

Wind River Test Diagnostics 3.3

As the amount of software embedded within intelligent devices continues to increase, the challenge of integrating, verifying, and validating it also expands. The system integration challenge is multiplied when complex new technologies are employed in the device, leading to complicated run-time defects. Typically traditional debuggers are too intrusive to be used on production software in the lab, thereby severely limiting developer access to device internals for root-cause diagnostics.

The isolation and elimination of defects is often made even more difficult when teams are geographically distributed

across the globe. Once a defect is discovered in a remote lab, development engineers are challenged to isolate or reproduce and repair it, then resubmit the software for test and validation. This is often a time-consuming, unpredictable cycle that uses essential resources. Current methods require lengthy debug, code, build, reload, and test steps that together limit diagnostic efficiencies and delay time-to-market.

The Wind River Test Diagnostics system is a productivity solution for device development and system integration teams. It provides a run-time diagnostics and patching solution based on Wind

River's unique sensorpoint dynamic instrumentation technology. A server-based, workgroup solution, it provides a collaboration platform and diagnostic information repository that can be shared across distributed teams. It also includes an integrated virtual lab manager (VLM) for network-based target management that enables optimal use of lab device assets by embedded development teams. Wind River Test Diagnostics is a functional subset of the Wind River Test Management system, sold separately.

A New Approach to Run-Time Diagnostics

The Wind River Test Diagnostics system provides a unique approach to addressing the complex run-time defects that surface during system integration and test. It is a scalable system that links product development teams in an intelligent, collaborative workflow so they can efficiently validate embedded devices and rapidly resolve issues. The product leverages sensorpoints, a minimally intrusive dynamic instrumentation technology, to probe the run-time system for forensic data, input or read run-time data values, or insert patch code, all without stopping, reloading, or rebooting the device.

Build Higher-Quality Devices Faster at Less Cost

Wind River Test Diagnostics enables development teams to adopt a more repeatable process for diagnosing problems with embedded systems. It helps users detect bugs earlier and correct faults as much as 10 times faster. It provides unprecedented visibility into

Table of Contents	
A New Approach to Run-Time Diagnostics	1
Build Higher-Quality Devices Faster at Less Cost	1
Optimize Diagnostic Cycle Time for Faster Time-to-Resolution	2
Manage Resources More Effectively	2
Accelerate System Integration and Verification	2
Diagnose Run-Time Defects Dynamically	2
Patch Software on the Fly	3
Manage Targets Remotely in Virtual Labs	3
Collaborate Across the Globe	4
A Scalable, Distributed Architecture	4
Open, Scalable Server Environment	4
Web-Based Applications	4
Developer Plug-Ins	4
Downloadable Agent	5
Technical Specifications	5
Target Device OS Support	5
Target Device Processor Support	5
Developer Plug-ins Host OS Support	5
Wind River Test Diagnostics Server OS Support	5
Subset of Wind River Test Management	5
About Wind River	5
Partner Ecosystem	5
Professional Services	5
Education Services	5
Support	6

device internals at run-time for developers. The system streamlines diagnostic workflow and mitigates risks and delays so products can get to market faster at lower cost.

Optimize Diagnostic Cycle Time for Faster Time-to-Resolution

Wind River Test Diagnostics eliminates the delays caused by symptom reporting, hit-or-miss fault isolation, and long rebuild cycles. It offers dynamic instrumentation and patching technology that can be deployed without a debugger to gather run-time data. When integration teams detect defects they can dynamically deploy diagnostics probes in devices without stopping them. Developers can access resulting fault logs to quickly identify and debug defects in their own environments. When the problem is fixed, software patches can be “hitlessly” applied to running lab devices for validation, dramatically accelerating the defect resolution cycle (see Figure 1).

Manage Resources More Effectively

Wind River Test Diagnostics lets distributed teams work more effectively. Users can create and manage repositories of reusable diagnostics assets and patches and share them across distributed teams throughout the product life cycle. Its virtual lab manager allows distributed target device

management and scheduling. It allows better management of device utilization to help prevent team inefficiencies when waiting for target hardware and helps save money by minimizing underutilized capital equipment.

Accelerate System Integration and Verification

Central to Wind River Test Diagnostics is Wind River’s unique sensorpoint technology. Sensorpoints let users gain visibility into running devices without stopping or reloading them. A sensorpoint is C language software used to instrument “live” C and C++ applications dynamically, without modifying the application source code, rebuilding the application, reflashing boards, or rebooting the device. Sensorpoints allow developers and testers to inject custom code into a running executable to gather data, add functionality, or change the flow of control. Sensorpoints dynamically patch the running binary software and can be inserted on entry or on exit of a function or, in certain conditions, on a specific line of code. Sensorpoints can access local and global variables within the scope of a function (see Figure 2).

Sensorpoints are minimally intrusive on device performance and footprint. They can be disabled and enabled remotely, individually, or in groups, separate from

```
foo ( arg1)
{
  int error = 0;

  if (arg1 < 500)
  {
    process();
  }
  else error = errorHdl();

  return (error)
}
```

Annotations in the code block show dynamic injection of code:

- At the function entry: `log($arg1);`
- At the `if (arg1 < 500)` line: `$arg1 = 501;`
- At the function exit: `log($error);`

Figure 2: Sensorpoint instrumentation allows the dynamic injection of code on entry, on line, or on exit of a software function

their installation. Sensorpoints do not need to be defined upfront when original source code is written. Their power stems from the fact that they can be defined and applied very late in the development cycle. The target application does not have to be rebuilt and the device does not need to be restarted in order to put sensorpoints to work.

Diagnose Run-Time Defects Dynamically

An inevitable outcome of development, system integration, and testing is the discovery of defects. Wind River Test Diagnostics allows developers to focus on an area of concern by temporarily injecting instrumentation (e.g., printf or log statements) to rapidly collect data that can point to the source of the problem.

Diagnostic sensorpoints are created by software developers using the developer plug-ins (or command-line equivalents) included with Wind River Test Diagnostics (see Figure 3) and then published to the Test Diagnostics server. Developers or testers deploy the sensorpoints to the failing device and re-run the tests to collect data. Any log data that is generated by sensorpoints is captured in the Test Diagnostics system and viewed via a browser-based viewer

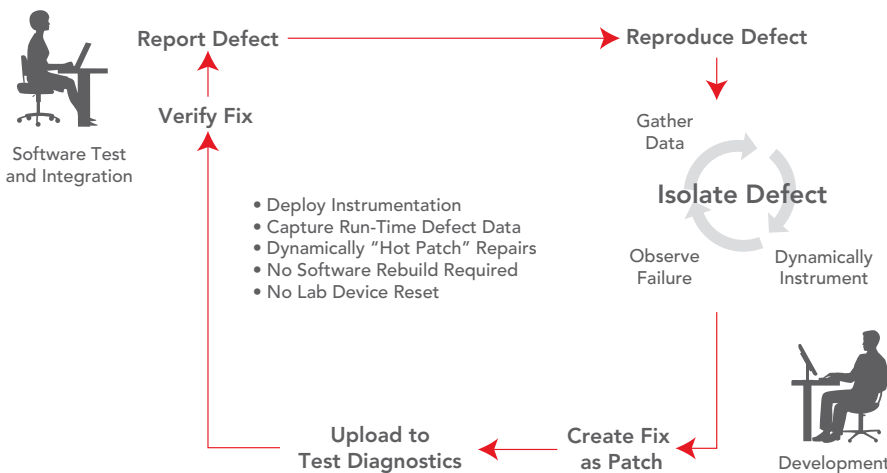


Figure 1: Access defective device run-time data, skipping time-consuming build, reload, and restart cycles

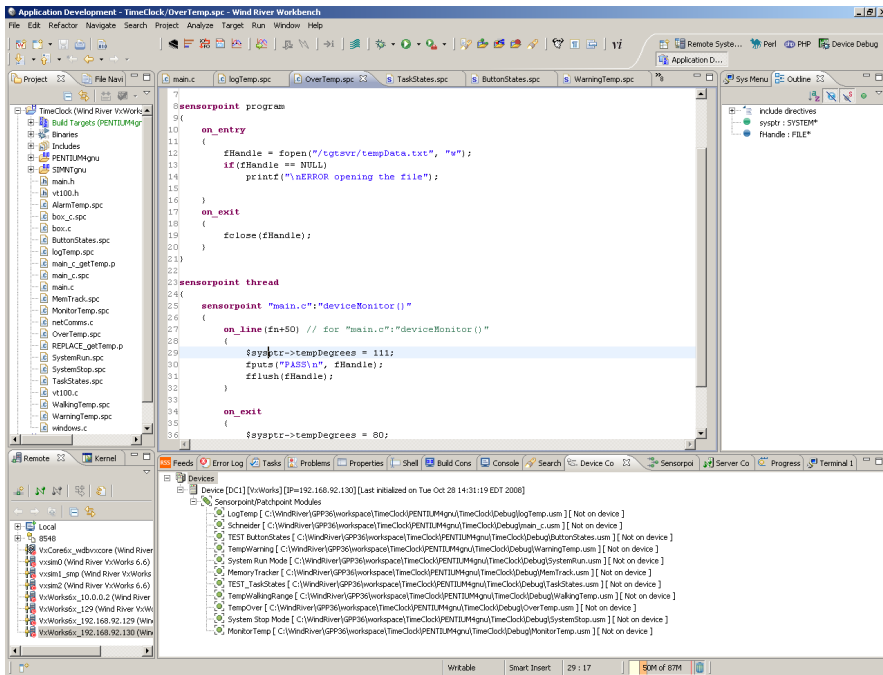


Figure 3: Software developers create dynamic sensorpoint instrumentation or “hitless” patchpoints with Eclipse-based Developer plug-ins or command-line tools

or developer’s desktop plug-ins. For VxWorks users, core images and error detection and recovery (ED&R) logs can also be captured by the Test Diagnostics system for analysis.

This cycle can be repeated as often as necessary to rapidly isolate the root cause of the problem. Sensorpoints are managed as a growing knowledgebase in the Test Diagnostics system along with relevant device and build information.

The following are example uses of sensorpoints for diagnostics:

- Inject specific values into variables.
- Log internal device data generated at run-time for validation against expected parameters.
- Alter the system’s environment (e.g., change values read from sensors, set CPU and peripheral controllers register values, modify streaming data).
- Set the device to a specific state that may only be reached after hours of continuous operation (e.g., buffers full, queues empty).
- Eliminate the use of sophisticated test equipment (e.g., forcing the hop count to a large value in a data packet, stubbing out interfaces).

Patch Software on the Fly

When developers isolate the defect to a given function or module, they typically create a fix by modifying the source code in their development environment at their desks. Once complete, the fix needs to be validated in the actual device. Typically the process to validate is to check in the fix, rebuild the device software, and reload and retest it on the device. This process can be quite time consuming and can hold up test execution for hours or days. There are further delays and resource requirements if the lab has to be reset and any long-run test suites run again to the point of failure.

Wind River Test Diagnostics provides a dynamic patching mechanism where once a fix is created by developers, it can be published in the form of a patchpoint. Patchpoints are variations of sensorpoints that perform dynamic patching of code. The main difference is that patchpoints *replace* existing code with new C or C++ code, while sensorpoints *insert* new C code within existing code.

Patchpoints allow the defect repair cycle to be dramatically reduced and allow

prospective fixes to be added to the failing device in the lab, ideally while still in its failing state, so that the repair can be quickly validated by the test team.

If the fix does not actually address the problem or, as is often the case, the fix introduces other problems, the developer can gather further diagnostic data with sensorpoints and iterate on subsequent patches until validation is complete. Once validated, patchpoints can be made persistent and reused indefinitely in the lab or alternatively can be made obsolete when the fix is permanently committed to a subsequent build.

Manage Targets Remotely in Virtual Labs

Many companies today have expensive capital equipment spread across the many labs in their companies for use by developers. However, it’s often hard to know what devices are available that have the desired configuration, and it’s difficult to provision them with appropriate software for diagnostics. This problem is compounded when teams are located across multiple sites. Wind River Test Diagnostics provides a virtual lab manager application that lets users remotely reserve and provision desired equipment and access it over wide area networks.

Lab administrators define a Virtual Lab view of the available devices that captures device configuration, location, connectivity, and various other parameters. A “device” is defined as a single instance of a running lab diagnostics agent, so the system can manage multiple boards in a chassis or even multiple cores as separate devices. Lab devices can be connected directly through their Internet Protocol (IP) address and/or indirectly through a terminal server to its serial port. An APC switch can also be used to remotely power cycle/reboot devices. Administrators can also set up and manage permissions and reservation policies such as the maximum time a device can be reserved. Users can search for desired equipment characteristics and check out and reserve devices for testing

and track their utilization. Developers can then reserve a device, provision it with the latest build, and execute diagnostics on it (see Figure 4).

All the actions of deploying sensorpoints and patchpoints and gathering data and logs are managed through the VLM. Wind River Test Diagnostics saves time and hassle when working with lab devices and allows maximum utilization of people and capital equipment resources.

Collaborate Across the Globe

Isolating and resolving defects in embedded devices often requires the collaboration of a team. Ideally system integrators and testers can assist in the isolation of defects so they can provide developers the details they need to resolve the problem. Ideally developers have access to detailed device operational data rather than just observed symptoms so they can get to root cause fast.

However, given that many development and test organizations today are in different physical locations and time zones, this can significantly limit diagnostics productivity in solving tough

problems. Wind River Test Diagnostics lets teams collaborate to gather forensic information, validate prospective fixes, and capture reusable knowledge.

A Scalable, Distributed Architecture

Wind River Test Diagnostics is designed to provide a scalable, distributed architecture that brings together all the players into a common environment:

- The Wind River Test Diagnostics server manages device and diagnostics data. It resides on a Windows or Linux server on the user's network.
- Web browser-based application interfaces run in standard browsers and allow users to access the Test Diagnostics system over their intranet.
- Developer plug-ins are installed on software developers' desktops to allow sensorpoint and patchpoint development within their existing command line, Eclipse-based, or Wind River Workbench programming environments.
- A device management agent is built in conjunction with the user's software and downloaded to the lab device on demand to manage dynamic instrumentation.

Open, Scalable Server Environment

At the center of Wind River Test Diagnostics is a scalable server environment. Built on standard, proven Java/Java 2 Platform, Enterprise Edition server and relational database technology, the server acts as the diagnostics data repository and the central collaboration point for all users. The Test Diagnostics server stores diagnostic data, manages the virtual labs, deploys sensorpoints and patchpoints on demand, and gathers logs and device data from managed devices.

The server has built-in security at multiple levels that protect system access or device information. Users have secure logins mapped to assigned privileges; all application and device communication can be encrypted and authenticated.

Web-Based Applications

Users interact with Wind River Test Diagnostics through their standard, Flash-enabled web browser via a licensed application seat. Whether sharing sensorpoints, managing labs, or patching remote systems, all the functions of the system can be carried out though an intranet. The interactive user interface speeds user operation while avoiding the cumbersome system administration common with heavyweight desktop clients.

Developer Plug-Ins

Developer plug-ins extend the Wind River Workbench Eclipse-based integrated development environment (IDE) with tools for creation and deployment of sensorpoints and patchpoints. The plug-ins let developers simply point to a function or line of code within the Workbench graphical editor and create a sensorpoint or patchpoint where desired. The plug-ins support sensorpoint compilation and the recovery and analysis of resulting logs and diagnostics data.

All this developer capability is alternatively available via command-line tools or a set of general Eclipse plug-ins that will work with other environments.

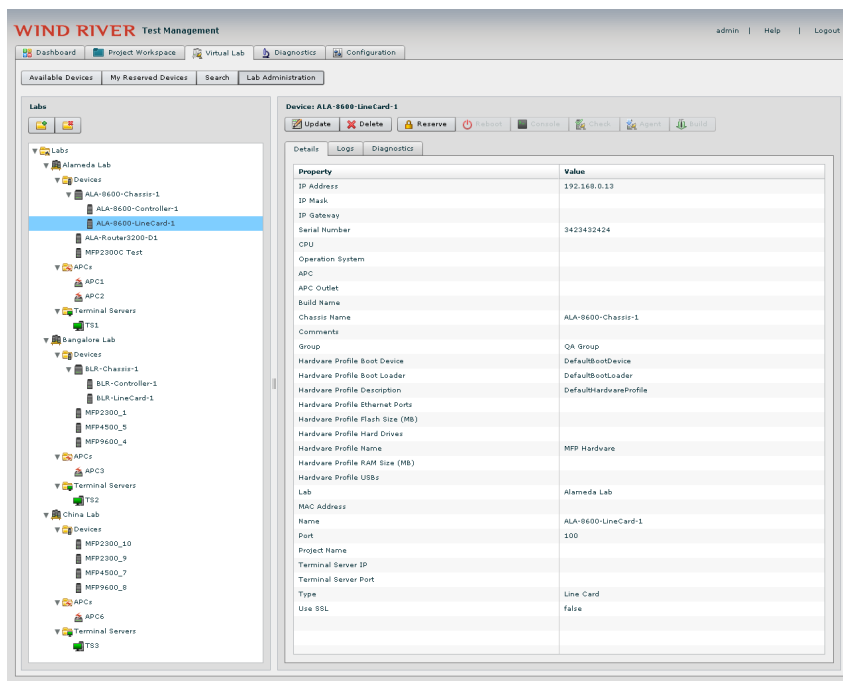


Figure 4: The virtual lab manager lets worldwide teams find, reserve, provision, and access devices

Downloadable Agent

The sensorpoint and patchpoint dynamic technology is enabled by a small run-time device management agent that can be embedded in the device software or downloaded on demand. The device management agent performs the housekeeping functions required for dynamic instrumentation such as sensorpoint load/unload placement within the running device and enable/disable on demand. The agent is also configured to gather and offload logs and core dumps from the device as needed.

For dynamic loading of the device management agent into lab devices, the system leverages VxWorks downloadable kernel modules (DKM) or Wind River Linux RPM package manager. Users can use the virtual lab manager to query whether the device management agent is resident in a target device; if not, it can initiate agent download, install, and startup. The agent can be left in the device or software build or removed automatically after diagnostics is complete.

Technical Specifications

Wind River aims to support the widest range of processor, operating system, host, and infrastructure software combinations with Wind River Test Diagnostics. This list is subject to change and continuously expanding to cover both Wind River and non-Wind River platforms.

The following is the support matrix as of the time of this document's publishing. Contact your Wind River representative for the latest information on supported platforms, hardware requirements, and technical specifications.

Target Device OS Support

- VxWorks 6.1–6.8 platforms
- Wind River Linux 1.4, 1.5, 2.0.2, 3.0 based platforms
- VxWorks 5.5.1
- Other Linux, UNIX, and real-time operating systems

Target Device Processor Support

- ARM/XScale
- IA-32
- MIPS
- PowerPC

Note: VxWorks 5.5.1 is currently supported only on PowerPC and IA-32; 64-bit processors are not yet supported.

Wind River Test Diagnostics generally supports processors within these architecture families that are supported by the VxWorks or Wind River Linux platforms. Contact Wind River Sales for further details and exceptions.

Developer Plug-ins Host OS Support

- Windows XP Professional SP2 or Vista
- Red Hat Enterprise Linux 4 or 5, Red Hat Fedora Core 7
- SUSE Desktop Linux 10, SP1, SUSE Linux/openSUSE 10.2
- Solaris 9 (update 9/05) or 10

Note: For VxWorks 5.5.1, only Windows hosts are supported.

Wind River Test Diagnostics Server OS Support

- Red Hat Enterprise Linux Workstation with option 5
- openSUSE Linux 11.0
- Microsoft Windows 2000 Professional (Service Pack 4) or XP Professional (SP2)

Subset of Wind River Test Management

Wind River Test Diagnostics is a subset of the Wind River Test Management system for device test automation. Wind River Test Management includes all the features of Wind River Test Diagnostics. It shares the same foundation of sensorpoint technology, collaboration server environment, and virtual lab manager features and then builds on this with additional applications for testing and quality management.

Wind River Test Management enables manual and scripted test management, automated and optimized test execution, fault injection, and results analysis. The system also leverages dynamic

instrumentation to provide run-time test coverage, test-to-code traceability, and performance profiling capability for use in the system test lab. Wind River Test Diagnostics users can upgrade to the Wind River Test Management solution to add test management and automation features to their diagnostics platforms.

About Wind River

Partner Ecosystem

Wind River's world-class partner ecosystem assures tight integration with solutions from a wide range of premier software providers. Our partners extend the capabilities of Wind River Test Diagnostics by offering out-of-the-box integration and support for key technologies in a number of fast-moving industries.

Professional Services

Wind River Professional Services, a CMMI Level 3–certified organization, enables you to reduce risk and focus on development activities that add value and differentiate your design. Our team delivers device software expertise within structured engagements that directly address key development challenges and contribute to the success of our clients. Our track record of timely delivery and in-depth understanding of market and technology dynamics makes Wind River a valuable implementation partner for clients worldwide.

Wind River Professional Services offers a variety of services around Wind River Test Diagnostics to meet your needs, including installation and configuration, tool integration, deployment best practices, and platform migration.

Education Services

Education is fundamentally connected not only to individual performance but also to the success of a project or entire company. Lack of product knowledge can translate into longer development schedules, poor quality, and higher costs.

The ability to learn—and to convert that learning into improved performance—creates extraordinary value for individuals, teams, and organizations. To help your team achieve that result, Wind River offers flexible approaches to delivering Wind River Test Diagnostics product education that best fits your time, budget, and skills development requirements.

Support

Wind River provides full technical support for Wind River Test Diagnostics at centers worldwide. Support is also available 24 hours a day at our Online Support website (www.windriver.com/support) or by email at support@windriver.com. Visit Wind River Online Support for fast access to product manuals, downloadable software, and other problem-solving resources for Wind River products. Additional features, including patches and technical tips for common problems, are available for all customers on subscription. Online Support visitors can also access a community of developers to discuss their issues and experiences.

WIND RIVER

Wind River is a world leader in embedded and mobile software. We enable companies to develop, run, and manage device software faster, better, at lower cost, and more reliably. www.windriver.com

© 2010 Wind River Systems, Inc. The Wind River logo is a trademark of Wind River Systems, Inc., and Wind River and VxWorks are registered trademarks of Wind River Systems, Inc. Other marks used herein are the property of their respective owners. For more information, see www.windriver.com/company/terms/trademark.html. Rev. 06/2010