Programming Exercises Evaluation Systems: An Interoperability Survey

Ricardo Queirós and José Paulo Leal
Faculty of Sciences of University of Porto

4th International Conference on Computer Supported Education (CSEDU’12)
Porto, Portugal

April 16, 2012
Outline

1. Introduction
2. Related Work
3. Interoperability Analysis
4. Synthesis
5. Conclusion
Context

- Learning programming requires solving programming exercises.
- Manual assessment of exercises:
  - Time consuming - teachers need to assess a large number of exercises (e.g. large classes)
  - Error prone - hinders the consistency and accuracy of assessment results
- This issue triggered the appearance of Programming Exercises Evaluation Systems
Programming Exercises Evaluation Systems

- Automatic tools to
  - grade students’ programming exercises
  - give feedback on the quality of students’ solutions

- Used on different learning scenarios:
  - Curricular (e.g. practical classes, assignments and exams)
  - Competitive (e.g. programming contests)
    - IOI - for secondary school students
    - ACM-ICPC - for university students
    - IEEE Extreme - for IEEE student members

- Examples:
  - AutoGrader, BOSS, Hustoj, Mooshak, WEB-CAT, etc.
Several surveys enumerate and compare PES features...

- how the analysis of the code is made
- how the tests are defined
- how grades are calculated

...but neglects the interoperability feature. Organized at two levels:

- content - how a programming exercise is described
- communication - how it should be shared among systems:
  - Learning Management Systems
  - Learning Objects Repositories
  - Integrated Development Environments
  - ...
Work description

- An interoperability survey on existing PES

Main goal

- to gather information on the interoperability features of existent PES
- to compare them regarding a set of predefined criteria such as
  - content specification
  - standard interaction with other tools

The intended benefit of this survey is twofold

- to fill the gap on PES interoperability features found in most surveys
- to help instructors, educational practitioners and developers when they have to choose a PES to integrate in their e-Learning environments
Evolution of assessment tools

- **Evolution of # systems**

- **Evolution of features**

---

**Early Assessment Systems (1960 - 1985)**
- Punched cards
- Support for one language (e.g., Algol)
- No feedback
- No administration facilities

- Command-line interfaces (manual operation of scripts)
- Support for few languages (e.g., C, JAVA)
- Limited feedback
- Content management
- Static analysis
- Student automated testing
- Grading-support system
- Competitive learning (e.g., contests)

**Web-Oriented Systems (2000 - ...)**
- Web-based interfaces
- Multi-languages (Prolog, SQL, FORTRAN)
- Richer and incremental feedback
- Course/student administration facilities
- Sophisticated testing approaches
- Automatic test generation
- Plagiarism detection
- Service-oriented
- Integration with LMS
Recent surveys

Five surveys:

- **Douce et al. (2005)**
  - Methodology: details features of PES organized in 3 generations
  - Trend: evaluation of GUI programs, meta-testing (evaluation of the students’ tests), SOA and use of interoperability standards

- **Kirsti AlaMutka (2005)**
  - Methodology: organizes PES in dynamic and static evaluators
  - Trend: content and communication standardization.

- **Liang et al. (2009)**
  - Methodology: details dynamic and static analysis methods of PES
  - Trends: security, algorithms for automatic generation of test data and content standardization

- **Ihantola et al. (2010)**
  - Methodology: discuss PES (2006-2010) features (e.g. tests definition, resubmission policies and security)
  - Trends: integration with LMS and assessment of web applications

- **Romli et al. (2010)**
  - Methodology: approaches for test data generation
  - Trends: test data generation techniques, interoperability and security
Survey existing PES regarding their interoperability features

Multi-criteria approach for the selection of tools
- based on its effective use (flexibility on the exercises and users management)
- selected tools: AutoGrader, BOSS2, CourseMaker, CTPracticals, DOMJudge, EduComponents, GAME, HUSTOJ, Moe, Mooshak, Peach3, Submit!, USACO, Verkkoke, Web-CAT

Multi-criteria approach for the selection of facets
- based on surveys trends and our background
- selected facets: programming exercises, users and assessment results
These facets are synchronized with PES main objective

to evaluate a **user’s attempt to solve a programming exercise** and produce an **assessment result**.

Each facet includes three interoperability maturity levels:

- Level 0 - manual configuration of data
- Level 1 - data import/export
- Level 2 - services invocation
Programming Exercises

Survey levels in the **Programming Exercises facet**
- Level 0 - manual configuration of exercises
- Level 1 - import/export of exercises
- Level 2 - integration with repository services

<table>
<thead>
<tr>
<th>Systems</th>
<th>Level 0</th>
<th>Level 1</th>
<th>Level 2</th>
</tr>
</thead>
<tbody>
<tr>
<td>AutoGrader</td>
<td>F</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>BOSS2</td>
<td>F</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>CourseMaker</td>
<td>F</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>CTPracticals</td>
<td>F</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>DOMJudge</td>
<td>F</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>EduComponents</td>
<td>F</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>GAME</td>
<td>F</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>HUSTOJ</td>
<td>F</td>
<td>P</td>
<td>-</td>
</tr>
<tr>
<td>Moe</td>
<td>F</td>
<td>P</td>
<td>-</td>
</tr>
<tr>
<td>Mooshak</td>
<td>F</td>
<td>F</td>
<td>F</td>
</tr>
<tr>
<td>Peach3</td>
<td>F</td>
<td>P</td>
<td>-</td>
</tr>
<tr>
<td>Submit!</td>
<td>F</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>USACO</td>
<td>F</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>Verkkoke</td>
<td>F</td>
<td>F</td>
<td>-</td>
</tr>
<tr>
<td>Web-CAT</td>
<td>F</td>
<td>F</td>
<td>P</td>
</tr>
</tbody>
</table>

- All systems support the configuration of exercises
- 6 tools export exercises; 3 bidirectional support; few systems use exercises formats
- 2 tools support communication with repositories through SOA

Queirós & Leal

PES: An Interoperability Survey

16-04-2012
Users

Survey levels in the **Users facet**

- Level 0 - manual configuration of users;
- Level 1 - import/export of users;
- Level 2 - integration with user directories services (authentication) and AMS (authorization)

<table>
<thead>
<tr>
<th>Systems</th>
<th>Level 0</th>
<th>Level 1</th>
<th>Level 2</th>
</tr>
</thead>
<tbody>
<tr>
<td>AutoGrader</td>
<td>F</td>
<td>F</td>
<td>P</td>
</tr>
<tr>
<td>BOSS2</td>
<td>F</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>CourseMaker</td>
<td>F</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>CTPracticals</td>
<td>F</td>
<td>F</td>
<td>P</td>
</tr>
<tr>
<td>DOMJudge</td>
<td>F</td>
<td>F</td>
<td>P</td>
</tr>
<tr>
<td>EduComponents</td>
<td>F</td>
<td>F</td>
<td>P</td>
</tr>
<tr>
<td>GAME</td>
<td>F</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>HUSTOJ</td>
<td>F</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>Moe</td>
<td>F</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>Mooshak</td>
<td>F</td>
<td>F</td>
<td>P</td>
</tr>
<tr>
<td>Peach3</td>
<td>F</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>Submit!</td>
<td>F</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>USACO</td>
<td>F</td>
<td>F</td>
<td>-</td>
</tr>
<tr>
<td>Verkkoke</td>
<td>F</td>
<td>F</td>
<td>-</td>
</tr>
<tr>
<td>Web-CAT</td>
<td>F</td>
<td>F</td>
<td>-</td>
</tr>
</tbody>
</table>

- All systems support the manual configuration of users
- 8 tools allow the import/export of users in non-standard formats
- 5 tools communicates with authentication services (LDAP)
Assessment results

Survey levels in the **Assessment results facet**
- Level 0 - visualization of evaluation results
- Level 1 - export of assessment results
- Level 2 - integration with LMS

<table>
<thead>
<tr>
<th>Systems</th>
<th>Level 0</th>
<th>Level 1</th>
<th>Level 2</th>
</tr>
</thead>
<tbody>
<tr>
<td>AutoGrader</td>
<td>F</td>
<td>F</td>
<td>P</td>
</tr>
<tr>
<td>BOSS2</td>
<td>F</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>CourseMaker</td>
<td>F</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>CTPracticals</td>
<td>F</td>
<td>F</td>
<td>P</td>
</tr>
<tr>
<td>DOMJudge</td>
<td>F</td>
<td>F</td>
<td>-</td>
</tr>
<tr>
<td>EduComponents</td>
<td>F</td>
<td>F</td>
<td>P</td>
</tr>
<tr>
<td>GAME</td>
<td>F</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>HUSTOJ</td>
<td>F</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>Moe</td>
<td>F</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>Mooshak</td>
<td>F</td>
<td>F</td>
<td>-</td>
</tr>
<tr>
<td>Peach3</td>
<td>F</td>
<td>F</td>
<td>-</td>
</tr>
<tr>
<td>Submit!</td>
<td>F</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>USACO</td>
<td>F</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>Verkkoke</td>
<td>F</td>
<td>F</td>
<td>P</td>
</tr>
<tr>
<td>Web-CAT</td>
<td>F</td>
<td>F</td>
<td>-</td>
</tr>
</tbody>
</table>

- All systems present the evaluation results to users
- The majority allows its exportation in non-standard formats
- 4 systems support the communication with LMS
Mooshak, Web-CAT, Verkkoke offer the best interoperability levels
Half of the systems studied did not reach 50% of the maturity rate
There are a lot to do regarding PES interoperability

% of maturity
Interoperability features coverage

- There is no specific trend on interoperability features.
- Nevertheless, programming exercises facet presents the lower results.
- Need to standardize:
  - the description of programming exercises.
  - the communication of PES with other systems (e.g., repositories).
This work presents an interoperability survey on PES.

Based on a multi-criteria approach we select 15 tools and organized the survey based on three interoperability facets:
- programming exercises
- users
- assessment results

For each facet each PES was characterised based on its interoperability level.

Two issues were detected that can hinder the proliferation of PES:
- the lack of standards to describe programming exercises
- the lack of standards to communicate with other e-Learning systems

The benefit of this survey was:
- to fill the gap on PES interoperability features found in most surveys
- to help instructors, practitioners and developers when they have to choose a PES to integrate in their e-Learning environments.