

Stripping framework

Anton Poluektov

The University of Warwick, UK,
Budker Institute of Nuclear Physics, Novosibirsk, Russia

What is needed from stripping framework:

- Keep physics selections "under the same roof"
- Control streams and selections
- Control DST and ETC writing
- Tools to debug and monitor selections

A good starting point is Hlt2 based on Lines.

The first release of stripping framework is available in DaVinci v23r1.

- Phys/StrippingConf
 - Python configuration classes (StrippingConf, StrippingLine, StrippingMember)
 - C++ configurable (StrippingAlg)
- Phys/StrippingSelections
 - Placeholder of stripping selections (StrippingSelections/options/)
 - Examples of selections and DaVinci jobs (StrippingSelections/tests/)

Do "getpack Phys/StrippingSelections" first.

Consider the sequencer that does $B \rightarrow Dh$ selection is there (B2DhFilterSequence). Then:

```
line = StrippingLine('B2Dh'  
    , prescale = 1  
    , algos = [ B2DhFilterSequence ]  
    , postscale = 1  
    , stream = 'BExclusive'  
)
```

will integrate the selection into stripping framework. This will create a line corresponding to B2DhFilterSequence selection, and add it to the list of lines. The selection will be assigned to the stream "BExclusive".

Selections are stored in \$STRIPPINGSELECTIONSROOT/options/. The new selection has to be added (via importOptions) to \$STRIPPINGSELECTIONSROOT/options/StrippingSelections.py (this is the main file loaded by default).

StrippingMember is implemented similarly to Hlt2Member. Cloning of lines may be convenient for e. g. signal and sidebands selections. Example:

```
combine = CombineParticles("Bd2KstarMuMuCombine")

filter = StrippingMember (FilterDesktop
    , "Filter"
    , InputLocations = ["Bd2KstarMuMuCombine"]
    , Code = "BLABLA_1"
)

line1 = StrippingLine('B2DPi_signal'
    , prescale = 1
    , algos = [ combine , filter ]
    , stream = 'BExclusive'
)

line2 = line1.clone('B2Dpi_sideband'
    , prescale = 0.2
    , FilterDesktopFilter = {"Code" : "BLABLA_2"}
)
```

No `SharedParticles` functionality as in the case of `Hlt2` is implemented in stripping framework.

It is intended that common states such as $K^* \rightarrow K\pi$ or $\phi \rightarrow KK$ will go into `CommonParticles` package to avoid running duplicate combinatorics.

If the existing `Std...` state is not loose enough for your purpose, you are welcome to add it to `CommonParticles` (work coordinated by Marta Calvi).

These are the minimum options needed to run stripping under DaVinci:

```
from Configurables import DaVinci
from StrippingConf.Configuration import StrippingConf

StrippingConf().OutputType = "ETC"    # or "DST" or "NONE"

DaVinci().EvtMax = 200
DaVinci().DataType = "2008"
DaVinci().Simulation = True
DaVinci().ETCFile = "etc.root"
```

This will load all selections from Phys/StrippingSelections and produce an ETC file called "etc.root" with selection results.

Alternatively,

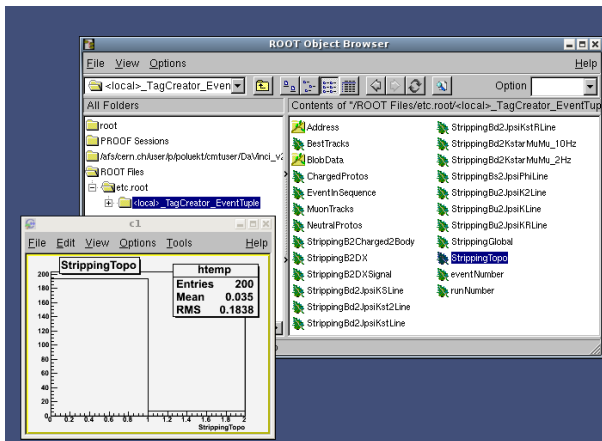
```
StrippingConf().OutputType = "DST"
```

will produce a separate DST file for each stream (e. g. BExclusive.dst for stream named "BExclusive").

StrippingConf options

- `StrippingConf().MainOptions = "path/selections.py"`
Main options file to import.
- `StrippingConf().ActiveLines = ["B2HH", "B2Dpi"]`
List of active lines (take all if empty). Can be useful for debugging a specific selection.
- `StrippingConf().ActiveStreams = ["MuonPair", "DStar"]`
List of active streams (take all if empty).
- `StrippingConf().DSTPrefix = "test1_"`
Prefix for DST files.
- `StrippingConf().OutputType = "ETC"`
Output type: "ETC", "DST", or "NONE"

Selection decisions are stored in the ETC file under names "Stripping[LineName]" (e. g. "StrippingB2DH").



Eventually, there will be a "StrippingGlobal" decision (logical OR of all lines) and ORs of selections for each stream.

Selections under `$STRIPPINGSELECTIONS/options/`:

- B2Charged2Body by Angelo Carbone.
- B2DH by Vava Gligorov and Jacopo Nardulli.
- Bd2JpsiKstar, Bd2JpsiKS, Bs2JpsiPhi, Bu2JpsiK by Greig Cowan.
- Bd2KstarMuMu by Rob Lambert.
- Topological (2, 3, and 4-body) by Vava Gligorov.
- Bu2JpsiK, Bd2JpsiKstar by Diego Martinez Santos (not in the release yet).

There are also versions of these selections modified for STEP09 under `$STRIPPINGSELECTIONS/options/STEP09` (Bd2KstarMuMu and B2Charged2Body selections relaxed to ensure reasonable rate for L0 minimum bias).

- First version of stripping framework is available. Some selections are implemented in Phys/StrippingSelections.
- Definitely not all functionality is there yet (monitoring, debugging). Suggestions are welcome.
- See TWiki page for more information:
<https://twiki.cern.ch/twiki/bin/view/LHCb/StrippingLines>