

Progress update on tracking efficiencies

Martin Tat

Heidelberg University

23rd January 2026



**UNIVERSITÄT
HEIDELBERG**
ZUKUNFT
SEIT 1386

Current meeting time: Tuesdays 10:30

- Is this slot suitable for everyone? Or should we move it?
 - As RD RTA liason, I have biweekly RTA WP3 at 10:00 meetings
- Note: As I'm not working full time on tracking efficiencies, it's likely I'll only organise these meetings biweekly instead of weekly
- I'll provide minor updates on MatterMost

TrackCalib2 update: Environment

- Issue: The TrackCalib2 environment setup has caused headache
 - Conda environment takes several hours to set up and occupies significant disk space
 - Personally, the `mamba` instructions never worked for me on `lxplus`
 - Different machines seem to throw different warnings/errors
- Solution: Set up an `lb-conda` environment
 - MR has been approved and merged: [!247](#)
 - I tested on `lxplus` and it worked out-of-the-box
 - Anyone can clone TrackCalib2 and have it running in a few seconds

TrackCalib2 update: Environment

Clone TrackCalib2 repository:

```
git clone ssh://git@gitlab.cern.ch:7999/lhcb-rta/trackcalib2.git  
git submodule init && git submodule update
```

Load environment: ← now takes seconds instead of hours

```
lb-conda trackcalib2
```

Run TrackCalib2:

```
python trackcalib.py prepare -year 2024_Block1  
python trackcalib.py fit -year 2024_Block1
```

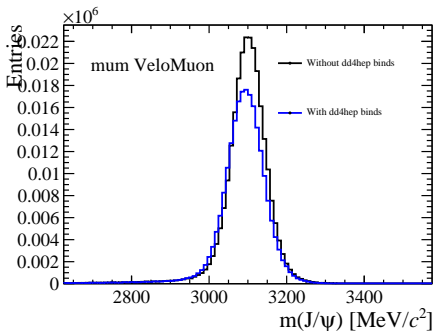
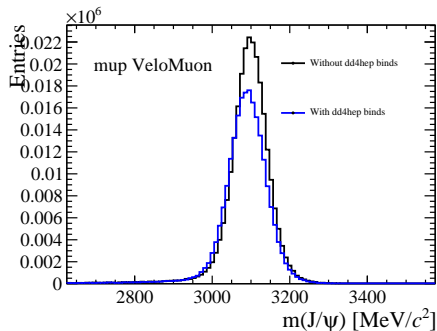
Missing DD4HEP binds

```
if UseDD4Hep:
```

```
    # This needs to happen before the public tools are instantiated,  
    # which means we cannot put it inside make_streams().  
    from PyConf.Tools import TrackMasterExtrapolator, TrackMasterFitter  
    TrackMasterExtrapolator.global_bind(  
        ApplyMultScattCorr=False,  
        ApplyEnergyLossCorr=False,  
        ApplyElectronEnergyLossCorr=False)  
    TrackMasterFitter.global_bind(ApplyMaterialCorrections=False)
```

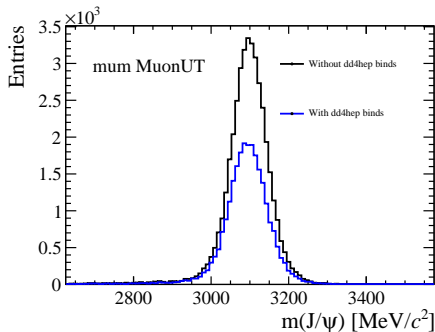
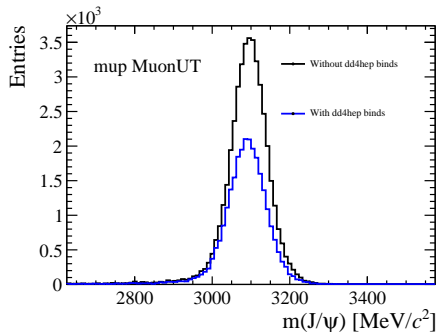
- Do I understand correctly that multiple scattering and energy loss corrections are disabled for data, but not MC? Why...?
- I ran a quick comparison between MC with and without these corrections, and differences are huge for VeloMuon and MuonUT

Missing DD4HEP binds



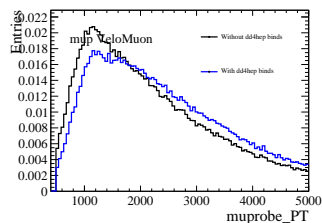
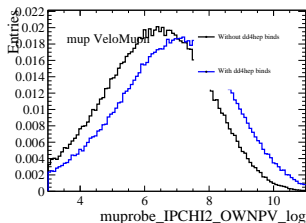
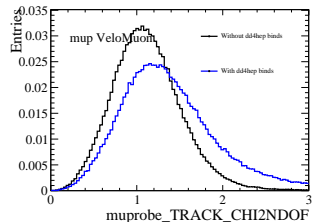
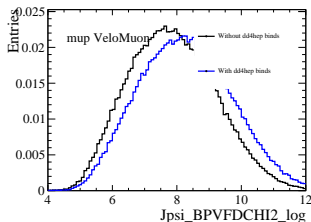
- Fewer candidates with DD4HEP binds for VeloMuon

Missing DD4HEP binds



- Difference is much larger for MuonUT!

Missing DD4HEP binds



- It's mostly VeloMuon probe muon distributions that are different...
- ... but I don't see much difference in MuonUT distributions

Sprucing decision filter

- Reminder: Michel discovered a prescale in blocks 7/8
- Events failing the prescale but triggered by other lines are kept when rerunning reconstruction
- Solution: Apply sprucing decision filter directly in Moore
- Rowina discovered that applying the filter in blocks 5/6 also changed the tracking efficiencies, and I confirmed this in TrackCalib2 as well...
- ... and for the number of J/ψ candidates, I see a 10% discrepancy (!!!) in block 5/6

Sprucing decision filter

- Check impact of DD4HEP binds using block 6 data
- The numbers are surprisingly consistent with Moore v55r12p3:

Spruce decision	DD4HEP binds	VeloMuon	Downstream	MuonUT
No	No	197/183	719/706	121/45
Yes	No	180/169	719/706	115/39
No	Yes	195/178	719/706	131/46
Yes	Yes	195/178	719/706	131/46

Sprucing decision filter

- We need to run Moore v55r12p8 due to bug fixes
- Here there are differences, but they are very small:

Spruce decision	DD4HEP binds	VeloMuon	Downstream	MuonUT
No	No	193/182	665/665	114/43
Yes	No	176/168	661/660	108/37
No	Yes	195/172	665/665	123/44
Yes	Yes	193/172	661/660	123/44

Missing DD4HEP binds and sprucing decision filter

- Sprucing decision filter definitely needed to get correct efficiencies in blocks 7/8
- But clearly we don't get consistent numbers in blocks 5/6 unless DD4HEP binds are included
- Don't quite understand the DD4HEP binds yet...
- ... but I will rerun all data APs with them

TrackCalib development 1: Remove Run 1/2 code

- Run 1/2 used different methods/variables/cuts/matching... huge amount of code unnecessary for Run 3
- I suggest separate releases for Run 1/2 and Run 3
 - Less code \implies Easier to maintain

TrackCalib development 2: Change to RDataFrame

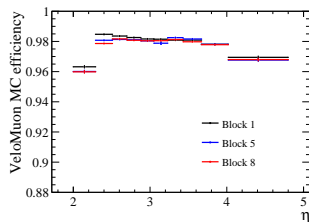
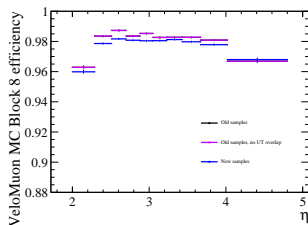
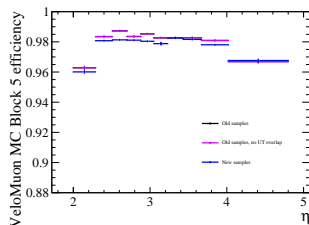
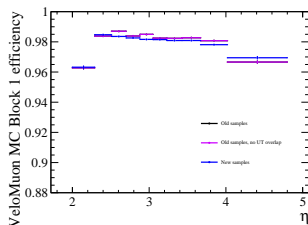
- Uproot processes hundreds of files one by one
- Preparing samples can take hours
 - Very difficult to perform studies that require reprocessing the samples
- I know ROOT dependency was removed in the past, but for sample preparation RDataFrame is much more suitable in my opinion
 - Multi-thread support
 - Lazy evaluation \implies Only need to loop once to apply all cuts and matching criteria
- After this change, sample preparation only takes a few minutes!
 - Allows me to prepare new samples and produce tracking efficiencies almost instantly once AP is ready

TrackCalib development 3: New data/MC samples

- Sim10d \rightarrow Sim10g
- Use overlap functors for matching
- All cuts and matching criteria updated (after discussion with Maurice and Rowina)
- Tracking efficiencies were compared before and after to understand differences
 - Small differences understood to be sprucing decision filter and wrong/outdated matching criteria in TrackCalib2
- Note: Plots I'm going to show don't have DD4HEP binds, which is expected to have a large impact

TrackCalib2 development 3

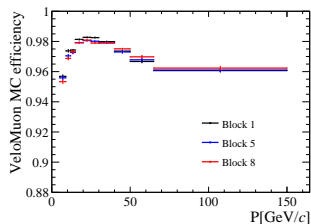
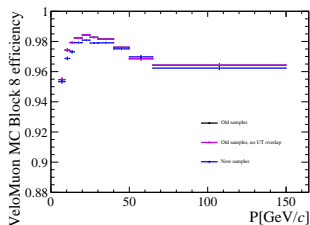
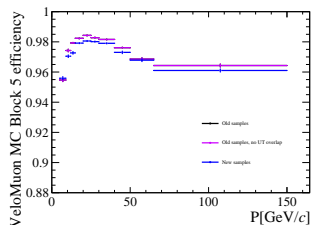
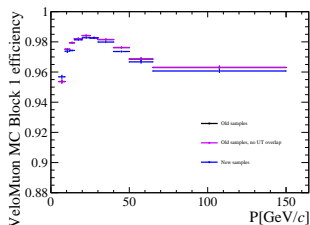
TrackCalib2 development 3: New data/MC samples



MC VeloMuon track efficiencies in η

TrackCalib2 development 3

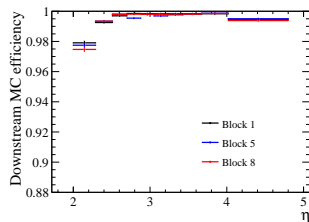
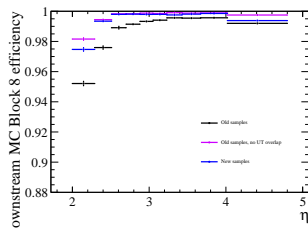
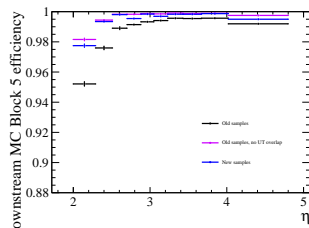
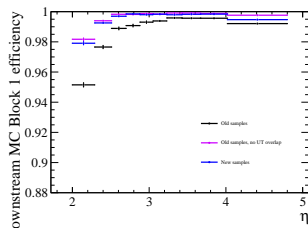
TrackCalib2 development 3: New data/MC samples



MC VeloMuon track efficiencies in p

TrackCalib2 development 3

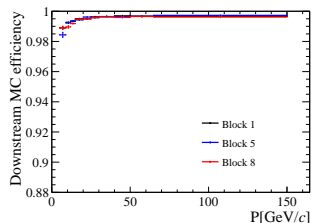
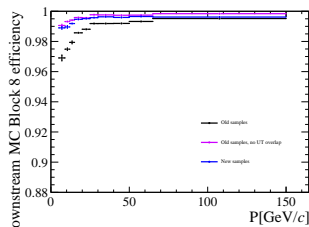
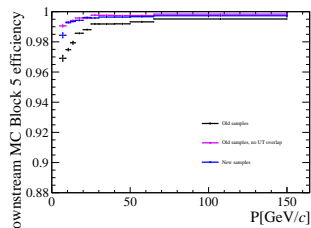
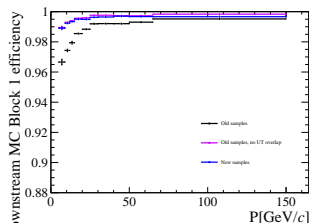
TrackCalib2 development 3: New data/MC samples



MC Downstream track efficiencies in η

TrackCalib2 development 3

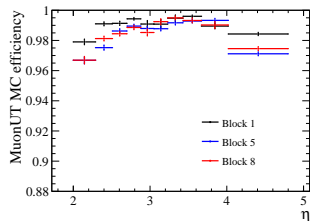
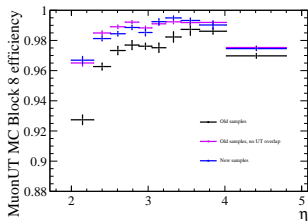
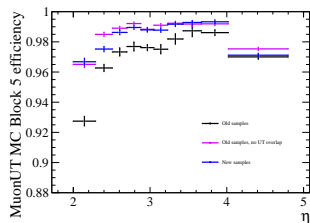
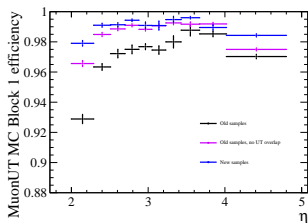
TrackCalib2 development 3: New data/MC samples



MC Downstream track efficiencies in p

TrackCalib2 development 3

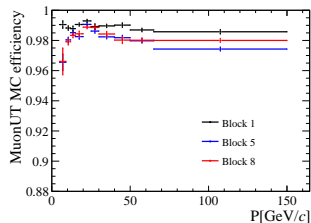
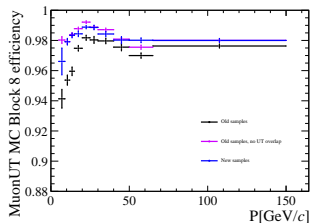
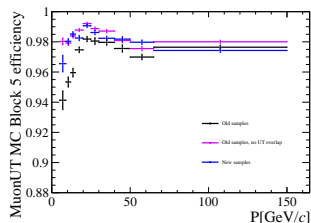
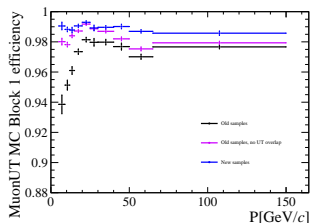
TrackCalib2 development 3: New data/MC samples



MC MuonUT track efficiencies in η

TrackCalib2 development 3

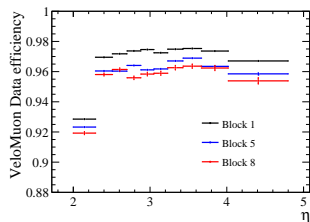
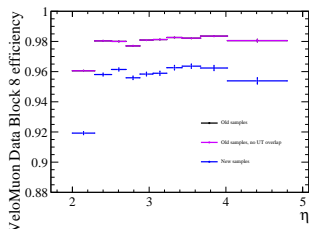
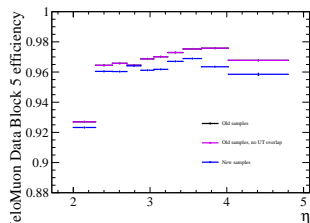
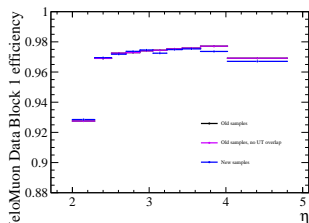
TrackCalib2 development 3: New data/MC samples



MC MuonUT track efficiencies in p

TrackCalib2 development 3

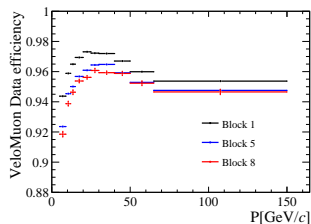
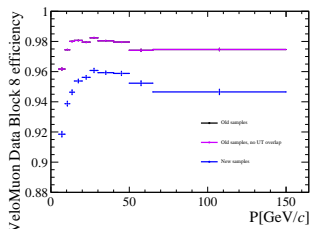
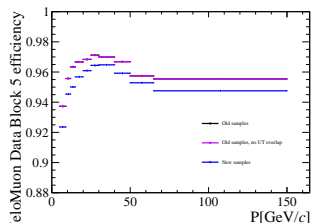
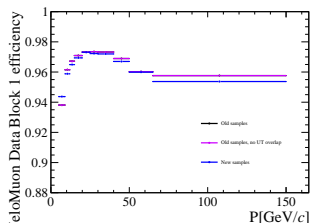
TrackCalib2 development 3: New data/MC samples



Data VeloMuon track efficiencies in η

TrackCalib2 development 3

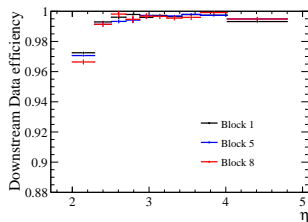
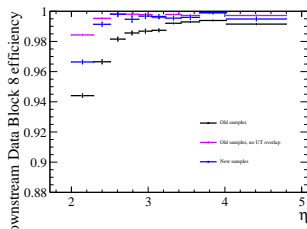
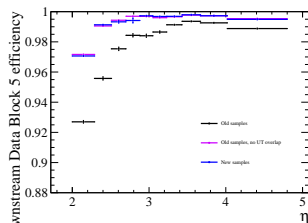
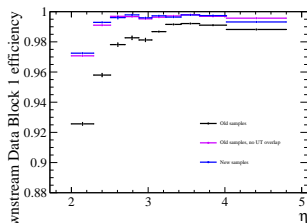
TrackCalib2 development 3: New data/MC samples



Data VeloMuon track efficiencies in p

TrackCalib2 development 3

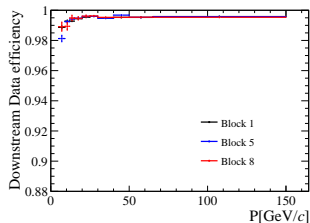
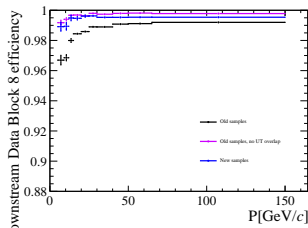
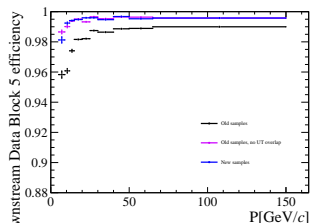
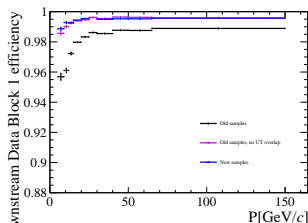
TrackCalib2 development 3: New data/MC samples



Data Downstream track efficiencies in η

TrackCalib2 development 3

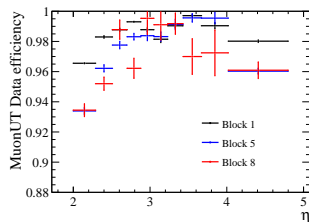
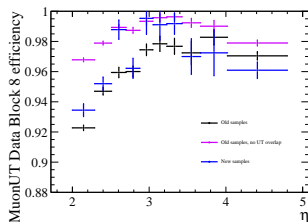
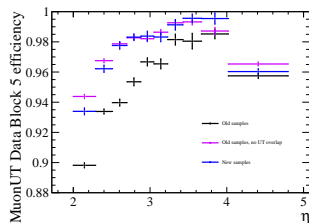
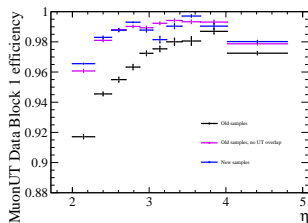
TrackCalib2 development 3: New data/MC samples



Data Downstream track efficiencies in p

TrackCalib2 development 3

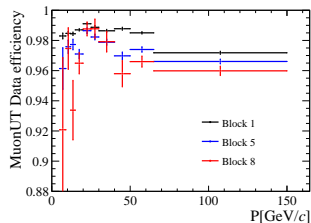
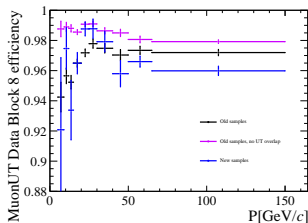
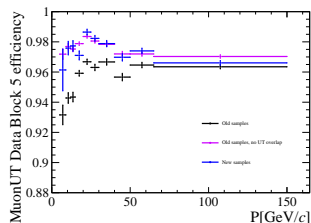
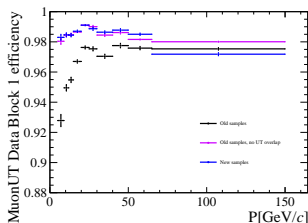
TrackCalib2 development 3: New data/MC samples



Data MuonUT track efficiencies in η

TrackCalib2 development 3

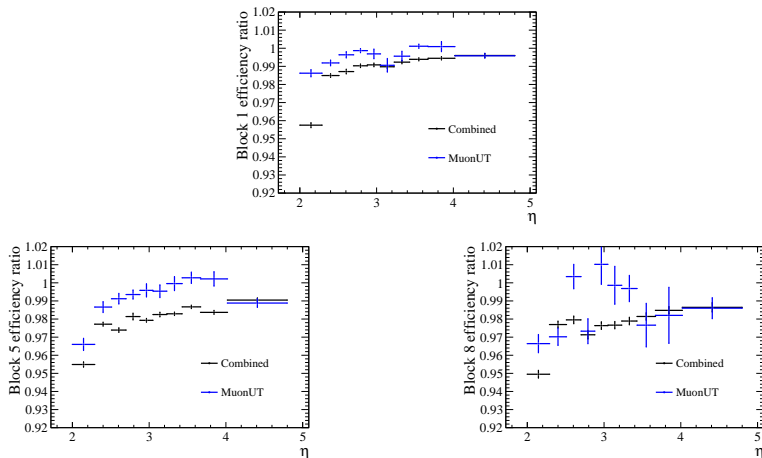
TrackCalib2 development 3: New data/MC samples



Data MuonUT track efficiencies in p

TrackCalib2 development 3

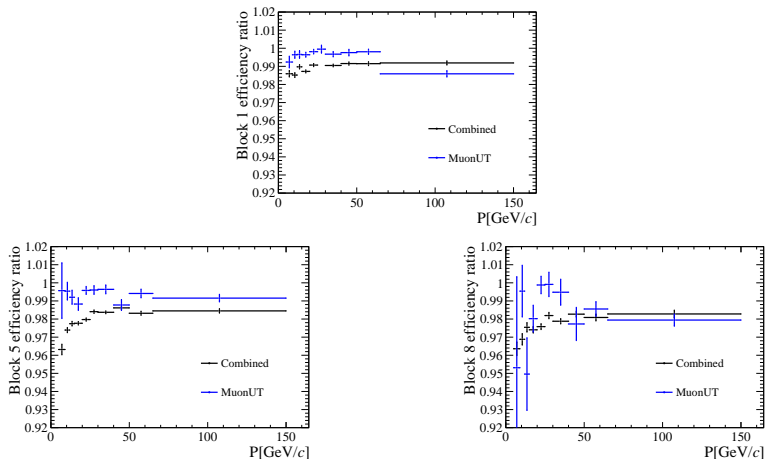
TrackCalib2 development 3: New data/MC samples



Data/MC ratio for Combined/MuonUT track efficiencies in η

TrackCalib2 development 3

TrackCalib2 development 3: New data/MC samples



Data/MC ratio for Combined/MuonUT track efficiencies in η

Preparing data/MC samples

- ① AP for blocks 1, 5, 6, 7, 8 of 2024 data
 - AP ready: [!4808](#)
 - Include sprucing decision and DD4HEP binds
 - Apply tag muon TOS filter (removes 75% of candidates)
 - Remove tag muon overlap variables (removes 20% of branches)
- ② We should run over block 2 as well
 - I requested MC yesterday
- ③ MC for block 1b
 - MC request has been stuck for 1 month: [!1640](#)
- ④ 2025 pre-TS data
 - MC is ready, need to create AP over data/MC
- ⑤ 2024 *pp* reference run
 - MC is ready, need to create AP over data/MC
- ⑥ Tidy up AP
 - Perhaps remove older samples that are not used
 - Track efficiency AP currently takes up 13 TB(!)

Summary

- ① Standard environment for TrackCalib2 works
- ② Briefly looked at impact of missing DD4HEP binds, impact seems to be significant but not understood yet
- ③ Sprucing decision inconsistency in block 5/6 has been resolved, understood to be inconsistent DD4HEP binds when rerunning reconstruction
- ④ Several updates to TrackCalib2:
 - Run 1/2 stuff removed, we should create a separate release for Run 3
 - RDataFrame for sample preparation is much faster
 - New data/MC samples
 - Overlap functors for matching
 - Sprucing decision filter included
 - New tracking efficiencies are consistent with previous samples once accounting for new matching criteria and sprucing decision

Next steps

- 1 Submit AP over 2024 data and rerun everything
- 2 Look at 2025 pre-TS tracking efficiencies
- 3 Add block 2 and pp ref run
- 4 Optimise 2D binning scheme and produce 2D data/MC ratio tables for all 2024 blocks and μ^\pm , μ^+ , μ^-
- 5 Unfortunately I've had limited time to look into Combined/MuonUT discrepancies, but I'll try to do this in parallel