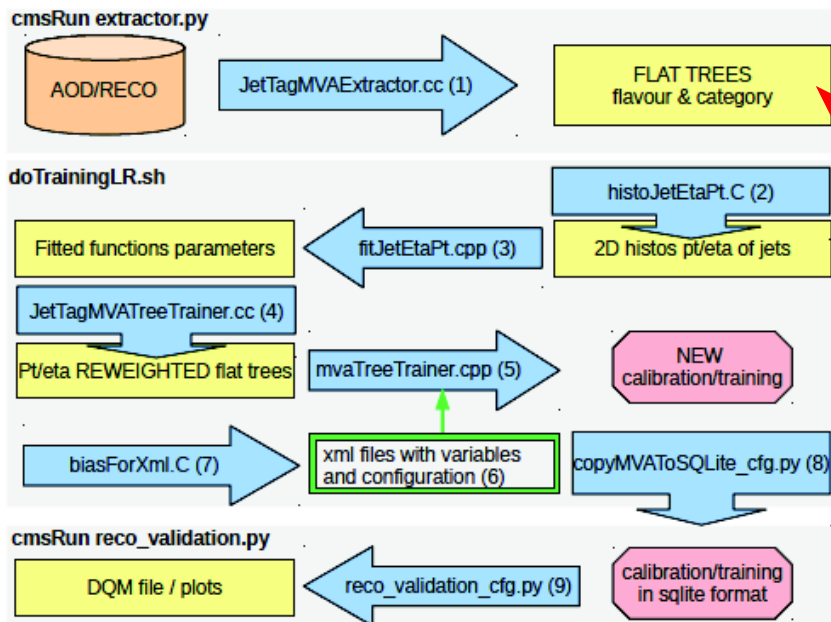


Training of CSV algo @ HLT level

Work is based on the [documentation CMS/AN-12-441](#) from Petra Van Mulders & al:
Implementation and training of the Combined Secondary
Vertex MVA b-tagging algorithm in CMSSW

Package used: RecoBTau/JetTagMVALearning

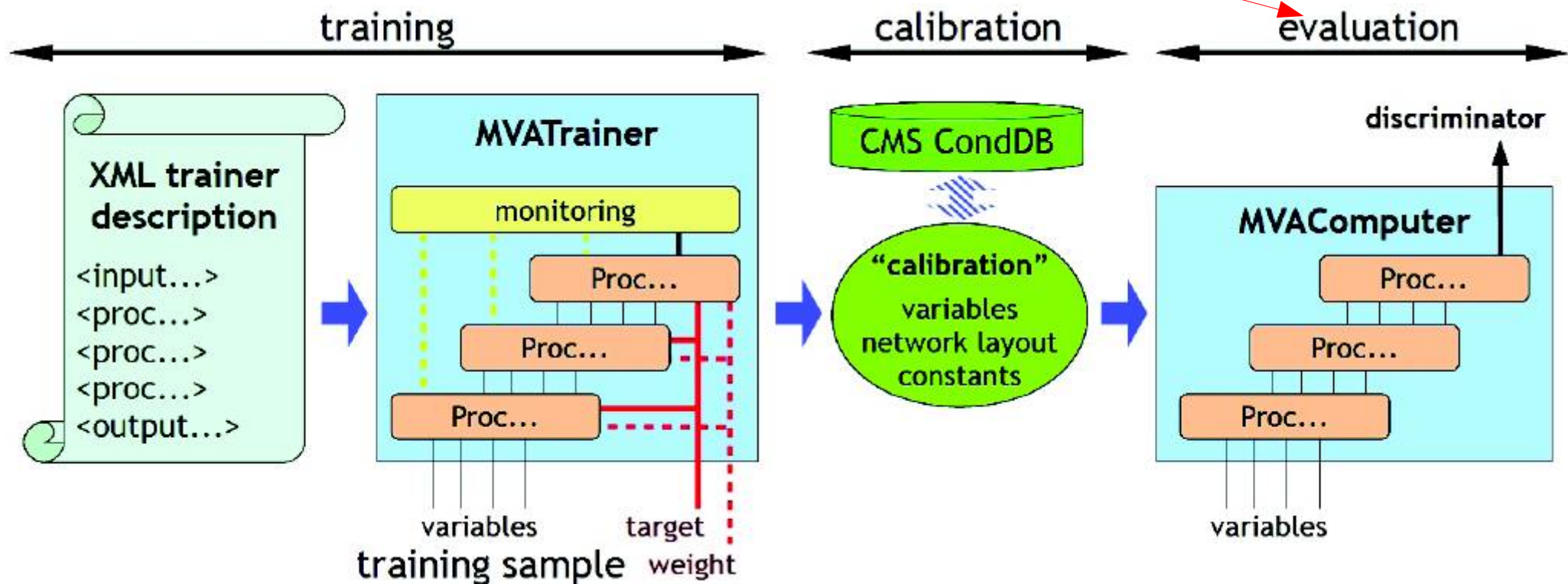
Scheme of the CVS training



For the training @ HLT, the main different is to be able to access HLT objects and fill the flat tree.

=> I've already a working recipe

Evaluation of performance is also different for HLT btagging



Ongoing work

1st step: identifying what are the inputs of JetTagMVAExtractor

Path investigated: HLT_DiCentralPFJet30_PFMET80_BTagCSV07_v5

Input collections for the training related to that path:

- **Tracks:** hltFastPixelBLifetimeRegionalCtfWithMaterialTracksHbb
- **SVTagInfo:** hltL3SecondaryVertexTagInfos
- **IPTagInfo:** hltFastPixelBLifetimeL3TagInfosHbb
- **Jets:** hltSelectorJets20L1FastJet

2nd step: establishing a recipe for the production of the flat-tree

- **Run on GEN-SIM-RAW**
- **Run the HLT paths**
- **Keep all the mandatory collections**
- **Run the JetTagMVAExtractor on that collection**
- **Store all the root file (one for each jet flavor)**

Samples: QCD & ttbar [prod. ongoing]

Other steps: identical to offline training

Extension of the work

As far as I understood, as long as the input collections (tracks, IP, SV, jet) inherit from the basic classes of CMSSW, one can try the run CVS algo at any level (L2.5, ...) of the trigger.