Change and Evolution in Web-based Learning Technology Systems

Claus Pahl
Dublin City University
School of Computing
Centre for Next Generation Localization (CNGL)
Dublin 9, Ireland
cpahl@computing.dcu.ie

Abstract: With the inception of the Web now being more than 20 years ago, many Web-based learning technology systems (LTS) have had a long life and have undergone many changes, both affect content and infrastructure technologies. In an earlier paper (Pahl, 2003), we introduced a facetted dimension model to capture the various factors of change and suggested a change-aware design of LTS. The purpose of the paper here is to add empirical results to the facets in order to show the relevance of the dimension model by specifically looking at an LTS that has been developed, maintained and extended over a period of more than 20 years. A key observation is an unexpectedly high impact of environment constraints, but also of new opportunities emerging over time.

Introduction

The World-Wide Web has changed the way teaching and learning is facilitated using digital means. With the inception of the Web now being more than 20 years ago, many Web-based learning technology systems (LTS) have had a long life and have undergone many changes, both affect content and infrastructure technologies (Palmer and Tulloch, 2001). In an earlier paper (Pahl, 2003), we introduced a facetted dimension model to capture the various factors of change and suggested a change-aware design of LTS. The purpose of the paper here is to add empirical results to the facets in order to show the relevance of the dimension model.

We specifically look at an LTS that has been developed, maintained and extended over a period of more than 20 years (Smeaton, 1991; Smeaton and Crimmins, 1997; Pahl, Barrett, and Kenny, 2004; Kenny and Pahl, 2008). During this period some notable changes (beyond the expected content updates to reflect changes in the subject domain and similar problems) include different teachers, an integration of content and infrastructure with the university's learning technology platform, the export of learning objects to sharing platforms and repositories and the shared delivery of with several universities across different countries. A key observation is an unexpectedly high impact of environmental constraints, i.e. aspects that define how the LTS relates to its technical and organizational environment, but also of new opportunities emerging over time (Devedžić, 2006).

Eight years on after the publication of our LTS change and evolution discussion paper (Pahl, 2003), the proposed solution shall be revisited. Some issues include the question whether the proposed dimension model is still adequate. Changes do not only happen as a once-off activity, but are part of a long lifecycle. What is the impact of changes in technology or general teaching and learning research on LTS? What are the long-term costs? How effective would a methodology for change-aware LTS design be? We outline the lessons we learned using a concrete LTS over two decades.

Learning Technology Systems and IDLE

Learning technology systems (LTS) are computer-based (now often Web-based) environments that provide learning content through a range of specific (Web-based) software features. We look at the evolution of an interactive database learning and training environment, embedded into an online courseware system – called the Interactive Database Learning Environment IDLE (Pahl, Barrett, and Kenny, 2004; Kenny and Pahl, 2005) – for a second year undergraduate computing course. A language training part forms a central part of this course as database programming is one of the core learning objectives of the course. Programming (i.e. defining, updating, and querying database tables) is a skill that needs to be trained by the student. Understanding and mastering the overall development process of a database application is equally important. Database programming in the language SQL also requires conceptual understanding of the underlying data model with its structures, operations, and constraints. The IDLE LTS provides four different features for the different aspects of SQL programming:

- Conceptual knowledge. Conceptual knowledge is presented in a virtual lecture system.
- Procedural knowledge. SQL and parts of its underlying data model are about the execution of instructions. Procedural knowledge is presented in an animated tutorial system.
- Programming skills. SQL programming is the core activity, supported by an interactive tutorial that guides the student through exercises to be worked on within the system.
- Development skills. SQL programming is part of the overall database application development process, which supported by an integrated lab environment with modelling, programming, and analysis features.

The system is implemented as a Web-based system, i.e. only a Web-browser and standard plug-ins are required by the student. Conceptual understanding of principles and concepts of the topic is of course required before practical work can start. However, the aim of the tutorial system is to allow students to go quickly into the practical features by supporting a learning-by-discovery style, allowing them to acquire skills, but also to construct and deepen their conceptual knowledge through activities in meaningful and realistic problems. Consequently, the practical features are well linked to the respective background.

IDLE is a feature-rich LTS that uses multimedia and interactive technologies for student access, but also an extensive support software architecture on the server side. In addition to its primary role of being an LTS aiming at improving the learning experience, IDLE is also an instrument and mechanism for research into learning technology and software engineering, technology transfer, and collaboration and internationalization – which has affected how IDLE has change and evolved.

Change and Evolution in LTS

Change Dimensions

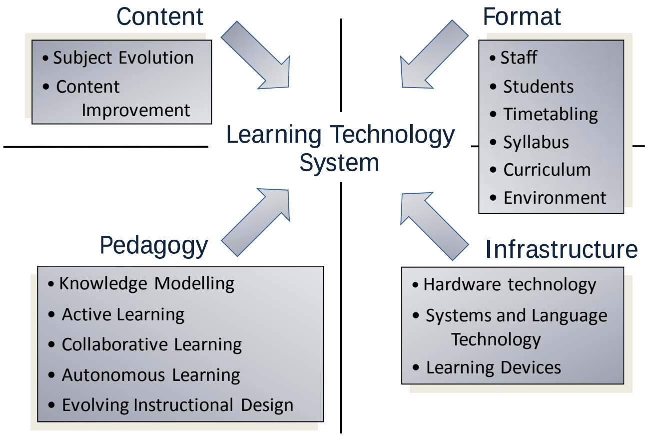

Figure 1. Change and Evolution Dimensions and Facets.

In (Pahl, 2003), we presented a dimension model that categorises change factors into a number of facets. This categorization serves not only as a change and evolution analysis and impact determination technique (post-development), but also as a change-aware development/methodology (pre-development). The change and evolution models captures the following dimensions (see Fig.1):

- *Content* – the subject-oriented perspective – refers to the subject taught and the representation of knowledge in the LTS and captures changes relating to the subject that is taught.
- *Format* – the organisational perspective – comprises attributes determined by the institutional context: curriculum, syllabus, staffing, etc. and captures changes related to staff and students involved or to curriculum and syllabus.
- *Infrastructure* – the technical perspective – relates to the hardware and software environment in which the LTS is deployed and captures changes due to developments in hardware/software technology or learning devices.

- *Pedagogy* – the educational perspective – refers to the instructional design of the LTS determining in which way the course is taught and captures the evolution of teaching and learning in computer-supported environments.

In (Pahl, 2003), we looked at the LTS-internal perspective, i.e. the effect of change on an LTS. Here, we apply its results and look at an external higher-level perspective, i.e. at drivers of change and how impact is dealt with.

Impact of Change and Evolution

IDLE has changed continuously over the past 20 years – caused by different factors which shall be summarized using the detailed change dimension model from Fig. 1. Each change aspect shall be categorized in terms of its impact (high/medium/low) on the IDLE LTS (approximating the required change effort/costs).

Content [low – continuous, but moderate]

- *Subject evolution*: The course subject itself evolves – an external factor.
  - *Low* – minor additions to content have been made
- *Content improvement*: Content is changed in order to improve the material in a planned process.
  - *Low* – minor corrections to address presentation and technical issues have been carried out

Format [low – course shortened as the most important change in this category]

- *Staff*: Changes relating to educators, course developers, or technical support staff.
  - *Medium* – in total 4 lecturers – with 3 of them familiar (of which 2 permanent academics and 1 replacement instructor) and 1 unfamiliar with the underlying technology) – staff training has become an issue over time as recent instructors were not familiar with underlying technology
- *Students*: The student body changes in terms of numbers, qualifications, or mode of learning.
  - *Low to medium* – profile changed slightly (no effect after one significant increase of numbers affecting server support)
- *Timetabling*: Changes related to where and when a course takes place.
  - *Medium* – part-time evening delivery and support at that time has been a problem for a period
- *Syllabus*: Content and organisation of the course content can change.
  - *Low* – Content has been shortened as part of a degree restructuring
- *Curriculum*: Organisational needs require changes in level, extent, or prerequisites of courses.
  - *Medium* – changed cause syllabus update (see above)
- *Environment*: Issues such as the legal or financial environment require changes.
  - *Low* – none in our case, but possible issues could be behaviour tracing of identifiable students

Infrastructure [high – significant hardware and software changes]

- *Hardware technology*: Communications and network technology, computing power, and computer platform have constantly improved.
  - *High* – enabling and disabling changes (more powerful database server, but also outdated server support without maintenance contract)
- *Systems and language technology*: Minor changes within the technology, technology leaps, legacy or pre-eminent technologies are frequent issues.
  - *High* – new technologies (e.g. flash, XML, Semantic Web) emerged and have been incorporated.
- *Learning devices*: Software and hardware devices such as smart objects, information infrastructures and virtual environments serve as learning devices.
  - *Low* – new devices (e.g. mobile) have not been supported, but a national learning object repository can be considered as another content storage and delivery platform.

Pedagogy [medium – from online to blended]

- *Knowledge modelling*: Acquisition, modelling of and access to educational knowledge.
  - *High* – in the context of the provision of LTS components through a national repository, explicit annotation and access infrastructure needed to be provided (packaging)
- *Active learning*: Engaging the student through interactive systems.

- *High* – initially considerable investment into developing active learning support (tutorials and labs)

- *Collaborative learning*: Supporting communication and collaboration through communications systems.
  - *Low* – not done
- *Autonomous learning*: Personalisation and independent learning through adaptive systems.
  - *High* – instruction and support develop with an emphasis of independent learning at later lifecyle stages (including automated correction and feedback mechanisms)
- *Evolving instructional design*: Planned evolution integrated in design through course evaluation.
  - *Medium* – often neglected through a research prototype focus, only partly compensated by the instructors' technical expertise.

Observations and Lessons learned

We now look at the sustainability of change and factors that influence this sustainability by analyzing the change we recorded.

Observations. The dimension model has proven itself to be useful. All actual changes that have occurred do fit into the categorisation scheme. We would consider the dimension taxonomy stable, apart from the pedagogy dimension which is a container to capture important learning technology trends. An important observation is that the impact of change factors varies:

- The more research-driven the LTS is, the more important are the infrastructure and pedagogy dimensions (and consequently the more costs/effort is required).
- Equally, the longer the life of the LTS, the more important these two dimensions become.

That means that two factors determine the impact significantly: the role of the LTS and its age, i.e. external factors need to be correlated with internal change impact on the LTS.

What can be observed is that the TLS lifecycle reflects a software ageing process. Software ageing is characterized by increasing maintenance costs. The maintenance and evolution process causes ageing, negatively impacted through research development prototyping before proper systematic production methodologies are applied.

In addition to this internal perspective, we can also look at the IDLE lifecycle in the context of developments in two domains: learning technology research (LT) and Web and Internet software technology development (ST) – see Fig. 2. Most significant trends have been incorporated – some adopted early (even before they became mainstream), indicated through shorter arrows, some adopted later, indicated through longer arrows. Research, however, causes a form of opportunistic evolution (exploiting a technology opportunity to follow educational trends). This is in principle costly, but can to some extent alleviated through reuse and knowledge transfer, e.g. by incorporating externally developed sharable content.

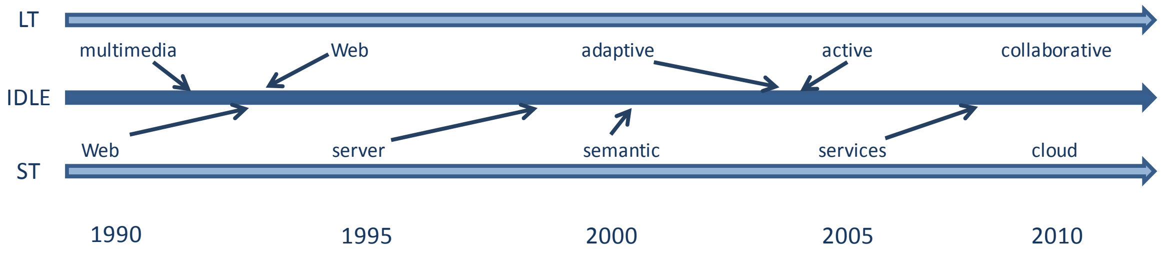

Figure 2. IDLE Evolution in the Context of Learning Technology LT and Software Technology ST Evolution.

Success and Limitations. Overall, IDLE has been a success. In educational terms, student attainment in written examination and practical work has improved continuously while we extended IDLE (Kenny and Pahl, 2005). The IDLE system and approach to active learning have also received high appreciation by the students (more than 1000 students used the system). Another benefit is the fact that IDLE has introduced many of our students to online and blended life-long learning technologies, which many will encounter again during further education throughout their careers. In terms of research, IDLE as a research prototype has led to a significant number of publications, established the reputation of the research group and led to various contacts and collaborations with other researchers, groups and institutions.

However, high-end features in LTS such as IDLE are showpieces that raise the profile of the developers. IDLE is a system that has turned out to be costly and difficult to maintain. New investments and new developments are required if new concepts are to be implemented. For many years, we have attempted to include new features and update the infrastructure, but eventually, the cost of maintaining the system infrastructure and training have cause us recently to discontinue the support of the system. If a system cannot be developed further as in our case, sometimes a community might be willing to continue development and support (for instance in an open-source setting).

Costs. Responsible for the eventual winding down of the system are the increasing maintenance costs. The development costs have been largely covered through funded research projects, but has also significantly relied on the unpaid contributions of the faculty members and project students involved. However, maintenance and other activities interoperability have not been as adequately covered:

- Maintenance of hardware and software – both have incurred costs, particularly after more than 10 years in operation have resulted in, first, the removal of some software features and the retirement of the whole server system. Training as a maintenance concern on a human level has also been a problem.
- Exchange and collaboration have partly been supported, through grants allowing us to make some components available as reusable learning objects for a national learning object repository, which in turn allowed us to deliver the course in collaboration with universities in two other countries.

Without an institutional strategy, such systems are difficult to maintain over longer periods.

The Future. While IDLE as a system is not supported any further, at least part of it will remain in the form of reusable learning objects in a national learning object repository. This makes content and infrastructure available to other instructors (subject to some IP agreement). Another avenue that might be open to similar systems is commercialization. IDLE is not a general-purpose LTS, but a very subject-specific system. However, as the example of a similar system by the University of Canterbury in New Zealand show (see http://www.aw-bc.com/databaseplace/), the commercialization of such a system as textbook companion available through a Web tool, is still an option.

Conclusions

While change and evolution in learning technology systems are, not unexpectedly, common, the problem is often aggravated by a number of factors. In academic and research environments, research-oriented prototypes often have a longer-than-expected life span. The rapid evolution of particular Web and Internet platform and supporting multimedia technologies causes the need to constantly update and migrate. Learning technology research takes new platform technology aboard to facilitate new forms of learning. These factors together cause a rapid aging process in an LTS lifecycle process to take place.

We have used a change and evolution model to characterise and categorize the changes to an LTS that we have developed, maintained and used for over 20 years. Primary change factors such as changes in the subject domain or changes in the organizational context of the course in questions will always have a certain, but usually predictable impact. However, our observations have demonstrated that some factors such as platform and pedagogy have a higher impact the longer the system runs.

- Content and Format: These change factors cause LTS and content erosion – a gradual, evolutionary process based on incremental changes. These changes are part of the usual remit of an instructor.
- Infrastructure and Pedagogy: These factors cause more abrupt degradation – cause by distinct, often discontinuing events. These changes have implications beyond the usual instructor's remit.

Without additional support, resulting costs can made an LTS unsustainable for an instructor alone. Often costs are only justifiable if the LTS serves different roles, e.g. as a delivery platform, but also as a research vehicle.

These potentially high-impact factors all relate to an LTS and its links and connectivity to its wider context in terms research and technology and communities involved here. While some developments have a negative impact and are prohibitive in terms of sustainability, even complex systems have a change of survival (at least in parts) based of the same connectedness with the environment. With a culture of sharing and community-based development (as learning object repositories in different countries, but also open-source software development shows), efforts made by individual researchers or groups can benefit larger communities and can be utilized and developed further beyond the abilities of the original developers.

Acknowledgements

This material is based upon works supported by the Science Foundation Ireland under Grant No. 07/CE/I1142 as part of the Centre for Next Generation Localisation (www.cngl.ie) at Dublin City University (DCU). The author is also greatly indebted to Alan Smeaton, who started work on the IDLE system and kept extending it for 10 years, and Claire Kenny, who made an important contribution to IDLE through her M.Sc. research work. Thanks also go to all other students who contributed through projects and internships.

References

Boyce, S. and Pahl, C. (2007). The Development of Subject Domain Ontologies for Educational Technology Systems. Journal of Educational Technology and Society (ETS). 10(3):275-288. IEEE.

Devedžić, V. (2006). Semantic Web and Education. Berlin: Springer-Verlag.

Holohan, E., McMullen, D., Melia, M., and Pahl, C. (2005). Adaptive Courseware Generation based on Semantic Web Technologies. Proceedings International Workshop on Applications of Semantic Web Technologies for E-Learning (SW-EL2005) at 12th Intl Conf on Artificial Intelligence in Education AIED2005.

Kenny, C. and Pahl, C. (2005). Automated tutoring for a database skills training environment. ACM SIGCSE Symposium 2005. St. Louis, US. Pages 58-64. ACM Press.

Kenny, C. and Pahl, C. (2008). Personalised Correction, Feedback, and Guidance in an Automated Tutoring System for Skills Training. International Journal of Knowledge and Learning (IJKL). 4(1):75-92.

Melia, M. and Pahl, C. (2009). Constraint-based validation of adaptive e-learning courseware. IEEE Transactions on Learning Technology, 2(1):37–49.

Pahl, C. (2003). Evolution and Change in Web-based Teaching and Learning Environments. Computers & Education. 40(1): 99-114.

Pahl, C., Barrett, R. and Kenny, C. (2004). Supporting Active Database Learning and Training through Interactive Multimedia. International Conference on Innovation and Technology in Computer Science Education ITiCSE'04. ACM.

Pahl, C. and Holohan, E. (2009). Applications of Semantic Web Technology to Support Learning Content Development. Interdisciplinary Journal of E-Learning and Learning Objects (IJELLO), Vol. 5: 1-25.

Palmer, S. and Tulloch, W. (2001). The Evolution of Online Teaching and Learning in Engineering at Deakin University. Journal of Computing in Higher Education, 14 (1).

Reynard, R. (2008). Changing Designs of Online Learning: The Evolution of Digital Learning Systems Through Customization. THE Journal. Retrieved Jan 26, 2011 from http://www.thejournal.com/articles/22954

Robbins, S.R. (2002). The evolution of the learning content management system. Retrieved January, 26, 2011 from: http://www.learningcircuits.org/2002/apr2002/robbins.html

Smeaton, A.F. (1991). Using Hypertext for Computer-Based Learning, Computers and Education, 17(3), 173-179.

Smeaton, A.F., & Crimmins, F. (1997). Virtual Lectures for Undergraduate Teaching. Proc. ED-MEDIA'97 World Conference on Educational Multimedia and Hypermedia.