Today’s CAE cycles involve a great number of standardized processes which, given a certain input, must produce a result that meets a set of requirements. Process automation capabilities of ANSA can accurately capture any pre-processing task making any quality criteria and requirements inherent to the process. As a result, the productivity of engineering teams is improved and the quality of models is assured, leading to a drastic reduction of the CAE turnaround time.

Task manager

At the heart of the integrated process automation lies the Task Manager, a tool where all individual steps of the development of a simulation model are included in the form of Task items. Tasks in Task Manager are built up by analysis experts and reflect all the distinct actions that must be performed for the set-up of a high-quality simulation model. Once a Task is completed, it can be saved as a template process and can then be used by inexperienced users to generate the required analysis models, promoting knowledge transfer within the organization.

Tasks are not bound to any particular set of data, thereby enabling the repeating use for similar analyses on different models. Additionally, advanced tools such as the Parts Representation Manager, Batch Mesh Manager and Connection Manager are orchestrated by Task Manager, forming seamless processes that cover the complete pre-processing workflow. Task Manager is capable of tracking dependencies between modeling actions, assuring that, whenever a change takes place in model parameters, all implicitly related definitions are properly updated.

Scripting

The BETA Scripting Interface is an Application Programming Interface that is an extension of the Python programming language. An ANSA script is a Python script. It can be used to extend the core functionality of ANSA. It can automate procedures of CAD data translation, meshing, boundary condition definition and model output. A rich library of functions enables the creation and manipulation of model data. Custom graphical user interfaces can be generated with the aid of a Graphical User Interface designer. Additionally scripting can be used to generate a big number of capabilities such as custom connection models and tailor-made model checks. The Python Programming language allows the use of numerous external libraries (i.e. mathematical, scientific) which offer great capabilities in the hands of the user.

Features

- Python programming language
- Customization of core functionality
- Complete process capturing
- Process templates
- Customizable Graphical User Interface

Benefits

- Process standardization
- Action-dependency tracking
- Model quality assurance
- Automatic model adaptation to changes
- Reusability
- Repeatability
µETA provides the analyst with a unique range of powerful tools to set up automated 3D and 2D post processing tasks. Special action based language (session), Python programming and user defined toolbars can boost productivity offering post-time reduction, limitless automation capabilities and reliable repeatability.

Manual post-processing is recorded, parametrized and executed through intuitive GUI without the need for text editing. New processes can be built up inside editor tools with specifically developed functionality for online help and debugging. Toolbars can be shared on group or company level while processes on local data can easily be packaged and transferred.

Session files

µETA is a GUI operated software that also supports and assigns commands to all actions, so that everything can be recorded in text files and be reapplied sequentially without the need for user interference. The Session Editor, a tool integrated in the software, facilitates the creation, editing and parametrization of session files through error checking, coloring of the command arguments, identification & listing of argument types (e.g. filenames), and debugging. Furthermore, sessions with local data used by this can be packed together in order to enable their easy transportation. Finally, sessions can also be executed in batch mode without the need to have GUI.

User toolbars

Better organization of work and time efficient post-processing can be achieved through the use of toolbars. All necessary analysis-oriented commands can be gathered inside a single interface and assigned to intuitive gui entities including buttons, textboxes, radio buttons and many more. A tool is available for the interactive creation and designing of toolbars, whereas storage location can be controlled in order to manipulate the level of common use.

Features

- Automation of 3D & 2D post processing tasks
- Easy set up & parameterization
- Editors for the easy creation of sessions / scripts
- Custom GUI
- Direct access to all simulation results
- Full reporting automation
- Sharing of processes among group/company level
- Stiffness calculation toolbar
- Pedestrian toolbar
- Bore Distortion Analysis toolbar
- Optimizer Setup toolbar
- Laminates toolbar

Benefits

- Time-efficient post processing
- Effortless realization of frequent tasks
- Fast model comparison
- Feasibility of advanced procedures
- Automatic results reporting
- Minimizes cost and time to market
- Novel features lead to results faster, while ensuring efficiency and quality.
Script files

In order to automate more advanced and complicated procedures that cannot be achieved through session files, script files -text files with the common in ANSA and μETA Python programming language- can be used. The capabilities of scripting are practically limitless, as everything inside μETA can be accessed, used and modified and complex programming structures can be set up. Apart from single script execution, the user can call a script inside μETA tools, whenever an advanced manipulation of data is needed, for example inside an annotation's text, in a 2D plot legend, in mathematical operations or in the statistics tool.

Further script features include the reading/writing of ASCII and binary files, the direct session commands incorporation, and the extensive interface customization for data input.

The Script Editor, a tool developed for the creation and editing of script files, also provides error checking, debugging functionality and a full library of script functions with online help and ready to run examples.

Variables and built-in functions

μETA includes tools in order to aid the easy retrieval and storage of values or other data. In the Variables tool the user can define and manipulate all available variables, static or dynamic, system or user defined, whereas in the Built in functions tool data from all entities can be retrieved through an intuitive GUI. Moreover, inherent μETA variables are automatically created to pinpoint significant session states.

Automatic procedures’ execution

In μETA the user can define some actions to be applied at each change of state/step/subase.

Furthermore, when the advanced filter tool is used inside a focus command, an identification or an annotation, the tool can be locked so that the filters are reactivated every time the state changes and the respective action is applied on the new filtered entities.

Tools associativity

The interoperability among μETA’s major automation tools can offer users great flexibility. Session parts can be combined with script parts, sessions and scripts can be called from other sessions or scripts. The session editor can be called to edit commands behind a user toolbar button. A user toolbar can be created directly from the session editor. Script code can be assigned directly to a toolbar entity.