

# crimsonHex: an interoperable repository of programming problems

José Paulo Leal    and    Ricardo Queirós  
zp@dcc.fc.up.pt      ricardo.queiros@eu.ipp.pt

The University of Valladolid (UVA) maintains an archive of programming problems with problem sets of various ICPC competitions, including all world finals and regional of the past seven years. The EduJudge project aims to integrate this archive into an effective educational environment and an eLearning platform is being developed for that purpose. The EduJudge platform adheres to a service oriented architectural model and integrates three main types of services: Learning Objects Repository (LOR), to store programming exercises; Evaluation Engine (EE), to evaluate and grade the students attempts; and Learning Management System (LMS), to manage the presentation of exercises to students.

This paper presents the current state of the LOR of EduJudge. We start with the definition of programming problems as learning objects. Then we present the design and implementation of crimsonHex, an interoperable repository of programming problems, starting with its overall architecture and proceeding to its core functions and users interface.

The definition of programming problems as learning objects is based on the IMS CP specification. An IMS CP learning object assembles resources and meta-data into a distribution medium, in our case a file archive in zip format, with its content described in a manifest. Meta-data in the manifest file conforms with the IEEE LOM schema, but can be extended with other schemata. We used this possibility and defined specialized meta-data for programming problems, to identify particular types of resources, such as problem descriptions, test cases and program solutions, among others.

The architecture of crimsonHex is driven by the need for interoperability. An API is the heart of its architecture, defining functions implemented by a core component and required both by other crimsonHex components and external consumers of crimsonHex's services.

The core component of the crimsonHex provides a minimal set of functions based on the IMS DRI specification. This specification prescribes a list of basic functions exposed by SOAP web services. We extended this specification both with a RESTfull interface and with new functions, for reporting learning objects' usage data and managing the structure of the repository. The latest type is intended mostly for the user interface layer.

The design of the users interface followed usability guidelines. First of all, to cope with the diversity of users we identified several tasks profiles, including: the archivist responsible for the repository, the author, the reviewers and the consumers of learning objects. For sake of simplicity and consistency the layout of the users interface follows and intuitive single-page model. The web interface has two main regions: browse and view. The former allows the user to navigate through the repository structure and to select items - collections and LOs - using a tree-based control. The later aggregates commands over the selected items, using a tab page control.

The paper describes also the implementation of both layers of crimsonHex. The core functions are based on an XML database and required a validation module to check IMS CP conformance. The users interface layer uses an Ajax toolkit to implement the single-page model chosen for the interface.

A few tasks remain to complete this work-package of the EduJudge project. We conclude this paper with the approaches planned for these tasks, with populating the repository using programming problems from legacy repositories on the top of the list.