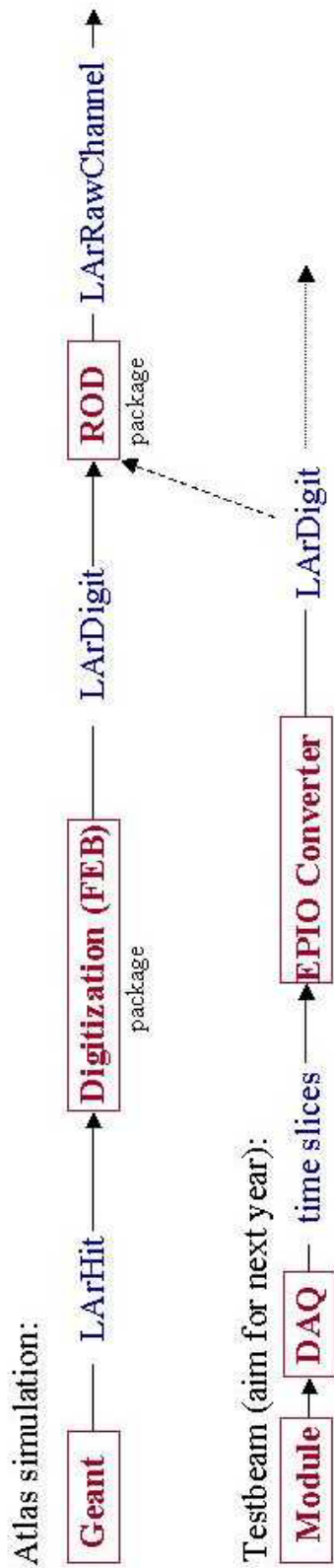


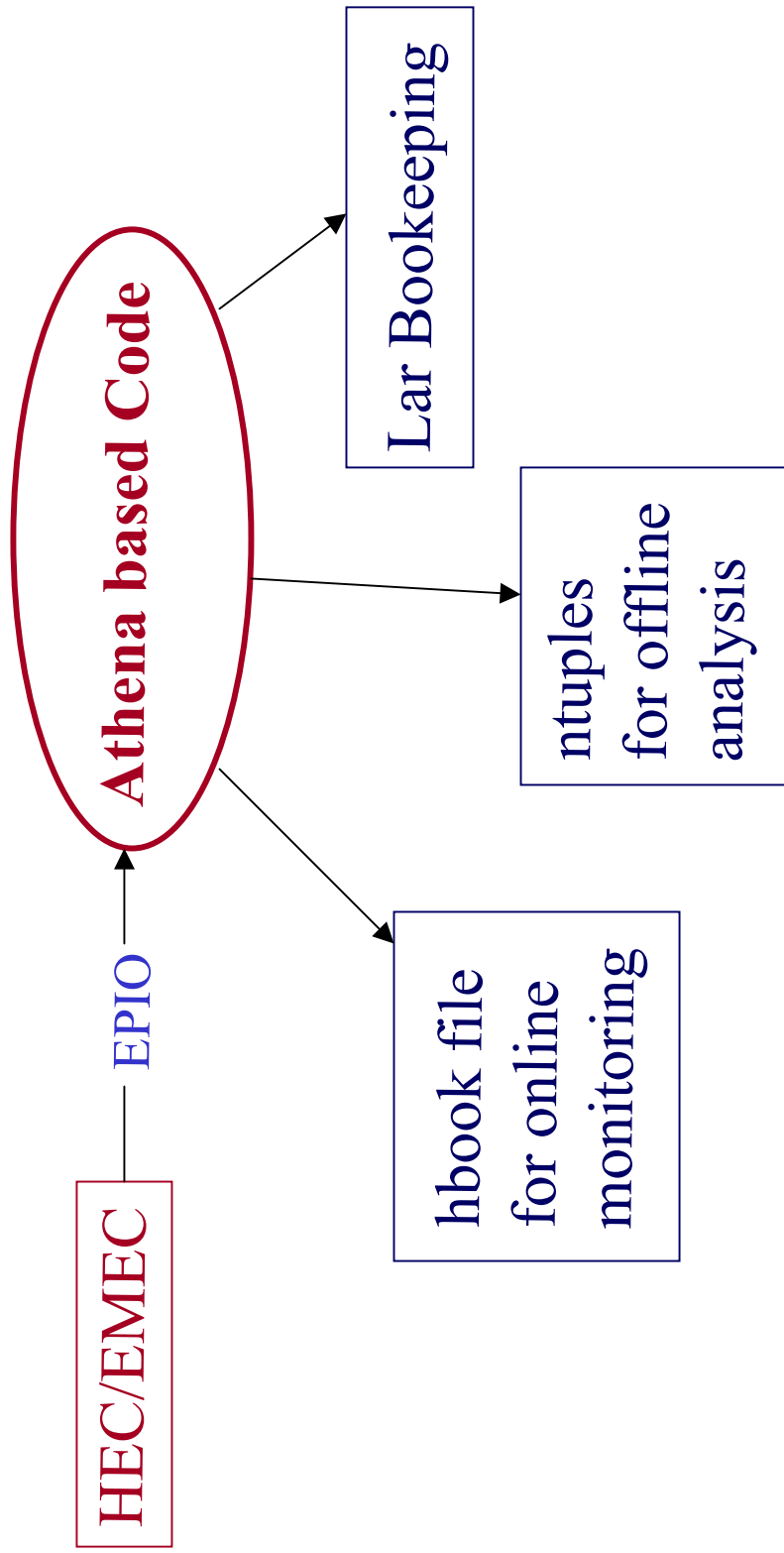
## Moving towards an OO Data Format

- Atlas data will be stored in an OO format
- Simple overview of data flow
- Testbeam data will be a good opportunity to gain experience with OO data
- hec\_adc - laid out in an OO philosophy  
“road map” is intended to be a guide for C++ code development
- First step ‘EPIO converter’ by Naoko Kanaya (next talk) already accomplished

# Data Flow



## Goal for OO Testbeam package:



### Milestones:

1. EPIO converter ✓
2. Hec\_adc ntuple (Dec.?)

**Click here ---> [□](#)  
to start **hec\_adc** road map**

Some useful road map overview figures!

**hec\_adc top level ps file**

**hec\_online tree ps file**

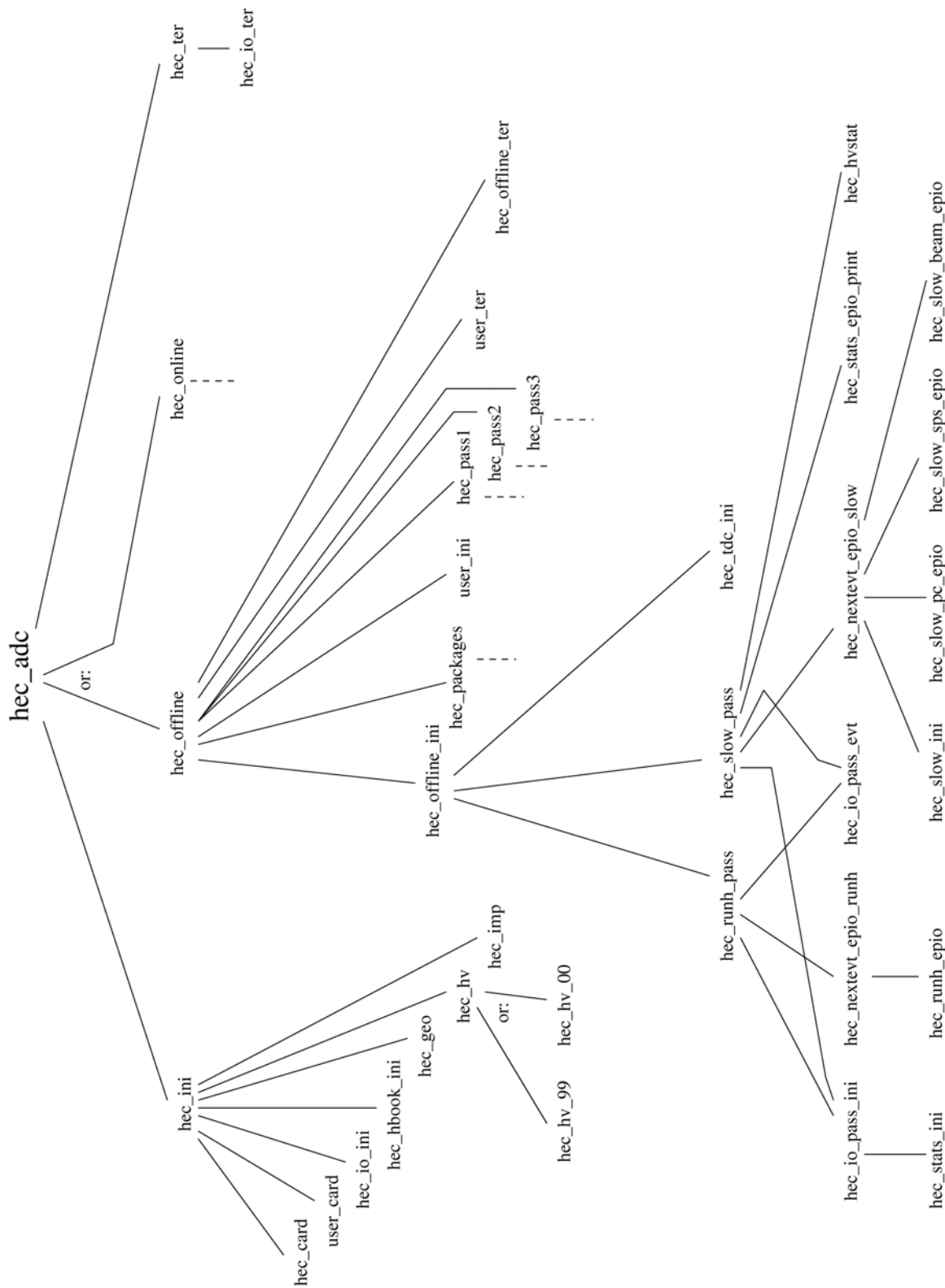
**hec\_online\_pass1 ps file**

**packages 1st page , 2nd page , 3rd page , 4th page , 5th page**

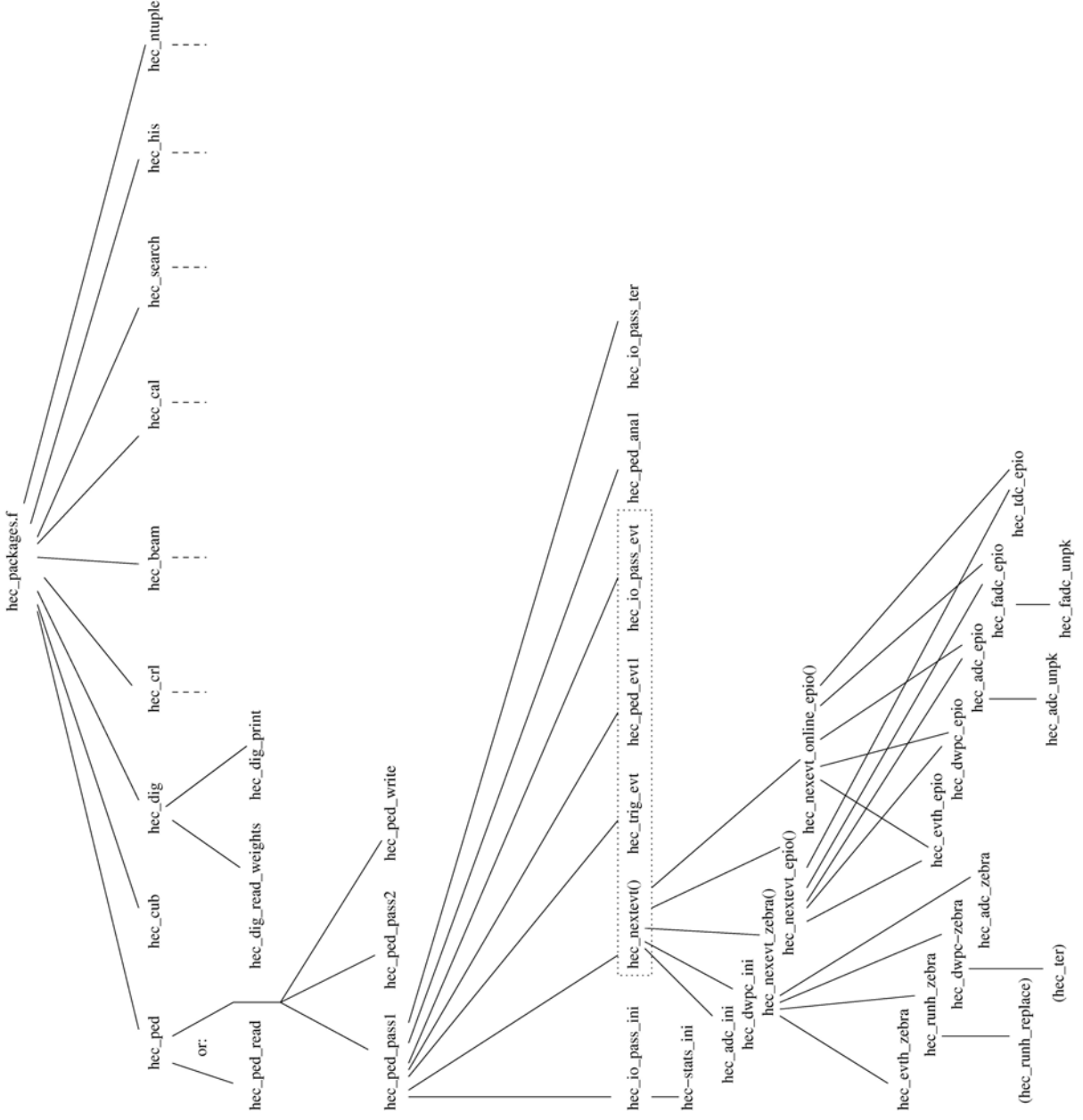




```
      subroutine hec_offline
c
c offline mode of hec_adc
c
      implicit none
c
c user datacard values common
c
      include'hec_par.inc'      !shared parameters
      include'hec_datacard.inc' !datacard values
c
c offline initialization
c
      call hec_offline_ini
c
c call system packages main routine
c
      call hec_packages
c
c user global initialization
c
      call user_ini
c
c pass 1 over data set for user
c
      if (n_pass .ge. 1) call hec_pass1
c
c pass 2 over data set for user
c
      if (n_pass .ge. 2) call hec_pass2
c
c pass 3 over data set for user
c
      if (n_pass .ge. 3) call hec_pass3
c
c user termination
c
      call user_ter
c
c offline termination
c
      call hec_offline_ter
c
      end
```







# Summary

- `hec_adc` - “road map” has been created to be a guide for C++ code development
- Next talk by Naoko Kanaya:  
EPIO converter