PROOSIS is a stand-alone, flexible and extendible object-oriented simulation environment for modelling gas turbine engines and other systems (control, electrical, thermal, hydraulic, mechanical, etc.). It was originally developed by Empresarios Agrupados Internacional S.A. and an aeronautics consortium of European universities, research institutes and corporate companies. It is based on EcosimPro, the European Space Agency’s preferred tool for rocket propulsion, environmental control and life support systems, among others.

PROOSIS has an advanced Graphical User Interface and uses a high-level, “engineer-friendly” object-oriented language (EL) for modeling engine systems and state-of-the-art technologies in areas such as numerical solvers, non-causal modeling of reusable libraries, XML file formats, map handling etc.

Any gas turbine engine configuration or system can be constructed graphically by ‘dragging-and-dropping’ the required component symbols from the included libraries to a schematic window. Using EL, users can also create new components and libraries, or extend the existing ones.

PROOSIS is capable of steady state and transient simulations as well as customer deck generation (dll, exe, ARP4868, AS4191). Different types of calculations can be performed (single or multi-point design, off-design, test analysis, sensitivity, parametric and optimisation studies, mission analysis, diagnostics, control system design and test, etc).

PROOSIS can also perform multi-fidelity, multi-disciplinary and distributed simulations. These are greatly facilitated by its open architecture, which allows it to connect to external commercial (Excel, Matlab, COM) or in-house tools and link with codes written in C, C++ and FORTRAN.

These features make PROOSIS a useful tool for all phases of the engine life cycle, from preliminary and detailed design to post-certification and in-service support, and allow it to serve as a common framework in multi-partner collaborative engine projects providing common standards and methodologies.

Lastly, PROOSIS also provides a multi-domain simulation platform for the simulation of gas turbines, engine/aircraft systems and power plants.
**BASIC CONCEPTS**

The most important concept in PROOSIS is the "Component". A "Component" contains a mathematical description of the corresponding real-world component. For example, a combustion chamber is a Component. Components communicate with one another through their Ports. Ports define the set of variables to be exchanged between connected Components. Components and ports are stored in reusable Libraries, like the TURBO library for gas turbine modeling.

The development of libraries requires a user (Level 3) expert on the physical and mathematical formulation of component models and with deep knowledge of the PROOSIS modeling language EL.

Using existing libraries, a user (Level 3) can construct a model graphically by dragging-and-dropping the required component icons from one or more library palettes to a schematic diagram. The Components, the appropriate ports and editing their attributes. A model, for example, could be a single component, a sub-assembly or a complete engine and its subsystems (oil, fuel, control, etc.).

The mathematical representation of a model is called a "Partition" in PROOSIS and it is set with the help of wizards. For a given "Partition", a user (Level 3) can define different simulation cases ("Experiments") using either wizards or the object-oriented language EL.

Experiments can run either in batch mode or graphically using the Monitor application. A log file with the user-defined level of information is produced for debugging.

From a working experiment and through a wizard, different types of Decks can be generated for delivering a model to a customer (Level 4 user).

PROOSIS also gives engineers a multi-domain simulation environment for modeling gas turbines and engine/aircraft systems, as shown below:

**MAIN FEATURES**

- EL modeling language for the creation of new components and libraries. Its non-causal approach allows the modeler to focus on the intrinsic physical model to solve issues. Its object orientation allows extending and easy to maintain the libraries. LSB and FORTRAN functions can be called from the EL language, and integration with XML files is easy through the XLM parser.
- Models and calculations can be encapsulated and exported in a stand-alone application (deck) with user-defined accessible input/output variables. The deck can be used in command line mode, through a graphical user interface, or integrated into C or FORTRAN applications by means of standard input/output.
- The symbolic mathematical kernel automatically extracts the mathematical model from the graphical schematic. The user selects the governing variables (e.g. pressure, flow, heat, temperature, etc.) and the numerical algorithms generate robust mathematical models. Symbolic calculations and equation re-ordering. The tool also allows selecting any iterative variables when non-linear equations are formed, and provides valuable information about the mathematical problem to be solved by the user.
- The available internal solvers allow dealing with stiff or non-stiff dynamic problems (DIASS, RICK and Adams-Moulton). The algebraic problems are tackled by an improved Powell's hybrid-based method. Inequality groups (conditional equations or limits) can be used in both steady and transient simulations. Convergence issues can be addressed with the help of the simulation log, in which the evolution of the variables, residues, simulation errors and statistics are summarized.
- The symbolic mathematical kernel automatically extracts the mathematical model from the graphical schematic. The user selects the governing variables (e.g. pressure, flow, heat, temperature, etc.) and the numerical algorithms generate robust mathematical models. Symbolic calculations and equation re-ordering. The tool also allows selecting any iterative variables when non-linear equations are formed, and provides valuable information about the mathematical problem to be solved by the user.
- A model can also be exported ready to be used in a HIL environment, where it can be integrated into C or FORTRAN applications by means of standard input/output.
-EL modeling language for the creation of new components and libraries. Its non-causal approach allows the modeler to focus on the intrinsic physical model to solve issues. Its object orientation allows extending and easy to maintain the libraries. LSB and FORTRAN functions can be called from the EL language, and integration with XML files is easy through the XLM parser.
- Moreover, with the PROOSIS toolbar for Ms Excel, models can be directly integrated into stand-alone spreadsheets.
- It is also possible to interact from MATLAB with PROOSIS models. That’s why, the model can be exported as an encapsulated S-Function block, which means it is possible, for example, to develop the engine with PROOSIS and the control system with Simulink.
- A model can also be exported ready to be used in a HIL environment, where it can be integrated into C or FORTRAN applications by means of standard input/output.
- A model can also be exported ready to be used in a HIL environment, where it can be integrated into C or FORTRAN applications by means of standard input/output.
- Moreover, with the PROOSIS toolbar for Ms Excel, models can be directly integrated into stand-alone spreadsheets.
- It is also possible to interact from MATLAB with PROOSIS models. That’s why, the model can be exported as an encapsulated S-Function block, which means it is possible, for example, to develop the engine with PROOSIS and the control system with Simulink.
- A model can also be exported ready to be used in a HIL environment, where it can be integrated into C or FORTRAN applications by means of standard input/output.

**APPLICATION FIELDS**

The existing PROOSIS application libraries make it useful for a wide range of applications:

**Engine performance simulation**

PROOSIS has the TURBO library for the modeling of gas turbine systems. It includes most of the components used in any gas turbine system, thus making it possible to simulate any complex configuration.

The flexible object-oriented based modeling approach allows extending the library with new components or inserting new functionalities.

Together with TURBO/PROOSIS, the included GT sample libraries of models and calculations offer a ready to use simulation environment.

Please refer to the TURBO/PROOSIS brochure for more information.

**Integrated engine/aircraft systems simulation**

Several off-the-shelf libraries are available for the simulation of other stationary and aircraft gas turbine related systems, as Organic Rankine Cycles, fuel and lubrication oil fluid loops and their thermal integration with other systems, as well as electric systems (electric machines, power electronics, electrical networks) and environmental control systems.

Please refer to the PROOSIS Aircraft Systems brochure for more information.

**Power Plant simulation**

Several off-the-shelf libraries are available for all sorts of analysis of industrial gas turbines and power plants (transient and steady state as well as thermal balance).

Please refer to the PROOSIS Power Plants brochure for more information.

**Other applications**

- Space propulsion systems: cycle design, analysis and dynamic simulation of liquid, solid and hybrid rocket engines.
- Environmental control and life support systems in manned spacecraft.
- Cryogenic systems: dynamic simulation of large cryogenic installations.

Level 1: Modeling new components libraries

Level 2: Simulation results visualization

Level 3: Graphical modeling of complex systems

Level 4: Simulating a PROOSIS engine model from Simulink.