Target Control and Debugging of VxWorks 6.x Real-Time Processes using Rational Rose RealTime

Sreenivasulu Valmeti
Kinjal Shah
Kalyan Prasad

December 09, 2008

Introduction

IBM® Rational® Rose RealTime is a modeling tool that you can use to create models of software systems based on the Unified Modeling Language constructs, to generate the full implementation code, compile it, and then run and debug the resulting application. This article explains how you can perform the target control and debug operations of RTPs built with Rational Rose RealTime, using both the Rational Rose RealTime toolset and the VxWorks Workbench.

Traditionally in Wind River's VxWorks 5.x, there were only kernel mode tasks. VxWorks 6.x has introduced user mode processes called RTPs (real-time processes). Rational Rose RealTime supports the target control operations (such as run, terminate, debug, and so on) for RTPs built using Rational Rose RealTime. The run, terminate, and debug operations of RTPs are different from the traditional kernel mode tasks. For example, there are no load and unload operations for RTPs. VxWorks 6.x also introduced a new IDE called Workbench, which has an integrated debugger that enables you to debug more than one RTP at the same time.

This article explains how you can perform the target control and debug operations of RTPs built with Rational Rose RealTime, using both the Rational Rose RealTime toolset and the VxWorks Workbench.

About this article

This section gives a brief introduction to Rational Rose RealTime and Wind River VxWorks 6.x. It also discusses the advanced features of Wind River Workbench.
Finally, it shows you step-by-step instructions about how to use Rational Rose RealTime to perform Target Control operations on RTP applications that are built using Rational Rose RealTime.

Introduction to Rational Rose RealTime

IBM® Rational Rose® RealTime is a modeling tool that you can use to create models of software systems based on the Unified Modeling Language constructs, to generate the implementation code, compile it, and then run and debug the application. It is a robust Model-Driven Development (MDD) solution for system architecture design.

Rational Rose RealTime unifies the project team by providing an extensive set of tool integrations to meet the needs of the entire team, from requirements capture through to high-performance code generation and debugging for real-time operating system targets. The Rational Rose RealTime user interface looks like that shown in Figure 1.

Figure 1. Rational Rose RealTime user interface

Introduction to VxWorks 6.x

VxWorks historically provided only kernel mode, requiring you to develop exclusively in this mode. With VxWorks 6.x, Wind River introduced the Real-Time Process (RTP) models. The RTP model
introduces user and kernel modes demarcation for VxWorks applications. Application designs based on the RTP model will benefit from the high-performance VxWorks kernel, with the addition of reliable resource reclamation and memory protection.

**Introduction to real-time processes**

A typical VxWorks system consists of an operating system, tasks running in kernel mode, and a number of user applications, as shown in Figure 2. A user mode application is an Executable and Linking Format (ELF) executable and executes as an RTP. RTPs are isolated from the kernel that contains the VxWorks operating system, as well as from each other. This isolation allows applications to execute independently, and provides code, data, and symbol namespace separation.

The process owns all of the resources associated with the application. Memory protection to enforce process isolation is provided by a CPU’s memory management unit (MMU), such that a fault occurring within an application will cause only that application to fail, but will not impact the kernel or other applications.

**Figure 2. RTP architecture block diagram**

RTP support for VxWorks 6.x allows you (a Rational Rose RealTime user) to model and build RTP mode applications.

**Introduction to Wind River Workbench**

Wind River Workbench is based on the Eclipse Platform, an industry-standard framework for building development suites.
Wind River Workbench 2.x (shown in Figure 3) is a development suite that provides an efficient way to develop real-time and embedded applications with minimal intrusion on the target system. The Workbench debugger allows you to view and debug applications in the original source code. You set breakpoints, single-step, examine structures, and so on at the source level, using a convenient graphical interface. Workbench allows you to debug more than one application at a time. Currently, the latest Workbench version supported by Rational Rose RealTime is Workbench 2.6.

Figure 3. Wind River Workbench 2.2 in the Application Development perspective

Introduction to Target Control operations

Target Control refers to the Rational Rose RealTime toolset features that load, unload, execute, and terminate a Rational Rose RealTime-generated application, as well as the ability to reset a remote target platform.

In case of RTP mode applications, there are no load and unload operations. Only execute, terminate, and reset operations are relevant for RTP mode applications. This is because, when you execute an RTP mode application it gets loaded automatically, and the RTP will be created on target. Similarly, when you terminate an RTP mode application, it will be unloaded automatically.

Target Control operations can be performed in any of these three modes:

- **Manual mode**
  In Manual mode, Rational Rose RealTime does not provide any Target Control functionality. You are responsible for performing Target Control operations (such as loading and executing). After the target application starts, you can direct the Rational Rose RealTime toolset to connect to the executing target application for target observability.
• **Basic mode**  
  In Basic mode, Rational Rose RealTime uses the target environment's APIs to control the execution of the target application.  
  Target Control operations for RTP mode applications are as follows:  
  • **Reset:** resets the target  
  • **Execute:** starts the execution of application on target  
  • **Terminate:** stops the execution of application on target

• **Debugger mode**  
  Debugger mode provides the same capabilities as Basic mode, and, in addition, enables you to work with a C++ source debugger (for example, Tornado, Workbench, and so on) to set source code level breakpoints from within the UML model.  
  When these source breakpoints are hit at run-time, control of the executable is passed to the source debugger. When the application is continued, control of the executable is passed back to the Rational Rose RealTime toolset.  
  Debugger mode provides an integrated debug environment that permits a simultaneous use of source code and UML debugging styles.

**Rational Rose RealTime support for real-time processes**

This section explains the steps that user needs to follow to explore the Rose RealTime support for real-time processes.

**Getting Started**

In order to perform any Target Control operations, in Basic mode or Debug mode, you needs to start the Wind River Workbench.

Once the Workbench is started, you can follow these steps to complete the initial setup:

**Steps to create a new target connection from Wind River workbench**

1. Select **New Connection > WindRiver Target Server Connection for VxWorks.**
2. Enter appropriate values in the Connection Properties dialog. For example, consider:  
   a. The target IP address is 9.182.243.189  
   b. The Host machine name is win-rosert01  
3. Next, enter the Target Connection name as, for instance, 9.182.243.189.  
4. Right-click the newly created target and select **Connect.**  
5. The Target Server is connected to the target board.

**Steps to be followed on the Rational Rose RealTime Toolset side**

1. Start the Rational Rose RealTime, and open or create the model.  
2. Build the Real-Time Process application.  
3. In the Deployment view, open the Processor Specification dialog of the processor.  
4. The Processor Specification dialog opens, as shown in Figure 4.  
5. Switch to the **Detail** tab and set the properties with appropriate values:
a. **CPU:** Select the appropriate CPU type (for example, ppc)
b. **OS:** Set it to VxWorks
c. **Address:** Set the Target board IP address to 9.182.243.189
d. **Server:** Set the Target Server name to 9.182.243.189@win-rosert01
e. **Load script:** Set to $ROSERT_HOME\bin\tc\win32\workbench\rtp

Figure 4. Setting the target details in the Processor Specification dialog

6. Click the **Apply** button.

**Target Control support for VxWorks 6.x**

Once initial setup is completed, you can perform the Target Control operations on RTP mode applications built in Rational Rose RealTime.

**Running RTP mode application in Basic mode**

In order to run the Rational Rose RealTime-built RTP mode application in Basic mode, you need to perform the following steps.

**Steps to set the Operation mode to Basic mode**

1. In Rational Rose RealTime, open the Component Instance Specification dialog of the component instance, as shown in Figure 5.
2. Switch to the **Detail** tab and ensure that the **Operation Mode** is **Basic**, which is the default mode.
3. Click the **Apply** button.
Figure 5. Setting the operation mode to Basic

Steps to run the RTP application

1. In Rational Rose RealTime, select the Component instance in the Deployment view.
2. Right-click and select **Run** to run the RTP mode application, as shown in Figure 6.

Figure 6. Running a RTP application from toolset
Steps to terminate the RTP application

1. In Rose RealTime, select the Component instance in the Deployment view.
2. Right-click and select **Shutdown** to terminate the RTP mode application, as shown in Figure 7.

Figure 7. Terminating a RTP application from the toolset

Running an RTP mode application in Debug mode

In order to run the Rational Rose RealTime-generated RTP application in Debug mode, you need to perform the following steps.

Steps to set the Operation mode to Debug mode

1. In Rational Rose RealTime, open the Component Instance Specification dialog of the component instance.
2. Switch to the **Detail** tab and ensure that the **Operation mode** is **Debugger-WorkBenchRTP**, as shown in Figure 8.
3. Click the **Apply** button.
Figure 8. Setting the Debug mode

![Component Instance Specification for New...](image)

Setting the breakpoints for RTP mode applications

RTP mode applications do not require load and unload operations, whereas load and unload operations are mandatory for kernel mode applications. Hence, the Target Control operations in debugger mode are different from that of Kernel mode applications.

As soon as you set the Operation mode to Debugger-WorkBenchRTP in the Component Instance Specification, you can set the breakpoints on a state machine or for operations within the model.

Steps to set or unset the model-level breakpoints

1. In Rational Rose RealTime, right-click the component instance and select the Open Breakpoint Diagram option.
2. This opens the state machine diagram for the top state of the top capsule associated with that component instance.
3. The opened breakpoint diagram will be like that shown in Figure 9.
Although, the Breakpoint Diagram dialog is very similar to a State Monitor dialog, you cannot modify the state diagram that appears on the Breakpoint Diagram dialog: you can only set or unset breakpoints.

The browser for the Breakpoint Diagram dialog shows states, similar to a State Monitor browser. However, instead of displaying the elements (such as attributes, end ports, and probes), it shows all of the capsules associated with the component for the Component Instance opened in the initial state monitor.

**Note:** To open a Breakpoint Diagram dialog for a capsule that is not the top capsule, right-click that capsule on the Breakpoint Diagram dialog and select **Open Breakpoint Diagram**.

4. To set the breakpoint:
   a. Select the Breakpoint icon from the Breakpoint Diagram toolbar.
   b. Set the desired breakpoint on the diagram
5. To unset the breakpoint:
   a. Select the Breakpoint in the Breakpoint diagram
   b. Right-click and select **Delete**, as shown in Figure 10
Steps to set or unset the breakpoints for Operations

1. In Rational Rose RealTime, on the Model View tab in the browser, right-click an operation and select Add Breakpoint.
   **Note:** The Add Breakpoint command exists on the context menu for an operation only when there is at least one component instance that has a debugger specified in the Operation mode box.

2. To remove the breakpoint, select the Remove Breakpoint option.
   **Note:** The Remove Breakpoint command is enabled only when there is a breakpoint currently set for the selected operation.

Steps to view the breakpoints

To view all of the breakpoints that you have specified (those associated with the selected component instance)

1. In Rational Rose RealTime, select the Component instance in the Deployment view.
2. Right click and select View Breakpoints.
   **Note:** The View Breakpoints menu command is available only when the component instance specifies a debugger (for example, Debugger-WorkBenchRTP) in the Operation mode box.
3. The View Breakpoints dialog opens, as shown in Figure 11.
4. The **Location** and **Context** for each breakpoint displays in the list.
5. Click **Remove** to delete the currently selected breakpoints.

**Steps to Run the RTP application**

1. In Rational Rose RealTime, select the Component instance in the Deployment view.
2. Right click and select **Run** to run the RTP mode application, as shown in Figure 12.

**Figure 12. Running a RTP application in debug mode**

**Steps to Attach to the Workbench for source code debugging**

1. After the appropriate target connection is created in the Workbench, and the RTP application is run on the target, the RTP task will be shown in the Workbench. You need to attach to the RTP task in Workbench in order to perform source code debugging.
2. In Workbench, you can attach to the RTP task as follows:
   a. Select the running RTP task in **Target Manager**
   b. Right-click it and select **Attach to Real Time Process**, as shown in Figure 13.
   c. The debugger attaches without stopping the program.
3. As soon as you attach to the RTP task, the debugger perspective in the Workbench will open as follows. Because no breakpoint is hit yet, debug options (such as `step` and so on) are disabled, as shown in Figure 14.
4. When a breakpoint hit occurs, the source code file will automatically open in the Workbench, as shown in Figure 15.
5. Now, you can perform typical source code debugging operations.
Steps to terminate the RTP application

1. In Rational Rose RealTime, select the Component instance in the Deployment view.
2. Right click and select **Shutdown** to terminate the RTP mode application, as shown in Figure 16.

Figure 16. Terminating a RTP application in debug mode

What you have learned

This article has provided a brief introduction to the Target Control operations of Rational Rose RealTime, and the real-time processes of VxWorks 6.x. Also, it has described the Target Control operations, which you can perform on RTP applications in both Basic mode and Debug mode.
Related topics

- Subscribe to an IBM developerWorks newsletter.
- Download a free IBM product evaluation.
- Try IBM Bluemix free for 90 days. Bluemix is a cloud platform as a service (PaaS). Bluemix supports several programming languages and services as well as integrated DevOps to build, run, deploy and manage applications on the cloud.

© Copyright IBM Corporation 2008
Trademarks
(www.ibm.com/developerworks/ibm/trademarks/)