An Application Development and Deployment Platform for Satellite Clusters

Daniel Balasubramanian, William Emfinger, Pranav Kumar, William Otte, Abhishek Dubey, Gabor Karsai
ISIS, Dept of EECS, Vanderbilt University, Nashville, TN 37235, USA

Subject areas: Model-based software development, software modeling, spacecraft software architectures

A fractionated spacecraft is a system comprising of a cluster of wirelessly communicating satellites, each with sensors, embedded processors, data storage, attitude control and propulsion. These clusters have several advantages over monolithic satellites: they offer redundancy, the economy of smaller satellites and resource sharing. However, there are other challenges beyond those of distributed software systems, such as intermittent networking via fluctuating communication links.

To address the challenges, this presentation will describe a software architecture for fractionated satellites that we call DREMS: Distributed REaltime Managed System (http://www.isis.vanderbilt.edu/drems). DREMS consists of two main parts: (1) a design-time toolsuite for the modeling, analysis, synthesis, integration, debugging and testing of application software built from reusable components, and (2) a run-time software platform for deploying and operating application software on a cluster of satellites. The design-time toolsuite supports a model-based design and implementation paradigm, where models represent software components with interfaces and the architecture of applications is built from interacting components. Models are used to generate infrastructure code and complex configuration data. The run-time software platform consists of two layers: an operating system (OS) and middleware. The OS layer provides fundamental services, such as process scheduling and resource management, and the middleware layer works with the OS to perform the high-level component scheduling and provide the infrastructure for component interactions spread across different nodes. The OS scheduler supports mixed criticality applications using both temporal and spatial task partitioning (similar to ARINC-653 partitions [1]), as well as priority-based preemption that allows sporadic, critical tasks to execute as soon as possible.

DREMS is particularly well-suited to applications like flight control software because of its support for the publisher/subscriber communication pattern, which facilitates distributed development and integration and has previously been found to be a good fit for flight software [2]. Other unique features of DREMS include support for multi-level security (MLS) and mandatory access control (MAC) for secure information flows, and a model-based deployment and configuration tool that assists with deploying and updating software on the cluster, both of which have been cited as needs for future space architectures [3], [2].

Our presentation will also describe an experiment implemented on a multi-node testbed running DREMS on which a cluster of three satellites running a (mockup) cluster flight control application along with (mockup) image processing applications was emulated. This experiment demonstrates the entire DREMS infrastructure, including model-based development, deployment, management and operation.

Our ongoing work includes design-time methods for analyzing and planning application network utilization. The goal is to generate an application schedule that guarantees network bandwidth to applications throughout the orbital cycle, during which the bandwidth and availability of network links can greatly fluctuate. Other ongoing work addresses the design-time analysis of the component-based application architecture: the goal is to develop a framework for verifying the end-to-end timing properties of applications based on the knowledge of the timing of the components and the composition of the architecture.

Acknowledgments: This work was supported by the DARPA System F6 Program under contract NNA11AC08C.

REFERENCES

[1] *Document No. 653: Avionics Application Software Standard Inteface (Draft 15)*, ARINC Incorporated, Annapolis, Maryland, USA, Jan. 1997.

[2] L. Fesq and D. Dvorak, “Nasa’s software architecture review board’s (sarb) findings from the review of gsfc’s "core flight executive/core flight software",” in *Workshop on Spacecraft Flight Software (FSW)*, 2012.

[3] P. Wood, “Cyber attacks: An emerging threat to satellites,” in *Workshop on Spacecraft Flight Software (FSW)*, 2012.