

# Table of Contents

<b>Introduction.....</b>	<b>1</b>
<b>Source Code.....</b>	<b>2</b>
Supported platforms.....	2
Dependencies.....	2
GIT repository @ gitlab.cern.ch.....	2
Structure.....	2
Compiling.....	3
<b>Running the code.....</b>	<b>4</b>
<b>testBeamAna containers and functions.....</b>	<b>5</b>
<b>testBeamAna results.....</b>	<b>6</b>

# Introduction

Documentation and examples of testBeamAnalysis. Intended as a clustering analysis for RAW FE-I4 data from the RCE DAQ system (USBPix - in progress).

**STATUS:** Under development!

# Source Code

## Supported platforms

- Mac OSX
- RHEL
- Runs on LXPlus

## Dependencies

The software was developed in C++98 and with a bash shell. The analysis itself can be run as is but if use of the plotting functions is required, access to ROOT [libraries](#) is needed and is expected to be compiled and findable. This framework was developed with version 5.34.

## GIT repository @ gitlab.cern.ch

The source code can be found in this git repo:

- URL: <https://gitlab.cern.ch/TestBeamTeam/TestBeamAna.git>

Contact the author to get access. Require a Cern computing account.

## Structure

The code is structured to facilitate easy merging with multiple developers (hopefully!). There is a separate folder (subdirectory) for what we think is each main stage of the analysis:

- Data I/O
- Clustering
- Tracking/alignment
- Analysis (a general umbrella that utilises the products of the previous 3 items on this list)

Each subdirectory has the same structure and a makefile that expects files to be organised accordingly. Here is a list of folders inside each subdirectory:

- Makefile : Makefile to compile the code
- README : Textfile with basic information about what this subdirectory contains, how to run some of the tests.
- ana/ : Analysis objects and namespaces (all source files)
- bin/ : Folder for compiled executables
- build/ : Folder for compiled objects
- src/ : Folder for entry points/main functions/test programs

To help the makefile distinguish between source files and test/main files it is required that: source/class/namespace files end in '.cpp', header files end in '.h', and main files to end in '.cxx'. This is just to help keep the code organised.

Because of the identical subfolder structure switching development from one task to another should be fairly simple.

Since the idea is that all of these subdirs contain analyses that will mostly be used together, if not sequentially,

they should all share the same 'tools' or container objects. These are all stored in the shared 'tools' directory and are detailed later in this page.

## Compiling

The makefile should take care of everything. You have two options with this framework: either compile and run just in the subdir or compile everything in the base directory. The base makfile simply calls all other makefiles in parallel but can also be called to compile sequentially if necessary. If you want to use methods from other subdirectories simply add them to the makefile but the framework is designed such that anything that requires use of multiple subdirs should be made in the Analysis subdir. Examples to follow.

Expected output in subdir make:

```
$ make
[Compiling] build/Algorithm.o
[Compiling] build/Cluster.o
[Compiling] build/DataReader.o
[Compiling] build/Event.o
[Compiling] build/Hit.o
[Compiling] build/Plot.o
[Compiling] build/RCEDataReader.o
[Creating] build/libTB_ana.a
[Compiling] build/sgnTest.o
[Linking] bin/sgnTest
[Compiling] build/testMain.o
[Linking] bin/testMain
.
.
.
```

# Running the code

**Currently** the main executable is:

- anaData

**Currently** the only parameter passed to the program is the raw data file. This may change so watch the repo for updates. Run like:

```
./bin/anaData $PATH/filename.dat
```

# testBeamAna containers and functions

Short description of some of the tools that can be used in an analysis. Click the links for more details.

Name	Filename	Author	Description
DataReader - base class	DataReader.h	Rebecca	Raw data reader base class.
RawDataReader - derived class	RawDataReader.h	Rebecca	RCE raw data formatted record file reader and import
RootDataReader - derived class	RootDataReader.h	Rebecca	RCE ROOT data formatted record file reader and import
Event	Event.h	Rebecca	Event object, stores all hits with the same LV1ID
Algorithm	Algorithm.h	Rebecca	Clusters hits in each event within 1 bcid of each other. Please see class description for details.
Cluster	Cluster.h	Rebecca	Cluster object, stores hits and other cluster information.
Hit	Hit.h	Rebecca	Hit object, stores the bcid, tot, column, and row of each hit.
Plot	Plot.h	Rebecca	Plotting functions, makes use of root libraries to save and display plots.

# testBeamAna results

A link to the graphs from analysis

Sensor type	Angle [degrees]	Run, date (dd/mm/yy), log file	Description	Data added by	Date added (dd/mm/yy)	plots (CCE)
3D	2	lorem ipsum	dolor	bla	sometime	link
Planar	2	lorem ipsum	dolor	bla	sometime	link

---

## Major updates:

-- RebeccaCarney - 30-November-2014

---

This topic: Main > TestBeamAna

Topic revision: r3 - 2015-06-16 - RebeccaCarney



Copyright &© 2008-2019 by the contributing authors. All material on this collaboration platform is the property of the contributing authors.

Ideas, requests, problems regarding TWiki? Send feedback