PF2PAT

25/10/2011

- Particle Flow :
  - Reconstruction des PFCandidates
    - Déjà présents dans les AOD
    - Divisés en types
      - h
      - e
      - mu
      - gamma
      - h0
      - h_HF
      - egamma_HF
      - X
  - PF2PAT (aujourd'hui)
  - Pat Data Format

Quelques liens

- https://twiki.cern.ch/twiki/bin/view/CMSPublic/PF2PAT
- https://twiki.cern.ch/twiki/bin/view/CMSPublic/SWGuidePF2PAT
- https://twiki.cern.ch/twiki/bin/view/CMSPublic/SWGuidePATRecipes (configuration)
- https://twiki.cern.ch/twiki/bin/view/CMSPublic/WorkBookPF2PAT
- https://twiki.cern.ch/twiki/bin/view/CMSPublic/WorkBookPATConfiguration
- https://twiki.cern.ch/twiki/bin/view/CMSPublic/WorkBookPATDataFormats
- https://twiki.cern.ch/twiki/bin/view/CMSPublic/SWGuidePATFAQs (ne semble pas mis à jour pour 4_X_Y)
- https://twiki.cern.ch/twiki/bin/view/CMSPublic/SWGuidePATTools

PatSequence
(based on data – version4_2_X)

process.patPF2PATSequence = cms.Sequence(process.PF2PAT
+process.patDefaultSequence)

```
process.PF2PAT =
cms.Sequence(process.pfNoPileUpSequence
+process.pfAllNeutralHadrons
+process.pfAllChargedHadrons
+process.pfAllPhotons
+process.pfMuonSequence+process.pfNoMuon
+process.pfElectronSequence
+process.pfNoElectron
+process.pfJetSequence+process.pfNoJet
+process.pfTauSequence+process.pfNoTau
+process.pfMET)
```

```
process.pfNoPileUpSequence = cms.Sequence(process.pfPileUp
+process.pfNoPileUp)
```

```
process.pfPileUp = cms.EDProducer("PFPileUp",
       PFCandidates = cms.InputTag("particleFlow"),
       Enable = cms.bool(True),
       checkClosestZVertex = cms.bool(True),
       verbose = cms.untracked.bool(False),
       Vertices = cms.InputTag("offlinePrimaryVertices")
   )
process.pfNoPileUp =
cms.EDProducer("TPPileUpPFCandidatesOnPFCandidates",
       bottomCollection = cms.InputTag("particleFlow"),
       enable = cms.bool(True),
       topCollection = cms.InputTag("pfPileUp"),
       name =
cms.untracked.string('pileUpOnPFCandidates'),
       verbose = cms.untracked.bool(False)
   )
```

```
process.PF2PAT = cms.Sequence(process.pfNoPileUpSequence
+process.pfAllNeutralHadrons+process.pfAllChargedHadrons
+process.pfAllPhotons+process.pfMuonSequence+process.pfNoMuon
+process.pfElectronSequence+process.pfNoElectron
+process.pfJetSequence+process.pfNoJet+process.pfTauSequence
+process.pfNoTau+process.pfMET)
```

```
process.pfAllNeutralHadrons = cms.EDFilter("PdgIdPFCandidateSelector",
   pdgId = cms.vint32(111, 130, 310, 2112),
   src = cms.InputTag("pfNoPileUp")
)

process.pfAllChargedHadrons = cms.EDFilter("PdgIdPFCandidateSelector",
   pdgId = cms.vint32(211, -211, 321, -321, 999211,
      2212, -2212),
   src = cms.InputTag("pfNoPileUp")
)

process.pfAllPhotons = cms.EDFilter("PdgIdPFCandidateSelector",
   pdgId = cms.vint32(22),
   src = cms.InputTag("pfNoPileUp")
)
```

```
process.pfAllNeutralHadrons = cms.EDFilter("PdgIdPFCandidateSelector",
    pdgId = cms.vint32(111, 130, 310, 2112),
    src = cms.InputTag("pfNoPileUp")
)

process.pfAllChargedHadrons = cms.EDFilter("PdgIdPFCandidateSelector",
    pdgId = cms.vint32(211, -211, 321, -321, 999211,
        2212, -2212),
    src = cms.InputTag("pfNoPileUp")
)

process.pfAllPhotons = cms.EDFilter("PdgIdPFCandidateSelector",
    pdgId = cms.vint32(22),
    src = cms.InputTag("pfNoPileUp")
)
```

```
process.PF2PAT =
cms.Sequence(process.pfNoPileUpSequence
+process.pfAllNeutralHadrons
+process.pfAllChargedHadrons
+process.pfAllPhotons+process.pfMuonSequence
+process.pfNoMuon+process.pfElectronSequence
+process.pfNoElectron+process.pfJetSequence
+process.pfNoJet+process.pfTauSequence
+process.pfNoTau+process.pfMET)
```

```
process.pfMuonSequence = cms.Sequence(process.pfAllMuons+process.pfMuonsFromVertex
+process.pfSelectedMuons+process.pfMuonIsolationSequence+process.pfIsolatedMuons)
process.pfAllMuons = cms.EDFilter("PdgIdPFCandidateSelector",
    pdgId = cms.vint32(-13, 13),
    src = cms.InputTag("pfNoPileUp")
)
process.pfMuonsFromVertex = cms.EDFilter("IPCutPFCandidateSelector",
    d0Cut = cms.double(0.2),
    src = cms.InputTag("pfAllMuons"),
    dzSigCut = cms.double(99.0),
    d0SigCut = cms.double(99.0),
    vertices = cms.InputTag("offlinePrimaryVertices"),
    dzCut = cms.double(0.5)
)
process.pfSelectedMuons = cms.EDFilter("GenericPFCandidateSelector",
    src = cms.InputTag("pfMuonsFromVertex"),
    cut = cms.string('pt>5')
)
process.pfMuonIsolationSequence = cms.Sequence(process.pfMuonIsoDepositsSequence
+process.pfMuonIsolationFromDepositsSequence) ---> Je ne rentre pas dans les détails pour l'instant.
process.pfIsolatedMuons = cms.EDFilter("IsolatedPFCandidateSelector",
    src = cms.InputTag("pfSelectedMuons"),
    isRelative = cms.bool(True),
    combinedIsolationCut = cms.double(0.15),
    isCombined = cms.bool(True),
    isolationValueMaps = cms.VInputTag(cms.InputTag("isoValMuonWithCharged"),
cms.InputTag("isoValMuonWithNeutral"), cms.InputTag("isoValMuonWithPhotons")),
    isolationCuts = cms.vdouble(10, 10, 10)
)
```

```
process.PF2PAT =
cms.Sequence(process.pfNoPileUpSequence
+process.pfAllNeutralHadrons
+process.pfAllChargedHadrons
+process.pfAllPhotons+process.pfMuonSequence
+process.pfNoMuon+process.pfElectronSequence
+process.pfNoElectron+process.pfJetSequence
+process.pfNoJet+process.pfTauSequence
+process.pfNoTau+process.pfMET)
```

```
process.pfNoMuon = cms.EDProducer("TPPFCandidatesOnPFCandidates",
    bottomCollection = cms.InputTag("pfNoPileUp"),
    enable = cms.bool(True),
    topCollection = cms.InputTag("pfIsolatedMuons"),
    name = cms.untracked.string('noMuon'),
    verbose = cms.untracked.bool(False)
)
```

```
process.PF2PAT =
cms.Sequence(process.pfNoPileUpSequence
+process.pfAllNeutralHadrons
+process.pfAllChargedHadrons
+process.pfAllPhotons+process.pfMuonSequence
+process.pfNoMuon+process.pfElectronSequence
+process.pfNoElectron+process.pfJetSequence
+process.pfNoJet+process.pfTauSequence
+process.pfNoTau+process.pfMET)
```

PatSequence

```
process.patPF2PATSequence =
cms.Sequence(process.PF2PAT
+process.patDefaultSequence)
```

```
process.patDefaultSequence =
cms.Sequence(process.patElectrons+process.patMuons
+process.patJetCorrections+process.patJetCharge
+process.patJets+process.makePatMETs
+process.patCandidateSummary
+process.selectedPatCandidates
+process.countPatCandidates)
```

```
process.patMuons = cms.EDProducer("PATMuonProducer",
    embedTpfmsMuon = cms.bool(True),
    embedHighLevelSelection = cms.bool(True),
    embedCaloMETMuonCorrs = cms.bool(True),
    caloMETMuonCorrs = cms.InputTag("muonMETValueMapProducer","muCorrData"),
    resolutions = cms.Pset(
    ),
    userIsolation = cms.Pset(
    ),
    embedPFCandidate = cms.bool(True),
    pfMuonSource = cms.InputTag("pfIsolatedMuons"),
    efficiencies = cms.Pset(
    ),
    embedStandAloneMuon = cms.bool(True),
    useParticleFlow = cms.bool(False),
    userData = cms.PSet(
       userCands = cms.PSet(
          src = cms.VInputTag("")
       ),
       userInts = cms.PSet(
          src = cms.VInputTag("")
       ),
       userFloats = cms.PSet(
          src = cms.VInputTag("")
       ),
       userClasses = cms.PSet(
          src = cms.VInputTag("")
       ),
       userFunctionLabels = cms.vstring(),
       userFunctions = cms.vstring()
    ),
    embedTrack = cms.bool(False),
    addEfficiencies = cms.bool(False),
    usePV = cms.bool(True),
    embedTcMETMuonCorrs = cms.bool(True),
    pvSrc = cms.InputTag("offlinePrimaryVertices"),
    addTeVRefits = cms.bool(True),
    embedCombinedMuon = cms.bool(True),
    genParticleMatch = cms.InputTag(""),
    beamLineSrc = cms.InputTag("offlineBeamSpot"),
    muonSource = cms.InputTag("muons"),
    addGenMatch = cms.bool(False),
    addResolutions = cms.bool(False),
    tpfmsSrc = cms.InputTag("tevMuons","firstHit"),
    pickySrc = cms.InputTag("tevMuons","picky"),
    isoDeposits = cms.Pset(
    ),
    embedGenMatch = cms.bool(False),
    tcMETMuonCorrs = cms.InputTag("muonTCMETValueMapProducer","muCorrData"),
    embedPickyMuon = cms.bool(True)
```

```
process.patDefaultSequence =
cms.Sequence(process.patElectrons+process.patMuons
+process.patJetCorrections+process.patJetCharge
+process.patJets+process.makePatMETs
+process.patCandidateSummary
+process.selectedPatCandidates
+process.countPatCandidates)
```

```
process.patCandidateSummary =
cms.EDAnalyzer("CandidateSummaryTable",
   logName = cms.untracked.string('patCandidates|
PATSummaryTables'),
   candidates = cms.VInputTag(cms.InputTag("patElectrons"),
cms.InputTag("patMuons"), cms.InputTag("patTaus"),
cms.InputTag("patPhotons"), cms.InputTag("patJets"),
      cms.InputTag("patMETs"))
)
```

```
process.patDefaultSequence =
cms.Sequence(process.patElectrons+process.patMuons
+process.patJetCorrections+process.patJetCharge
+process.patJets+process.makePatMETs
+process.patCandidateSummary
+process.selectedPatCandidates
+process.countPatCandidates)
```

```
process.selectedPatCandidates =
cms.Sequence(process.selectedPatElectrons
+process.selectedPatMuons+process.selectedPatJets
+process.selectedPatCandidateSummary
```

```
process.selectedPatMuons =
cms.EDFilter("PATMuonSelector",
    src = cms.InputTag("patMuons"),
    cut = cms.string('')
)
```

```
process.patDefaultSequence = cms.Sequence(process.patElectrons
+process.patMuons+process.patJetCorrections+process.patJetCharge
+process.patJets+process.makePatMETs
+process.patCandidateSummary+process.selectedPatCandidates
+process.countPatCandidates)
```

```
process.countPatCandidates =
cms.Sequence(process.countPatElectrons+process.countPatMuons
+process.countPatLeptons+process.countPatJets)
```

```
process.countPatMuons = cms.EDFilter("PATCandViewCountFilter",
    maxNumber = cms.uint32(999999),
    src = cms.InputTag("selectedPatMuons"),
    minNumber = cms.uint32(0)
)
```

PatDataFormat

- https://twiki.cern.ch/twiki/bin/view/CMSPublic/WorkBookPATDataFormats#PatMuon

- Dans le cadre de PF2PAT, il reste à discuter de :
  - Soustraction des neutres issus du PU
  - pfElectrons
  - Isolation des leptons
  - pfJets + corrections
  - pfTau
  - pfMET
- Il faudrait aussi étudier la reconstruction Pflow et les objets PFCandidate