



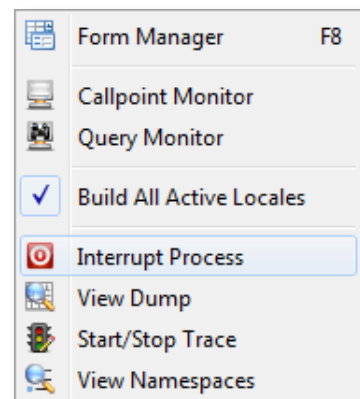
## New Ways to Debug Barista

**T**he Barista® Application Framework now includes several new debugging features that simplify your troubleshooting efforts in your Barista and custom callpoint code. Now armed with the ability to interrupt the process by pressing the [ESCAPE] key, you can debug and dot-step through the code! In addition, you can view the dump to see the contents of the workspace, and start and stop tracing. Lastly, you now have the option to

view the namespaces and the contents of the namespace variables. Read on for more details about these great new debugging features.

Listed under Development in the Barista MDI menu are the four new functions as shown in **Figure 1**; Interrupt Process, View Dump, Start/Stop Trace, and View Namespaces. Launching a standard maintenance form, grid maintenance form, or header/detail form automatically enables these new functions.

Let's take a look at how they work.



**Figure 1.** New debug options in Barista's development menu



**By Ralph Lance**  
Software Engineer

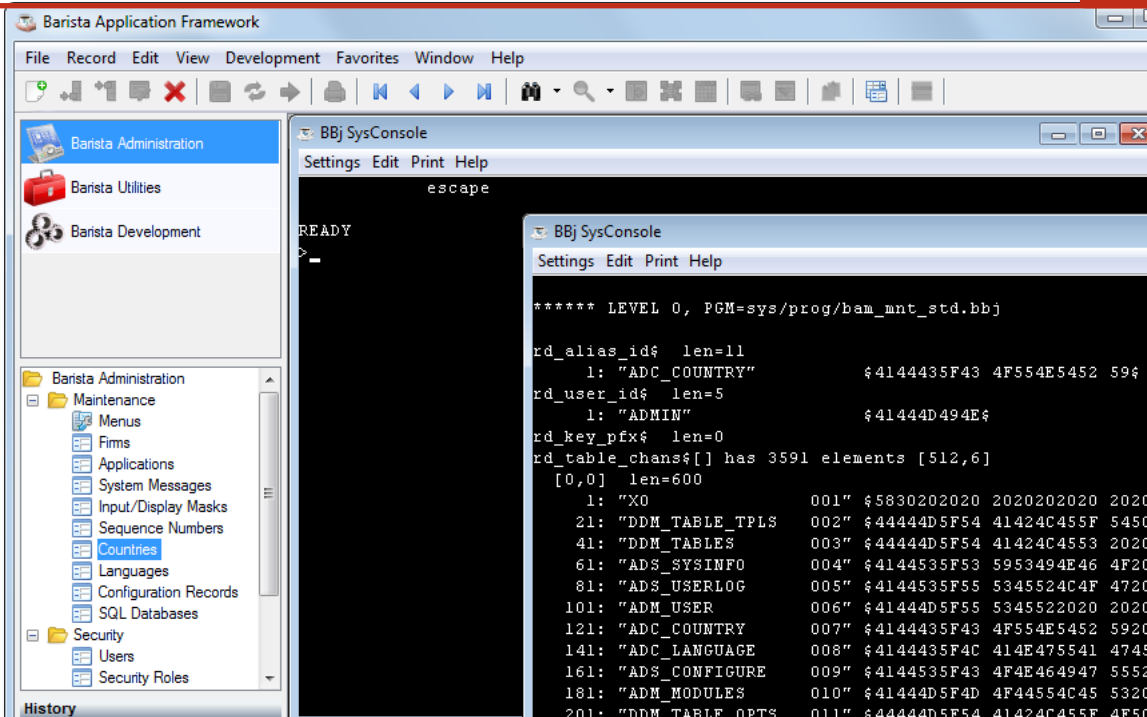


Figure 2. Interrupting a process in Barista

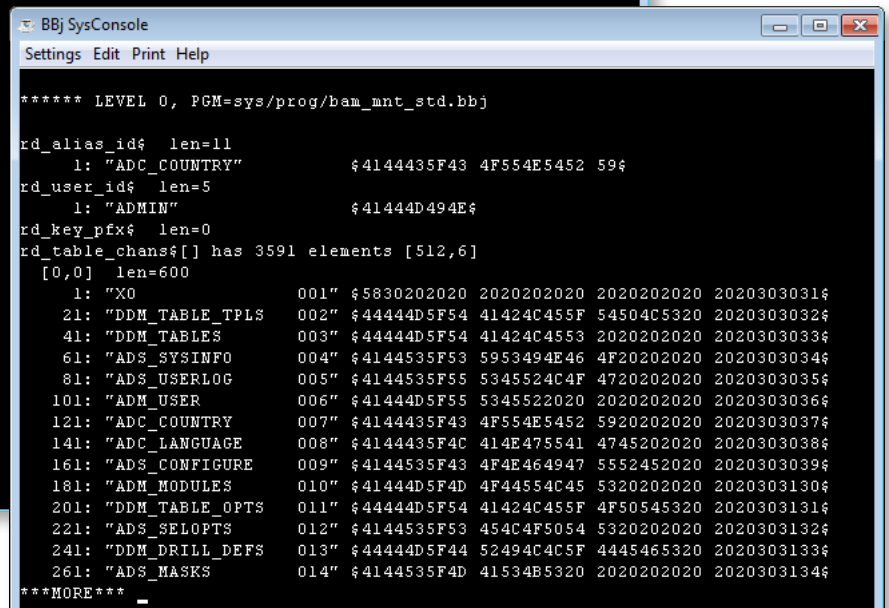


Figure 3. Result of a DUMP command to view the current workspace variables and their values

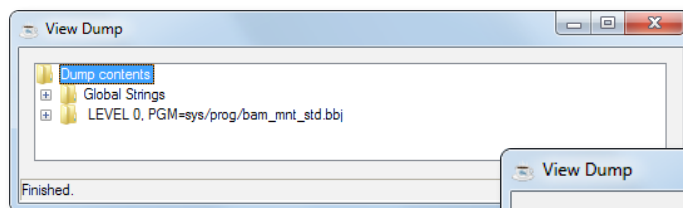


Figure 4. Result of Barista's View Dump menu option

## Interrupt Process

Interrupt Process sends an escape to the Barista process as shown in **Figure 2** and then falls to the BBj® SysConsole, if allowed.

From here, you can easily list non-protected programs, interrogate variables, execute dot-step commands, or perform a DUMP command (see **Figure 3**).

## View Dump

One alternative to reviewing dump data is to use the menu item View Dump, which presents the data in a developer-friendly tree with global strings separated from the program variable data (see **Figure 4**).

The global strings and program variables appear alphabetically to help find the item in question quickly (see **Figure 5**). Optionally, you can view the individual elements of array variables as well as the template and contents of templated strings.

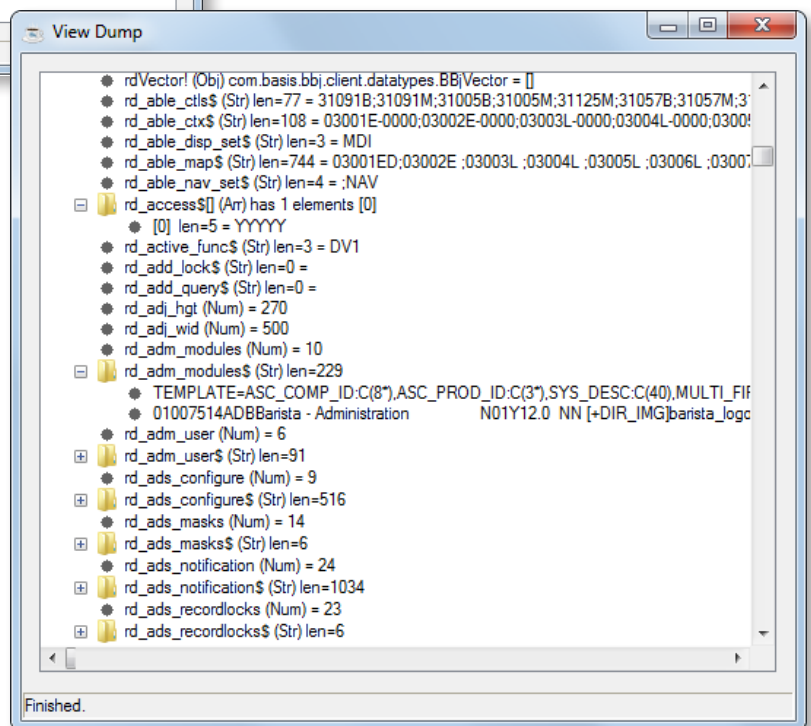
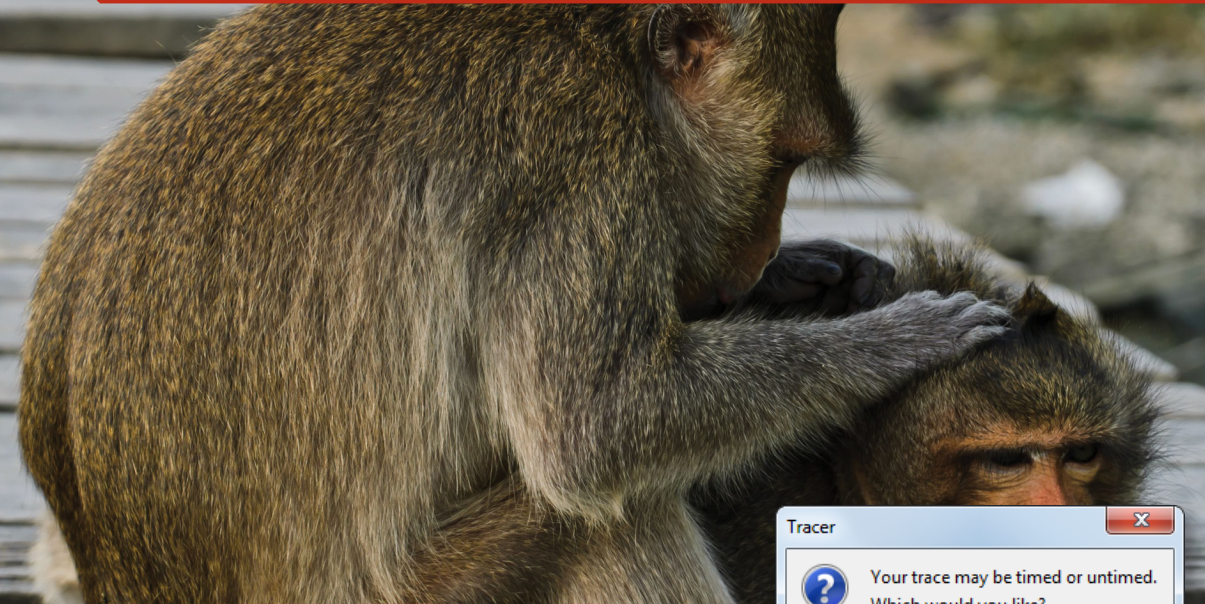


Figure 5. Expanded view of variables





### Start/Stop Trace

Starting a trace displays a dialog as shown in **Figure 6** in which you can choose whether the trace should be timed or untimed. The advantage of a timed trace is that you can use the BASIS Performance Analyzer to analyze the program trace file for potential optimizations. An untimed trace is handy for viewing a step-wise path through the code (handy for something like an infinite program loop, for example.) Barista names the trace file using a universally unique identifier (a "UUID" generated for us with a Java function) plus ".trc" and stores it in the Barista `/workarea` directory.

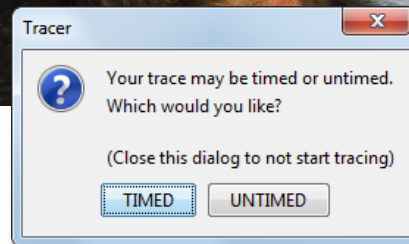
After tracing through the desired processing steps, choose the menu item again to stop the trace and respond to the dialog. If the trace was timed, then you may optionally choose to invoke the Performance Analyzer to immediately analyze the trace file as **Figure 7** shows.

### View Namespaces

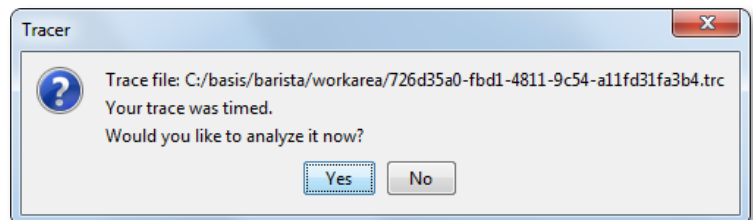
Similar to the View Dump option, View Namespaces presents a tree view of the GlobalNamespace and GroupNamespace folders, which expands to display the alphabetically sorted contents (see **Figure 8**).

### Summary

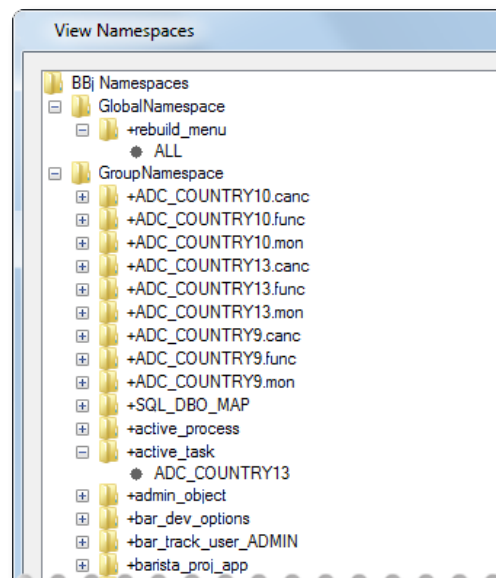
As robust a tool as the Barista Application Framework is, it is even more powerful and valuable to the developer with its new debugging features. You now have the tools you need to track and monitor your applications as they interact with the Barista Framework. Using Barista is better than ever so add it to your toolset and enjoy the easy and efficient way it can help you be more productive in support of your customers. ■



**Figure 6.** Starting a trace



**Figure 7.** Finishing the trace



**Figure 8.** Result of Barista's View Namespaces menu option



- Read *Tuning the Performance Analyzer* at [links.basis.com/05tuning](http://links.basis.com/05tuning)
- Find additional Barista resources at [links.basis.com/baristaref](http://links.basis.com/baristaref)