

# THE SIERRA COMPUTATIONAL MECHANICS FRAMEWORK: PARALLEL ERROR HANDLING

J. R. Overfelt,<sup>a</sup> H. C. Edwards<sup>b</sup> and R. A. van de Geijn<sup>c</sup>

Sandia National Laboratories  
Dept. 9143, MS 0827  
Albuquerque, NM 87185-0827  
<sup>a</sup>jroverf@sandia.gov  
<sup>b</sup>hcedwar@sandia.gov

<sup>c</sup>Computer Science Department  
The University of Texas at Austin  
Austin, TX 78712  
rvdg@cs.utexas.edu

SIERRA is a modern computational framework developed at Sandia National Laboratories to support parallel mechanics application code development. SIERRA can be thought of as a suite of basic and advanced “services”, which are delivered through a set of C++ objects with a well-defined API. The services provided are primarily mechanics independent, allowing applications to be written with the addition of mechanics-specific functionality. SIERRA supports unstructured, fully-distributed meshes. Its low-level parallelism uses MPI and is not threaded.

A number of mechanics codes have been written using SIERRA, including thermal, quasistatics, explicit transient dynamics, and nearly-incompressible fluid mechanics applications. Several more applications are currently in development. The main idea behind SIERRA is to simplify and accelerate the development of sophisticated mechanics codes running on large parallel platforms. By providing generic, mostly computer-science related services, code developers are free to concentrate on mechanics algorithm research and development.

This presentation will focus on the recently implemented SIERRA framework service for parallel error handling. A description of the framework of a SIERRA application will be given. One fundamental abstraction within the SIERRA framework is the concept of a mechanics, the implementation of a physics model which is applied to a specific subset of a mesh. A mechanics module is supplied by an application to support multiple tightly or loosely coupled physics. Because of the support supplied by SIERRA, a mechanics module is developed independently of the complex communication patterns needed to support the inter-mesh coupling of mechanics, the dynamic load balancing, dynamic mesh redistribution due to h-adaptively and parallel error handling. Parallel error handling involves the separation of the concerns between local (single processor) mechanics-specific responsibilities and global (multi processor) tasks performed by the SIERRA services. A mechanics is responsible for defining a local error state and message. The SIERRA services are responsible for efficiently generating a consistent, globally synchronized error state from the local error state generated by the mechanics module. SIERRA manages output among processors and provides for proper termination of all processes in the event of a non-recoverable error. The error handling services are implemented simply and efficiently due to object oriented implementation of the mechanics module interface within SIERRA.

## References

- [1] H.C. Edwards and J.R. Stewart, “The SIERRA Computational Mechanics Framework: I. Parallel Infrastructure,” Proceedings of the Sixth U.S. National Congress on Computational Mechanics, 2001, pg 115.
- [2] J.R. Stewart and H.C. Edwards, “The SIERRA Computational Mechanics Framework: II. Parallel Data Transfer and H-Adaptivity Services,” Proceedings of the Sixth U.S. National Congress on Computational Mechanics, 2001, pg 116.