



## **EUTelescope, the JRA1 tracking and reconstruction software: a status report (Milestone)**

A. Bulgheroni<sup>\*</sup>,

P. Roloff, J. Behr<sup>†</sup>,

A. F. Zarnecki<sup>‡</sup>,

Y. Furltova<sup>§</sup>

December 16, 2008

### **Abstract**

The main goal of the JRA1 within the EUDET project is the construction of new and the improvement of existing test beam infrastructures. The JRA1 collaboration developed a high resolution, low material pixel telescope to characterize Devices Under Test. Along with this hardware instrument, a software tool performing all the off line procedures needed to extract from the data acquired by the DAQ the precise spatial information and the DUT figures of merit has been developed as well and made available to the community.

This paper is complementing the first status report of the software already described one year ago in the EUDET-Memo-2007-20 with the latest developments.

---

<sup>\*</sup>INFN, Milano, Italy

<sup>†</sup>DESY, Hamburg, Germany

<sup>‡</sup>Institute of Experimental Physics, University of Warsaw, Warsaw, Poland

<sup>§</sup>University of Bonn, Bonn, Germany

# 1 Introduction

The main goal of the JRA1 within the EUDET project is the construction and the improvement of test beam infrastructures. One of the main activity currently on going is the development of a high resolution, low material pixel telescope along with a dedicated software framework called EU Telescope. The main objectives of EU Telescope together with the overall analysis strategy and some results were already deeply described in the EUDET-Memo-2007-20. In this paper, an exhaustive report about the new features introduced this year is presented.

During this summer test beam campaign many DUT users were willing to try to use or at least to compare the results obtained with their own software with EU Telescope. It was immediately clear that while the software framework was definitely working fine for the pixel telescope, a few minor corrections have to be introduced in order to properly deal with a data stream coming from the reference telescope and one or more other coming from a detector under test. Together with these small changes also a request for more usability was raised to make the transition from user code to EU Telescope as seamless as possible.

## 2 New and revised features

According to Marlin<sup>1</sup> prescription, each feature is implemented into a separate module also called Processor performing a specific operation on an input collection of data and consequently producing an output collection or modifying data already existing in the current event. Here below, the new features introduced since the previous Memo are described along with some other processors modified to fulfill specific user needs.

### 2.1 EUTelNativeReader

At the beginning of the development it was decided to have the data acquisition system writing the acquired data directly in LCIO<sup>2</sup> format, but even if a feasibility test proven that this was possible, both the DAQ and the analysis experts decided to keep a very simple platform and system independent native format that can be decoded also by users not using the ILCSOFT infrastructure. This native data format is promptly converted into the LCIO data collections used by EU Telescope to perform all the analysis steps. For this purpose the **EUTelNativeReader** processor has been coded with the aim of converting all the sub events found within the input native data stream. In this way, an external user, willing to use EU Telescope for his own analysis, has just to write a little excerpt of this universal reader specifying how his detector data should be decoded from the native stream and re-encoded into the LCIO data model. The idea behind this reader is the universal replacement for the EUTelMimoTelReader that was focused on the conversion

---

<sup>1</sup>Marlin: Modular Analysis and Reconstruction for the LINear collider ([http://ilcsoft.desy.de/portal/software\\_packages/marlin/](http://ilcsoft.desy.de/portal/software_packages/marlin/))

<sup>2</sup><http://lcio.desy.de/>

of events containing only Mimotel sensors and finally also Mimosa 18.

The `EUTelNativeReader` will guess from the Beginning Of Run Event (BORE) which kind of sensors are present in the telescope setup and fill in a list of `EUTelBaseDetector` derived classes, each of them is implementing a precise sensor configuration. For each group of sensors there will be a common output collection containing the raw data information.

## 2.2 EUTelCorrelator

This is a very useful diagnostic tool that users can use to understand if the data quality is acceptable and if the geometry description they provided is effectively compliant with the reality. **EUTelCorrelator** is taking as input either a collection of clusters (group of pixels on the same sensor) or a collection of hits (space points in the global telescope frame of reference) and filling series of two dimensional plots showing the correlation of the cluster / hit position with respect to all other cluster / hit not belonging to the same plane.

## 2.3 EUTelMAPSDigi

This was the last piece of code needed to make our software a complete suite for analysis and reconstruction. **EUTelMAPSDigi** is digitization processor. A realistic model based on the beam test results is used to calculate the expected response of MAPS sensors for the Geant4 simulated hits, as returned by Mokka. The conversion from global reference frame to local reference frame of individual sensor, based on the GEAR geometry description, was adopted from `EUTelHitMaker`. Each Mokka hit is divided into a number of smaller steps (based on energy deposit and path length). For each step corresponding charge deposit in sensor pixels is calculated by 2D integration of the expected charge density over the pixel surface. Charge capture in silicon (signal attenuation) and charge reflection from the epitaxial layer boundary are also taken into account. The core of the algorithm is implemented in TDS (Track Detailed Simulation) package by Piotr Niezurawski ([pniez@fuw.edu.pl](mailto:pniez@fuw.edu.pl)). With this new processor, users can simulate realistic reference data sample and estimate their DUT figures of merit before collecting real test beam data.

## 2.4 EUTelMille

The **EUTelMille** processor is the core of the new alignment processor that is taking advantage from Millepede<sup>3</sup> a professional tool used for precise alignment of detectors with thousands of degrees of freedom. In the case of the pixel telescope, the number of parameters is quite limited but nevertheless the use of Millepede guarantee an unbiased determination of the best parameters. `EUTelMille` loops over all desired events containing hits in the global telescope frame of reference and calculate local fit results. The

---

<sup>3</sup><http://www.desy.de/~blobel/mptalks.html>

latest version of EUTelMille is also able to estimate a good starting point for the global minimization process and run pede directly inside Marlin.

## 2.5 EUTelPedestalNoiseProcessor

This processor has been revisited to comply with some specific user requests, like the possibility to perform a Common Mode rejection on a row wise base instead of full sensor and different dead / noisy / hot pixels rejection algorithms. These new features are now available to the user community.

## 2.6 Histogramming package

This is not really a Marlin processor but a very useful set of ROOT<sup>4</sup> macros opening the histogram files produced by the EUTelescope processors and showing relevant histograms and profiles making the data quality monitor much simpler and more effective.

## 3 GRID installation

The EUTelescope software, being part of the ILCSoft, has been installed on the Experiment Software Area of all the Computing Elements accepting job submission by ILC VO members. In this way, users with a limited knowledge and experience of the GRID can perform their analysis using EUTelescope in a very simple way being all the needed libraries already installed and tested on the target working node. The developers are also thinking of preparing a submission script running standard analysis steps either on the local computer or, if needed, directly on the GRID exploiting parallel processing. This new installation has been used to carry on all the analyses done on the summer 2008 test beam campaign.

## 4 Conclusion and Future development

The tracking and reconstruction software for the beam telescope is ready and working according to the specification, hence the corresponding milestone has to be considered accomplished. Nevertheless, the framework is such that future developments and modifications are still possible especially in view of integrating more and more detectors under test. In this respect the developers will continue working in collaboration with end users trying to help them in getting the best from what was coded so far, and eventually implement what is required by their specific needs and still missing in the framework. Moreover, the new digital telescope will soon be available and a couple of modifications have to be included in the existing processors to make them compatible with a binary detector type.

---

<sup>4</sup><http://root.cern.ch>

## **Acknowledgement**

This work is supported by the Commission of the European Communities under the 6<sup>th</sup> Framework Programme "Structuring the European Research Area", contract number RII3-026126.