The Calorimeter COMMON package v. 2.00 README

Emiliano Mocchiutti
5th December 2005

In this README file there is a description of how to install the calorimeter packages and how to configure the PAMELA working environment. A scheme of dependencies of packages is shown in section 2 of this document and can be found as text file in the doc directory.

INDEX:

---

1) INSTALLATION:

The software has been written to satisfy the PAMELA repository requirements but it can be compiled and installed also outside the PAMELA environment.
NOTICE: when you compile using the PAMELA environment programs will look for libraries and calibrations in the PATH determined by the PAMELA environmental variables. If you move directories and change accordingly your environmental variable the programs will still work. Moreover you will have installed under the “macro” directory all the ROOT macros.
When you compile outside the PAMELA environment macros will not be installed and only standalone programs will work. Path to calibrations will be statically linked inside executables, if you change the path to calibrations the compiled programs will NOT work anymore unless you recompile them.

The calorimeter COMMON package is required to install the other packages.
Dependencies are shown in section 2, the installation order if you want to install the full set of packages is (when on the same line the installation order does not matter):
- calorimeter COMMON
- ROOT2PAW, calorimeter QLOOK
- EVENTVIEWER
- calorimeter LEVEL2
- calorimeter UTILITIES

1A) Downloading software.

You can find the software on the PAMELA afs repository:

/afs/ba.infn.it/user/pamela

or you can download it from the WEB, in this case go to:

http://pcba28.ba.infn.it/cgi-bin/cvsweb.cgi

from the menu “CVS Root” choose “PAMELA repository”, click on the package name you need and click on “Download this directory in tarbll or zip archive” depending on your preferences.

1B) Installing software USING the PAMELA environment.

a) once you have downloaded your package unpack it somewhere.

b) enter the created directory, you will find these file and directories:

doc/ macros/ inc/ src/ bin/ lib/ Makefile calib/ data/ install.sh

c) make sure you have set up your PAMELA environmental variables and directories. To do so, choose a path where you want to install the PAMELA software (let's say /mydirectory/pamela/) and create the following directories: bin , lib , src , inc , macros , docs , calib .
Then edit your login configuration file (people using bash shell will edit $HOME/.bashrc , people using tcsh will edit $HOME/.tcshrc and so on), and add the following environmental variables:

```
export PAM_BIN=/mydirectory/pamela/bin
export PAM_LIB=/mydirectory/pamela/lib
export PAM_SRC=/mydirectory/pamela/src
export PAM_INC=/mydirectory/pamela/inc
export PAM_MACROS=/mydirectory/pamela/macros
export PAM_DOC=/mydirectory/pamela/docs
export PAM_CALIB=/mydirectory/pamela/calib
export PAM_YODA=/mydirectory/pamela/yoda/
```

The last two variables determines which is the installation directory of YODA. It must contain the “lib” directory in which the file libyoda.so is located and the “include/yoda/” directory where it is possible to find the YODA headers.
At this point export the LD_LIBRARY_PATH environmental variable:

```
export LD_LIBRARY_PATH=$LD_LIBRARY_PATH:$PAM_LIB:$PAM_YODA/lib:$ROOTSYS/lib
```

if you are able to run ROOT then the ROOTSYS variable should already been defined, if not define it as the installation path for ROOT.
I also suggest to add the PAMELA bin directory to your default bin path but this is not essential:

```
export PATH=$PATH:$PAM_BIN
```

People using tcsh will use “setenv” instead of “export” and will put a space at the place of the sign “equal”. This set up has to be done once forever.

d) once the PAMELA environment is set, just cd to the package directory and give:

```
./install.sh
```

the program will check the system and create the needed libraries. Then it will install the package files in the paths you have given.

1C) Installing outside the PAMELA environment

cd to the package directory and give:

```
./install.sh --no-env --install-dir=/mydirectory/pamela/
             --with-yoda=/mydirectory/pamela/yoda/
```

where --with-yoda must contain the same path as if PAM_YODA was used while --install-dir gives the installation path.

---

2) PACKAGES DEPENDENCIES:

PACKAGE	REQUIRES	PROVIDES
CALOCOMMON	(none)	CaloFunctions.h
		calomysqlstruct.h
		caloclasses.h
		caloclasses_h.so
		libqumy.so
		libremy.so
		libqumy_C.so
		libremy_C.so
		rootlogon.C
		CaloFINDCALIBS.c
		CaloFINDCALIBS_c.so
		CaloFINDCALIBS.C
		CaloADC2MIP.root
		calorimeterCOMMON.README.pdf
		packages
		calocommon
ROOT2PAW	CALOCOMMON	liboptrklev1.so
		libretrklev1.so
		libcltrklev1.so
		liboptrklev2.so
		libretrklev2.so
		libcltrklev2.so
		libopaclev1.so
		libfiaclev1.so
		libclaclev1.so
		liboptrklev1_C.so
		libretrklev1_C.so
		libcltrklev1_C.so
		liboptrklev2_C.so
		libretrklev2_C.so
		libcltrklev2_C.so
		libopaclev1_C.so
		libfiaclev1_C.so
		libclaclev1_C.so
		liboptoflev1.so
		libretoflev1.so
		libcltoflev1.so
		liboptoflev1_C.so
		libretoflev1_C.so
		libcltoflev1_C.so
		aclev1class_h.so
		ctrkstruct.h
		ctrkinclude.h
		ctofstruct.h
		ctofinclude.h
		cacstruct.h
		trklev1struct.h
		aclev1class.h
		GrounDataConvert.c
		GrounDataConvert_c.so
		GrounDataConvert
		root2paw.README.pdf
		examples.c

CALOQLOOK	CALOCOMMON	CaloCHKCALIB.c
		CaloLEVEL1.c
		CaloMATRA.c
		CaloMIP.c
		CaloPLANES.c
		CaloPULSE.c
		CaloCHKSTRIPS.C
		CaloLEVEL1.C
		CaloMATRA.C
		CaloMIP.C
		CaloPLANES.C
		CaloPULSE.C
		CaloCHKCALIB_c.so
		CaloLEVEL1_c.so
		CaloMATRA_c.so
		CaloMIP_c.so
		CaloPLANES_c.so
		CaloPULSE_c.so
		CaloQLOOK.C
		CaloQLOOK.c
		CaloQLOOK_c.so
		QLpathtoc_c.so
		CaloCHKCALIB
		CaloLEVEL1
		CaloMATRA
		CaloMIP
		CaloPLANES
		CaloPULSE
		CaloQLOOK
EVENTVIEWER	CALOCOMMON	libptlevel1.so
	ROOT2PAW	libtrack.so
		libreadb2maps.so
		libptlevel1_C.so
		libtrack_C.so
		libreadb2maps_C.so
		EVpathtoc_c.so
		eventviewer.h
		EventViewer_c.so
		EventViewer.c
		EventViewer
		filter.c
		measure_n4_110402_corrected.rz
		measure_n3_290302.rz
		External_top_map_n4_150402.rz
		EventViewer.README.pdf
CALOLEVEL2	CALOCOMMON	libopcalol2.so
	ROOT2PAW	libcrcalol2.so
	libtrack.so (EVENTVIEWER)	libclcalol2.so
	libreadb2maps.so (EVENTVIEWER)	libopcalol2_C.so
		libcrcalol2_C.so
		libclcalol2_C.so
		ccal2struct.h
		CaloLEVEL2_c.so
		CaloLEVEL2.c
		CaloLEVEL2.C
		CaloLEVEL2
CALOUTILITIES	CALOCOMMON	CaloADC2MIP.c

libtrack.so (EVENTVIEWER)	CaloADC2MIP.C
readb2maps.so (EVENTVIEWER)	CaloCALIBSCAN.c
ccal2struct.h (LEVEL2)	CaloCALIBSCAN.C
	CaloTRKCALOALIG.c
	CaloTRKCALOALIG.C
	CaloTRKCALOALIG_c.so
	CaloADC2MIP_c.so
	CaloCALIBSCAN_c.so
	UTpathtoc_c.so
	CaloADC2MIP
	CaloCALIBSCAN
	CaloADC2MIPf.root
	CaloCALIBdata.root
	caloutilities