THE COMMON COMPONENT ARCHITECURE (CCA) APPLIED TO SEQUENTIAL AND PARALLEL COMPUTATIONAL ELECTROMAGNETIC APPLICATIONS

Daniel S. Katz*, E. Robert Tisdale, Charles D. Norton
Jet Propulsion Laboratory, California Institute of Technology
4800 Oak Grove Drive, Pasadena, CA, 91109, USA
Phone: +1.818.354.7359, +1.818.354.1295, +1.818.393.3920
Fax: +1.818.393.6141, +1.818.393.3134, +1.818.393.3134
{Daniel.S.Katz, E.Robert.Tisdale, Charles.D.Norton}@jpl.nasa.gov

ABSTRACT

The development of large-scale multi-disciplinary scientific applications for high-performance computers today involves managing the interaction between portions of the application developed by different groups. In the business world, component-based software engineering, including Microsoft's Component Object Model (COM) [1,2] and Sun's Enterprise JavaBeans (EJB) [3,4], is often used. These technologies are often not appropriate for scientific computing, and the Common Component Architecture (CCA) [5,6] is an alternative.

The CCA Forum is developing a component architecture specification to address high-performance scientific computing, emphasizing scalable (possibly-distributed) parallel computations. The forum is also developing a reference framework, various components, and supplementary infrastructure, and is collaborating with practitioners in the high-performance computing community to design suites of domain-specific abstract component interface specifications. [7]

NASA's ESTO-CT (Earth Science Technology Office's Computation Technologies) project has so far been successful in "Demonstrating the power of high-end, scalable, and cost-effective computing environments..." [8]. The project is now emphasizing frameworks and interoperability for large-scale high performance scientific software development and maintenance. The material in this presentation is part of an ongoing study of the CCA Forum's technology by the ESTO-CT project. [9]

This paper presents an examination of the CCA software in sequential and parallel electromagnetics applications using unstructured adaptive mesh refinement (AMR). The CCA learning curve and the process for modifying Fortran 90 code (a driver routine and an AMR library [10]) into two components are described. The performance of the original applications, and the componentized versions are measured and shown to be comparable.

KEYWORDS

Parallel computing, component architecture, computational electromagnetics, unstructured adaptive mesh refinement.

*Corresponding and presenting author

REFERENCES

[1] Microsoft COM Web page, see http:///www.microsoft.com/com/about.asp.

[2] R. Sessions, *COM and DCOM: Microsoft's Vision for Distributed Objects*, John Wiley & Sons, 1997.

[3] R. Englander, *Developing Java Beans*, O'Reilly, 1997.

[4] R. Monson-Haefel, *Enterprise Java Beans*, O'Reilly, 1999.

[5] R. Armstrong, D. Gannon, A. Geist, K. Keahey, S. Kohn, L. C. McInnes, S. Parker, B. Smolinski, "Toward a Common Component Architecture for High-Performance Scientific Computing," *Proceedings of High Performance Distributed Computing*, pp. 115–124, 1999.

[6] Common Component Architecture Forum, see http://www.cca-forum.org/.

[7] B. Norris, S. Balay, S. Benson, L. Freitag, P. Hovland, L. McInnes, and B. Smith, "Parallel Components for PDEs and Optimization: Some Issues and Experiences," in review as an invited paper in a special issue of *Parallel Computing*, Feb. 2002.

[8] ESS Project Plan, June 2000 (Project homepage: http://ct.gsfc.nasa.gov/)

[9] Daniel S. Katz, E. Robert Tisdale, and Charles D. Norton, "A Study of the Common Component Architecture (CCA) Forum Software," *High Performance Embedded Computing (HPEC-2002)*, Boston, MA, 2002.

[10] Charles D. Norton, John Z. Lou, and Thomas Cwik, "Status and Directions for the PYRAMID Parallel Unstructured AMR Library," *8th Intl. Workshop on Solving Irregularly Structured Problems in Parallel (15th IPDPS)*, San Francisco, CA 2001.