

## **HPEC 2002 abstract**

### **Data Reorganization Interface (DRI) -**

First, corresponding, and presenting author:  
Mr. Kenneth Cain Jr., Principal Software Engineer  
Mercury Computer Systems, Inc.  
Riverneck Road, Chelmsford, MA 01824-2820  
Phone ( 978) 967-1645, Fax (978) 244-0520, kcain@mc.com

Second author: Anthony Skjellum, President, MPI Software Technology, Inc. and  
Associate Professor of computer science, Mississippi State University.  
Mississippi State University and MPI Software Technology, Inc.  
101 Lafayette Street, Suite 33, Starkville, MS 39759  
Phone (662) 320-4300, Fax (662) 320-4301, tony@mpi-softtech.com

This presentation is *not* restricted to the US-only session.

#### **Topic Areas:**

- Middleware Libraries and Application Programming Interfaces
- Software Architecture, Reusability, Scalability, and Standards
- Algorithm Mapping to High Performance Architectures

The Data Reorganization Interface (DRI) emphasizes the need for high performance and easy-to-use “transpose and reshape” operations frequently needed in data-parallel signal and image processing applications. Existing middleware standards such as message passing interface (MPI) provide too little and relatively inefficient support for such operations, and provide a low level of abstraction that exposes tedious details of parallel programming to the end user. DRI enhances existing communication middleware (both standard and proprietary) with common syntax, semantics and powerful abstractions for data reorganization of clique and bipartite forms.

Planned transfer paradigms, together with abstractions related to datacubes and buffering semantics, deliver high performance and scalability to these essential operations. Additionally, significant ability to overlap communication and computation—an essential capability for efficient, modern multicomputer and cluster systems—is part and parcel of this specification and its implementations.

This presentation will cover the highlights of the first version of the DRI API, to be completed by the time of the workshop. Both the features included in DRI 1.0, and those that are deferred for later specification (including rationale and near term provisions for those that want the excluded functionality) will be discussed. The current value of DRI to embedded signal/image processing applications will be presented, in terms of its application programming interface (API) and how its implementations address some common application requirements. The presentation will include suggestions of ways the HPEC community can derive additional value from DRI by focusing on additional application requirements, either in the context of additional DRI specification or in subsequent HPEC software efforts.