Logging system

As of svn repository revision 799, mpifxcorr uses a new logging infrastructure. Prior to this change logging information was sent to stdout or stderr by using the C++ streams cout and cerr. This was a bit limiting as each MPI process would have output going to the same place, often with the result that two log messages would be spliced into one unintelligible string. Now there are 7 new output streams that work just like cout and cerr that are defined in alert.h and alert.cpp. These are:

1. **cfatal** message related to an event causing immediate termination of mpifxcorr, including illegal command line parameters
2. **csevere** message related to OS or software failure (most certainly a bug in the program)
3. **cerror** message related to a problem likely causing loss of data
4. **cwarn** message related to a possible loss of data or use of poorly tested code
5. **cinfo** informational message
6. **cverbose** informational message of minor importance
7. **cdebug** message related to debugging of intereste to developer only

Please use these instead of cout and cerr within mpifxcorr. Source files that use these simply need to include alert.h .

Note! messages using these streams must begin with *cxxx* << `startl` << and end with << `endl;` . These two *stream manipulators* are required for thread safety.

The output of these functions depends on the configuration of mpifxcorr at compile and run time. If difxmessage support was not compiled into mpifxcorr, then these messages will be directed to cerr or cout depending on the severity (cerr gets the first three streams: cfatal, csevere, and cerror). If difxmessage support is compiled into mpifxcorr, two environment variables will specify where the information is sent. If DIFX_MESSAGE_PORT is set to a positive integer and DIFX_MESSAGE_GROUP is set to a valid multicast group, then these messages, along with an appropriate severity code, will be sent via a multicast XML message to the specified group/port. Otherwise the messages will be redirected to cerr and cout. Note that all messages going to cfatal will get a copy sent to cerr even if they are also multicast to ensure that the person running the program knows why it crashed. A message to cfatal should precede any unexpected exit of mpifxcorr. Additionally, such an unexpected exit should exit with MPI_Abort(MPI_COMM_WORLD, 1), not exit( ) , to maximize chances of a clean shutdown.