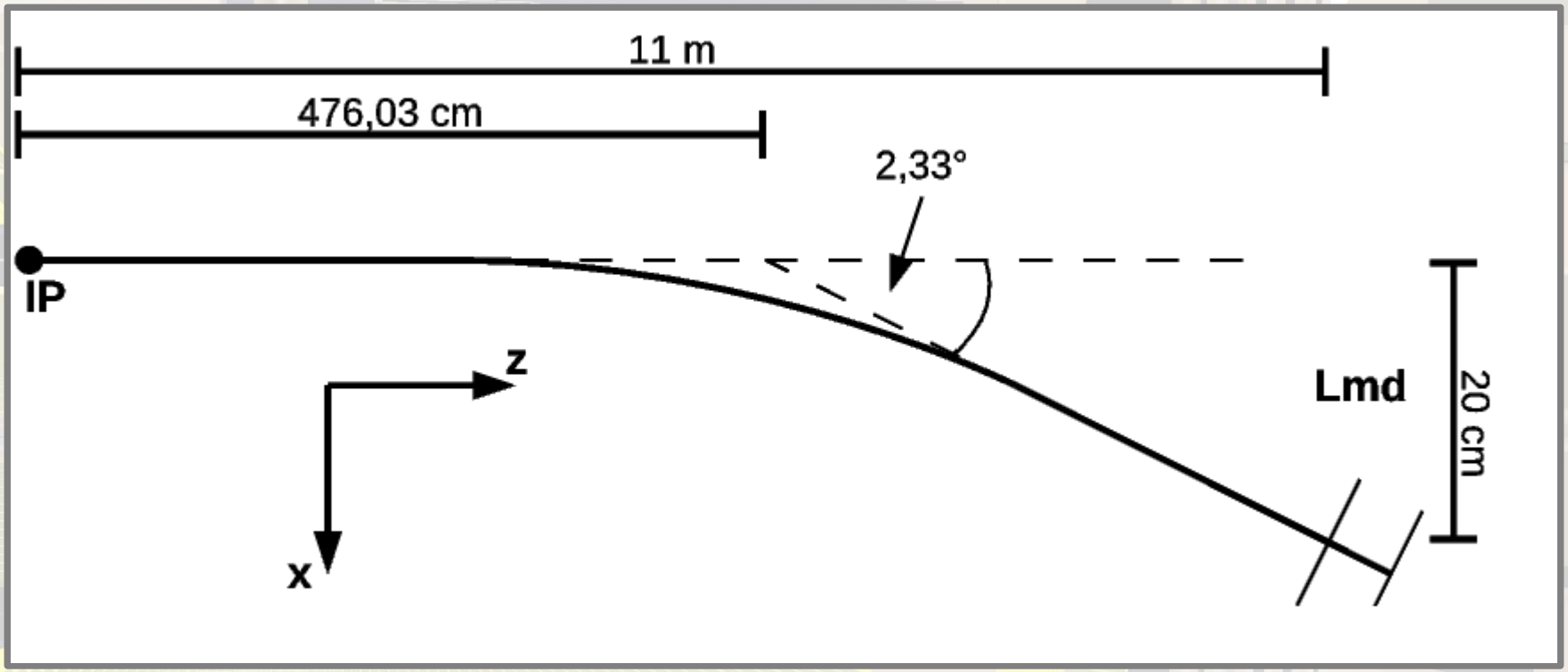
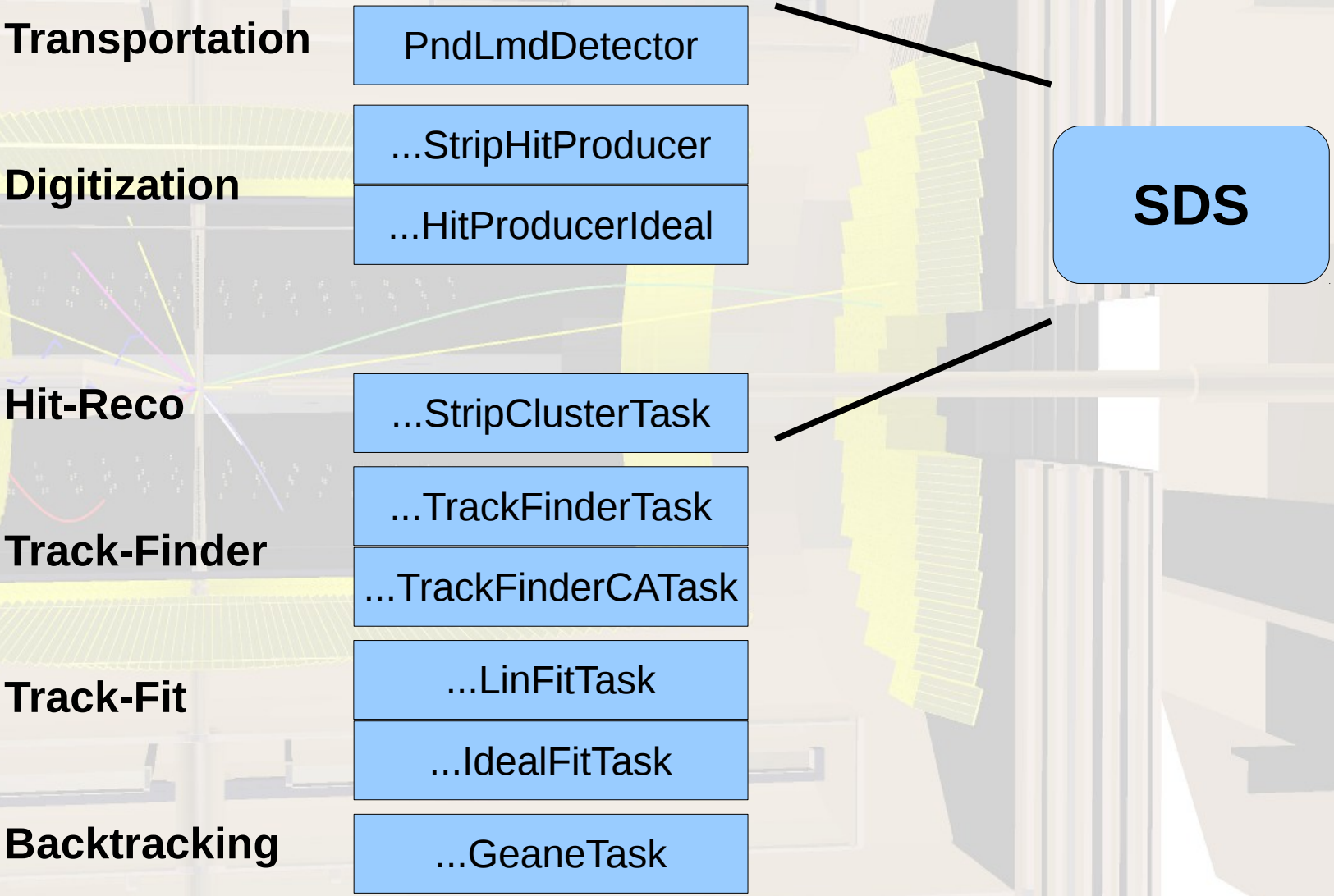
# Lumi Software Update: Pixels

- Software Status
- New Design
- Software Update
- Differences to Strip-Sensor Design
- Outlook

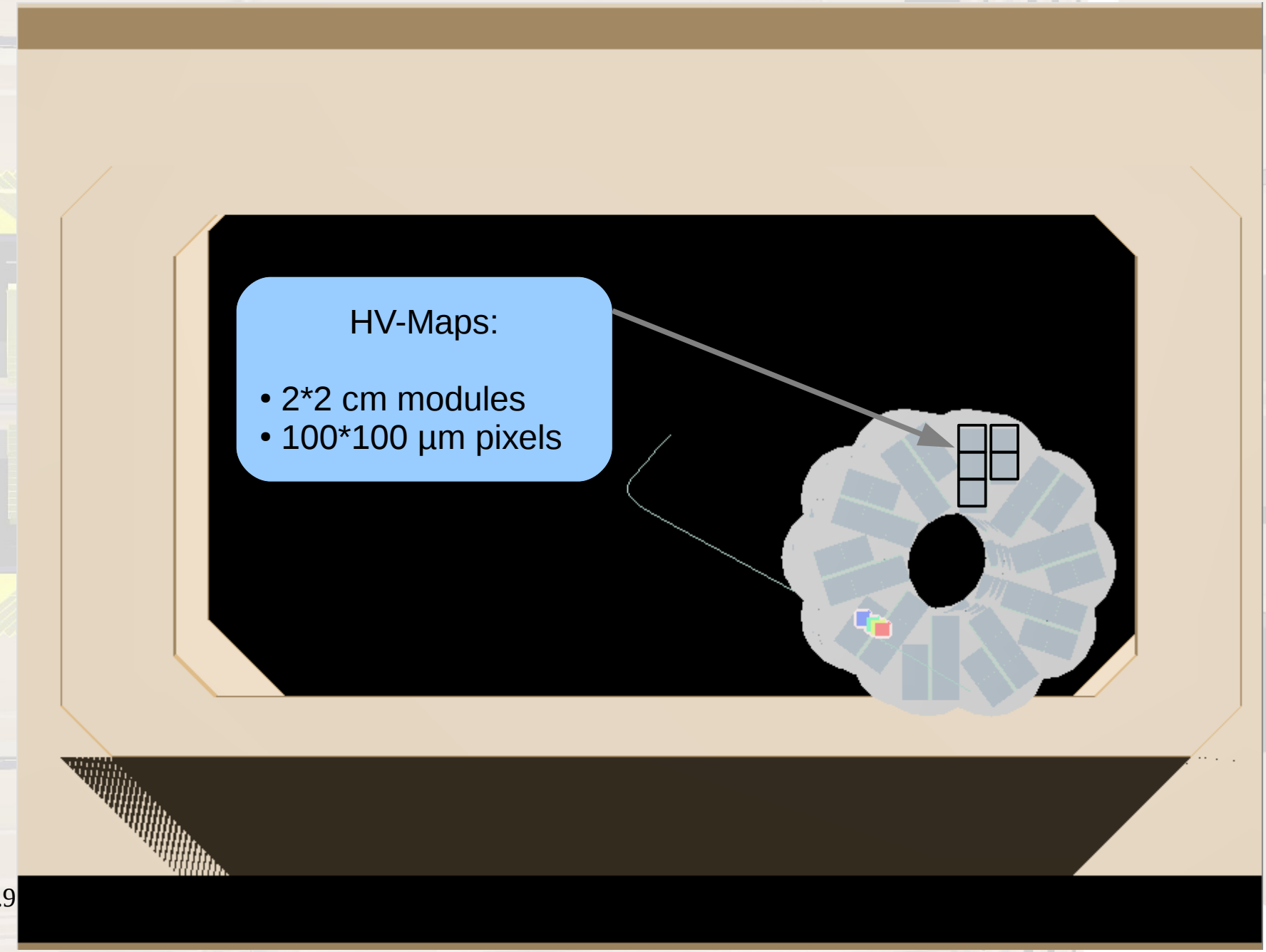
# Luminosity Detector (Lumi)



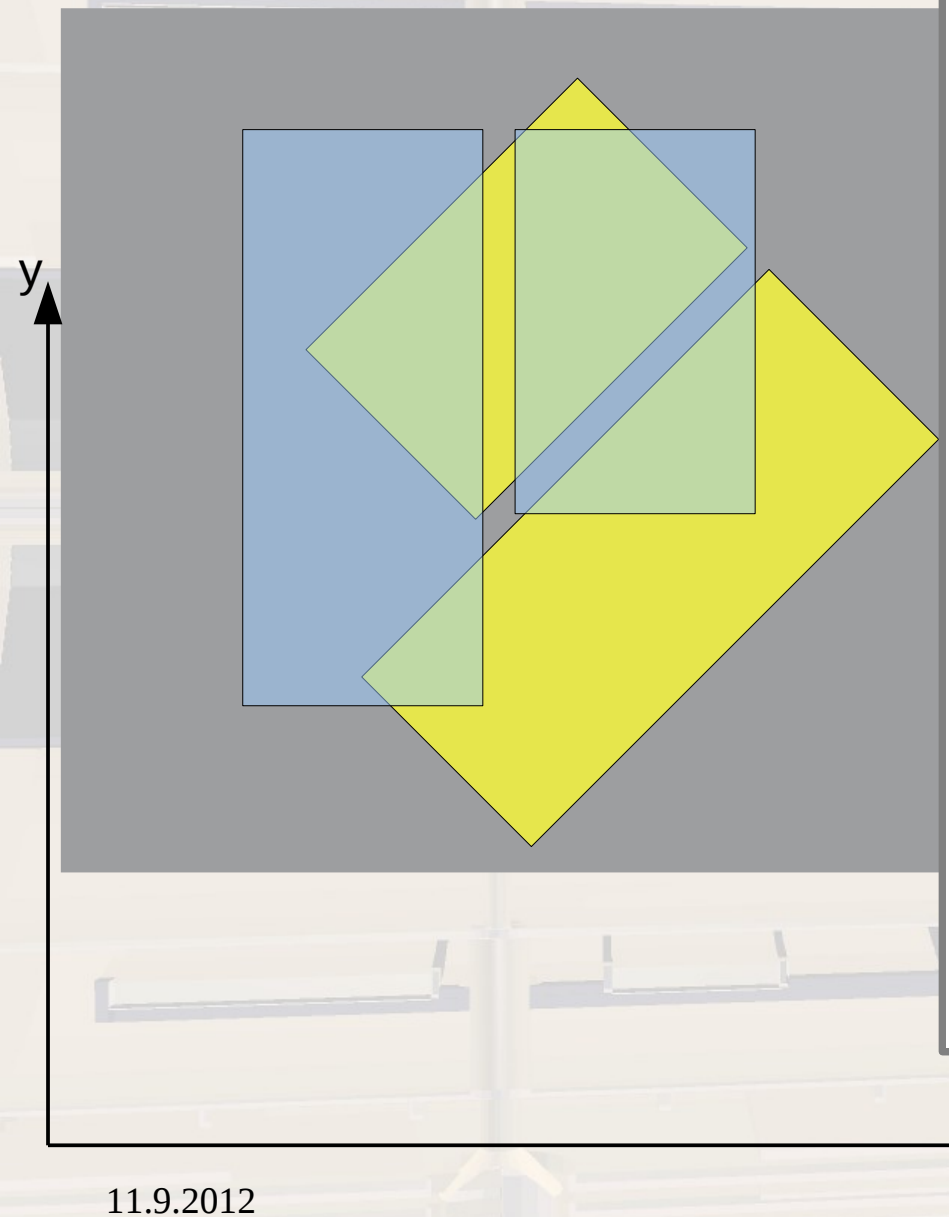
# Lumi Software before Pixels

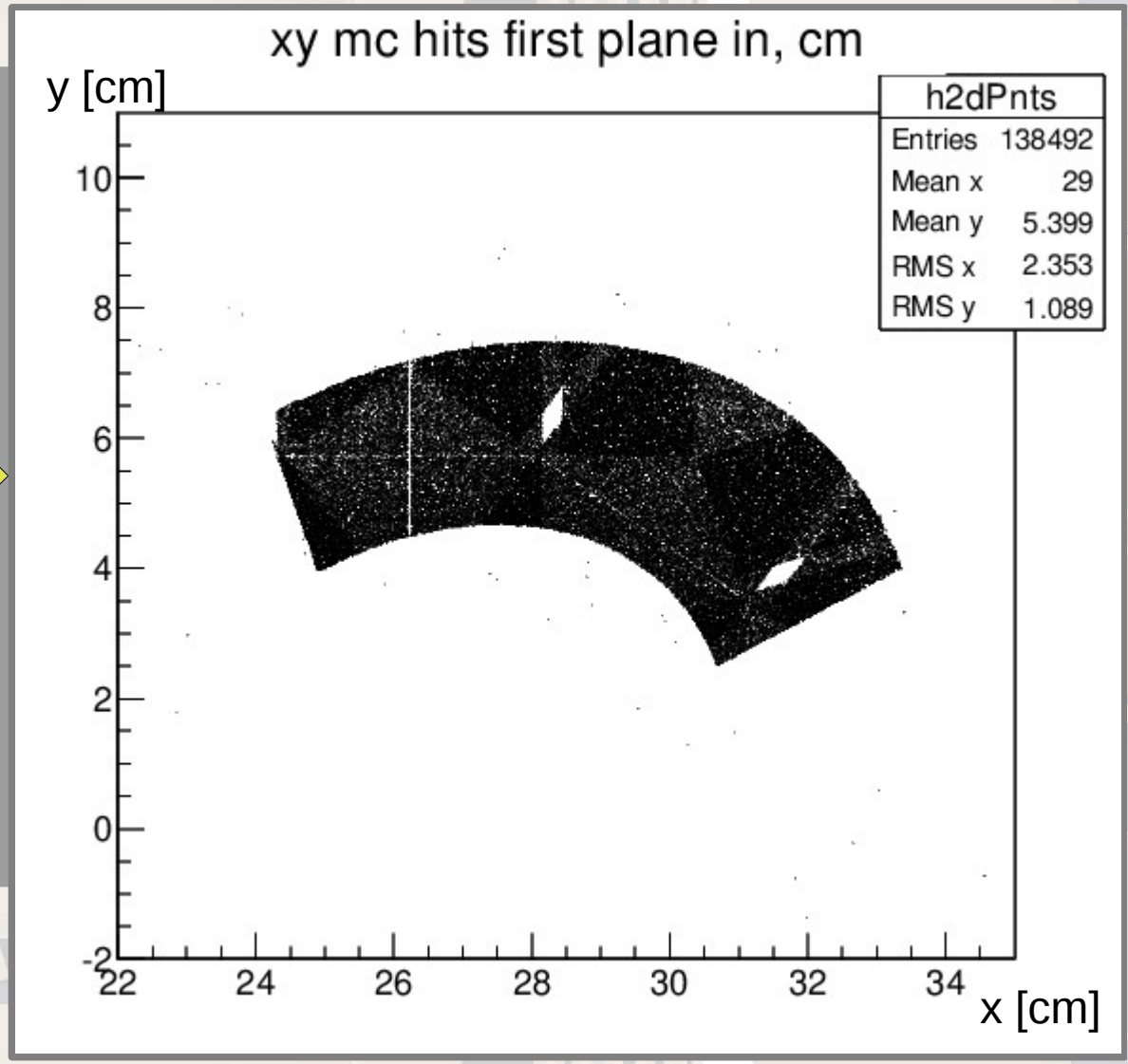


# New Lumi Design with Pixels

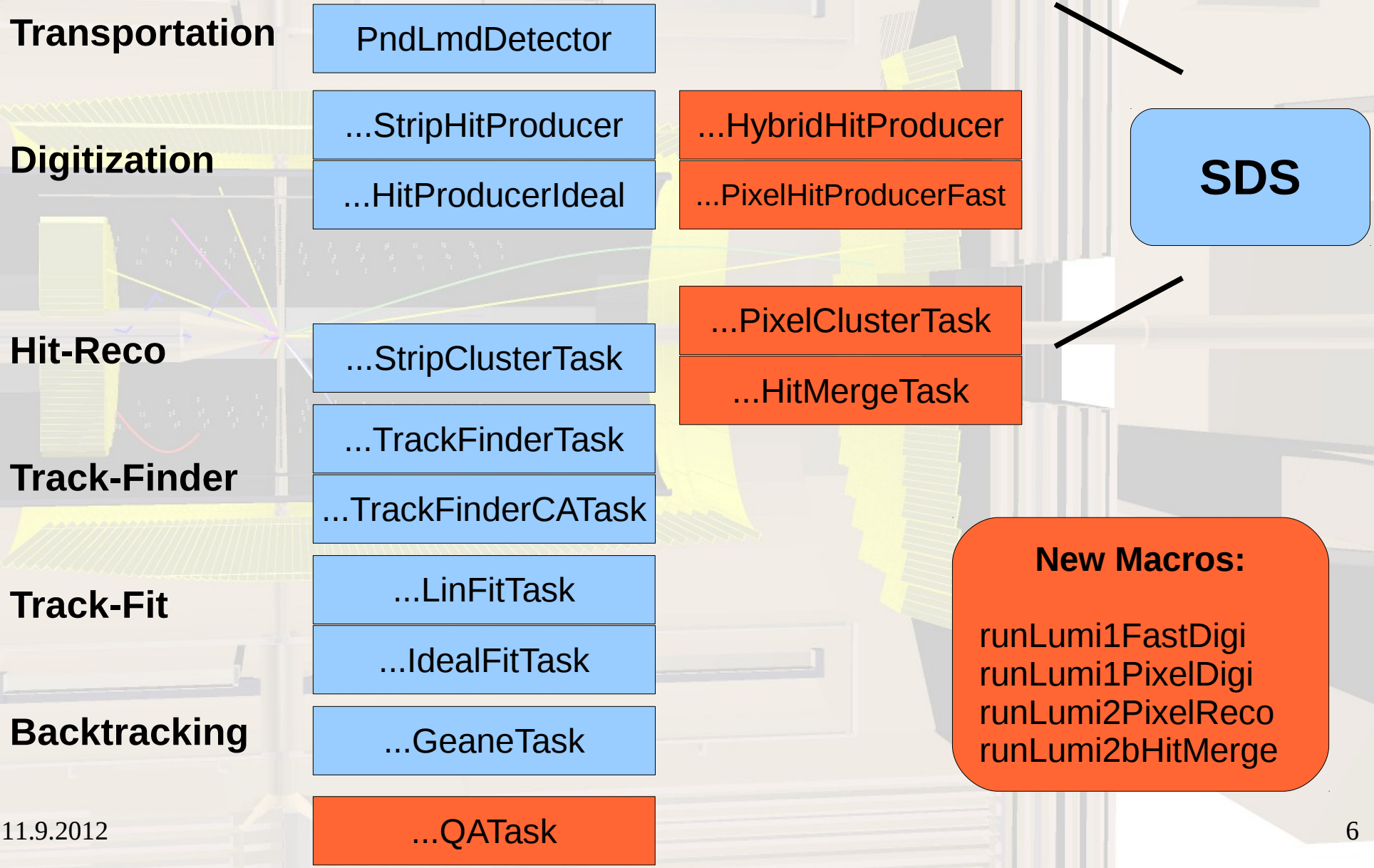


# New Lumi Design with Pixels



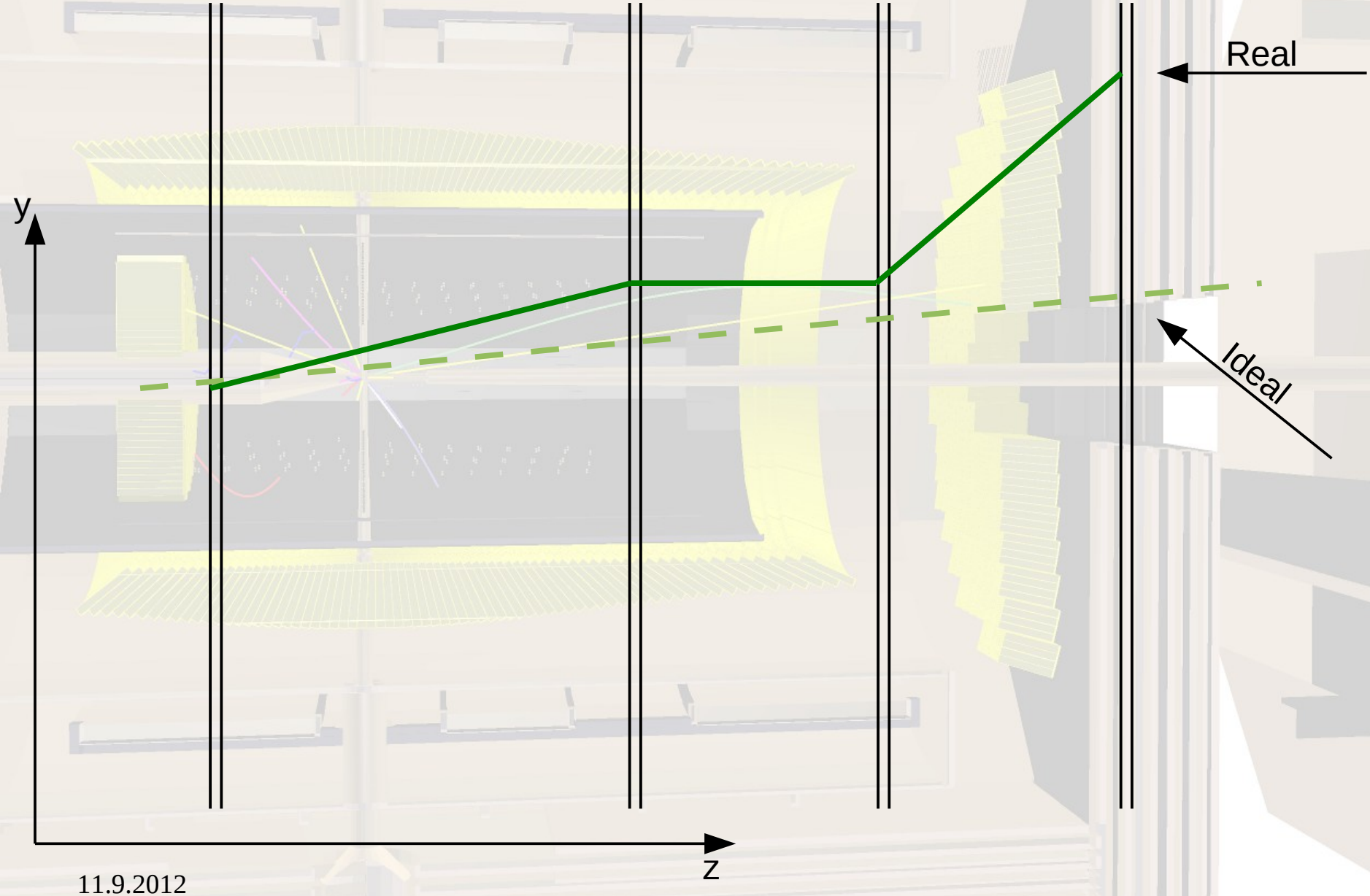
# Lumi Software Update



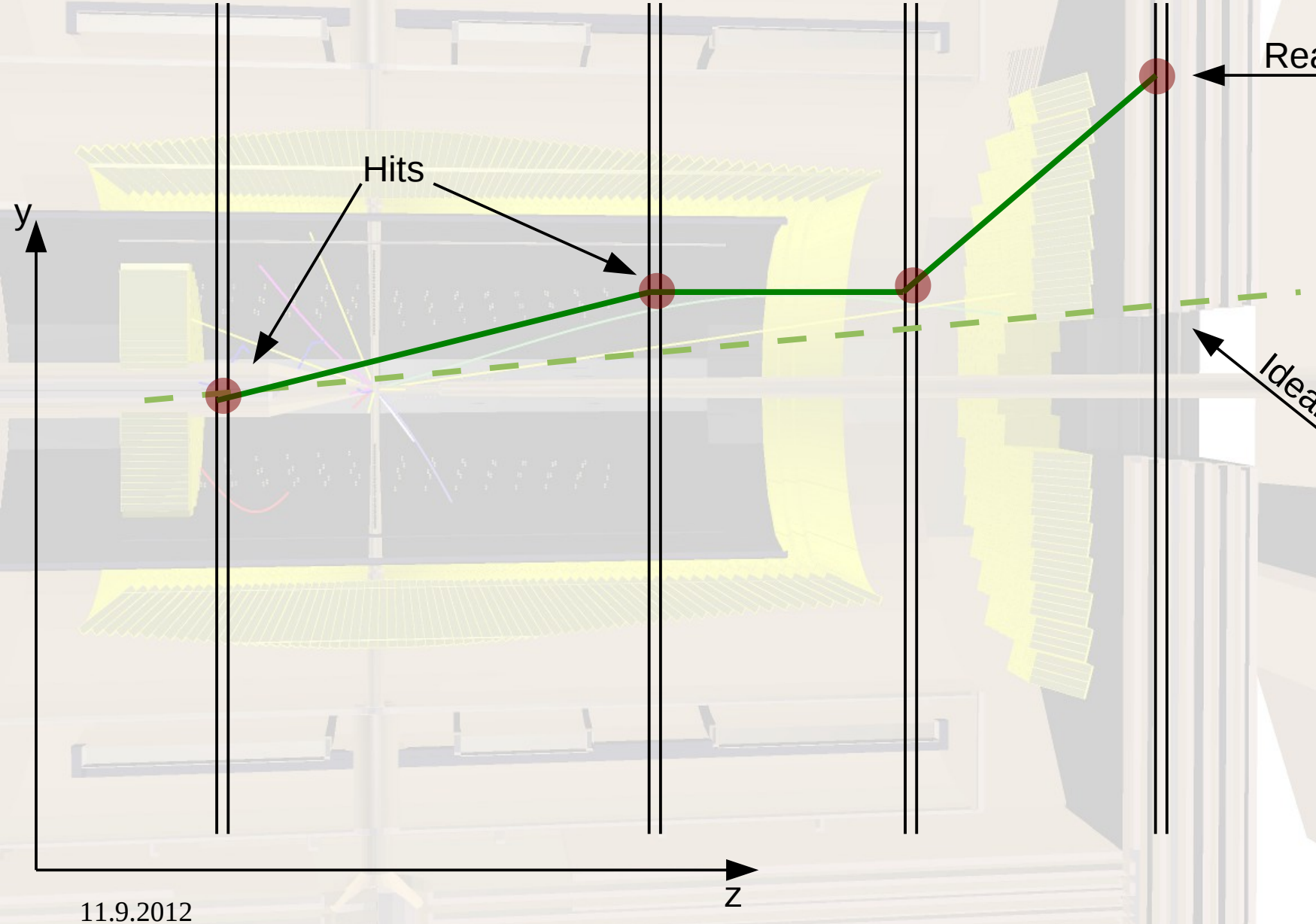

**New Macros:**

runLumi1FastDigi
runLumi1PixelDigi
runLumi2PixelReco
runLumi2bHitMerge

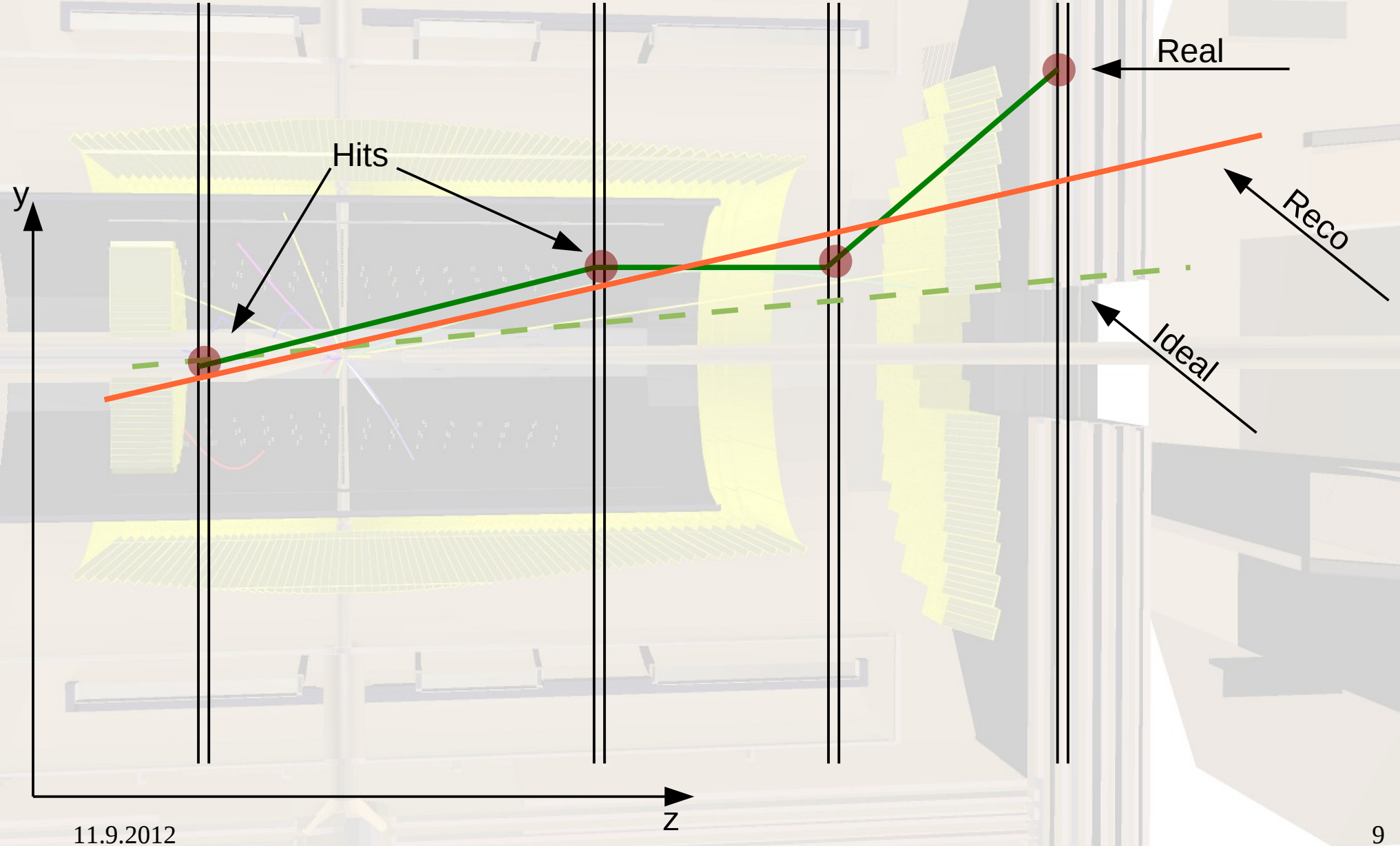
# Pandaroot LmdPixel Update



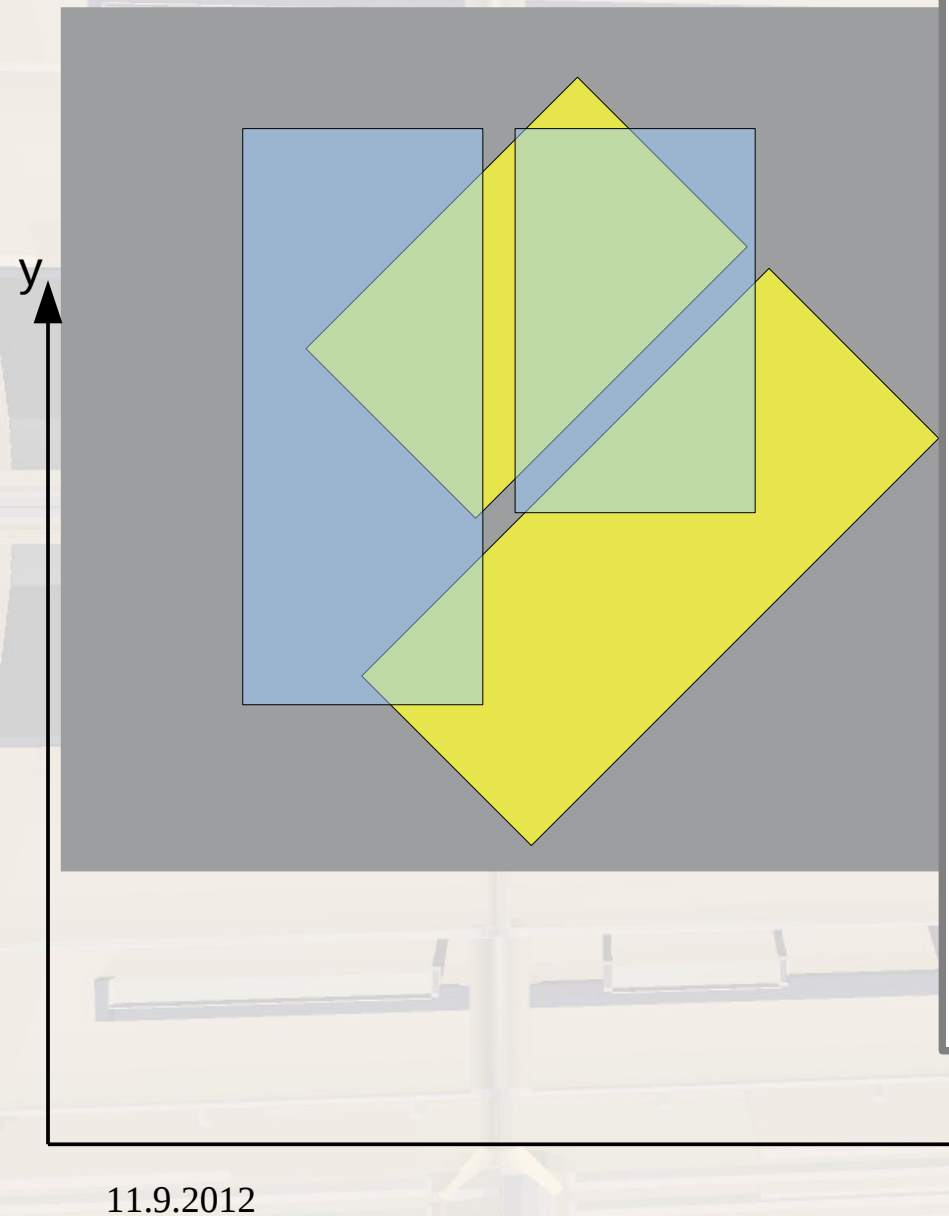
# Pandaroot LmdPixel Update

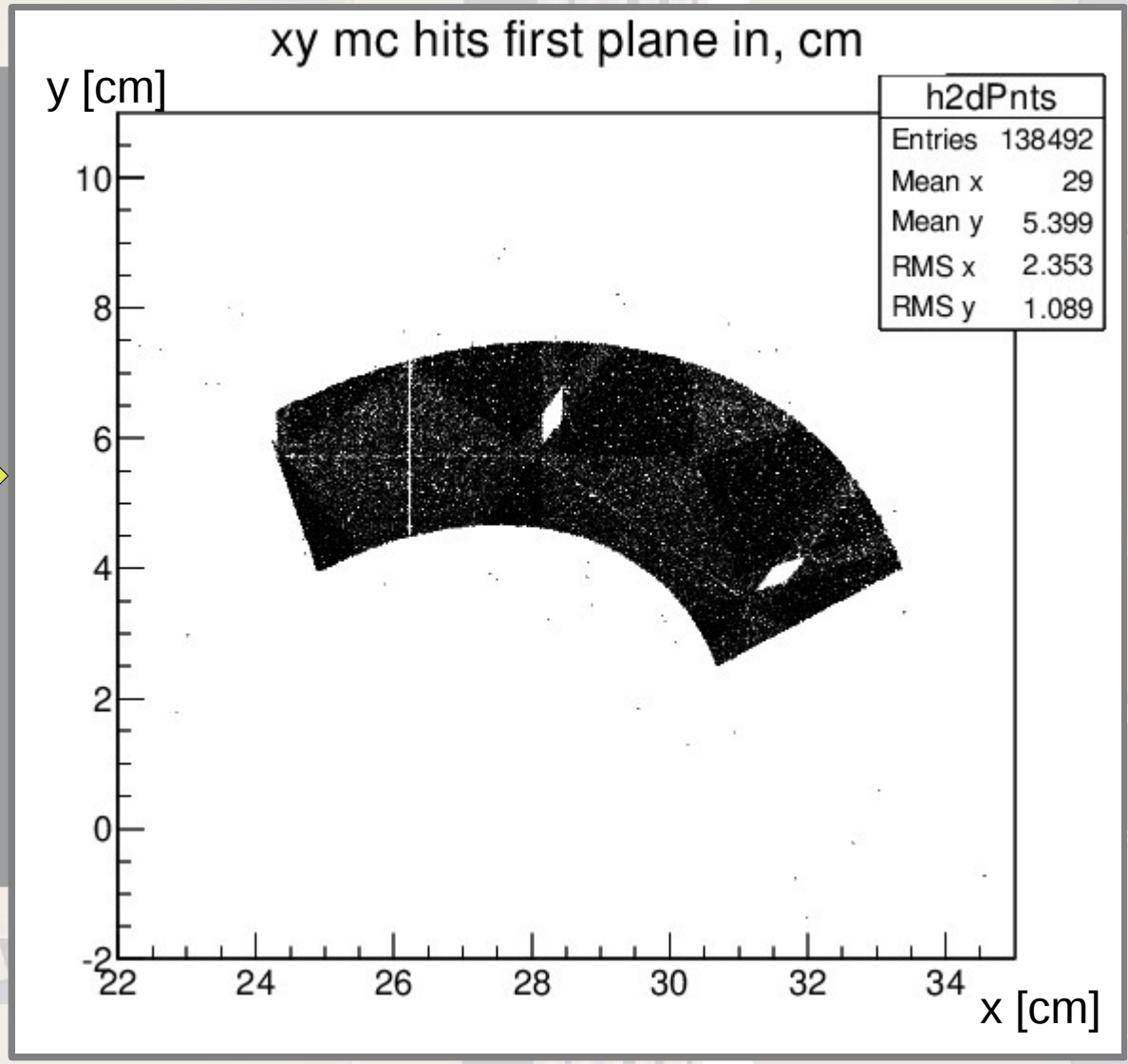


# Pandaroot LmdPixel Update

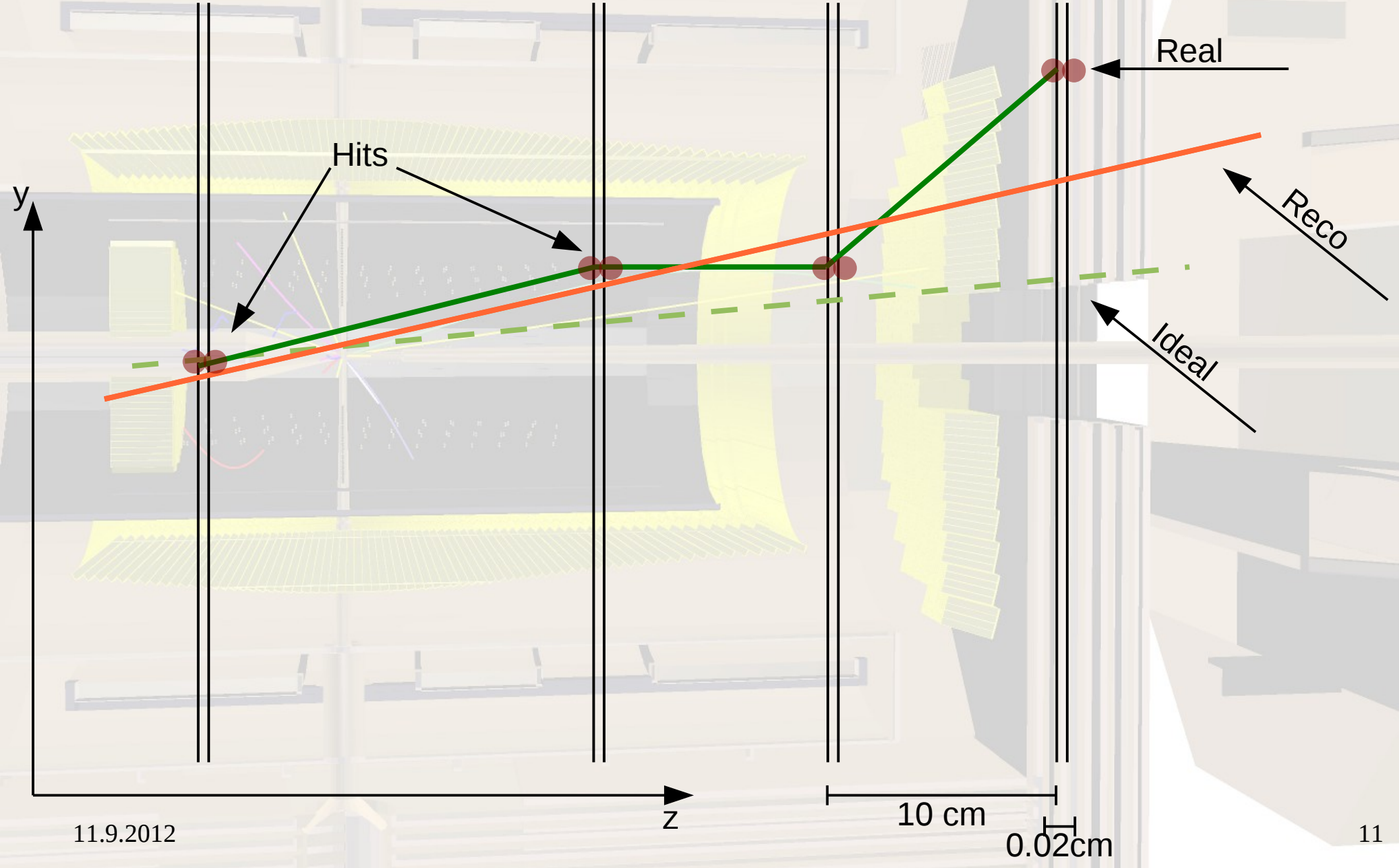


# Pandaroot LmdPixel Update

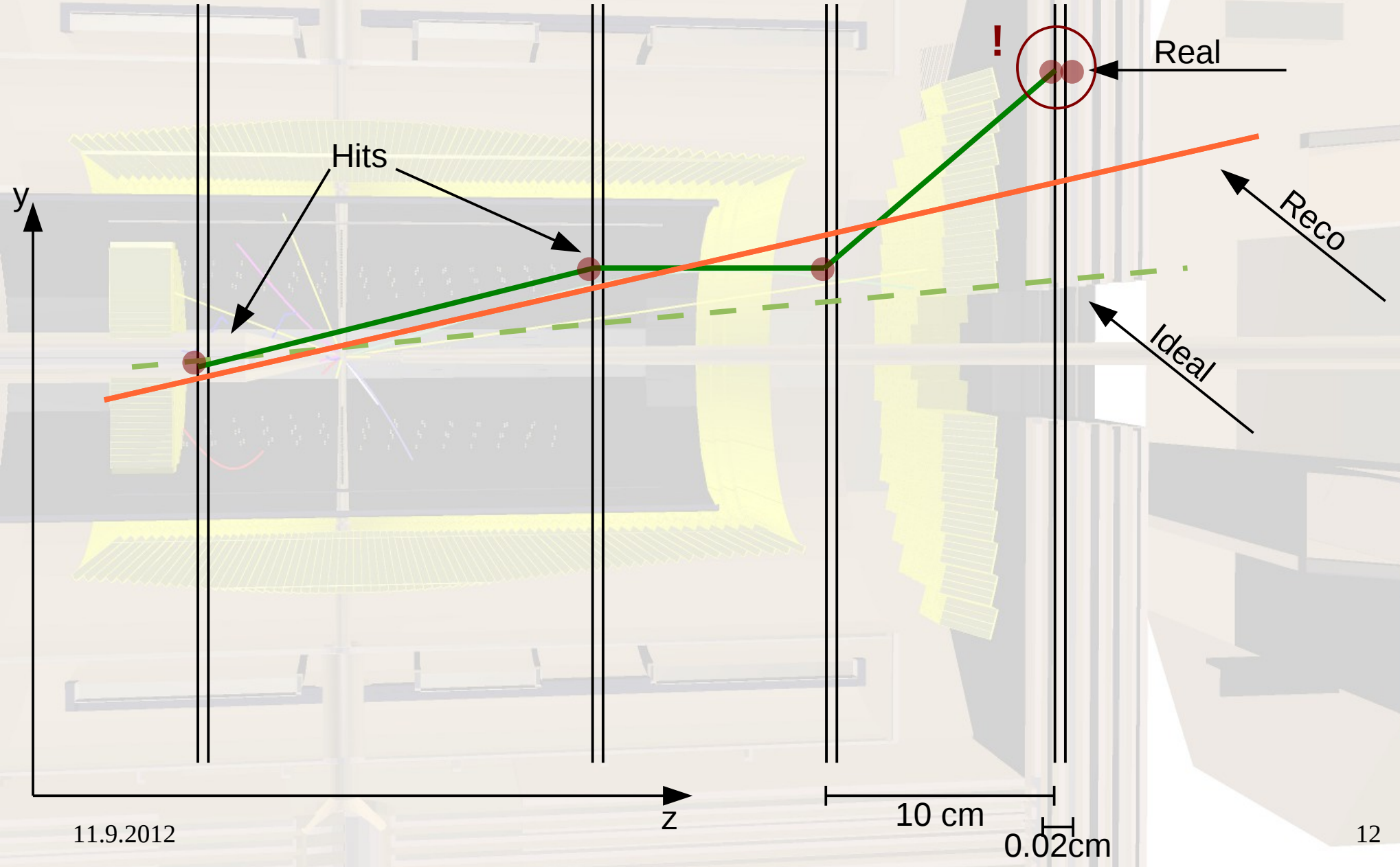


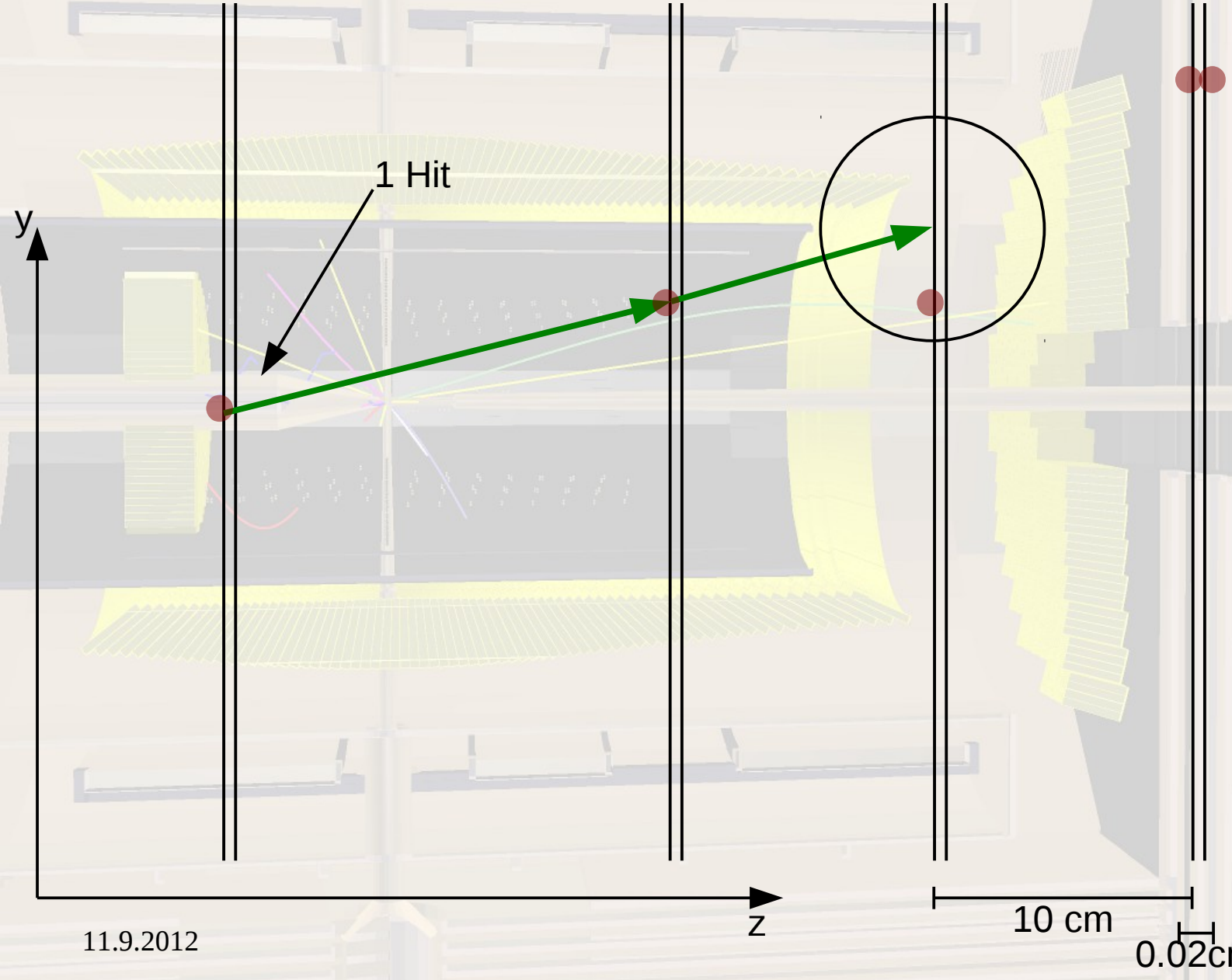
# Pandaroot LmdPixel Update



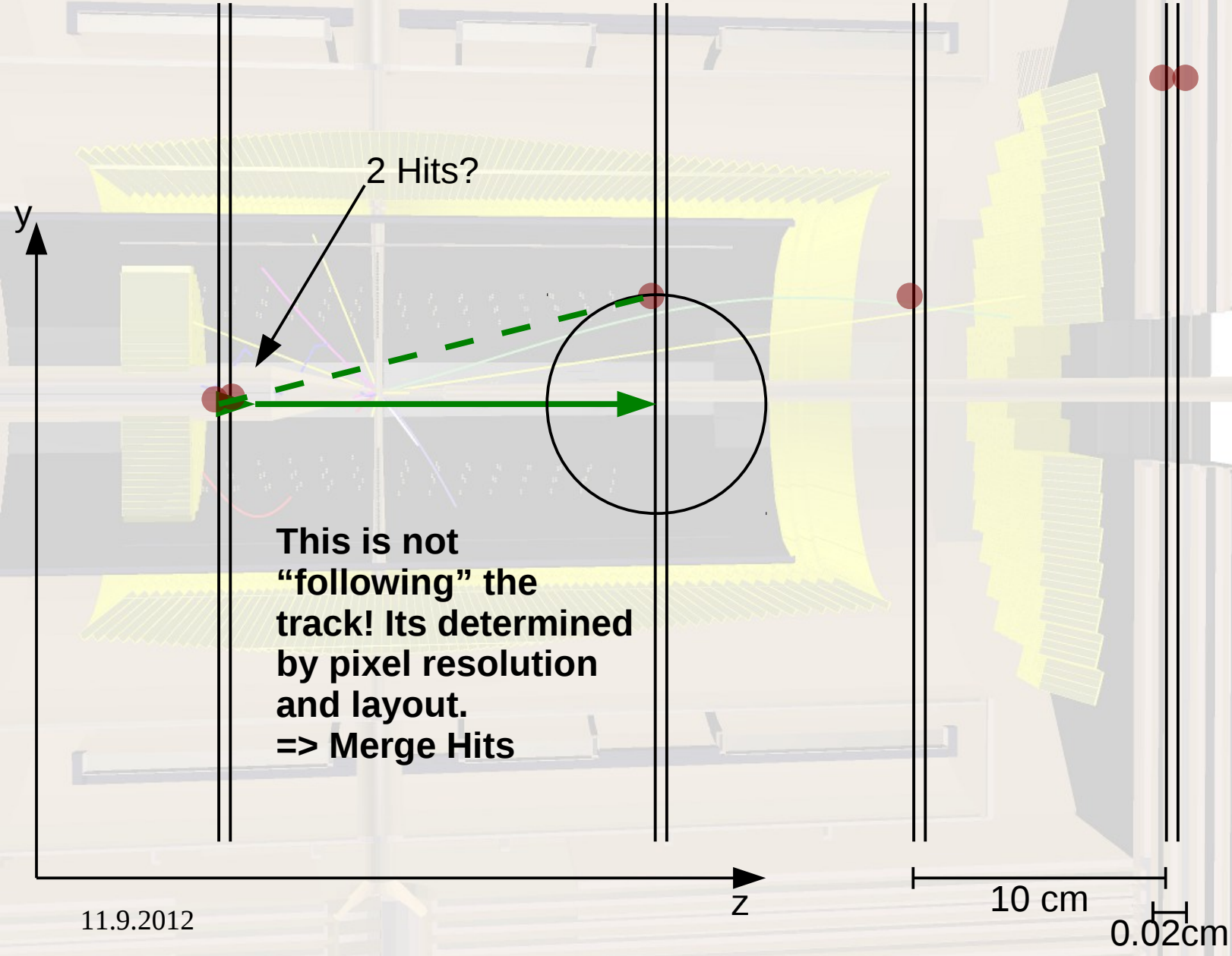
# Pandaroot LmdPixel Update



# Effect on Track-Follower



# Effect on Track-Follower

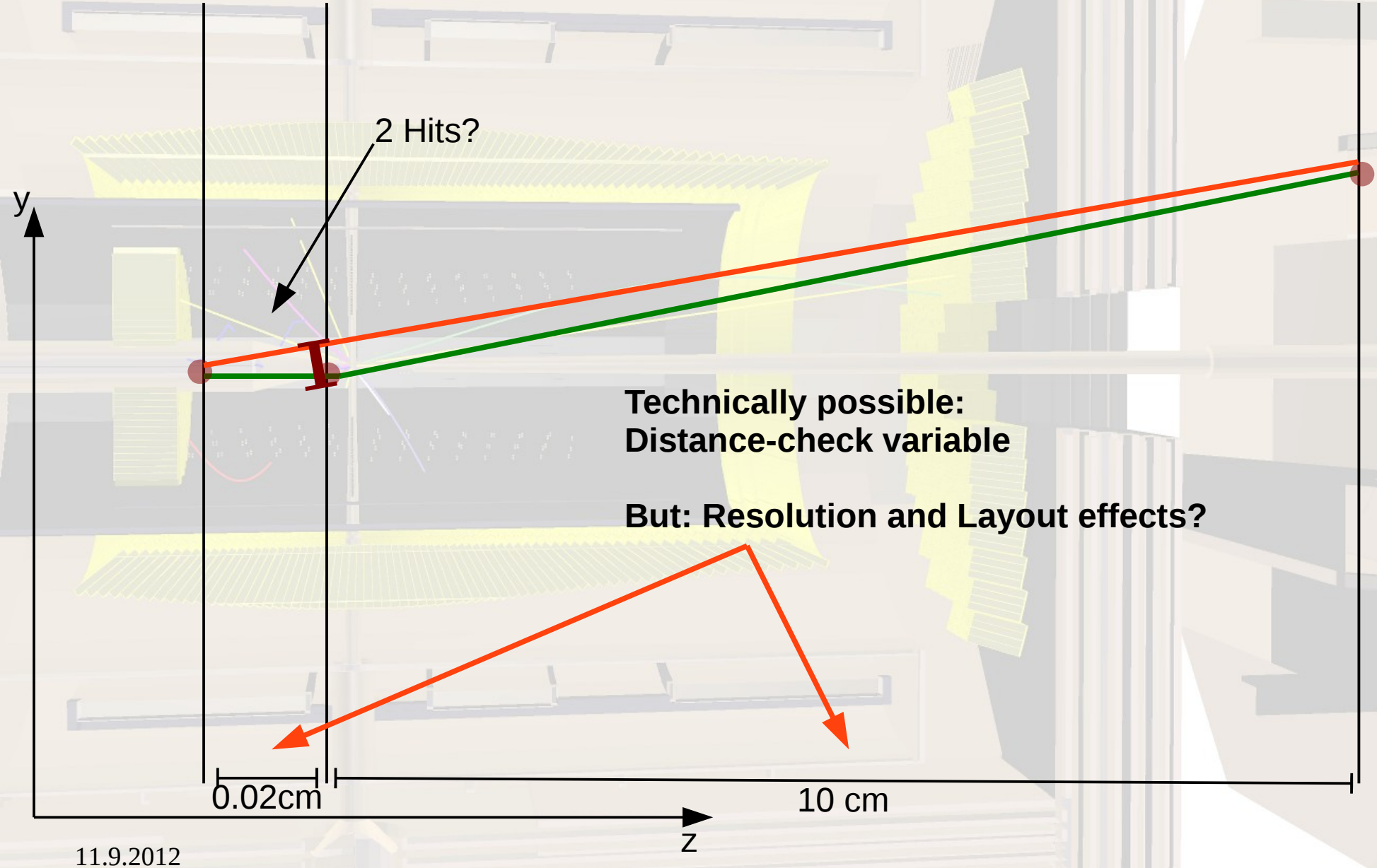



**This is not "following" the track! Its determined by pixel resolution and layout.**
**=> Merge Hits**

Hit-Merge:

- Loop hits in plane
- Check for other hits close by
- Mean Position

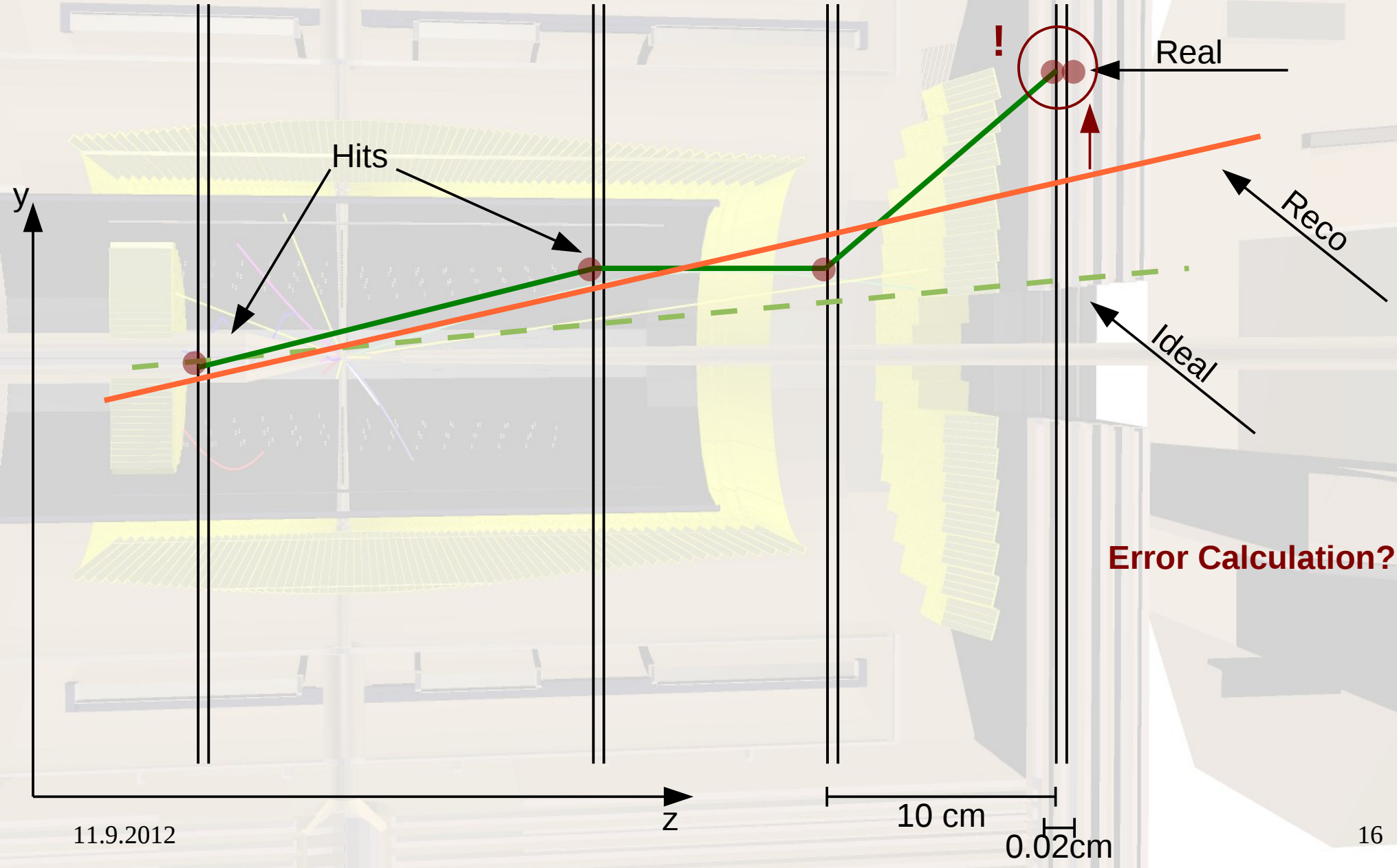
# Effect on Cellular Automaton




**Technically possible:**
**Distance-check variable**

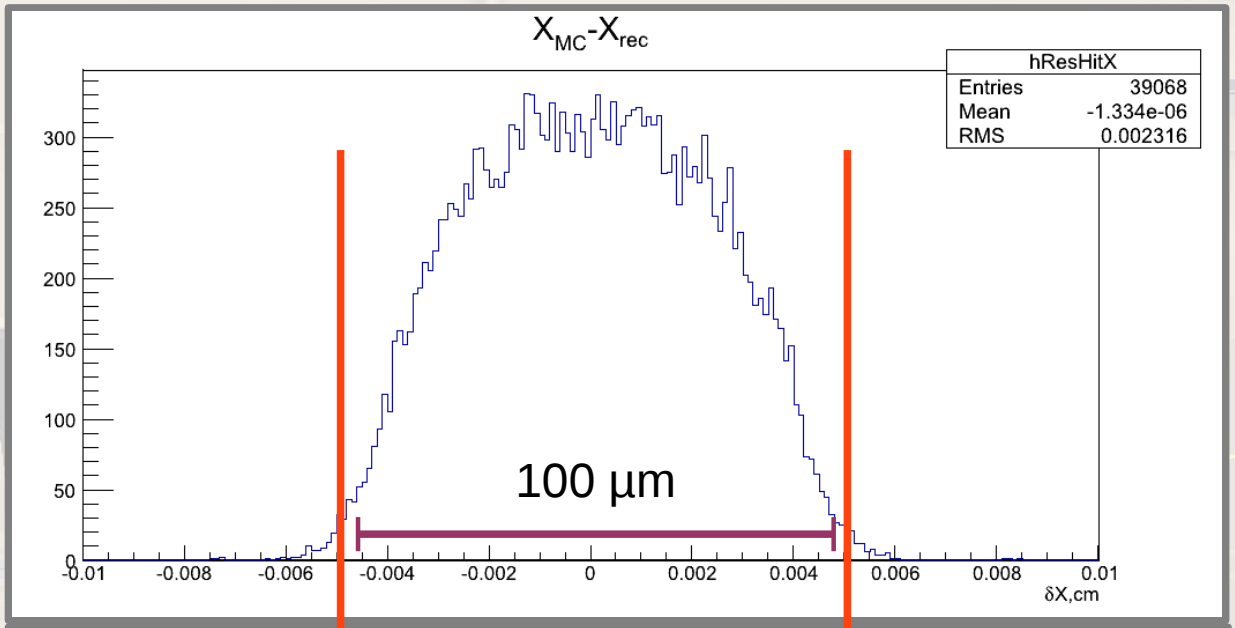
**But: Resolution and Layout effects?**

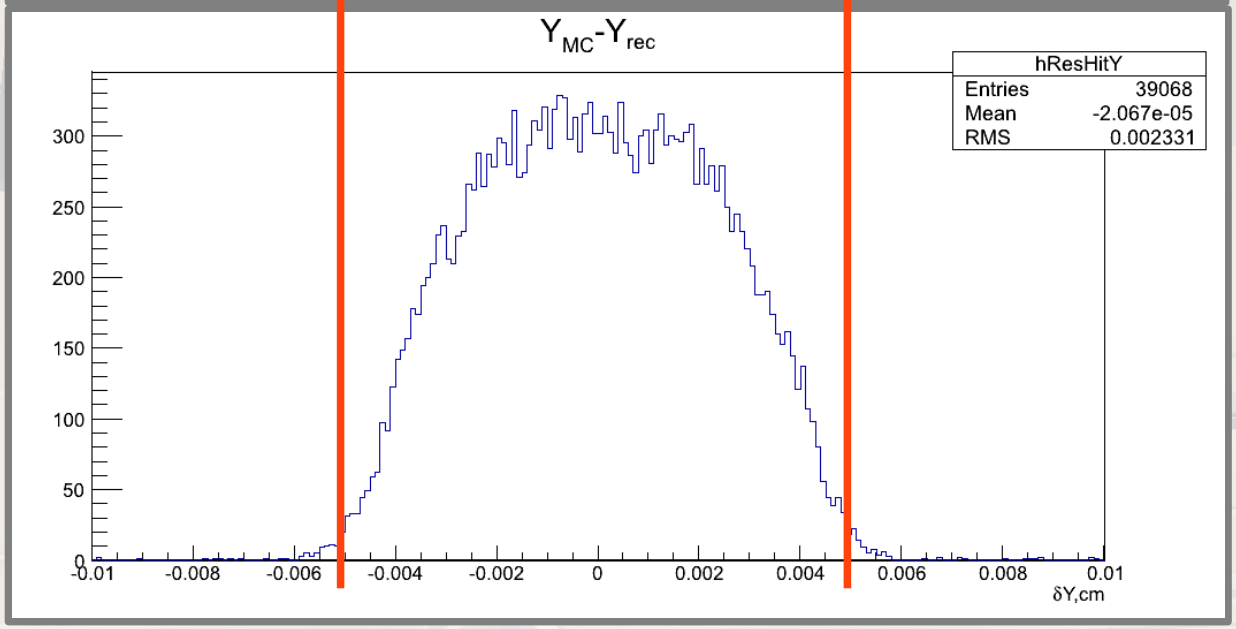
# Effect on Trackfit

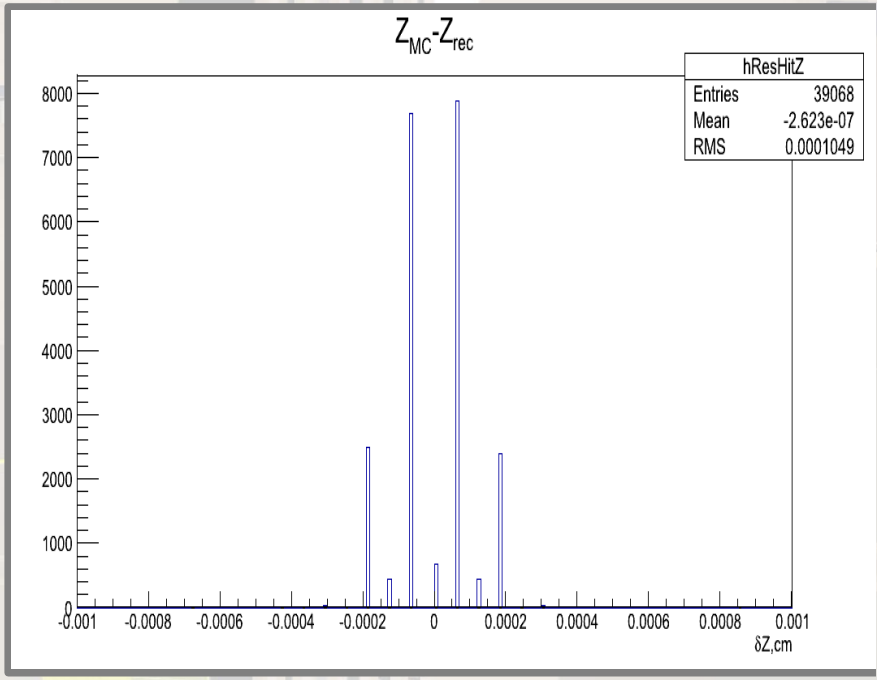



**Error Calculation?**

# Hit-Resolution



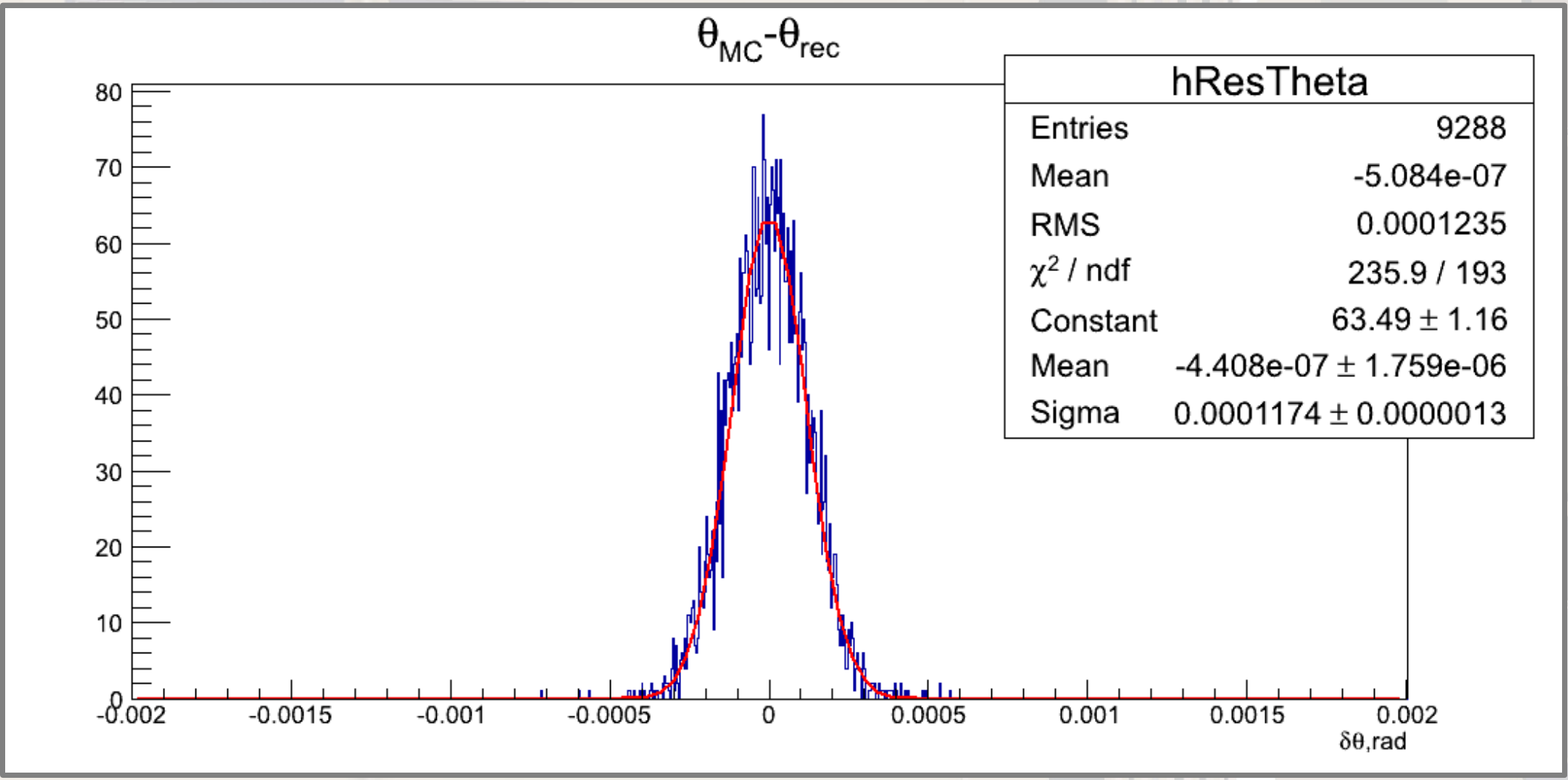




4 µm

Dipole rotation
=> Backmapping to
local Lumi coordinates

# Polarangle-Resolution



# Outlook

**Hit Reconstruction:**

- Pixel Reco in local coordinates

**Tracking:**

- Update Cellular Automaton
- Test “multiple hit per plane” search
- Test “multiple hit per plane” fitting
- Check more sophisticated trackfit like Kalman

**General:**

- Optimize Pixel-Design
- Comparison with Strip-Design
- Check effects of hole in acceptance