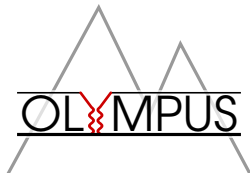


Offline analysis

Jan C. Bernauer



Collaboration Meeting September 2011



Massachusetts Institute of Technology

Analysis will be divided in steps, roughly:

- from raw data to hits.
- from hits to tracks.
- from tracks to physics.

Analysis will be divided in steps, roughly:

- from raw data to hits.
- from hits to tracks.
- from tracks to physics.

But we also need independent steps, e.g. calibration.

Try to not reinvent the wheel for every step.

Framework:

- Works on root trees: One input, zero or one output trees
- Individual Functions (Detector, Track reco, ...) packaged in libraries.
- XML based init file, XML based recipe file
- One driver program.

Command line syntax

```
cooker (<options>) <recipe> <input> (<output>)
```

Transforms the tree in the <input> root file into a tree in the <output> root file, using the recipe.

- Self contained: only recipe and input file name needed.
- Cooker handles XML parsing, plugins handle tree data.
- Cooker does not depend on tree structure. One program for all steps.

Recipe XML

```
<?xml version="1.0" encoding="UTF-8" ?>
<olympus>
  <init>init/demo.xml</init>
  <source>ORT</source>
  <destination>tofhits</destination>

  <plugins>
    <plugin> <name>ToF</name> <file>libToF</file> </plugin>
  </plugins>

  <startup>
    <ToF>startup</ToF>
  </startup>

  <execute>
    <ToF>process</ToF>
  </execute>

  <finalize>
    <ToF>done</ToF>
  </finalize>
</olympus>
```

Recipe XML

```
<?xml version="1.0" encoding="UTF-8" ?>
<olympus>
  <init>init/demo.xml</init>
  <source>ORT</source>
  <destination>tofhits</destination>

  <plugins>
    <plugin> <name>ToF</name> <file>libToF</file> </plugin>
  </plugins>

  <startup>
    <ToF>startup</ToF>
  </startup>

  <execute>
    <ToF>process</ToF>
  </execute>

  <finalize>
    <ToF>done</ToF>
  </finalize>
</olympus>
```

Recipe XML

```
<?xml version="1.0" encoding="UTF-8" ?>
<olympus>
  <init>init/demo.xml</init>
  <source>ORT</source>
  <destination>tofhits</destination>

  <plugins>
    <plugin> <name>ToF</name> <file>libToF</file> </plugin>
  </plugins>

  <startup>
    <ToF>startup</ToF>
  </startup>

  <execute>
    <ToF>process</ToF>
  </execute>

  <finalize>
    <ToF>done</ToF>
  </finalize>
</olympus>
```


Recipe XML

```
<?xml version="1.0" encoding="UTF-8" ?>
<olympus>
  <init>init/demo.xml</init>
  <source>ORT</source>
  <destination>tofhits</destination>

  <plugins>
    <plugin> <name>ToF</name> <file>libToF</file> </plugin>
  </plugins>

  <startup>
    <ToF>startup</ToF>
  </startup>

  <execute>
    <ToF>process</ToF>
  </execute>

  <finalize>
    <ToF>done</ToF>
  </finalize>
</olympus>
```

Recipe XML

```
<?xml version="1.0" encoding="UTF-8" ?>
<olympus>
  <init>init/demo.xml</init>
  <source>ORT</source>
  <destination>tofhits</destination>

  <plugins>
    <plugin> <name>ToF</name> <file>libToF</file> </plugin>
  </plugins>

  <startup>
    <ToF>startup</ToF>
  </startup>

  <execute>
    <ToF>process</ToF>
  </execute>

  <finalize>
    <ToF>done</ToF>
  </finalize>
</olympus>
```

Recipe XML

```
<?xml version="1.0" encoding="UTF-8" ?>
<olympus>
  <init>init/demo.xml</init>
  <source>ORT</source>
  <destination>tofhits</destination>

  <plugins>
    <plugin> <name>ToF</name> <file>libToF</file> </plugin>
  </plugins>

  <startup>
    <ToF>startup</ToF>
  </startup>

  <execute>
    <ToF>process</ToF>
  </execute>

  <finalize>
    <ToF>done</ToF>
  </finalize>
</olympus>
```

Init XML

```
<?xml version="1.0" encoding="UTF-8" ?>
<olympusinit xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:noNamespaceSchemaLocation='demo.xsd'
xmlns:xi="http://www.w3.org/2001/XInclude">

  <xi:include href="../src/detector/ToF/init/ToF.xml" parse="xml"/>

  <ToF>
    <date time="2011-01-01T00:00:00">
      <tdcoffset> <id id="0">20</id> </tdcoffset>
    </date>
  </ToF>
</olympusinit>
```

Init XML

```
<?xml version="1.0" encoding="UTF-8" ?>  
<olympusinit xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"  
xsi:noNamespaceSchemaLocation='demo.xsd'  
xmlns:xi="http://www.w3.org/2001/XInclude">
```

```
<xi:include href="../src/detector/ToF/init/ToF.xml" parse="xml"/>
```

```
<ToF>  
  <date time="2011-01-01T00:00:00">  
    <tdcoffset> <id id="0">20</id> </tdcoffset>  
  </date>  
</ToF>  
</olympusinit>
```

Init XML

```
<?xml version="1.0" encoding="UTF-8" ?>
<olympusinit xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:noNamespaceSchemaLocation='demo.xsd'
xmlns:xi="http://www.w3.org/2001/XInclude">

  <xi:include href="../src/detector/ToF/init/ToF.xml" parse="xml"/>

  <ToF>
    <date time="2011-01-01T00:00:00">
      <tdcoffset> <id id="0">20</id> </tdcoffset>
    </date>
  </ToF>
</olympusinit>
```

- Official repo is *olympus_release* on *oweb*.
- Each group should have own repo clone, either on own accessible computer or *oweb*. So far:
 - packaging repo: *olympus*
 - Lumi-GEMs: *olympus_lumigem*
 - MWPC: *olympus_MWPC*
 - ToF: *olympus_ToF*
 - personal repos: *olympus_axel*, *bsh*, *funke*, *winnebeck*
- For access: I need the public key!
- Commit often!
- Release: Send me a pull request with id and repo location. I will merge, test and release.

Up to now, simulation is MIT svn repo. Merge necessary!

- Will import FULL history.
- Switch will take place 9/26/2011.
- Merge your stuff before 9/25.
- Get access to the git repo!

Short term:

- *ocompile* not meant for large jobs.
- Options:
 - Develop on personal computers + *ocompile*, get files via AFS
 - Buy a development system: about 2k Euro for 16 cores, 8GB Opteron 2.2GHz in 1U.

Short term:

- *ocompile* not meant for large jobs.
- Options:
 - Develop on personal computers + *ocompile*, get files via AFS
 - Buy a development system: about 2k Euro for 16 cores, 8GB Opteron 2.2GHz in 1U.

Long term:

- For bulk calculations (Track finding, etc.)
 - Buy our own cluster
 - MIT cluster
 - access only for MIT people
 - DESY cluster
 - might cost money

Objectives

Short term (2 month)

- Integrate slow control data
- For each detector: From raw data to hits, and from MC - hits to raw data.
- Data quality monitoring: Is the data any good?
 - Efficiency

Objectives

Short term (2 month)

- Integrate slow control data
- For each detector: From raw data to hits, and from MC - hits to raw data.
- Data quality monitoring: Is the data any good?
 - Efficiency

Mid term (before February beamtime):

- Luminosity
- More data quality monitoring
 - Elastic rate
- Tracks

Objectives

Short term (2 month)

- Integrate slow control data
- For each detector: From raw data to hits, and from MC - hits to raw data.
- Data quality monitoring: Is the data any good?
 - Efficiency

Mid term (before February beamtime):

- Luminosity
- More data quality monitoring
 - Elastic rate
- *Tracksbsh*

Long term:

- Reiterate
- Physics analysis

Detectors:

- Slow control in cooker: Anton, Christian, JCB
- ToF: Inti, (Alexander W., JCB),...?
- MWPC: Alexander K., ...?
- Lumi-GEM: Jürgen, Alexander W., ...?
- Moeller: ?
- Wire Chambers: Doug, MIT students,... ?

Other tasks:

- Tracks: Doug, Alexander K., MIT students, JCB... ?
- MC: Doug, MIT students, DETECTOR GROUPS!
- Luminosity: ?

Further Info

Analysis Meeting

Sign up on the doodle poll:

<http://doodle.com/6yew5vf9pqpdztix>

Mailing list

Sign up for olympus-offline:

<http://mailman.mit.edu/mailman/listinfo/olympus-offline>

Read/Edit the WIKI

Start point for offline analysis:

<https://olympus-docu.hiskp.uni-bonn.de/dokuwiki/doku.php?id=ana-doc>

Additional slides

Recipe XML (1)

Header

```
<?xml version="1.0" encoding="UTF-8"?>  
<olympus>
```

Define which init file to use

```
<init>init/demo.xml</init>
```

Input and output tree name ~ tree type

```
<source>ORT</source>  
<destination>tofhits</destination>
```

Plugin definitions

```
<plugins>  
  <plugin> <name>ToF</name><file>libToF</file></plugin>  
  ...  
</plugins>
```

Recipe XML (2)

Startup routines

```
<startup>  
  <ToF>startup</ToF>  
  ...  
</startup>
```

Per Event routines

```
<execute>  
  <ToF>process</ToF>  
  ...  
</execute>
```

Finalize routines

```
<finalize>  
  <ToF>done</ToF>  
  ...  
</finalize>
```