**Use of open-source software at ENEA for nuclear reactor safety simulations**

A. CERVONE1, S. MANSERVISI2, P. MELONI1, M. POLIDORI

1 ENEA FSN-SICNUC, via Martiri di Monte Sole 4, Bologna, Italy

2 UNIBO, Industrial Engineering Department (DIN), via dei Colli 16 Bologna, Italy

Email contact of corresponding author: antonio.cervone@enea.it

In the past 10 years, ENEA has introduced and developed the use of open-source software in the simulations performed for nuclear reactor safety assessments. Historically, the division for nuclear safety and security built their capabilities and competences on domain-specific proprietary thermal-hydraulics simulation tools, such as RELAP (distributed by US-NRS) and CATHARE (developed and distributed by CEA), as well as proprietary severe accident simulation tools such as MELCOR (distributed by US-NRC) and ASTEC (developed and distributed by IRSN). This path was common among many institutions in the nuclear safety field, due to the difficulty in resources involved in the development of in-house codes and especially in their validation. Only a very little part (if not any) of nuclear safety specific codes have been available as open-source.

The introduction of CFD as a support tool for nuclear safety assessments led to the usage of some general purpose simulation tools such as FLUENT and ANSYS-CFX. These tools were (and are) still proprietary software, but some open-source alternatives are available, differently from the case of the dedicated nuclear safety tools mentioned above. The main driving force that led to the transition from proprietary software to open-source were two: coupled simulations involving thermal-hydraulics codes and CFD tools, and massive parallel simulations of large and detailed domains. In fact, the flexibility offered by open-source software has been crucial in the development of such application domains.

For what concerns the coupling of system thermal-hydraulics codes with CFD, the motivation to use open-source software has been the possibility to perform online in-memory coupling of the codes, without the usage of external files that are stored and re-read from disk (offline coupling). Proprietary software gives limited access to its inner components by design, making it very difficult (when not impossible) to gain direct control of the inner data structures and procedures, while open-source software can be manipulated by knowledgeable users, that can develop the missing pieces that can be used to access to desired data. This approach has been supported in particular by the French code environment and a series of European projects (NURISP, NURESIM, NURESAFE) where a dedicated coupling platform, SALOME, has been developed as open-source software. This platform has been extended to cover many functionalities on top of the coupling, such as mesh generation, run management and post-processing of the simulation results, leveraging many other open-source software packages, such as OpenCASCADE, ParavIew, etc. The heart of the coupling methodology is implemented in the MED and MEDCoupling libraries, also available as stand-alone open-source projects.

The coupling with the proprietary software, such as CATHARE, is instead managed by a proprietary library named ICoCo, that offers similar functionalities to the MED library, giving access to the inner data structures of the code for their manipulation by the coupling interface. The usage of this approach has enabled the coupling of the thermal-hydraulics codes with some open-source CFD software such as Trio\_U (later split in the TRUST library and the Trio\_CFD extension), CODE\_Saturne (and its proprietary extension for two-phase flows NEPTUNE\_CFD), libmesh and OpenFOAM. The former two, Trio\_U and CODE\_Saturne, are also developed and distributed by the French nuclear institutes and therefore where already included in the open-source coupling platform, while the latter where not. In fact, the coupling interfaces for the latter two codes have been developed by ENEA in collaboration with the University of Bologna. Libmesh is a general purpose finite element open-source library, that can be successfully used for CFD applications as well as many other simulation fields such as structural mechanics and porous media, that is now the base of the MOOSE library developed by Idaho National Laboratories, that has been extensively used for simulations in the nuclear field. OpenFOAM is the most renowned open-source library in the CFD field, based on the finite volume approach. The library is constituted by many modules that aim to be on par with commercially available software such as FLUENT.

The other application domain where the open-source software has been selected at ENEA is in performing massive parallel CFD simulations that leverage the on-premise HPC machine CRESCO. While proprietary software is available on the cluster, the number of license is limited and the cost for licenses that would cover all available computation nodes is prohibitive. The open-source stack, based on the GCC compiler with its OpenMP implementation, the OpenMPI library and the petsc linear algebra library have been therefore deployed on the cluster, together with many other open-source supporting libraries, in order to use the same open-source tools described in the previous section.

Another field that has seen the extensive usage of open-source software is the atmospheric dispersion of nuclear pollutant. In particular, the Lagrangian particle simulation tool FLEXPART has been adopted for its capabilities in simulating a large number of particles and taking into account for the dispersion, advection, deposition and decay mechanisms in the atmosphere. This software has been available as open-source from the start, and leverages meteorological data distributed by the European Centre for Medium-Range Weather Forecast (ECMWF). The meteorological data, at least for the past dates and not for the forecast, is openly available in the Copernicus EU initiative.

Aside from the main application software, many open-source tools have been adopted and even developed to support the simulation activities. In particular, for mesh generation the main tools used are the aforementioned SALOME Platform and Gmsh, while the post-processing is handled almost exclusively with Paraview, that supports many software formats, while some additional plug-ins have been developed to extend its use to some specific simulation domain. Open-source packages such as spack have been adopted for the consistent deployment of software on the workstations and clusters. The python software ecosystem, with its packages, is regularly used to process pre- and post-processing jobs, analyse results and, recently, develop Artificial Intelligence (AI) and Machine Learning (ML) models to aid the traditional simulation tools.

The use of open-source tools has been also very valuable with students that have developed their theses or internships at ENEA. In fact, promoting the usage of open-source software, that remains available to the student beyond the period at ENEA and therefore can be used in other domains, has been positively received by the applicants and has been a major focus on the list of activities that are and will be proposed.

The adoption of open-source software has come with many challenges that are more or less similar for all the different simulation tools. An extensive knowledge of operating systems and software deployment is required to have a working and performing installation of the tools, especially for effective parallel simulations. The most prominent disadvantage, however, is often the lack of documentation and support, that leads to further knowledge that a user may require, such as programming skills and debugging, in order to analyse in depth the code and understand the code behaviours that are not described in the documentation.

Generally speaking, however, the open-source approach has demonstrated to be invaluable in the tasks that have leveraged it, with the benefits largely outweighing the drawbacks, with the intention of extending its usage in the future. This strategy can be particularly useful for the safety division at ENEA that does not have the human resources required to develop in-house a major software, but that can launch and contribute to other initiatives under the open-source umbrella.