**Workshop Description:**

The Workshop on Massively Parallel Processing is a forum that focuses on computer systems that utilize thousands of "processing elements" that work together to solve a single problem. As this is a workshop, the focus is on identifying new and novel ideas rather than proving incremental advances. By concurrently exploring architecture, programming models, algorithms and applications, this workshop seeks to advance the state-of-the-art of MPP systems. The Fifth Workshop on Massively Parallel Processing (WMPP'05) builds on the success of the four previous successful workshops, held as part of IPDPS'01, IPDPS'02, IPDPS'03, and IPDPS'04 respectively. WMPP has picked up momentum and is now pushing the limits of single day format. Technical presentations have come from industry (e.g. IBM), from national laboratories (e.g. UCAR, JPL), and from academic institutions from across the globe. The First and Second workshops featured invited keynote talks by Peter Kogge of Notre Dame and David Bader of the University of New Mexico, respectively. The Third workshop featured an invited keynote talk by Thomas Sterling of Cal Tech and NASA JPL. The Fourth workshop featured an invited keynote talk by Mootaz Elnozahy, Senior Manager and Master Inventor at IBM Research in Austin Texas.

Topics of interest include but are not limited to:

- **Architectures and Experimental Systems Architecturally**
  - How can a massive amount of "processing elements" communicate and coordinate their activities?
  - What is a "processing element" and how are they interconnected and coordinated?
  - What metrics can we use to demonstrate new and novel MPP systems?
  - How can we model and simulate MPP systems?

- **Parallelism, Programming Models and Algorithms**
  - Where is the parallelism (i.e. which part of what problems should be placed in the parallel paradigm)?
  - How can we program MPP systems?
  - How can we improve the productivity of MPP systems?
  - How can we exploit characteristics of certain problem domains to improve the system's performance, programmability and/or user productivity?
  - How can we debug MPP programs?

- **Perspectives, Positions and Problems**
  - Where are we now?
  - What have we learned positively and negatively about our current systems?
  - How are we doing with our large systems?
  - What must be changed to utilize "massively parallel" systems?
  - What are the problems that need to be addressed?

- **Future Factors?**
  - What are some of the future issues that we need to start considering now to achieve MPP systems?
  - How can we utilize power more efficiently?
  - How can systems handle unreliable components?
  - Can we program our way around some of these issues or must the system appear to be reliable?

---

**General Chair:**
Raymond Hoare, University of Pittsburgh, USA

**Program Chair:**
John Dorband, NASA, USA

**Steering Committee:**
Robert Walker, Kent State University, USA
Johnnie Baker, Kent State University, USA
Philip A. Wilsey, University of Cincinnati, USA

**Publicity Committee:**
Michael Scherger, Kent State University, USA

**Program Committee:**
Nael Abu-Ghazaleh, SUNY Binghamton, USA
David Andrews, University of Kansas, USA
Johnnie Baker, Kent State University, USA
Thomas Braunl, The University of Western Australia, Australia
Ray Hoare, University of Pittsburgh, USA
Mahmut Kandemir, Penn State University, USA
Peter Kogge, Notre Dame University, USA
H. J. Siegel, Colorado State University, USA
Theo Ungerer, University of Augsburg, Germany
Robert Walker, Kent State University, USA
Philip A. Wilsey, University of Cincinnati, USA
Jie Wu, Florida Atlantic University, USA