

Software Environments for Scenario-based Learning

A “snapshot” comparison of some available
tools

Terry Stewart and Scott Symonds
Centre for Academic Development and e-Learning
Massey University
8/20/2009



An Ako Aotearoa publication.

Support for this work was provided by Ako Aotearoa through its Regional Hub Project Funding scheme

Five software packages designed to support scenario-based learning were examined. This report compares their attributes against desirable features which support this paradigm.

INTRODUCTION

In its broadest sense scenario-based learning refers to learning that occurs in a context, situation or social framework. It's based on the concept of situated cognition which is the idea that knowledge can't be known and fully understood independent of its context (Kindley, 2002). In this sense it is seen as an advanced learning paradigm and one that assists with both student motivation and the uptake of skills.

Another definition, more specific and focusing on the paradigm within an e-learning environment, is offered by The University of London (2009):...

“Scenario-based learning puts the student in a situation or context and exposes them to issues, challenges and dilemmas and asks them to apply knowledge and practice skills relevant to the situation. The student navigates through by choosing options and is given feedback based upon their choice.”

Typically, a lesson designed around an interactive problem-based scenario would involve some kind of narrative, story or “setting”, some kind of tutor guidance and feedback where appropriate, assessment components (quizzes etc,) and the ability for students to make decisions which may have consequences in their scenario.

Although scenario-based e-learning lessons can be authored and delivered using simple webpages, there are some software programs that have been designed specifically for this paradigm. The authors have a role in encouraging and supporting staff wishing to develop scenario-based lessons at Massey University. One of the motivations for this study was to enable the authors to choose the best “fit-to-purpose” package for scenario-based lessons staff may be interested in producing, and also for their own research purposes into this paradigm. It was also felt a comparison of this nature would be of interest to other tertiary teachers developing lessons of this nature.

A QUALIFIER

It must be said from the outset that the senior author of this report has considerable input in the concepts behind one of the packages in this comparison, namely SBL interactive. This package is used to support the author's own scenario-based learning. It is familiar software, and its features determined many of the criteria listed below used in the comparison.

Given the above there was always an inherent risk of bias in the study. However, it must be emphasized that the senior author has no commercial stake in SBL interactive, which is being sold by the code owners, the University of Queensland. The comparison was made as objectively as possible looking at each of the specified criterion. We think the aim of a fair comparison has been realized, but it must be acknowledged that the criteria itself was determined by using SBL interactive as reference. In essence then, the study examined how similar packages “stacked up” alongside SBL interactive and it should be read in that light.

One other point needs to be made. Software changes quickly. Some packages disappear and others add new features or fix deficiencies. This study is a snapshot of the versions existing in early 2009. This report can act as a starting point for selection but it is important that potential users check out subsequent developments by visiting the product URLs listed below.

PROGRAMS EXAMINED

The packages for comparison were found from a combination of published literature and web searches. Five pieces of software specifically designed for scenario-based learning were selected for a closer examination. These were ...

- SBL-interactive, published by the Centre for Biological Information and Technology, University of Queensland (www.sblinteractive.com)
- Adobe Captivate, published by Adobe Inc (www.adobe.com/products/captivate)
- Udutu, published by cedarlearning (www.udutu.com)
- Stochasmos from the University of Cyprus (www.stochasmos.org)
- Emergo, from the Open University of the Netherlands (<http://emergo.ou.nl/emergo/community/EN/overemergo.htm>)

WHAT WE DID

A list of features was developed, reflecting what was desirable or useful in software designed to author and deliver teaching scenarios. Assessment was then carried out by downloading the packages, examining the support literature and websites, and authoring a small scenario in each of the packages to evaluate the features.

The findings were reported alongside each feature, and a short summary was written about each package.

WHAT WE DID NOT DO

Some of the packages listed SCORM-supported LMS integration in regards many of the reporting features. Integration into an LMS is not straightforward and requires expertise in server technologies. This was beyond the scope of the project and the ease at which this could be done was not assessed.

Also we did not “rate” the packages or provide some kind of scorecard. The programs were very diverse. Scenario-based learning can be used in many different ways and in many contexts and features that help support a particular scenario for particular learning objectives may be of no value in another and vice versa. Often there is a trade-off between complexity and power. What’s best depends on the particular learning objectives being addressed.

FEATURES

The following properties of each package were investigated:

- Availability
- Authoring and delivery platforms (i.e. Client-based, Server-based or both?)
- Ease of installation
- Student Interface
- Functionality of constructed scenarios to support scenario-based learning
 - Non-linearity (ability to explore a scenario in a non-linear fashion)

- Causality (where doing one activity can affect the availability or change the nature of another activity)
- In-built assessment components (tests, quizzes etc.)
- Attributing values to activities (i.e. ability to link time, costs, and scores to activities and tests)
- Logging of student activity and input
- Collecting objects (where items can be picked up and put down at various places in the scenario)
- Tailored Feedback depending on choice
- Ability to embed resources (PDF's) within the scenario (as opposed to simply linking to them)
- Ability to pass student data (scores, results, submissions, logs etc.) to an LMS
- The ability to restrict the students to a single pass through a scenario
- Team Play or groupwork functions
- The ability to export scenario content into other formats
- Ease of authoring
- Scenario portability
- Price and configurations
- Maturity
- Vendor and User Community Support
- Assistance in scenario planning
- Interoperability (Compliance with standards such as SCORM etc.)

FINDINGS

Four of the five packages were examined in detail. It was decided that the fifth package, Emergo, was not yet mature enough to be scrutinized in any depth. However, some information on Emergo is included at the end of this section.

Although all packages supported scenario-based learning we found the way they did this differed quite markedly from one to the other. For example, the SBL interactive interface represented a 2-D real-world environment where locations, objects and tasks in those locations could be examined, transported and manipulated. Working through a scenario was more akin to the 2-D “adventure” approach of classic computer games. Adobe Captivate and Uduu on the other hand, took more of an individual “page” approach with appropriate navigation options. Finally, Stochasmos was different again, closer to an LMS where cases were presented surrounded by a number of activity options.

Given this diversity, straight comparisons under the simple criteria listed above are not enough to give the whole picture as regards a particular package. Readers are encouraged to make their own judgment as to which package may suit their particular requirements.

1. ADOBE CAPTIVATE

Adobe Captivate is a relatively mature product from Adobe Systems Incorporated. The essence of Adobe Captivate is the ability to author (or convert from other formats) software demonstrations, simulations and scenarios, include assessment components such as random quizzes, and output

these in a widely used format such as swf (i.e. a native FLASH file). It can also convert swf to avi format if required.

Screen shots from other applications are easily captured and it is a straightforward task to add images and graphics. These can be made interactive by using hotspots and may be included in quiz questions.

Here's how it fared based on the features outlined above:

(i) Availability

The package is readily available from retailers or direct from Adobe Inc.

(ii) Authoring and delivery platforms

Authoring is entirely local on a client PC. As the output is a widely adopted file format (i.e. swf) supported by nearly all microcomputers, there is no need for specific server based software to deliver a Captivate-generated teaching module. Output can be either uploaded to a Learning Management System (LMS) or standard webserver, or it can be published as a stand alone item in a number of formats. This allows for both client-based and server-based delivery. Any student management required would be handled through LMS integration.

(iii) Ease of installation

Installation on a client computer is very straightforward using a standard installer. However, assistance with more advanced reporting features and plug-in installation for LMS integration would most likely require specialised help.

(iv) Student Interface

The basic environment as far as the student would see it during delivery of a Captivate lesson, is simply a Flash-based movie player with a timeline, play/pause, forward, backward and volume buttons. Scenarios allow audio, video, animation and drag/drop functionality.

The 'movies' (i.e. scenarios) that are created are entirely customisable. The down-side to this is that it places emphasis on the author to ensure that slides and scenarios are well laid-out and clear. For example it is possible to create highly intuitive objects that require little or no explanation in order for the student to use them successfully. However, it is also possible to create cluttered and counter-intuitive objects. Slide layout and design needs to be considered carefully so a uniform and consistent approach is taken across any learning object.

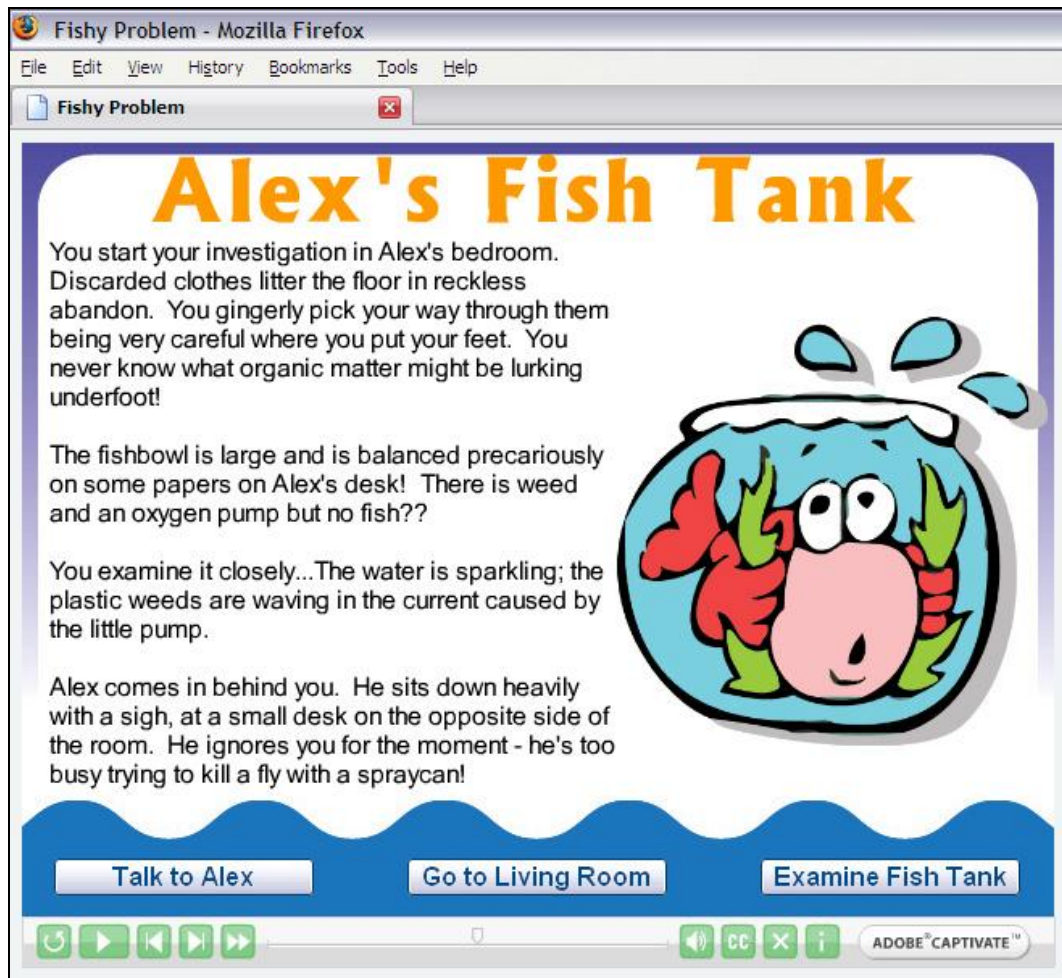


Figure 1a. A Student View of an Adobe Captivate Scenario

Captivate comes with a series of templates and styles that are clean and easy to customise for a particular look and purpose.

- (v) Functionality of constructed scenarios to support scenario-based learning
- Non-linearity - There is plenty of flexibility for the author to create non-linear scenarios.
 - Causality – This is facilitated by accumulated score, which can be used to open up new branches to screens as the user navigates through the scenario. This facilitates the use of mastery tests before progress is allowed.
 - Built-in assessment components. - Assessment and reporting facilities exist including quiz question databases. Questions can be in a variety of formats: multiple choice, short answer, fill in the blank, true/false, matching, drag/drop visuals. All allow scoring and branching based on the answer or score. Mastery quizzes from randomised banks of questions are possible.
 - Attributing values to activities - Points can be allocated for slides and questions. These can be tracked, accumulated and reported. Questions and slides can have a real-time limit, and this can be linked to points scoring. There is no provision for adding a specific monetary 'cost' value to activities.
 - Logging of student activity and input - Powerful tracking and reporting features are available through log files. Captivate has the ability to record user access, slide views, quiz results and other user interactions such as time spent on each slide, clicked objects etc. These logs can be emailed to nominated addresses from the Captivate file.
 - Collecting objects - Captivate does not try to mimic real-world environments, with separate locations and examinable and moveable objects. It is possible for authors to

use the software in this manner, but many features (such as the ability to collect objects) are not supported. Scenarios are best driven by analysis and assessment of what's shown on a series of linked and/or branching slides which are used to present the scenario. Consequently the interface is very much determined by the author. It's not really practical to provide 'maps' or 'inventories' using Captivate alone. One would need to be proficient with Flash and Actionscript.

- g. Tailored feedback depending on choice - Feedback can be instant and tailored to each answer that the user gives. Feedback can also summarise student performance at the end of a series of tasks or the scenario.
- h. Ability to embed resources (PDF's) within the scenario - External files can be embedded into an Adobe Captivate scenario.
- i. Ability to pass student data (scores, results, submissions, logs etc.) to an LMS - LMS integration is carried out through AICC and SCORM compatibility. All quizzes, interactions logs and slide views can be recorded and reported back to LMS.
- j. Restrict to single pass – Not available.
- k. Team Play functionality – Adobe Captivate itself does not provide this function, but groupwork can be organized through integration with an LMS or Adobe Connect Enterprise Server.
- l. Export of scenario content into other formats - Content can be exported to a variety of formats: Flash SWF, Adobe Connect, XML, Word or standalone .exe

(vi) Ease of Authoring

Authoring is relatively simple. 'Slides' are created in a similar fashion to PowerPoint. Indeed PowerPoint slides can even be imported as starting points. The 'timeline' is similar in use to that in Adobe Flash and those used in simple video editing software such as iMovie or Windows Movie Maker. All buttons and interactivity can be added through a simple WYSIWYG editor and a series of menus. An overview section shows the relationship between slides.

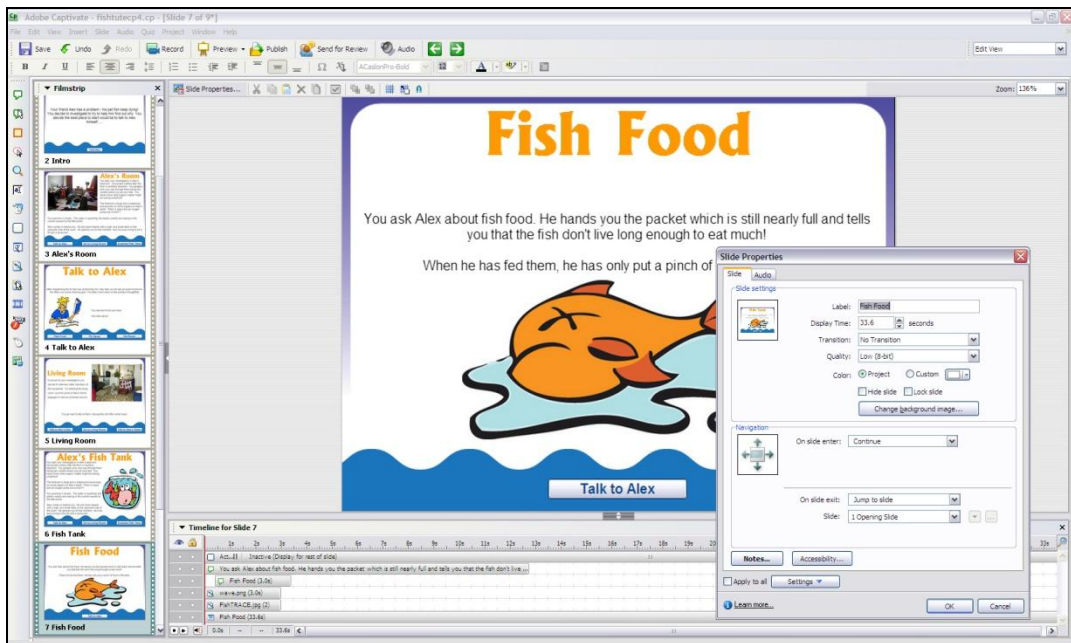


Figure 1b. The Adobe Captivate authoring interface showing a scenario in development

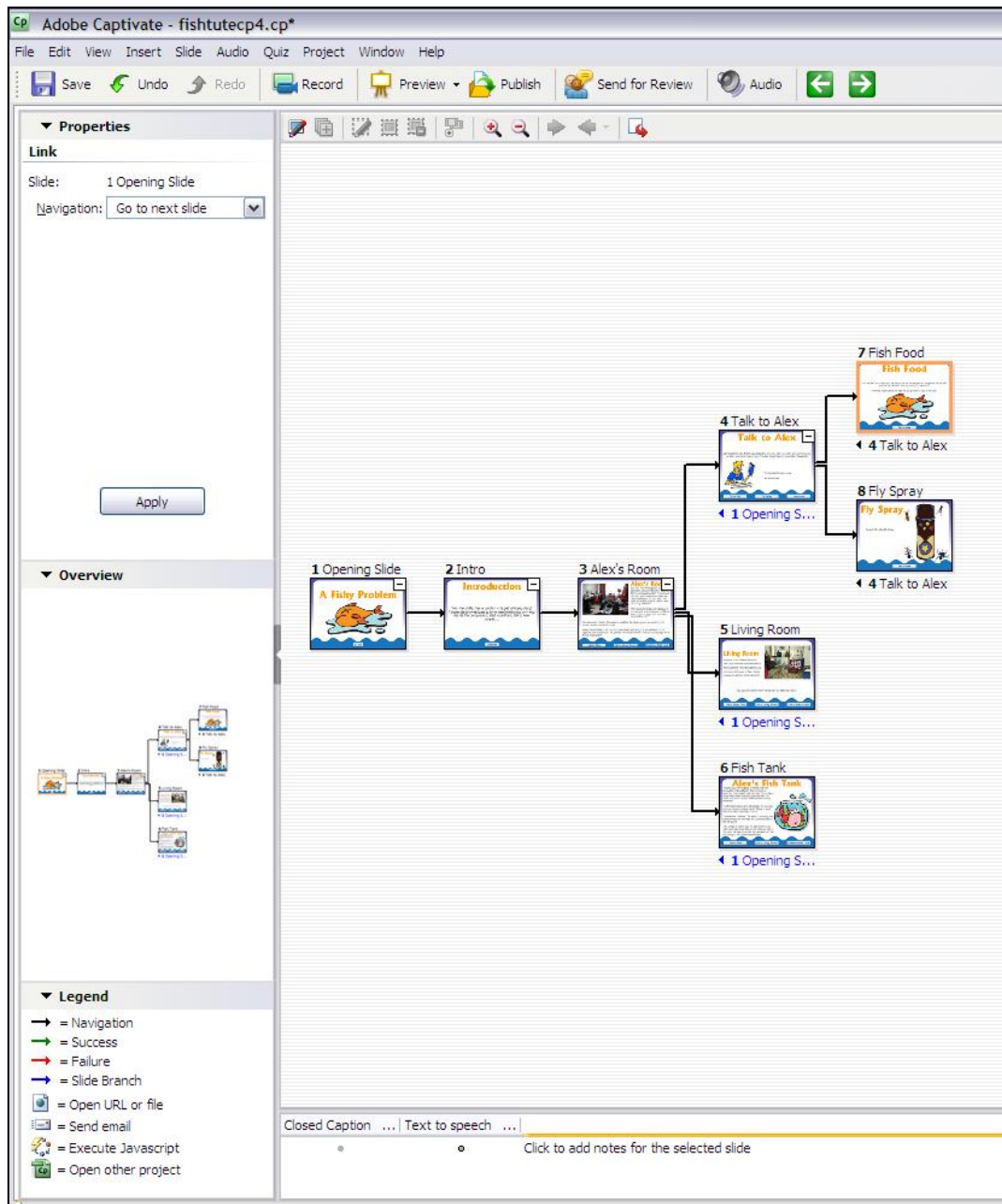


Figure 1c. The overview screen in Adobe Captivate, showing the relationship between slides

A variety of different authoring views are available including a branching view specifically for scenario building. There is no scripting required, although Captivate does allow scenarios to be exported to Flash for more advanced editing and scripting if needed.

(vii) Scenario portability

Content is very much a standalone package and can be delivered from a server or as an offline EXE file. Logs can be emailed to the tutor on completion of standalone scenarios. Software requires Adobe Flash to play. However, Flash is almost 'web-standard' these days with most PC users having it already installed.

(viii) Price and configurations

At the time of writing the latest version retails for around \$750. Subsidised rates for educational use are also available. The copy used for this review cost \$250. Free, fully functional 30 day demo are available for download from Adobe website. Upgrades to older versions are also available as downloads from Adobe.

(ix) Maturity

Captivate is still in its 'youth' when compared to many other Adobe products, but it is a refined and user friendly application as you'd expect from such a large software developer. During the trial Captivate 4 was released, which has features for collaboration, and interacts with other Adobe products like Bridge, Soundbooth, Photoshop and Adobe Device Central.

Captivate 3 (used in this evaluation) felt very well thought-out and useable. The added features of Captivate 4 are welcome and show Adobe's commitment to pushing the development of this piece of software.

(x) Vendor and User Community Support

Vendor support is obviously a strong point with this application. Captivate is a relatively recent Adobe acquisition (originally being a Macromedia product) and they seem intent on turning it into the standard for online learning applications. This is evidenced by the swift (almost annual) release of updated versions with new and improved features. Active Adobe Captivate forums exist along with online tutorials. There exists an increasing community of educational users who are expanding the knowledge base with online tips and techniques.

(xi) Assistance in scenario planning:

Adobe Developer Centre tutorials exist including video walkthroughs. The product has excellent built-in help file linking to online support and tutorials. There is also an abundance of third party web-based tutorials, most of them free of charge. Several quality third party books/guides exist on content creation.

Adobe tutorials tend to approach content from a commercial training point-of-view and there is less emphasis on pedagogical approach to the production of learning objects and how they might be supported in lessons. However, there is a growing base of educational users of Captivate and many are posting tutorials, hints and tips to blogs and various forums. Many of these postings consider pedagogical approaches and applications. Google is an invaluable resource for finding these!

(xii) Interoperability

Scenarios can be output as SCORM and AICC compliant packages. Reporting parameters can be set from within Captivate. User privileges would be assigned through the LMS.

Summary

Captivate is a useful tool for the creation of interactive learning experiences. Its use in exploratory or game-based scenario development is possible, but this is not the software's strength. It would be

more suited to a simple 'case-based' approach, where students are set a series of questions to a problem, with their answers (or diagnoses) affecting the outcome of the scenario. It is possible to provide formative feedback on the way, and for scenarios to be used for more formal, summative assessment with the powerful reporting features that the software contains.

The Flash objects that Captivate produces have several benefits: they're portable, easy to use and upload and advanced authors can import scenarios into Adobe Flash for further editing and refinement. Flash capabilities are well suited to embedding multimedia, and the quiz feature allows multimedia to be built into questions. No server is required and simple scenarios can be run straight off local media like a hard drive or CD-ROM.

The product appears to have good LMS integration and compatibility with essential formats such as AICC and SCORM.

Regarding its limitations, Captivate is not designed specifically for free-flowing exploratory scenarios with a high amount of causality, and many features that are available in other products do not appear. It's a proprietary product that was initially aimed at commercial training. It's relatively expensive.

There is no inbuilt facility for group work. This requires LMS integration or use with Acrobat Connect Enterprise Server.

2. UDUTU

Udutu is a simple online course and scenario creation service that offers free authoring tools and paid-for hosting of content. Using the Udutu web site, authors can create their scenarios, then download it as a package of HTML and/or FLASH documents that can be viewed in any browser, offline or online.

(i) Availability

The authoring tool is available for use at www.udutu.com. Connection speed might be the only hurdle, as the authoring process does require reasonable bandwidth. Screens are refreshed frequently. There have been no issues on the networks it has been tested on.

(ii) Authoring and delivery platforms

Authoring is entirely online from your account on the Udutu website. The authoring tool is entirely free. Teacher created content can either be hosted on the Udutu website for a fee or exported for use in a learning management system or for standalone delivery. Any student management required would be handled through LMS integration.

The authoring server is also available for institutional delivery at a negotiated price. A client-based authoring tool available on purchase is also planned.

(iii) Ease of installation

No local installation is necessary. The authoring tool (myUdutu) is accessible simply by creating a free account on the Udutu website.

(iv) Student Interface

The interface is fully multimedia, simple, clear and intuitive. Like Adobe Captivate, presentation is very much screen by screen. Simple scenarios are easy to navigate through. Icons and graphics can be added, but authors are very much limited as to how and where these can be used within slides. It is largely a screen by screen approach similar to a series of linked pages. The collection of 'objects' or an inventory is not supported.

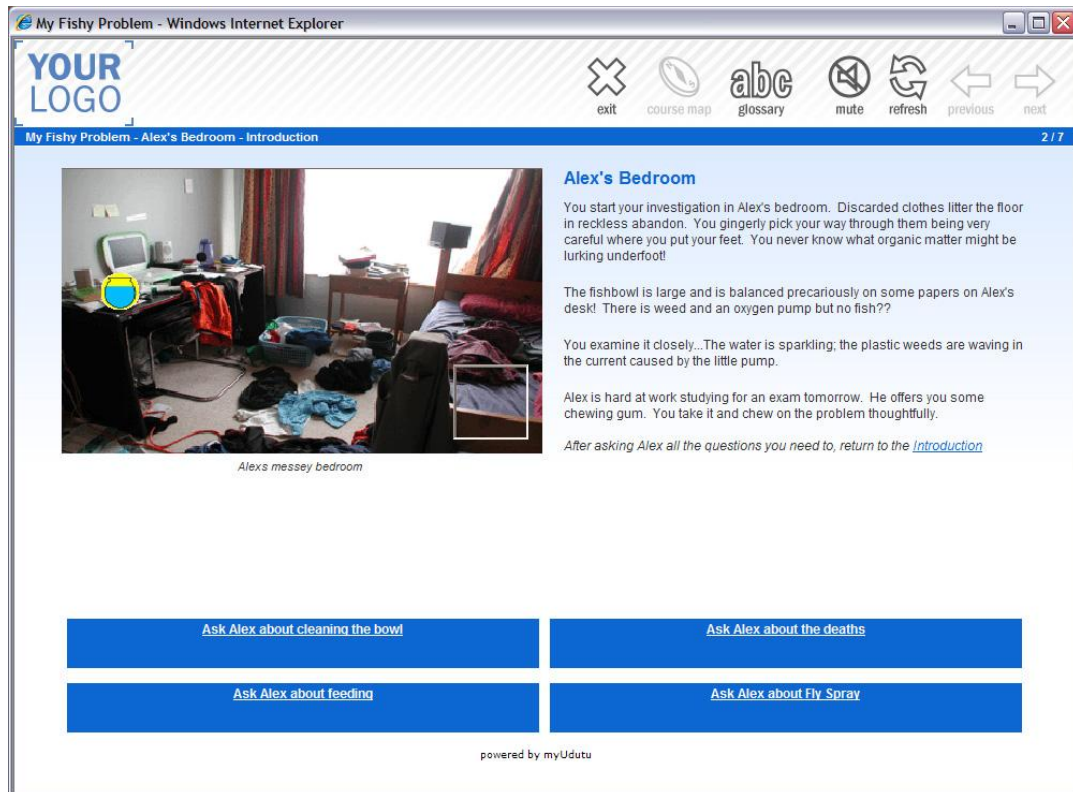


Figure 2a. A Student's view of a typical scenario in Udutu

(v) Functionality of constructed scenarios to support scenario-based learning

- a. Non-linearity – There is flexibility for authors to create non-linear scenarios. Branching is supported.
- b. Causality – The use of pre-requisites is limited. Tripping variables, such as time or score to open other options or routes through the scenario is limited to using assessment slides. After an assessment slide, scenarios can branch dependent on the responses students make. This can be used to 'stream' students and create mastery exercises.
- c. Built-in assessment components - A variety of assessment slides can be used: multiple choice, labeling images, order stepping, phrase matching etc. These can be scored or self assessed.
- d. Attributing values to activities – Scores can be added to assessment components. The only other activities students might carry out are clicking a navigation link or multimedia component. No values can be attributed to these
- e. Logging of student activity and input – Log files do not appear to be available with the hosting service. As a Udutu scenario is simply a collection of HTML and FLASH files it is conceivable that once integrated into an LMS, logging could be arranged for the pages visited. This doesn't appear to be automatic though.
- f. Collecting objects – An Object based inventory is not supported. As with Captivate, the software does not attempt to mimic real world environments.

- g. Tailored feedback depending on choice - Feedback can be given specific to each answer a student gives. It's also possible to incorporate audio feedback for correct and incorrect answers.
- h. Ability to embed resources (PDF's) within the scenario - External files can be embedded in scenarios and opened by using a 'link' button. All external files used within scenarios are zipped in a scenario package when exported.
- i. Ability to pass student data (scores, results, submissions, logs etc.) to an LMS – The standard package does not seem to offer this.
- j. Restrict to single pass – Not present in the standard version
- 1. Team Play functionality – There is no group work or group interaction facility in the existing versions. However, this could be managed with an LMS.
- k. Export of scenario content into other formats – Not supported.

(vi) Ease of Authoring

Very simple, template-based approach, which is very straightforward to use. It presents a WYSIWYG interface with HTML text editing as an option for more advanced authoring. As with Adobe Captivate, a scenario framework is displayed showing the relationship between slides.

The screenshot shows the UduTU authoring interface. At the top, there are navigation tabs for 'Workspace', 'Library', and 'Administration'. Below these are numbered steps: 1 create, 2 author, 3 distribute. A toolbar contains icons for 'add', 'basic screen', 'assessment screen', 'advanced screen', 'group', 'add scenario', 'import powerpoint', and 'edit glossary'. The main area is titled 'edit basic screen' and shows the 'My Fishy Problem > (2) Alex's Bedroom > 2. Introduction' screen. On the left, a 'course outline' tree shows the structure of the scenario. The main editing area has sections: 'B choose a basic screen layout' with three layout icons; 'C personalize your basic screen' with a 'screen name' field (containing 'Introduction'), a 'description' field, and checkboxes for 'completed', 'share this screen in your library', 'self assessment', and 'scored'; and 'D add content' with a 'save' button. Below this is a preview of 'Alex's Bedroom' featuring a photo of a messy bedroom and text describing the scene and a fishbowl. At the bottom, there are four blue buttons: 'Ask Alex about cleaning the bowl', 'Ask Alex about the deaths', 'Ask Alex about feeding', and 'Ask Alex about Fly Spray'. The final section is 'E add narration if required' with a 'title' field, a 'stream narration' checkbox, and 'add', 'library', and 'save' buttons.

Figure 2b. Authoring a Scenario in UduTU

(vii) Scenario portability

Scenario content can be exported as zipped SCORM 2004 or SCORM 1.2 compliant packages which are HTML and Flash-based. The zipped SCORM exports can be used to run scenarios and learning objects offline with similar functionality to those hosted in an LMS or by UduTU. However, in our assessment we found there appeared to be some problems with functionality of Flash swf files in offline scenarios, such as lockups and crashes in certain situations.

(viii) Price and configurations

Use of the authoring tool is free. Udu charge \$US 1 per screen to host content, although you are not obliged to use their servers and are free to export content with no charge. Where institutions want to purchase a Udu server of their own, prices seem to be negotiable on a case by case basis.

(ix) Maturity

Udu is still a developing product. Updates seem reasonably frequent and support via Udu forums seems adequate. There are elements that still feel under-developed or lacking, for example the user interface could be tidier. Udu has very recently introduced Section 508 compliance features that allow authors to meet US standards for disability provision in their content. This demonstrates a commitment to ongoing development.

(x) Vendor and User Community Support

The application seems relatively well supported. Questions posted in the forums appear to be answered reasonably quickly by developers. The user community is small at the moment, so there is a lack of third-party support. Udu make their money from paid-for hosting, technical support and course development. Solid technical support for 'free' users may not be high on the Udu list of things to provide, which is understandable.

(xi) Assistance in scenario planning:

There are fewer independent third party support materials and written guides/books than other more-established commercial software but what is available is good quality. Multimedia tutorials exist on how to build courses in Udu (including a PDF 'getting started' guide) along with several tutorials on pedagogy and theory of instructional design which is commendable.

The web site contains User forums with sections devoted to instructional design and technical issues.

For a fee, Udu course developers are available to assist with the development of courses.

(xii) Interoperability

Scenarios can be exported as SCORM compliant packages.

Summary

The Udu web-based service is free and will produce learning objects that can be used within Moodle or other LMS programs with some work. It's also very simple to use and cleanly laid out. It lacks many features that would add depth or functionality to scenarios, and Udu reporting/logging functions are poorly developed and implemented at this moment in time. This obviously limits the possibilities for using the application for summative testing. It is worth noting though that a customisation service exists and, for a price, these server-based features could probably be set up.

It is unclear as to how well reporting/logging would work through Moodle as this was not tested, but it is assumed that Moodle would have to be responsible for this for the most part, and this would put a great emphasis on the Moodle/Udu plug-in and how well it works.

In many ways Udutu is similar to Adobe Captivate, although the Adobe package is far more versatile. Again, a page by page approach to scenarios would be beneficial here, with less emphasis on a story or location based 'adventure'. Used within its limitations, Udutu could be a valuable content creation tool to have in a digital learning arsenal.

3. SBL INTERACTIVE

SBL Interactive (SLBi) is owned by the Centre for Biological Information Technology (CBIT), University of Queensland. The idea for SBL interactive arose from past collaborative work between CBIT and Massey University, and is essentially an improved commercial version of PBL which was designed on contract as part of a TEC-sponsored e-learning Collaborative Development Fund (eCDF) project in 2005-2006. This was made available to all government - sponsored tertiary teaching institutions free of charge (Stewart, 2007).

SBL Interactive has a distinct WYSIWYG interface for authoring scenarios. Rather than being single screens or slides, content is presented in a 2-D representation of reality with an area for the locations, scenes and objects pertinent to the scenario to appear, be examined, collected and transported.

(i) Availability

The authoring tool is available for download at www.sblinteractive.org. Trial versions are available.

(ii) Authoring and delivery platforms

The software consists of a client-based authoring tool (a Builder) and a server-based player which allows scenarios to be run in a web browser. The builder also has a player mode for testing and can export scenarios with a client-based player for offline viewing.

SBL interactive is only available for Microsoft Windows operating Systems.

(iii) Ease of installation

Installation for the builder is straightforward. NetFramework 2.0 is required but this is installed with the main program if it is required. The player server software is also easy to install but would normally require specific tweaking to fit into an institutional setting.

(iv) Student Interface

The student interface consists of four panes. The top left hand side shows a series of clickable icons representing locations. These can be physical locations in the scenario or conceptual ones such as "Reflections" or "Introduction". The right hand top window shows a graphic associated with each location (normally a scene) and can also hold clickable (and collectable) items. The lower left hand window shows a series of hyperlinks representing actions or (via a tab) a student's current collection of items. Content from clicking on locations, items and actions (collectively known as objects) is shown in the lower right hand screen.

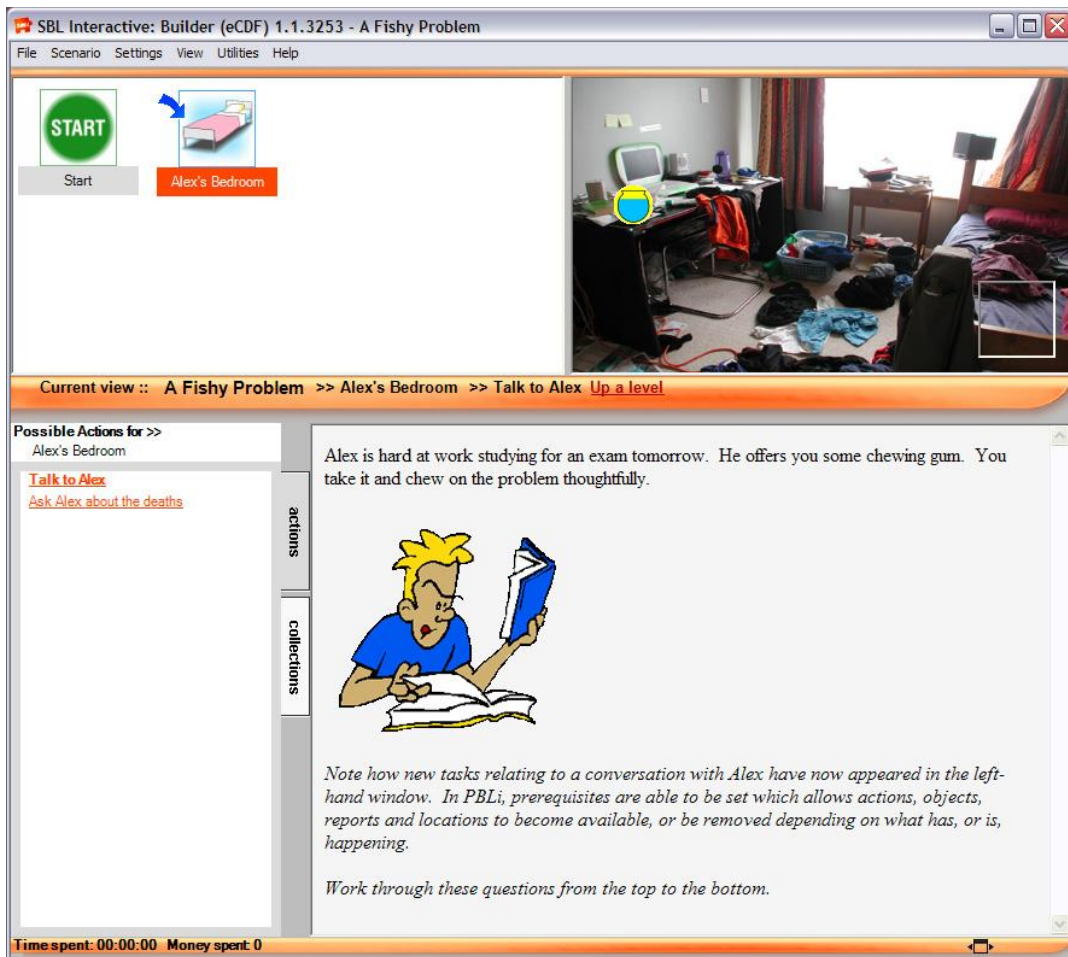


Figure 3a. A SBLi Scenario in Player Mode

Locations in scenarios can be “nested” in that clicking on a location can act as a doorway to further locations and activities. Clicking on an exit door can return to the original locations. In this way very large scenarios with multiple environments can be accommodated.

- (v) Functionality of constructed scenarios to support scenario-based learning
- Non-linearity – Authors can create non-linear scenarios allowing free exploration. Limited one-way branching can be accommodated through the use of a pre-requisite functionality which imposes conditions on whether an object appears or not.
 - Causality – The prerequisite functionality can be applied to all objects and assessment components like reports. A prerequisite will hide or reveal an object depending on what was done before. You can also have a number of prerequisites on a single object. However in a Boolean sense multiple prerequisites are associated only by AND not by OR. All conditions must be satisfied before the object can be hidden or revealed and this is a limitation in constructing rich scenarios.
 - Built-in assessment components – Multiple choice, tick box, multiline and single text boxes are available. Multichoice and tick boxes can be scored. Scores from the same assessment component can be accumulated if the student repeats the test and these scores can be used to trigger pre-requisites. They are not accumulated throughout the scenario though, at least not in the present version.
 - Attributing values to activities – Clicking on any object can increment a cost or scenario time score which is cumulative.

- e. Logging of student activity and input - Log files are available both from the client builder and player (via email to a nominated SMTP server), and from the administration interface in the server player
 - f. Collecting objects – An object-based inventory is supported. Items in SBLi can be collected and transported to locations elsewhere.
 - g. Tailored feedback depending on choice - Feedback can be given specific to each answer a student gives in one of the assessment components. Also, feedback on the relevance or otherwise of all objects clicked on can be given at the end of the scenario as “customized feedback”
 - h. Ability to embed resources (PDF's) within the scenario - External files can be embedded in scenarios and will stay with the scenario data file if this is moved.
 - i. Ability to pass student data (scores, results, submissions, logs etc.) to an LMS – This functionality does not exist at the moment in SBLi but a Blackboard plug-in is being developed by CBIT. A Moodle plug-in is also planned.
 - j. Restrict to single pass – In the server player students can be restricted to a single pass through a scenario.
2. Team Play functionality – Team play functionality exists where students can be arranged in groups and can chat to each other in a chat box while exploring a scenario.
 - k. Export of scenario content into other formats – Scenarios can be exported as HTML and PDF flat files showing content and the associated properties of each object.

(vi) Ease of Authoring

SBLi has a WYSIWYG interface. Objects can easily be created and properties assigned to them through a specific point-and-click properties panel. No scripting is required, even for advanced functions. A library of icons and images is also available from within the program.

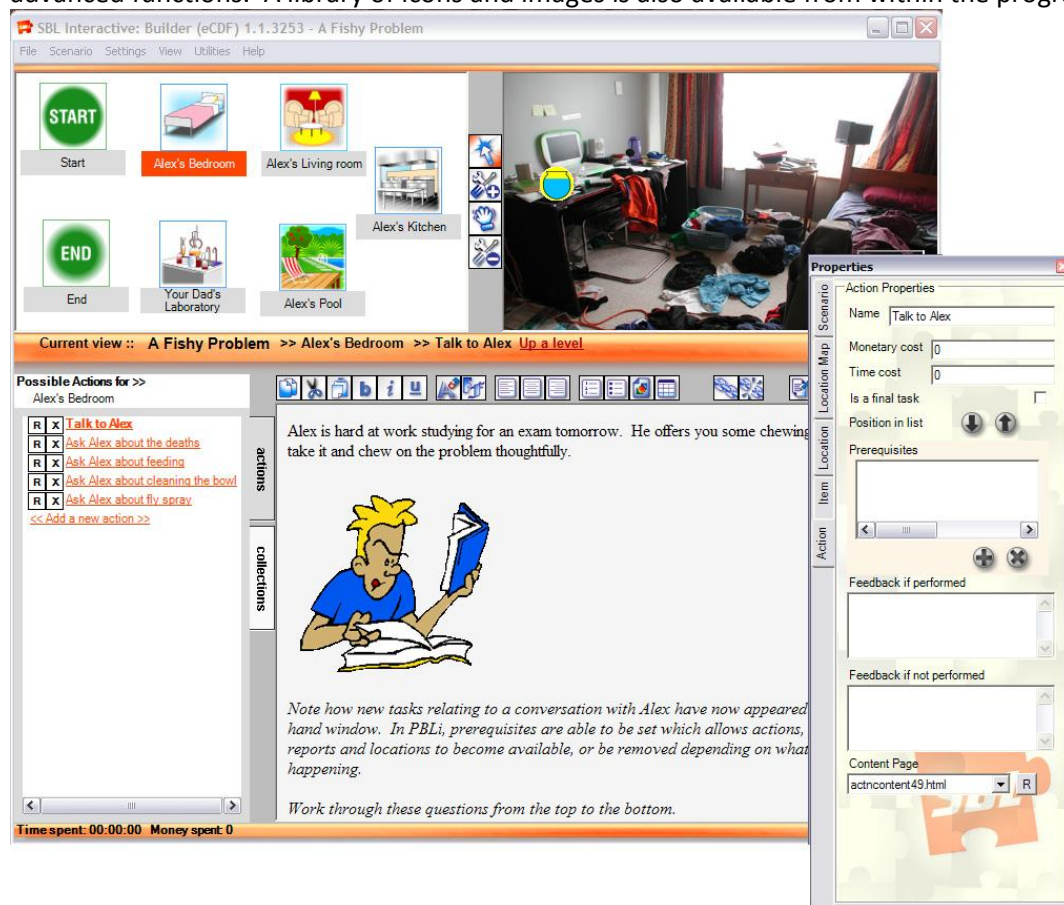


Figure 3b. The authoring interface of SBL Interactive

(vii) Scenario portability

Although scenarios are self contained files and can be transported, they are specific to SBLi. They consist of standard XML content but they are not SCORM compliant.

(viii) Price and configurations

Usually \$295 AUD for authoring/player application, and \$1500 AUD for server version. CBIT will host SBLi scenarios at additional charge of \$499 AUD per year.

A limited-functionality demo of the Builder is available for free download.

Tertiary institutions in New Zealand who received PBL Interactive through the eCDF project mentioned above are eligible for a free upgrade to the current version of SBLi.

(ix) Maturity

Like all of the other products assessed, SBLi Interactive is still maturing. Enhancements like adding a score value for clicking objects and providing Boolean AND enhancements to the prerequisite functionality are planned.

(x) Vendor and User Community Support

The application is well supported by both developers and its user community, which along with the University of Queensland, involves Massey University and the University of Manchester, U.K.. Updates and revised versions are released regularly, and there is pedagogical support from several institutions. Much of the effort to date from CBIT involves embedding the tool in the University of Queensland.

(xi) Assistance in scenario planning:

An example tutorial is provided with the authoring tool and there are some resources on the website. There is also a third-party manual (plus CD-ROM) which takes the user through planning a scenario featuring numerous examples.

(xii) Interoperability

SBLi scenarios are not SCORM compliant and can only be used in the SBLi environment. However, a BlackBoard plug-in is under development and a Moodle plug-in is being considered.

Summary

SBLi has been designed specifically for exploratory scenario-based learning using an “adventure game” metaphor and as such includes many features (inventory of objects, time and monetary costs, complex pre-requisite system) that are simply not supported or are massively under-developed in the other products.

While SBLi can manage simple branching through the use of prerequisites, complex branching is difficult. The addition of “OR” functionality within multiple prerequisites assigned to an object would aid this enormously.

The ability to publish scenarios as self-contained, standalone objects which can be run in a web-browser with no extra software, in a similar manner to that of Captivate or Uduu, would add greatly to the flexibility of the package.

Although the server player management interface facilitates student management and reporting, SBLi does not interact or report directly to learning management systems. However, CBIT is actively developing this capability. .

Despite these issues, SBLi offers users the ability to create complex and engaging scenarios that would be difficult to create with any other software, without complex programming.

4. STOCHASMOS

Stochasmos is an inquiry-based learning “environment” that encourages problem solving and the construction of evidence-based explanations through individual and collaborative activities. It is essentially a collection of webtools and templates integrated into a coherent environment.

(i) Availability

The website for this learning environment can be found at www.stochasmos.org – The tool, which is expected to be released as open source once complete, is not publicly available as yet. There are several institutions piloting the application and there may be opportunities for collaboration with the developers. We were given access to an authoring account by the developers for this evaluation.

(ii) Authoring and delivery platforms

Authoring is entirely online using a Stochasmos server, which is also used for delivery. The Stochasmos server requires a Windows XP or Vista platform.

(iii) Ease of Installation

No local installation is necessary. The authoring tool is accessible via the developers through an authoring account.

(iv) Student Interface

The interface presents a clean, well laid out and easy to navigate interface using tabs and menu bars. Some features include:

- An ‘Inquiry Environment’ with information, data and situations for the student to investigate.
- ‘Workspace’ for students to write responses, collect thoughts and compile answers to problems. ‘Sharing’ area for students to collaborate in pairs or groups and compile thoughts and answers to problems. These areas are similar to an online exercise book or journal and can have permissions assigned to them by both the teacher and the student.
- Notebook for thoughts and ideas.
- Image capture tool for students to take screenshots. Screenshots can be placed into reports by students to support their analyses.

- Forum for whole-group (or wider community) discussions.

Possible issues could arise with successfully 'linking' the various areas within any given assignment so that students utilise each area to its fullest potential. Spaces are very much teacher defined and as such much of the responsibility of clarity and navigation falls on the teacher.

The screenshot displays the Stochasmos web-based reflective inquiry environment. The interface includes a top navigation bar with tabs for Project, Templates, Inquiry Environment, Add Group, Review, Group Pairing, Partnership, and Forum. A user profile for Terry Stewart is visible in the top right. The main content area is titled 'A Fishy Problem' and features a navigation menu with 'Start', 'Alex's Room', 'Talk To Alex', 'Alex's Fish Tank', and 'Fly Spray'. The 'Introduction' section contains a cartoon illustration of a pink fish in a blue tank and a text block that reads: 'Your friend Alex has a problem... His pet fish keep dying! You decide to investigate to try to help him find out why. You decide the best place to start would be to talk to Alex himself....'. A 'Hints' icon is also present. The footer includes copyright information for Stochasmos and a 'POWERED BY nqcontent' logo.

Figure 4. The fishy problem in Stochasmos

The student/work interface is entirely customisable and can be saved as a template. Templates can be shared on the Stochasmos server between teachers and can be imported to a project and edited.

(v) Functionality of constructed scenarios to support scenario-based learning

- Non-linearity – Very non-linear. Students can visit pages in any order, revisit pages, make notes wherever they like and collaborate wherever they like. There is no formal provision for branching but there is a great deal of flexibility to how the environment can be set up for students. 'Nesting' can be facilitated by creating pages and linking to them in various ways.
- Causality – Pre-requisites aren't supported, but they don't fit into the method of delivery Stochasmos supports. Scenarios are very much inquiry-based where students are set a problem and presented with data that they need to analyse to provide a solution to the problem.
- Built-in assessment components – Usual web form components can be used.
- Attributing values to activities – This is not supported
- Logging of student activity and input – Student activity can be captured on the Stochasmos server
- Collecting objects – Not supported

- g. Tailored feedback depending on choice - The application was designed with blended learning and face to face teaching in mind. It is expected that the teacher would provide input and feedback throughout the task. Assistance can be provided on each page in the form of a Hint button. The teacher can review student work at any time. It is possible to provide asynchronous feedback at any stage and on any of the work carried out by students by using 'Post-It' notes. It is also possible to provide synchronous feedback using the Chat function. The nature of the application and teaching approach makes automated feedback difficult to employ and scoring is limited to a final teacher assessed grade and comment.
- h. Ability to embed resources (PDF's) within the scenario – Available.
- i. Ability to pass student data (scores, results, submissions, logs etc.) to an LMS – As with SBL Interactive, all student management is done through the Stochasmos teaching/authoring tool. LMS integration would require a plug-in, which does not appear available at the moment.
- j. Restrict to single pass – Not available. The environment is quite open.
- k. Team Play functionality – Students can be assigned to groups, and classes can be made up of groups of students from different teachers registered on Stochasmos, making inter-class and even international collaboration a possibility.
- l. Export of scenario content into other formats – Content is not portable outside of Stochasmos, although any content that is created can be shared with other Stochasmos users

(vi) Ease of authoring

Generally well laid out with HTML based WYSIWYG editors and provision for content management. However, relatively complex with links to certain tools appearing in different places. The problems associated with a multi-faceted environment are reflected in the authoring tools. It's often difficult to visualise how the different views and areas relate to each other. Occasionally the server was slow and non-responsive, but this could be a technical issue.

(vii) Scenario portability

Content is standalone and contained on the Stochasmos servers. There doesn't appear to be any way to export content in any form.

(viii) Prices and Configurations

Unknown at this stage but the development is cited as "open-source".

(ix) Maturity

At the time of writing Stochasmos is still a work in progress. It seems to have grown out of a number of research projects. Testing and development is ongoing. There are a number of known bugs and issues with the application that the developers are striving to iron out. There are several partner institutions using the software and assisting with beta testing.

The application itself feels solid and well constructed. The environment can run a little slowly at times and it is assumed that users would require reasonably fast internet or use local network connections.

(x) Vendor and user community support

The product appears to be well backed by several organisations. The developers were quick to respond to any emails that were sent and updates occur regularly. Whilst the application is a work in progress, it feels professionally constructed and supported which inspires confidence that it will be around in a few years time.

(xi) Assistance in scenario planning

Demonstrations, templates and examples appear mostly developed for middle and high school use. There are simple schemes of work to accompany the several demonstration inquiry environments that are currently available.

There are fewer independent third party support materials and written guides/books than for other more established commercial software. However, user forums exist with sections devoted to instructional design and technical issues.

There is an extensive PDF manual available that is focused on how to use the authoring tool with a section specifically aimed at students. It is largely of a technical nature.

There is good provision for sharing resources, templates and environments through Stochasmos servers.

Stochasmos was not designed purely for distance and standalone use, but as a way to facilitate investigative and collaborative learning in the classroom and to support other methods of delivery. As such, much of the support material is geared towards this kind of usage.

(xii) Interoperability

As with SBLi, lessons are largely self-contained within the Stochasmos environment, rather than being developed with a view to LMS integration. Hence there doesn't appear to be considerations of SCORM compliance in this regard.

Summary

Stochasmos takes a very different approach to the other applications under review. It has primarily been designed as a web environment for use in blended learning as a means to facilitate collaborative, inquiry-based learning. As such it contains few of the features (such as branching, quizzes, scoring etc) that are found in most other products reviewed and focuses on student investigation, collaboration, analysis and reporting, with the teacher being integral to this process in terms of guidance and feedback.

The application functions in a similar way to a learning management system, although it has far more specific functionality and focus when it comes to exploring scenarios.

EMERGO

Emergo appears as an interesting application in that, working from published papers, (Nadolski *et al*, 2008), the developers seem to be specifically developing a methodology and toolkit for the

development of “serious games”. In other words, engaging educational scenarios that use a gaming approach. There is serious attention being paid to scenario design in this regard, linking back to educational principles. Scenario (and hence) lesson design is a problematic area for scenario-based learning.

The tool is open-source but on investigation seems still at the development stage. It was not included therefore, but is one to watch for in the future.

CONCLUSION

The comparison proved an interesting exercise. All products had pros and cons, and choice would depend on what the teacher was attempting to do, and the environment they were working in.

SBL interactive allows rich exploratory scenarios with good feedback and testing functionality to be created relatively quickly with no scripting. The interface helps scaffold a 2-D virtual world which the student can explore and interact with objects within it. The downside is that even simple scenarios require client-based software (for stand-alone machines) or server-based software (for web-interaction) to view them. In an institutional setting, this means someone (often an IT department) is required to install, maintain and often tweak the configurations. Although web-based scenarios can be launched from an LMS, there is no direct inter-operability between that and SBLi meaning student authentication and management must co-exist in both systems without specific customisation.

Adobe Captivate also allows a series of simple navigatable content screens plus assessment components to be produced fairly quickly. The advantage over SBLi is that Captivate files can simply be run in a web-browser. Hence there is no need for a separate server or player, and captivate scenarios can be easily played offline or integrated as learning components into LMS systems. Adobe is a large company and on-going development and support is likely. However, developing anything more than linear click-through screens takes some skill and the absence of a scaffolding interface means that creating a coherent world for a student to explore would take a large amount of design effort.

For basic scenarios, the Udutu free on-line authoring tool is quick and easy to use. What’s more the output is standard SCORM-compatible flash or html based content which can be run in a browser with or without a webserver. Like Adobe Captivate, it is largely page-based, but in contrast to that product advanced functionality is limited. More functionality may be available with the server product, but like Adobe Captivate, use of this would require specialist skills.

Finally Stochasmos seems to have more in common with an LMS, although one specifically for inquiry-based learning rather than a catch-all. In this sense alone it is useful although, as with SBLi, running two learning server-based environments in tandem can be problematic for institutions unless the interface between them is seamless. Stochasmos is still in development. Both it and Emergo will be worth watching in the future for those interested in this particular paradigm.

REFERENCES

Kindley, R.W. (2002). Scenario-Based E-learning: A Step Beyond Traditional E-learning. ASDT online magazine - Learning Circuits 3 (5), [ONLINE] http://www.astd.org/LC/2002/0502_kindley.htm . Accessed 17th July, 2009

Nadolski, R.J., Hummel, H.G.K et al (2008). EMERGO: A methodology and toolkit for developing serious games in higher education. *Simulation & Gaming* 39: 338-352. DOI:10.1177/1046878108319278

University College of London. (2009). Scenario-based Learning. [ONLINE] <http://www.ucl.ac.uk/learningtechnology/content/sbl/>. Accessed 17th July, 2009

Stewart, T.M. (2007). Tools and techniques for scenario based e-learning for New Zealand tertiary students: Prototype to adoption. ICT: Providing choices for learners and learning. Proceedings ascilite Singapore 2007. Centre for Educational Development, Nanyang Technological University, Singapore, 2-5 December. [ONLINE] <http://www.ascilite.org.au/conferences/singapore07/procs/stewart-t.pdf>. Accessed 17th July, 2009



This work is published under the [Creative Commons 3.0 New Zealand Attribution Non-commercial Share Alike Licence \(BY-NC-SA\)](https://creativecommons.org/licenses/by-nc-sa/3.0/). Under this licence you are free to copy, distribute, display and perform the work as well as to remix, tweak, and build upon this work noncommercially, as long as you credit the author/s and license your new creations under the identical terms.