Introduction to Wind River Workbench 4.x Support

This topic provides basic information about the capabilities of the C/C++test Plugin for Wind River Workbench 4.x.

Integration

You can integrate C/C++test with Wind River Workbench by installing the C/C++test Eclipse Plugin into your Wind River Workbench 4.x installation – see In stalling the C/C++test into Wind River Workbench for details.

See IDE Support for information about Wind River Workbench IDE versions supported by C/C++test.

Supported Practices

- Static Analysis – Static Analysis with the C/C++test Wind River Workbench Plugin
- Unit Testing – Unit Testing with the C/C++test Wind River Workbench Plugin
- Application Monitoring – Application Monitoring with the C/C++test Wind River Workbench Plugin

Supported Project Types

Static analysis, unit testing, and application monitoring is supported for the following project types:

- VxWorks Downloadable Kernel Module
- VxWorks Real Time Process Project

Depending on the project structure, you may need to test your code through Test Projects – separate projects that enable testing. See Using Test Projects with the C/C++test Wind River Workbench Plugin.

Known Limitations

- C/C++test does not support command line testing for Wind River Workbench projects. The command line interface is supported only for Test Projects.
- The following compiler keywords are not supported:
    - extended
    - __interrupt__ and interrupt
    - __packed__ and packed
    - bool, pixel, vec_step, and vector AltiVec Keywords
    - __ev64_* Keywords
    - __accum, __fixed, __X, and __Y DSP Extensions
    - pascal
- C++test may fail parsing when the following intrinsic functions or type specifiers are used in the source code:
    - ARM: ffi
    - PPC: multiply-accumulate instructions (PPC405), AltiVec instructions group
    - Sh: __fixed type
    - Spar: __scan and __divs