Bender v7r0 as your analysis environment

Vanya BELYAEV
References

- **Bender Pages** and **Bender pages** by *Lena Mayatskaya*
- **Bender mailing list**
- **Bender Savannah portal** (news, bugs, tasks, ...)
- **Bender Tutorial**: slides & instructions
- **Bender Examples**
  - including nice scripts from *Diego* for $B_s \rightarrow \mu \mu$ background studies
    - getpack Ex/BenderExample v7r0
- “**Bender-helpdesk@lhcb.cern.ch**”
  - **1-R-010** at CERN
  - +41 (0) 22 767 89 28
When use Bender

- Python: perfect for prototyping
  - e.g. develop the cuts for preselection/stripping
- **Interactive**: perfect for “short” (“supervising”) tasks
  - resolutions
  - spectra
  - “reflections”
- **Flexible & Friendly**:
  - good for “the final” analysis of small data sets
  - combine with Root, Panoramix, RooFit, …
When no Bender

• Stripping does not support Bender.
• Reasons?
  • 😞 *Some CPU penalty* for Bender selections vs LoKi selections is unavoidable (Python vs C++)
  • could be visible/sizeable for “minimal” job
    • mainly comes from the explicit loops, ntuples and explicit manipulations with dictionaries:
      \[ \text{sqrt}(p.px() * p.px() + p.py() * p.py()) \]
    • could be very small for realistic selection
    • And of course for well-coded lines
  Negligible with patterns (C++) 😊
Bender v7r0

• The most fresh version of Bender, based on DaVinci v19r1 - official DC06 stripping version

• The tutorial slides are attached to the agenda
  • Here only some highlights:
    • It is already slide #5, and I have only 30 minutes

• *If somebody needs, I would be happy to organize “hands-on” Bender tutorial similar to tutorials in Beijin & Dortmund or semiprivate tutorial for HLT guys.*
from bendermodule import *

gaudi.config( files = ['MyOptionsFile.opt'])

gaudi.run(10)

gaudi.exit()

Well, It is not Bender, it is GaudiPython

Take care about input data!!

../solution/Minimalistic_0.py
Minimal **Bender** module

```python
from bendermodule import *

def configure() :
    gaudi.config( files =
                 [‘MyOptionsFile.opts’])
    return SUCCESS

if __name__ == ‘__main__’ :
    configure()
    gaudi.run(100)

../solutions/Minimalistic.py
```

**Application and Components Configuration**

**Job steering**
Scripts vs modules

- Dilemma in Python: scripts vs modules
- Scripts are a bit more intuitive and a bit easier to write
  - Problems with reusing 😊
- Modules require some discipline & conventions 😞
  - full power of OO, including classes & extensions
  - Easy to import and reuse 😊
  - the only way to assemble “large” application from pieces
- Be friendly: code modules
  - loose nothing
  - gain a lot
Scripts versus modules

• Script above:
  
  import myscript

  Will execute everything out of control

• Module above:
  
  import mymodule
  mymodule.config()
  gaudi.run(100)
"Hello, World!" (I)

- The simplest possible BENDER "algorithm"
- Follow LoKi's style:
  - *inherit the algorithm from useful base class*
  - (re)implement the "analyse" method

```python
class HelloWorld(Algo) :
    def analyse( self ) :
        print 'Hello, World!'
    return SUCCESS

../solutions/HelloWorld.py
```
"Hello, World!" (II)

- One needs to instantiate the algorithm
  \[
  \text{alg} = \text{HelloWorld}(\text{'Hello'})
  \]

- Add it to the list of 'active' algorithms
  \[
  \text{gaudi.addAlgorithm( alg )}
  \]

- Execute 😊
  \[
  \text{gaudi.run(10)}
  \]

../solutions/HelloWorld.py

Part of job steering block
Access to the data (LoKi’s style)

- **C++: GaudiAlgorithm/LoKi**

```cpp
const MCParticles* mcps = get<MCParticles>("MC/Particles");
```

- **Python: Bender**

```python
mcps = self.get('MC/Particles')
```

Semantics to be improved

../solutions/DataAccess.py
Access to the data using service

• Inside the algorithm

```python
dataSvc = self.evtSvc()
hdr     = dataSvc['Header']
print 'Event #', hdr.evtNum()
```

• Outside the algorithms

```python
dataSvc = gaudi.evtSvc()
hdr     = dataSvc['Header']
print 'Run #', hdr.runNum()
```
Attributes and (python) loops

for mcp in mcps:
    print 'ID=', nameFromPID( mcp.particleID() )
    print 'PX=', mcp.momentum().px()
    print 'PY=', mcp.momentum().py()

• To know the available attributes:
  
  help( obj )
  help( type( obj ) )
  dir(gbl)

• ON-LINE help for ALL Python/Bender functions/classes, sometimes it is VERY useful
  
  ../solutions/DataAccess.py
Lets start with physics analysis

- >95% of LoKi’s idioms are in Bender
- The semantic is VERY similar
  - In spite of different languages
  - few ‘obvious’ exceptions
- In the game:
  - All Functions/Cuts
    - a bit more round braces are required
  - All (v,mc,mcv) select methods
  - loops , plots
  - for N-Tuples the functionality is a bit limited
    - A lack of template methods,
    - ‘farray’ need to be validated

Start from MC-truth (requires no special configurations)
**MCselect statement**

- **Selection of MCParticles which satisfy the certain criteria:**

  \[
  \text{mcmu} = \text{self.mcselect}( \text{`mcmu' ,} , \\
  \text{`mu+' == MCABSID} )
  \]

  \[
  \text{beauty} = \text{self.mcselect(}`beauty' , BEAUTY )
  \]

- **Refine criteria:**

  \[
  \text{muFromB} = \text{self.mcselect ( `muFromC' ,} \\
  \text{mcmu ,} \\
  \text{FROMMCTREE( beauty ) )}
  \]

  \[
  \text{muPT} = \text{self.mcselect( `withPT' ,} \\
  \text{muFromB ,} \\
  \text{( MCPT > 1000 ) )}
  \]

  Select $\mu^+ \text{ & } \mu^-$

  Everything which has $b$ or $\bar{b}$

  Everything from "decay" trees (incl. decay-on-flight)

-LUG, Tab. 13.4, p.84

-..solutions/MCmuons.py
Change input data

• Get and configure EventSelector

```python
evtSel = gaudi.evtSel()
evtSel.open( "file"
```

OR

```python
evtSel.open( [ "file1", "file2"]
```

• e.g.

```python
evtSel.open ( 'LFN:/lhcb/production/DC04/v1/DST/00000543_00000017_5.dst')
```
Find MC-tree (IMCDecayFinder)

Brilliant tool from O.Dormond

- find the MC-decay trees:

\[
\text{mc} = \text{self.mcFinder()}
\]
\[
\text{trees} = \text{mc.find(}
' [B_s0 \rightarrow (J/\psi(1S) \rightarrow \mu^+ \mu^-) \phi(1020)]cc' 
\text{)}
\]

- find MC-decay tree components:

\[
\text{phis} = \text{mc.find(}
' \phi(1020) : [B_s0 \rightarrow (J/\psi(1S) \rightarrow \mu^+ \mu^-) \phi(1020)]cc' 
\text{)}
\]

- extract ‘marked’ MC-decay tree components:

\[
\text{mus} = \text{mc.find(}
' [B_s0 \rightarrow (J/\psi(1S) \rightarrow \mu^+ \mu^-) \phi(1020)]cc' 
\text{)}
\]

../solutions/MCTrees.py
Add simple histos!

for mu in mus:
    self.plot(MCPT(mu)/1000, 'PT of muon from J/psi', 0, 10)

The default values: bins = 100, weight = 1

• Configuration for HBOOK histograms:
  gaudi.HistogramPersistency = 'HBOOK'
  hsvc = gaudi.service('HistogramPersistencySvc')
  hsvc.OutputFile = 'myhistos.hbook'

../solutions/MCTrees.py
Add the simple N-Tuple

tup = self.nTuple(‘My N-Tuple’) 
zOrig = MCVXFUN( MCVZ )
for mu in mus :
    tup.column(‘PT’, MCPT ( mu ) )
    tup.column(‘P’, MCP ( mu ) )
    tup.column(‘Z’, zOrig ( mu ) )
tup.write()

**Configuration:**

myAlg = g.algorithm(‘McTree’)  
myAlg.NTupleLUN = ‘MC’  
ntsvc = g.service(‘NTupleSvc’)  
ntsvc.Output =  
[“MC DATAFILE=‘tuples.hbook’ TYP=‘HBOOK’ OPT=‘NEW’ ”]  

../solutions/MCTrees.py  

To be improved  

Vanya BELYAEV
Component Properties

**Algorithms**

```python
alg = gaudi.algorithm('MyAlg')
alg.NTupleLUN = 'LUNIT'
```

**Services**

```python
hsvc = gaudi.service('HistogramPersistencySvc')
hsvc.OutputFile = 'histo.file'
```

**Tools**

```python
MyAlg.PhysDesktop.InputLocations = ['Phys/StdLooseKaons']
```

---

1st June’2k+7  LBD group meeting

Vanya BELYAEV
The algorithm itself

```python
class MCSTrees(AlgoMC):
    """The algorithm itself""

    ## the main analysis method
    def analyze(self):
        """the main analysis method""

        # get the MDDecayFinder wrapper
        self.MDDecayFinder = self.MDDecayFinder()

        # find all NC trees of interest
        self.finder = self.finder()
        for (E, p, theta) in self.decays:
            self.finder.find()

        print('found NCtrees/Phis/5s: %s' % len(self.decays))

        # fill the histogram
        self.plot(self, self.decays, self.decays, self.decays, self.decays)
```

---

```python
# configure the job
def configure():
    """configure the job""

gaudi.config()  
```
Go from MC to RC data

• At this moment one knows how to:
  • Deal with MC trees, decays, particles
  • Perform simple (python) loops
  • Deal with the histograms & N-Tuples
    • Some knowledge of ‘configuration’

• For RC data one must perform non-trivial algorithm configuration to be able to run
  • Input for RC particles (or ParticleMaker)
  • Dependency on ‘other’ algorithms (‘PreLoad’)

1st June'2k+7  LBD group meeting
Vanya BELYAEV
Algorithm configuration

desktop = gaudi.property('MyAlg.PhysDesktop')
desktop.InputLocations = ['Phys/StdLooseKaons']

* Similar semantic in configuration (* .opts ) files:
  MyAlg.PhysDesktop.InputLocations={"Phys/StdLooseKaons"} 

../solutions/RCSelect.py
select/loop statements

muons = self.select ( 'mu',
            ( 'mu+'== ABSID ) & ( PT > (1*GeV) )
)
kaons = self.select ( 'K',
            ( 'K+'== ABSID ) & ( PIDK > 0 )
)

• Loops:
  pxis=self.loop( 'mu mu', 'J/psi(1S)' )
  phis=self.loop( 'K K', 'phi(1020)' )

..solutions/RCSelect.py
Inside the loops (I)

dmcut = ADMASS(‘J/psi(1S)’) < 50
for psi in psis :
    if not 2500 < psi.mass(1,2) <3500 : continue
    if not 0 == SUMQ( psi ) : continue
    if not 0 <= VCHI2( psi ) < 49 : continue
    self.plot ( M(psi)/1000 ,
               “ di-muon invariant mass” ,
               2.5 , 3.5 )
    if not dmcut( psi ) : continue
    psi.save(‘psi’)

psis = self.selected(‘psi’)
print ‘# of selected J/psi candidates:‘, psis.size()
Inside the loops (II)

dmcut = ADMASS('phi(1020') < 12
for phi in phis :
    if not phi.mass(1,2) < 1050 : continue
    if not 0 == SUMQ( phi ) : continue
    if not 0 <= VCHI2( phi ) < 49 : continue
    self.plot ( M( phi ) / 1000 ,
            " di-kaon invariant mass" ,
            1.0 , 1.050 )
    if not dmcut( phi ) : continue
    phi.save('phi')

phis = self.selected('phi')
print '# of selected phi candidates:', phis.size()
Inside the loops (III)

dmc cut = ADMI ASS('B_s0') < 100
bs = self. loop ( 'psi phi', 'B_s0' )
for B in bs :
    if not 4500 < B.mass(1,2) < 6500 : continue
    if not 0 <= VCHI2( B ) < 49 : continue
    self.plot ( M( B ) / GeV ,
                " J/psi phi invariant mass" ,
                5.0 , 6.0 )
    if not dmc cut( B ) : continue
    B.save('Bs')

Bs = self. selected('Bs')
print "# of selected Bs candidates: ", Bs.size()
if not Bs.empty() : self.setFilterPassed ( TRUE )

../solutions/RCSelect.py
The last step: MC-truth match

- The simplest case: check if RC particle originates from the certain MC-(sub)tree
  - The most frequent case
    - Check for efficiencies
    - Resolution
- The opposite task: what MC particle “corresponds” to RC particle
  - similar (MC\text{TRUTH} \to \text{RC\text{TRUTH}})
- NB: LoKi (and Bender) uses own concept of MC “loose” matching
  - LUG, chapter 15
MC-truth match

finder = self.mctruth('some name')

* Select MC-particles

mcBs = finder.find(
    ` B_s0 -> (J/psi(1S) -> mu+ mu-) phi(1020) ]cc ' `)

mcPhi = finder.find(
    ` phi(1020) : [B_s0 -> (J/psi(1S) -> mu+ mu-) phi(1020) ]cc ' `)

mcPsi = finder.find(
    ` J/psi(1S) : [B_s0 -> (J/psi(1S) -> mu+ mu-) phi(1020) ]cc ' `)

* Prepare 'MC-Truth cuts'

match = self.mcTruth('some name')

mcCutBs = MCTRUTH ( match , mcBs )

mcCutPhi = MCTRUTH ( match , mcPhi )

mcCutPsi = MCTRUTH ( match , mcPsi )

../solutions/RCMCSelect.py
The last step: MC-truth match

for psi in psis :
    if not mcCutPsi ( psi ) : continue
...
for phi in phis :
    if not mcCutPhi ( phi ) : continue
...
for B in bs :
    if not mcCutBs ( B ) : continue

• Alternatively :

for B in bs :
    psi = B(1)
    phi = B(2)
...
    tup.column ( 'mcpsi' , mcCutPsi( psi ) )
    tup.column ( 'mcphi' , mcCutPhi( phi ) )
    tup.column ( 'mc'    , mcCutBs ( B ) )
    tup.write()

../solutions/RCMCSelect.py
1st June'2k+7  LBD group meeting

Vanya BELYAEV

```python
# neutral combination:
if not 0 <= B mass ( psi1 ) : continue
## check the chi2 of the vertex fit
if not 0 <= VCHI2 ( psi1 ) : continue
self.plot (H[psi1] / 1000 ,
    " dimuon invariant mass ",
    2.5 , 1.5 )
if not dmPhi( psi1 ) : continue
psi1.save( psi1 )
## save J/psi
## delta mass cut for phi
dmPsi1 = ADEMASS ( psi1(1020) ) < 20
## prepare the loop over dimuons
phi1 = self.loop ( 'K' , 'phi(1020)' )
for phi in phi1 :
    ## use ONLY* Monte-Carlo cuts
    if not mcCutPhi( phi ) : continue
    ## ATTENTION: only true phi
    if phi.mass( 1,2 ) > 1050 : continue
    # neutral combination ?
    if not 0 <= B mass ( phi ) : continue
    if not 0 <= VCHI2 ( phi ) : continue
    self.plot ( H[phi] / 1000 ,
        " dimuon invariant mass ",
        1.0 , 1.050 )
    if not dmPhi( phi ) : continue
    phi.save( phi )
## delta mass cut for Bc
dmPsi2 = ADEMASS ( Bc ) < 100
## prepare the loop over psi1+phi1 combinations
hs = self.loop ( 'psi1 phi' , 'Bc' )
for h in hs :
    ## use ONLY* Monte-Carlo cuts
    if not mcCutBc ( h ) : continue
    ## ATTENTION: only true Bc
    if 0 <= Bc.mass( 1,2 ) / 1000
    if not 4.5 < m : continue
    if not 0 < VCHI2 ( Bc ) : continue
    self.plot ( H[Bc] / 1000 ,
        " psi1 phi1 invariant mass ",
        5.0 , 5.0 )
    if not dmBc ( h ) : continue
    Bc.save( Bc )
# check selected particles:
Bc = self.selected( 'Bc' )
if not Bc.empty() : self.setFilterPassed [ True ] ) # FILTER PASSED
```

1st June'2k+7  LBD group meeting
• Algorithm: 81 lines
  • 55% - comments
• Configuration & steering: 44 lines
  • 40% comments
• Select true “reconstructed” Bs with loose cuts: fine for cuts investigation
Other features, out of scope

- Nice printout of trees, particles, events
- Various “extractors” and metafunctions
- HepMC + HepMCParticleMaker
- Jets, Jets maker, LoKi-kt-Jet
- Tools for background origin studies
- Patterns
- “Hybrid”: now also for MCParticles
  - “IFilterCriterion” in python
  - “IMCParticleSelector” in python
- and much much more...

As concerns the functionality needed for analysis, Bender is full scale application, widely used for physics studies
References again...

- **Bender Pages** and **Bender pages** by *Lena Mayatskaya*
- **Bender mailing list**
- **Bender Savannah portal** (news, bugs, tasks, …)
- **Bender Tutorial**: slides & instructions
- **Bender HyperNews**, TWiki, FAQ, User Guide and Manual: 😞 not yet. still in the bottle of inc
- **Bender Examples**
  - including nice scripts from *Diego Martitez Santos* for $B_s \rightarrow \mu \mu$
  - background studies
    - getpack Ex/BenderExample v7r0
- "**Bender-helpdesk@lhcb.cern.ch**"
- **1-R-010 at CERN**
- +41 (0) 22 767 89 28