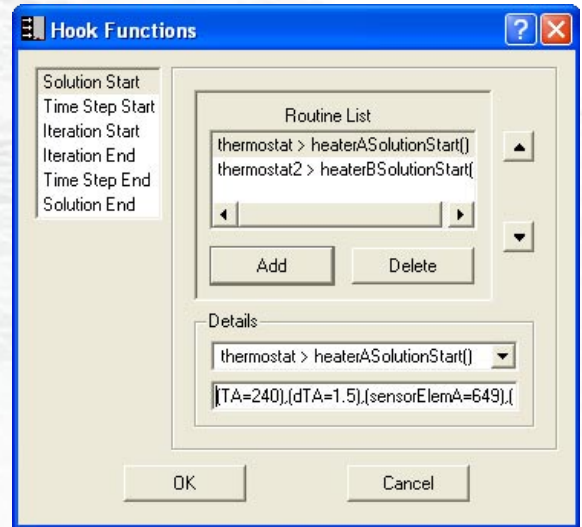


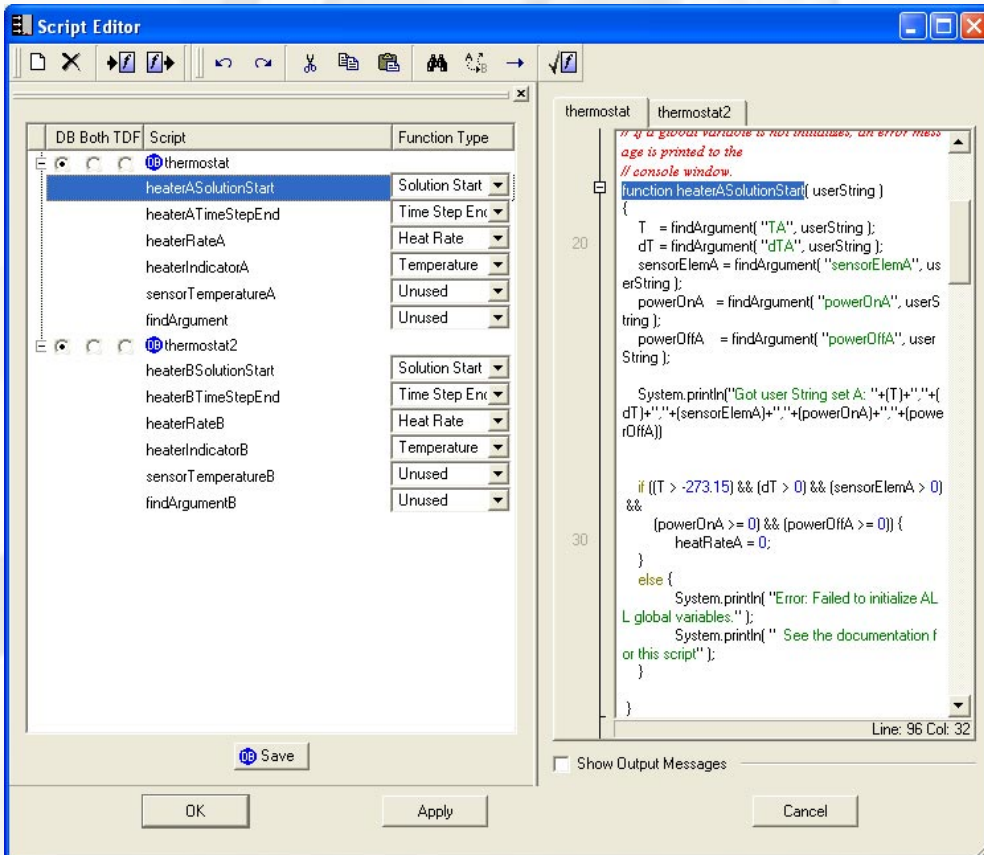


ThermoAnalytics Software Feature Focus: User Routines

Version 7.1 of ThermoAnalytics' software RadTherm (including IR and RT) and MuSES Pro includes support for user-defined scripts and routines. Your team can create custom functions to affect the solution or export data to a file. Where in the past you could specify a value or curve, you now can control such parameters with outputs of a script or routine. Furthermore, with Hook Functions you can access node-level data during the solution, and utilize that data in the routines. For example, at each time step you could check the temperature of two nodes, and impose a heat rate based on the temperature difference, similar to a PID controller. Spatially and/or temporally varying boundary conditions can also be imposed through scripts, allowing you to simulate movement through manufacturing operations like heat treatment or welding. There is no limit to the creative potential your team can exercise when developing scripts and routines - adding greater value and accuracy to your engineering analysis.

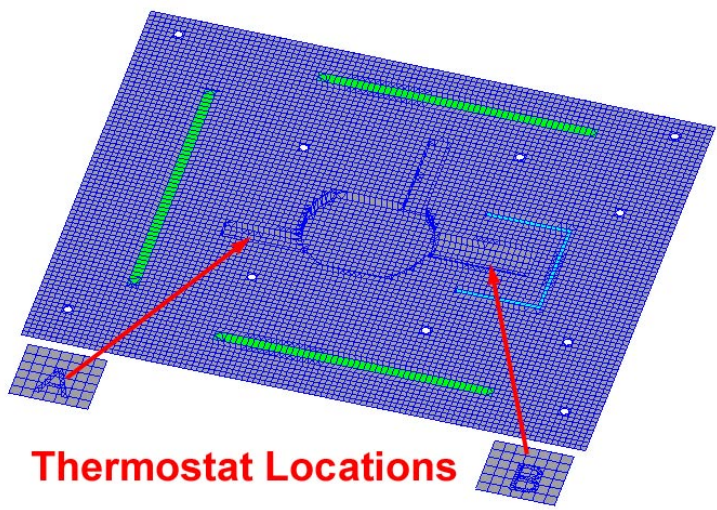


Hook Functions can be executed at the start/end of the solution, timestep, or iteration, allowing the user complete access to the solution process.

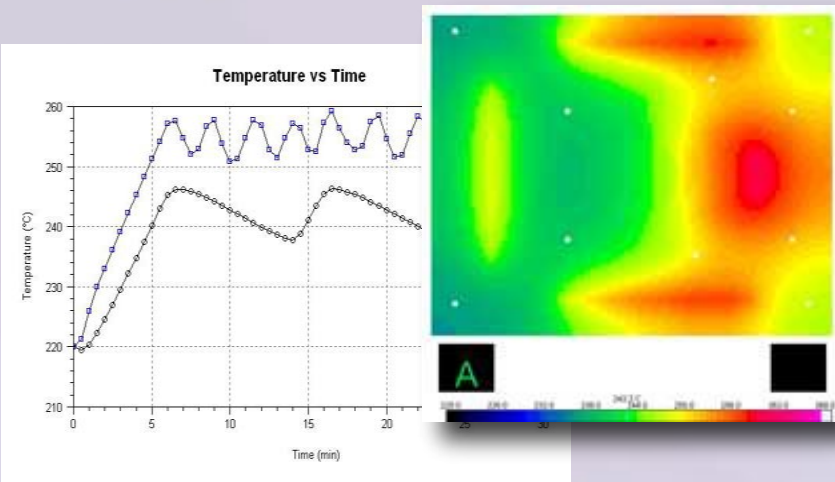


The built-in Script Editor window in Version 7.1 is a comprehensive script development tool. Functions, comments, and arguments are colored for easy editing. The function type (when it should be called during the solution) is set by a simple pull-down menu. Over 80 internal data sets can be accessed through hook function calls and data can be written to external ASCII files for use in other analytical codes. Scripts can be imported and exported to file. Common mathematical functions are supported, allowing your team to develop complex relationships within the thermal analysis environment.

Feature Focus: USER ROUTINES



Thermostat Locations



Internal Scripts VS Compiled Routines

User-defined functionality can be implemented in two ways with Version 7.1: through internal scripts or external compiled libraries. External libraries are compiled functions from FORTAN or C++ that are called from within our software. The major advantages of compiled functions are speed and security. They execute faster than internal scripts and the source code of the executable is not visible to the user. This allows your team to develop a proprietary library of functions that can be used by others, but the source code remains inaccessible and cannot be edited.

Scripts have reciprocal advantages. They can be shared in complete form and edited by the end user. This allows your entire team to modify the functions as required to solve their engineering problems. Furthermore, no compiler is required to develop and execute internal scripts.

Example: Thermostats

A simple example of a user-defined script is a thermostat. The example shown here utilizes two thermostat scripts that independently control two heaters embedded in an aluminum plate. The thermostat script reads a user string at the solution start to obtain the sensor element number, the therostatic centerpoint temperature, allowed error, and heater power setting for the on/off states.

At the end of each timestep, the script is again called and the temperature of the sensor element is checked to determine if the heater should be engaged. The output of the heater script is a wattage that is imposed onto the heated area of the plate. The heated thermal nodes are the interior nodes of a 3-layer part. This allows accurate simulation of embedded heating elements. The transient heat-up time and cycling of the heaters is accurately captured by the simulation. This model and scripts are available for free download at the following URL: www.thermoanalytics.com/applications/mom/mom-05-2004/

