PWGCF: Hands-on Session
Femto framework(s)

O2 Analysis Tutorial 4.0

17.10.2024

Anton Riedel, Georgios Mantzaridis, Jaime Gonzalez Gonzalez

Overview

- There are a few frameworks available in O2Physics under PWGCF folder:

```
[anton@silver] ~/alice/O2Physics
> find PWGCF/ -type d -iname "Femto*"
PWGCF/Femto3D
PWGCF/FemtoDream
PWGCF/FemtoUniverse
PWGCF/FemtoWorld
```

First implementation Specialized for analyzing p-p data

Overview

- There are a few frameworks available in O2Physics under PWGCF folder:

```
[anton@silver] ~/alice/O2Physics
> find PWGCF/ -type d -iname "Femto*"
PWGCF/Femto3D
PWGCF/FemtoDream
PWGCF/FemtoUniverse
PWGCF/FemtoWorld
```

First implementation Specialized for analyzing p-p data

- Data formats of the different frameworks are partially different -> will be unified/updated in very near future

```
[anton@silver] ~/alice/O2Physics
> find PWGCF/DataModel/ -name "Femto*"
PWGCF/DataModel/FemtoDerived.h
```

Derived data & Analysis

- Full chain of femto analysis

AO2D.root

ROOT file(s)

DF:
- O2collision
- O2track
- O2muon
- ...

Derived data & Analysis

- Full chain of femto analysis

AO2D.root

ROOT file(s)

DF:
- O2collision
- O2track
- O2muon
- ...

Producer task

Store	Binding	Description
[ ]	FDExtMCParticles	FDEXTMCPARTICLE
[ ]	FDExtParticles	FDEXTPARTICLE
[ ]	FDMCLabels	FDMCLabel
[ ]	FDMCParticles	FDMCPARTICLE
[x]	FDCollisions	FDCOLLISION
[x]	FDParticles	FDPARTICLE

Derived data & Analysis

- Full chain of femto analysis

AO2D.root

ROOT file(s)

DF:
- O2collision
- O2track
- O2muon
- ...

Producer task

Store	Binding	Description
☐	FDExtMCParticles	FDEXTMCPARTICLE
☐	FDExtParticles	FDEXTPARTICLE
☐	FDMCLabels	FDMCLabel
☐	FDMCParticles	FDMCPARTICLE
☑	FDCollisions	FDCOLLISION
☑	FDParticles	FDPARTICLE

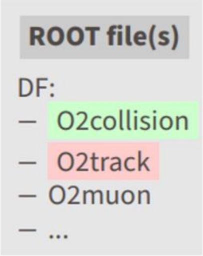
```
o2-analysis-cf-femtodream-producer
```

- Fill femto tables for analysis
- Only store events and tracks/V0/… fulfilling loosest selections
- Compute bitmask for track and PID selections
- Generate extended and/or MC tables for QA and more

Derived data & Analysis

- Full chain of femto analysis

AO2D.root

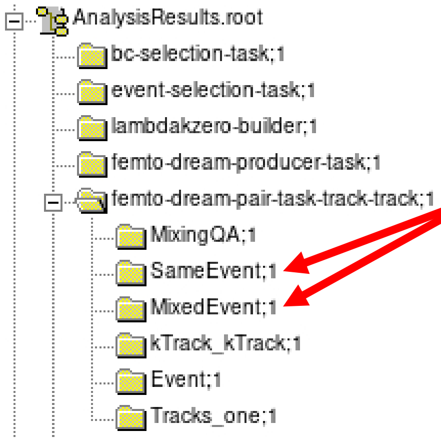


Producer task

Store	Binding	Description
☐	FDExtMCParticles	FDEXTMCPARTICLE
☐	FDExtParticles	FDEXTPARTICLE
☐	FDMCLabels	FDMCLabel
☐	FDMCParticles	FDMCPARTICLE
☑	FDCollisions	FDCOLLISION
☑	FDParticles	FDPARTICLE

Pair task

AnalysisResults.root



Produce same and mixed event distributions for post-processing

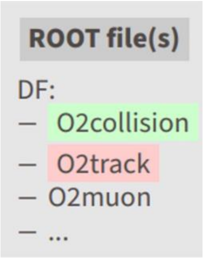
`o2-analysis-cf-femtodream-producer`

- Fill femto tables for analysis
- Only store events and tracks/V0/… fulfilling loosest selections
- Compute bitmask for track and PID selections
- Generate extended and/or MC tables for QA and more

Derived data & Analysis

- Full chain of femto analysis

AO2D.root

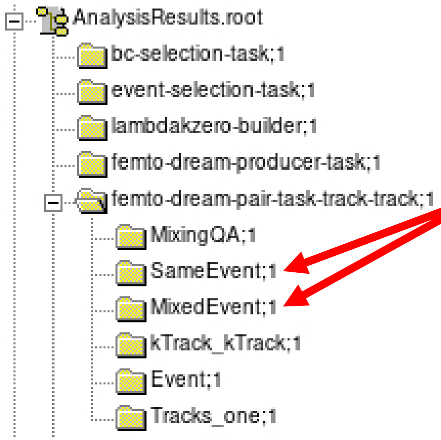


Producer task

Store	Binding	Description
☐	FDExtMCParticles	FDEXTMCPARTICLE
☐	FDExtParticles	FDEXTPARTICLE
☐	FDMCLabels	FDMCLabel
☐	FDMCParticles	FDMCPARTICLE
☑	FDCollisions	FDCOLLISION
☑	FDParticles	FDPARTICLE

Pair task

AnalysisResults.root



Produce same and mixed event distributions for post-processing

`o2-analysis-cf-femtodream-producer`

`o2-analysis-cf-femtodream-pair-track-track`

- Fill femto tables for analysis
- Only store events and tracks/V0/… fulfilling loosest selections
- Compute bitmask for track and PID selections
- Generate extended and/or MC tables for QA and more

- Select particles according to the bitmask
- Pair tracks/V0s in single event for same event distribution
- Pair tracks/V0s across multiple events for mixed event distribution

Derived data & Analysis

- Full chain of femto analysis

AO2D.root

ROOT file(s)

DF:
- O2collision
- O2track
- O2muon
- ...

Producer task

Store	Binding	Description
☐	FDExtMCParticles	FDEXTMCPARTICLE
☐	FDExtParticles	FDEXTPARTICLE
☐	FDMCLabels	FDMCLabel
☐	FDMCParticles	FDMCPARTICLE
☑	FDCollisions	FDCOLLISION
☑	FDParticles	FDPARTICLE

Pair task

AnalysisResults.root

- AnalysisResults.root
  - bc-selection-task;1
  - event-selection-task;1
  - lambdakzero-builder;1
  - femto-dream-producer-task;1
  - femto-dream-pair-task-track-track;1
    - MixingQA;1
    - SameEvent;1
    - MixedEvent;1
    - kTrack_kTrack;1
    - Event;1
    - Tracks_one;1

Produce same and mixed event distributions for post-processing

- Run full chain in one go
  - Femto tables are **generated in memory and immediately consumed** by the pair task, **only analysis output is saved to disk**
- Computationally very expensive to run full chain
  - Use **only on small/medium dataset** for testing and QA

Derived data & Analysis

- Full chain of femto analysis

AO2D.root

ROOT file(s)

DF:
- O2collision
- O2track
- O2muon
- ...

Producer task

Store	Binding	Description
☐	FDExtMCParticles	FDEXTMCPARTICLE
☐	FDExtParticles	FDEXTPARTICLE
☐	FDMCLabels	FDMCLabel
☐	FDMCParticles	FDMCPARTICLE
☑	FDCollisions	FDCOLLISION
☑	FDParticles	FDPARTICLE

Pair task

AnalysisResults.root

- AnalysisResults.root
  - bc-selection-task;1
  - event-selection-task;1
  - lambdakzero-builder;1
  - femto-dream-producer-task;1
  - femto-dream-pair-task-track-track;1
    - MixingQA;1
    - SameEvent;1
    - MixedEvent;1
    - kTrack_kTrack;1
    - Event;1
    - Tracks_one;1

Produce same and mixed event distributions for post-processing

Run producer task alone and save its output to disk **-> Derived data**

AO2D.root Containing femto tables only

- AO2D.root
  - DF_2247591081241500;1
    - O2fdcollision;1
    - O2fdparticle;1
  - parentFiles;1

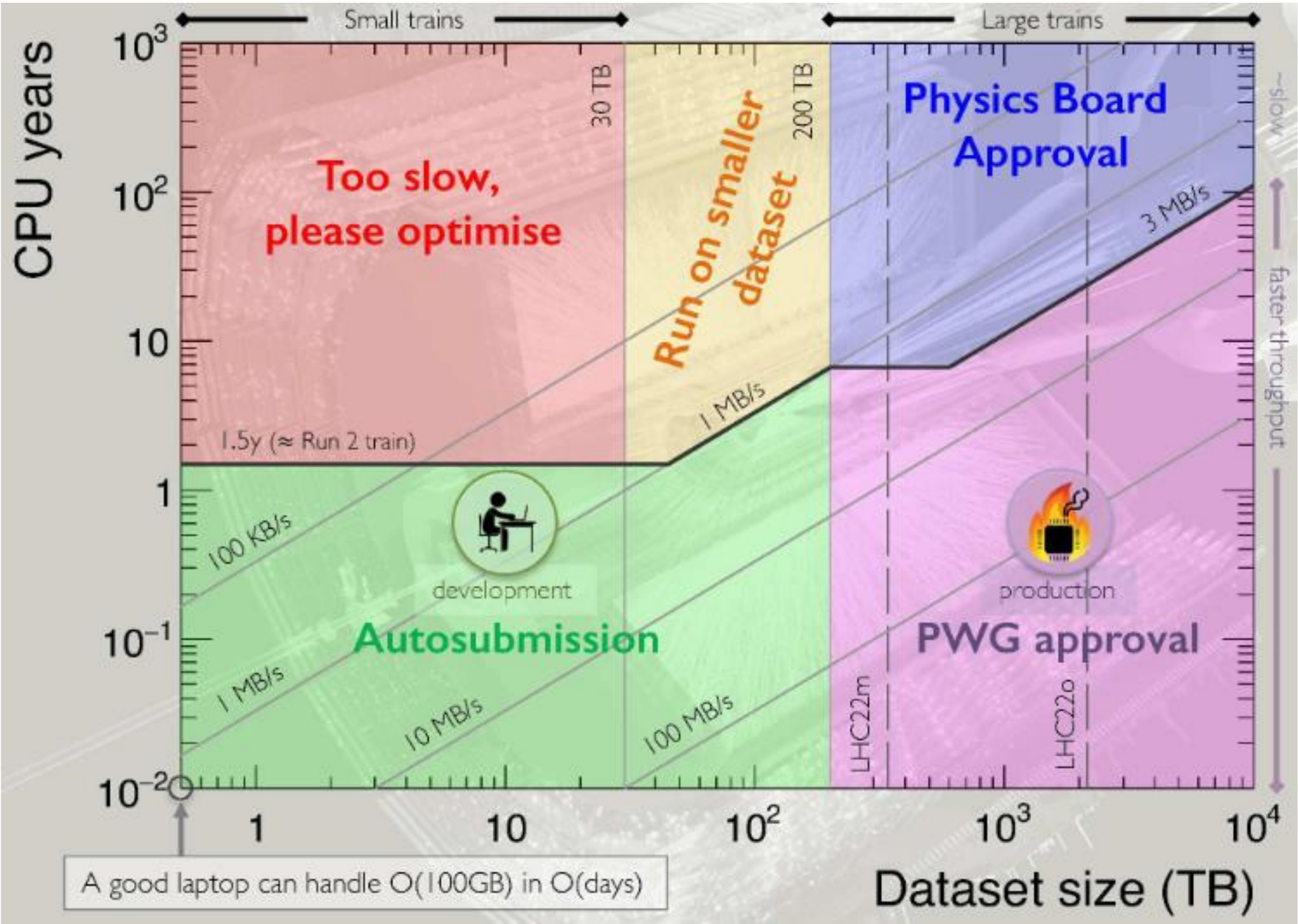
Run pair task alone with **derived data as input**

- Split (final) analysis into **2 parts**
  1. Run skimming up front and **remove unnecessary information to reduce the size of the data**
  2. Run the **analysis on derived data**
- Computationally **very efficient, especially for doing systematics**

Analysis of LHC22 MB pass4 (3PB)

Hyperloop test of femtodream producer task

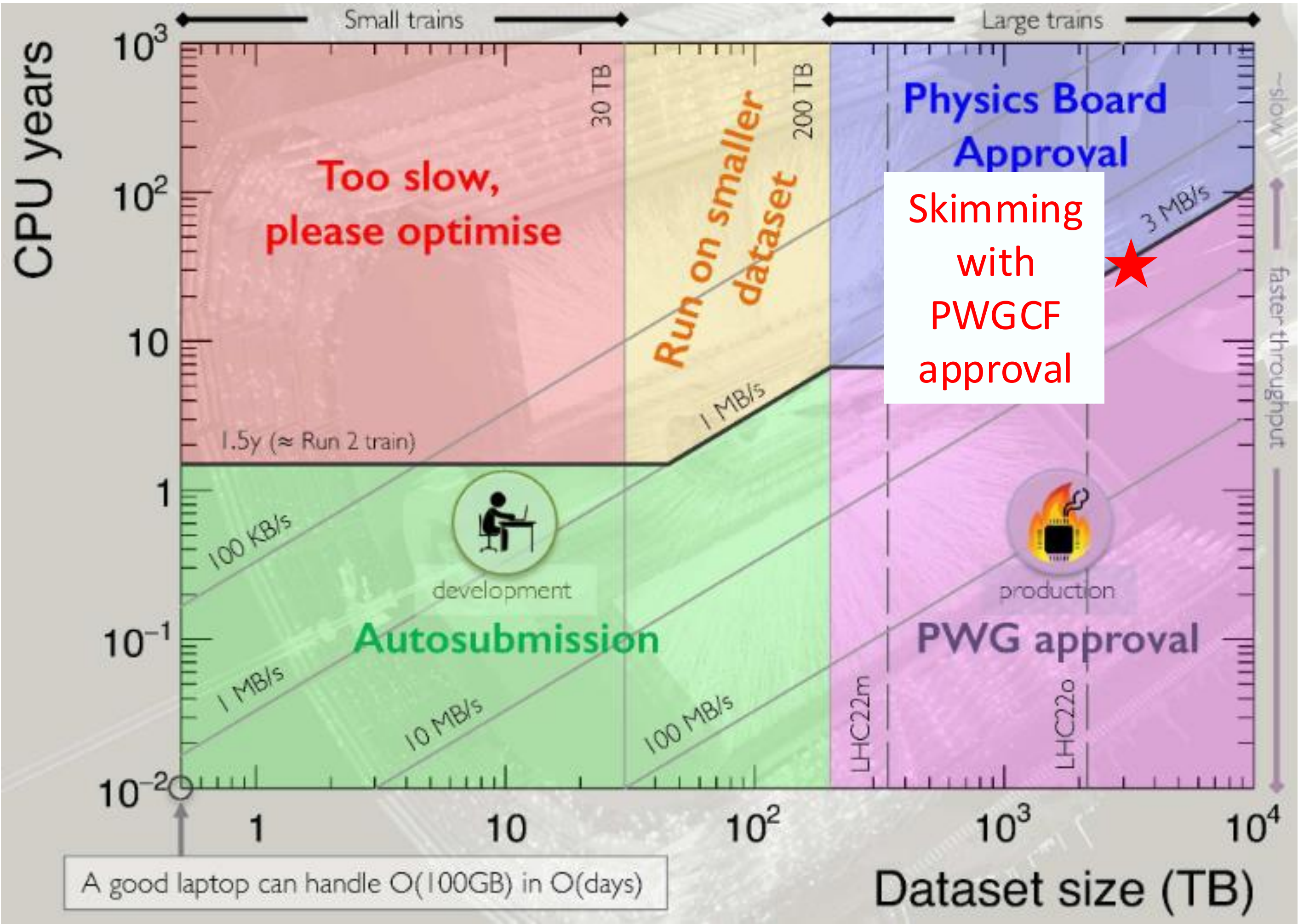
Number of input files	1
Input size	2.4 GB
Output size	12.4 MB
Output size (AO2D only)	12.3 MB
Reduction Factor	196
PSS Memory	Max: 3.7 GB Avg: 3.6 GB Slope: 1.4 MB/s
Private Memory	Max: 2.6 GB Avg: 2.4 GB Slope: 594.1 KB/s
Timing	CPU: 9m 40s Wall: 12m 44s
Throughput	3.2 MB/s/core
Expected resources	33y 291d 16.4 TB



Skimming & Analysis of whole 22 data (3PB)

Hyperloop test of femtodream producer task

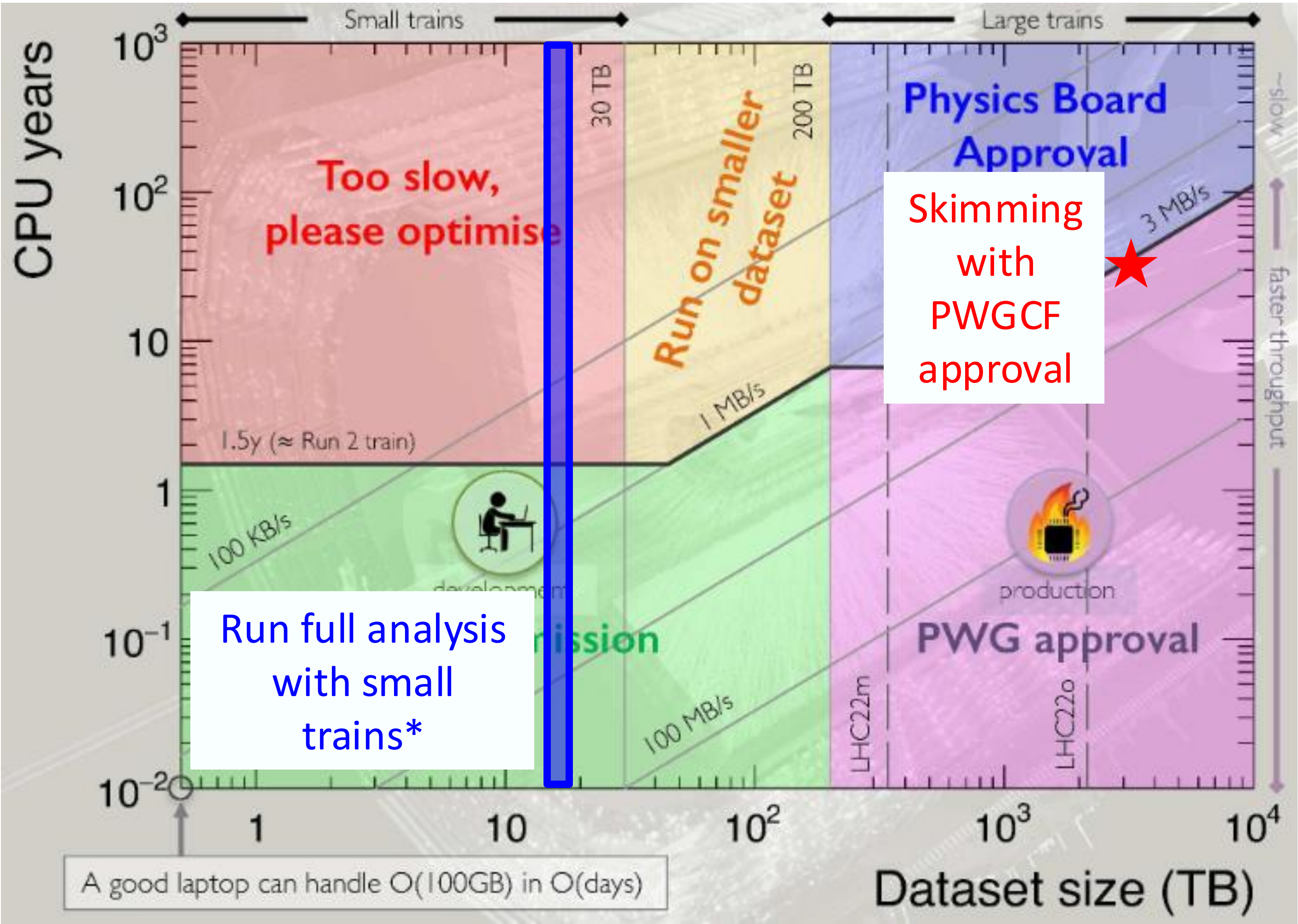
Number of input files	1
Input size	2.4 GB
Output size	12.4 MB
Output size (AO2D only)	12.3 MB
Reduction Factor	196
PSS Memory	Max: 3.7 GB Avg: 3.6 GB Slope: 1.4 MB/s
Private Memory	Max: 2.6 GB Avg: 2.4 GB Slope: 594.1 KB/s
Timing	CPU: 9m 40s Wall: 12m 44s
Throughput	3.2 MB/s/core
Expected resources	33y 291d 16.4 TB



Skimming & Analysis of whole 22 data (3PB)

Hyperloop test of femtodream producer task

Number of input files	1
Input size	2.4 GB
Output size	12.4 MB
Output size (AO2D only)	12.3 MB
Reduction Factor	196
PSS Memory	Max: 3.7 GB Avg: 3.6 GB Slope: 1.4 MB/s
Private Memory	Max: 2.6 GB Avg: 2.4 GB Slope: 594.1 KB/s
Timing	CPU: 9m 40s Wall: 12m 44s
Throughput	3.2 MB/s/core
Expected resources	33y 291d 16.4 TB



Bitmask for track cuts

dpl-config.json

```
"ConfTrkPtMin": {
    "values": [
        "0.5",
        "0.4",
        "0.6"
    ]
},
"ConfTrkPtMax": {
    "values": [
        "4"
    ]
},
"ConfTrkEtaMax": {
    "values": [
        "0.8",
        "0.77",
        "0.83"
    ]
},
```

O2track from AO2D.root

O2Tracks from AO2D.root			
Index	1	2	…
p_T	0.55	0.42	…
eta	0.79	0.9	…
…	…	…	…

Index	p_T min			p_T max	|eta| max			…
	>0.4	>0.5	>0.6	<4	<0.83	<0.8	<0.77	…
1								…
2								…

Bitmask for track cuts

dpl-config.json

```
"ConfTrkPtMin": {
    "values": [
        "0.5",
        "0.4",
        "0.6"
    ]
},
"ConfTrkPtMax": {
    "values": [
        "4"
    ]
},
"ConfTrkEtaMax": {
    "values": [
        "0.8",
        "0.77",
        "0.83"
    ]
},
```

O2track from AO2D.root

O2Tracks from AO2D.root			
Index	1	2	…
p_T	0.55	0.42	…
eta	0.79	0.9	…
…	…	…	…

Track 1 passed pT>0.5, set bit to 1

Track 2 did not pass pT>0.5, set bit to 0

Index	p_T min			p_T max	|eta| max			…
	>0.4	>0.5	>0.6	<4	<0.83	<0.8	<0.77	…
1	1	1	0					…
2	1	0	0					…

Bitmask for track cuts

dpl-config.json

```
"ConfTrkPtMin": {
    "values": [
        "0.5",
        "0.4",
        "0.6"
    ]
},
"ConfTrkPtMax": {
    "values": [
        "4"
    ]
},
"ConfTrkEtaMax": {
    "values": [
        "0.8",
        "0.77",
        "0.83"
    ]
},
```

O2track from AO2D.root

O2Tracks from AO2D.root			
Index	1	2	…
p_T	0.55	0.42	…
eta	0.79	0.9	…
…	…	…	…

Index	p_T min			p_T max	|eta| max			…
	>0.4	>0.5	>0.6	<4	<0.83	<0.8	<0.77	…
1	1	1	0	1				…
2	1	0	0	1				…

Bitmask for track cuts

dpl-config.json

```
"ConfTrkPtMin": {
    "values": [
        "0.5",
        "0.4",
        "0.6"
    ]
},
"ConfTrkPtMax": {
    "values": [
        "4"
    ]
},
"ConfTrkEtaMax": {
    "values": [
        "0.8",
        "0.77",
        "0.83"
    ]
},
```

O2track from AO2D.root

O2Tracks from AO2D.root			
Index	1	2	…
p_T	0.55	0.42	…
eta	0.79	-0.9	…
…	…	…	…

Index	p_T min			p_T max	|eta| max			…
	>0.4	>0.5	>0.6	<4	<0.83	<0.8	<0.77	…
1	1	1	0	1	1	1	0	…
2	1	0	0	1	0	0	0	…

Bitmask for track cuts

- Use cutculator in femtodream to compute the bitmask for the selections you need
- Cutculator takes as input the configuration of your producer task

```
[O2Physics/latest] ~ $> o2-analysis-cf-femtodream-cutculator dpl-config.json
Welcome to the CutCulator!
Found femto-dream-producer-task in dpl-config.json
Do you want to work with tracks or V0s (T/V)? >T
Do you want to manually select cuts or create systematic variations(M/V)? >M
Selection: Sign of the track - (-1 1 )
 > 1
Selection: Minimal pT (GeV/c) - (0.4 0.5 0.6 )
 > 0.5
Selection: Maximal pT (GeV/c) - (4 )
 > 4
Selection: Maximal eta - (0.83 0.8 0.77 )
 > 0.8
```

Invoke cutculator with the configuration of your producer as input

Bitmask for track cuts

- Use cutculator in femtodream to compute the bitmask for the selections you need
- Cutculator takes as input the configuration of your producer task

```
[O2Physics/latest] ~ $> o2-analysis-cf-femtodream-cutculator dpl-config.json
Welcome to the CutCulator!
Found femto-dream-producer-task in dpl-config.json
Do you want to work with tracks or V0s (T/V)? >T
Do you want to manually select cuts or create systematic variations(M/V)? >M
Selection: Sign of the track - (-1 1 )
 > 1
Selection: Minimal pT (GeV/c) - (0.4 0.5 0.6 )
 > 0.5
Selection: Maximal pT (GeV/c) - (4 )
 > 4
Selection: Maximal eta - (0.83 0.8 0.77 )
 > 0.8
```

Select the cuts you want
from the available options

Bitmask for track cuts

- Use cutculator in femtodream to compute the bitmask for the selections you need
- Cutculator takes as input the configuration of your producer task

```
[O2Physics/latest] ~ $> o2-analysis-cf-femtodream-cutculator dpl-config.json
Welcome to the CutCulator!
Found femto-dream-producer-task in dpl-config.json
Do you want to work with tracks or V0s (T/V)? >T
Do you want to manually select cuts or create systematic variations(M/V)? >M
Selection: Sign of the track - (-1 1 )
 > 1
Selection: Minimal pT (GeV/c) - (0.4 0.5 0.6 )
 > 0.5
Selection: Maximal pT (GeV/c) - (4 )
 > 4
Selection: Maximal eta - (0.83 0.8 0.77 )
 > 0.8
```

```
+++++++++++++++++++++++++++++++++
CutCulator has spoken - your selection bit is
00000001100000000111010010101010 (bitwise)
25195690 (number representation)
PID for these species is stored:
Proton : 0
+++++++++++++++++++++++++++++++++
```

Use the bitmask as input for your pair task to select tracks

Configuration of pair-task-track-track

ConfCutPartOne	25195690
ConfCutPartTwo	25195690

Bitmask for PID cuts

$$n\sigma_{TPCTOF} = \sqrt{n\sigma^2_{TPC} + n\sigma^2_{TOF}}$$

dpl-config.json
(Producer)

```
"ConfTrkPIDnSigmaMax": {
    "values": [
        "3",
        "2.5",
        "3.5"
    ]
},
```

PID information of tracks from helper tasks			
Index	1	2	…
NsigmaTPC (Proton)	1.2	2.8	…
NSigmaTOF (Proton)	0.5	5.6	…
…	…	…	…

dpl-config.json
(Producer)

```
"ConfTrkPIDspecies": {
    "values": [
        "4"
    ]
},
```

Species	Proton					
	3		2.5		3.5	
Index	<|TPC|	<|TPCTOF|	<|TPC|	<|TPCTOF|	<|TPC|	<|TPCTOF|
1						
2						

Bitmask for PID cuts

$$n\sigma_{TPCTOF} = \sqrt{n\sigma_{TPC}^2 + n\sigma_{TOF}^2}$$

dpl-config.json
(Producer)

```
"ConfTrkPIDnSigmaMax": {
    "values": [
        "3",
        "2.5",
        "3.5"
    ]
},
```

PID information of tracks from helper tasks			
Index	1	2	…
NsigmaTPC (Proton)	1.2	2.8	…
NSigmaTOF (Proton)	0.5	5.6	…
…	…	…	…

dpl-config.json
(Producer)

```
"ConfTrkPIDspecies": {
    "values": [
        "4"
    ]
},
```

from PID.h

```
static constexpr ID Electron = 0;
static constexpr ID Muon = 1;
static constexpr ID Pion = 2;
static constexpr ID Kaon = 3;
static constexpr ID Proton = 4;
static constexpr ID Deuteron = 5;
static constexpr ID Triton = 6;
static constexpr ID Helium3 = 7;
static constexpr ID Alpha = 8;
```

Species	Proton					
	3		2.5		3.5	
Index	<|TPC|	<|TPCTOF|	<|TPC|	<|TPCTOF|	<|TPC|	<|TPCTOF|
1						
2						

Bitmask for PID cuts

$$n\sigma_{TPCTOF} = \sqrt{n\sigma^2_{TPC} + n\sigma^2_{TOF}}$$

dpl-config.json
(Producer)

```
"ConfTrkPIDnSigmaMax": {
    "values": [
        "3",
        "2.5",
        "3.5"
    ]
},
```

PID information of tracks from helper2222 tasks			
Index	1	2	...
NsigmaTPC (Proton)	2.8	-3.2	...
NSigmaTOF (Proton)	0.5	5.6	...
...	...	...	...

Track 1 passed
|nsigma_TPC|<3, set bit to 1

Track 2 did not pass
|nsigma_TPC|<3, set bit to 0

dpl-config.json
(Producer)

```
"ConfTrkPIDspecies": {
    "values": [
        "4"
    ]
},
```

Species	Proton					
	3		2.5		3.5	
Index	<|TPC|	<|TPCTOF|	<|TPC|	<|TPCTOF|	<|TPC|	<|TPCTOF|
1	1		0		1	
2	0		0		1	

Bitmask for PID cuts

$$n\sigma_{TPCTOF} = \sqrt{n\sigma^2_{TPC} + n\sigma^2_{TOF}}$$

dpl-config.json (Producer)

```
"ConfTrkPIDnSigmaMax": {
    "values": [
        "3",
        "2.5",
        "3.5"
    ]
},
```

PID information of tracks from helper2222 tasks			
Index	1	2	...
NsigmaTPC (Proton)	2.8	-3.2	...
NSigmaTOF (Proton)	-0.5	5.6	...
	...	...	...

Track 1 passed |nsigma_TPCTOF|<3, set bit to 1

Track 2 did not pass |nsigma_TPCTOF|<3, set bit to 0

dpl-config.json (Producer)

```
"ConfTrkPIDspecies": {
    "values": [
        "4"
    ]
},
```

Species	Proton					
	3		2.5		3.5	
Index	<|TPC|	<|TPCTOF|	<|TPC|	<|TPCTOF|	<|TPC|	<|TPCTOF|
1	1	1	0	0	1	1
2	0	0	0	0	1	0

Support for TOF only selections will be added in the future

Bitmask for pid cuts

$$n\sigma_{TPCTOF} = \sqrt{n\sigma_{TPC}^2 + n\sigma_{TOF}^2}$$

- Cutculator will compute all possible PID selections

```
PID bits for these species are available:
+++++++++++++++++++++++++++++++++
Species Proton with |NSigma|<3.5
Bit for Nsigma TPC: 65536
Bit for Nsigma TPCTOF: 32768
+++++++++++++++++++++++++++++++++
Species Proton with |NSigma|<3
Bit for Nsigma TPC: 4096
Bit for Nsigma TPCTOF: 2048
+++++++++++++++++++++++++++++++++
Species Proton with |NSigma|<2.5
Bit for Nsigma TPC: 256
Bit for Nsigma TPCTOF: 128
```

Bitmask for pid cuts

$$n\sigma_{TPCTOF} = \sqrt{n\sigma^2_{TPC} + n\sigma^2_{TOF}}$$

- Cutculator will compute all possible PID selections

```
PID bits for these species are available:
+++++++++++++++++++++++++++++++++
Species Proton with |NSigma|<3.5
Bit for Nsigma TPC: 65536
Bit for Nsigma TPCTOF: 32768
+++++++++++++++++++++++++++++++++
Species Proton with |NSigma|<3
Bit for Nsigma TPC: 4096
Bit for Nsigma TPCTOF: 2048
+++++++++++++++++++++++++++++++++
Species Proton with |NSigma|<2.5
Bit for Nsigma TPC: 256
Bit for Nsigma TPCTOF: 128
```

- Select particle species of interest

Bitmask for pid cuts

$$n\sigma_{TPCTOF} = \sqrt{n\sigma^2_{TPC} + n\sigma^2_{TOF}}$$

- Cutculator will compute all possible PID selections

```
PID bits for these species are available:
+++++++++++++++++++++++++++++++++
Species Proton with |NSigma|<3.5
Bit for Nsigma TPC: 65536
Bit for Nsigma TPCTOF: 32768
+++++++++++++++++++++++++++++++++
Species Proton with |NSigma|<3
Bit for Nsigma TPC: 4096
Bit for Nsigma TPCTOF: 2048
+++++++++++++++++++++++++++++++++
Species Proton with |NSigma|<2.5
Bit for Nsigma TPC: 256
Bit for Nsigma TPCTOF: 128
```

- Select particle species of interest
- Select limit of $|n\sigma|$

Bitmask for pid cuts

$$n\sigma_{TPCTOF} = \sqrt{n\sigma^2_{TPC} + n\sigma^2_{TOF}}$$

- Cutculator will compute all possible PID selections

```
PID bits for these species are available:
+++++++++++++++++++++++++++++++++
Species Proton with |NSigma|<3.5
Bit for Nsigma TPC: 65536
Bit for Nsigma TPCTOF: 32768
+++++++++++++++++++++++++++++++++
Species Proton with |NSigma|<3
Bit for Nsigma TPC: 4096
Bit for Nsigma TPCTOF: 2048
+++++++++++++++++++++++++++++++++
Species Proton with |NSigma|<2.5
Bit for Nsigma TPC: 256
Bit for Nsigma TPCTOF: 128
```

- Select particle species of interest
- Select limit of |nσ|
- Select limit for TPC or TPCTOF

Bitmask for pid cuts

$$n\sigma_{TPCTOF} = \sqrt{n\sigma^2_{TPC} + n\sigma^2_{TOF}}$$

- Cutculator will compute all possible PID selections

```
PID bits for these species are available:
+++++++++++++++++++++++++++++++++
Species Proton with |NSigma|<3.5
Bit for Nsigma TPC: 65536
Bit for Nsigma TPCTOF: 32768
+++++++++++++++++++++++++++++++++
Species Proton with |NSigma|<3
Bit for Nsigma TPC: 4096
Bit for Nsigma TPCTOF: 2048
+++++++++++++++++++++++++++++++++
Species Proton with |NSigma|<2.5
Bit for Nsigma TPC: 256
Bit for Nsigma TPCTOF: 128
```

- Select particle species of interest
- Select limit of |nσ|
- Select limit for TPC or TPCTOF
- Read of the bitmask

Bitmask for pid cuts

$$n\sigma_{TPCTOF} = \sqrt{n\sigma^2_{TPC} + n\sigma^2_{TOF}}$$

- Cutculator will compute all possible PID selections

```
PID bits for these species are available:
+++++++++++++++++++++++++++++++++
Species Proton with |NSigma|<3.5
Bit for Nsigma TPC: 65536
Bit for Nsigma TPCTOF: 32768
+++++++++++++++++++++++++++++++++
Species Proton with |NSigma|<3
Bit for Nsigma TPC: 4096
Bit for Nsigma TPCTOF: 2048
+++++++++++++++++++++++++++++++++
Species Proton with |NSigma|<2.5
Bit for Nsigma TPC: 256
Bit for Nsigma TPCTOF: 128
```

- Select particle species of interest
- Select limit of |nσ|
- Select limit for TPC or TPCTOF
- Read of the bitmask
- Configure the task

Momentum threshold to switch from TPC to TPCTOF

Track1.PIDThres	0.75

$$n\sigma_{TPCTOF} = \sqrt{n\sigma^2_{TPC} + n\sigma^2_{TOF}}$$

PP TASK FORCE

ALICE

Bitmask for pid cuts

- Cutculator will compute all possible PID selections

```
PID bits for these species are available:
+++++++++++++++++++++++++++++++++
Species Proton with |NSigma|<3.5
Bit for Nsigma TPC: 65536
Bit for Nsigma TPCTOF: 32768
+++++++++++++++++++++++++++++++++
Species Proton with |NSigma|<3
Bit for Nsigma TPC: 4096
Bit for Nsigma TPCTOF: 2048
+++++++++++++++++++++++++++++++++
Species Proton with |NSigma|<2.5
Bit for Nsigma TPC: 256
Bit for Nsigma TPCTOF: 128
```

Here: Select protons with |nσ|< 3 for TPC and TPCTOF with a momentum threshold of 0.75 GeV

- Select particle species of interest
- Select limit of |nσ|
- Select limit for TPC or TPCTOF
- Read of the bitmask
- Configure the task

Important: Set momentum threshold to switch from TPC to TPCTOF

Track1.PIDThres	0.75
Track1.TPCBit	4096
Track1.TPCBit_Reject	0
Track1.TPCTOFBit	2048

How do I know that my bitmasks are correct?

- Full chain of femto analysis

AO2D.root

ROOT file(s)

DF:
- O2collision
- O2track
- O2muon
- ...

Producer task

Store	Binding	Description
☐	FDExtMCParticles	FDEXTMCPARTICLE
☑	FDExtParticles	FDEXTPARTICLE
☐	FDMCLabels	FDMCLabel
☐	FDMCParticles	FDMCPARTICLE
☑	FDCollisions	FDCOLLISION
☑	FDParticles	FDPARTICLE

```
o2-analysis-cf-femtodream-producer
```

- Fill femto tables for analysis
- Only store events and tracks/V0/… fulfilling loosest selections
- Compute bitmask for track and PID selections
- **Generate extended and/or MC tables for QA and more**

How do I know that my bitmasks are correct?

- Full chain of femto analysis

AO2D.root

ROOT file(s)

DF:
- O2collision
- O2track
- O2muon
- ...

Producer task

Store	Binding	Description
☐	FDExtMCParticles	FDEXTMCPARTICLE
☑	FDExtParticles	FDEXTPARTICLE
☐	FDMCLabels	FDMCLabel
☐	FDMCParticles	FDMCPARTICLE
☑	FDCollisions	FDCOLLISION
☑	FDParticles	FDPARTICLE

`o2-analysis-cf-femtodream-producer`

- Fill femto tables for analysis
- Only store events and tracks/V0/… fulfilling loosest selections
- Compute bitmask for track and PID selections
- **Generate extended and/or MC tables for QA and more**

Debug task

AnalysisResults.root

- femto-dream-debug-track;1
  - Event;1
  - Tracks;1
    - hPt;1
    - hEta;1
    - hPhi;1
    - hDCAxy;1
    - hMomentumVsEta;1
    - hMomentumVsPhi;1
    - hEtaVsPhi;1
    - hCharge;1
    - hTPCfindable;1
    - hTPCfound;1
    - hTPCcrossedOverFindable;1
    - hTPCcrossedRows;1
    - ...

Distributions with track properties and PID

`o2-analysis-cf-femtodream-debug-track`

- **Select particles according to the bitmask**
- Fill distributions of all available track properties and PID for all supported particle species
- **Check distributions to see if bitmask is applied correctly**

Continued development

- Internals of FemtoDream still very much under active development
  - Performance optimizations
  - Reduction in data size
  - Support for cascades/resonances/…
  - More modular design of the producer task
  - Generic correlator tasks for track-cascade, V0-cascade, cascade-cascade, …

- Unification/Update of exiting data formats

HANDS-ON START

Exercises for the femto tutorial can be found here:

```
[anton@silver] ~/alice/O2Physics
> ls Tutorials/PWGCF/FemtoFramework/src/*
Tutorials/PWGCF/FemtoFramework/src/CFTutorialTask1.cxx
Tutorials/PWGCF/FemtoFramework/src/CFTutorialTask2.cxx
Tutorials/PWGCF/FemtoFramework/src/CFTutorialTask3.cxx
Tutorials/PWGCF/FemtoFramework/src/CFTutorialTask4.cxx
Tutorials/PWGCF/FemtoFramework/src/CFTutorialTask5.cxx
Tutorials/PWGCF/FemtoFramework/src/README.md
Tutorials/PWGCF/FemtoFramework/src/run.sh
```

The source files contain comments on what needs to be implemented

HANDS-ON START

GOAL: Implement an analysis task in O2Physics for pairing tracks to compute same and mixed event distributions using derived data as input (similar to the pair-track-track task that is already available in femtodream)

There are 5 exercise Tasks:

- **Task 1:**
    - Implement filters for tracks and events
- **Task 2:**
    - Implement track selections using bitmasks (-> Slide 14-20)
- **Task 3:**
    - Implement PID selection using bitmasks (-> Slide 21-31)
- **Task 4:**
    - Implement particle pairing in same events to compute same event distribution
- **Task 5:**
    - Implement event mixing and particle pairing across multiple events to compute the mixed event distribution

Tasks build on top of each other, meaning the solution of Task 1 is the starting point of Task 2 and so on