Selecting a Software Simulation Package

Introduction

Once that it has been decided to switch to software simulation to deliver training exercises and tests, it is necessary to create a plan for selecting the appropriate software simulation package.

The purpose of this document is to:

- Briefly describe what a software simulation package does and how it does it
- Outline a process for selecting the appropriate package
- List the critical features and functions that simulation packages should contain

Software Simulation: a description

Definition

A software simulation application is a computer program that reproduces the look and feel of another application and mimics the functionality of that application (usually in a limited way).Typically, software simulations capture the computer screen as a picture at each step of specific procedure. These screen shots are then annotated either with text or with audio narration (or both). Click areas and/or text entry boxes can be overlaid on the screen shots to allow the user to interact with the simulation in the same way they would with the actual computer program.

Presentation Modes

Once the screen shots have been captured and annotated, the simulation can be presented in the following modes:

- **Demonstration**. In this mode, the simulation plays like a movie. Interaction is limited to starting, stopping and replaying, much the same as the way you control a video in a VCR.
- **Interactive Exercise**. In this mode, the user practices performing the procedure by clicking on the appropriate controls and entering text into the appropriate areas. Incorrect actions can be responded to with on-screen text feedback and/or audio narration. Typically, performance is not scored.
- **Testing**. By removing the instructional annotations from the simulation, the interactive exercise can function as a test. Performance is scored and the scores can be transmitted to a variety of Learning Management Systems (LMS). Some software simulation packages have two levels of testing: self tests which report the scores to the user only and criterion tests which capture the scores for formal student evaluation.

Software Simulation: Critical Functions and Feature

Presentation Mode Options

The first step in selecting a package is to determine what presentation modes are required. All packages under consideration can create demonstrations and

interactive exercises. These functions may be used in classroom training, self-paced Web-based training and "just-in-time" training embedded in online help.

However, if you want to provide student testing, you face a strange anomaly. Most simulation packages are not capable of sophisticated text entry analysis. Instead, most packages only provide for "exactly correct" or "don't score" text entry analysis. This means that simple typing errors will be scored as wrong answers, an unacceptable limitation in my view.

Text analysis tools have been part of computer-based training (CBT) packages from the beginning in the 1970's. These text analysis tools include options for the use of "wild-card" characters to allow for spelling errors, single word or phrase identification to assess essay or short-answer responses, and so forth. If your software requires extensive text entry, the ability to provide standard CBT text analysis is an important consideration.

So, if we decide that scored testing is a function we must be able to provide, this narrows the choices among the currently considered simulation packages to one. Obviously, we should then search for other packages that provide sophisticated text entry evaluation to widen the choices before making a final decision.

On the other hand, if testing is not a function you intend to provide, then you can choose among the packages based on their other features.

Development Features

The method of creating the simulations varies between these packages. Some capture screen shots automatically as you perform the procedure while others require you to manually capture the screen shots: some packages provide the developer the option of manual or automatic capture. Some provide automatic generation of exercises and tests from the initial demonstration; others require manual manipulation of the initial demonstration to create the other output modes. Some capture just the screen shots and require that all interactivity be specified by the developer, others capture the interactivity directly. The obvious trade off is between level of control of the simulation and the speed of development. If the automatic output is deemed acceptable, then automatic generation of different presentation modes can speed development significantly.

The ease of editing a captured simulation is also a critical feature: the more information that is tied to a given frame, the greater the consequence of deleting or moving that frame within the simulation. On the other hand, the more information captured within a frame, the less work required to complete the simulation. Given a schedule of constant enhancements to the software, ease of editing existing simulations becomes an important consideration.

Other features that impact development time include:

- Global text search and replace
- Global spell checking.
- Style sheets for global "look and feel" customization

- Built-in graphics tools for creating initial or intervening information screens within the simulation
- Linking to existing CBT/help systems

Presentation Features

Presentation Context. Most of these packages provided individual simulations that may be integrated into a variety of presentation contexts (e.g., a self-paced CBT, online help, individual training exercises presented during instructor-led training). One of these packages embeds their simulations in a CBT structure with a course, module and lesson hierarchy. Depending on how we intend to present our simulations, this CBT context may help or hinder our development.

Most vendors provide additional instructional modules that integrate with the simulation software. These may include test data-capture databases and simplified or sophisticated Learning Management Systems to track student attendance and performance. In the absence of an existing client student tracking system, the availability and quality of these modules may be an important consideration.

Presentation Modes. These modes have been mentioned before: demonstration, exercise, self-test (not scored) and criterion test. One package also provides mini help screens with the procedural steps captured in the simulation. You need to consider our immediate as well as future needs. All of the simulation software vendors are revising their software at least twice a year. It is conceivable that all these packages may provide needed presentation features in the future. However, there is no guarantee that any given software package will be enhanced with missing features that are desired, so it is more realistic to make the choice based on the current feature set. It would be wise to sign non-disclosure agreements with the main contenders to discover their enhancement schedule and direction: it may be that planned enhancements may meet future needs.

Customization. The ability to customize the "look and feel" of the presentation for different audiences may be a prime consideration. Obviously, if "look and feel" is controlled by external style sheets it will be much easier to make the necessary modifications than if the "look and feel" is embedded directly in the presentation output.

Output File Formats. Most packages now provide Macromedia Flash output. All current browsers contain the Flash plug-in so this is a viable output format for corporate delivery (unless the corporation has blocked Flash at the firewall or removed/deactivated Flash at the browser level). The other main value of the Flash format in addition to its ubiquity is that file sizes are minimized for rapid delivery.

Each simulation package provides other output file formats in addition to Flash:

- HTML/DHTML/JavaScript that simulates the Flash functionality but doesn't require any special plug-ins
- Java applets (requires the Java run-time plug-in)
- An ".exe" format for direct execution on Microsoft Windows client computers

- Proprietary formats that require special browser plug-ins (file size for these proprietary formats may vary significantly between vendors and may be a concern to the corporate IT department)
- Text output: some vendors provide the screen shots and accompanying annotation text in a Microsoft Word document format, suitable as a printed user guide. Others capture the actual procedural steps as text, which can be exported to a user manual.

In general, the more output options the better chance of meeting a specific client's needs.

Delivery Requirements. You need to determine the server requirements for each of the different packages. Theoretically, there is no limit to the number of simultaneous users of a simulation, so we will need to set a reasonable goal for the load requirements. You will need to investigate the ease of mounting and replacing simulations as well as other technical concerns.

Compatibility. If you are required to provide tracking and testing data for your courses, such data must be provided in a format compatible with the existing LMS. While all vendors claim this compatibility, data translation middleware may be required, especially if the LMS has been customized. You must consider the possibility that an LMS may be changed in the future. You must make sure that the chosen vendor can demonstrate compatibility with the most common LMSs before purchasing the system and can provide the development expertise to ensure compatibility with future LMS requirements.

Vendor Characteristics

Software Reliability. In addition to a robust feature set that meets our requirements, software reliability is an issue that must be tested in advance of purchase. Also, the ability of the developer to reliably predict the outcome of creation and editing choices is critical. If you can't be sure of how the software will react to the actions you take, frustration increases and development time is wasted. An intuitive interface and reliable operation ensure maximum efficiency.

Vendor Viability. Part of the due diligence must include an investigation of the viability of the vendor, itself. This is a competitive field; companies are being acquired by other companies as the field is "shaking out." Others may not be able to survive. We need to have some assurance that the simulation package we purchase will be around for the foreseeable future and that it will continue to be supported and enhanced by the vendor.

Training and Support. The quality of training, documentation and support is also a major concern. Some of these packages are easy to learn, others require training to master. The learning curve must be taken into account, especially if more than one developer will be required in the future.

A proposed selection methodology

1. Specify required output modes: demonstration, interactive simulation, practice test, certification test
2. List the critical features for the specified modes
3. Rate simulation software packages on each of the features
4. Select the two top contenders and do extensive due diligence on each, concentrating on production ease and speed; also test their technical support
5. Do appropriate due diligence on the companies themselves
6. Sign a non-disclosure agreement with each company to learn their future plans and direction
7. Test compatibility with delivery environment and LMS
8. Select the best compromise and negotiate purchase price, training and maintenance

APPENDIX A: SIMULATION SOFTWARE FEATURE CHART

DEVELOPMENT FEATURES

#	IMPORTANCE	FEATURE	SIGNIFICANCE	SOFTWARE
1.	A	Ability to select the key or key combination that "captures" the screens.	This allows you to select a key or key combination that isn't used by the application you are simulating.	
2.	A	Accept both single, double and right button mouse clicks	Obvious	
3.	A	Allow multiple keys to end text entry. Example: complete text entry in a field by either pressing Enter or Tab.	This is necessary to mimic the real operation of the application.	
4.	A	Automatic capture of procedural steps that works **reliably**	Makes capturing screens a "no–brainer" and much faster than manual capture. HOWEVER, few programs do this well (necessitating using the manual capture mode) and some programs don't offer this feature at all.	
5.	A	CBT-style text entry evaluation	Provides sophisticated text entry analysis using: • "wild card" characters (*, ?) to allow spelling mistakes • Specify specific word or phases that must be present • Specify specific word or phases that must *not* be present	

#	IMPORTANCE	FEATURE	SIGNIFICANCE	SOFTWARE
6.	A	Control timing of demo playback on overall and individual frame basis	Set a "base rate" for the duration of all frames, which can then be modified it as necessary on a frame-by-frame basis. NOTE: Some software will only allow minor modifications to individual frame timing, automatically setting the frame duration on the length of text in the frame's text boxes.	
7.	A	Create non-simulation frames easily from within program	Such frames can be used for introductory information, summary information, performance results and menus to jump to different parts of the simulation	
8.	A	Customization of "look and feel" to match client corporate standards	This allows for seamlessly integrating simulations into existing Website	
9.	A	Include control keys in keystroke entry	Allows for: • CTRL/SHIFT/ALT+<some letter/number key> • Function key input • Allows for the use of the Delete/Home/End keys	
10.	A	Integrated narration recording capability	It is much harder to prerecord narration than to record the narration from within the program	
11.	A	Mouse click (single/double) and keystroke sounds	Program should provide these sounds that can be added with a menu choice. If the program doesn't provide these sounds, you should be able to install them in a sound library for easy addition to a frame.	
12.	A	Multiple level undo	The ability to recover from mistakes is critical to rapid development time	

#	IMPORTANCE	FEATURE	SIGNIFICANCE	SOFTWARE
13.	A	Permitting multiple valid responses to achieve a single result	Especially for testing, students should not be penalized for using an alternate valid method of achieving the desired result (e.g., the cut/paste buttons on the button bar vs. CTRL+X, CTRL+V)	
14.	A	Provide tools and environment to support multiple developers	Providing a common environment for the storage of all simulations and their assets as well as functions such as "check-out" of specific simulations that keeps multiple developers from creating separate updates or modifications of a single simulation supports a multiple developer environment.	
15.	A	Reliably edit existing simulations by adding/subtracting/moving frames	Some programs capture mouse positions as part of the frame data. Deleting a frame can result in unpredictable behavior in the preceding/following frame because this data was deleted with the frame. Experience with the program can prepare the developer to anticipate such effects, but not having to deal with these problems at all will save considerable development time, especially if the only solution is to recapture the procedure (start all over again).	
16.	A	Simulation of selected text and overtyping of selected text	Data entry often requires overtyping automatically or manually selected text	
17.	A	Spell check of all text in the simulation	Obvious	
18.	A	Spell check of current frame only	Obvious	
19.	B	.RTF Text	Ability to highlight individual words or phrases within text boxes using bold, italic, color, font changes	

#	IMPORTANCE	FEATURE	SIGNIFICANCE	SOFTWARE
20.	B	Align cursor position with next or previous frame	Allows for correcting inadvertent mouse movement during recording.	
21.	B	Automatic creation of graded test from demonstration capture	Speeds development time HOWEVER, unless controllable by the developer, the program's automated output must be acceptable "as-is" NOTE: Even though this is a test, it may be necessary to retain or insert prompts (esp. for specific text data entry). So the automated output must be editable to be useful.	
22.	B	Automatic creation of interactive exercise from demonstration capture	Speeds development time HOWEVER, unless controllable by the developer, the program's automated output must be acceptable "as-is"	
23.	B	Automatic creation of self-test from demonstration capture	Speeds development time HOWEVER, unless controllable by the developer, the program's automated output must be acceptable "as-is" NOTE: Even though this is a test, it may be necessary to retain or insert prompts (esp. for specific text data entry). So the automated output must be editable to be useful.	
24.	B	Edit spelling dictionary	Add customer-specific or technical terms to dictionary	
25.	B	Embed text graphically in background plate	Frame timing is unaffected by text length Text can remain visible across sequential frames Text not restricted as to font, size, effects, bounding box	

#	IMPORTANCE	FEATURE	SIGNIFICANCE	SOFTWARE
26.	B	Setting of defaults for such things as the shape and color of the text boxes	Text boxes come in two standard formats: rectangular boxes for information and "balloon-style" boxes for instructions. The balloons come in different shapes that have an extended "pointer" that points left, right, up, down, etc. It is a time-saver to be able to set the default shape or, alternatively, to have the program automatically select the balloon shape based on the previous selection. The ability to set other style defaults is very time-saving, also.	
27.	B	Use of external style sheets	Enables almost instanteous changes to "look and feel" to match multiple clients' requirements or to update specific client's output	
28.	C	Automatic creation of job-aid (Procedural steps than can be accessed from within the target application)	This feature provides "just-in-time" step-by-step instructions for performing the procedure in the form of a "help text box". Duplicates the standard online help.	
29.	C	Customization of the text box color and border style	Allows customization of the text prompts that can be superimposed on the background plate (screen capture).	
30.	C	Customization of VCR controls	Allows changing the "look and feel" of the VCR controls	
31.	C	Enhance background plate (screen capture) with graphics from within program	Saves having to export background plate to an external graphics program	
32.	C	Selection of pointer style	Allows customization of the graphic "pointer" that indicates the location of the next action.	

#	IMPORTANCE	FEATURE	SIGNIFICANCE	SOFTWARE
33.	C	Spell check “as you type”/spelling errors highlighted á là Word	Handy to catch spelling errors immediately	
34.	C	Use custom dictionaries	Create custom spelling dictionaries for individual clients that include client-specific terms	

DISPLAY FEATURES

#	IMPORTANCE	FEATURE	SIGNIFICANCE	SOFTWARE
35.	A	Easy integration into existing CBT/online help	If simulations are integrated into existing CBT/WBT or online help, the transitions should appear seamless to the end user.	
36.	A	Speed of an interactive exercise should match the speed of using the real application	Long delays in response time are confusing and frustrating to end-users and have a negative impact on learning. Also, end users may make unjustified negative judgments about the actual application based on their experience with the simulation if they experience atypical delays.	
37.	B	Customization of the visual elements	It should be possible to customize text boxes, pointers, etc. to match a desired "look and feel"	
38.	B	Display demo text entry as characters typed individually or as one step Check data entry as individual characters or only after the entry is completed.	The ability to show typed input as if it were being typed adds realism, the option to show it as a single step speeds up the demo, especially if there is a great deal of text entry. Some programs capture text entry as individual characters but allow it to be "collapsed" into a single entry after the fact. For testing and interactive exercises, the ability to test each character or the entire entry each has its advantages.	
39.	C	Customization of VCR controls	Allows changing the "look and feel" of the VCR controls	

DEPLOYMENT FEATURES

#	IMPORTANCE	FEATURE	SIGNIFICANCE	SOFTWARE
40.	A	Accurate data transfer to a variety of LMS and/or guarantee of middleware to ensure such data transfer	Claimed SCORM/AIIC compliance isn't a guarantee that data will accurately transfer to a specific LMS. Companies should both guarantee and be willing to demonstrate in advance that student data will transfer accurately and reliably to the LMS in use. Since there is always a chance that a company will change to a different LMS, software simulation companies that have already created the middleware for a variety of existing LMSs provide a better choice than those who only link to a given LMS.	
41.	A	Output file types that match the corporate network delivery restrictions	IT security restrictions may prevent certain types of files from being delivered to internal end-users and/or browser plug-ins from being installed. Alternate file outputs provide a greater chance of reliable distribution/deployment.	
42.	A	Reasonable streaming server requirements	File size, scaling to meet increasing demand, etc. are critical criteria that must be investigated to ensure reliable and timely delivery to the student population. Installing and maintaining files for simulation distribution should be as easy as possible. Procedures for updating simulations by developers should be investigated with an eye toward minimal IT involvement.	
43.	C	Customization of VCR controls	Allows changing the "look and feel" of the VCR controls	