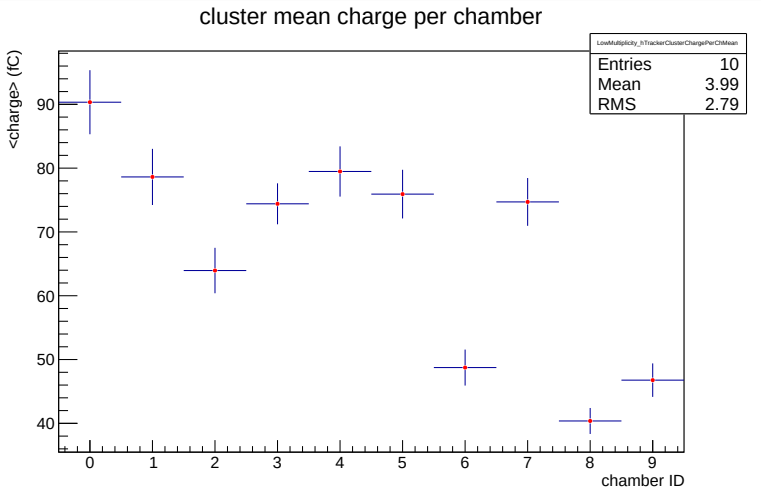




PbPb 2015 on the left, pp 2016 on the right.

Cluster Mean Charger per Chamber

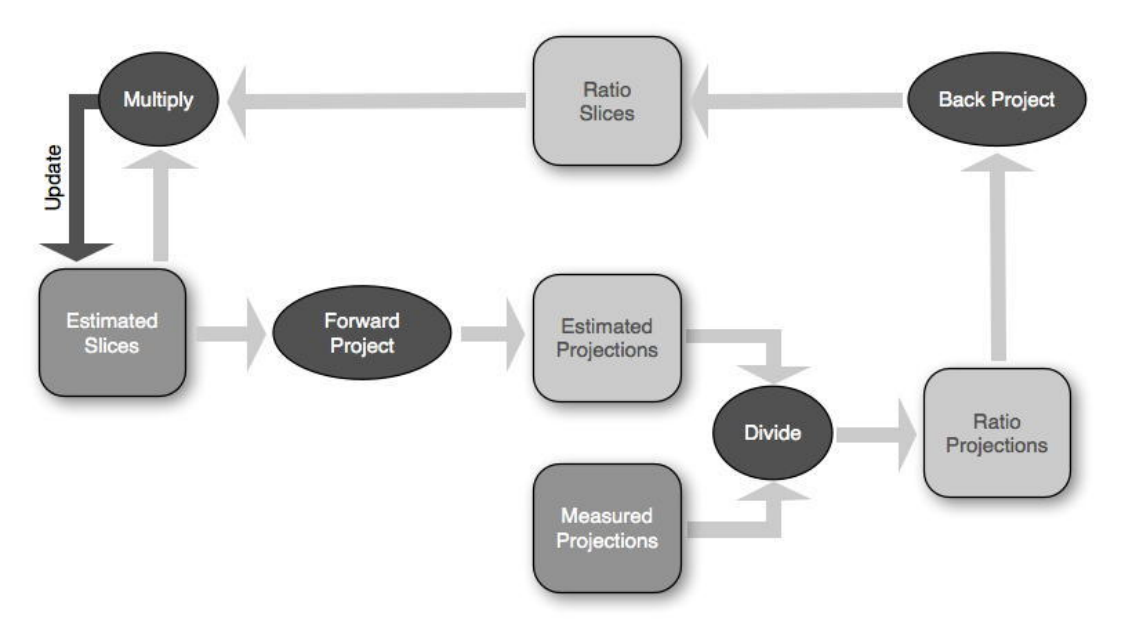




PbPb 2015 on the left, pp 2016 on the right.

Maximum Likelihood - expectation maximization (MLEM) Algorithm

Used extensively in medical imaging, particularly PET scanners.



MLEM

$$\lambda_j^{k+1} = \frac{\lambda_j^k}{\sum_i^m C_{ij}} \sum_i^m \frac{C_{ij}}{\sum_j^m C_{ij} \lambda_j^k} \tag{1}$$

- λ_j^k value of reconstructed charge at j for k^{th} iteration
- y_i measured projection data at i-th bin
- C_{ij} detection probability pixel j to projection bin i. The overlapping area pads and pixels in our case.

Logical Parts

1. Add Virtual pads if necessary, smooth out holes.
2. MLEM until sufficiently small pixel size is achieved.
3. Find Center of Gravity around maximal pixel
4. loop back with smaller and smaller pixels
5. Remove Pixels with low signal or visibility, preserving charge to nearest neighbor.
6. MLEM algorithm again.

MLEM :

- 4 way nested dependent loops.

Speed up

- look for dead code 11% (Finding nearest neighbor)
- change data or reorder to hit caches better. 5%
- get a faster cpu. sadly not going to happen
- gpu 30%, but 4 nested dependent loops
- fpga - try to avoid.
- do something completely different.

Performed on a i7 with nvidia gtx 980, will have to be repeated on other hardware.

Something else

At this stage now.