RDi Tidbits and Goodies

Note:
WDSC = RDi = RDPi

by Aaron Bartell
of Krengel Tech

Copyright © Aaron Bartell 2010
aaronbartell@mowyourlawn.com
Abstract

Have you been using RDi for awhile and are looking for the little **tidbits** to make you even more productive? Everything from modifying the max memory allowed to learning the best font for viewing RPG code to the best plugins! Come and see what I love best about RDi and how to customize it to best suit a programmers needs. If you have a tip for your fellow RDi RPG programmer please bring it with to this session and let your voice be heard!
Service Entry Points allow you to “mark” a program for debug. When the program gets invoked a debug session will automatically start in RDi.

This is incredibly cool because you don't have to know the job number ahead of time, it simply watches for the program to run under a specific user profile!

It is one of those things you have to see to believe, so I created a video tutorial...

See Service Entry Point video tutorial here:

http://tinyurl.com/2m3ee7
If you need to view more memory than what the default Monitor Expression shows (i.e. 1024), an alternative is to use the Monitor Memory option which allows you to keep scrolling through memory to your hearts content!

To use this simply highlight the variable whose content you want to view, right click, select Monitor Memory->EBCDIC.
Optimal font for viewing RPG code in LPEX editor is Lucida Console 8pt. To change this go to Window->Preferences->LPEX Editor->Appearance. This allows you to see many more lines of code at a time while still giving you a pleasant viewing experience.

Note that you can gain even more space for viewing code by making a few more modifications. Go to the Window->Preferences->LPEX Editor->Controls and deselect Status line, Message Line, and Command line.
We are all able to code in /FREE form right? As you are working in those fixed form programs it would be great to be able to easily convert code to /FREE form. RDi does this with great ease by simply highlighting the code you want converted, right click on the code and select Convert Selection to Free-Form.

### Before

```plaintext
C      EVAL      Pos = %SCAN(InpSch :InpStr :Str)
C      IF        Pos = *ZERO
C      EVAL      OutStr = InpStr
C      ELSE
C      EVAL      OutStr = InpStr + %SUBST(InpStr :Str :Pos - Str) + InpRpl
C      EVAL      Str = Pos + %LEN(InpSch)
C      EVAL      // Enough left to search
C      IF        Str + %LEN(InpSch) <= %LEN(InpStr)
C      EVAL      Pos = %SCAN(InpSch :InpStr :Str)
C      ELSE
C      EVAL      Pos = *ZERO
C      ENDSL
C      ENDDO
C      // Tack on trailing characters
C      IF        Str <= %LEN(InpStr)
C      EVAL      OutStr = OutStr + %SUBST(InpStr :Str :%LEN(InpStr) - Str + 1)
C      ENDSL
C      ENDIF
C      RETURN
```
Code Templates work great if you find yourself repeating the same section of code multiple times.

To view existing Code Templates go to Window->Preferences->Remote Systems->Remote Systems LPEX Editor->iSeries Parsers->ILE RPG->Templates

Use Ctrl+Space to bring up the window to select a template.

Templates can be exported and imported so they can easily be shared amongst other developers.

Only works when doing Ctrl+Space between /FREE and /END-Free.

Produced from DOW_loop template
Snippets

Alternative to code templates. Allows you to store snippets of source that aren’t meant to be used with Ctrl+Space functionality. To display this view select Window->Show View->Other->General->Snippets. By default there are no iSeries related snippets. Right click anywhere on the view and select Customize. Click New to add a Category and name it RPG.

Once a category has been created you need to create Items within.
Name the item and give it variables with names to be occupied when this snippet is inserted during development. Paste in the code you want to be involved in the snippet (in this case a skeleton of an RPG sub procedure).

Place your cursor in the Template Pattern area where you want one of the variables and press Insert Variable Placeholder.

Last, place your cursor in a source member and double click the new item you just created. It will prompt you for the parameters and insert the snippet into your program – rock on!
I used to go to the command line before I found this feature. If you are unsure of a command simply type the initial characters followed by an asterisk and hit F4 to prompt. Select the command you want and then you will be prompted for that commands values.
Ever wonder where the ending to a particular IF statement is because there are multiple IF’s embedded? Or maybe the IF statement spans many lines of code. Simply place your cursor at the beginning of a control structure (i.e. SELECT, DOW, DOU, IF, FOR, etc) and press Ctrl+Shift+M and your cursor will be place at the corresponding end of that control structure.

Continue pressing Ctrl+Shift+M to alternate between the two locations!

With SELECT statements you can also place your cursor on a WHEN statement and press Ctrl+Shift+M to go to the next WHEN statement.

Also, using Ctrl+M will highlight the entire code section.
Using the SEU exclude block characters (i.e. XX) you can hide a chunk of code. You can do this in SEU so no big deal, but WDSC adds the ability to easily expand that section of hidden code by placing a plus sign next to the hidden section.

Expandable sections are also enabled when you right click anywhere in an RPG source member and select Filter view->Comments (or any of the other options from within Filter view).
If you want to easily see all the places where a variable is used within a program simply highlight the variable and click Selected->Filter Selection.

Note the plus (+) signs off to the far left. Expand these to see the code immediately following the line where the variable was found.

To revert back to regular source viewing select Ctrl+W (Show All).
RDi allows you to have two windows open in edit mode for a single source member! This is a big advantage over SEU as we only have browse capabilities if we want to display a member through a second editor. As you type in one editor it immediately shows up in the other!

To bring up the dual edit mode simply select Ctrl+2 in the editor you would like to open a second time. To close the second window simply right click and select View->Close View.

Both open for edit!
The Find and Replace capabilities in the LPEX are second to none – even when you consider the Java editors equivalent. The best part about the Find/Replace in LPEX is it doesn’t open up a dialogue box but simply places a small form at the bottom of the editor to enter parameters.

Note that regular expressions are also a feature here which makes find/replace capabilities incredibly fine tunable.
The Tasks view displays tasks that you add manually. You can associate a task with a resource in the Workbench (e.g. RPG source member), but this is not required. If Task View is not open go to Window->Show View->Other->General->Tasks.

Double click entry to go to location, even if the source isn’t open!!
The more I develop with a modular mindset (i.e. RPG *MODULE’s and *SRVPGM’s) the more I find my /COPY statements growing. To easily view a copy book’s contents simply place your cursor on the /COPY line, then right click and select /COPY Member->Edit/Browse.
Traditionally whenever I need to see the structure of a physical file I simply use green screen command line tools like LSTFFD (from www.BVStools.com), but if you are in RDi you can simply right click on a F spec and select Show fields. Note that the file must be compiled. The iSeries Field Table View will then be opened (if it isn’t already) and populated.
I do all new development in /FREE format, but I still have to work with fixed format on a semi regular basis where I am not allowed to change it to /FREE. To make it bearable I frequently use the iSeries Indent feature to learn how a piece of code is organized.

To display this view simply select Source->Show Indentation from the RDi menu.

Note that this works for /FREE format code also in the event a programmer didn’t do a good job of indenting their code.
RDi has a good/fair source compare tool that allows you to see the changes from one source member to the next. A good scenario for this is source that is in development vs. testing or production. The reason it is only a good/fair tool is because www.Softlanding.com has one that doesn’t include the date area for comparison which is good, because if somebody does a re-sequence of the line numbers then ALL lines will appear changed. To use this simply select both source members from RSE (use Ctrl key for multiple selects), right click and select Compare With- >Each Other.
Softlanding.com’s Compare Source utility is quite nice in that it doesn’t include the date area when comparing. Notice the tick marks to the far right on the compare editor. This allows you to easily navigate to differences in the source.

To get the free RSE extensions from Softlanding go to this URL: http://www.softlanding.com/websphere/rse.htm
When you compile from RDi it will return a list of errors and when you double click on one it will insert a pink message right below the location of where the error is located. This is very nice for quickly resolving errors.

To remove the pink lines select Ctrl+F5.
Right out of the box the iSeries Error List shows ALL compiler messages including RNF7031’s. Most of the time these are of no concern and can be ignored. To remove them from the iSeries Error List select the downward triangle->Show Severity->Information (deselect).
If you have been playing with the RSE perspective by moving things around and closing views you may find that you have closed something you need. Sometimes it is hard to find an existing view through the Window->Show View->Other approach and it may be best to reset the entire perspective. To do this simply select Window->Reset Perspective.

Alternatively if you are at a point in organizing the RSE perspective and want to save it for ease of reference, you can do so by selecting Window->Save Perspective As.

To gain access to saved perspectives simply open it like any other attempt to open a perspective: Window->Open Perspective->Other->Aaron’s RSE
In a never ending effort to maximize screen real-estate I found that you can drag certain views off of the main RDi window as shown to the right. This is a really cool feature for those that have dual monitors! Note that if you have a laptop all you need to do is get an additional monitor and you can easily do this by using the software built into Microsoft Windows.
Fast views are hidden views, which can be quickly made visible. They work the same as normal views, only when they are hidden they do not take up screen space on the Workbench window. Fast Views are great when you want to maximize the amount of viewable source code.

To make a view hidden simply right click on the tab and select Fast View. It will then be placed in the Fast View area represented by an icon. To show a view that is hidden simply left click on the corresponding icon. To make it a regular non-hidden view again simply right click on the icon and deselect Fast View.
In previous versions of RDi it was not easy to switch from one workspace to the next. Starting RDi 6.x it is as simple as File->Switch Workspace. There are a variety of reasons you may want to switch workspaces. Mine simply relate to needing a clean environment for developing demo’s, or if I want to try out a new feature in RDi (i.e. iSeries Projects) I can do so without messing up my existing workspace. Maybe you create a workspace for each client you work with (for consultants).

I recommend keeping your workspace as close to the root of your hard drive as possible because eclipse tends to create long file and folder names that sometimes cause problems for Windows. I keep my workspaces in c:\sync\java for instance.
It can be a pain to traverse all the way through RSE simply to open a member that you know the full location and name of anyway. To bypass that you can use the Ctrl+Shift+A shortcut and simply enter in the library, source physical file, member and whether to open with browse or edit.
With WSDC you can easily search across multiple source physical files in the same library by using Search->iSeries from the top menu or by right clicking on a library/source physical file and selecting Find String. From the resulting Remote Search view you get a sample of each line where the search string was found and can access it by simply double clicking on that line which will open up the source member and place your cursor on that exact location – very nice!
The Link With Editor feature allows you to easily navigate back to the location within the RSE tree of an open source member. This can be a huge time saver when working on a project with many libraries and source physical files.
Sometimes a source physical file can contain many hundreds if not thousands of source members and it can be a daunting task simply to scroll down to the source member(s) you wish to access. Instead of manually using the scroll bar simply highlight the first member in the source physical file and start typing the name of the source member you are after. The highlighted source member will change as you continue to type up to the 10 character member name.

This feature is similar to what Windows offers in Windows Explorer.
With RDi’s ability to have many source members open at a given time it can be a daunting task to go back and forth between members – especially when you close a member and have to go back to edit something you forgot.

RDi offers two great buttons to ease this pain. Simply select the Last Navigation Location button (reverse or forward). It will bring to front the source that was most recently accessed.
Switching editors using the mouse can be annoying when doing heads down programming (yes I know you hunt and peckers are out there!). A solid alternative that keeps you on the keyboard is using Ctrl+E to bring up the list of editors.
Quick Marks

When you are at a certain section of code in a large program and want to move to another but know you are going to be back, you can use Ctrl + Q to set a quick mark, and then navigate to the other section of code. To get back to that quick mark simply press Alt + Q. And to go back to the second spot of code press Alt + Q again - very nice!

This space intentionally left blank
I really like the FireFox browser because of it’s non-intrusive ‘Find’ capabilities (i.e. no popup dialog to deal with like Internet Explorer). To change the default browser for RDi Help to be FireFox go to Window->Preferences->Help->General->Web Browser, select Firefox. RDi Help is brought up whenever you press F1 in RPG source code.
By default Microsoft Windows comes with a lot of eye candy that in the end costs precious CPU cycles. You can change from traditional XP look and feel and adjust for best performance. To change this go to Start->Settings->Control Panel->System, select the Advanced tab, select the Settings button in the Performance section. Select the Adjust for best performance radio button.
iSeries Commands

RDi has a nice alternative to heading to the green screen command line for certain commands. Open the iSeries Commands Log view and use the triangle to select a system connection. Then enter a command (e.g. CHGLIBL) and select the prompt button. If it is a supported command (i.e. doesn’t need to be interactive) then you will be presented with the commands input fields.
I have always liked being able to copy a block of code while in a green screen session and paste it elsewhere. In Windows editors this is usually a foreign concept, but the LPEX editor fully supports it as detailed here.

Place your cursor at the location where you would like to start the block copy and right click to Select->Select rectangle (or use shortcut Alt+R).

Then reposition the cursor to the location where you want to insert the block paste and select Alt+Z. Make sure you don’t double click at the location you want to paste as that will cause the selected text to be de-selected.
The “RPG Next Gen” Eclipse based IBM i source member editor is the new kid on the block that aims to compete with RDi. The tool is free and open source, and you can download it from www.rpgnextgen.com. Make sure to submit feature requests and issues you find on the same site.

- Small Download: 30MB
- Installs in less than 5 minutes
- Uses around 50MB of memory
- No cost
- Open source
- Site: www.RPGNextgen.com
- Try it out!
Duplicate the current line of code **CTRL + D**
Open split LPEX view **CTRL + 2**
Emulate right click in source **SHIFT + F10**
Close split LPEX view **SHIFT + F10, V, C**
Go to specific source line **CTRL + L**, enter line number
Go to command line area **ESCAPE**
Moves the screen up and down without moving the cursor **CTRL + Up Arrow** or **CTRL + Down Arrow**
Place cursor in sequence area **SHIFT + TAB** (once or twice)
Remove inserted compile errors **CTRL + F5**
Bring up Find section **CTRL + F**
Undo last edit **CTRL + Z**
Copy **CTRL + C**
Paste **CTRL + V**
Get list of all shortcuts **ESCAPE, enter help seu**
Use **Ctrl+Shift+L** to bring up ALL the short-cuts for any given view in Eclipse
www.midrange.com – WDSC/RDi mailing list

http://www.softlanding.com/websphere/rse.htm (free plugin)

http://wdsc.wordpress.com/ (RDi developers frequently post to this blog)


http://www.i5podcast.com/Training/WDSC/index.html
Questions?

Aaron Bartell

aaronbartell@mowyourlawn.com

Lead developer of **RPG-XML Suite (www.rpg-xml.com)**

Owner of **www.MowYourLawn.com**