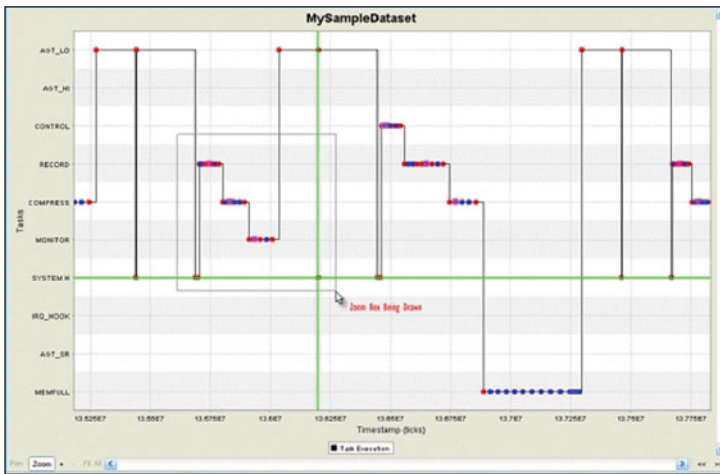


EDGE Profiler

There are times when troubleshooting embedded development is like a black art, where problems occur mysteriously and for no apparent reason. The EDGE Profiler was introduced to demystify these cases by letting you see what the underlying OS and application code are doing. In addition, we added performance-measuring capabilities to the product, letting you see what's happening inside the black box.



EDGE Profiler tasks view shows OS and user-defined events in the order that they occur.

The EDGE Profiler consists of two software components: an embedded agent running on the target and a GUI-based analysis residing on your host PC. The EDGE Profiler doesn't require additional hardware like trace buffers or specialized connection modules to run, which also means that it is easily portable across a wide variety of processors.

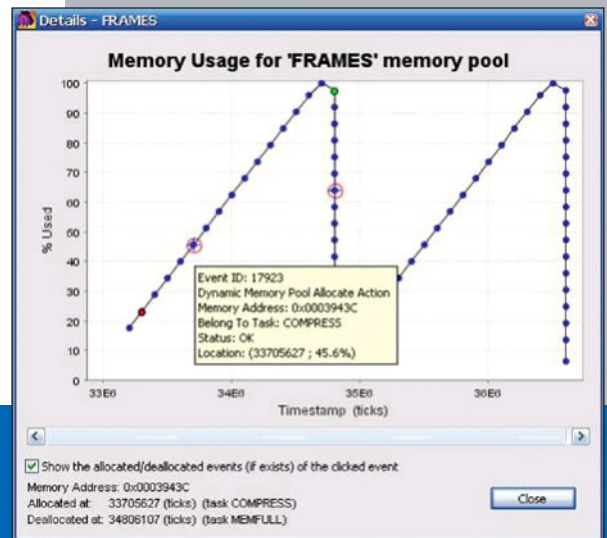
The target side of the EDGE Profiler works by adding specialized code into the OS that captures system events, and sends them to the agent to transmit back to the host for analysis. To make the product even more useful, we built in the capability to allow developers to add their own hooks into their application code.

The host side of the tool is an Eclipse-based plugin. This shows the events in the order that they occur, with timestamps. Memory pool allocations and deallocations are shown, and time measurement between any two points in the display is as easy as a mouse click.

The EDGE Profiler already knows about the Nucleus OS, and can let you monitor the status and performance history of all the OS elements. By extending the capability to insert user-defined events into application code, developers can get a quantitative measurement of how well their algorithm is tuned. In addition, event capture filters can be set and downloaded to the target without recompiling the code.

Advanced Features

- The host-side part of the EDGE Profiler is a plug-in into the Eclipse framework, familiar to EDGE IDE and EDGE Debugger users
- The target-side part of the EDGE Profiler knows about all of the functionality of the Nucleus operating system
- Users can add customized instrumentation hooks into their application code easily for additional performance metrics
- Filters can selectively capture the events of interest without having to recompile the target application code
- Broad processor support means that most mainstream processors are supported with little work needing to be done by the developer
- Ethernet communication can be used to transfer event data from the target to the host



The EDGE Profiler memory view shows OS memory allocations and deallocations.

- Blue dots represent match memory pool allocations and deallocations.
- A red dot represents memory pools that have been deallocated without a matching allocation. These areas could represent bad pointers.
- A green dot represents memory pools that have been allocated without a matching deallocation. These dots could be memory leaks.



739 North University Blvd • Mobile, Alabama 36608
Phone: 251.208.3400 • Fax: 251.343.7074 • Toll free: 800.468.6853

embedded_info@mentor.com • www.mentor.com/embedded