

PELICANS: an Implementation Tool for Solvers of PDEs

The French Institute for Radiological Protection and Nuclear Safety (IRSN) has been developing for many years simulation softwares of thermo-mechanical engineering, involving various domains like heat transfer, fluid mechanics (laminar or turbulent, reactive, multi-species, multi-phase, Newtonian or non-Newtonian) and solid mechanics (elasticity, viscoplasticity, damage and fracture). In answer to the growing need for capturing the commonalities between these developments, an abstracted view of the task of building numerical simulations has been elaborated. The starting point was the recognition that the overall process could be split into three steps: in the mathematical modeling stage, the physical problem under consideration is stated as a *system of partial differential equations* (PDEs), then a numerical method is designed for their solution and, finally, a computational code is developed. For a wide range of thermo-mechanical simulations, it appeared that the complexity and the cost of this very last step could be dramatically reduced with the use of implementation tools focusing on the PDEs themselves rather than on the physical background. The emphasis of this abstracted view is on the implementation purpose.

In this context, a C++ application *framework* with a set of integrated *reusable components*, called PELICANS, has been designed at IRSN to simplify the task of developing applications of numerical mathematics and scientific computing, particularly those concerning PDEs and initial boundary value problems. PELICANS provides functionalities for dealing with multi-dimensional discretizations on structured and unstructured meshing with finite volume and finite element like methods. The following situations are accessible: moving boundaries, grid motion, characteristic methods, adaptative multilevel mesh refinement, non conforming multi-domain. Moreover, facilities are supplied for distributed processing and consistent coupling with external established numerical libraries.

Nowadays PELICANS is the numerical kernel of several industrial codes developed at IRSN for nuclear safety studies.

PELICANS defines a “prescriptive” approach, without being unnecessarily restrictive, to building applications. It provides both standard component types with a great deal of built-in behavior and standard structures for code. The standard *components* can be combined in many ways to build a variety of applications. They can be extended by customizing them or by building new ones with a similar structure. They represent code application developers don’t have to write, because it is already done for them. The standard code structures, or *framework*, provide pre-built applications with all their generic features already implemented. Developers use object-oriented techniques like inheritance to build on the generic application and tailor *components* to the new application. Thus PELICANS allows developers to focus on problems unique to the application at hand. Elements commonly used are handled in the *framework*. This increases the speed of development, reduces the amount of new code written, enables wide-scale software reuse and reduces application maintenance costs.

A specific C++ usage methodology has been adopted in the development of PELICANS (specialized tools have been developed to this end). The spirit of this approach is threefold. Firstly, a support for *Design By Contract*, which is considered to be of major importance for *Component-Based Development*, has been included. The relationships between software *components* are thus rationalized (in particular, the meaning of inheritance is restricted by the *Liskov Substitution Principle* formalized in terms of assertion overriding). Secondly, *Self-Documentation* is supported consistently with the *Design By Contract* assertions. Finally, the C++ capabilities whose use is considered delicate and tricky have been ignored, which maintains the accessibility of PELICANS to non C++ experienced programmers. This consistent and validated approach also represents an output of the PELICANS project.

PELICANS is distributed under the CeCILL-C license agreement (an adaptation of LGPL to the French law, see <http://www.cecill.info>). While being the property of IRSN, it is fully free, open-source, and may be reused in softwares distributed under any license.

The project page is: <https://gforge.irsn.fr/gf/project/pelicans>