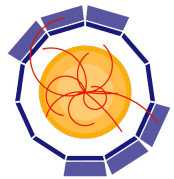


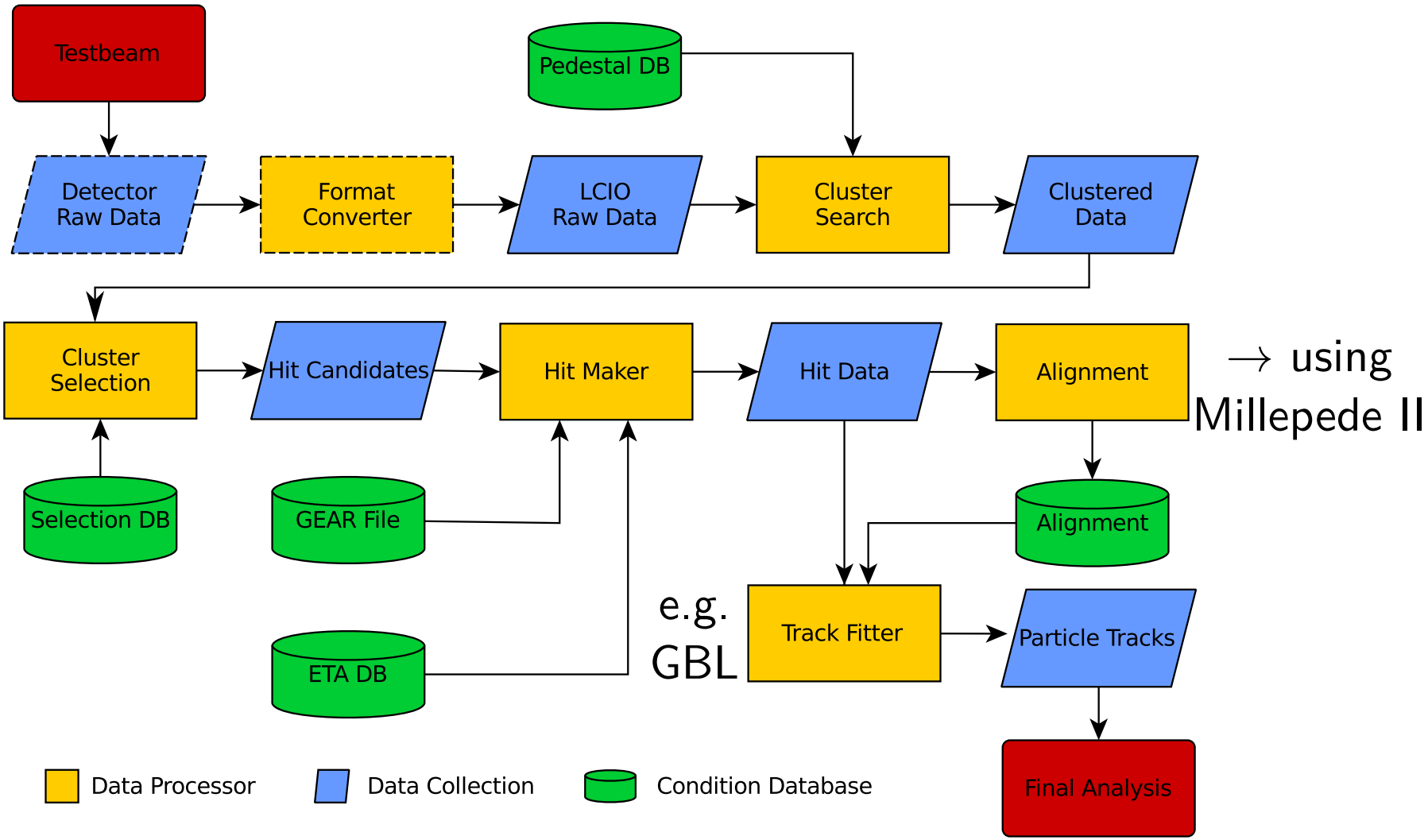
AIDA

Extending the EUTelescope Data Analysis Framework for the AIDA Telescope

Tobias Bisanz, Thomas Eichhorn, Phillip Hamnett, Denys Lontkovskyi,
•Hanno Perrey, Igor Rubinskiy, Simon Spannagel, Hongbo Zhu

EUTelescope Analysis Framework

- Library of processors running in ILCSoft's Marlin framework
- Step-wise transition: single pixel array → fitted tracks in global frame



Toward a User-Friendly "no-duct-tape" Framework

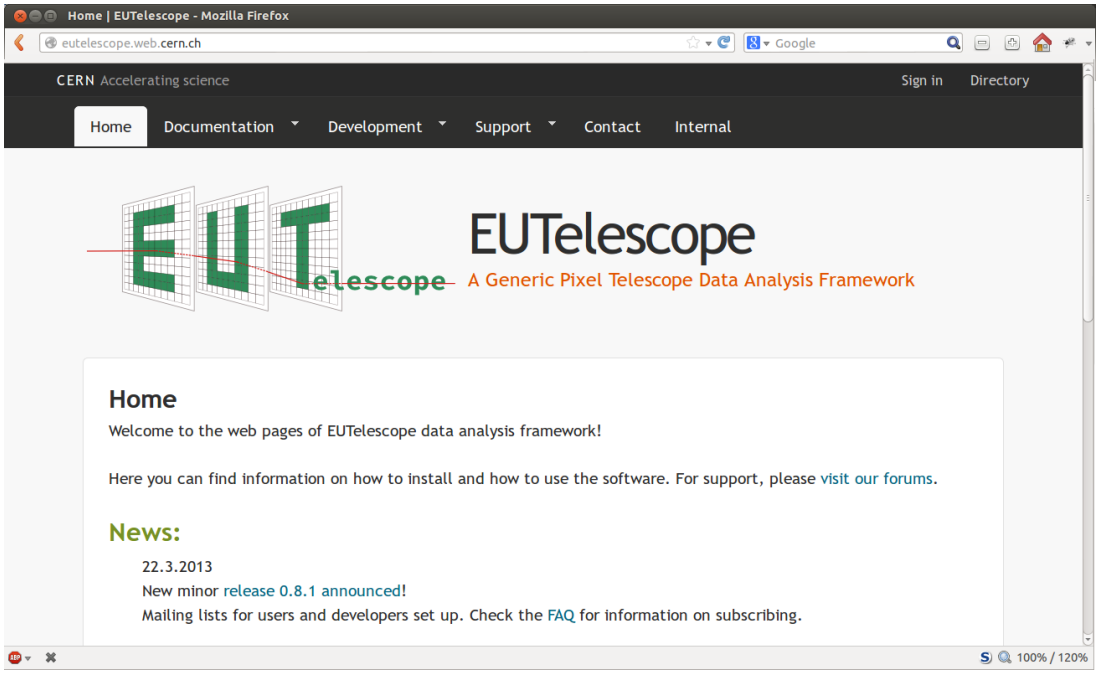
Thanks to the busy test beam in 2013, we have had new users (almost) every week.
With releases 0.8.x (from March '13) and 0.9.x (from August '13) we've focused on the user experience:

- easy installation $\rightarrow$ simplified+verified installation on many platforms
- working examples $\rightarrow$ examples $\widehat{=}$ test cases
- stability $\rightarrow$ maintaining releases through bug fixes
- centralized documentation $\rightarrow$ new web pages
- a place to ask questions $\rightarrow$ new online forum

Goal: Easier start/smoother learning curve/better usage experience

Large collaborative effort in the last $\sim$ 14 months: already addressed many issues in recent releases!

http://eutelescope.desy.de



central location for. . .

- announcements
- documentation
  - step-by-step guides for installation/updating
  - hints for new developers
  - FAQs
  - . . .
- support forums
- issue tracker

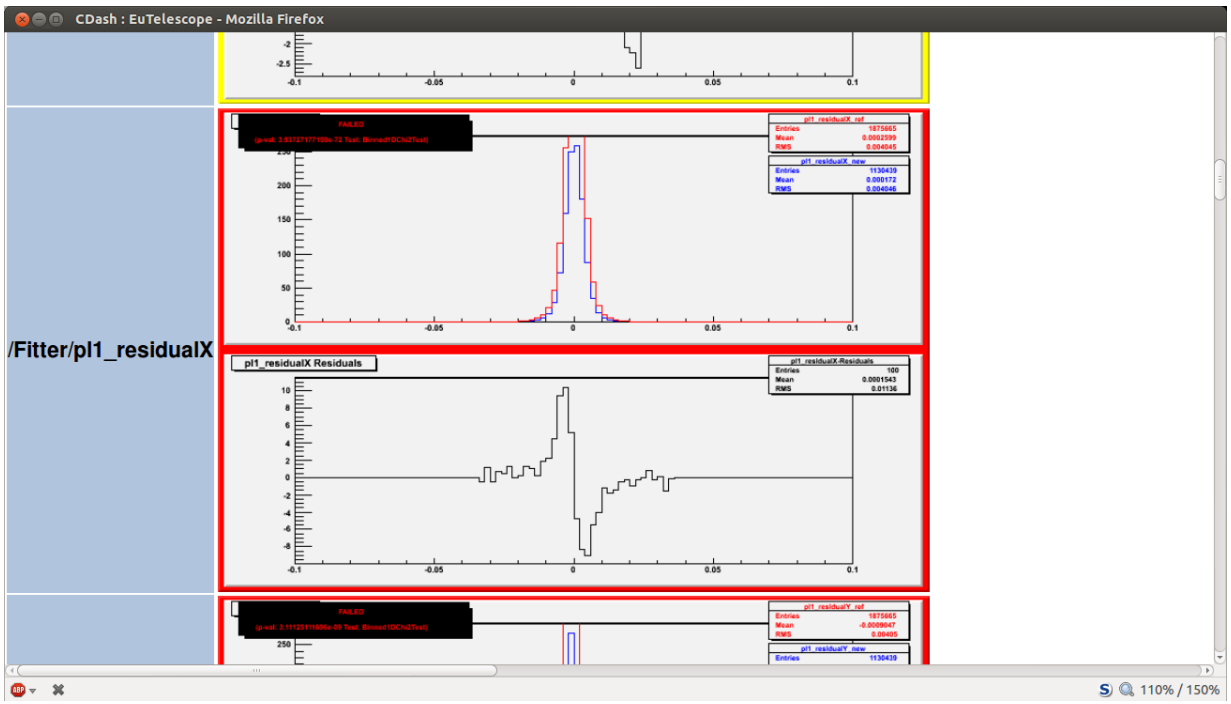
- Using drupal content management system (hosted by CERN)
  $\rightarrow$ easy to add content and to keep it up-to-date
- Actively frequented by users:
  $\sim$ 280 posts in 50 discussions over 12 months

Important tool for user support and collaboration!

Code Validation Through Nightly Data-Driven Tests

- Regression tests: ensure stability of software & analysis code
- Run nightly: code checkout, configure, build, tests → submit to server
- Fail/pass conditions determined from exit status & shell output
- Complex output (e.g. ROOT) validated through external tools
- Warn devs via mail if tests fail

⇨ Test ability to run code and ensure consistency with previous results



CTest/CDash record & visualize:

- compilation messages
- fail/pass conditions
- console output
- test duration
- numeric & graphical test results

Overview of recent and ongoing changes

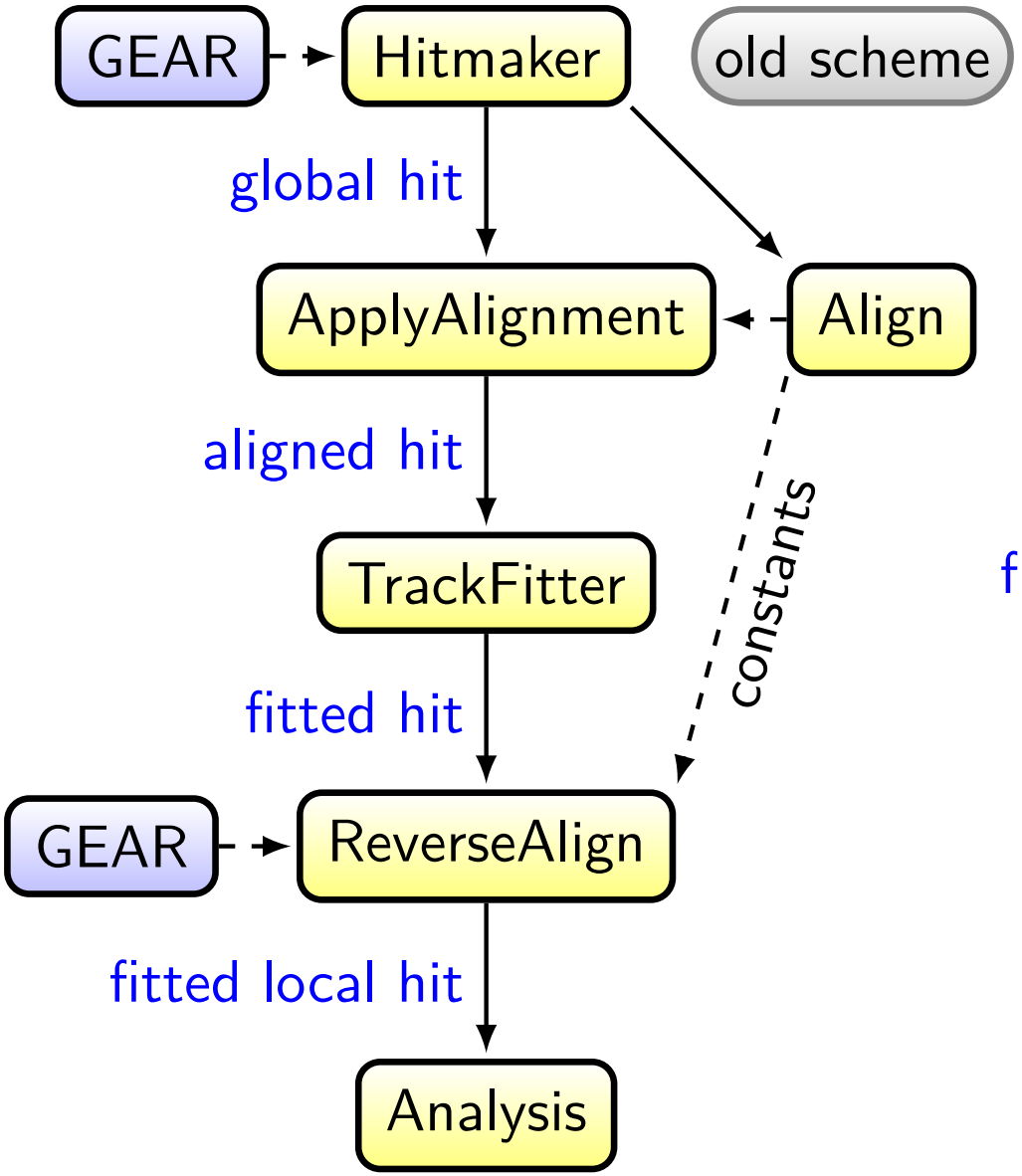
- Tracking based on General Broken Lines (GBL)
  - better treatment of multiple scattering due to inactive material
- Revised coordinate transformation handling (see next slides):
  - new central geometry class for transfer from local $\leftrightarrow$ global coordinate frame based on ROOT::TGeo
  - Alignment outputs aligned geometry description (GEAR)
    $\rightarrow$ allows iterative alignment
- Make the framework more closely follow ILCSoft's Marlin paradigm:
  - single-purpose processors that can be flexibly combined or extended
  - better exploit object linking and object properties in LCIO
- Revising clustering to be more flexible/generalized

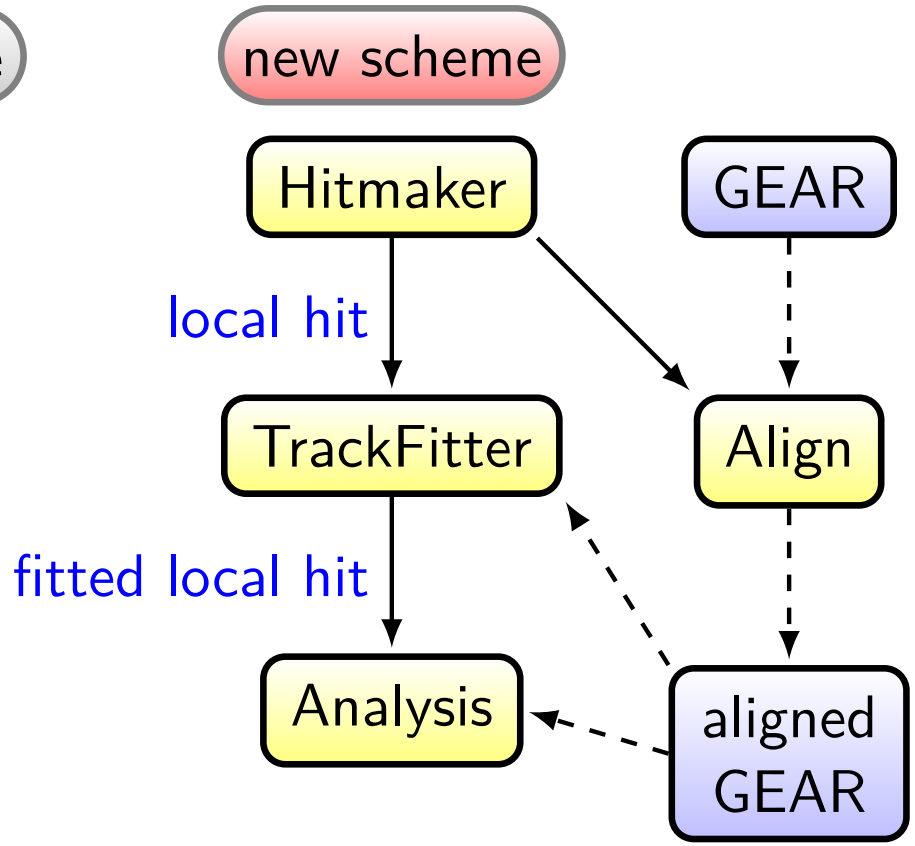
$\Rightarrow$ Simplifies framework, reduces code paths and run time

- Moved to git/GitHub – makes it *much* easier to contribute back
- Provide scripts to run analysis on NAF2 resources

Important changes – require changes to existing analyzes, however

Geometry Transformations in EUTelescope 1.0





- Keep hits in local frame
- Transform into (aligned) global frame on-the-fly

Challenges for Analysis of AIDA Telescope Data

- LCIO data format used by EUTelescope is inherently event-based
- Current assumption of one data block for every trigger from every device not valid for AIDA telescope DAQ:
  - data blocks potentially span many triggers, contain timestamps only

$\Rightarrow$ Prior to track finding, the data streams have to be merged according to trigger ID/timestamp

- Online monitoring faces same challenges $\rightarrow$ development strategy:
  1. start with offline only merging tool
     $\rightarrow$ option of using EUTelescope for immediate offline-analysis
  2. then extend to networked, partial data stream merger running within/next to EUDAQ

'Semi-online' Monitoring: Immediate Offline Analysis

- Challenges of data stream merging faced by online monitor as well but can be both CPU and IO intensive
- Offline analysis chain could be tuned to provide quick (mid-run if flushing mechanism in place) results for feedback during data taking
- Advantages to using EUTelescope:
  - very modular approach
  - all work spent on integration benefits the later analysis
  - more than monitoring: complete tracking (including alignment) possible
- Requirements:
  - write merging processor (*done* for EUDAQ1.0 data format)
  - provide safe access to data written by EUDAQ mid run (flush & transfer data, while limiting IO and net usage)
  - produce monitoring/correlation histograms useful during data taking (already largely *done*)
  - provide geometry information up front (e.g. within EUDAQ)
  - optimize the analysis chain to work out-of-the-box in as many cases as possible (on-going work of the past year)

Summary and Outlook

- EUTelescope is a modular framework for test beam data analysis from clustering to fitted tracks in a global frame
- Besides new features, we have put a lot of effort into making EUTelescope a reliable tool especially for the many new users.
- Version 1.0 will bring completely revised geometry and coordinate system transformation back-end
- Work on (online) data stream merging tool needed for AIDA telescope – progressing, but could use support!
- Long-term goal: EUTelescope usable with minimal interaction directly after data-taking $\rightarrow$ useful for (semi-)online monitoring

To follow the development, consider signing up for *eutelescope-users* and *eutelescope-developers* mailing lists at e-groups.cern.ch, and/or sign up for forum notifications at eutelescope.desy.de

Overview Backup Slides

Running tests locally

- run CTest to verify code changes before committing to SVN
- prerequisites:
  - data & known-good results (currently on DESY AFS)
  - tests enabled and test files included in `CMakeLists.txt`
- available tests:
  - static code analysis using `cppcheck`
  - analysis chain for Datura telescope @DESY w/o DUT
  - analysis chain for Anemone telescope with ATLAS FEI4
  - check of output files with LCIO tools
  - verification of results (e.g. residuals) with `stattest` (from GEANT4)
- configure & modify tests through
  - example configs & steering files
  - `testing.cmake` file in example directory
- example CTest commands:

```
cd $EUTELESCOPE/build # output will be in ./Testing
ctest -N # list available tests
ctest -I 2,6 # run tests 2 to 6
ctest -V # run all tests, show output
```