

Ucesb & R³BROOT Data flow from DAQ to =)

Bastian Löher

R³B Collaboration Meeting June 2015

Goals

- Analyse full experiments in **2017**
 - Simulation, calibration and reconstruction
- Needs some **practice on existing data**:
 - Analysis of latest S438B experiment
- **Working online analysis** during data taking

R³BROOT

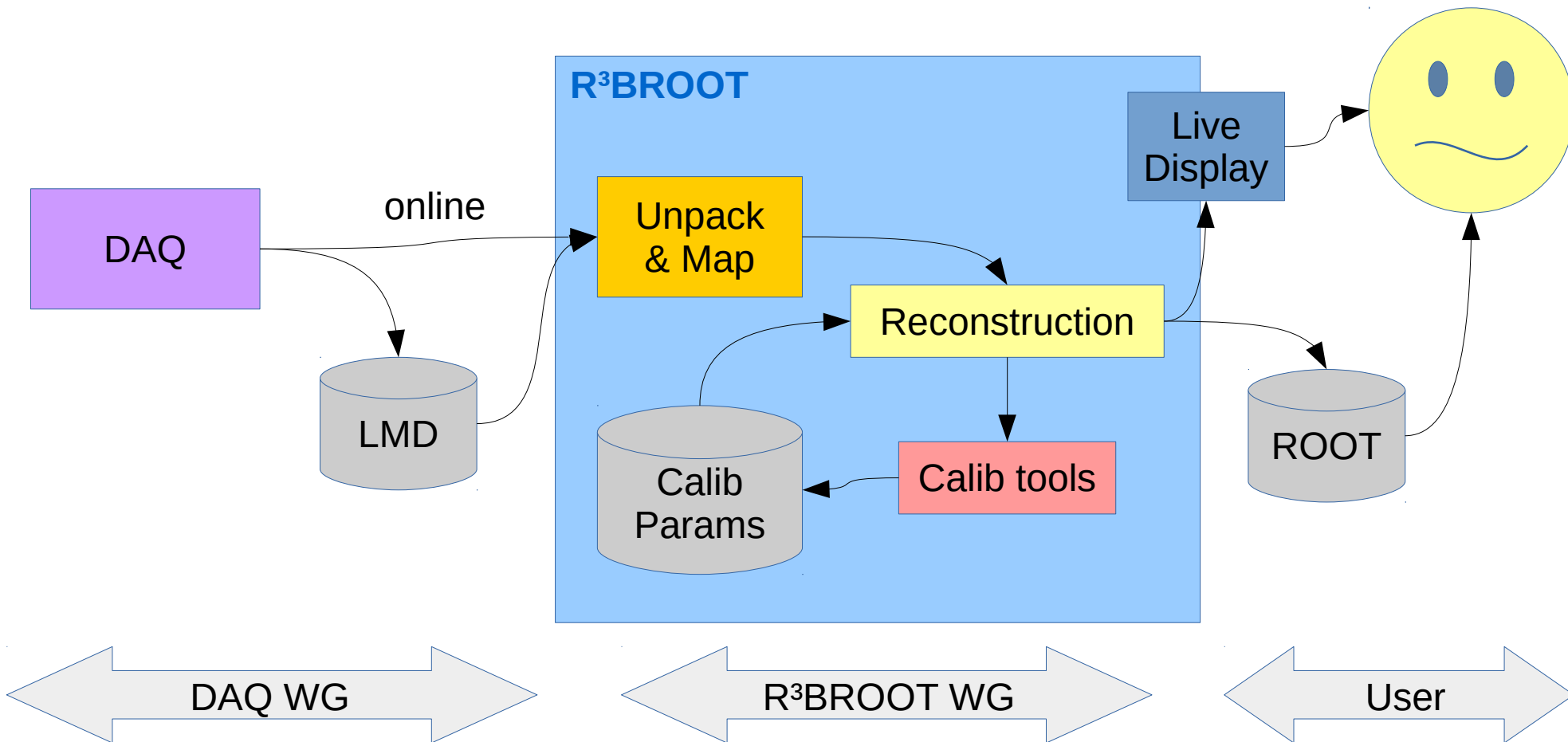
<http://r3broot.gsi.de>

- Proposed analysis tool for current and future experiments
- Based on Fairroot → inherits a lot of functionality (file handling, simulation, ...)
- Does not ship with unpacking functionality
 - ➔ Unpackers must be written by hand
 - Even for testing / debugging
- Same for channel mapping



Data flow

- Pure R³BROOT



But there is ucesb!

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

- Generic unpacker generator with user friendly syntax (no C++ necessary)
- Includes strict data checking for free
- Makes debugging easy
- Reads data from various sources (file, network, pipe...)
- Can act as online data server
- Writes to file, network, pipe or external programs

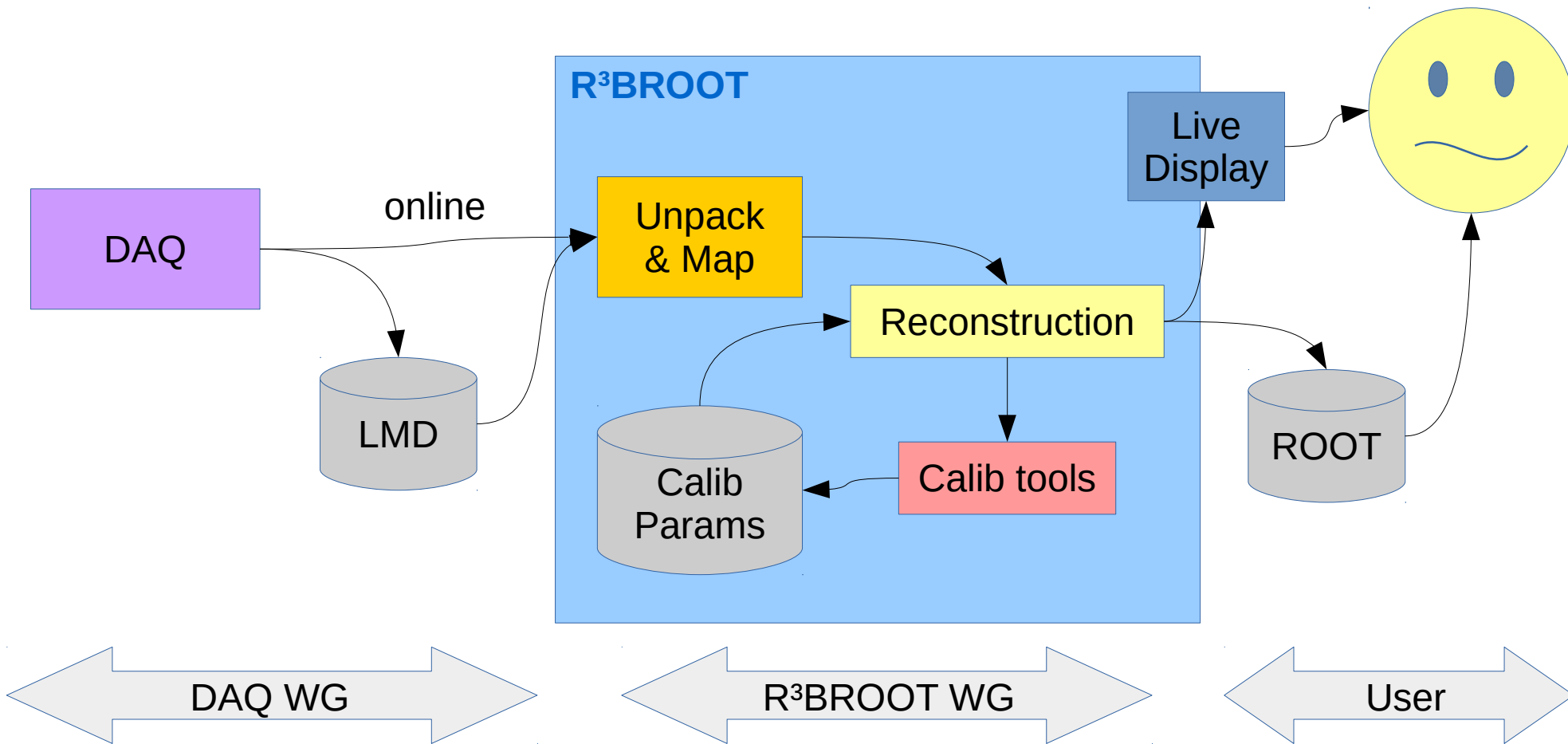
But there is ucesb!

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

- Generic unpacker generator with user friendly syntax (no C++ necessary)
- Includes strict data checking for free
- Makes debugging easy
- Reads data from various sources (file, network, pipe...)
- Can act as online data server
- Writes to file, network, pipe or external programs

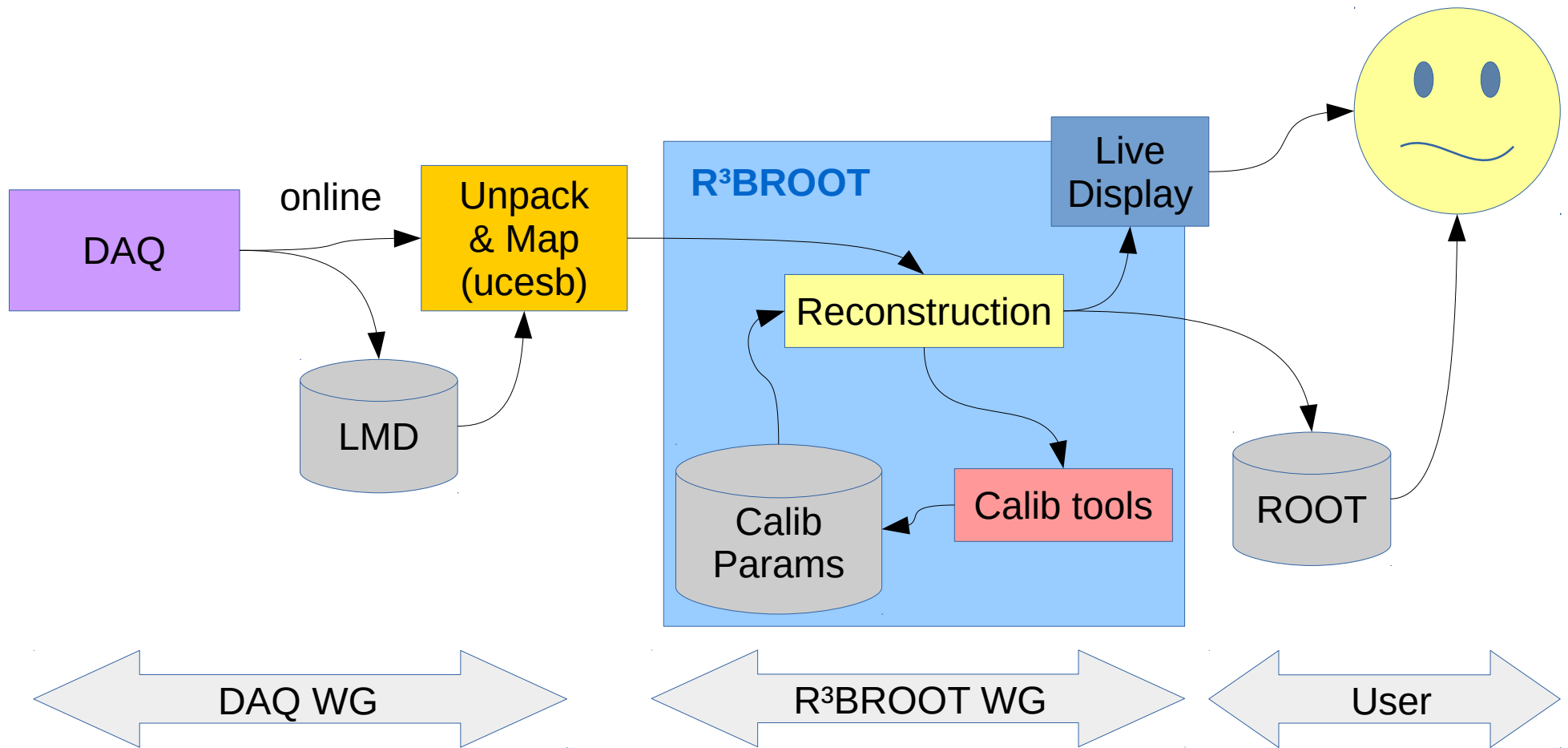
Data flow

- Pure R³BROOT



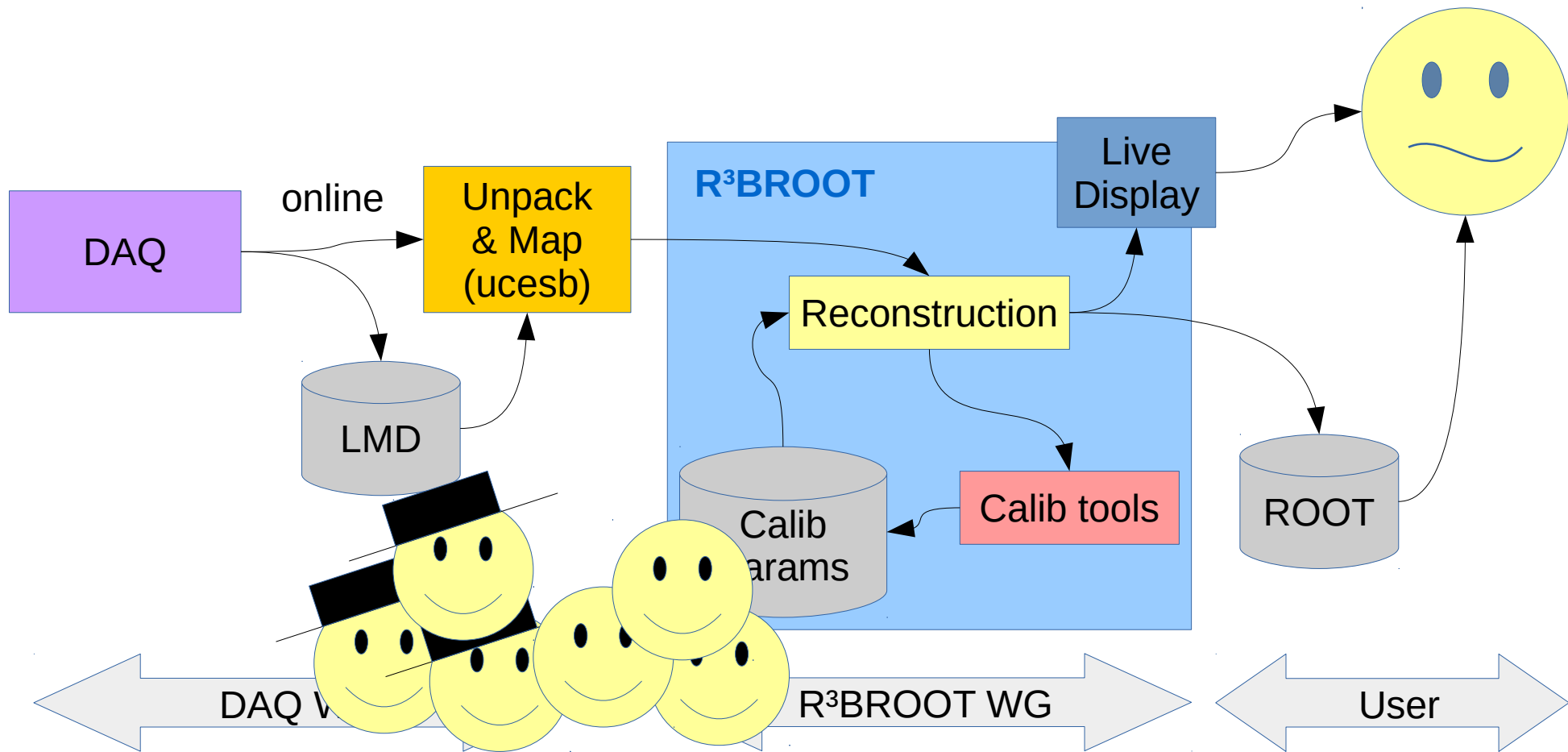
Data flow

- R³BROOT & ucesb



Data flow

- R³BROOT & ucesb



Other reasons

- **Project plans** for both scenarios:
 - [http://web-docs.gsi.de/~land/tjps/r3broot/Overview%20\(native\).html](http://web-docs.gsi.de/~land/tjps/r3broot/Overview%20(native).html)
 - [http://web-docs.gsi.de/~land/tjps/r3broot/Overview%20\(ucesb\).html](http://web-docs.gsi.de/~land/tjps/r3broot/Overview%20(ucesb).html)
 - Using ucesb as unpacker can save ~4 months of development time (assuming 2.8 developers)
- **Ucesb development** needed anyways
 - Unpackers for most detectors exist already
 - Improvements for DAQ operation
 - Apply directly to the R³BROOT use case

How exactly?

- **Interface** between ucesb & R³BROOT
 - struct_writer and ext_data_client (→ [Håkan's talk](#))
 - R³BROOT forks off a 'ucesb' to input data
 - **class FairUcesbSource**
to read ucesb input and
copy from C struct → TClonesArray
 - Deliver unpacked and mapped data directly to
R³BROOT user Task (RAW level for now)

ucesb side → Done
R³BROOT side → Work in Progress!

Next up:

- Gory details



- Questions?