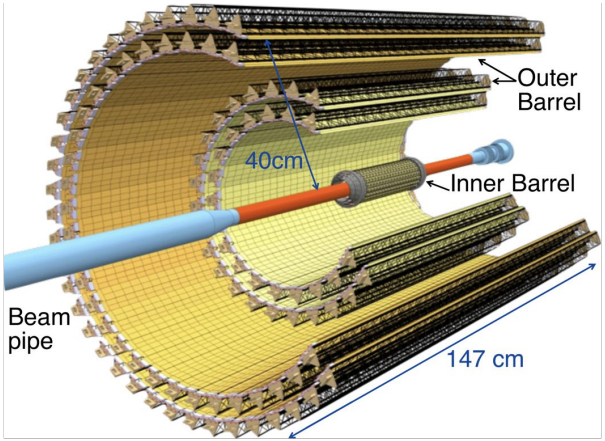
HF O^2 analysis framework

Vít Kučera, Francesco Prino,
Gian Michele Innocenti
on behalf of the HF O^2 team

HF O^2 hackathon
7 Dec 2021

HF challenges in Run 3



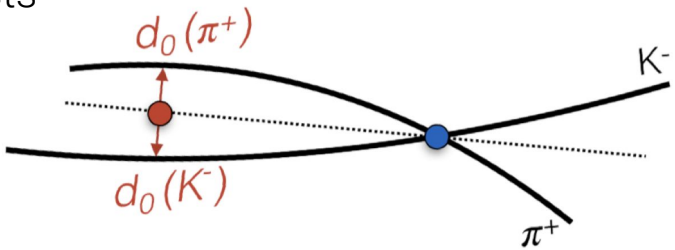
Precise HF measurements down to p_T = 0

- 100 times more Pb–Pb data than in Run 2
- Large combinatorial background
- Small S/B ratio, difficult triggering

→ HF reconstruction and selection as the most challenging analysis process in Run 3

Framework-design requirements:

- Minimize disk space occupied by derived analysis objects
- Maximize CPU performance
- Flexible structure, skims,...



A special thanks and a lot of credit to those who developed, optimized and maintained the AliPhysics heavy-flavour software from which we enormously profited (A. Rossi's talk)!

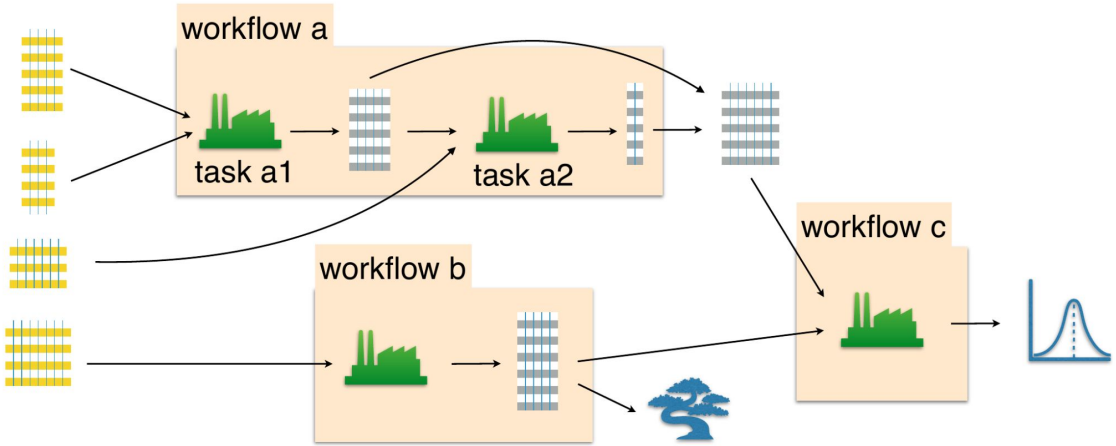
HF reconstruction and analysis in O^2

Recap of the O^2 analysis framework

Data format and handling
- Data stored in flat tables
- Tables interlinked via indices

Table processing
- Declarative/imperative

Tasks and configuration
- Tasks organised in workflows
- Subscriptions to input tables determine topology.
- Configuration via JSON



See Anton's talk

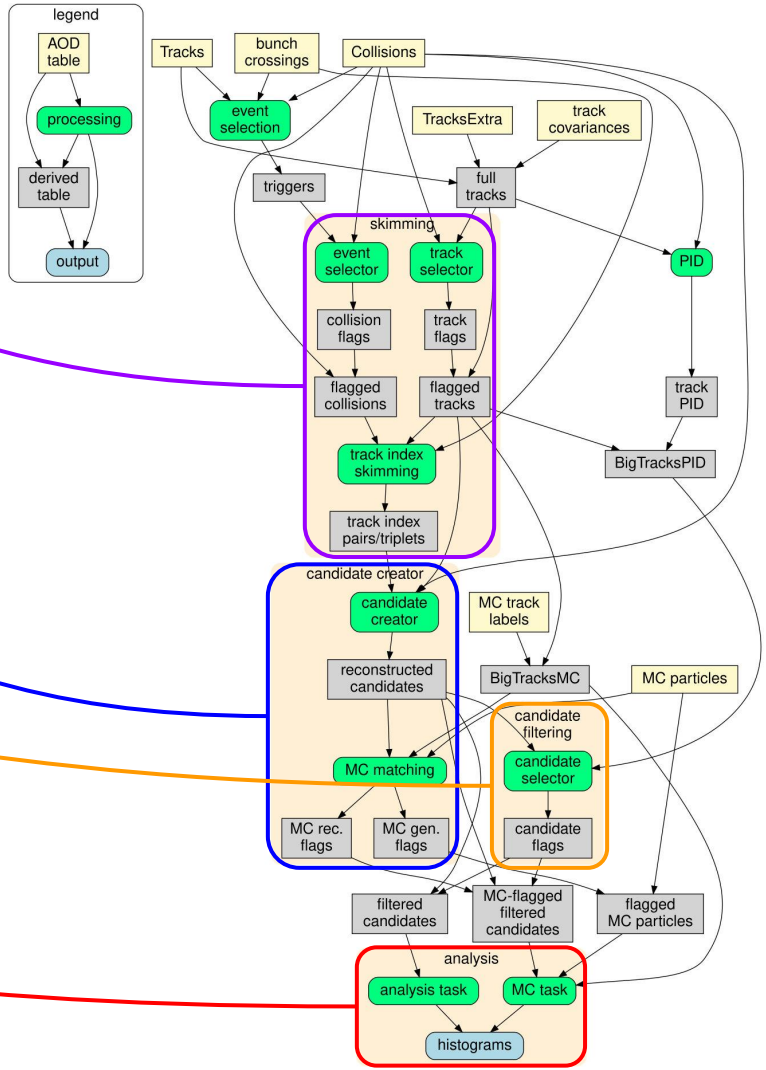
HF O^2 analysis framework

Track index skim creator

Candidate creator

Candidate selector

Analysis task



Track skim creator

Track index skim creator

Candidate creator

Candidate selector

Analysis task

Input: tracks, collisions

Event selection

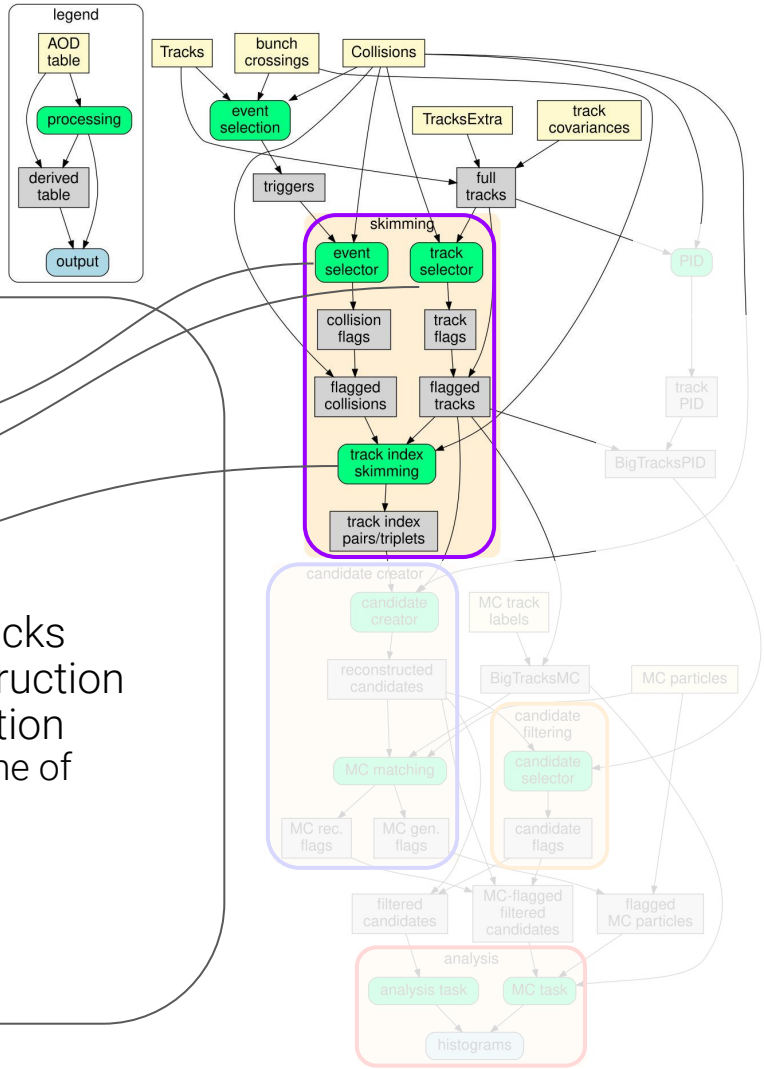
Track selection

- p_T, η, DCA, quality

Skimming

- Double/triple loop over tracks
- Secondary-vertex reconstruction
- Loose candidate preselection
  - invariant mass, p_T, cosine of pointing angle,...

Output: track skims (skimmed pairs/triplets of track indices)



Candidate creator

Track index skim creator

Candidate creator

Candidate selector

Analysis task

Input: track skims

Candidate creation

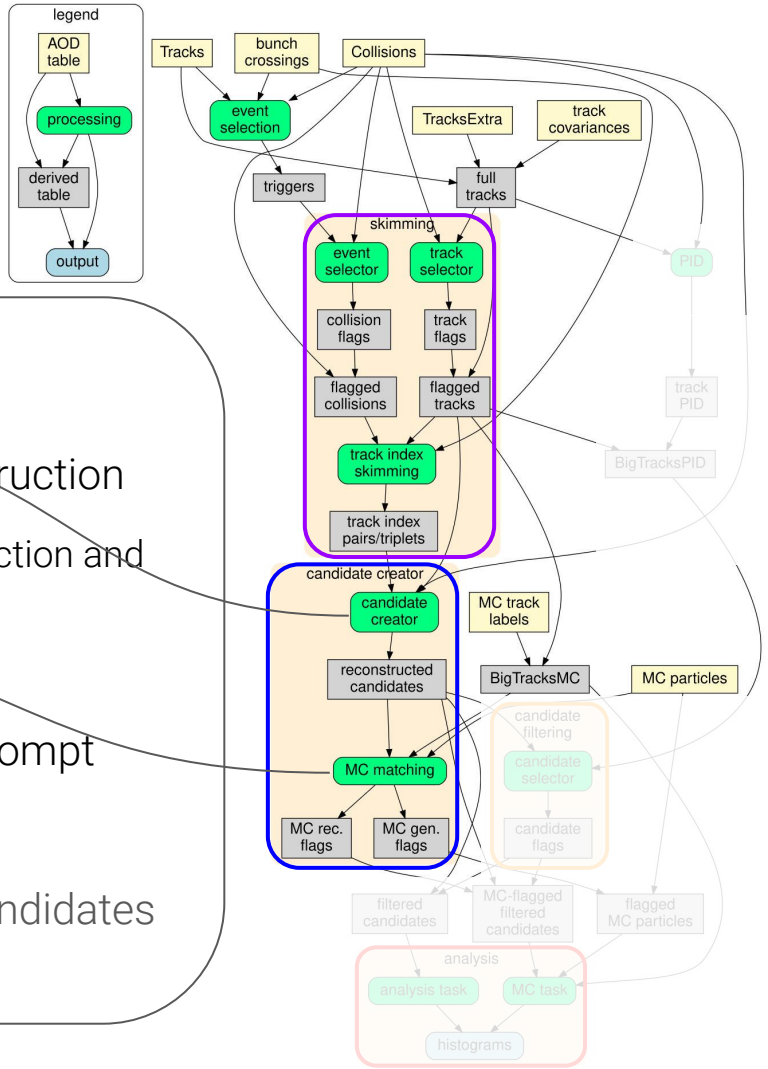
- Secondary-vertex reconstruction
- Candidate building
  - full information for selection and analysis

MC matching

- Rec. level (candidate)
- Gen. level (MC particle)
- MC origin tracing (non-)prompt
  - (from c/b quark)

Output:

- Fully reconstructed HF candidates
- MC flags



Candidate selector

Track index skim creator

Candidate creator

Candidate selector

Analysis task

Input:

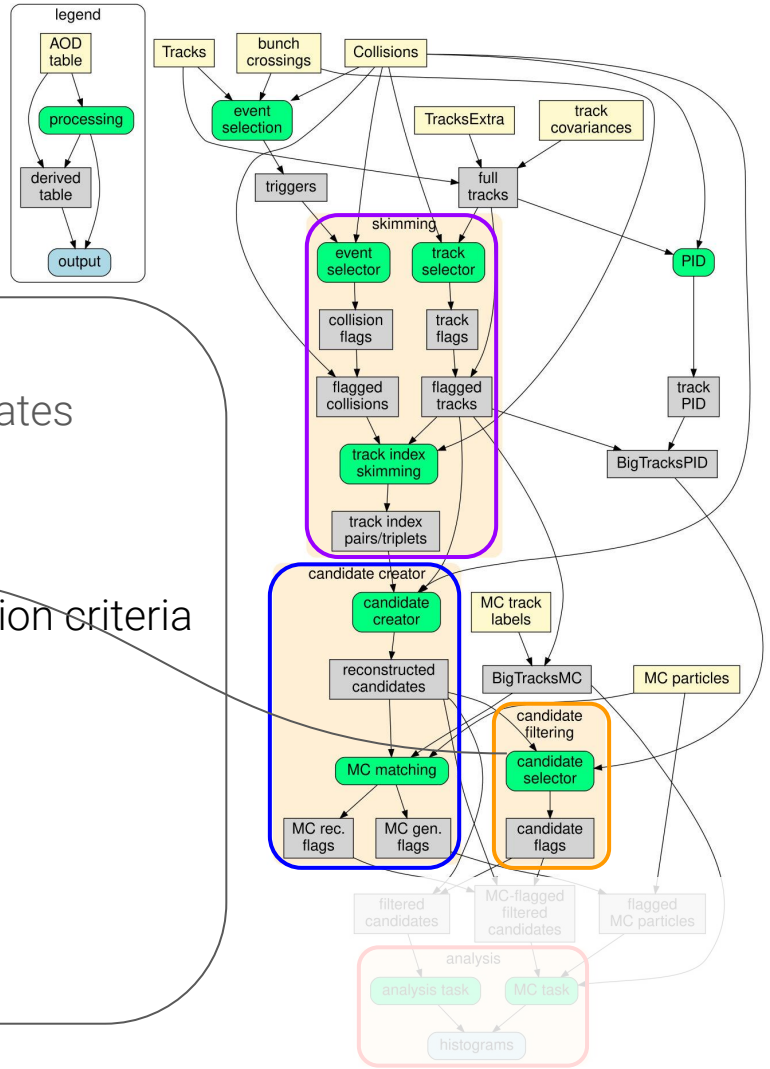
- Reconstructed HF candidates
- Track PID

Candidate selection

Application of candidate selection criteria

- Topological cuts
- Daughter PID cuts

Output: selection flags



Analysis task

Track index skim creator

Candidate creator

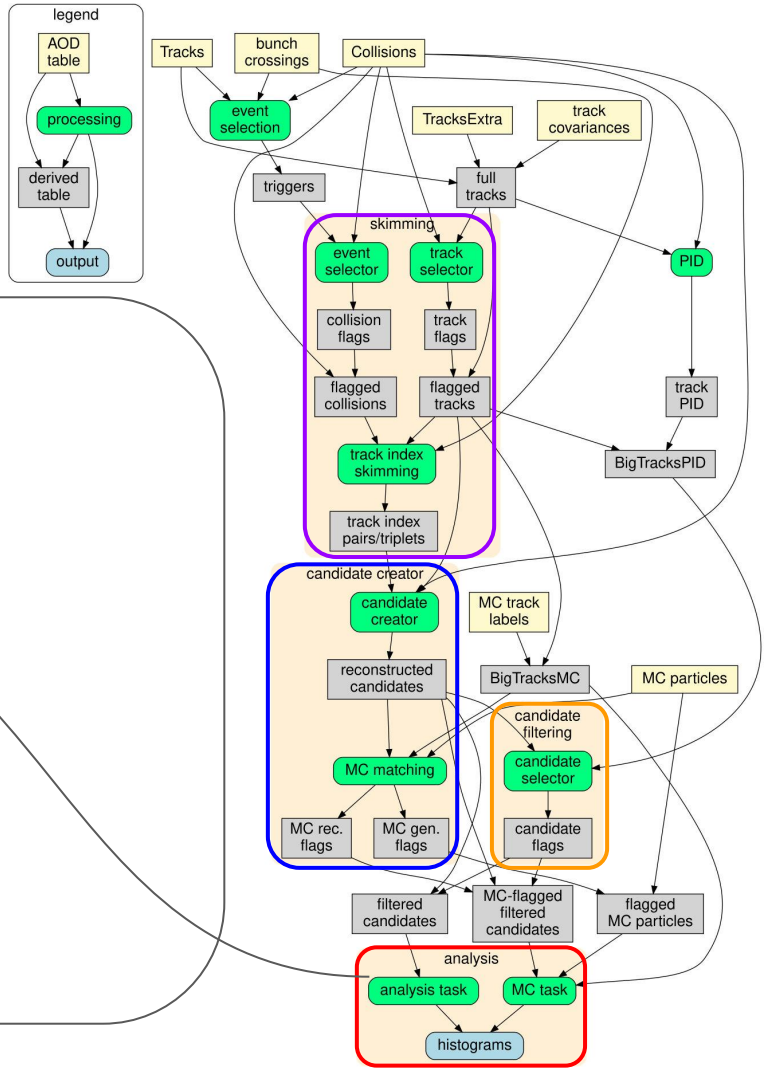
Candidate selector

Analysis task

Input:
- Selected candidates
- MC particles
- MC flags

Analysis task
- Filter application
- Histogram filling

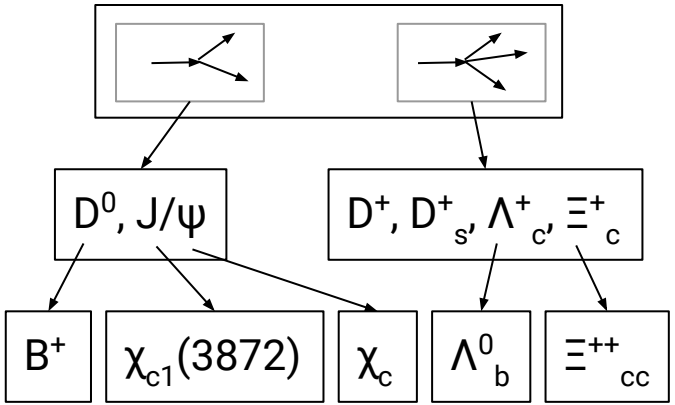
Output: histograms
- Kinematic properties
- Signal vs. background
- Reconstruction efficiency
- Prompt vs. non-prompt



More complex analyses

Modularity of O^2 workflows allows to build analyses of multi-stage decays on top of analyses of direct decays.

E. g.: $B^+ \rightarrow D^{0bar}\ \pi^+$:
Track skim creator
→ 2-prong candidate creator
→ D^0 selector
→ B^+ candidate creator
→ B^+ candidate selector
→ B^+ analysis task



$D^0 \rightarrow \pi^+ K^-$
$D^{0/+}$–$D^{0bar/-}$ correlations
$B^+ \rightarrow D^{0bar}\ \pi^+$
$D^+ \rightarrow \pi^+ K^-\ \pi^+$
$D^+_s \rightarrow \pi^+ K^-\ K^+$
$\Lambda^+_c \rightarrow p\ K^-\ \pi^+$
$\Xi^+_c \rightarrow p\ K^-\ \pi^+$
$\Xi^{++}_{cc} \rightarrow \Xi^+_c\ \pi^+$
$\Lambda^+_c \rightarrow p\ K^0_S$
$J/\psi \rightarrow e^+\ e^-$
$J/\psi \rightarrow \mu^+\ \mu^-$
$\chi_{c1}(3872) \rightarrow J/\psi\ \pi^+\ \pi^-$
$\chi_c \rightarrow J/\psi\ \gamma$

Heavy-flavour analyses on AliHyperloop

All HF O^2 workflows configured as wagons on the Grid and tested on converted Run 2 data sets

Wagon	LHC15o_test	LHC20a7_cent	LHC17pp_test	LHC20e3a	LHC17p_pass1	LHC17q_pass1	LHC20g2a_2	Last run
hf-candidate-creator-2prong	✕	✕	✕	✕	✕	✕	✕	4402
hf-candidate-creator-3prong	✕	✕	✕	✕	✕	✕	✕	4581
hf-candidate-creator-cascade	✕	✕	✕	✕	✕	✕	✕	
hf-candidate-selector-D0-toKPi	✕	✕	✕	✕	✕	✕	✕	4402
hf-candidate-selector-DPlus-toPiKPi							✕	
hf-candidate-selector-Jpsi							✕	
hf-candidate-selector-Lc-toPKPi							✕	4580
hf-candidate-selector-Xic-toPKPi							✕	
hf-mc-validation							✕	
hf-qa-efficiency							✕	
hf-task-D0							✓	4402
hf-task-DPlus							✓	4581
hf-task-Jpsi							✓	
hf-task-Lc-toKs0p								
hf-task-Lc-toPKPi							✓	4580
hf-task-Xic							✓	
hf-track-index-skims-creator-2prong	✕	✕	✕	✕	✕	✕	✕	4402
hf-track-index-skims-creator-3prong	✕	✕	✕	✕	✕	✕	✕	4581

hf-candidate-selector-D0-toKPi

D0_to_pi_K_cuts

#	m	DCA	cos theta*	pT K	pT Pi	d0K	d0pi	d0d0	cos pointing angle	cos pointing angle xy	normalized dec
pT bin 0	0.4	0.03	0.8	0.5	0.5	0.1	0.1	-0.00	0.8	0	0
pT bin 1	0.4	0.03	0.8	0.5	0.5	0.1	0.1	-0.00	0.8	0	0
pT bin 2	0.4	0.03	0.8	0.4	0.4	0.1	0.1	-0.00	0.8	0	0
pT bin 3	0.4	0.03	0.8	0.4	0.4	0.1	0.1	-0.00	0.8	0	0
pT bin 4	0.4	0.03	0.8	0.7	0.7	0.1	0.1	-0.00	0.9	0	0
pT bin 5	0.4	0.03	0.8	0.7	0.7	0.1	0.1	-0.00	0.9	0	0
pT bin 6	0.4	0.03	0.8	0.7	0.7	0.1	0.1	-0.00	0.85	0	0
pT bin 7	0.4	0.03	0.8	0.7	0.7	0.1	0.1	-0.00	0.85	0	0
pT bin 8	0.4	0.03	0.8	0.7	0.7	0.1	0.1	-0.00	0.85	0	0
pT bin 9	0.4	0.03	0.8	0.7	0.7	0.1	0.1	-0.00	0.85	0	0
pT bin 10	0.4	0.03	0.8	0.7	0.7	0.1	0.1	-0.00	0.85	0	0
pT bin 11	0.4	0.03	0.8	0.7	0.7	0.1	0.1	-0.00	0.85	0	0

Save

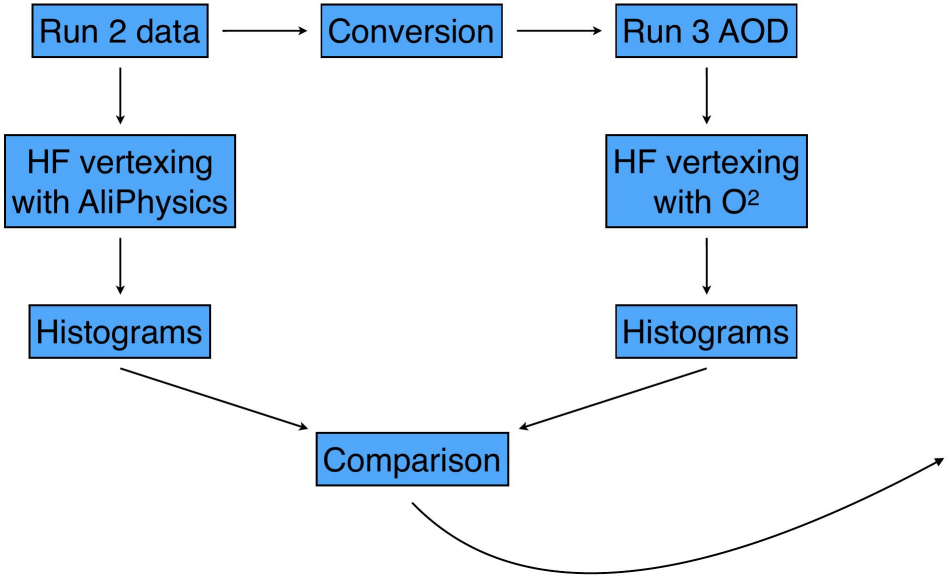
https://alimonitor.cern.ch/hyperloop

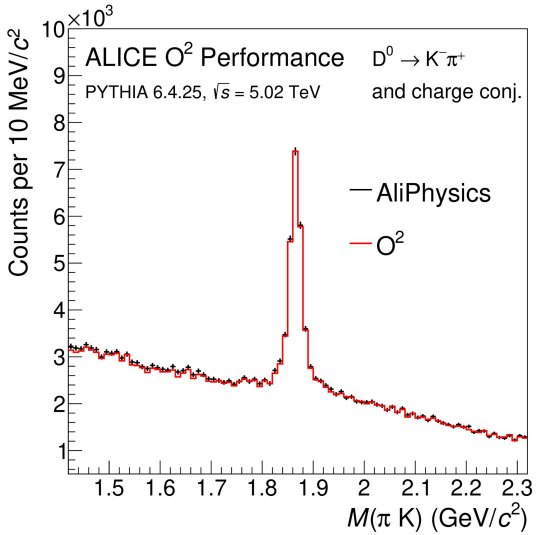
Validation framework

Validation framework

Tool for an easy execution, testing and validation of O^{2}Physics tasks
https://github.com/AliceO2Group/Run3Analysisvalidation

- ESD → AO2D conversion
- Flexible execution of arbitrary [Ali/O^2]Physics analysis tasks
- Job parallelisation, output merging, error checking, postprocessing
- Easy generation of the O^2 command from a database of workflows and options





Execution steps

Run: `bash runtest.sh` (symlinked in `codeHF`)

- Load settings
  - Set parameters based on specified input and requested tasks
- Clean directory
  - Delete files produced by previous runs
- Generate list of input files
- Modify JSON file
  - Activate MC, apply candidate selection, set collision system, Run 2/3/5,...
- Run ESD → AO2D conversion
- Run AliPhysics tasks
- Run O^2Physics tasks
- Run postprocessing
  - Comparison AliPhysics vs O^2, efficiency plots,...
- Clean directory
  - Delete temporary files

Input configuration

`config_input.sh` defines what data will be processed and how.

`NFILESMAX` Maximum number of processed input files
`NFILESPERJOB_[CONVERT,ALI,O2]` Number of input files per [conversion, AliPhysics, O^2] job
`NJOBSPARALLEL_O2` Maximum number of simultaneously running O^2 jobs

`JSON` Path to the JSON file (`dpl-config.json`)

`INPUT_LABEL` Data description
`INPUT_DIR` Input directory
`INPUT_FILES` Input file pattern
`INPUT_SYS` Collision system
`INPUT_RUN` Run (2, 3, 5)
`ISMC` Is the input MC data? (0, 1)
`ISINPUTO2` Is the input in the O^2 format? (0, 1)
`ISALICE3` Is the input from the ALICE 3 detectors? (0, 1)

Task configuration

`config_tasks.sh` defines which steps will be executed and what they will do.

- Generates job scripts: `script_ali.sh`, `script_o2.sh`, `script_postprocess.sh`

Step activation: `DOCLEAN`, `DOCONVERT`, `DOALI`, `DOO2`, `DOPOSTPROCESS`
Activation of O^2 workflows: `DOO2_SKIM`, `DOO2_SEL_D0`, `DOO2_TASK_D0`, `DOO2_…`
Application of selection cuts: `APPLYCUTS_D0`
Save derived tables as trees: `SAVETREES`

`AdjustJson` Function that modifies the JSON file based on the input parameters
`MakeScriptO2` Function that generates the O^2 script containing the full O^2 command

- O^2 command generated by a Python script based on a YAML database of workflows, their dependencies and options (`workflows.yml`).

How to add a new workflow

1. **Add the workflow specification in the database** (`workflows.yml`).
   `o2-analysis-hf-my-analysis-wf:`
   `executable:` Name of the workflow executable
   Allows to define the same workflow multiple times with different names and options.
   `dependencies:` Required workflows (follows table subscriptions)
   `requires_mc:` Does the workflow require MC input?
   `options:` Command line options
   Format: `str`, `list` or `dict` (if `dict`: `default`, `real`, `mc`)
   `tables:` Derived tables, same format as `options`
   Unnecessary settings can be omitted.

2. **Add the workflow activation in the task configuration** (`config_tasks.sh`).
   ```
   DOO2_MYWF=0
   [ $DOO2_MYWF -eq 1 ] && WORKFLOWS+=" o2-analysis-hf-my-analysis-wf"
   ```

3. **Add the workflow configuration in the JSON file** (`dpl-config.json`).
   `"my-analysis-task": {...}`

Extras

Debugging mode

See more details in the terminal

`bash runtest.sh -d`

Job debugging tool

Display bad jobs, their input, warnings, errors

`bash exec/debug.sh`

Maintenance tool

Sync Git repos (upstream/master, origin/mybranch)

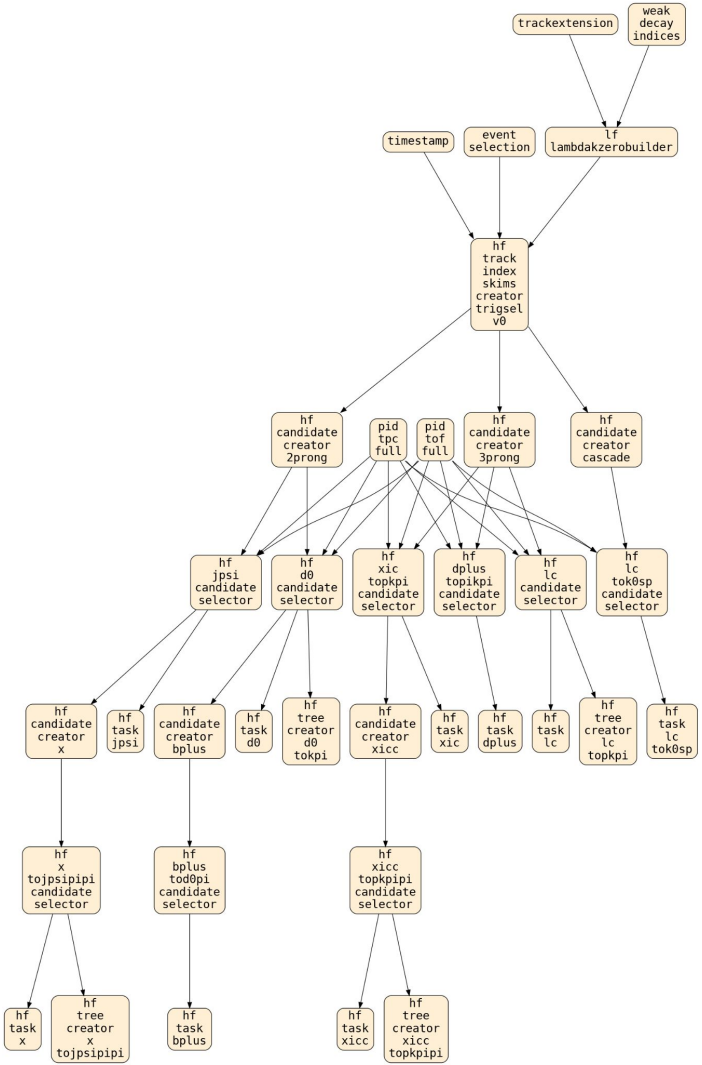
Build packages with aliBuild

Delete obsolete builds (deeper cleanup)

`bash exec/update_packages.sh`

Dependency diagram

Set `MAKE_GRAPH=1` in `config_tasks.sh`



References and useful information

Mattermost: https://mattermost.web.cern.ch/alice/channels/hf-o2-analysis-challenge

Documentation:
https://aliceo2group.github.io/analysis-framework/docs/framework/pwghf.html

O^2Physics code:
https://github.com/AliceO2Group/O2Physics/tree/master/PWGHF
Validation framework & postprocessing analysis tools:
https://github.com/AliceO2Group/Run3Analysisvalidation

Meetings:

- Weekly HF O^2 meetings: Tue 9:30, https://indico.cern.ch/category/9431/
  → Core HF development, utilities, processing, MC tools,…
- PAG/PWGHF meetings for all the analysis-related discussions

Ongoing developments: getting ready for real analyses!

Ongoing activities and topic “task forces”

HF O^2 framework is in an advanced state of development, but there is still a lot of work to do!
Four major areas of developments identified with the help of HF conveners and HF experts:

MC correction and reweighting

- Strategy for MC-based efficiency and selection corrections
- MC generation/configuration for HF simulation
- Multidimensional MC-reweighting to improve MC/data agreement

ML application, model preservation and selector refactory

- Application of training models with local and Grid resources
- ML model storage and long-term preservation
- “Modular” refactory of the selector classes:
  → Multi-purpose software unit to perform both rectangular or ML-based selection

Ongoing activities and topic “task forces”

HF O^2 framework is in an advanced state of development, but there is still a lot of work to do!
Four major areas of developments identified with the help of HF conveners and HF experts:

Post-processing (after O^2 processing) macros and QA:

- Optimization of the code to analyze the O^2 outputs to produce final results for analyses
  - Efficiency macros, cross section calculations, fitting, feed-down subtraction
- Optimized validation for HF MC simulations and reconstructed data

Skim creation, size/CPU estimation, triggering

- Optimize structure of derived data creation (*track skims, candidate skim, trees for offline optimization*)
- Processing using derived data as starting point
- Resource estimation and optimization
- Triggering strategy (based on the track skims)

Conclusions

→ Data are about to come, and it is still to complete and polish our software!
→ The contribution from the “analysis” community is now really critical.
  – Experts to help the steering of the projects and new developers

→ The menu of today
- Hands on session for analysis in O^2
- Hands on sessions for HF analysis in O^2
- Time for discussion!

Thanks for your attention