Creating a Driver using JAS3

HowTo write your own Analysis

This little tutorial will help you build your own analysis code step-by-step. After finishing this section you should be able to write a complete analysis inside the JAS3 framework.
You will learn what components are necessary to make an analysis Driver. The Driver presented here will load an existing Driver that puts an object in the event, read that object back from the event and plot a property of that object.
Let's get started: Create a new file with `File->New->Java File`. Save it as MyExampleClass.java.
At first we need to create a class that inherits from the `Driver` class:

```
import org.lcsim.util.Driver;

public class MyExampleClass extends Driver {

}
```

You will have to add a constructor in order for JAS3 to be able to load your class. The constructor **must be `public` and may take no arguments !**

```
public MyExampleClass(){
};
```

If you want to use many Drivers at the same time, you can compile and load them one by one. However, if your code depends on the order of Drivers (for example if you have a Driver that does the jetFinding in the event and your Driver takes the jets back from the event), you may want to use the `add()` member function.
So, let's add the other Drivers:

```
add(new MyJetFindingDriver()); // we rely on the output of this Driver so we use the add() member function to
ensure it's processed in the correct order
add(new Analysis101()); // we can also load many other Drivers with add()
```

The next essential component of our new Driver is the `process()` method. If you have added child Drivers, you **must** call `super.process(event)` to ensure that the child Drivers process the event first.

```
public void process(EventHeader event) {
    super.process(event); // this takes care that the child Drivers are loaded and processed.
}
```

You can now load the list of jets that the child Driver `MyJetFindingClass` has put in the event.

```
List<List<MCParticle>> jetList = (List<List<MCParticle>>) event.get("jetList");
aida.cloud1D("nJets").fill(jetList.size());
```

The complete file now looks like this.

```
import java.util.List;

import org.lcsim.event.EventHeader;
import org.lcsim.event.MCParticle;
import org.lcsim.util.Driver;
import org.lcsim.util.aida.AIDA;

public class MyExampleClass extends Driver {
    private AIDA aida = AIDA.defaultInstance();
    public MyExampleClass(){
        add(new MyJetFindingDriver());
        add(new Analysis101());
    };
    public void process(EventHeader event) {
        super.process(event); // this takes care that the child Drivers are loaded and processed.
        List<List<MCParticle>> jetList = (List<List<MCParticle>>) event.get("jetList");
        aida.cloud1D("nJets").fill(jetList.size());
    }
}
```

Save the file.
Open the file MyJetFindingDriver.java and compile it.
Open the Analysis101 example and compile it.
Compile and load the MyExampleClass example.