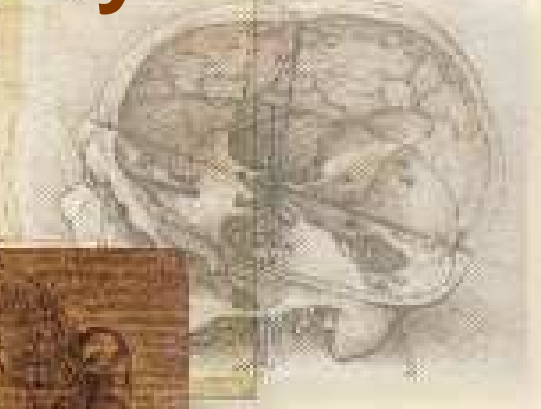


Why there are no stripping results yet



- **Reminder**
- **What works**
- **What doesn't**

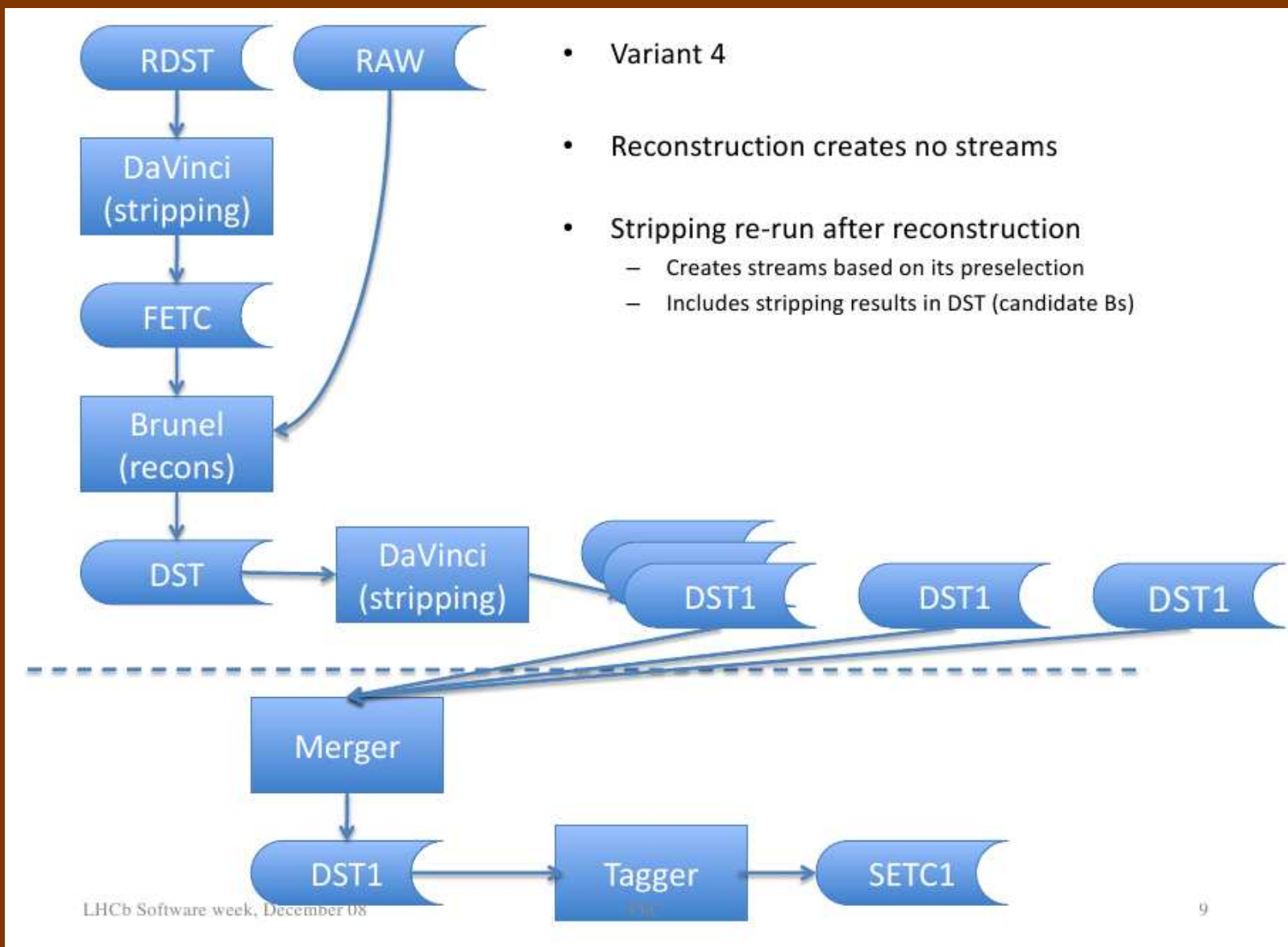
17/06/08 — Software week

Patrick Koppenburg
Imperial College
London





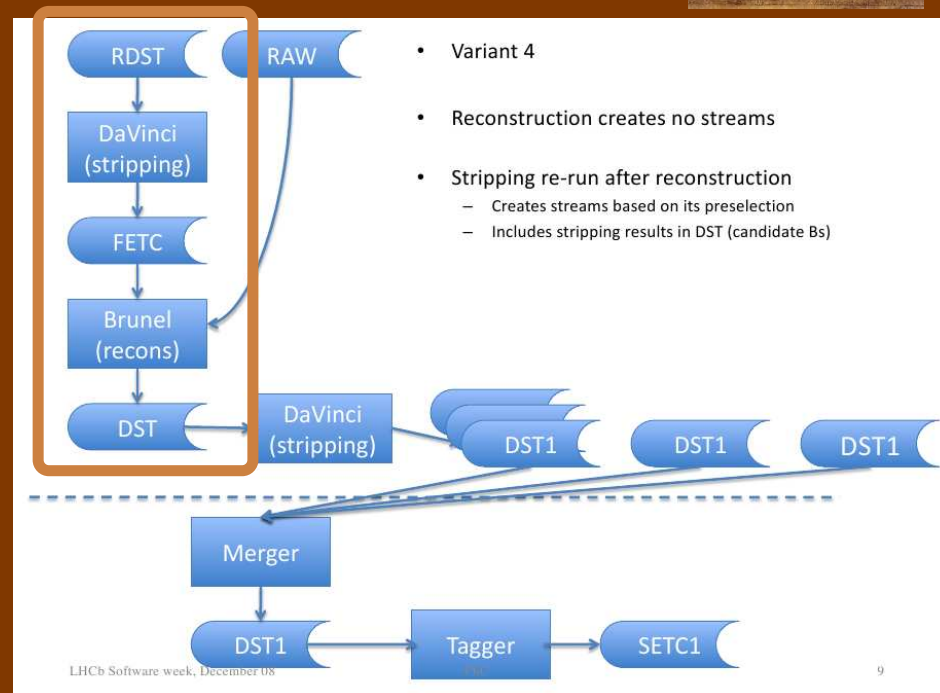
What do we want?





What do we want?

- So far only look at the first job
 - $rDST \rightarrow DaVinci \rightarrow ETC \rightarrow Brunel \rightarrow DST$
- The first step is the standard stripping
 - ✓ That works
- The second step is **Brunel** reading from ETC
 - ✗ Two problems found
- The good news is that I can run the stripping
 - ✗ But it needs a new release of **Brunel** and **DaVinci**



Problems



Brunel configurable:

- Stripping was never tested with recent data and latest **Brunel**
- A lot has changed on the **DaVinci** side, which has effects on **Brunel**
- ➔ Needs 3 patches in `Configuration.py`

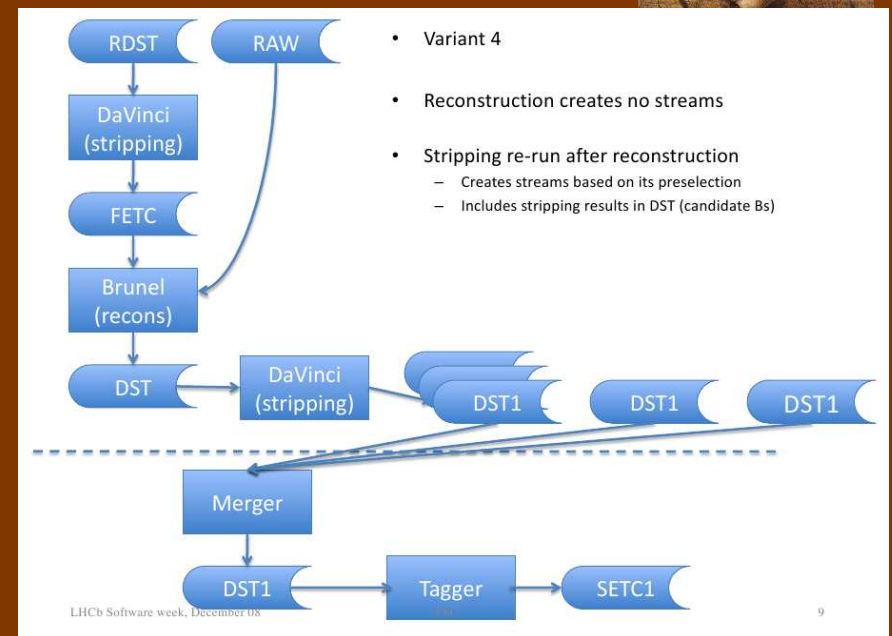
DIRAC:

- To select data I need :

```
"COLLECTION='TagCreator/EventTuple' DATAFILE='"+fname+"'  
TYP='POOL_ROOT' SEL='(StrippingB2Charged2Body>0) ||  
(StrippingB2DPi>0) || (StrippingBd2JpsiKSLine>0) ||  
(StrippingBd2JpsiKstLine>0) || (StrippingBd2KstarMuMu_10Hz>0)  
|| (StrippingBs2JpsiPhiLine>0) || (StrippingBu2JpsiKLine>0) ||  
(StrippingTopo>0)'"
```
- DIRAC needs to set `fname`, but we don't want to hard-code all selections
- ➔ Need one selection that is an or of all others
- ✓ Anton will provide a patch to stripping configuration

Conclusion

- ✓ Stripping works locally
- ✗ But is not yet ready for production
- ➔ Hope to start it next week





AND DON'T GIVE ME
THE CONDESCENDING
SIMPLE VERSION FOR
MANAGERS. I WANT A
FULL TECHNICAL
EXPLANATION.



Backup Slides

DST writing



DST writer: The new configurable `DaVinciWriteDST` allows to write DSTs.

- ✓ Dictionary of `file : sequencer` allows many DST files
- ✓ The same event can be written out several times

NEW
NEW

Candidate saving: We need to save the candidate to avoid re-doing the combinatorics on the whole event

- ➔ Reduces the risk of mistakes (*Stripping-Independent-of-Signal* events)
- ✓ `DaVinciWriteDST` writes the whole input DST, plus any other locations
- ✗ It writes all particles in that location
 - We want the candidate, plus the *needed* intermediates.

How? `MicroDST` can already do that. No need to re-invent the wheel.

- ➔ Call `MicroDST` configuration from `DaVinciWriteDST`

ETC writing



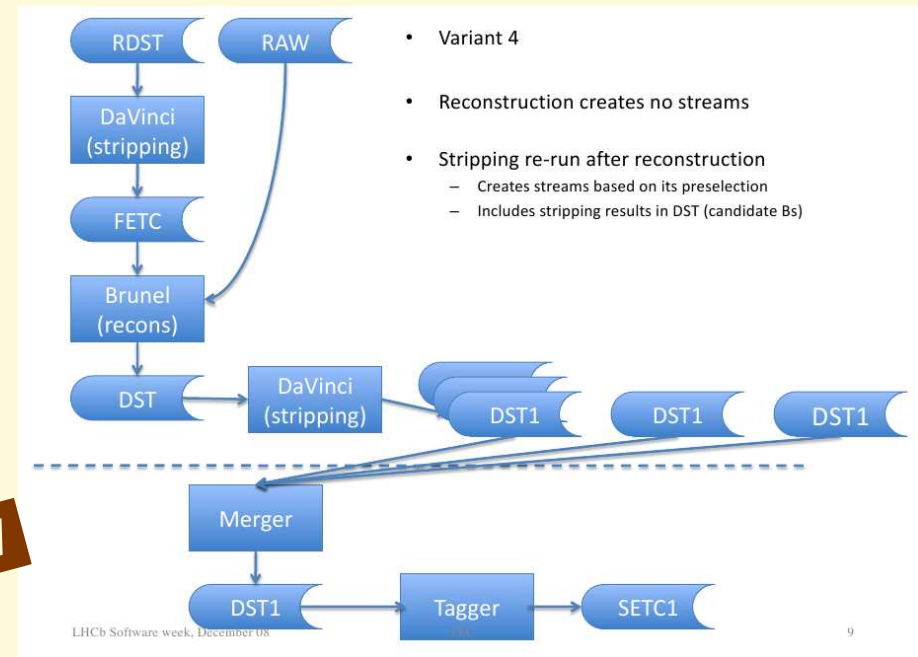
Stripping: Two steps save an event-tag-collection (ETC)

- Not clear if users will use it
- Mandatory if stripped events are reprocessed

ETC filling:

- EventTuple now allows to save the Tuple as an ETC **NEW**
 - TupleToolSelResults stores the `filterPassed` result of any algorithm.
- That's all we need.

ETC saving: `DaVinci()` now supports ETC writing **NEW**



Updated CommonParticles



- All common particles `StdLooseKaons`, `StdLooseKstar` ... used in the DC06 stripping are frozen and have been renamed to `StdDC06LooseKaons`, `StdDC06LooseKstar` ...
 - Don't use them for new selections
- New `StdLooseKaons`, `StdLooseKstar` have been created based on HLT shared particles
 - They are now in `Phys/CommonParticles/python/CommonParticles`
- They are still on demand, and predeclared in `AnalysisConf()`
- **These are templates. Discuss the definitions in the WGs!**

Organisation of Selections



- PhysSel / * and Tools / Stripping are obsolete
- We need new selections (using CombineParticles)
 - Based on new CommonParticles
 - Focusing on priority channels
- People in the WGs are working on that
- They should be organised and controlled by a configurable
 - 99%-identical to future Hlt2Conf
 - Organisation in sequencers and prescaled should follow Hlt2 structure
- We want Hlt2 and Stripping to be as much alike as possible
 - See my HLT talk tomorrow afternoon
- Templates to appear soon
 - Co-ordinated by Anton