**Running ResourceSubstitutable**

The ResourceSubstitutable model is written in Lazarus/Free Pascal. The latest version is available on github (<https://github.com/bkwiatkowski/ResourceSubstitutable>). ResourceSubstitutable can be run from the Microsoft Windows executable or by compiling the source code with the Lazarus IDE.

# Lazarus IDE

Lazarus is an open source, cross platform IDE. The latest version is available from the homepage, <https://www.lazarus-ide.org/index.php>.

# File Types

The ResourceSubstitutable model uses three types of files.

**Parameter Files (.par):** The parameter file contains the **initial state variable values** and the **parameter values** for a specific simulation.

**Driver Files (.drr):** The driver file contains the **external driver variables** for the simulation.

**Output Files (.out):** The output file is created by the model when a simulation is run. It contains the **state variables and process variables** for the entire simulation.

# Setting up a model run

The ResourceSubstitutable model opens with a green window (Fig. below). To run a simulation, you first specify file names for the parameter, driver, and output files. To open an existing file, single click on the “Parameter File” or “Driver File” label on the left application window; a dialog box will open allowing you to select a file. The values in the file will be read into memory. If the parameter and driver files do not exist, you can create them from within the model (see below). Output files are rewritten each time you run the model. You only need to specify the output file location and name; you can enter the path/filename directly into the text box or use the dialog box by clicking on the “Output File” label and entering the file name.



Click on the text “Parameter File” to open a dialog box to select the parameter file

**Creating and editing the parameter file.** The parameter file includes values for both the parameters and the initial values of all the state variables. Parameter files are space delimited text files. The easiest way to create them is through the model. To enter new initial state variable values or change values read from the parameter file, select 'Edit' from the main menu and then 'State Variables...'. An edit window will open where you can enter the initial values for each state variable; when done, click “OK”. Similarly, the parameters can be entered/edited by selecting “Edit” and then “Parameters…”. At this point the values you entered are stored in memory only. To save them select “Files” from the main menu, “Parameter File >” from the pull-down menu, and either “Save” to update the current parameter file or “Save As” to create a new parameter file.

**Creating and editing the driver file.** The driver file includes values for all the driver variables and how they change through the simulation. Driver files are comma separated text files. They can be created through the model, a spreadsheet program, or any text editor. To enter the driver values from within the model, select 'Edit' from the main menu and 'Drivers...' from the pull-down menu. An edit window will open with the current values from the driver file if a file was previously selected. The edit window is a basic text editor. If you are creating a new driver file, the first two lines of the file will be generated for you. The first line has the name of the driver variables (i.e., time and the drivers). The second line has the units for each of these variables. Each subsequent line must contain a value for time and each of the drivers, in the order listed in the first row. However, you only need to specify these data for 1) the initial values of the drivers and 2) for time steps when the drivers change. For example, if the drivers remain constant throughout the simulation, then only one line of data is needed specifying the driver values at time one. Alternatively, if you wanted to simulate an instantaneous change in year ten, but otherwise maintain everything constant then you only need to enter two lines of data. In the first would be a 1, for time one, followed by the initial values for all the drivers. In the second line would be a 10, for time ten, followed by values for ALL the drivers, even if their individual values have not changed. The drivers can change as many times as needed during a run. You can even specify the drivers for every time step of the simulation. In this case it is often easier to create the driver file using a spreadsheet program.

The drivers can also be ramped from one value to another over a specified time period. To create this ramp, specify the time and drivers for the beginning and end of the ramp as above, but put a minus sign (-) in front of the time at the end of the ramp. The model will interpolate linearly between these two end points.

Make sure to save the driver file when you finish editing it. Unlike the parameters and state variables, the driver variables must be saved to the driver file before running a simulation.

**Run the model.** Once you have entered initial state variables and parameters and have specified files for parameters, drivers, and output, enter a “start Time” (usually 1) and a “Stop Time” for the simulation. Then click on “Run”.

# Viewing model output

The model output includes the driver variables, the state variables, and values for all the processes for each time step of the simulation. The output file is a comma separated text file. It can be opened in any spreadsheet program, or you can view the output and create simple graphs from within the model by clicking “Output” on the main menu, then “Chart” from the pull-down menu. When the window opens, there will be a blank graph at the top and a list of all the output data at the bottom. Select the variables to plot by clicking on them (they will then be highlighted). As many as ten variables can be plotted at the same time. Once you have selected the variables you want to plot, simply click on the 'Update Chart' button. You can unselect a variable by clicking on it again. There is also a 'clear selection' button to unselect all variables.

# Rerunning the Model

At the end of a simulation the values of the state variables in memory are the value at the end of the simulation. To rerun the model, you can reset the state variables to the values stored in the parameter file by using the 'Reload' button at the lower right of the main window. You have the option to reload the state variables, the parameters or both by checking the appropriate checkbox. If you made changes to the parameters and did not save the parameter file after making the changes, reloading the parameters will erase those changes.