A Spaceborne Embedded COTS Cluster for Computational Optics

Daniel S. Katz, Paul L. Springer
{Daniel.S.Katz, Paul.L.Springer}@jpl.nasa.gov
Jet Propulsion Laboratory
California Institute of Technology
Pasadena, CA 91109

Over the last decade and continuing into the foreseeable future, a trend has developed in the spacecraft industry of both number of missions and the amount of data taken by each mission increasing faster than bandwidth capabilities to send these data to Earth. The result of this trend is a bottleneck between data gathering (on-board) and data analysis (on the ground), which many missions try to avoid by running instruments at a low duty cycle. Another alternative is to overcome this bottleneck by performing data analysis on-board and only transferring the results of this analysis to the ground, rather than the raw data. However, this demands a dramatic leap in capability of on-board computing. One attempt to do this is being made by the NASA HPCC Remote Exploration and Experimentation (REE) Project, which is developing spaceborne embedded COTS clusters. These clusters, while originally intended to solve one problem (of limited bandwidth causing low duty cycles,) may also provide answers to many other questions, such as how to ensure segmented mirror systems maintain a flat phase front and how to use traditional CCDs to take long images in a relatively high radiation environment. These questions are now being examined by the NGST Supercomputing Study Group working with REE, as one part of the REE's vision is to enable new science that had not previously been considered feasible.

We expect spaceborne embedded clusters will share many characteristics of the growing community of traditional, ground-based clusters (i.e.; Beowulfs) such as POSIX-compliant operating systems and message-passing applications, but they will also have significant differences, including packaging and the need for fault-tolerance and real-time scheduling in software. The similarities will allow software to be developed on standard workstations and ground-based clusters, and then ported with limited changes to the spacecraft. REE's intent is that as many of the differences as possible be hidden from the application developer by middleware. However, some of the differences, such as the fault rates that will occur when using commodity components in space will have some impact. This paper discusses both the similarities and the differences, and how they impact application development and application performance.