Status of TPC simulation and more

Jens Wiechula

Nov. 9 2017
ALICE offline week

Outline

- News
  - Removing TClonesArray depencence
  - Energy loss treatment
  - Configuration and parameter classes
  - Calibration classes
  - MC labels
  - TPC processing executable
  - Time frame simulation
- Next steps
  - Electron transport
  - Data formats
  - Data processing
- General discussions
- Summary

TClonesArray dependence

- Improvements on hits / digits / clusters (Sandro)
  - All now stored in std::vector instead of TClonesArray
    - Code optimisation: no dynamic casts etc.
  - Store and process hits sector-wise (smaller memory footprint)
  - Remove TObject dependence of digits
  - Remove cluster dependence of FairTimeStamp + make non-virtual

Energy loss treatment

Current status

- Ported latest modifications from AliRoot to O^2
  - Unified treatment in Geant 3 and Geant 4
  - Energy loss forced in step manager
  - Fewer steps (only every 2mm, not each collision)
  - → Faster and less memory and disk usage
  - Reduction in size of hit file
    - G3: 6.3MB -> 2.5MB
    - G4: 1.6MB -> 1.5MB
- Next steps
  - Understand difference in G3/G4 (physics processes, e.g. secondary production?)
  - Detailed comparison to AliRoot
  - Further optimisation in hits container?

Energy loss treatment

Future steps

- Increase step size further and move treatment to digitisation code (create electrons on the fly)
  - Faster processing in geant
  - Better control for tuning
  - Even smaller memory and disk usage

Configuration and parameters

Current status / next steps

- Configuration and Parameter classes introduced which are used to steer the simulation

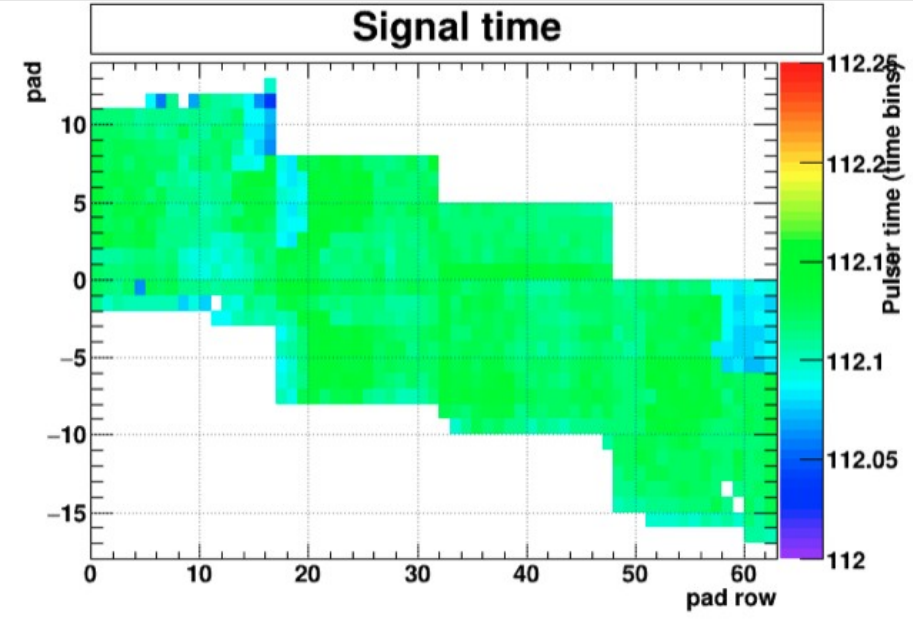
```
ParameterDetector
ParameterElectronics
ParameterGas
ParameterGEM
```

- Next steps
  - Develop TPC specific CCDB interface
  - Save and load from CCDB

Calibration classes

Current status / next steps

- Basic calibration classes (Pedestal, Pulser) implemented
- Already used for test beam analysis and in electronics lab

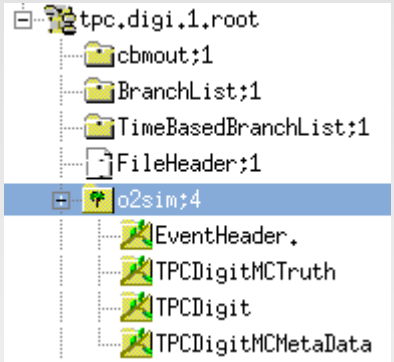


- Next steps for 2018
  - Gain variation and calibration
  - Space-charge correction

MC labels

Current status / next steps

- Labels are implemented in hits and digits
- Hits are stored on a track level → no redundant label information
- Digit labels in a separate branch
  - Separation from actual Digit information
  - using o2::dataformats::MCTruthContainer<o2::MCCompLabel>
- Next steps
  - Implementation on clusters being reviewed
  - Afterwards: implement for tracking



Running the TPC code

Simple executable

- A simple executable exists to run the different stages of the TPC code:

```
runTPC -h
Allowed options:
  -h [ --help ]                    Produce help message.
  -m [ --mode ] arg (=sim)         mode of processing, "sim", "digi", "clus",
                                   "track" or "all".
  -n [ --nEvents ] arg (=2)        number of events to simulate.
  -e [ --mcEngine ] arg (=TGeant3) MC generator to be used.
  -c [ --continuous ] arg (=1)     Running in continuous mode 1 - Triggered
                                   mode 0
```

- Simple box generator with 10 pions
- With **runTPC -m all** everything is run consecutively

Time Frame Simulation

Set up

- Use simple macro to produce 100 min bias Hijing events using AliRoot (Hijing not yet implemented in O^2)
  - Stored in Kinematics.root
- Use functionality implemented by Sandro to use Kiniematics as input for transport in O^2
  - o2sim -m PIPE ITS TPC --extKinFile SimKinematics.root -n 100 -g extki
  - Use only one TPC sector to minimize computing time and disk space
  - → Segfault after 72 events, to be checked!
- Use 14 * 72 events ~ 1000 events ~ 1 TF
- Hack realistic bunch structure in TPC digitisation
- Run TPC digitisation

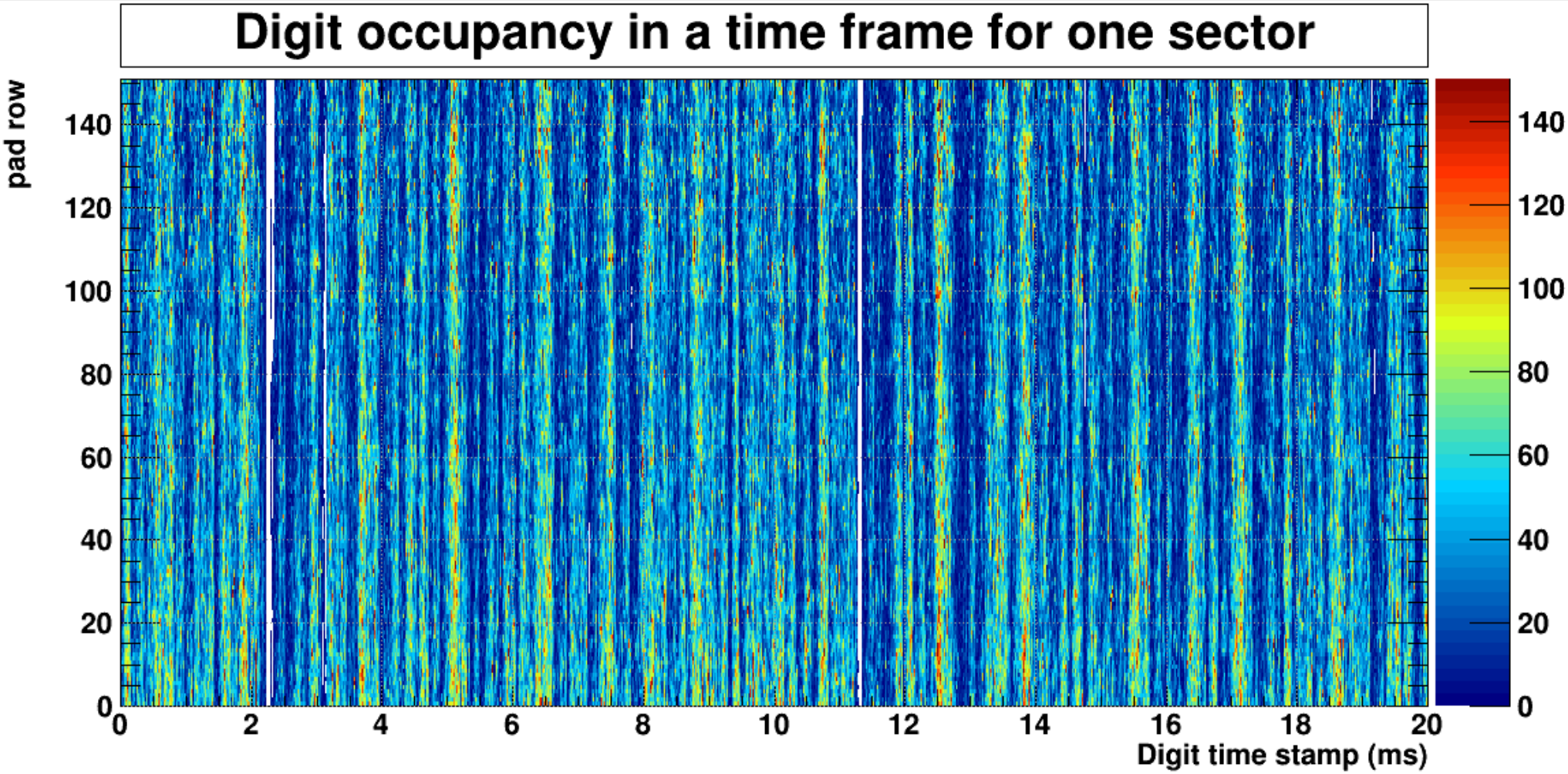
Time Frame Simulation

Results

- File Sizes
  - SimKinematics.root: 21M for 100 min bias Hijing
  - o2sim.root: 234M for 72 events, hits from only one sector
  - tpc.digi.root: 1,8G for ~1000 event, only one sector!
- Critical look to data classes and format needed

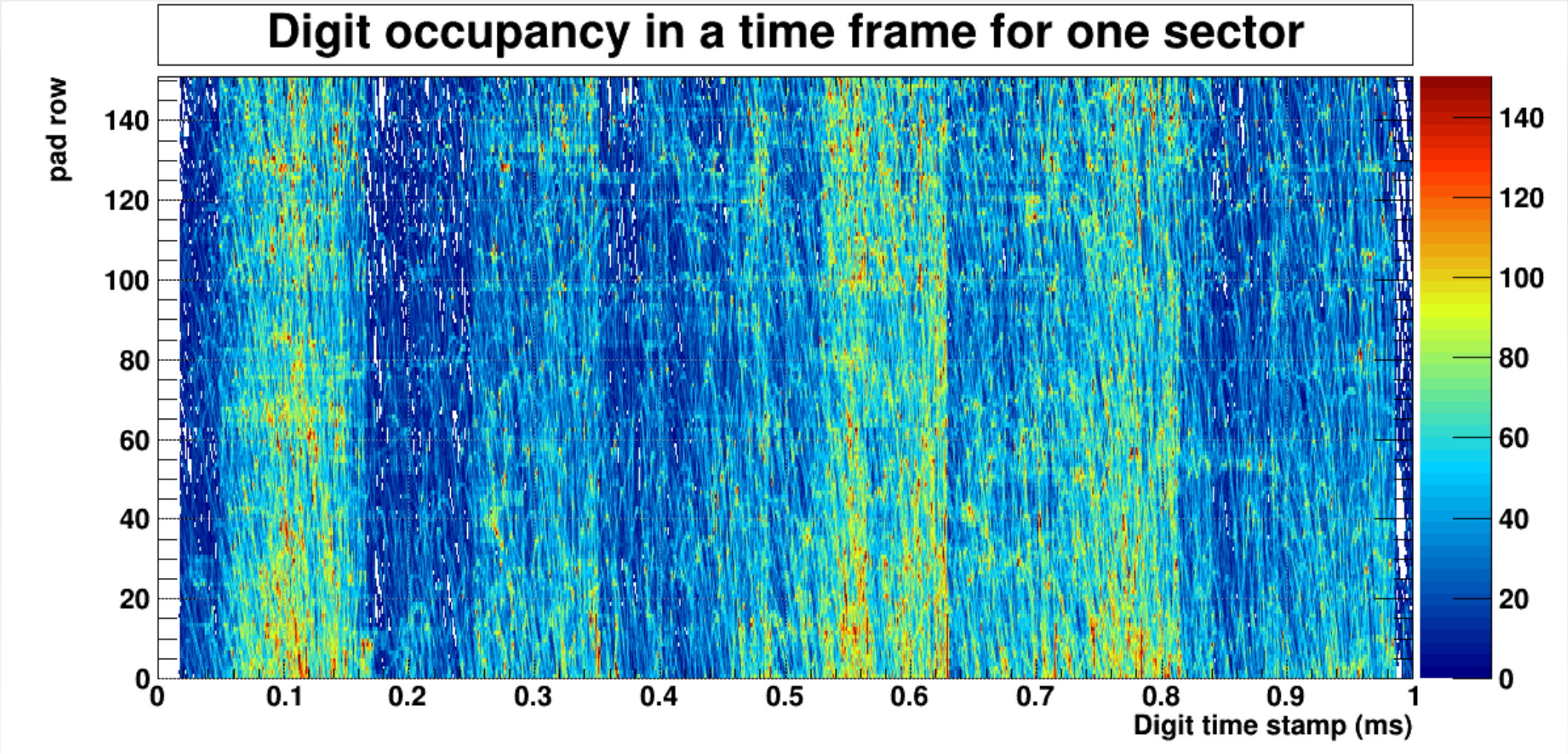
Time Frame Simulation

Visualisation



Time Frame Simulation

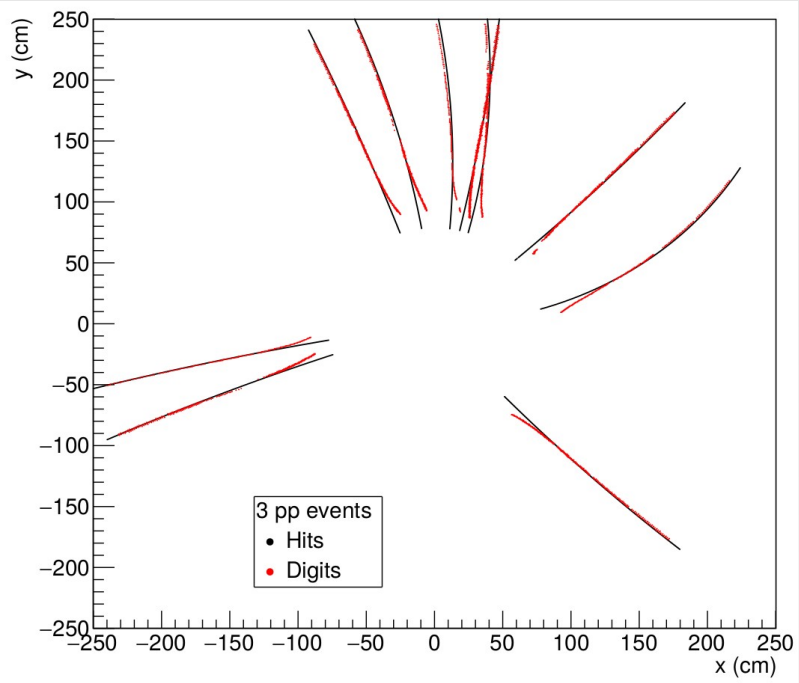
Visualisation



Next steps

Electron transport

- Ionisation electron transport only includes diffusion and attachment
- Important effects missing
  - ExB
  - Distortions
- Static distortions implemented in a private repository
  - Code should be used both in AliRoot and O^2
  - Build standalone library as for CA tracker?
- Dynamic distortions will be large effort



Next steps

Data format

- Data formats are being defined
  - Raw data
    - → decoded GBT frames, sorted in a natural way
  - Zero suppressed raw data (requires baseline correction)
    - Present FEE format + exchange time and pad direction
    - Or Bitmask for fired pads + raw ADC values
  - Clusters: format(s) proposal by David

    https://indico.cern.ch/event/678423/contributions/2778515/attachments/1553617/2442101/2017-11-07_HLT_TPC_Hands_on.pdf

    - Final property estimates and formatting of HW cluster from CRU will be done in an o^2 device
- Integrated digital currents (used for distortion corrections)
  - Will be calculated in the same device as above
  - To be defined
    - How to inject into the data stream and distribute over EPNs
    - Data format

Next steps

Data processing

- Up to now Simulation and reconstructed data stored in root trees
- Where to change from root based data to future online data formats?
  - Most natural for digitised data
  - Common format for simulated and real data
- Move parts of the simulation chain from ROOT based tasks to O^2 (FairMQ) devices (Taku)

Next steps

Even more

- Stress test with central Pb-Pb event
- Benchmark AliRoot against O^2
  - Possibility to use particles created in AliRoot as input for O^2 implemented (Sandro)

General discussions

Physics cut

- Physics cuts for geant
  - Global cuts set in Detectors/gconfig/SetCuts.C:
  - Dector material specific cuts in AliRoot treated globally in `data/galice.cuts`
    - Only for G3?
    - How about G4?
  - Where to do it in O^2?
    - Handled by the detectors individually?
    - Detector::SetSpecialPhysicsCuts() → called by FairRoot
    - Handled globally?

General discussions

Continuous simulation

- For continuous simulation it might be good to re-use hits
- It is important to simulate a realistic event time distribution (bunch structure)
- Develop custom file source inheriting from FairSource (similar to FairFileSource)?

Summary

- Full simulation chain running in O^2 hits → tracks
- MC Labels implemented for hits and digits, clusters and track will follow soon
- Digit time frame with min bias Hijing events and realistic bunch structure simulated
- Implementation of distortions in electron transport to be started as soon as possible
- Milestones mostly on track
  - need better definition for framework integration and QC
- Next challenge: Full continuous Pb-Pb simulation → started using one sector

Backup

Status overview / Milestones

Simulation

Type	Simple Implementation (Step 0)	Extended Implementation (Step 1)	Close to final (Step 2)
Pad plane mapping (Jens Wiechula)	✓		
Geometry and materials (Jens Wiechula)	✓ (AliRoot copy)	Q1 2018	
Step manager (energy loss) (Jens Wiechula / Peter Christiansen)	✓ (realistic energy loss)	Q4 2017 ✓ (synchronize AliRoot / O^2)	Q1/Q2 2018 (speed optimisation)
Microscopic Transport (Andreas Mathis, Ernst Hellbär)	✓ Diffusion/attachment	(✓) Q3/4 2017 Static distortions	Q1/2 2018 Dynamic dist.
Digitisation (Andreas Mathis)	✓	✓ (modularized code)	(✓) Q3/4 2017 Common mode/ pad response
Continuous readout (Jens, Andreas Mathis, Sando Wenzel)	✓ Q3/4 2017	Q1/2 2018 (code optimisation)	
Hit format optimisation (Jens, Andreas Mathis, Sando Wenzel)	✓ Q2/3 2017 (minimize redundancy)	Q1/2 2018 (compression)	

Status overview / Milestones

Reconstruction

Type	Simple Implementation (Step 0)	Extended Implementation (Step 1)	Close to final (Step 2)
Clusterisation (Sebastian Klewin)	✓ (basic software version)	(✓) Q3/4 2017 (tuned software version)	Q2/3 2018 (FPGA version)
Tracking (synchronous) (David Rohr)	✓ Standalone	(✓) Q3/4 2017 Porting to O2	
Loop finder (Lipi group, Malta group)	Q3/4 2017	Q2/3 2018	2020+ commissioning

Status overview / Milestones

Calibration

Type	Simple Implementation (Step 0)	Extended Implementation (Step 1)	Close to final (Step 2)
Pedestal (Jens Wiechula)	Q2 2017 ✓ For test beam	2018	2020+ commissioning
Pulser (Jens Wiechula)	Q3 2017 ✓ For test beam	2018	
Gain (Jens Wiechula)	Q1/2 2018 ? (requires de-calibration)		2020+ commissioning
ITS-TPC-TRD interpolation (Ole Schmidt, Ruben Shahoyan)	Q3/4 2017 After synch. Tracking → Q2/3 2018		2020+ commissioning
Current scaling (Ernst Hellbär, Lipi Group)	Q1/2 2018 Together with dyn. Dist.		
Further calibrations / calibration data storage	2018		

Status overview / Milestones

Framework

Type	Simple Implementation (Step 0)	Extended Implementation (Step 1)	Close to final (Step 2)
Usage of O^2 devices (Taku Gunji)	Q4 2017 (Simulation and reconstruction)		

- Framework integration not yet well defined

Status overview / Milestones

Quality control

Type	Simple Implementation (Step 0)	Extended Implementation (Step 1)	Close to final (Step 2)
Clusterisation / compression (???)	Q1/2 2018		2020+ commissioning
Basic tracking properties (???)	Q1/2 2018		2020+ commissioning
Full reconstruction QA	For sim. Challenge?	Q1/2 2019 For data challenge	2020+ commissioning

- Tasks for Quality control not yet well defined