

Configuring UEStudio or UltraEdit Integration for RSM

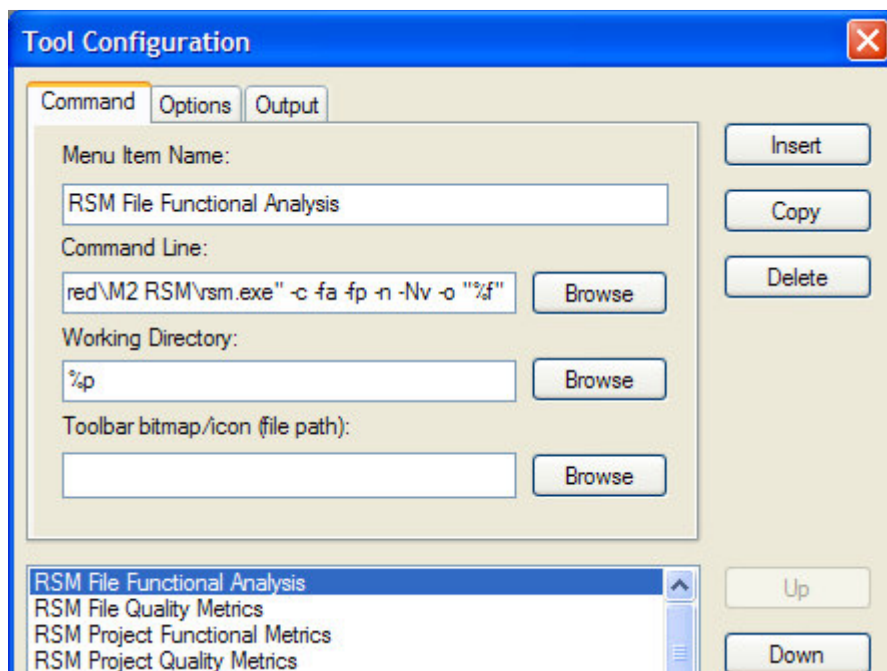
This article describes how to integrate RSM into the UEStudio or UltraEdit IDE. It will also show how to handle just the current file or an entire project. It does not aim to fully explore the options as each company and developer will choose which metrics to use and in what combinations. But the technique shown here can be used regardless of the actual metrics selected.

The IDE toolbar has an advanced menu entry on it. This is where the custom tool components can be loaded and configured. The Tool Configuration... entry is used to add, delete and configure the custom tool components.

Click on the Tool Configuration... menu item. This will show the configured items. The lower section might be completely empty if you haven't done this before. Click on the 'Insert' button in the top right corner to create a blank entry. This allows you to set the name for this item, define the command line to be executed, specify the working directory and also a custom icon if you want. I left the last item blank and so concentrated on the other three fields.

Analysis of the current editor file

Here is what the Command tab looks like for our file functional analysis. This performs an analysis on the file currently selected in the editor.

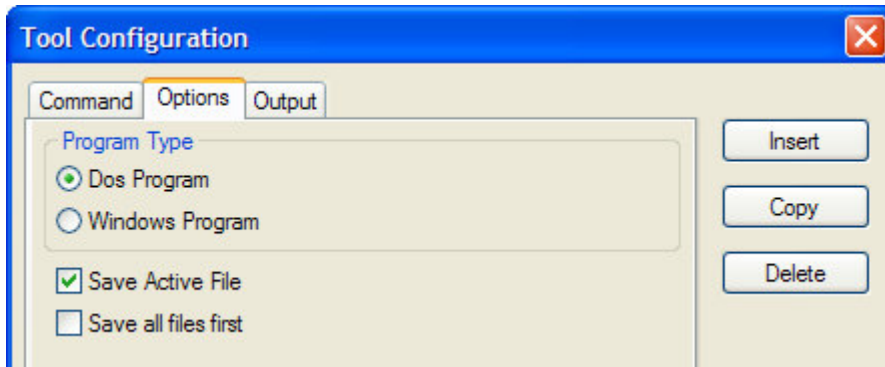


I chose to put the full path to RSM in as this avoids any issues associated with the order of directories in the path.

The flags afterward can be whatever you want them to be for your own notices. The "%f" at the end is translated into the full path and filename and so should be enclosed in quotes.

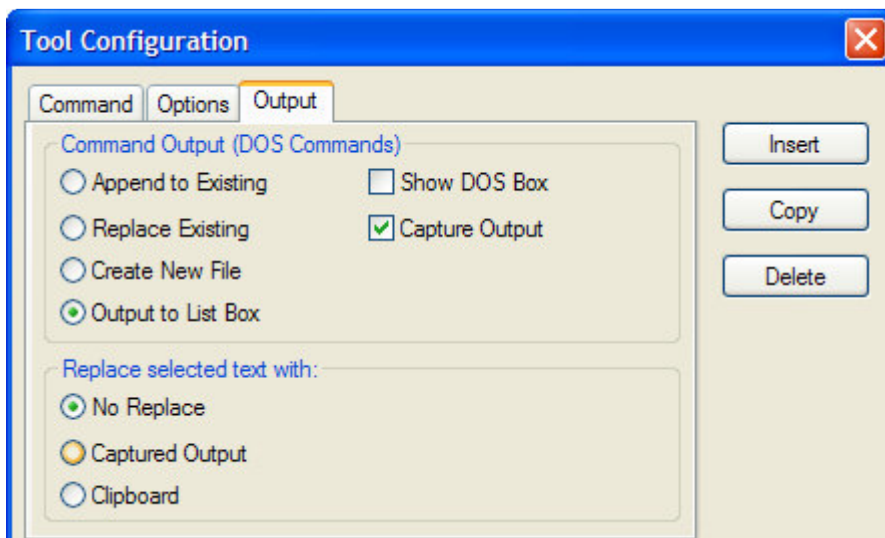
The Working directory is set to %p which means the same directory as the file is in. This is not so important for the file analysis but can be very important for the project level analysis if you do it the same way we did. Like many of these things, there are many options so we are just showing you one way that we know works.

Now there is the Options tab to look at.



This ensures the file is saved first since the analysis will be done on the file on disk, not in the editor window. It also tells the IDE that the program is not a windows program and so the output can be captured from standard streams.

Now for the Output tab.



The output is captured and sent to the list box which is the text box at the bottom of the IDE. The final section allows configuration for tools which place their output on the clipboard or run multiple passes and you want to capture each pass and add it onto the existing text. We chose to just overwrite everything and the No Replace option achieves this. So the text box will only contain the last analysis.

The particular flags we chose on the Command tab also mean the output is formatted so you just double click on the notice and it takes you to the line in the file that generated the notice. This is, of course, very convenient and speeds up the review process considerably.

So that's it. Your first RSM integration is done.

Next we will show you how we handle the project level version of this.

Analysis of a full project

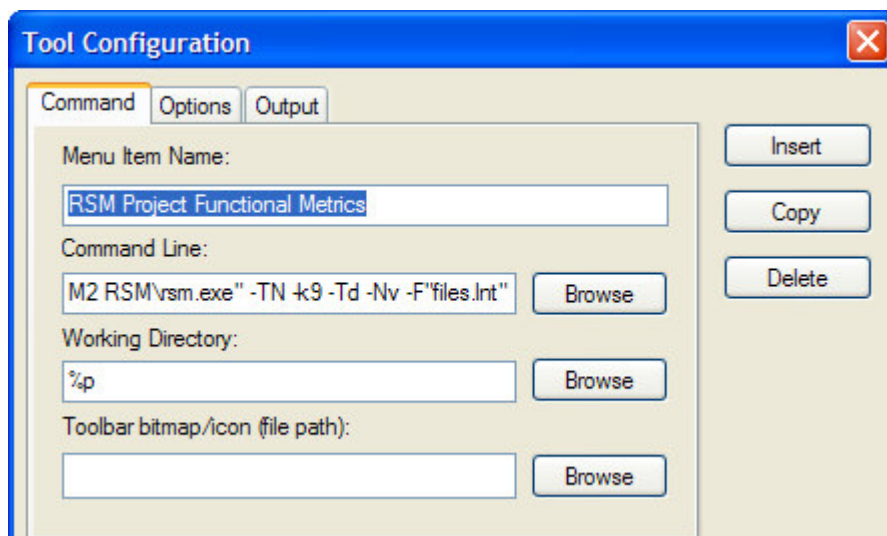
For a full project, we elected to put the list of all the files into a file. This means the same Advanced menu item will work with any project.

This can be done a couple of ways but since we also use other programs and they have more specific limitations than RSM we selected files.Int as the filename. However you can use any reasonable name you like such as files.lst, files.txt, ProjectFiles.lst etc.

The key thing is that the list of files must be in the same directory as an editor file you have open. You can have files spread out in a complex structure but the file list must be in the working directory.

The Options and Output tab are the same as for a file so only the Command tab is different. So we won't bother repeating them here. You can use the Copy button to copy an existing configuration so that also saves some typing.

Here are the Command tab fields.

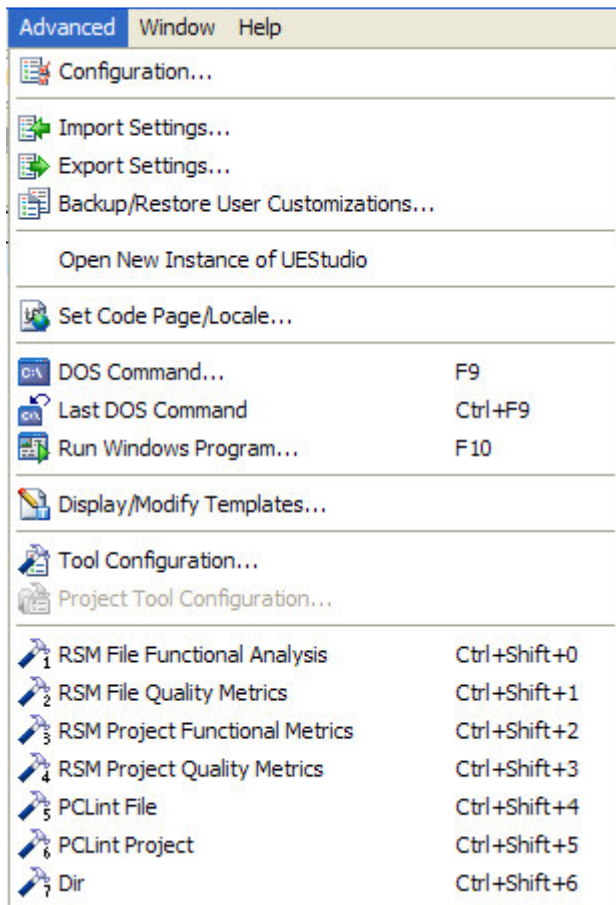


So this time the -F tag is used to indicate the files to analyse are in a file. Again we used the full path for RSM.

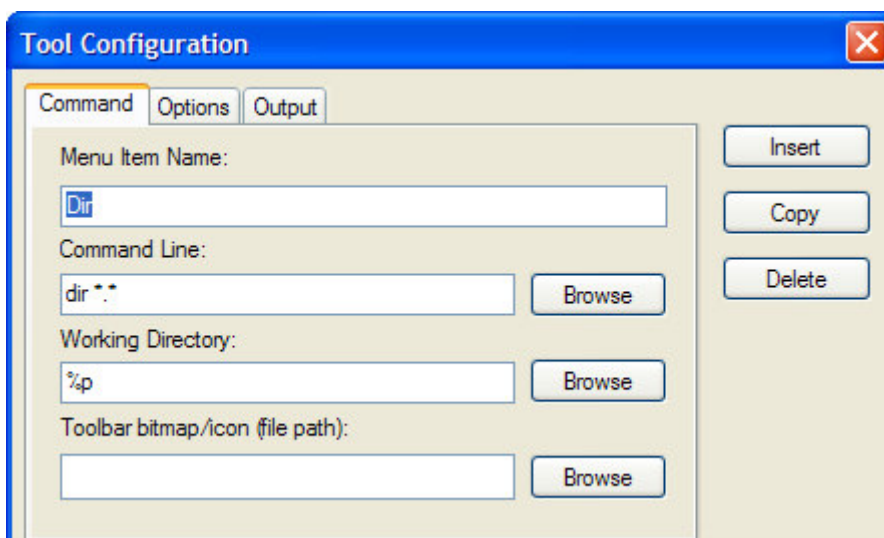
You can even open the files.Int file in the IDE and hit the project button menu entry. The output appears in the text window at the bottom of the IDE and the format has been set so double clicking on a notice opens the file and takes you to the line the notice refers to.

Putting it all together

So once we defined the quality reports we wanted to run we ended up with the Advanced menu looking like this:



The final item makes it easy to build a list of files. The configuration for it is similar to the other ones already reviewed except a bit simpler. Again, the Options and Output tabs are the same. The Command tab looks like this.



So this makes a directory listing of all the files in the directory of the currently selected editor file and puts it into the text window at the bottom of the IDE. Now you can copy and paste them into the files.Int file (or whatever you selected as a suitable name). Start with a new blank document if you are building the list for the first time. As the project grows you can add file as you need to. Since the command doesn't care which directory you are in then it is very flexible. You have to build the paths yourself if there is a structure involved.

Alternatively, if you are using a make file you can grab the items from that.

And of course you can be more specific if you only want a particular file type.

More advanced options

There are some excellent tutorials available on how to take the custom commands and place them as icons on the IDE toolbars and add other key mappings. If you wanted to take this further then check out the tutorials available at the UltraEdit site.

http://www.ultraedit.com/support/tutorials_power_tips.html#ues and

http://www.ultraedit.com/support/tutorials_power_tips/ultraedit/user_and_project_tools.html

Wrapping it up

So that looks pretty simple. It took a few different tries at options to select this as one that met every project need we had and like most of these activities the final outcome looks a lot more straight forward than the process that got us here.

We hope you find this useful. Don't hesitate to drop us a line if you need any further explanations.

Ray Keefe

Successful Endeavours Pty Ltd

71 Sydney Parkinson Avenue

Endeavour Hills 3802

Australia

+61-3-97004049

<http://www.successful.com.au>

raykeefe@successful.com.au