

Phoebe Development Map



Introduction

Phoebe is a prototype, proof of concept tool. Although broadly functional, it is therefore not as robust it would need to be for wide-scale uptake. However, with minor improvements to correct known faults and to extend functionality in certain key areas, Phoebe will be able to provide realistic opportunities for targeted implementation and wider piloting. In this way, it will be possible to conduct real-world investigations to extend our understanding of the further potential of planner tools.

Certainly, while both Phoebe and the LPP have been evaluated with real practitioners in a range of settings, the relative brevity of both projects has made it impossible to gauge their applicability to general practice. Real pedagogical planning can involve many people and take place over many months, and so evaluations conducted over longer time frames will almost certainly uncover usability issues and requirements that cannot be identified in a workshop lasting two hours or less.

With some additional development, therefore, we believe that it would be possible to conduct more meaningful pilots extending over longer periods. These would generate a more robust set of requirements on which to specify future systems, whether these build on Phoebe and/or the LPP or are designed afresh.

1. Current functionality

The current composition and functionality of Phoebe can be summarised as:

- 1.1 A self registration system
- 1.2 A library of shared learning designs
- 1.3 A set of guidance notes on learning design, and links to examples, covering:
 - 1.3.1 Context-specific help on areas of a learning design
 - 1.3.2 Introductory information about learning activities and sequences
 - 1.3.3 A mapping of activities to technologies
 - 1.3.4 Tools for e-learning
 - 1.3.5 Teaching approaches and techniques
- 1.4 A learning design creation and management interface providing the ability to:
 - 1.4.1 Create a new design from a specified template
 - 1.4.2 Create a new design from another design ("cloning")
 - 1.4.3 Edit one's own designs
- 1.5 A template creation and management interface providing the ability to:
 - 1.5.1 Create a template from another template
 - 1.5.2 Edit a template
 - 1.5.3 Create a design based on a template
- 1.6 A learning design authoring environment, including:
 - 1.6.1 The ability to fill in a learning design
 - 1.6.2 Access context-specific help and other guidance
- 1.7 A template authoring environment allowing
 - 1.7.1 The creation of new fields
 - 1.7.2 Changing of the links to context specific guidance

- 1.7.3 The reordering of fields
- 1.7.4 The authoring of default information
- 1.8 A learning design output environment providing the ability to:
 - 1.8.1 View a design in tabular format
 - 1.8.2 View a design in hierarchical format (as an indented outline)
 - 1.8.3 Export a design as XML code
 - 1.8.4 Print a design
- 1.9 Extensive user guidance about Phoebe

2. Improvements for immediate implementation

Although the current (beta) version of Phoebe can be (and is) used successfully, users' interactions are constrained by the need to work around a number of faults. Rectifying these, and making minor enhancements to functionality, would significantly improve users' experience in the short term (i.e. 3-6 months), pending the kinds of development envisaged in section 3 of this appendix.

These "immediate" improvements are:

- 2.1 General
 - 2.1.1 Ensure robustness in browsers other than Firefox
 - 2.1.2 Redesign saving across course areas to make consistent and logical for users
 - 2.1.3 Implement a search across shared designs
 - 2.1.4 Enable output of design in RTF format for easy editing
 - 2.1.5 Ensure robustness of the view design screen
 - 2.1.6 Ensure that terminology is consistent and meaningful to users: e.g. change "clone" to "copy" or "duplicate;" change "field" to something else.
- 2.2 Editing designs
 - 2.2.1 Allow formatted content to be pasted from MS Word
 - 2.2.2 Allow HTML codes to be included in the information entered in a design
 - 2.2.3 Make it possible to copy and rearrange certain fields in a design
 - 2.2.4 Keep hidden items hidden between sessions
- 2.3 Editing templates
 - 2.3.1 Allow the adjustment of a text entry box size in template
 - 2.3.2 Fix general usability bugs in template interface
- 2.4 Content
 - 2.4.1 Complete the current content
- 2.5 Features to explore in pilots:

We are aware of a number of new features which, if added to Phoebe, would significantly increase the value of any medium-term pilots and help identify requirements for the subsequent development phase. These features vary in complexity and in their implications for development. Some are dependent on collaborations with particular partners or user groups, but we believe all are fruitful areas for exploration. However, we suggest that the overhead incurred by implementing them (even in a tool which is likely to be redeveloped) will be rewarded by rich evaluation data to feed into the next tranche of requirements for development.

 - 2.5.1 Provision for managing a) different roles for individual users and b) different groups of users

- 2.5.2 Development of the community/sharing interface, which could operate both with guidance and with the learning designs produced by users
- 2.5.3 Streamlining the consumption of JISC and other outputs, through RSS feeds or other means¹
- 2.5.4 Creating customisable output
- 2.5.5 Developing customised content
- 2.5.6 Implementing a search for the guidance and the "Shared Designs" page
- 2.5.7 Exploration of the use of keywords or tagging for learning designs using, e.g., the LearnDirect taxonomy, free user tagging or folksonomies
- 2.5.8 Implementing interoperability with other systems as appropriate
- 2.5.9 Exploration of new content areas
- 2.5.10 Alternative means for mapping activities to tools

3. Wider development possibilities for future planner tool

The developments listed below have all been suggested by practitioners who have evaluated Phoebe or experimented with it. They are listed in no particular order, and we would expect that their desirability and priority will change in the light of experience with pilot installations and other evaluation activities.

- 3.1 Interoperability with other learning design tools through mappings between the Phoebe design structure and the structure used in the target tool.

It is likely that we will need to create an export facility which guides the user through the process, using their input to ensure that the data exported will result in a coherent design in the target learning design tool.

Note: The current design view/export mechanism has been designed to allow the addition of more export targets.

- 3.2 Interoperability with wider institutional and extra-institutional systems and tools that interface with the areas covered by Phoebe
- 3.3 Aggregating designs into a module or course, or in other ways determined by users
- 3.4 Ensuring full compliance with accessibility standards
- 3.5 Exploring other forms of representation for learning designs: e.g. concept maps (possibly through integration with Compendium LD)
- 3.6 Establishing a Phoebe guidance service as a long-term community project, separate from the actual tool
- 3.7 Integration with virtual learning environments such as Moodle or Sakai
- 3.8 Putting in place a stronger development discipline and process before the re-build, including the ability to run software project build systems, unit testing and continuous integration
- 3.9 Redesigning the software architecture around a code core and programming interfaces which can then be implemented by others, allowing the Phoebe team to focus on the core tool, and others to extend it
- 3.10 Redesigning the software architecture in accordance with standard or in-house XSD schemas as a design foundation, enabling database schemas, object models, and

¹ Our initial implementation of a feed consumer in the Phoebe guidance used a plug-in for the Trac platform which, although its basic operation worked, did not support many feeds. In particular, feeds using non-ASCII character sets were not displayed correctly. Moreover, security features in Trac prevented the use of javascript-based feed systems which would offer better feed support. To enable fully-functional feeds will require either further development on Trac, or the use of a platform on which this feature is more readily available, such as MediaWiki (see 3.11).

Web Services interfaces etc. to be derived from a single foundation. This would result in more options for flexibility, interoperability and integration

- 3.11 Migrate Phoebe guidance from Trac to a more suitable platform. Trac is more orientated to software development projects, so has many features that are not useful for the guidance and, conversely, lacks some that would be desirable.

Initial investigation suggests MediaWiki² as a good replacement. Notable MediaWiki features include:

- An Application Programming Interface (API) which should allow easy access to Phoebe guidance for other tools (learning design or otherwise), and also provide more scope for context-sensitive help in designs.
- Discussion pages, to enable community in-context discussion of guidance.
- A large developer community, which has produced several feed embedding extensions.³

- 3.12 Open the Phoebe guidance as a community project, allowing other institutions to insert their own help pages; and/or the development of standard help page modules that can be shared among several institutions.

- 3.13 Exploring the option to associate groups of sessions: e.g. to enable export of the type "Export all my session planning documents for Maths 101 Course for Term 3 to MS Word."

- 3.14 Developing and testing installation packages for institutions to host Phoebe locally and customise it accordingly.

- 3.15 Developing outputs from Phoebe that conform to standards such as IMS LD and others yet to be identified.

² <http://www.mediawiki.org/>

³ e.g. <http://www.mediawiki.org/wiki/Extension:FeedImport>