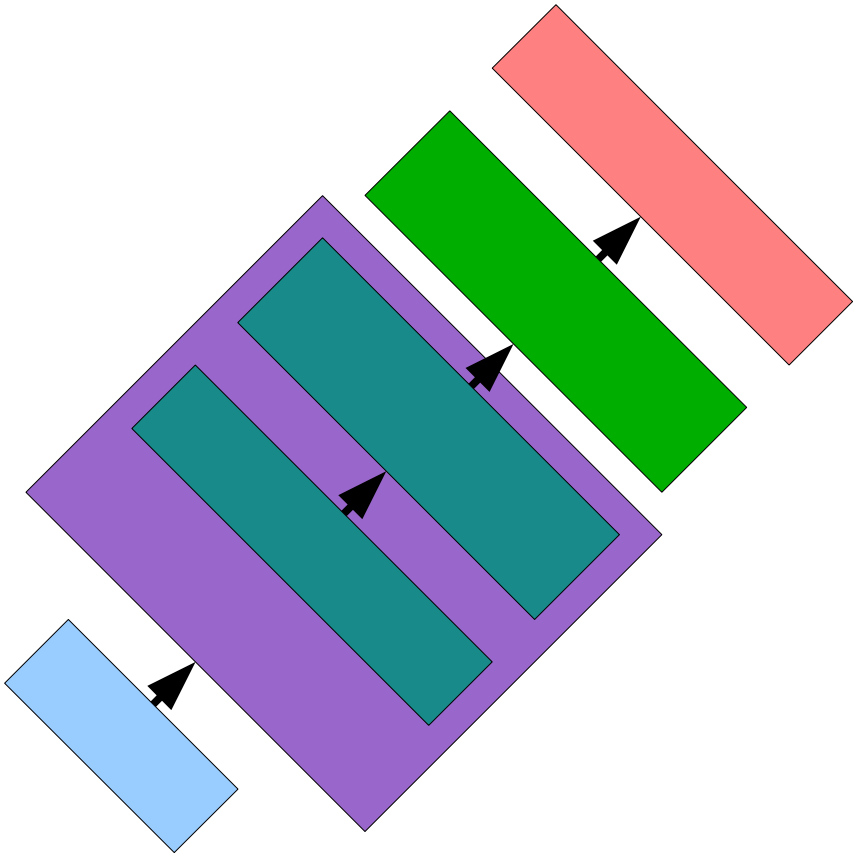
LAND02 – as we love to hate it



Håkan T. Johansson, Chalmers, Göteborg

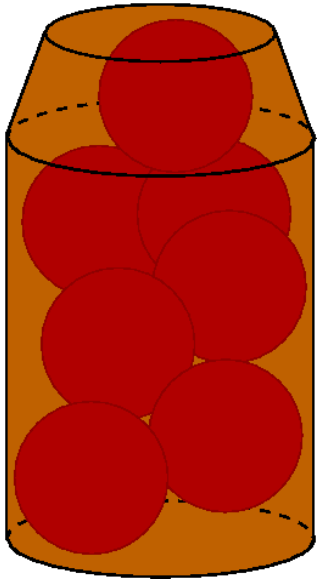
GSI, Darmstadt, May 2009

allies of LAND02

Interested and committed PhD students and postdocs, who 'had the priorities right', and did

not ask what the programs can do for you

but what you can do for the future experiments



gcc - .cc – compiler

make - .mk – build system

perl - .pl – scripts (glue)

flex - .lex – lexical analyser

bison/yacc - .y – parser generator

cpp - .hh – C preprocessor

---

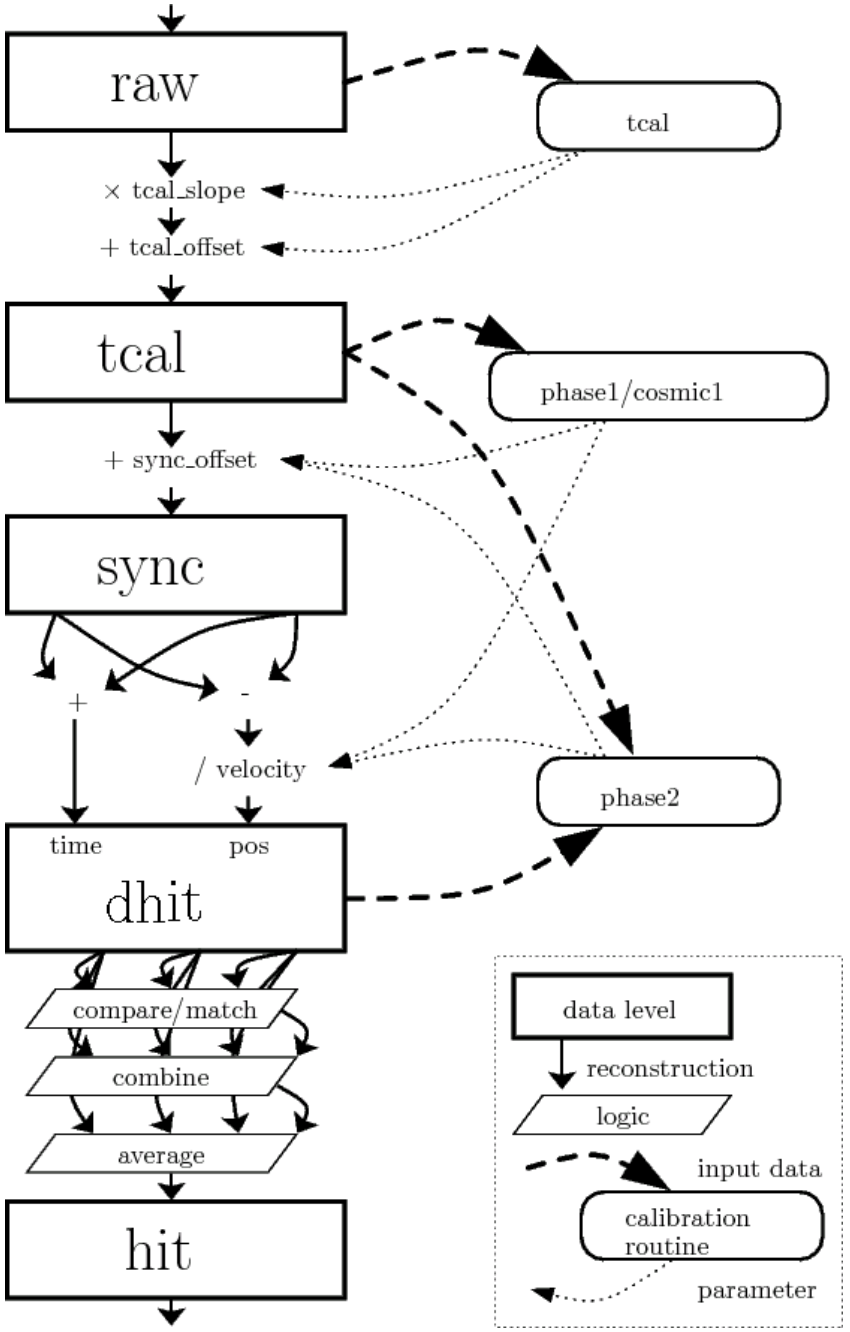
paw – analysis plotting results

root – more analysis and plots

(octave - .m – non-event plots)

LAND02 is not a package of programs.
It is a collection of source code which can be compiled into programs!

land02 - unorthodox?



No – just taking landpaw to the limits.

Or perhaps:

From a computing point of view,
it is quick-n-dirty and has the focus
on getting the job done!

Application of calibration parameters:

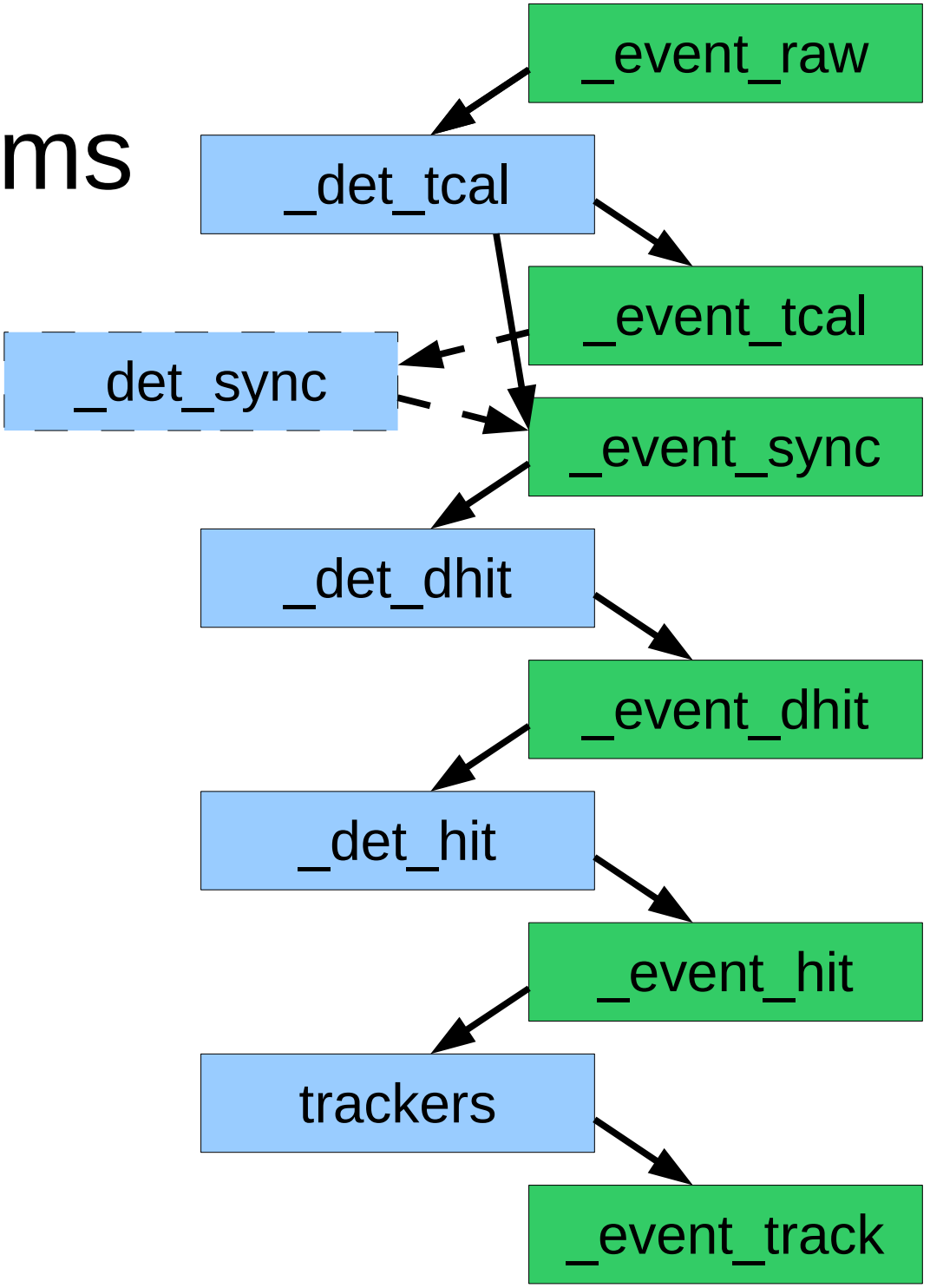
where it hurts the least

deliver 'least-surprise' data even for
non- & half-calibrated detectors

levels of data - levels of calib params

Event-wise data is held in the (zero-suppressed) event data structures.

Calibration parameters are kept in the reconstruction routines' data structures.



Calibration parameters

TIME_CALIB

// Comments!

LT_RANGE - time varying

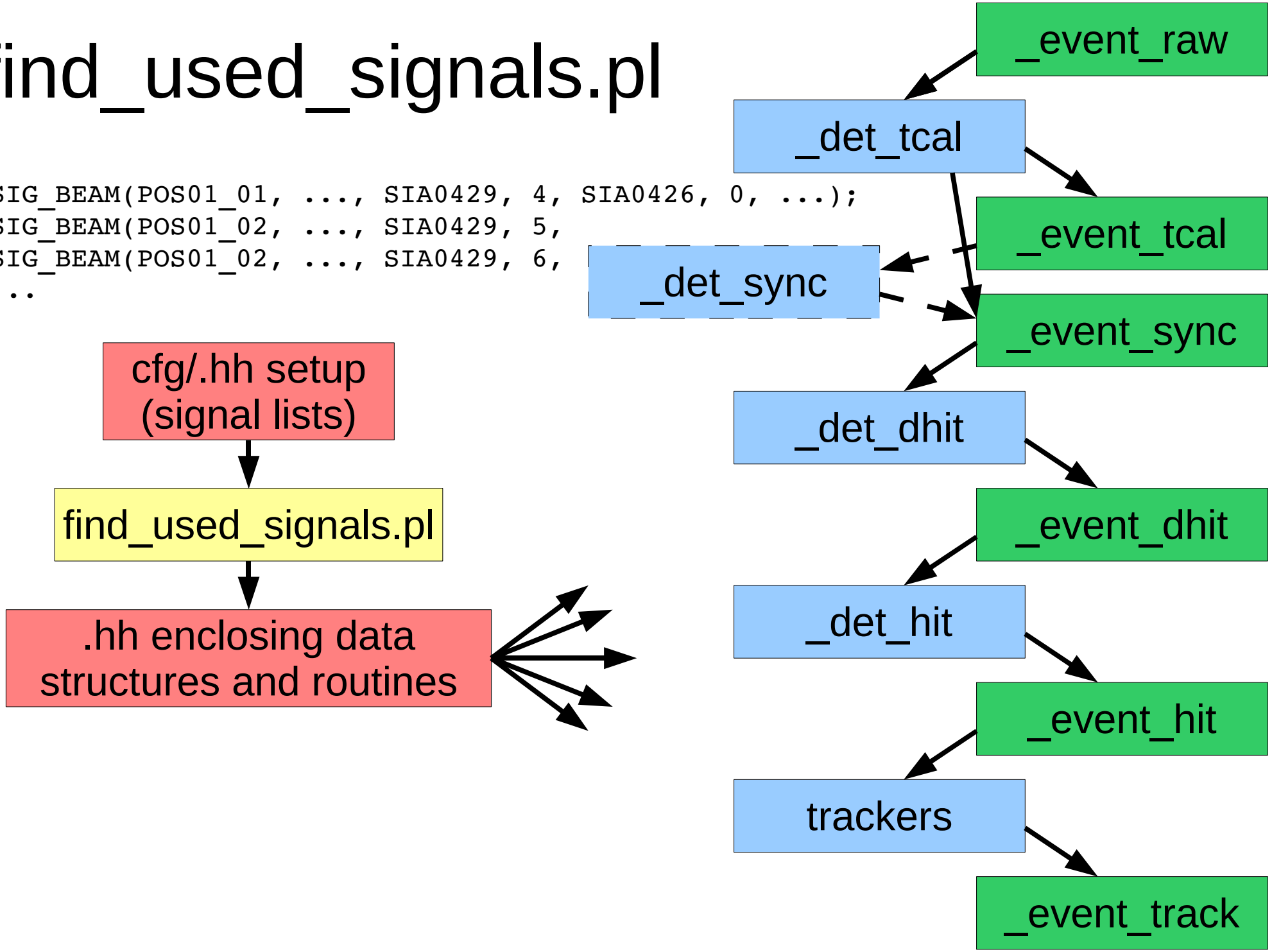
SIGNAL_ID

TIME_DIFF_OFFSET

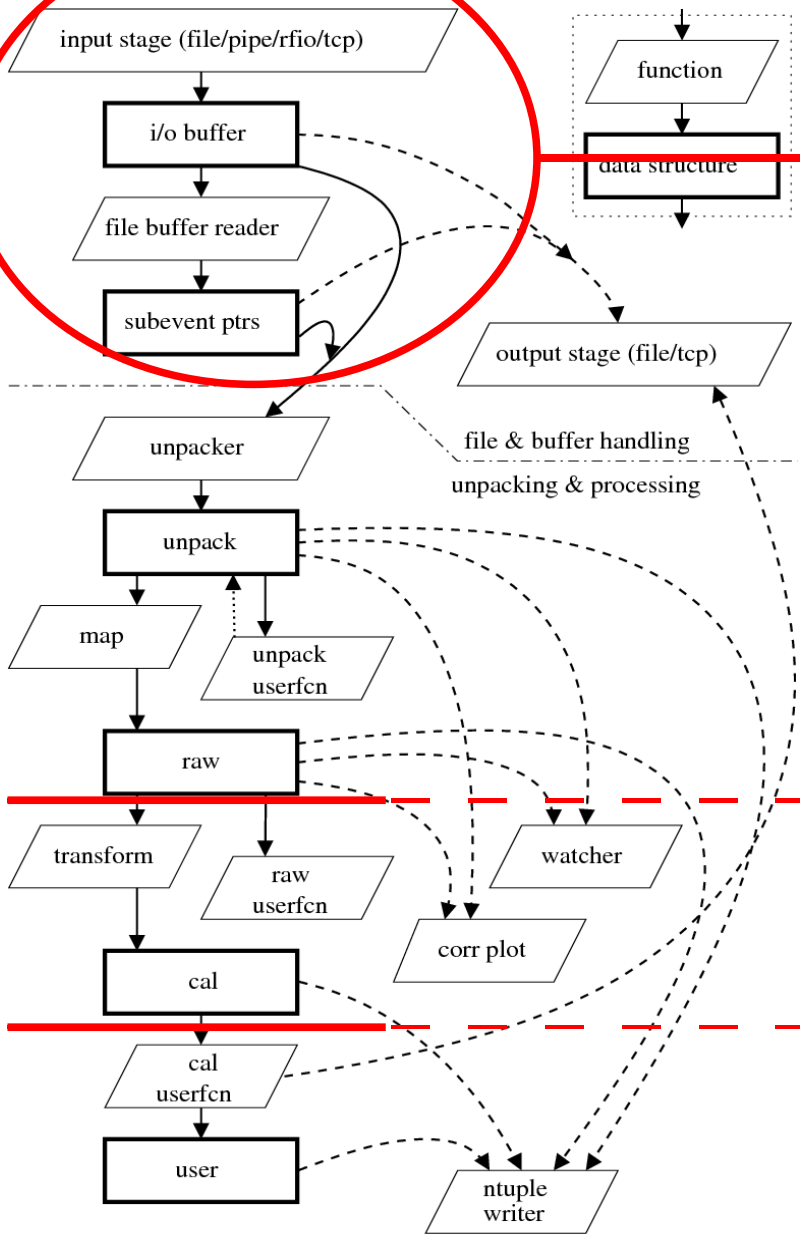
ENERGY_SYNC_GAIN

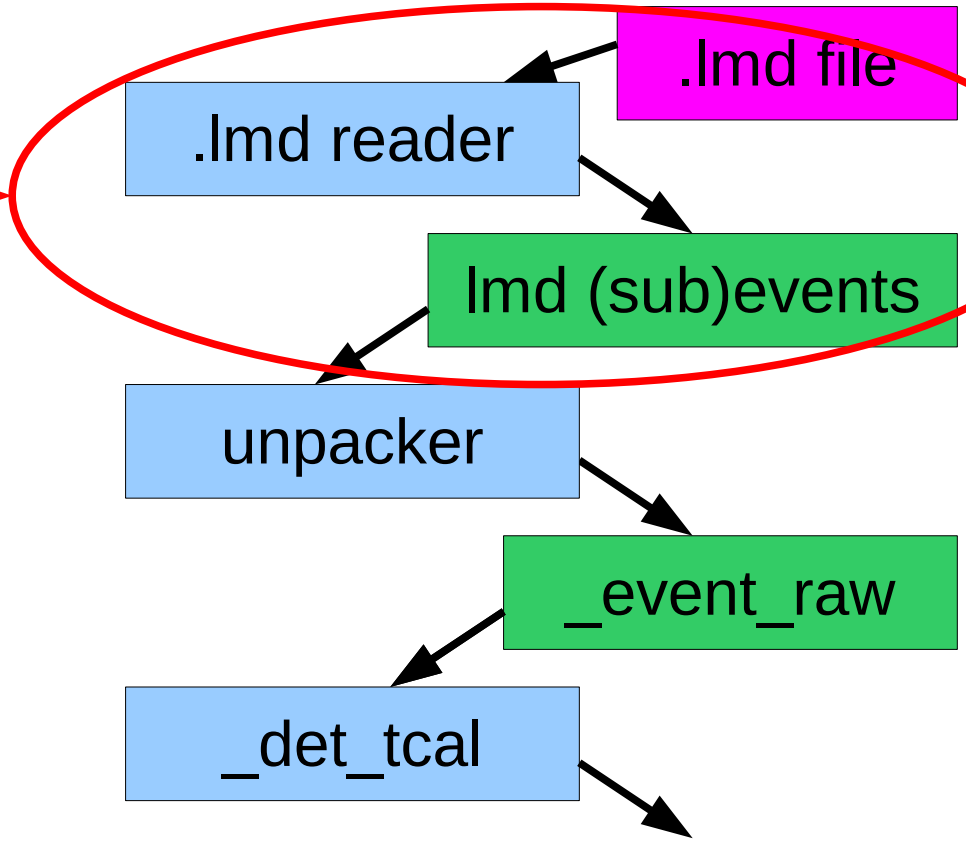
find_used_signals.pl

```
SIG_BEAM(POS01_01, ..., SIA0429, 4, SIA0426, 0, ...);
SIG_BEAM(POS01_02, ..., SIA0429, 5,
SIG_BEAM(POS01_02, ..., SIA0429, 6,
...
```



UCESB – unpack & check every single bit





R^{3}Broot (FAIRroot?)

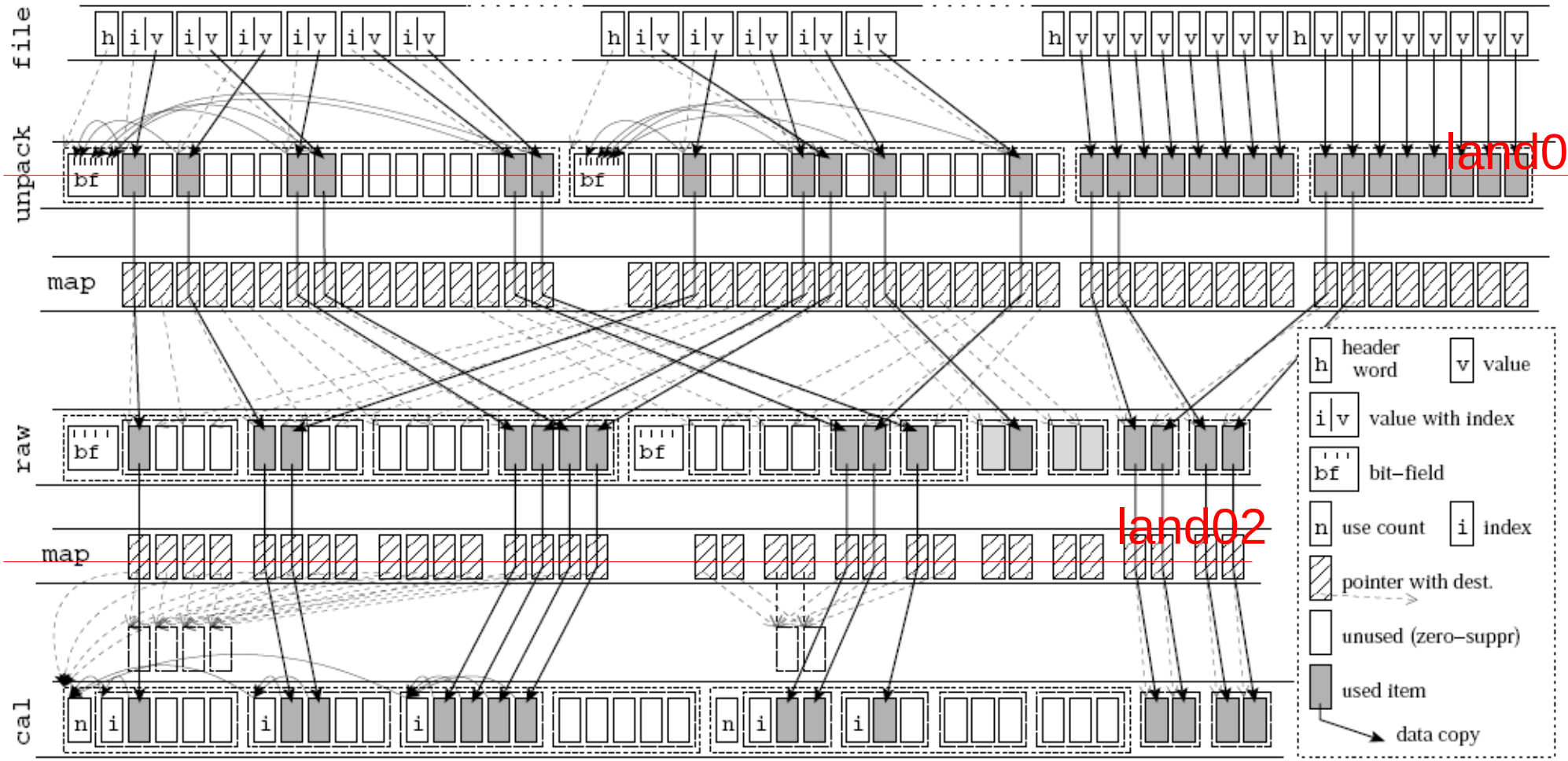
R^{3}Broot? (TCAL/SYNC)

http://fy.chalmers.se/~f96hajo/ucesb/

Data structures (UCESB)

"Show me your code and conceal your data structures, and I shall continue to be mystified. Show me your data structures, and I won't usually need your code; it'll be obvious."

Eric Raymond



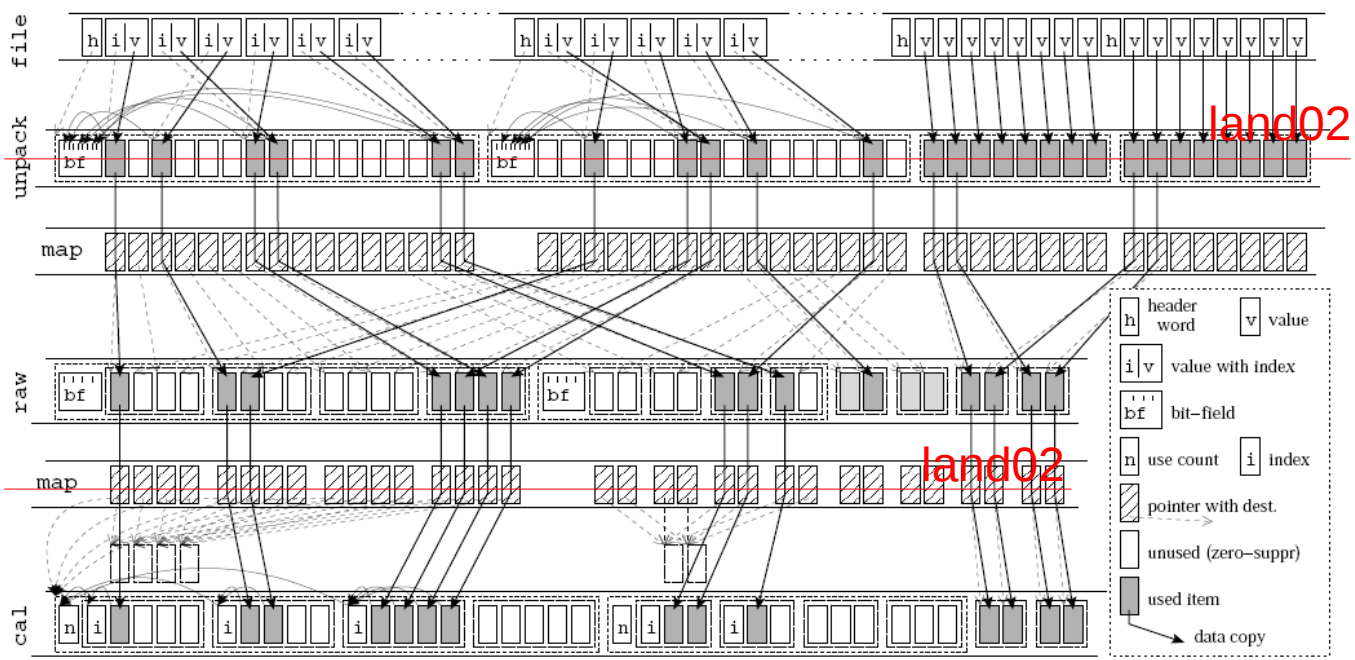
Data structures!

Organised to make the method implementation 'easy' (= by definition make certain things trivial, e.g. neighbour search).

At RAW level: arranged as arrays, bitmasks to allow easy random insertation (by the unpacking), and by design sorted.

At TCAL/SYNC level: arranged as lists (insertation is in order). Keeps sortings.

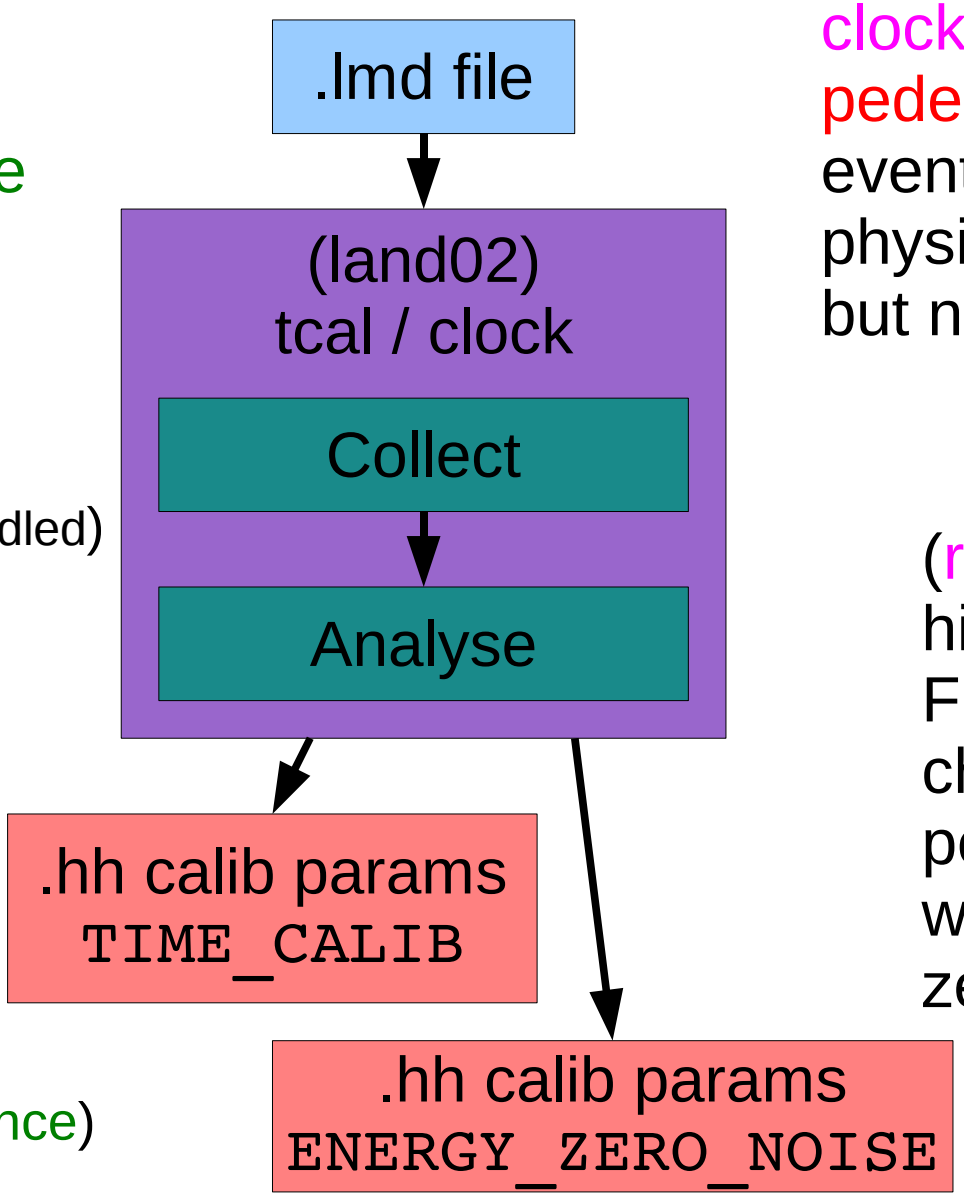
Templates to keep code generic, and let the **compiler** do the job!



tcal, clock, (range)

tcal: determine tdc gains from events with time calibrator data

clock: determine pedestals from events with no physics data, but normal gates



Four regimes:

- NIM tcal module
  (20 ns spacing, not handled)

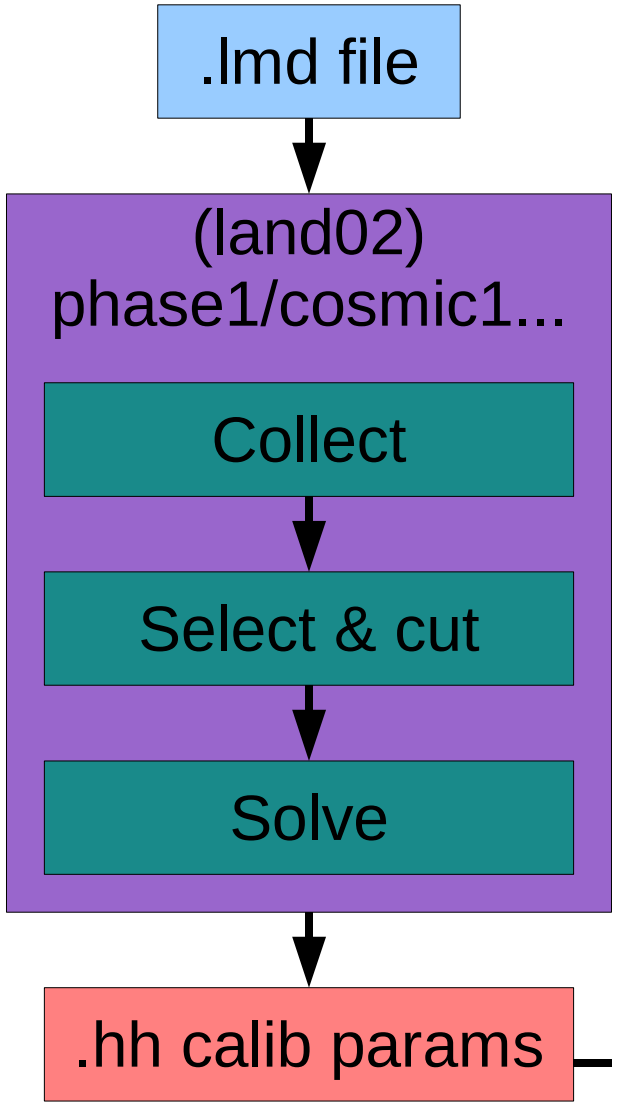
- CAMAC tcal module
  11 peaks, 10 ns

- CAMAC tcal module
  continuous samples,
  TDCt vs TCALt

- pulser+IOL+scaler+
  'clock' TDC (ntp-reference)
  (needs adaptions)

(range: try to adapt high/low range. Find noise level of channels without pedestals and working zero-suppression)

cosmic1(,2), phase1,2, phase1_gfi



Generally:

xxx1: bootstrap calibration, using LSQR to solve synchronisation problems

xxx2: handle semi-broken paddles

xxx3: improvements / more advanced models with Millepede II ?

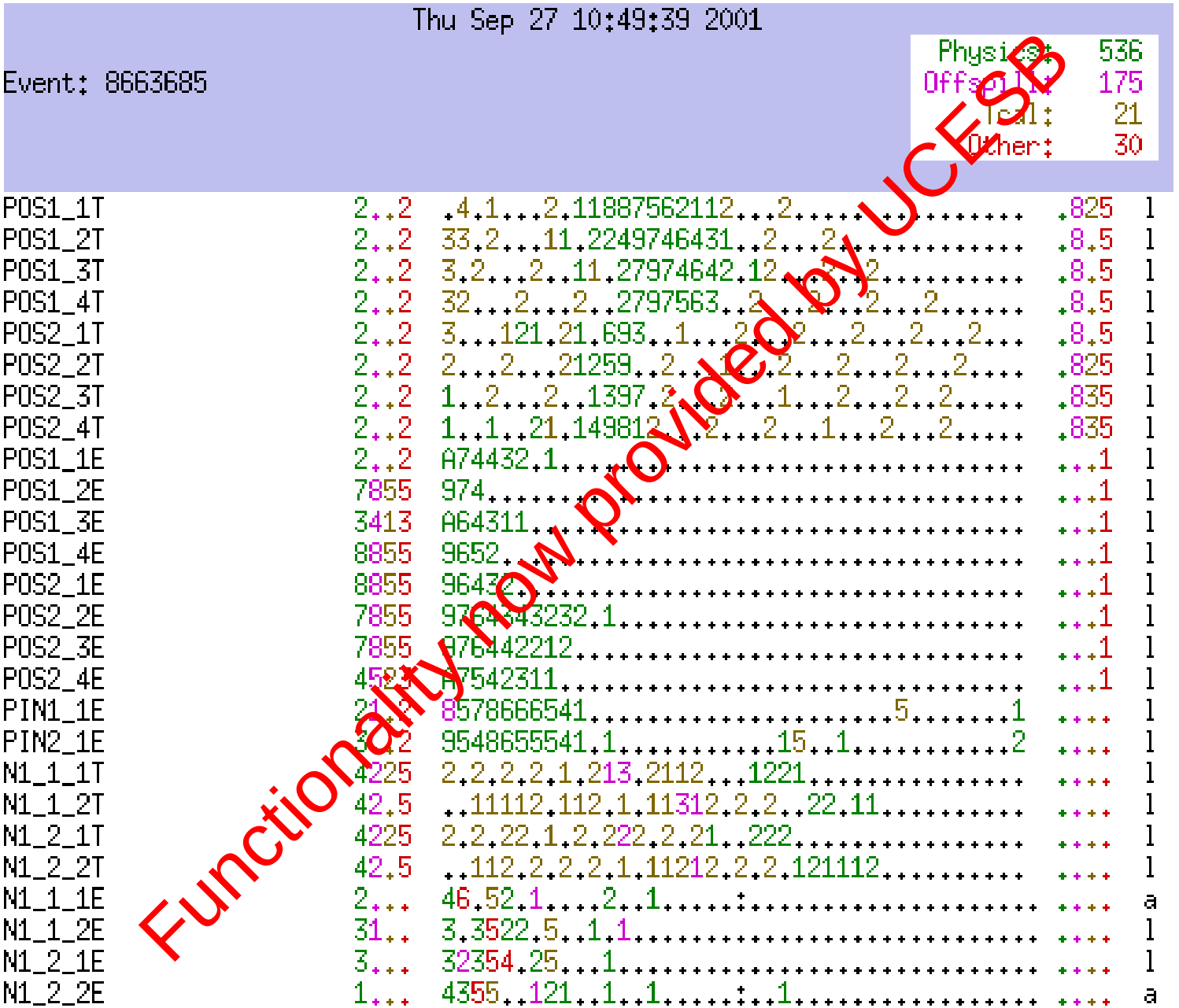
Needed: Handle time-varying parameters, and intrinsically floating calibration output:

showscaler

```
// ./showscaler /misc/scratch.land1/s287/lmd/run_068_1682.lmd \
// --scalerdiff=Clock,Proton,RAWTRIG10,COUNTER30
#EVENTNO   TRIG            Clock         TrigOR1          Proton          FRS S8
  67237163     2            62914               0               7             0
  67237183     2            71487               0              18             0
  67237203     2            65007               0              12             0
  67237216    12            35991               0               7             0
  67237236     1           289385              19              46          1292
  67237256     1            61236              20              20          1045
  67237276     1            53928              20              23          1177
```

```
// ./showscaler /misc/scratch.land1/s287/lmd/run_068_1682.lmd
SUMMARY IN SPILL:
=========================================
COUNTER01:                  R        6420
COUNTER02:                  O        6790
COUNTER03:                  L        4493
COUNTER04:                  U        4030
COUNTER05:               Pos1     2014149
=========================================
TPAT01=0x0001:       Min Bias    2001428    1535773      47993  23.3%  32.0    5
TPAT02=0x0002:       Fragment          0          0          0      -          -
TPAT03=0x0004:         GB+LND      68692      52286      52269  23.9%   1.0    0
TPAT04=0x0008:         GB+CsI      38055      29159      29156  23.4%   1.0    0
TPAT05=0x0010:           GB+P      46146      31735      31730  31.2%   1.0    0
TPAT06=0x0020:           Misc          0          0          0      -          -
```

watcher – the DAQscope



Each line is a histogram for one raw channel

Values are $\log_2$ of bin content

Stored zeros and overflow

Colour by most contributing trigger type

Spill synchronized

gamma1

.lmd file

(land02) gamma1

.root / .ntu
- Single crystal histograms
- Anti-Compton 'shielded'

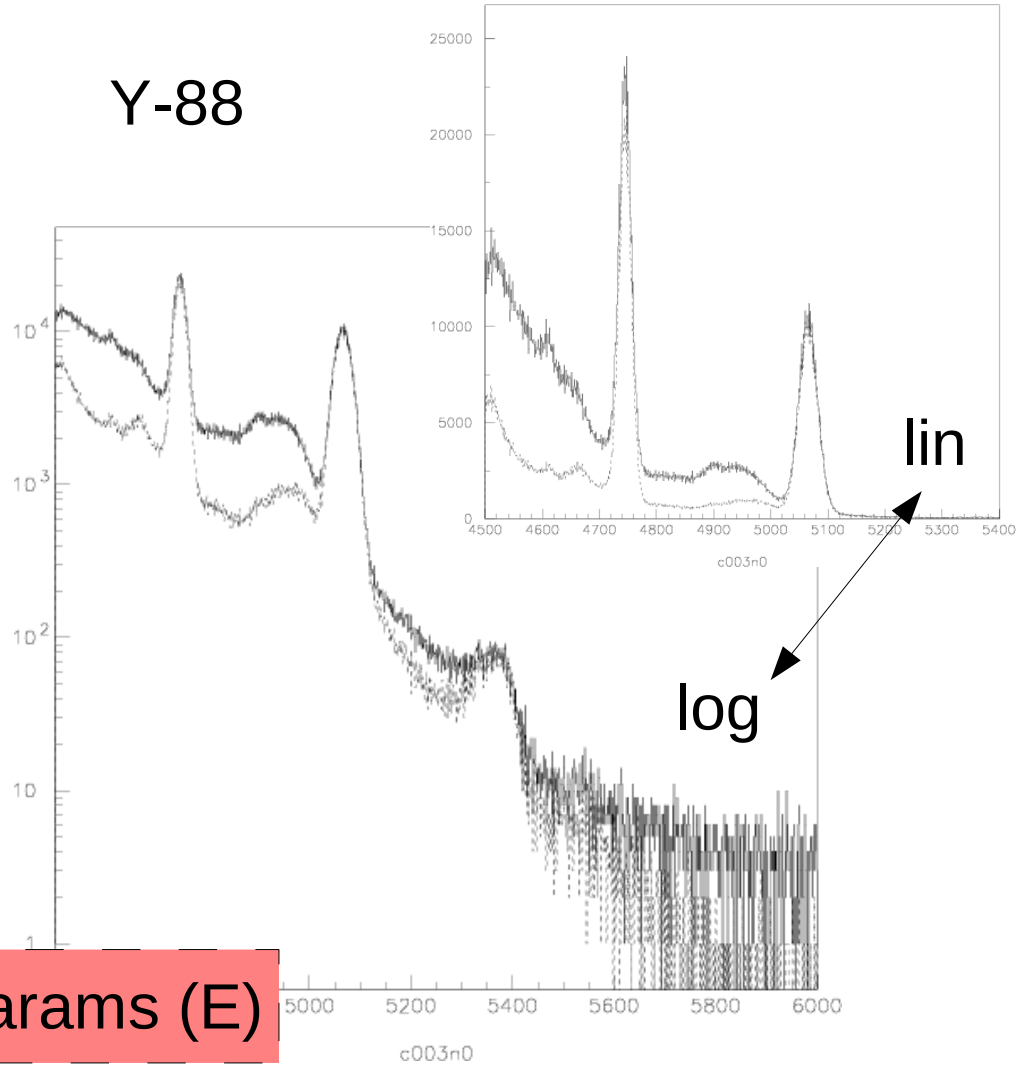
.root / .ntu
- e1e2-corr for dual read-out

.root / .ntu
- Tree with neighbour-pairs for direct compton

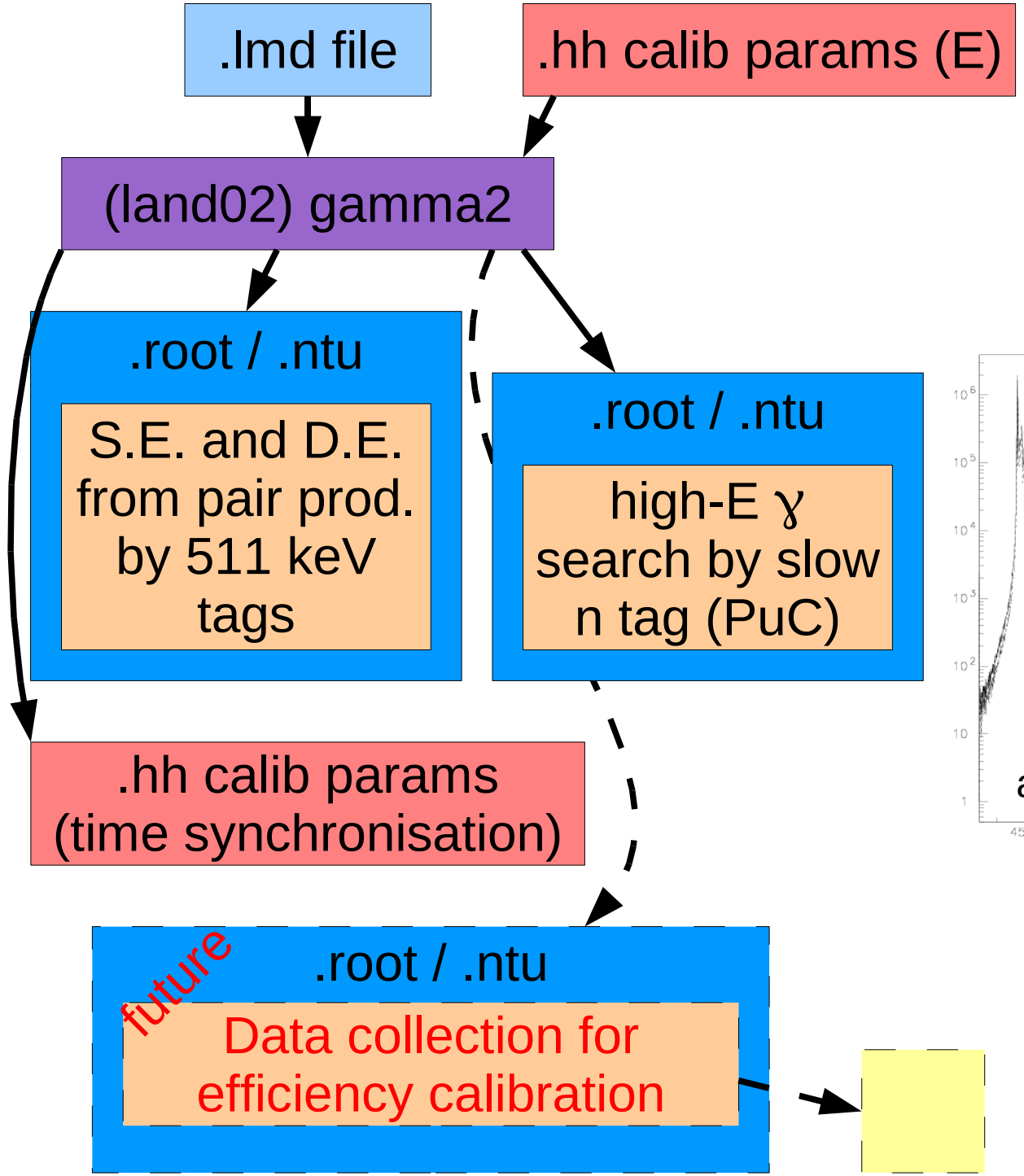
External scripts / root macros (peak fit)

.hh calib params (E)

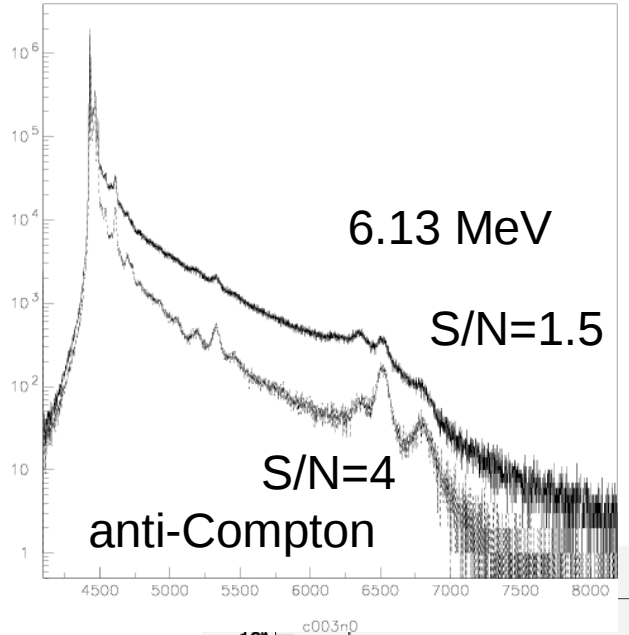
Y-88



gamma2

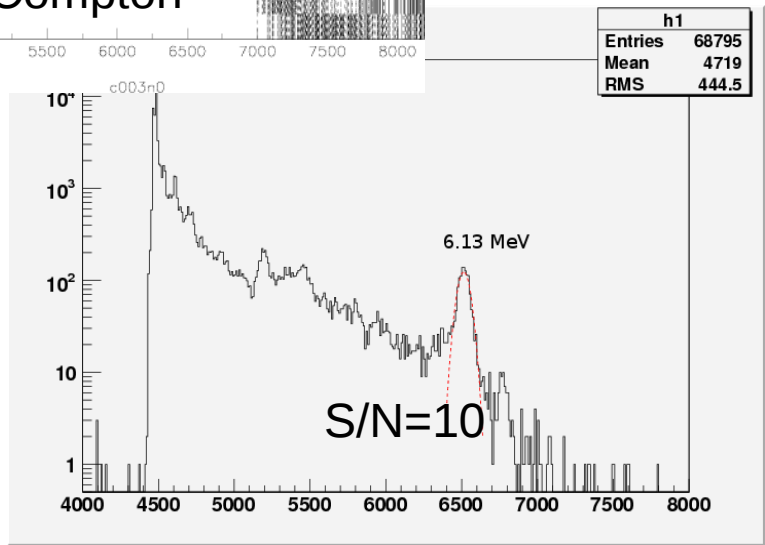


PuC source
$^{13}C(\alpha,n)^{16}O^*$
6.13 MeV line, neutrons



152 GB
raw data
$730*10^6$ ev.

land02:
20MB/s =
70GB/h

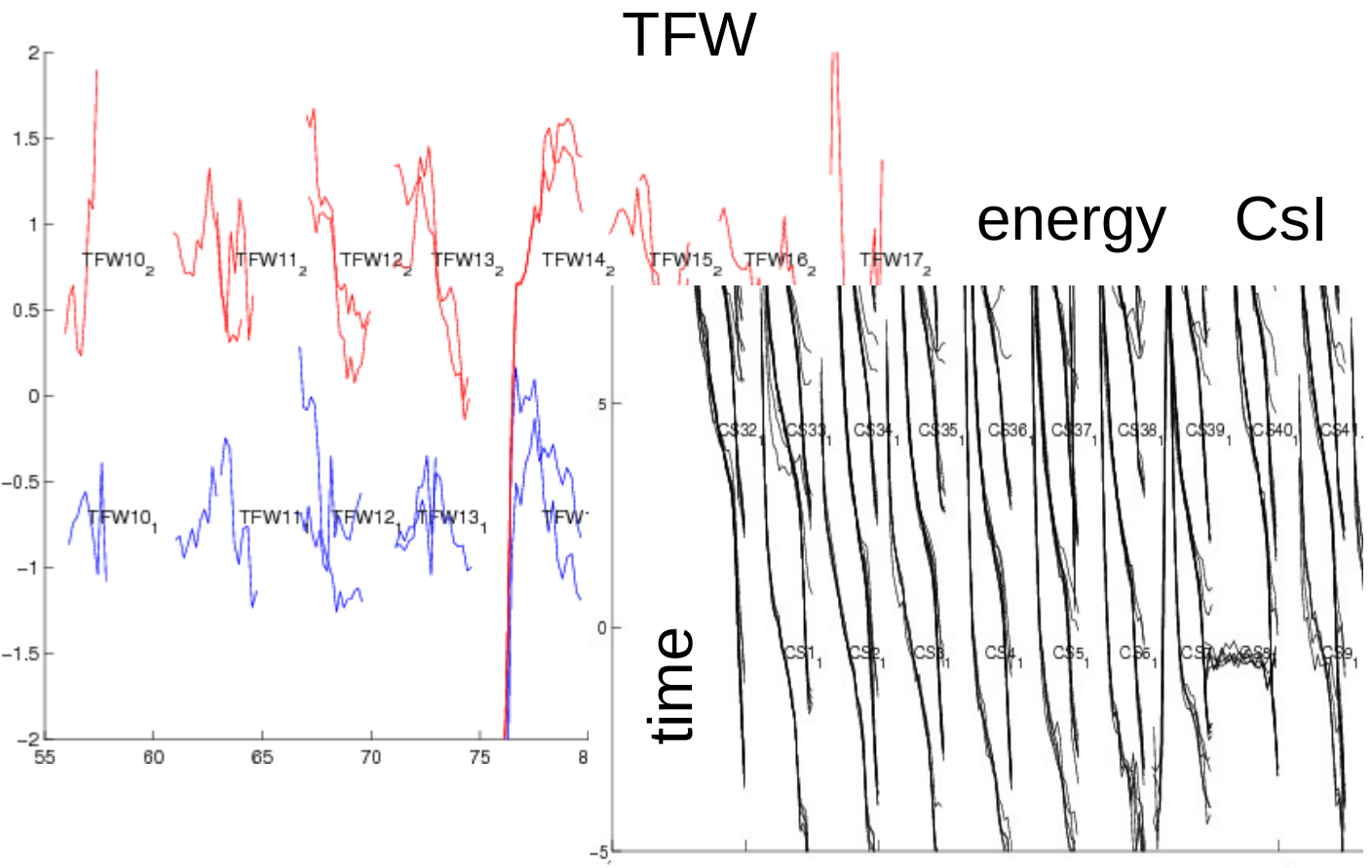


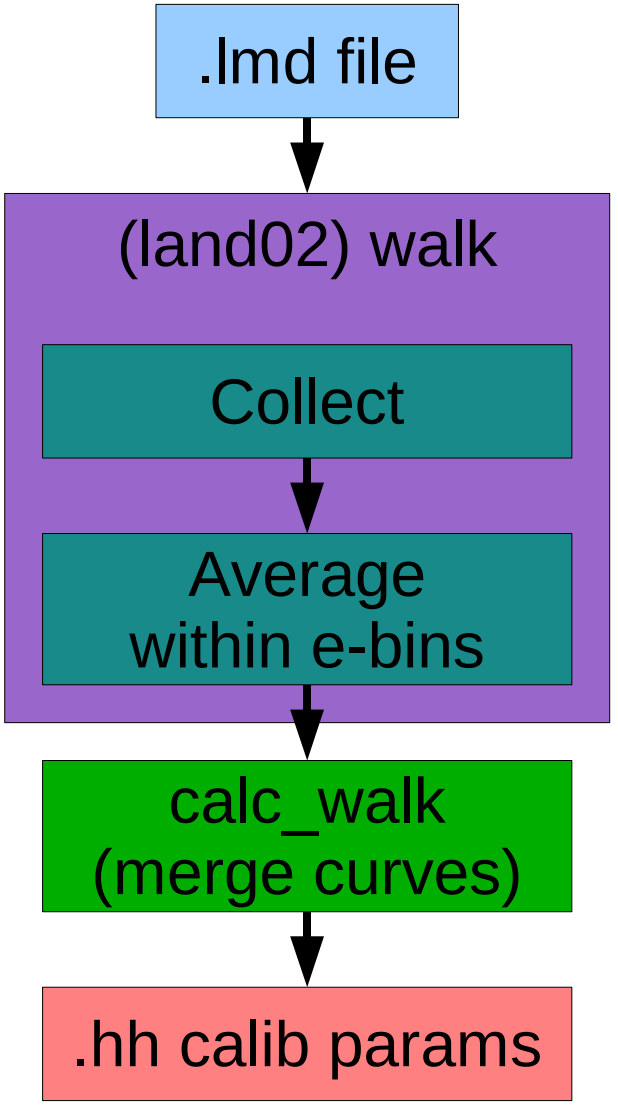
walk

$t_{real} = t_{measured} + f(e)$

$t_{real,1} = t_{real,2}$

$(t_{m,1} + f_1(e_1)) - (t_{m,2} + f_2(e_2)) = 0$



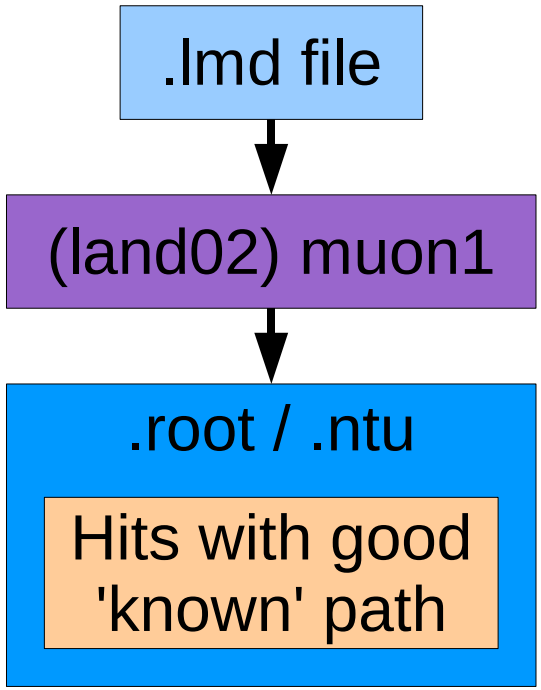


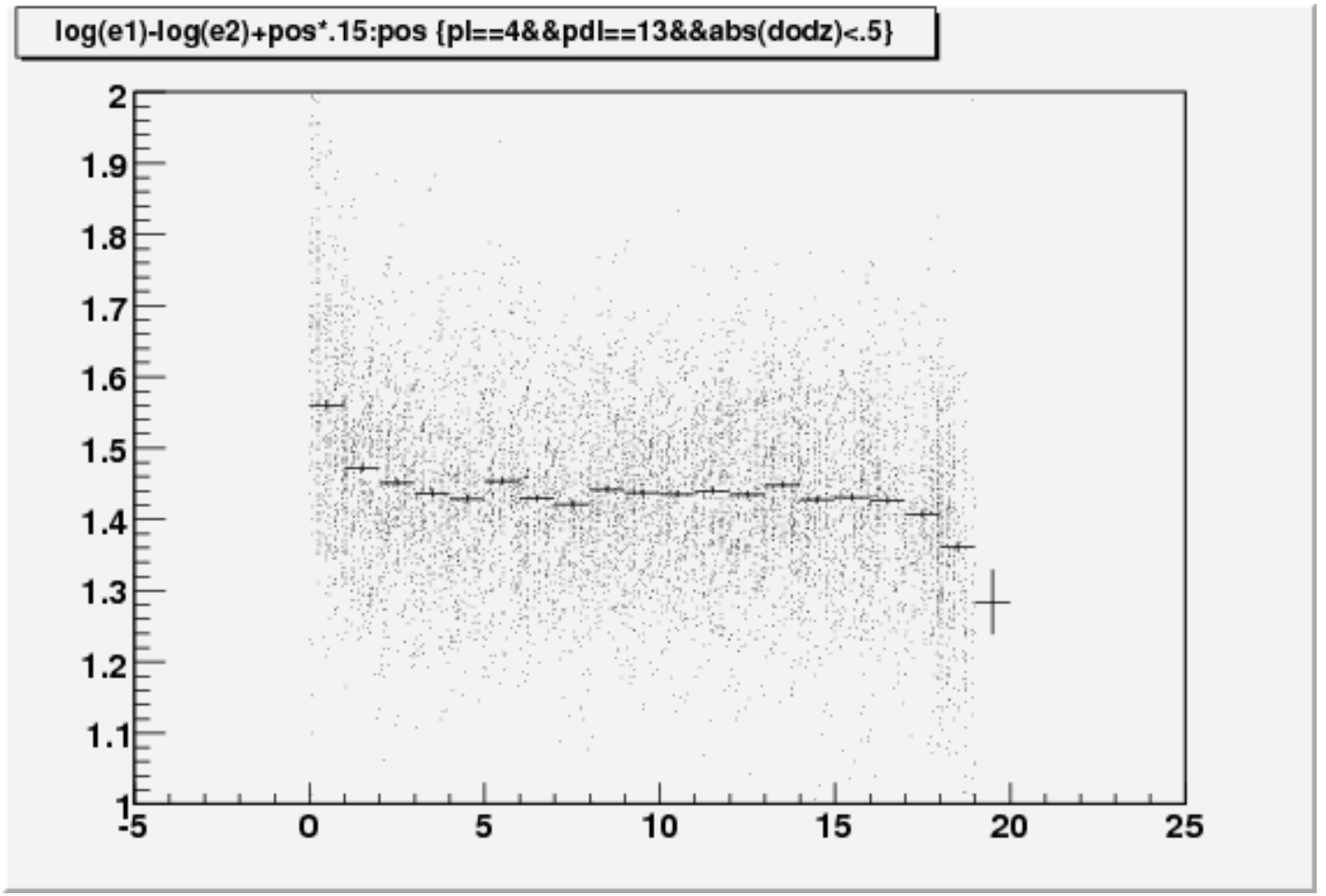
muon1

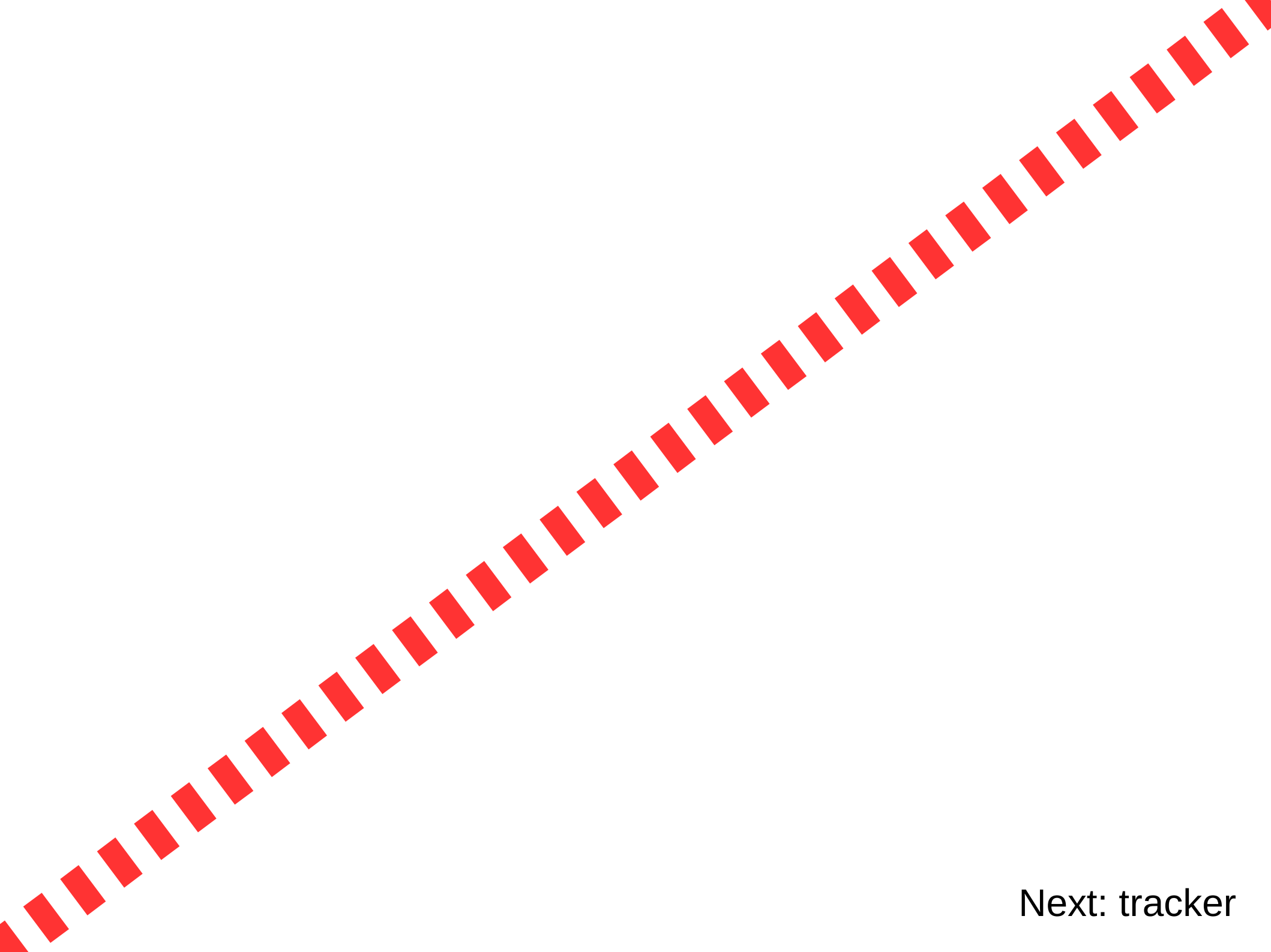
Find instances of 'clean' muon hits in paddles. Clean = well-defined path through the paddle.

Allows absolute calibration of the energy loss.

Search for non-linear behaviour (smiley).

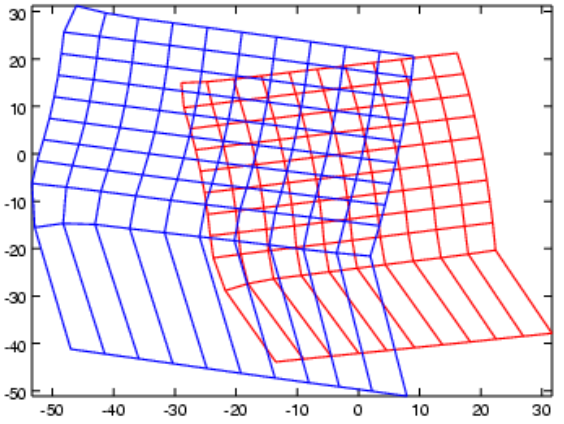






Next: tracker

Tracker



Multi-hit

Multi-track

Generic
(exp't indep)

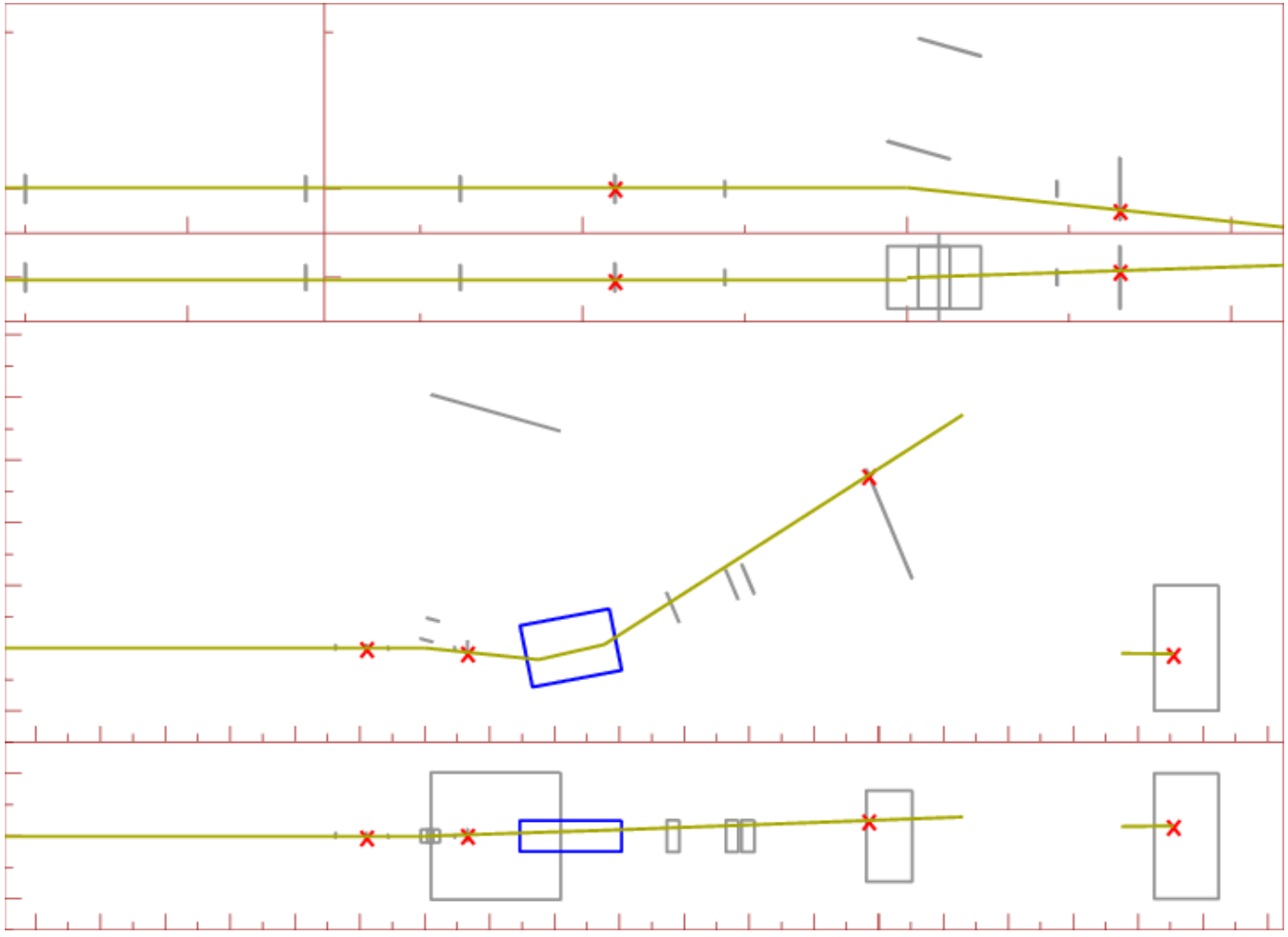
Field map

Energy loss (ATIMA)

Runge-Kutta

CPU-intensive
(interpolation)

Most likely track

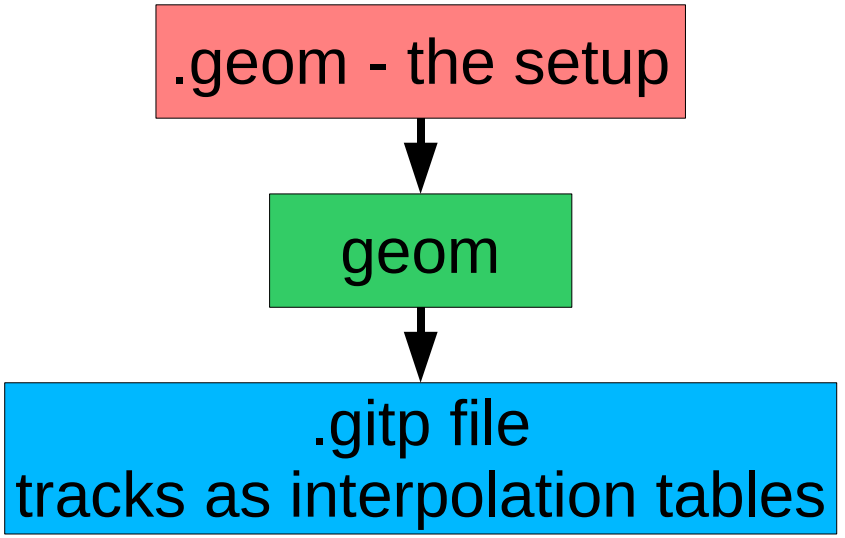


Specifying the setup geometry

Detector types are declared:

as planes separating materials

with control data (size, resolutions)
to control the interpolation granularity




```
DETECTOR(GFI)
{
  HALF_WIDTH(25.5 cm,25.5 cm);

  START_ACTIVE;
  MATERIAL(scintillator,0.1 cm);
  STOP_ACTIVE;

  RESOLUTION(x = 0.1 cm);
  RESOLUTION(y = 1.0 cm);  // to avoid interpolation to be picky
  RESOLUTION(t = 0.1 ns);
  RESOLUTION(E = 0.1 MeV); // ??? (lnE ??)
}
```

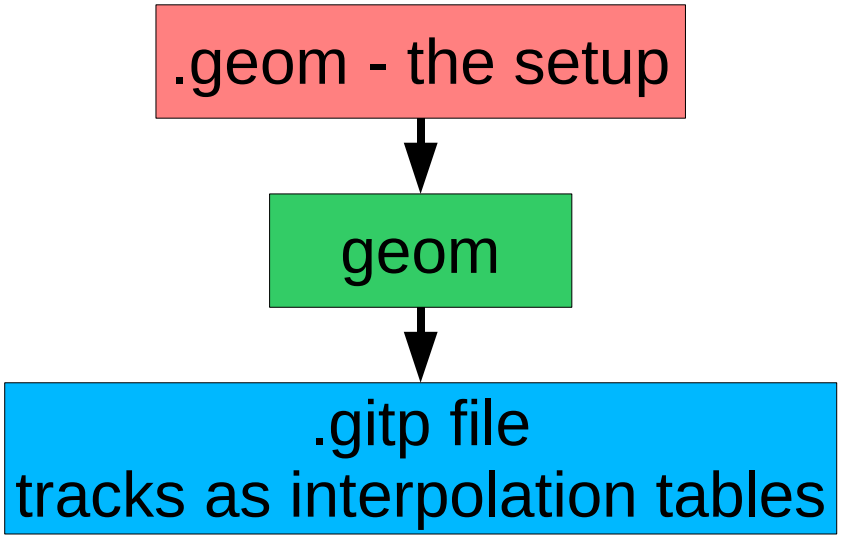
Specifying the setup geometry

Detector instances are used within coordinate systems

```
COORD_SYSTEM(cs_to_fragments)
{
  ROTATE(vector_y,DEG2RAD(22. deg));
  TRANSLATE({225.4} * vector_z);
  UPSTREAM(cs_magnet);

  MATERIAL(air);

  DETECTOR(GFI1)
    {
      TYPE(GFI);
      OFFSET_Z(169.8 cm);
      OFFSET_XY(0.0 cm, 0.0 cm);
      CAPABILITY(TOD_USABLE_X | TOD_USABLE_T);
      INTERP_ORDER_bweakx;
    }
...
```



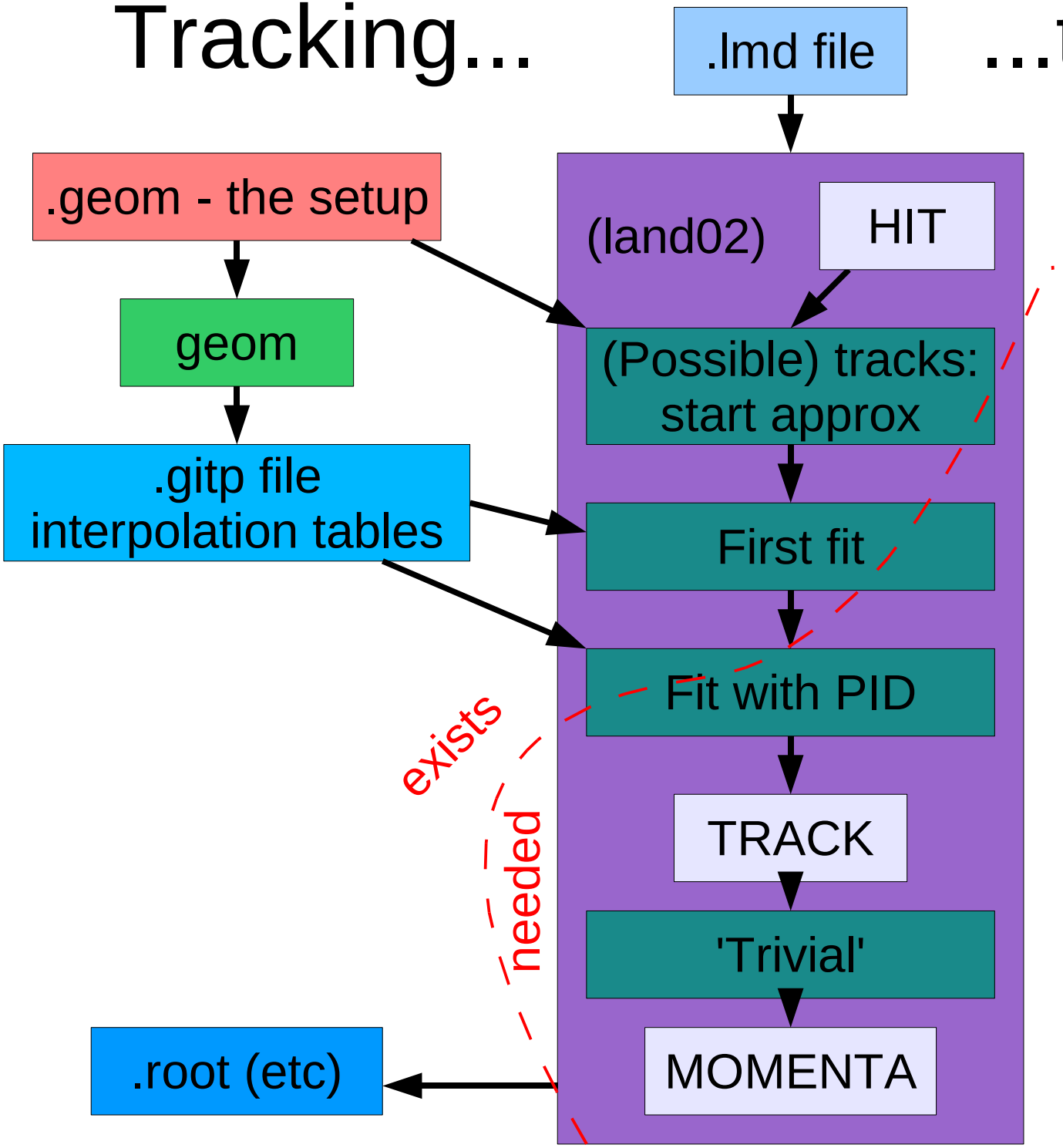
The coordinate system are located in the lab system,
and may have extra materials.

geom – interpolation tables

```
Segments:  3*3 3*3 9*3 -> 81*27*25=54675 , 10/322 params
 PIN2a:     0.00000    0.00000 (1: 6) (189/ 3/189) 50484
     b:     0.00000    0.00000 (1: 3) (189/ 3/189) 50484
     t:     0.00136    0.00041 (2: 6) (160/ 8/160) 50484
     e:     0.00812    0.00240 (2: 5) (160/ 8/160) 50484
 ZST2a:     0.00000    0.00000 (1: 5) (189/ 3/189) 50484
     b:     0.00000    0.00000 (1: 4) (189/ 3/189) 50484
     t:     0.00195    0.00058 (2: 6) (160/ 8/160) 50484
     e:     0.00022    0.00006 (2: 5) (160/ 8/160) 50484
 GFI1a:     1.09119    0.24276 (2: 7) (322/10/322) 50484
     b:     0.15824    0.02916 (2: 6) (322/10/322) 50484
     t:     0.16472    0.04261 (2: 6) (322/10/322) 50484
     e:     0.03038    0.00725 (2: 6) (322/10/322) 50484
 GFI2a:     2.01907    0.41493 (2: 7) (322/10/322) 50484
     b:     0.30486    0.05481 (2: 6) (322/10/322) 50484
     t:     0.23519    0.05804 (2: 6) (322/10/322) 50484
     e:     0.01562    0.00371 (2: 6) (322/10/322) 50484
 GFI3a:     2.27547    0.46530 (2: 7) (322/10/322) 50484
     b:     0.34481    0.06175 (2: 6) (322/10/322) 50484
     t:     0.25429    0.06225 (2: 6) (322/10/322) 50484
     e:     0.01580    0.00375 (2: 6) (322/10/322) 50484
  TFWa:     0.22839    0.04679 (2: 7) (322/10/322) 50484
     b:     0.35100    0.06206 (2: 6) (322/10/322) 50484
     t:     0.85631    0.20158 (2: 7) (322/10/322) 50484
     e:     0.01694    0.00400 (2: 6) (322/10/322) 50484
```

```
    3-He -    3.01493
----------------------------
    x: -1.50000 ..  1.50000
    y: -1.50000 ..  1.50000
    z:  0.00000 ..  0.00000
 dxdz: -0.10000 ..  0.10000
 dydz: -0.08647 ..  0.08647
 beta:  0.55000 ..  0.80000
   B0: -1.65000 .. -1.65000
----------------------------
Vary: dxdz, 2 others (dydz, beta) => 9
Segments:  1*5 -> 1*5*25=125 , 4/4 params
Segments:  2*4 -> 2*4*25=200 , 4/7 params
Segments:  3*3 -> 3*3*25=225 , 4/10 params
```

Tracking...



...to momenta

Detector reconstruction

float!

x_0, $^{dx}/_{dz}$, y_0, $^{dy}/_{dz}$, $^{dt}/_{dz}$, A, Z
with direct calc (linear fits)

Fit with 'good' e-loss
and track

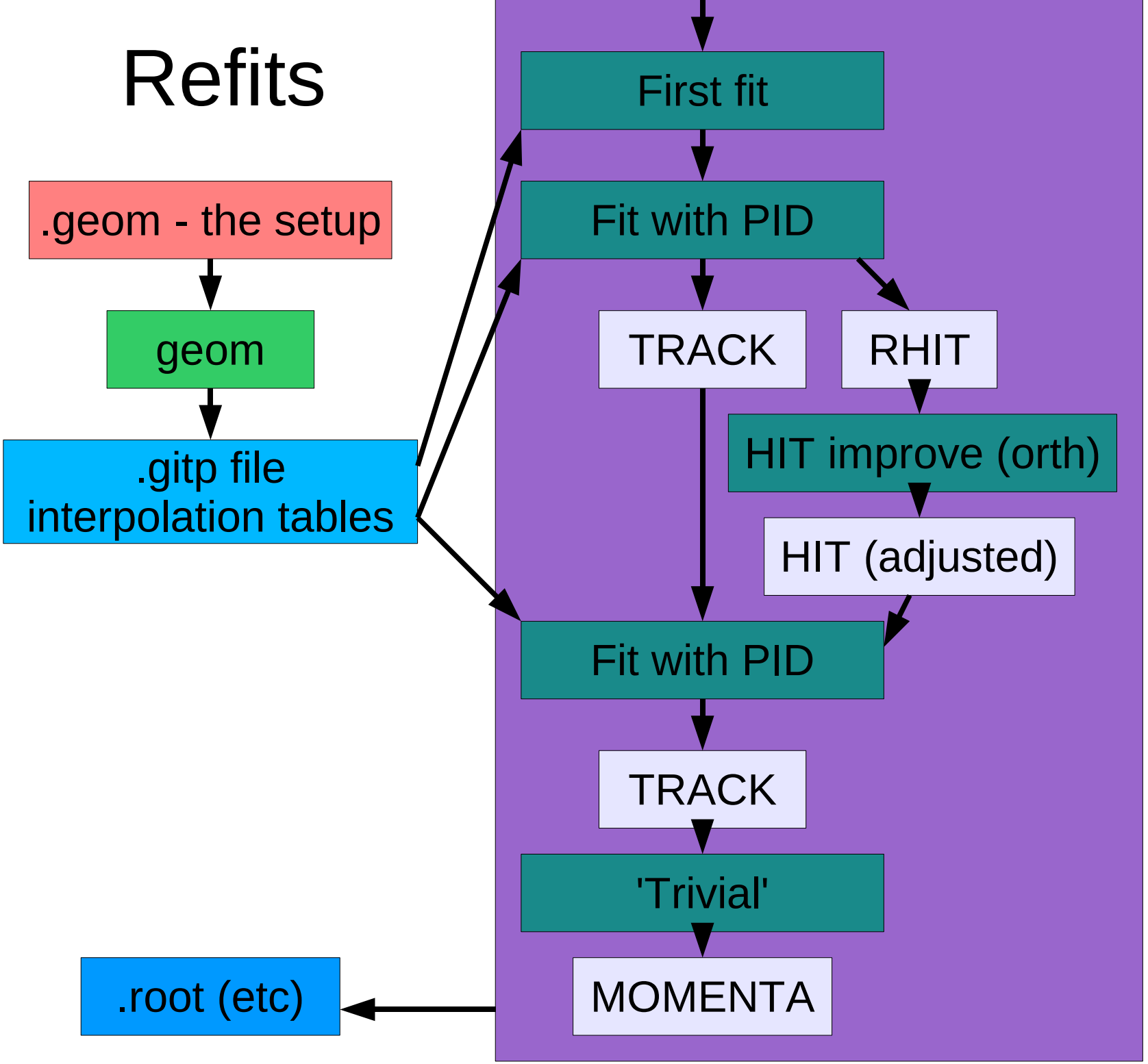
Fit with real A and Z

Parameters of the track

Done along with TRACK?

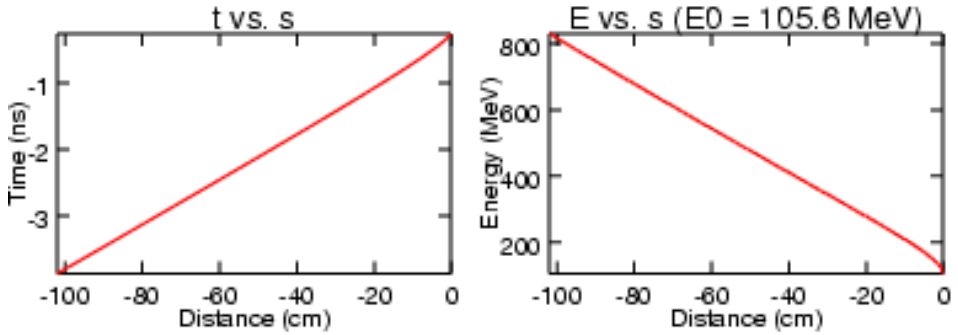
End of LAND02 mission

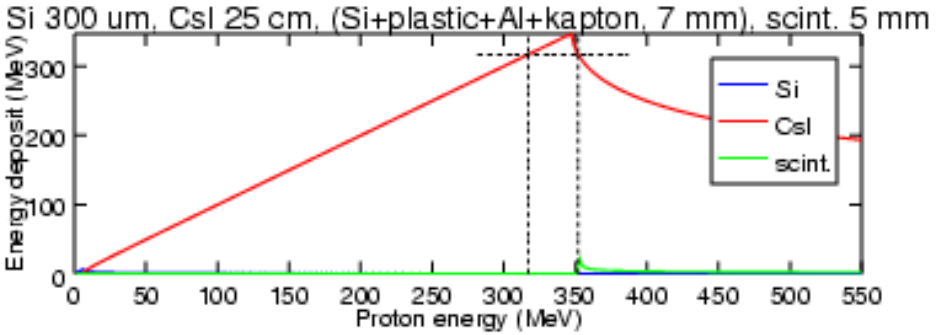
Refits

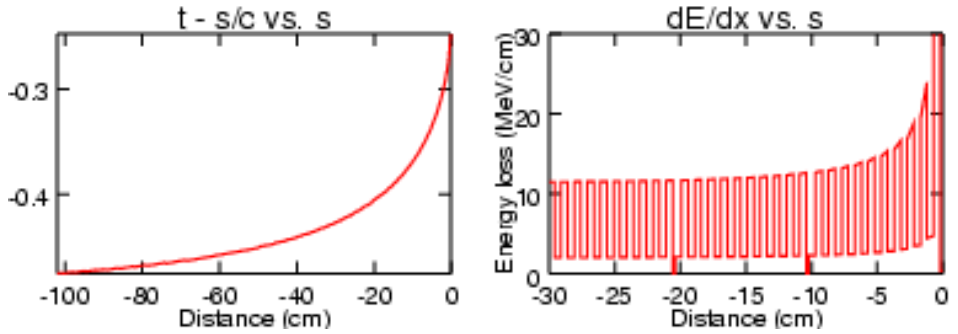

First fit
Fit with PID
.geom - the setup
geom
TRACK
RHIT
HIT improve (orth)
.gitp file
interpolation tables
HIT (adjusted)
Fit with PID
TRACK
'Trivial'
.root (etc)
MOMENTA

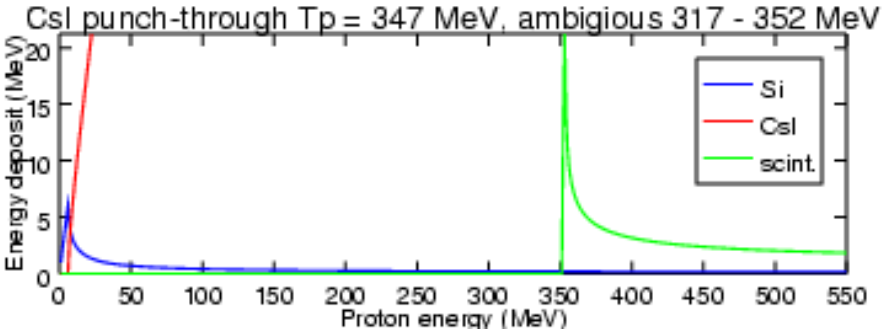
geom – command line tracks

```
./geom --geom=../s223/cfg/s223.geom --aladin \
--track=A=1,Z=1,T=500MeV/u,B0=-1900:20.:-1500T,cs_to_protons \
--show=B0,TFW:a,TFW:b,TFW:t
#     B0          TFW:a       TFW:b          TFW:t
#      (T)           (cm)        (cm)           (ns)
 -1900.00000   -13.31914     4.49640     46.84469
 -1880.00000   -10.20081     4.44766     46.84457
 -1860.00000    -7.08544     4.39818     46.84497
```





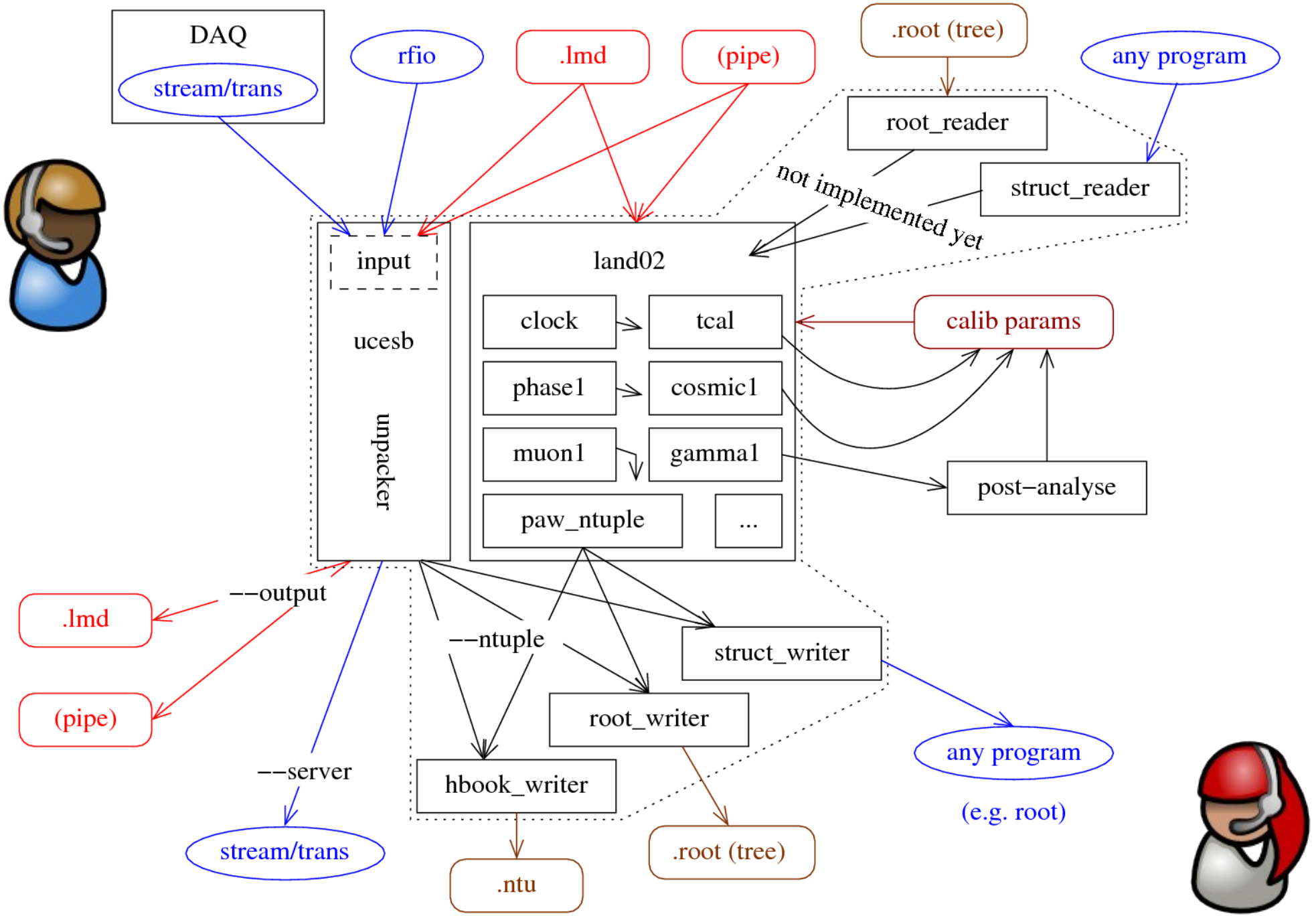


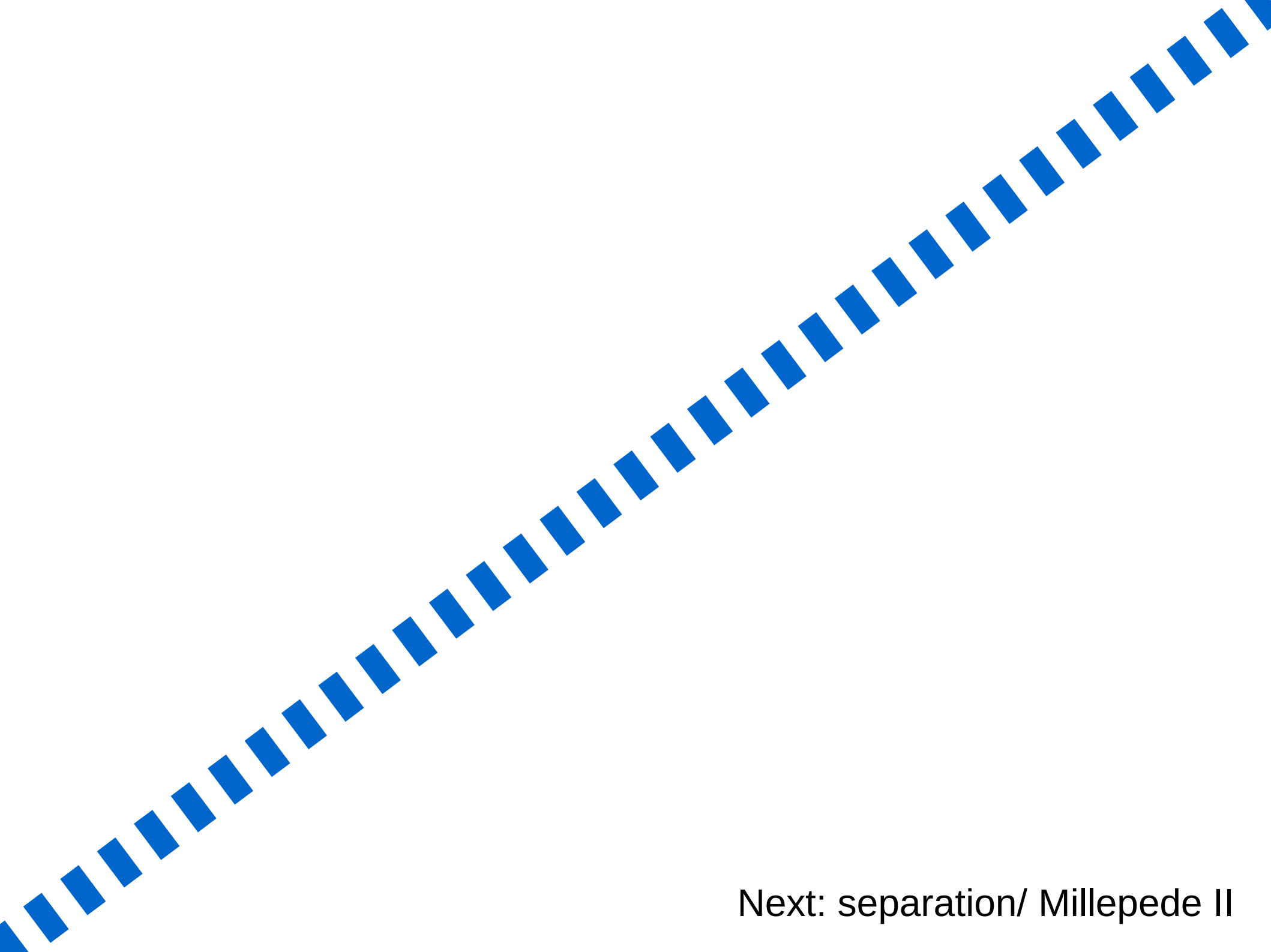


LAND02

Next: I/O connections

(ucesb)/land02 interaction (with outside world)

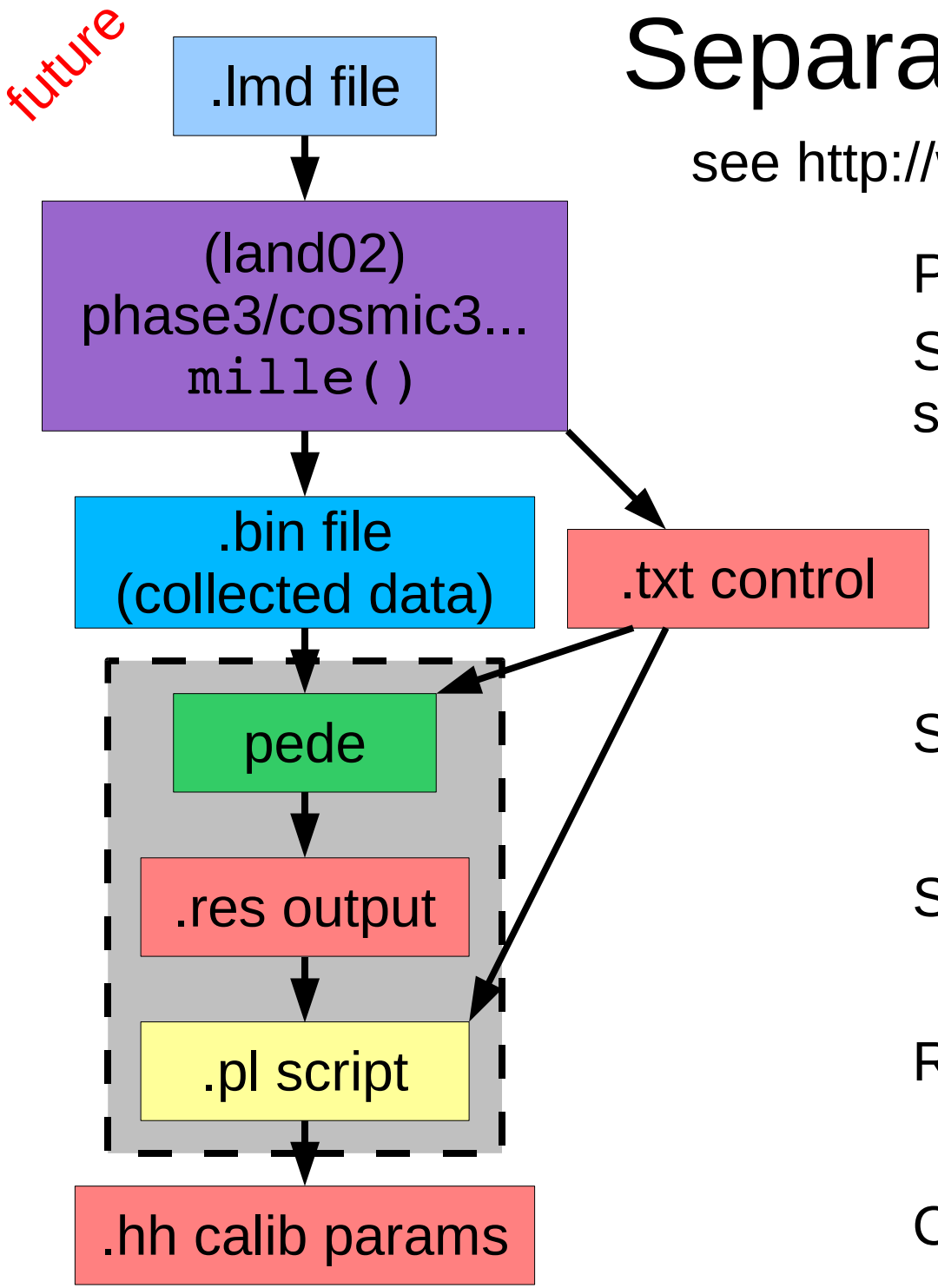




Next: separation/ Millepede II

Separation - Millepede II

see http://www.desy.se/~blobel/mptalks



Process input data and reconstruct

Select useful calibration events and store calibration data ('equations')

Stored equations / ancillary info

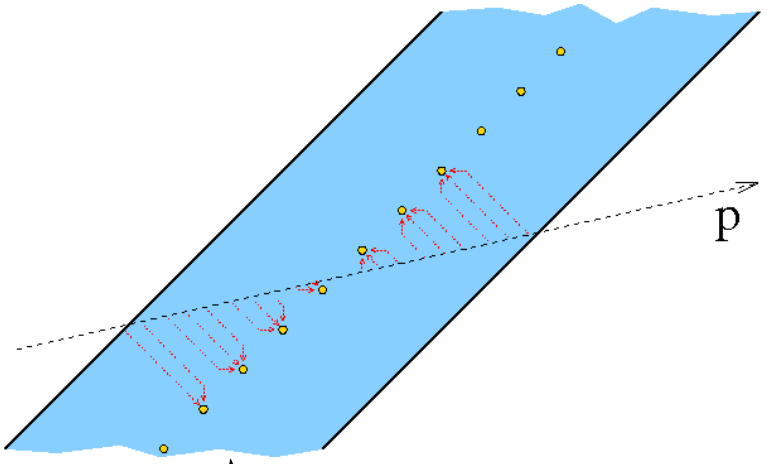
Solve equations

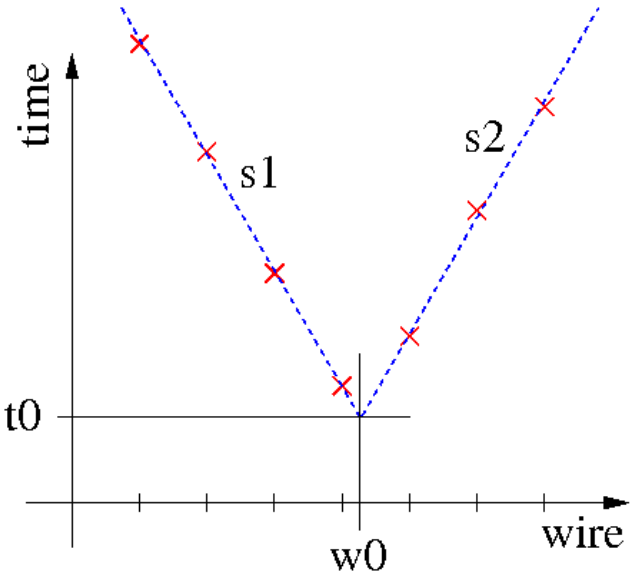
Solution

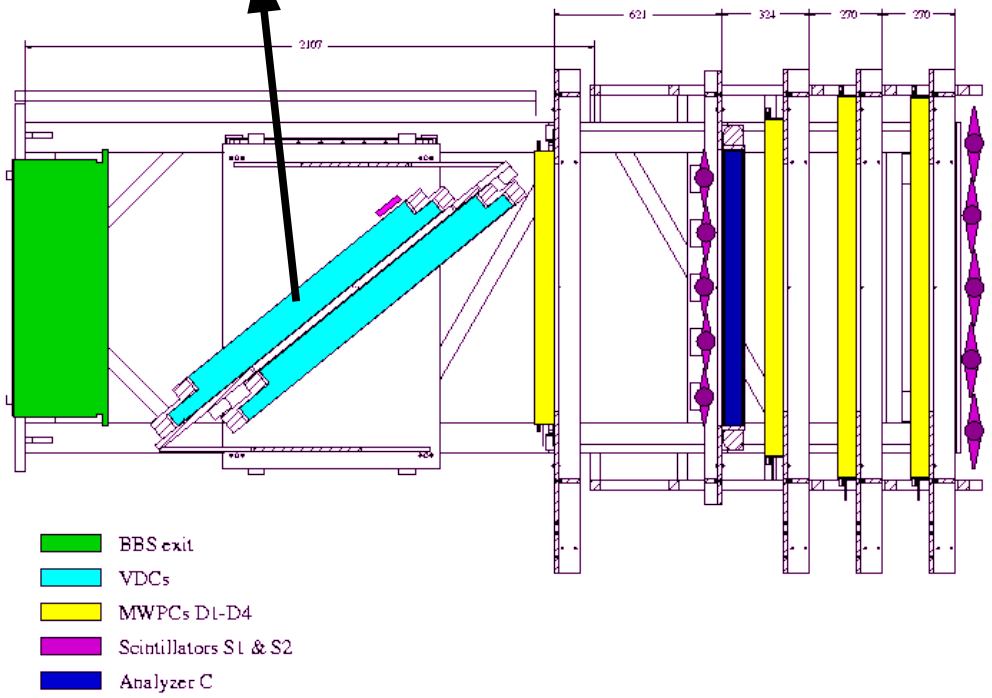
Reformat output to suit land02

Calibration parameters

EuroSuperNova (ESN) @ KVI
Millepede II – determining time offsets







t_0 = time of passage

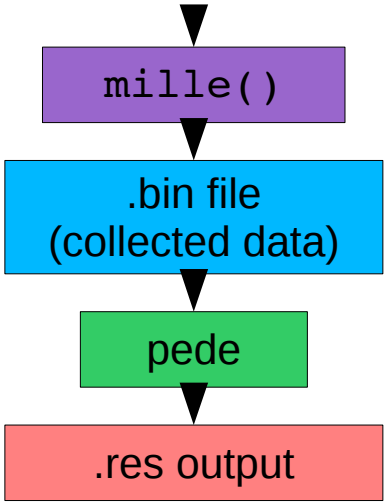
w_0 = position

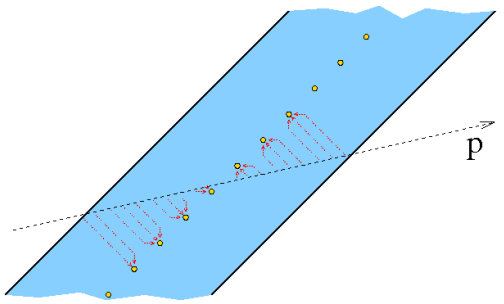
$s_{1,2}$ = slopes , depending on VDC drifttime, angle of incidence, proton speed

$$t_{i,e} = t_{0,e} + (w - w_{0,e}) * s_{n,e} + t_{off,i}$$

http://www.kvi.nl/~snovadoc/

Millepede II – determining time offsets





Given data for **many** tracks, pede solves for the global parameters, as if all tracks with **all** local and global parameters are fitted *simultaneously*

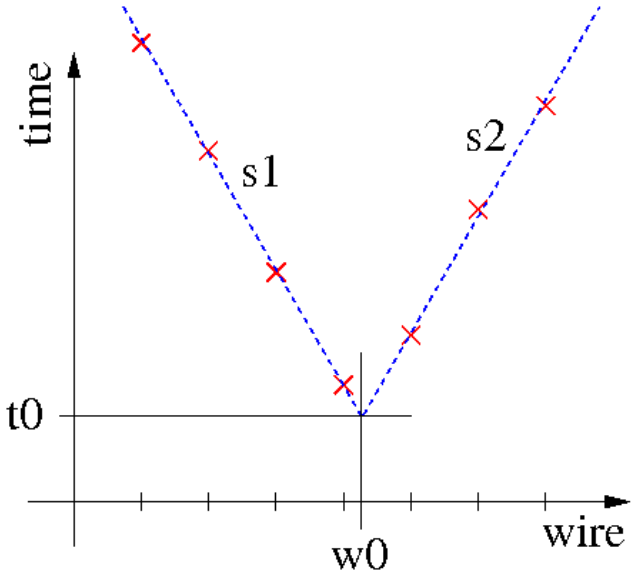
Input to `mille()` is the measured value (with sigma) and its derivatives w.r.t. the local and global params:

global

local (track)

measured

$$t_{i,e} = t_{0,e} + (w - w_{0,e}) * s_{n,e} + t_{off,i}$$



$$dt_{i,e}/dt_{0,e} = 1$$

$$dt_{i,e}/dw_{0,e} = s_{n,e}$$

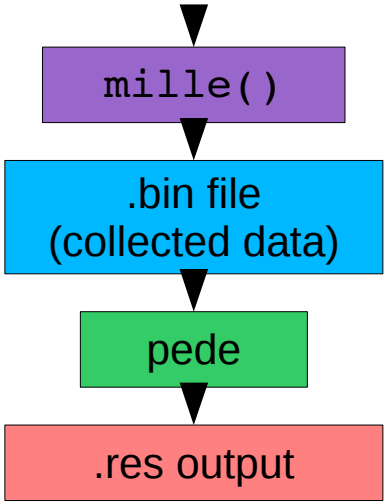
$$dt_{i,e}/ds_{n,e} = w - w_{0,e}$$

requires start approx

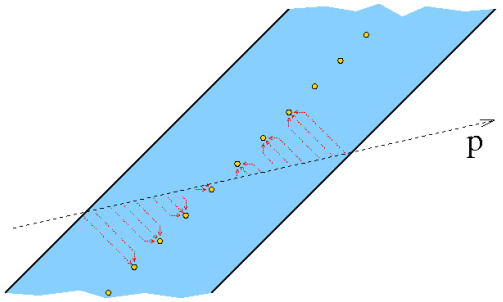
$$dt_{i,e}/dt_{off,i} = 1$$

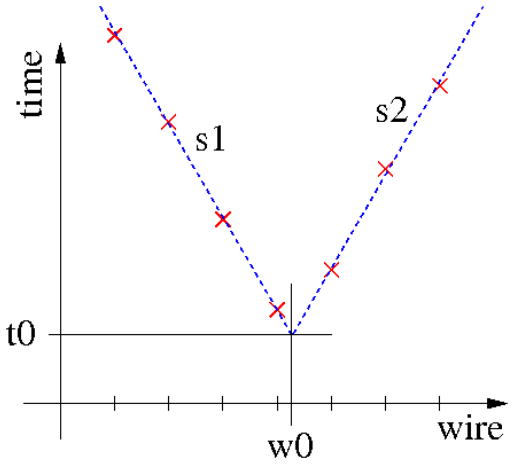
Abracadabra, Simsalabim ->

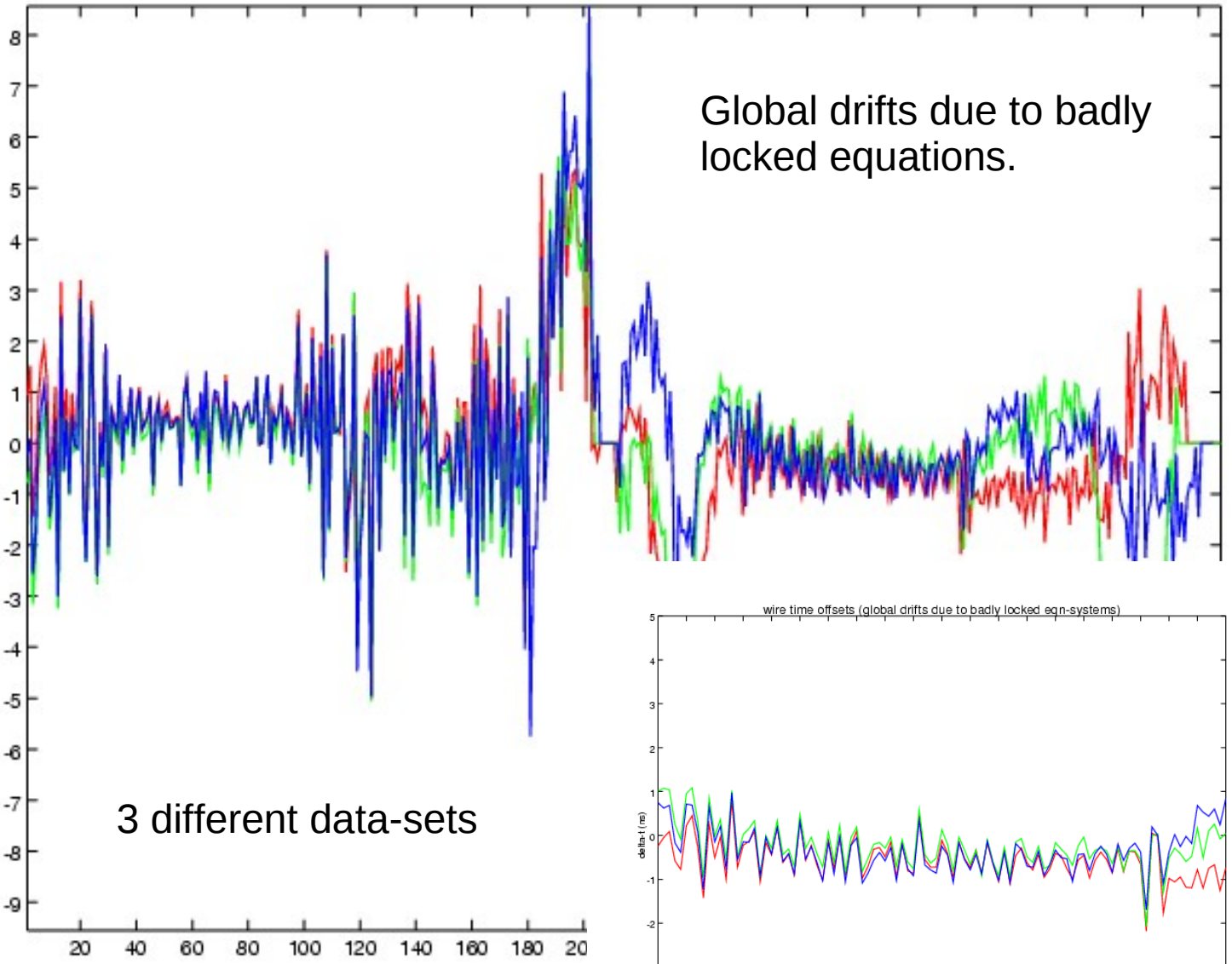
Millepede II – determined time offsets

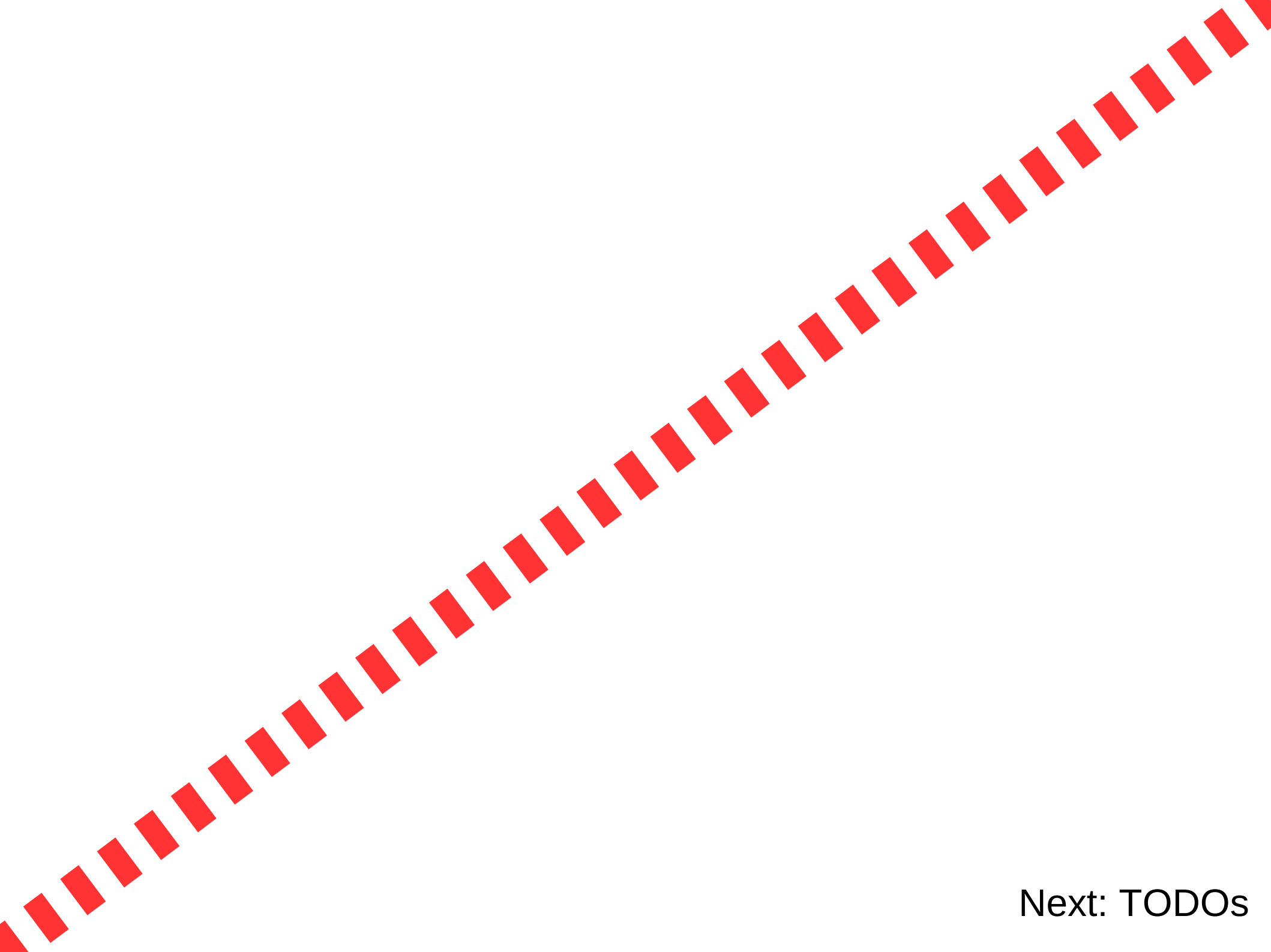


No Hocus Pocus! But Presto!









Next: TODOs

TODO – 'deep internals'

pseudo- random order

Use UCESB input stage (file reading, not unpacking)

Refactor calibration parameter parser and handling:
- At least linear time-variations support.
- Options for easier description of time (validity ranges)
- Units
- Cached

Input (simulated) data from file / external program (reversed ntuple/tree writer)

(Parse the cabling info tables instead of compiling)

Standalone command line (pipe-capable) wrapper for LSQR

((UNPACK level (á la UCESB) would be useful for some corrections. However, either from UCESB or separate, has major (internal) consequences))

TODO – 'internals'

pseudo- random order

Separate collection from solution in cosmic1/phase1/phase2

Get the tracker running

Get cosmic2 running (handle semi-broken paddles)

Millepede II calibrations, e.g. starting with cosmic3/phase3

Addback routine for γ detectors (baseline: landpaw-'translation')

TACQuilas are coming! – handle that

TACQuilas are coming! – get the walk handling into gear

RHIT level (hits on detectors as calculated from the tracks)

join_params – to handle / condense time varying parameters (not land02 'proper' internal)

pseudo- random order

TODO – 'internals' (cont'd)

Dump more internal state into ntuples/root trees:
- Error/uncertainty estimates
- Bitmasks?

pseudo random order

TODO – externals (land02-'independent')

Devise reconstruction correction and calibration for NTF smiley (small effect also in LAND, check TFW...)

More on-line graphs and spectra from struct_writer

How to 'calibrate' a tracker? I.e. determine detector positions...

SST calibrations

Post-tracker:

Further PDC calibration and reconstruction