Software Portability, Interoperability and Reuse: A Middleware Approach

Andrew Vandivort, Harold H. Smith III, Michael W. DaBose, Gerald E. Morris, Garrett J. Wright
Raytheon Missile Systems

Abstract – Poster

Within the Mil-Aero industrial community there is a great need for software architectures that inherently promote portability, interoperability and reuse. Toward this end the Missile Advanced Processing Systems (MAPS) Middleware Architecture IR&D is developing an infrastructure services layer (middleware) that will enable distributed, component based software development. Key themes of this middleware include:

- Layered, component based architecture.
- Described and captured in UML.
- External Entity Abstractions (EEAs) – decouple software applications from particular hardware realization, also promotes simulation/tactical software interoperability.
- Interprocess / Interprocessor Communication (IPC) mechanism with interchangeable communication protocol transports (e.g., Connexis, VxMP, RT CORBA)
- Loosely coupled components (e.g., asynchronous message passing).
- Hierarchical / distributed / multithreaded executive control shell.
- Decoupled event timing
- Use of *commercial-off- the-shelf* RTOS and OO-frameworks where applicable (e.g., Tornado II / VxWorks, Rational Rose RealTime)

This paper summarizes progress to date on the MAPS Architecture Middleware IR&D. More specifically we: 1) discuss how this middleware design promotes software portability by isolating and minimizing software elements that must be modified before deployment onto new hardware topologies; 2) show how these portability constructs also enable simulation/tactical software interoperability; 3) describe how software subsystem components potentially increase software reuse and lessen the pain of software modification. Finally, we will report middleware performance benchmarking results we have collected on both our host development platform (WindowsNT) and real-time embedded target (PowerPC 7400).

We have leveraged, as much as possible, a COTS based software development tool chain, ranging from embedded RTOS to application development framework and analysis tools. Many middleware services are derived directly from the Rational Rose RealTime run-time services library. Additional services are realized through the construction of novel middleware components, which also utilize these COTS technologies.

The MAPS team has successfully developed a middleware prototype, which has been deployed onto an eight-node WindowsNT development cluster and quad-PowerPC embedded processor board. This middleware has been demonstrated to span heterogeneous combinations of these two processing systems as well, without modification.

Current work is focusing on middleware performance optimizations, which include the integration of a DMA based transport into the Connexis IPC mechanism. Middleware system performance is being evaluated through a number of pilot programs within Raytheon, which are representative of the missile systems domain.

References:

B. Selic, J. Rumbaugh, “Using UML for modeling complex real-time systems”, Rational Software Corp, white-paper, 1998.

B. Selic, et. al., *Real-Time Object-Oriented Modeling*, John Wiley & Sons, 1994.