Status of HLT b-tagging with PU-50

J. Olzem, M. Aldaya (DESY)

Upgrade simulation technical meeting, 25.7.2011

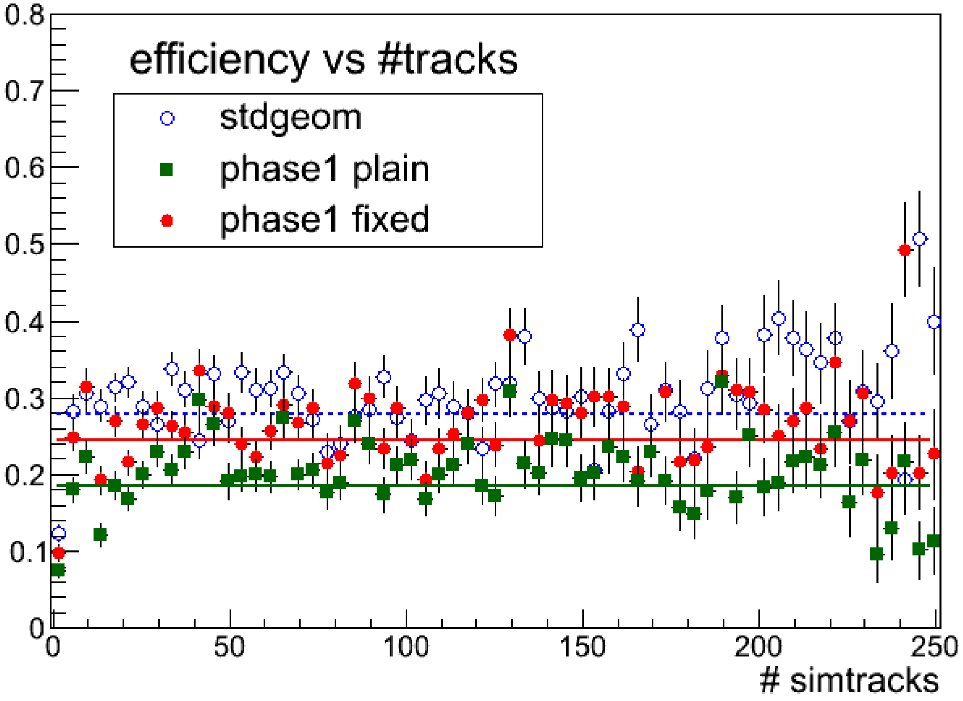
Divisive vertex finding in phase1

Found a problem with DVF vertexing (current hlt standard algo) in phase1

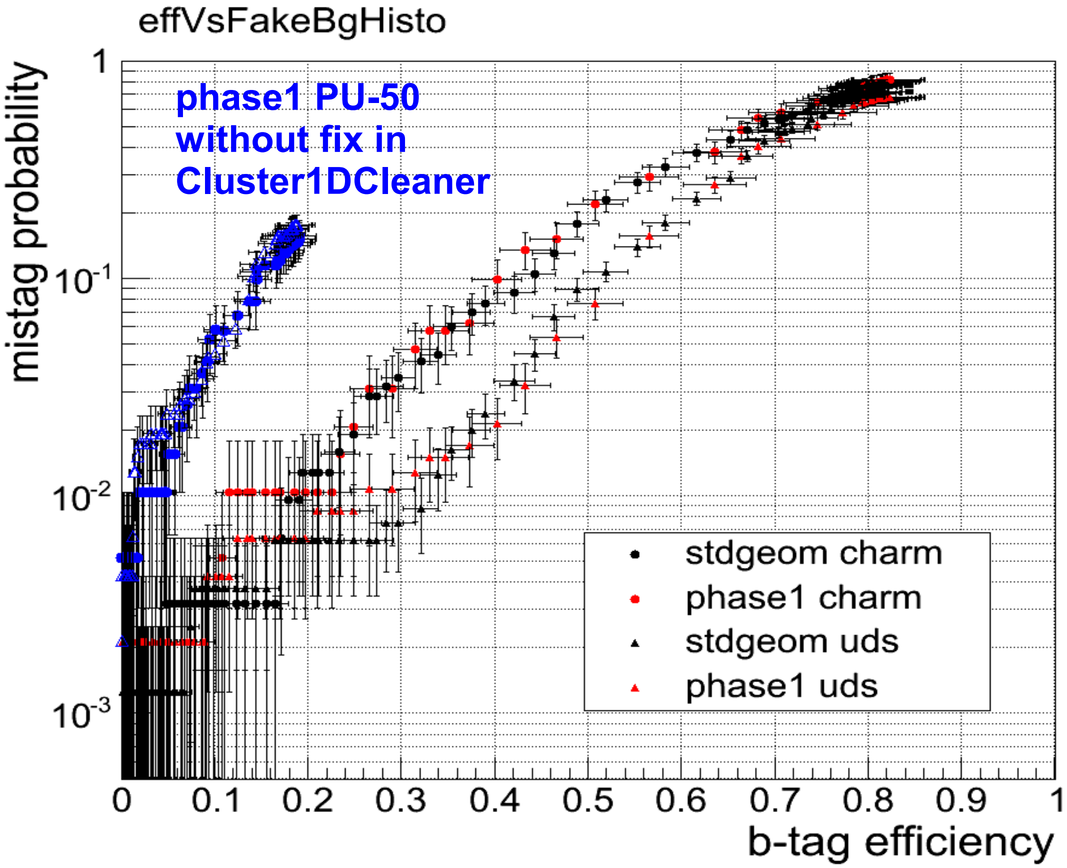
- sometimes (<1%) the upgraded pixel tracking code (new circle fit) cannot determine the track properties, e.g. **dz = nan**
- reason for this are **infinitesimal hit position errors** (see later slide)
- routine in DVF algorithm **rejects a vertex** if it contains such a track `Cluster1DCleaner<T>::average` in: `RecoPixelVertexing/PixelVertexFinding/interface/Cluster1DCleaner.h`
  → implemented a nan check as workaround

- vertexing efficiency raised by approx. 5% (almost level of stdgeom)

- vertices from hard interaction have many tracks
  → high probability of being discarded
  Effectively, this filtered ttbar vertices!

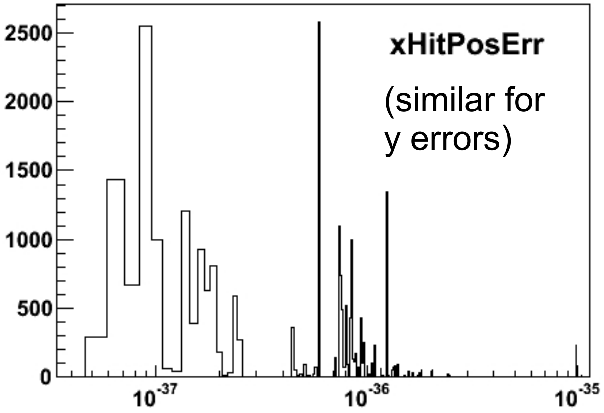


Phase1 PU-50 b-tagging performance



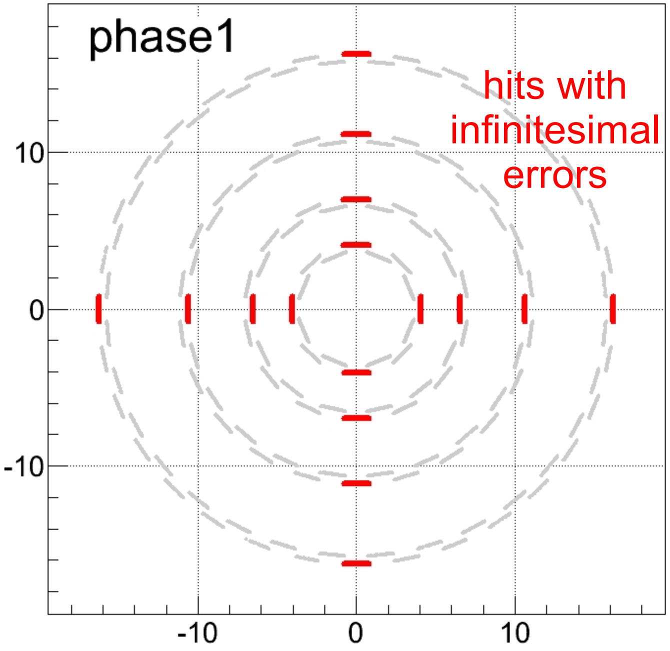
- fixing the vertexing brings phase1 b-tagging performance from almost 0 to same level as stdgeom
- again no improvement with upgraded geometry ...

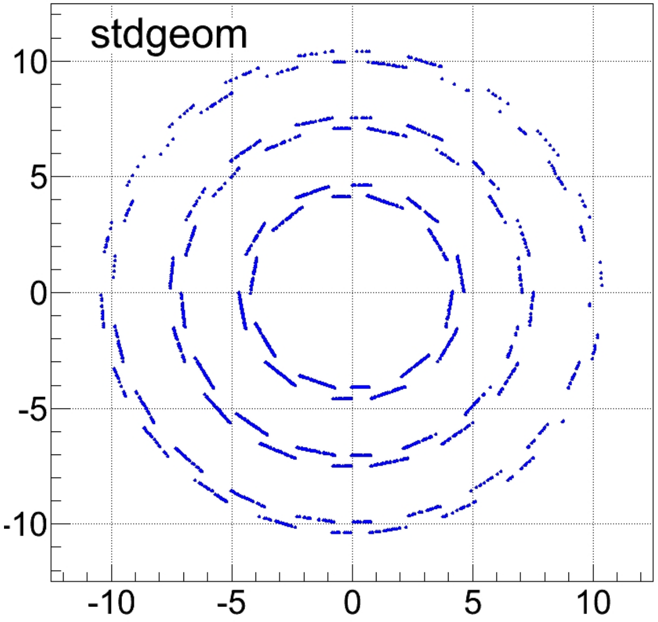
Infinitesimal PIX hit position errors



Phase1: pixel hits with infinitesimal x or y position errors

→ hits on modules which are parallel to x or y axes





- seems that the code projects localPositionErrors on the x and y axes via sinφ and cosφ – sensors parallel to x or y → unphysical values
- no such effect in stdgeom since no modules are exactly parallel to x or y
- will locate the code and see if that can be fixed