inlib:
- central logic: have a scene graph manager built over C++ and GL-ES inlib/sg.
- inlib/sg: a set of “nodes”: shapes (inlib::sg::cube, inlib::sg::vertices), cameras (inlib::sg::ortho, perspective, frustum, lrbt), inlib::sg::matrix to position objects, containers (inlib::sg::separator, group). An assembly of containers and leaf nodes is a scene graph. inlib::sg::action: to “traverse” a scene graph for doing something specific. A central action is the “render_GL” action that produces the GL-ES for doing the rendering. (inlib/sg is strongly inspired from the great OpenInventor).
- Various tools as histograms (inlib::histo::h1d), fitter, file reader/writer (aida, csv, root), plotter, etc…
- Have a maximum of pure header code to ease the build of apps on most platforms.

GUI:
- We attempt to do it with inlib code (inlib::sg::button, etc…).
- It simplifies a lot of things and permit to build very compact applications.
- Since over GL-ES, we can embed buttons in (3D) scenes visualizing data.
- Today the look and feel is not so great but we have not said our last word…

exlib:
- set of pure header classes doing the bridge with external packages such as GL-ES, Android, iOS, Cocoa, X11, Win32, expat, jpeg, png, freetype, cfitsio, dcmtk, graphvis, Geant4, etc…

ourex:
- in order to ensure a strong portability, the same comportement on all platforms, and to ease the “build and install” of apps; we bring/master the external packages we need, such as expat, jpeg, png, etc… in the ourex library.

2007/2010: Arrival of the stores (AppleStore, AndroidMarket) that permit an easy world wide distribution of “apps”.
2010: Apple gave to LAL/SI four machines and eight screens for an “ARTS” (Apple Research and Technological Support) projet.

⇒ We reconsider the way we do graphics and GUI: softinex

softinex = inlib, exlib, ourex libraries
+ applications as ioda, g4view, wall etc…

Ask for a tablet demo!