EuroTools SIG on
Problem-Solving Environments

Coordinator: José C. Cunha, Univ. Nova de Lisboa, Portugal
(Contact: jcc@di.fct.unl.pt)

31 August 1999, EuroPar’99, Toulouse

Index

- 1<sup>st</sup> part: Objectives
- 2<sup>nd</sup> part: Motivations
- 3<sup>rd</sup> part: Activities
Slide 2

Objectives

- To help end-users and tool developers to communicate and exchange ideas
- To survey the requirements, approaches, experiments and results of ongoing development and use of PSEs
- To address future challenges and open issues for building future generations of PSEs

Slide 3

Problem-Solving Environments

- Integrated environment supporting:
  - entire life cycle
    * development and execution steps
    * to solve a target class of problems in a given application domain
    * with easy access by an end-user,
Important Goals

- Provide transparency to the end-user concerning problem formulation.
- Allow evaluation of the more adequate solutions and problem-solving strategies.
- Allow the development of rapid prototypes to ease the experimentation with specific solutions.
- Allow the user to learn from experience.

Development Steps

- Help the user in producing a problem specification and its rapid prototyping so that it may be submitted to execution.
  - Rapid prototyping
  - Dependent on a specific domain
  - Expert assistance
**Execution Steps**

- Allow the user to interact with an ongoing experiment. To support online/offline observation and control of scientific experiments / simulation processes
- Activities performed on multiple heterogeneous components (application-specific and generic tools):
  - selection, evaluation and testing of individual components
  - configuration, activation, interconnection
  - monitoring, controlling

**Hetereogeneous Collection of Interconnected Components**

- Sequential, Parallel, Distributed Problem Solvers
- Tools for data and result processing, interpretation, visualization
- Interactive computational steering: user and agent driven
- Online interaction with large databases and other devices
- Multidisciplinary nature:
  - Heterogeneous / hybrid components / models
  - Interactions among multiple users, collaborative environments
Figure 1: Conceptual Layers

- **Application domain**
- **Tools**
  - Formal mechanisms
  - Coordination methods
  - Resource managing/Interconnection services
  - Monitoring and control layer
  - Heterogeneous hardware/software
- **PSE**
Activities in the EuroTools SIG on PSE

Main objective: To help end-users and tool developers to communicate and exchange ideas. To survey the requirements, approaches, experiments and results of ongoing development and use of PSEs. To address future challenges and open issues for building future generations of PSEs.

- Short term: Experiments in Building PSEs
  - Cooperation with experts from specific domains
  - Identification of user/application requirements
  - Early and incremental development of prototypes
  - Quick feedback from end-users

Make these PSE evolve towards advanced PSE to ease development and execution of complex applications.
To survey the requirements, approaches, experiments and results of ongoing development and use of PSEs.

To survey existing PSEs for specific problem domains, not only in Science and Engineering, but also in other domains.

Impact of PSEs upon Education and Learning at all levels of studies.

The EuroTools PSE home page as the main point to disseminate such information and to establish links to related sites, projects and relevant events in this topic:

URL: http://www.irisa.fr/EuroTools

Medium / Long term: To address future challenges and open issues for building future generations of PSEs

- To enable a cost effective exploitation of software and hardware facilities
- Using parallel and distributed computing
- Component-based programming
- Intelligent knowledge processing and discovery
- Virtual laboratories
- Collaborative computing
- Agent coordination models
Summary of Open Issues

Intelligence and Expert Assistance. Advising, explaining, expert systems / tools to assist the user during the development and execution steps.

High Interactive PSEs. Increased flexibility in user and component interaction. User and Agent Based Observation and Control. Computational Steering. Advanced Visualization.

Collaborative Environments. Multiple cooperative tools and users, sharing the state and controlling an ongoing experiment in collaborative environments.

Software Architectures for PSEs. To support the development of PSEs:

1. Formal specification of PSE components and their interactions.
2. Tools to reason about global system properties.
3. Tools to support transformation between software levels.
4. Middleware and supporting infrastructures: local and large scale operations (Metacomputing).

Generating PSEs. Generic vs specific architectures and tools for PSEs

Impact upon Education Methods. Impact of PSE upon Education and Learning

- Teaching how to build PSEs related to Computational Science and Engineering.
- Using PSE in Education at all levels of studies.