Introduction to DaVinci 3

- Standard Code
- Standard Particles

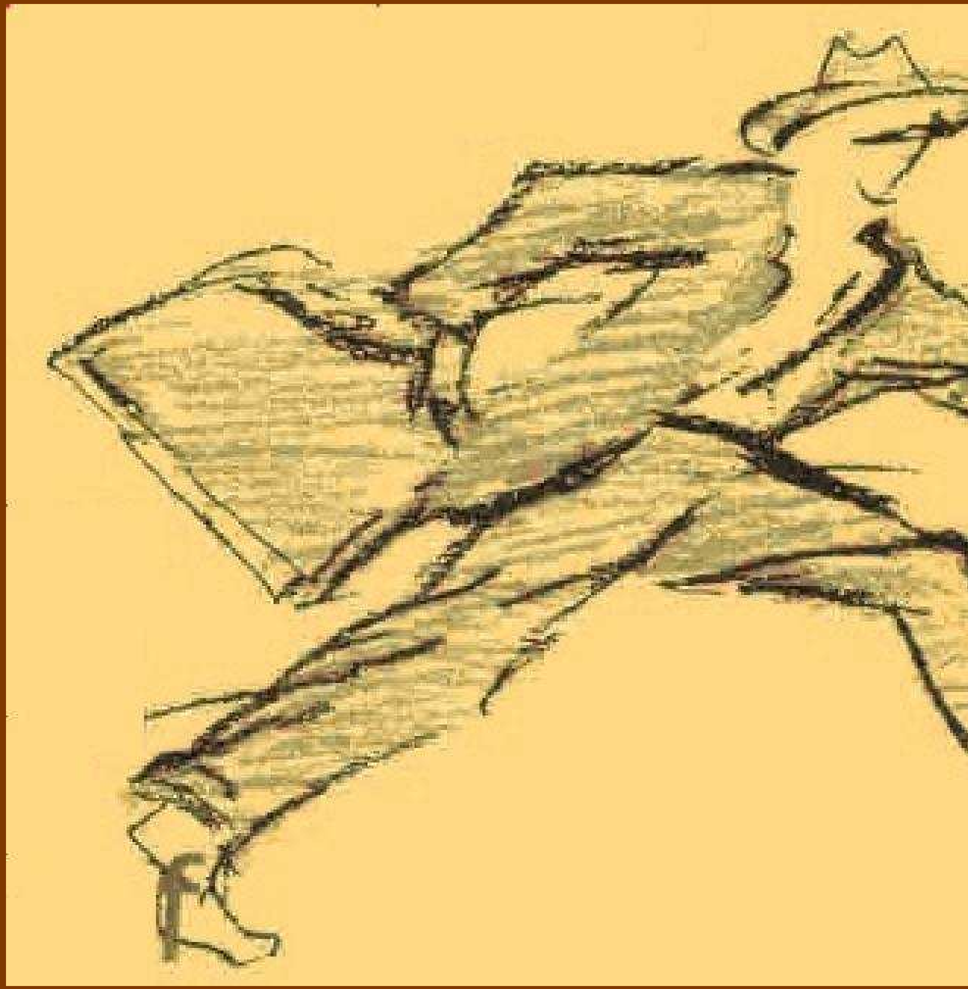
June 2006 Bologna Software Course

Patrick Koppenburg
Imperial College
London

Standards:

- Standard Code
- Filters
- More

Exercises are provided in the `Tutorial/Analysis` package.



Introduction

There are several ways to quickly get a physics result:

Plain C++: `DVAlgorithm` inherits from `GaudiAlgorithm` (and `GaudiTupleAlg` ...), some typing is saved

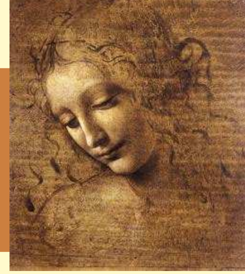
`LoKi`: “loops and kinematics”. Templated `C++`. More typing saved.

Bender: Interactive `python`.

Generic algorithms: The subject of this talk.

The common assumption is that physicists always do the same, hence any line of `C++` you type is a duplication of what your office-mate is typing right now.

Introduction (2)



If you did Exercise 3, and wrote an algorithm that can select $\mathbf{J}/\psi \rightarrow \mu\mu$ and $\phi \rightarrow \mathbf{KK}$ you probably thought:

> I could be even smarter and write an algorithm that can do the $\mathrm{B_s} \rightarrow \mathbf{J}/\psi\phi$ too . . . There must be a way to handle everything in a generic way.

Yes there is. . . And you're not the first one to think about that

The basic idea

Any ($\mathbf{B}$-) physics analysis is a sequence of A $\rightarrow$ B C (...), with some cuts in between.

Hence: An analysis algorithm should know:

1. Where to get the particles
2. What decay to reconstruct
3. What cuts to apply
4. Where to put the data

Anything else?

The basic idea

Any ($\mathbf{B}$-) physics analysis is a sequence of A $\rightarrow$ B C (...), with some cuts in between.

Hence: An analysis algorithm should know:

1. Where to get the particles
2. What decay to reconstruct
3. What cuts to apply
4. Where to put the data

I/O is handled by the `PhysDesktop`:

```
DsForBs2DsPi.PhysDesktop.InputLocations = { "Phys/Combined" };
```

The basic idea

Any (B-) physics analysis is a sequence of A → B C (. . .), with some cuts in between.

He

1.

2.

3.

4.

```
for { LHCb::Particle::ConstVector::const_iterator mK = KMinus.begin() ;
      mK != KMinus.end() ; ++mK ){
  for { LHCb::Particle::ConstVector::const_iterator pK = KPlus.begin() ;
        pK != KPlus.end() ; ++pK ){
    for { LHCb::Particle::ConstVector::const_iterator pi = Pions.begin()
          pi != Pions.end() ; ++pi ){
             [...]
```

Can be shorter:

```
for ( Loop Ds = loop( "K K pi", "D_s+", FitVertex ) ; Ds ; ++Ds ) {
```

Or:

```
DsForBs2DsPi.DecayDescriptor = "[D_s+ -> K+ K- pi+]cc"
```

Anything else?

The basic idea

Any (B-) physics analysis is a sequence of A → B C (. . .), with some cuts in between.

Hence: An analysis algorithm should know:

1. Where to get the particles
2. What decay to reconstruct
3. What cuts to apply
4.

Hard-coding cuts is a bad idea. . .

Better to use options of the algorithm

. . . or predefined filters configurable by options:

```
DsForBs2DsPi.MotherFilter.VtxFilterCriterion.MaxChi2 = 20.;
```

First implementations by Gerhard

`Select2ParticleDecay` (Jul.02): Makes decays to two (and more) particles. Used in DC03 for $\mathbf{B_s} \rightarrow \mathbf{J}/\psi\phi$

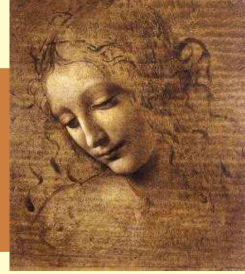
`RefineSelection` (Dec.03): Allows to “refine” a set of particles applying cuts.

`CombineParticles` (Jul.04): Replaces `Select2ParticleDecay`. Better syntax of options.

We decided to use them in the exclusive HLT.
Then it appeared that:

- Options syntax is incompatible between `RefineSelection` and `CombineParticles` $\rightarrow$ reshuffling of options is not straight-forward.
- `CombineParticles` is too slow: vertex fitting is applied before the mass cut.

Present implementation



`MakeResonances`: Yet another `CombineParticles`.

`FilterDesktop`: A `RefineSelection` with a similar syntax than `MakeResonances`

`ByPIDFilterCriterion`: Used by `MakeResonances` and `FilterDesktop`. That's where all the cuts are applied.

`IPlotTool`: They use `IPlotTool` for a quick plotting of some variables. Two implementations:

- **`SimplePlotTool`:** Plots any given set of variables for any particle
- **`RecursivePlotTool`:** Calls `SimplePlotTool` for each particle and it's daughters recursively.

All this is used since December 2004 in the HLT.
Since May 2006 it is used in 27/29 preselections.

A simple example

This framework has first been used offline for the $\mathbf{B^0} \rightarrow \mathbf{D^-}\pi$ preselection

```
ApplicationMgr.TopAlg+={"GaudiSequencer/SeqPreselBd2DPi"};
SeqPreselBd2DPi.Members += {"MakeResonances/DForPreselBd2DPi",
                            "MakeResonances/PreselBd2DPi"};
```

- Two instances of `MakeResonances`, one for the $\mathbf{D^-}$ and one for the $\mathbf{B^0}$.

1. The decay $D^- \to \pi^-\pi^-K^+$

```
DForPreselBd2DPi.DecayDescriptor = "[D- -> pi- pi- K+]cc" ;
DForPreselBd2DPi.Window = 50.*MeV ;
```

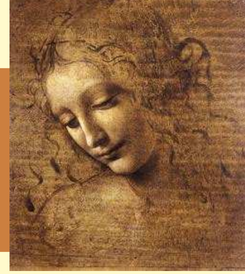
That's all! This makes all $\boldsymbol{\pi^-\pi^-K^+}$ combinations in a mass window of $\pm 50\ \mathrm{MeV}$ around the $\mathbf{D^-}$.

- The `DecayDescriptor` does not use the MC truth decay descriptor, but understands only simple things. Don't try

```
[B_s0 -> (J/psi(1S) -> mu+ mu- {,gamma} {,gamma})
         (eta -> pi+ pi- ( pi0 -> gamma gamma)]cc
```

- The mass window is applied on the sum of 4-vectors before vertex fitting
- The default vertex fitter is used. Change it if you like.

2. Some cuts



`MakeResonances` has two `ByPIDFilterCriterion` tools, one for the daughters, one for the mother.

Here are the cuts for the daughters:

```
DForPreselBd2DPi.DaughterFilter.Selections = {
                "K+  : KinFilterCriterion, PVIPFilterCriterion",
                "pi+ : KinFilterCriterion, PVIPFilterCriterion"};
DForPreselBd2DPi.DaughterFilter.KinFilterCriterion.MinPt = 300.*MeV ;
DForPreselBd2DPi.DaughterFilter.PVIPFilterCriterion.MinIPsignif = 1. ;
```

These options apply a $300\ \mathrm{MeV}\ p_T$ and a 1σ IP cuts to the kaons and pions before making the $\mathbf{D}^-$.

2. Some cuts

What if one wants to apply different cuts for the π and the $\mathbf{K}$?

```
DForPreselBd2DPi.DaughterFilter.Selections =
                              {  "K+   : KinFilterCriterion/KKin,
                                        PVIPFilterCriterion/KPVIP",
                                 "pi+  : KinFilterCriterion/PiKin,
                                        PVIPFilterCriterion/PiPVIP"};
DForPreselBd2DPi.DaughterFilter.KKin.MinPt = 300.*MeV ;
DForPreselBd2DPi.DaughterFilter.KPVIP.MinIPsignif = 1. ;
DForPreselBd2DPi.DaughterFilter.PiKin.MinPt = 500.*MeV ;
DForPreselBd2DPi.DaughterFilter.PiPVIP.MinIPsignif = 2. ;
```

Just rename the tool instances!

ByPIDFilterCriterion reference

Applies a list of `IFilterCriteron` (only “and” mode available) to a list of particles according to their PID.

Selections: a vector of strings with syntax `"Particle : FilterCriterion, FilterCriterion..."`

ApplyCC (Default: true): Apply same cuts to the anti-particle.

ExclusiveSelection (Default: false): If true, reject particles that do not appear in the list given in `Selections`. Else keep.

FilterByDescendents (Default: false): Filter composite particles according to criteria given for its descendants

```
FilterD0ForHLT.Filter.Selections = {
        "D0 : VtxFilterCriterion",
        "K+ : KinFilterCriterion", "pi- : KinFilterCriterion" };
FilterD0ForHLT.Filter.FilterByDescendents = true ; // use pi and K
```

All other `IFilterCriterion`

`BooleanFilterCriterion`: Combines them

`ByPIDFilterCriterion`: Combined them by PID

`DaughterMomentumSumFilterCriterion`: Sum of p of daughters

`DaughterPtProductFilterCriterion`: p_T of daughters

`DaughterPVIPFilterCriterion`: IP of daughters

`DaughterPVIPSumFilterCriterion`: Sum of IP of daughters

`DaughterVertexFilterCriterion`: Vertex of daughters

`DLLFilterCriterion`: PID DLL

`FlightDistanceFilterCriterion`: Flight distance

`KinFilterCriterion`: Momentum and p_T

`LifetimeSignificanceCriterion`: Lifetime significance

`MassDifferenceFilterCriterion`: Mass difference

`MassFilterCriterion`: Mass

`Momentum2FlightAngleFilterCriterion`: Aligned with PV

All other `IFilterCriterion`

`MomentumMotherDirectionFilterCriterion`: Daughter–mother

`MomentumMotherFlightAngleFilterCriterion`: Daughter–mother

`OverlapFilterCriterion`: Overlap

`PIDFilterCriterion`: PID

`PVIPFilterCriterion`: Impact parameter (on any PV)

`TrackFilterCriterion`: Track cuts

`TrackQualityFilterCriterion`: Track quality

`TrackTypeFilterCriterion`: Track Type

`TwoDaughterCloseApprFilterCriterion`: Distance of daughters

`TwoDaughterDistanceVertexFilterCriterion`: Distance of daughters to ver

`VtxFilterCriterion`: Vertex χ^2 and position

`VtxIsolationFilterCriterion`: Isolation of vertex

`TrueMCFilterCriterion`: Keeps particles from given decay

All other `IFilterCriterion`

`MomentumMotherDirectionFilterCriterion:` Daughter–mother

`MomentumMotherFlightAngleFilterCriterion:` Daughter–mother

`OverlapFilterCriterion:` Overlap

`PIDFilterCriterion:` PID

`PVIPFilterCriterion:` Impact p

`TrackFilterCrite`

`TrackQua`

`TrackTyp`

`TwoDaught` **`on:`** Distance of daughters

`TwoDaughte` **`xFilterCriterion:`** Distance of daughters to ver

`VtxFilterC` **`erion:`** Vertex χ^2 and position

`VtxIsolationFilterCriterion:` Isolation of vertex

`TrueMCFilterCriterion:` Keeps particles from given decay

> If you are missing a cut,
> don't be selfish:
> please provide a new
> `IFilterCriterion`!

4. Cuts on D^-

```
DForPreselBd2DPi.MinPt = 2000.*MeV ;
DForPreselBd2DPi.MotherFilter.Selections =
            { "D+ : VtxFilterCriterion, PVIPFilterCriterion,
                  FlightDistanceFilterCriterion/FDCut" };
DForPreselBd2DPi.MotherFilter.VtxFilterCriterion.MaxChi2 = 20. ;
DForPreselBd2DPi.MotherFilter.PVIPFilterCriterion.MinIPSignif = 2. ;
DForPreselBd2DPi.MotherFilter.FDCut.CutBestPV = true;
DForPreselBd2DPi.MotherFilter.FDCut.MinFSPV = 4.5 ;
```

This selects $\mathbf{D}^{\pm}$ with $\chi^2 < 20$, a 2σ IP on any PV and a 4.5σ flight separation from the PV to which it “points”.

As the mass `Window`, the `MinPt` cut is applied before the vertex fit. It’s quite helpful for the HLT. It does the same as

```
DForPreselBd2DPi.MotherFilter.Selections = { "D+ : KinFilterCriterion,
DForPreselBd2DPi.MotherFilter.KinFilterCriterion.MinPt = 3000.*MeV ;
```

5. Make the B^0

```
PreselBd2DPi.PhysDesktop.InputLocations = {"Phys/DForPreselBd2DPi",
                                   "Phys/CombinedForPreselBd2DPi"};
PreselBd2DPi.DecayDescriptor = "[B0 -> D- pi+]cc" ;
PreselBd2DPi.DaughterFilter.Selections = { "pi+ : KinFilterCriterion,
                                             PVIPFilterCriterion" };
PreselBd2DPi.DaughterFilter.KinFilterCriterion.MinPt = 500.*MeV ;
PreselBd2DPi.DaughterFilter.PVIPFilterCriterion.MinIPsignif = 2.5 ;
PreselBd2DPi.Window = 500.;
PreselBd2DPi.MotherFilter.Selections = { "B0 : VtxFilterCriterion,
                                               PVIPFilterCriterion,
                   Momentum2FlightAngleFilterCriterion/Mom2Flight" };
PreselBd2DPi.MotherFilter.VtxFilterCriterion.MaxChi2 = 20. ;
PreselBd2DPi.MotherFilter.PVIPFilterCriterion.MaxIPSignif = 6.;
PreselBd2DPi.MotherFilter.Mom2Flight.CosAngle = 0.999;
```

That was already the whole preselection: 38 lines of options!

`MakeResonances` reference

Builds mother particles according to given decay descriptor.

Options:

`DaughterFilterName`, **`MotherFilterName`** can be changed.

`Window`: Mass window

`UpperWindow`, **`LowerWindow`**: for asymmetric mass cuts

`MinMomentum`, **`MinPt`**: applied before vertexing

`KillOverlap` **(Default: true)**: Kill particles with overlap

`DecayDescriptor`: Actually a property of `DVAlgorithm`

`DecayDescriptors`: For the **very** busy people:

```
DiLeptonForPreselBu2LLK.DecayDescriptors = {"J/psi(1S) -> mu+ mu-" ,
                                            "J/psi(1S) -> e+ e-"};
```

Warning about mass windows

`Window`: Mass window

`UpperWindow`, `LowerWindow`: for asymmetric mass cuts

These windows are applied with respect to the nominal mass of the created particle.

```
MakeJpsi.DecayDescriptor = "J/psi(1S) -> mu+ mu-";
MakeJpsi.Window = 100 MeV ;
```

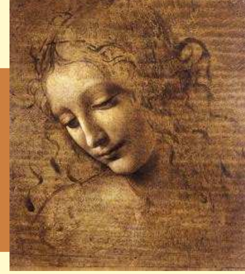
selects the range $m_{\mathbf{J}/\psi} \pm 100\,\mathrm{MeV} = [2997, 3197]\ \mathrm{MeV}$

In the $\mathbf{b} \rightarrow \ell\ell\mathbf{s}$ analyses we abuse the $\mathbf{J}/\psi$ as a dilepton container:

```
DiLepton.DecayDescriptors = {"J/psi(1S) -> mu+ mu-" , "J/psi(1S) -> e+ e-"};
DiLepton.LowerWindow = 3000.* MeV ; //MeV //
DiLepton.UpperWindow = 2500.* MeV ; //MeV //
```

This selects the range $[97, 5597]\ \mathrm{MeV}$.

Need plots?



More options for `MakeResonances`:

`HistoProduce` (false): produce plots

`DaughterPlotTool`, `MotherPlotTool`: Tool for plots. By default the `RecursivePlotTool`

`DaughterPlotsPath`, `MotherPlotsPath`: Plots path

```
PreselBd2DPi.MakePlots = true ;
PreselBd2DPi.DaughterPlots.Variables =
                              { "M", "Chi2", "Vz", "P", "Pt", "IPs"} ;
PreselBd2DPi.MotherPlots.Variables = { "WM", "M", "Chi2", "Vz", "P",
                                        "Pt", "IPs", "FS" } ;
PreselBd2DPi.DaughterPlotsPath = "BdIn" ;
PreselBd2DPi.MotherPlotsPath   = "BdOut";
```

This will produce a bunch of plots.

The plot tools

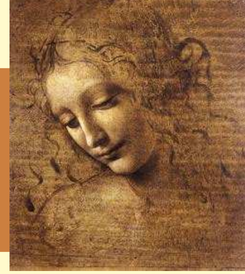
The `SimplePlotTool` and the `RecursivePlotTool` have the same options:

`Variables`: List of variables to plot. Defined are: `M` (mass), `WM` (wide mass), `DM` (mass diff.), `P`, `Pt`, `Chi2`, `IP`, `IPs` (IP signif.), `DPV` (Distance to PV), `FS` (Flight signif.), `Vz` , `Vr` , `Vx` , `Vy` (vertex positions)
The variable short-cuts will be revised someday. We'd like to be using the same names as in LoKi.

`Minima`, `Maxima`: Allows to give ranges to these variables.
Unfortunately it is mandatory to give ranges to either none or all variables.
This is also to be revised.

The `RecursivePlotTool` calls the `SimplePlotTool` for all descendants of the particle.

Need to filter without making?



Sometimes it is helpful to filter what is on the TES:

```
// Apply some pre-selection cuts on pions
HLTMakeResonances.Members += {"FilterDesktop/HLTPreselPions"};
HLTPreselPions.PhysDesktop.InputLocations = {"Phys/HLTDetachedPions"};
HLTPreselPions.Filter.Selections = {"pi+ : TrackTypeFilterCriterion,
                                           KinFilterCriterion"} ;
HLTPreselPions.Filter.KinFilterCriterion.MinPt = 300.; // MeV
HLTPreselPions.Filter.KinFilterCriterion.MinMomentum = 0.; // MeV
HLTPreselPions.Filter.TrackTypeFilterCriterion.RequireLong = true ;
HLTPreselPions.Filter.ExclusiveSelection = true ;
```

In HLT, the “detached pions” (with some IP) are refined to get the “pre-selected pions” requiring a minimum p_T of 300 MeV and long tracks.

`FilterDesktop` reference

`Filter`: the `ByPIDFilterCriterion`

`HistoProduce` (false): produce plots

`InputPlotTool`, `OutputPlotTool`: Tool for plots.

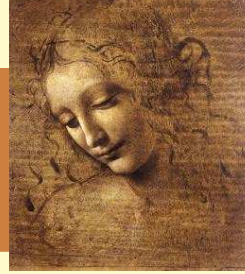
`InputPlotsPath`, `OutputPlotsPath`: Plots paths

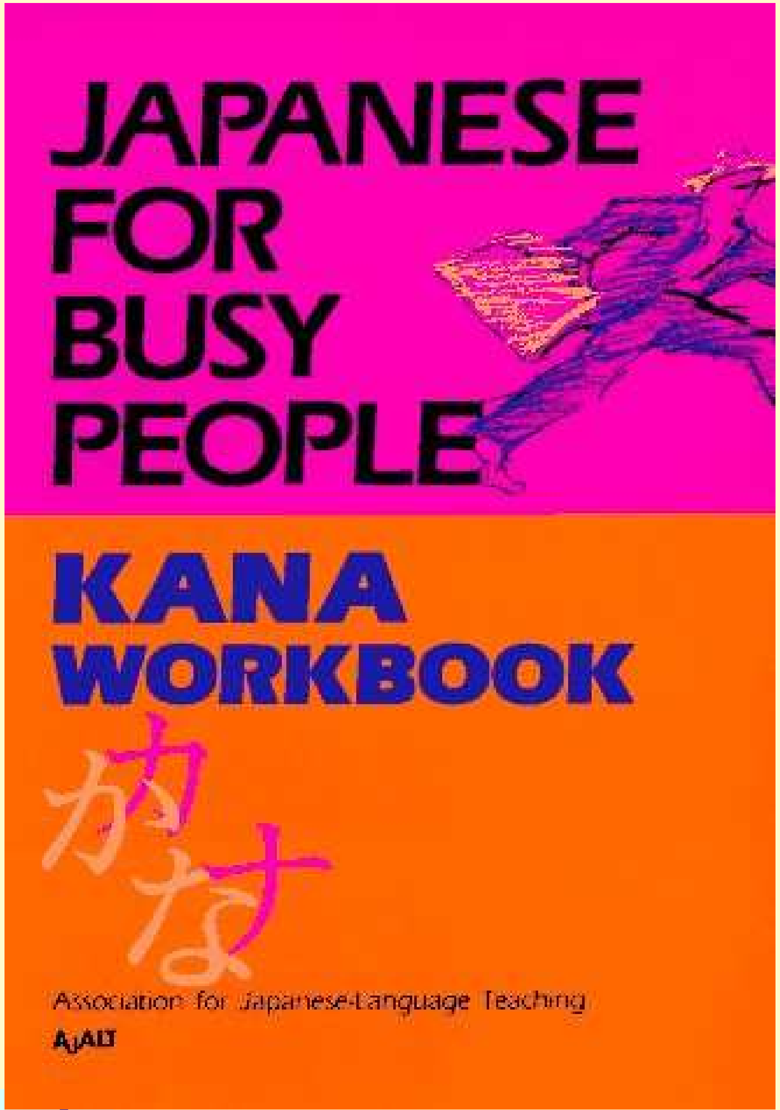
The filtered particles are cloned and put on another location (there is no way to store a pointer to a `Particle` on the TES and to treat it the same way.
This has the disadvantage that the original and the cloned particles are treated as two different particles for the MC association.

One more technicality

- `MakeResonances` is in `Phys/DaVinciTools`
- `FilterDesktop` is in `Phys/DaVinciFilter`
- The `IPlotTool` are in `Phys/DaVinciTools`
- All `IFilterCriterion` are in `Phys/DaVinciFilter` except `TrueMCFilterCriterion` in `Phys/DaVinciMCTools`

DaVinci for Busy People





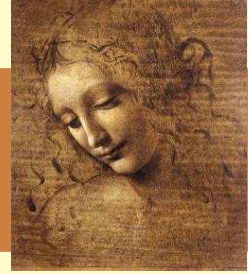
DaVinci for Busy People: (P. Koppenburg & L. Fernández) LHCb-note 2005-016 (somewhat obsolete)

Japanese for Busy People: The book for all those who feel obliged to learn some Japanese, but don't have any time to…

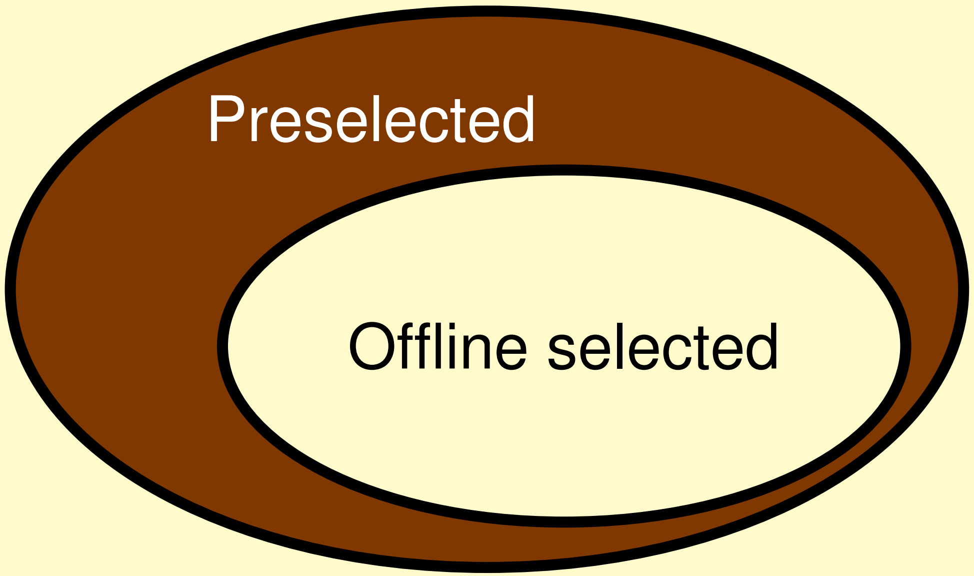
Standard Particles:

- Final states
- Composites (not yet in DC06)

Truisms

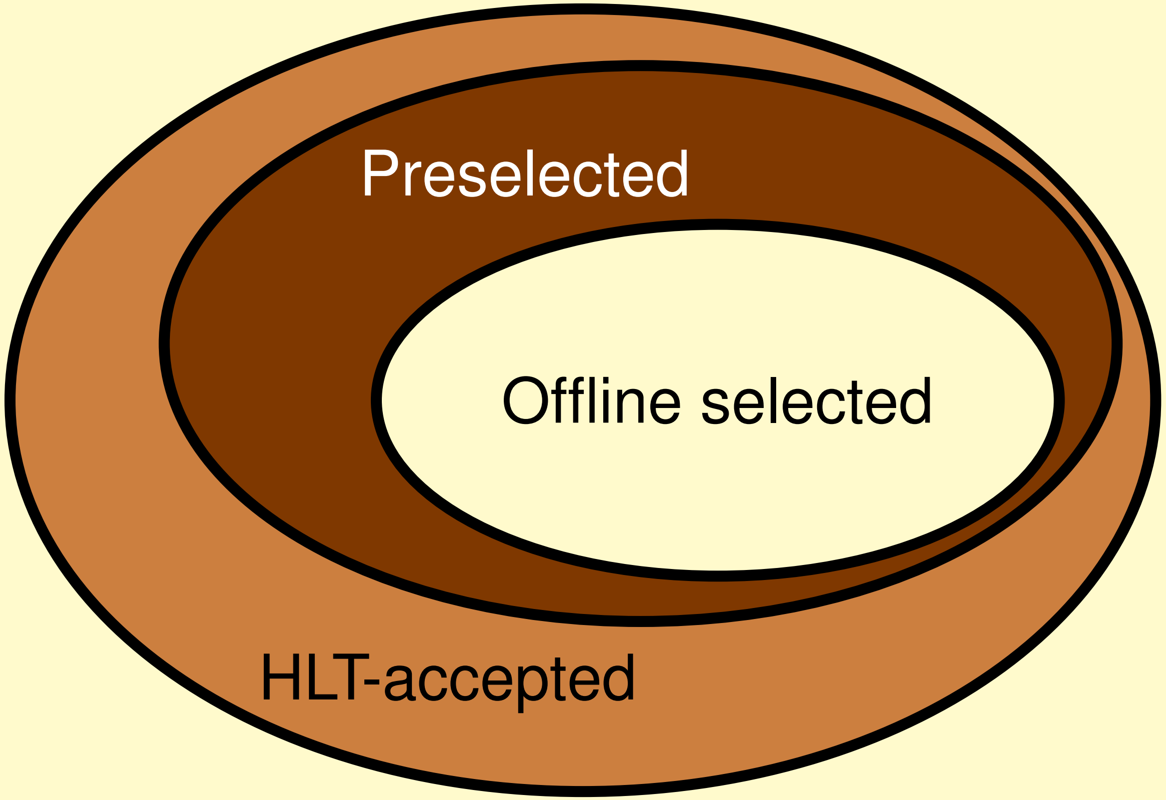


Offline candidates are a sub-sample of preselected.
$\longrightarrow$ Offline cuts are tighter.

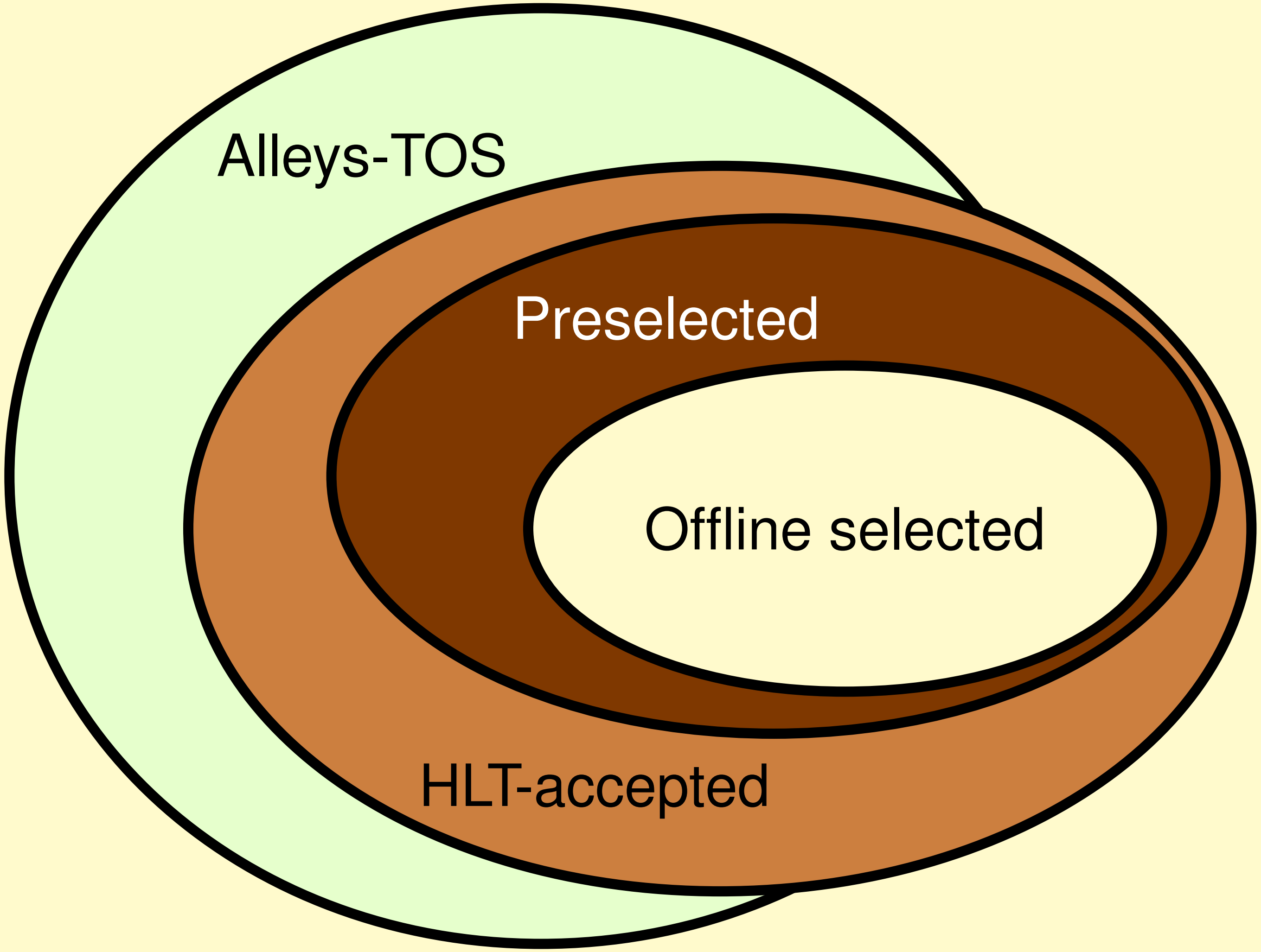


Truisms

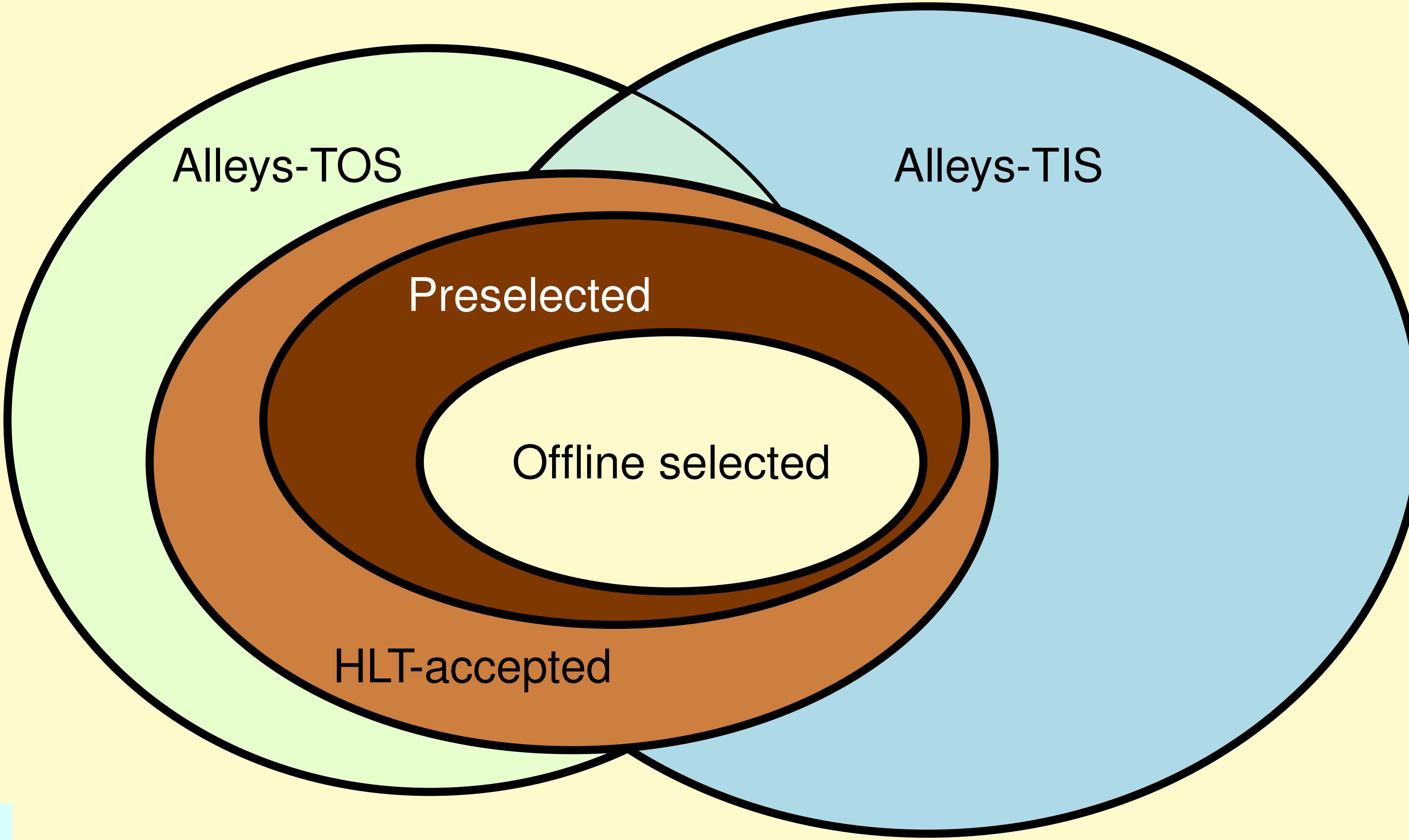
Preselected candidates are a sub-sample of HLT-selected.
$\longrightarrow$ Preselection cuts are tighter.



Truisms



Truisms

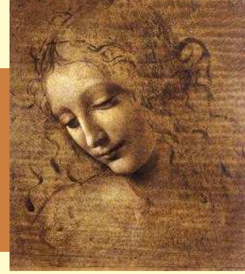

Alleys-TOS
Alleys-TIS
Preselected
Offline selected
HLT-accepted

Think Trigger!

- The HLT has timing and resolution constraints that favour simple over complicated cuts:
  - Cut tight in p_T and IP, loose in mass...
  - Make it simple.
  - → Use "standard" particles.
  - → This should improve the HLT efficiency by construction
- The design of the preselections **and of the HLT exclusive selections** is the responsibility of the WGs
  - Make sure that one can select your channel in the trigger
  - Decide if you need only TIS, TIS and TOS, or even TOB
  - In the end it's only the MB rate that counts. Forget about the factor 1/1000 in $\mathbf{B\overline{B}}$.

100% of your signal events will be in the 2 kHz

HLT & Preselections tuning



We are in an iterative process

```
while (!(LHC start)){
```

1. Tune offline selections
2. Tune preselections — We are here
3. Tune trigger
   - HLT selections not 100% efficient
   - Preselections and HLT selections not 100% correlated
4. Tune “standard” particles based on HLT to be used in all preselections
5. Start over

```
}
```

More Motivation

- In the various selections there a plenty of different definitions of what is a good kaon or $\mathbf{D^0}$ candidate.
  - Makes it difficult to compare selections
  - Is a source of systematic errors when you use one selection as a reference
  - Reduces correlations with the trigger, i.e. causes inefficiencies
  - Causes duplication of code
  - Hides mistakes made by inexperienced physicists
- Replace that by standard final state and intermediate state particles
  - Use them in preselections
  - Use them (and maybe cut harder) in offline selections
  - Base HLT standards on them (they will have to be looser)

Standard Final State Particles

Particle	StdLooseXxxx	StdTightXxxx
Muons	"det='MUON' mu-pi='-15.0'"	"det='MUON' mu-pi='0.0'"
Electrons	"det='CALO' e-pi='-5.0'"	"det='CALO' e-pi='0.0'"
Kaons	"det='RICH' k-pi='-5.0'"	"det='RICH' k-pi='0.0'"
Pions		"det='RICH' pi-k='-5.0'"
Protons	"det='RICH' p-pi='-5.0'"	"det='RICH' p-pi='0.0'"
Photons	PhotonParticleMaker	N/A
CnvPhotons	CnvPhotonParticleMaker	N/A

- They all also exist as `StdNoPIDs`,
- and there are also combinations (`hadrons` ...),
- and there are VTT and downstream π and $\mathbf{p}$.

Standard Composites

	Dimuon	$\mathbf{J}/\psi \to \mu\mu$	$\mathbf{J}/\psi \to \mu\mu$
	Jan Amoraal & PK		
ℓ p_T	max > 500 MeV		
$\mu\mu$-π DLL	max > -1.0		
$\mathbf{J}/\psi$ χ^2	< 6		
$\mathbf{J}/\psi$ window	2.5–10.1 GeV	± 400 MeV	± 400 MeV
$\mathbf{J}/\psi$ Flight			$> 2\sigma$

	$\phi \to \mathbf{KK}$	$\mathbf{K}^* \to \mathbf{K}\pi$	$\rho \to \pi\pi$
	Ivan Belyaev		
h IP	$> 2\sigma$	$> 1.5\sigma$	$> 2\sigma$
χ^2	< 25	< 25	< 25
Mass window	± 50 MeV	± 300 MeV	± 250 MeV

$\mathbf{K}^0_\mathbf{S} \to \pi\pi$	LL	DD
	Cristina Lazzeroni	
π p	> 3 GeV	
π IP	$> 4\sigma$	
π track χ^2	< 3.0	
χ^2	< 50	< 100
z range [mm]	-100–650	-100–2550
r range [mm]	$<< 0.5$	< 10
Mass window	± 110 MeV	± 300 MeV

$\mathbf{\Lambda} \to \mathbf{p}\pi$	LL	DD	LU	UL	LD	DL
	Federica Legger & Eric Conte					
$\mathbf{p}$ p_T [MeV]	> 400	> 1100	> 400	> 300	> 400	> 1100
π p_T [MeV]	> 100	> 250	> 50	> 100	> 250	> 100
$\mathbf{p}$ IP	$> 3\sigma$		$> 3\sigma$	$> 3\sigma$	$> 3\sigma$	
π IP	$> 3\sigma$	$> 2\sigma$	$> 3\sigma$	$> 3\sigma$	$> 2\sigma$	$> 3\sigma$
χ^2	< 50					
$\mathbf{\Lambda}$ p_T [MeV]	> 500	> 1000	> 500	> 500	> 1000	> 1000
Window	± 50 MeV					

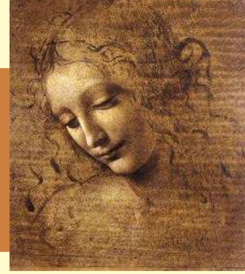
Standard D^0, D^+, D_s, D^*

	$D^0 \to K\pi$	$D^0 \to \pi\pi$	$D^0 \to KK$	$D^0 \to K^0_S\pi\pi$	$D^0 \to K^0_S KK$	$D^+ \pm KK\pi$	$D_s \to KK\pi$
	Vladimir Gligorov			Cristina Lazzeroni		Vava	Jérémie
h p_T	> 300 MeV	> 300 MeV	> 300 MeV	> 400 MeV	> 300 MeV	> 300 MeV	> 300 MeV
K^0_S p_T				> 1 GeV	> 800 MeV		
h p	> 2 GeV	> 2 GeV	> 2 GeV			> 2 GeV	> 2 GeV
h IP	$> 2\sigma$	$> 2\sigma$	$> 2\sigma$			$> 1\sigma$	$> 2\sigma$
D p_T	> 1000 MeV	> 1000 MeV	> 1000 MeV			> 500 MeV	> 1 GeV
D χ^2	< 25	< 25	< 25	< 30	< 40	< 25	< 20
D IP				$> 1\sigma$	$> 0.7\sigma$	$> 2\sigma$	$> 3\sigma$
D window	± 50 MeV	± 50 MeV	± 50 MeV	± 120 MeV	± 120 MeV	± 50 MeV	± 200 MeV

	$D^* \to \pi D^0(K\pi)$	$D^* \to \pi D^0(\pi\pi)$	$D^* \to \pi D^0(KK)$
	Vladimir Gligorov		
π IP	$> 1\sigma$	$> 1\sigma$	$> 1\sigma$
D^* p_T	> 1250 MeV	> 1250 MeV	> 1250 MeV
D^* χ^2	< 25	< 25	< 25
D^* window	± 50 MeV	± 50 MeV	± 50 MeV
D^*-D mass	± 20 MeV	± 20 MeV	± 20 MeV

- All preselections use these particles
- Cuts very similar between modes
- Warning: The default $D^0 \to K\pi$ does not contain DCS mode

How to



- The list of available states can be found here
  - which maps `$COMMONPARTICLESROOT/options/StandardOptions.opts`
  - They are not in **DaVinci** v16r1, but will be in the next I assume.
- To use them, simply include the proper path in your `PhysDesktop.InputLocations`
  - If they are used for the first time in the event this will call the `DataOnDEmandSvc` and trigger the execution of the relevant algorithm.

Exercises!

- Let's go for the exercises

Ex. 4 Write the $\mathrm{B_s}$ maker using `MakeResonances`

- Be very careful about typing! There is no compilation!

Optionally You can also re-write the ϕ and J/ψ makers