

Functional vs. Imperative Languages (Dataflow 2.0)

Functional languages and dataflow programming models get rediscovered by the programming community every 15 years or so because of their elegant expression of parallel constructs.

However, past forays into migrating the scientific community towards functional languages have met with strong resistance. If we want to move towards a more asynchronous execution models, then having bounded side-effects will be essential, which argues for stronger consideration of functional semantics and dataflow like concepts. This talk will discuss the merits of functional languages for simplifying parallelism in many dimensions. It will discuss the past successes and failures of functional approaches, lessons learned, emerging concepts for "modern" functional language ideas embodied in CILK, Ct, and library-based approaches such as MAGMA to see if there is a path forwards for broader adoption or if we will simply rediscover why these approaches failed to take root before.

Some background for this talk can be found in the SIAM PP Dataflow 2.0 minisymposium and UPCRC panel on the same topic

http://www1.nersc.gov/projects/SDSA/meetings/SIAM_PP08/

http://www.upcrc.illinois.edu/workshops/summit_feb2009/language/J_Shalf_PanelIII.pps