e-Assessment system architecture

General

The evolution of the architecture of e-assessment systems is generally similar to evolution of e-learning platforms or *learning management systems*. Such systems usually provide a holistic environment for managing and delivering educational experiences including assessment of their outcomes.[1] Examples of such systems are

- *Moodle*,
- *Sakai Project*,
- *Whiteboard*,
- *Desire2Learn*,
- *Gradepoint*,
- *WebCT/BlackBoard*.

Still, there are also a number of standalone e-assessment systems like

- *TCExam*,
- CASEC[2], or
- *The Questionmark*.

Evolution of the architecture

Evolution of the LMS architecture[3] and the evolution of the standalone e-assessment systems[4] can be described in three stages:

First generation

First generation of solutions (~ 1993-1999) for e-learning and e-assessment were monolithic black-box systems, usually oriented only on a specific course and offering very limited user-tracking. Examples of such systems were first versions of WebCT and Blackboard.[5]

Second generation

Second generation of solutions (~ 1999-) for e-learning and e-assessment like Moodle or Sakai offer more modular architectural design enabling easier integration of new functionality. Standards like SCROM, IMS Content Packaging, and IMS Learning Design were developed to support the ability to exchange courses or parts of the courses. In this type of systems, content is usually being separated from tools enabling them to be used as platforms for creating different courses and assessments.[6]

Service-oriented architecture

As a common conclusion of a number of recommended LMS frameworks (including JISC e-Learning Technical Framework (ELF), IMS Abstract Framework, and Open Knowledge Initiative), service-based architecture is expected to be the next architectural advance of e-learning and e-assessment systems.

Service-oriented computing (SOC) is

- "*an application architecture within which all functions are defined as independent **services** with well-defined invokable interfaces, which can be called in defined sequences to form business processes.*"[7]

The *services* in an SOA have the following characteristics:[8]

- they are autonomous, and external components do not know nor care how they perform their function,
- they have well defined invokable interfaces making it irrelevant weather they are local or remote.

Some of the reasons for adopting service-oriented architecture are[9]:

- development of a "coherent diversity"[10]
- easier collaboration between institutions,
- avoiding of limitations using single vendor solutions,
- easier connecting of components in custom and new ways,
- possibility of replacing of one service with another to offer same functionality in new ways.

[1], [3], [5], [6]

Dagger, D., A. O'Connor, S. Lawless, E. Walsh, and V. P Wade. Service-Oriented E-Learning Platforms: From Monolithic Systems to Flexible Services. IEEE Internet Computing 11, no. 3: 28-35, June 2007.
[2]

http://www.paaljapan.org/resources/proceedings/PAAL11/pdfs/11.pdf
[4], [9]

Armenski, Goce, and Marjan Gusev. The Architecture of an 'Ultimate' e-Assessment System. Association for Information and Communication Technologies ICT-ACT, 2009.
[7]

Channabasavaiah, K., Holley, K. and Edward M. Tuggle, Jr. Migrating to a service-oriented architecture. On demand operating environment solutions, White paper. 2004.
[8]

Papazoglou, Mike P, and et al. Service oriented architectures: approaches, technologies and research issues, 2007.
[10]

http://www.elearning.ac.uk/features/pedagandws