Charm++: What Have We Learned

Laxmikant V. Kalé

Department of Computer Science

University of Illinois

kale@cs.uiuc.edu

http://charm.cs.uiuc.edu

Genesis/History

- 1983-1986: Parallel Prolog, the Reduce-Or process model. First prolog system
- Recognition that many higher level systems such as Prolog need
  - a common base of support,
  - portability across parallel machines
- 1986-1990: Development of Charm/Chare Kernel: language concepts, implementation, load balancing.
- 1991 - Present : Refinements, libraries, tools, applications

Summary of Developments

- Tools:
  - Visual Programming,
  - Performance Feedback,
  - Expert Performance Analysis,
  - Automatic Runtime Optimizations
- Language Extensions:
  - Charm++: (1993): C++ based version
  - Dagger,
  - Structured Dagger,
  - Chare-Arrays.
  - DP-lib (DP-Charm)

Lessons

- Parsing/translation: major effort wasted
- Syntactic extensions: rethink
- Interoperability is essential: led to Converse
- C++ versus C
- Portability: recognized widely by now as useful.
- Message Driven Execution: the modularity/efficiency advantage
- Prioritization,
- Automatic dynamic load balancing: proved useful.