Subject: Re: switching off the information output
Posted by Florian Uhlig on Fri, 24 Feb 2012 14:13:43 GMT
View Forum Message <> Reply to Message

Hi Donghee,

To comment the lines is a way to get rid of the messages, but not the way you should use. There is much better way.
The FairLogger class takes care of all the logging and you can steer it from the macro. If you don't like output switch it off or dump it to a file. The way to do it is to
get a handle to the FairLogger in the macro and then to define it the way you want.
So you can switch of the output for a production run, but you can switch it on again in case your program crashes and you want to debug it.

```
  // This lines go to the macro

  // get handle
  FairLogger *logger = FairLogger::GetLogger();

  // define log file name
  logger->SetLogFileName("MyLog.log");

  // log to screen and to file
  logger->SetLogToScreen(kTRUE);
  logger->SetLogToFile(kTRUE);

  // Print very accurate output. Levels are LOW, MEDIUM and HIGH
  logger->SetLogVerbosityLevel("HIGH");

  // Set different levels of verbosity. In the example everything >=INFO goes to the
  // file and everything >= ERROR is printed on the screen
  // LogLevels are (FATAL, ERROR, WARNING, INFO, DEBUG, DEBUG1, DEBUG2,
DEBUG3, DEBUG4)
  logger->SetLogFileLevel("INFO"); // FATAL, ERROR, WARNING and INFO
  logger->SetLogScreenLevel("ERROR"); //Only FATAL and ERROR to screen
```

Hope this explanation helps to solve your problem.

Ciao

Florian