Future of Software and Software Research

Three Themes

William Mark
SRI International

New Visions for Software Design and Productivity Workshop
NSF
18 - 19 April 2001

Themes

- **Community Programming**
  - Software development is changing in some communities

    …and we can help
- **Vigilant Software**
  - Execution-time monitoring -- is it enough?
- **PhysInfo Co-Design**
  - Pervasive computing requires a new design paradigm

Community Programming

Practitioners express their expertise in terms of shared software

- **Practitioner-programmers**
  - Domain experts with some software training/experience
  - Community-centered
  - e.g., ESCOT
- **Frameworks**
  - Need to solve enough of the problem

    ...to make this old dream work
- **Open Software**
  - Right economics
  - Not a necessary condition

Vigilant Software

Make monitoring viable

- **Design for "monitorability"**
  - E.g., explicitly modularize for probabilistic stream-checking
- **Design the hardware too**
  - Use all those new gates!
    - Distributed "co-processors" for monitoring

PhysInfo Co-Design

Physical **constraints must be first class citizens**

- **Computation in the real world**
  - Embedded world imposes constraints on the information system
    - time, distance, energy, uncertainty...
    - that's what "embedded" means
  - Current software design has no place for physical constraints
    - programmers incorporate them (implicitly, inconsistently, unreliably) in their code
- **New computational models**
  - Embody physical constraints the way (some) models embody time
    - can we enforce "physics" in specs?