```
<< My adjustment of PHTimeStamp( in camDataIni.C ) >>

 PHRunHeaderRoot->set_run(runNumber);
  PHRunHeaderRoot->lookup_timeStamp("runtotime.txt");
  TimeStamp = PHRunHeaderRoot->get_timeStamp();
  cout << "PHRunHeader-I: Read in TimeStamp " << TimeStamp << "\n";
  cerr<<"timestamp calendertime           ="<<TimeStamp.getTics()<<endl;//yok
  TimeStamp += 13*3600;//yok
  cerr<<"timestamp calendertime(corrected)="<<TimeStamp.getTics()<<endl;//yok
  //  int a;cin>>a;//yok


<< My adjustment of PHTimeStamp( in camDataRun.C ) >>
   TimeStamp = PHRunHeaderRoot->get_timeStamp();

        cerr<<"eventloop time           ="<<TimeStamp<<endl;//yok
        cerr<<"eventloop time           ="<<TimeStamp.getTics()<<endl;//yok
        TimeStamp += 13*3600;
        cerr<<"eventloop time(corrected)="<<TimeStamp.getTics()<<endl;
        //        int a;cin>>a;//yok
--------------------------------------------------------------------
<< EMCal library code using PHTimeStamp from prdf? >>
PHBoolean mEmcCalibratorModule::event(PHCompositeNode *root)
{
...
  PHTimeStamp*  when ;

  PHNodeIterator i(root);
  PHDataNodeIterator DNiter(root);

  // Get Raw Data from the PRDF Node
  if (fVerbose > 0)
    cout << "mEmcCalibrator>>> Getting raw data from PRDF node " << endl;
  PHDataNode<Event> * thisEventNode =  (PHDataNode<Event>*) (i.findFirst("PHDataNode","PRD
F")) ;

 // Set up dEmcCalibTower and fill it
  if (fVerbose > 0)
    cout << "mEmcCalibrator>>> Getting raw data from PRDF node dstNode " << endl
;
  PHIODataNode<PHTable>* dEmcCalibTowerNode = (PHIODataNode<PHTable>*)i.findFirs
t("PHIODataNode","dEmcCalibTower");
  dEmcCalibTowerWrapper * dEmcCalibTower = static_cast<dEmcCalibTowerWrapper*>(d
EmcCalibTowerNode->getData());

 // Proccessing the event
 if(fVerbose>0)
   cout << "mEmcCalibrator>>> Processing event ..." << endl;
 fRda->GetDataExtractor()->processEvent(thisEventNode->getData());

 // when.setToSystemTime() ; // should be provided by rdo
 when =  thisEventNode->getData()->getTimeStamp();
```