After ICSE 2004

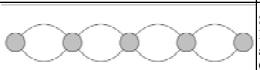
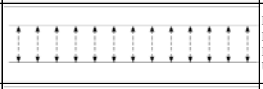
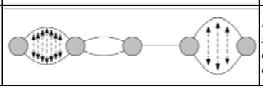
http://conferences.iee.org/icse2004/

Workshop on Directions in Software Engineering Environments, WoDiSEE2004

- Mehra, Grundy, Hosking (University of Auckland): Support Collaborative Software Design with a Plug-in, Web Services-based Architecture
- Van der Hoek, Redmiles et al. (University of California, Irvine): Continuous Coordination: A New Paradigm for Collaborative Software Engineering Tools
- Robillard, Murphy (University of British Columbia): Program Navigation Analysis to Support Task-aware Software Development Environments

http://www.cs.auckland.ac.nz/%7Eherm/WoDiSEE2004/

Formal, Informal, and Continuous Coordination

	Conceptual Visualization	*Strengths*	*Weaknesses*
Formal process-based coordination		Scalable; Control; Insulation from other activities; Group-centric	Resynchronization problems; Insulation becomes isolation
Informal, awareness-based coordination		Flexible; Promotes synergy; Raises awareness; User-centric	Not scalable; Requires extensive human intermediation
Continuous coordination		*Expected to be the strengths of both formal and informal coordination*	*To be discovered by future research*

Keynotes

- Richard Stallman, Free Software Foundation (http://www.stallman.org/)
- Karl Lieberherr, Northeastern University (http://www.ccs.neu.edu/home/lieber/)
- Janet Thornton, European Bioinformatics Institute (http://www.ebi.ac.uk/)

Panel on Design:
Supporting Reflective Practitioners

David Redmiles (University of California, Irvine)
Kumiyo Nakakoji (University of Tokyo)
Gerhard Fischer (University of Colorado at Boulder)
Yunwen Ye (SRA Key Technology Laboratories, Tokyo)
Alistair Sutcliffe (University of Manchester)
Sol Greenspan (National Science Foundation, Washington)

Supporting Reflective Practitioners

- Donald Schön
  - Professional workers are reflective practitioners
  - Design is a process of "reflection in action"
- Herbert Simon
  - Designers change existing states into preferred states
  - A "science of the artificial," including bounded rationality and a curriculum for design
- Lucy Suchman
  - Plans are resources
  - Action is situated
- Terry Winograd
  - Design is an activity of interpretation
- Morton Kyng and Joan Greenbaum
  - Social complexity is a greater challenge to design than technical complexity
- Fred Brooks
  - "Essential" and "accidental" elements of design

Stimulate Your Thinking About

- Different perspectives of design
- End users as reflective practitioners
- Software developers as reflective practitioners
- Design as a human activity

Overview of the Panel

- Design is a human centered activity, a social activity, not in isolation
- Ways to see / recognize / identify / name / frame what we normally do not see
  - Rational
  - Non-rational
- Education is key, new curricula are required

Paper: Preserving Versatility in Event-Based Middleware

- **Extensibility** is the ability to augment middleware with new functionality.
- **Functional Configurability** is the ability to combine and select different functionality corresponding to different application needs or heterogeneous hardware and software constraints.
- **Distribution Configurability** defines the selection of the place to perform event processing, whether in the publisher (producer), the event router (notification service), or in the subscriber (consumer).
- **Reuse** supports common requirements by referring to existing designs and implementations of services and components.
- **Usability** refers to the ease with which software engineers apply an infrastructure as well as the usefulness of that infrastructure's functionality.
- **Interoperability** is the ability to integrate new services.
- **Scalability** is the ability to support design and implementation choices required to cope with issues of magnitude, issues of quantity and size.

Publish/subscribe process decomposition