

E-learning: The Devil is in the Technical Details

Introduction

In the current literature on E-learning, one sometimes meets the claim that E-learning has met with problems of acceptance, not so much due to technical difficulties, but mainly due to a faulty educational philosophy. I would here like to comment on this hypothesis by also supplying some of my own experience, mainly from a project in E-learning over the Web, carried out in co-operation with *inter alia* our CEMS sister school in Bergen. The main conclusion of my experience is rather that, even in cases when E-learning is based on a sound pedagogical philosophy, it can meet with severe problems, often completely unanticipated, due to technological problems.

I have described some of this experience in more detail in a paper, Web-based Distance Learning System in Business Simulation – Experiences from an Inter-Nordic Course. *SSE/EFI Working Paper Series in Business Administration No 2002:4*, which is available at http://swoba.hhs.se/hastba/abs/hastba2002_004.htm. I shall here give a brief review of this, complemented with my experience from the continued work on this project carried out in the spring of 2003. I shall complement this also with some of the experience gained from developing a CD based textbook in collaboration with Professor Richard Born, Northern Illinois University.

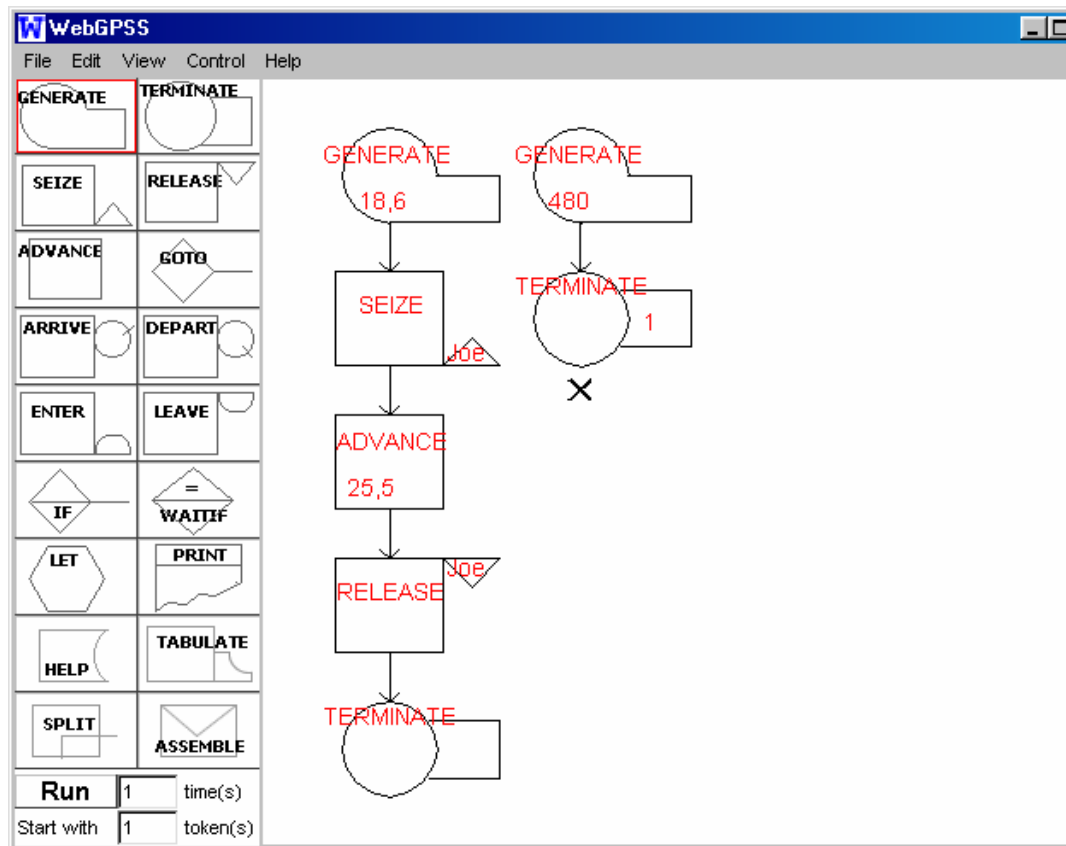
Background

I have for many in years been teaching simulation at the Stockholm School of Economics as parts of different courses. The original basis of the teaching of simulation was a software tool, called GPSS, developed originally by IBM. This software was, however, not sufficiently simple to be learnt well by business students within the limited time available in these courses. Based on feedback from over 5000 students, I gradually developed a much-simplified version of GPSS, called micro-GPSS. The use of micro-GPSS has spread to several other countries. In fact, teachers in over 50 different countries have asked for copies of this software.

Not the least because this software was regarded to be in line with sound pedagogical principles, we obtained funding of 1,000,000 SEK from the Swedish KK-foundation, which supports the development of didactic IT-tools, for producing a Web-based version of micro-GPSS. The interest in making a Web-based version stemmed from the advantages of providing simulation software on the Web, in particular with the aim that high school students should use it actively, e.g. in various courses, but also for project work during the last year in high school. The main advantages of a Web-based version, compared to a stand-alone version e.g. on a CD, are as follows:

1. The users can always be assured of using the **latest version** of the software.
2. The students can, after leaving school or college, be sure of getting **access to the software** wherever they are later going to work.
3. In many cases, a student or a teacher might want to have a **first look** at a new software product **without** having to take **risks** of e.g. viruses, when downloading it.

We developed this software in cooperation with computer scientists from the College of Karlskrona-Ronneby. This new software, called WebGPSS, was first presented in 1999 and is available at the sites webgpss.com and webgpss.hk-r.se It works as follows: A Java-based Applet is downloaded to the student computer's primary memory. This Applet provides primarily a Graphical Users Interface by which the student can build the simulation model of the studied system, e.g. a retail outlet. The student chooses, by clicking, among 18 different block symbols and builds the outline of the model in the form of a block diagram. An example of a model of a barbershop is shown below. Customers arrive 18 ± 6 minutes apart, a haircut takes 25 ± 5 minutes and the barbershop is closed after 480 minutes. Students produce this model during their very first hour of study. The parameters have been input by the use of dialogs, which also provide basic syntax information.



When this block diagram, including the parameter values, has been completed, the student clicks on a Run-button and the program is sent over the Web to the central server, where the basic GPSS engine executes the program and produces results in the form of text files, which, within a fraction of a second, are sent back to the student computer and are then presented in a number of windows in the form of tables, graphs, histograms, etc.

Besides this Applet, there is a Web-based tutorial, in the form of HTML pages, a great many Web-based program examples and an extensive system of Web-based HELP pages that explain technical details. It should be mentioned that the WebGPSS system requires that a Java plug-in from SUN be installed prior to running the system the first time on the client computer. The size of this plug-in is roughly 5 Mbytes.

This WebGPSS system is meant to be used not only in ordinary classes, but also, and in particular, for distance learning. Our first extensive experience with the use of WebGPSS in distance learning was in a joint effort of a distance learning course given from the Stockholm School of Economics (SSE) at the Norwegian School of Economics and Business Administration (NHH) in Bergen. This course, using WebGPSS, was run the first time in the spring of 2000, and has been repeated in 2001, 2002 and 2003. The courses have consisted in covering first the material corresponding to the 50 – 70 program examples available on the Web. After this introductory part of the course, the students have solved a larger exercise, dealing with pricing, production and inventory decisions of a furniture dealer. The remaining part of the course is spent on a student project, done by a group of students and dealing with a real business situation, which the students themselves choose. I have given similar courses, directly in class, for many years both in Sweden, and in the US. The experience from the distance courses given in Norway can be compared with this experience.

It should further be mentioned that the teaching was not only done over the Internet, but that the class met a few, usually three, times in Bergen with roughly a month's interval. It should also be mentioned that during the last two years the WebGPSS system has been complemented with a CD with PowerPoint slides, containing 36 lessons, each with on average roughly a dozen slides, for the learning of WebGPSS. These slides were meant to facilitate the self-study of WebGPSS.

Experiences from the use of WebGPSS in the distance course

I can compare running the course using WebGPSS in a **distance** mode with a meeting with the students only three times during the class and the **ordinary** mode, meeting the students in class on at least 15 different occasions. A main conclusion is that the result, measured in terms of the successful completion of the larger exercise as well as the individual projects, and the quality of the individual project work, does not seem to have been greatly influenced by the switch from the ordinary classroom method to the distance-learning format. At least, after a delay of a week or so, the participants in the distance course appeared to do just as well as the participants in the ordinary type of course had done. The delays had mainly been caused by technical problems.

The main disadvantage with WebGPSS has been that we during each of the courses have had cases when the server with WebGPSS was down. If the server went down in the evening or during the weekend, the students could in an extreme case have to wait more than two days before they could run the program again. Another technical problem that we encountered has been with getting the Java plug-in installed. In some years, the central computer service has, due to the risk of viruses, not allowed the installation of the plug-in. Fortunately, there was an experimental computer lab outside of the realm of the computer services, where we could install the Java plug-in. Similar problems of getting the plug-in installed has also been encountered when trying to spread WebGPSS to schools in Sweden.

Another problem that we experienced was that many of the students, who wanted to run WebGPSS at home on their own computer, found this too slow and preferred to travel to the NHH campus to run the programs in the NHH computer labs. In this regard the distance course was not truly “place independent”. The main problem mentioned was that the downloading of the Applet was experienced to be too slow. Hence, a demand for a stand-alone system, independent of the Web, arose in Bergen.

Another problem, though not encountered in this course at Bergen, has also implied a demand for such a stand-alone system. This comes from the fact that WebGPSS has also been used by other teachers in many different countries, since it has been generally available without cost. We have, however, even at times when we know with certainty that the server has not been down, received E-mail messages that the users have not been able to get into the WebGPSS system. These messages have come from southern Europe, from America and from Asia. Some of these complaints have come from universities with fast inter-net access. Although we have not been able to determine the source of this problem with absolute certainty, we guess that the Java Applet, which, as mentioned, is around 0.5 Mbyte large, got stuck in a queue at some transmitting node somewhere on the way from the far away country.

Another problem has appeared during the last years, due to the fact that Microsoft has changed its policy regarding to what extent the Internet Explorer browser supports Java. This has been caused by a strategic conflict between Microsoft and Sun. Earlier, there were no problems running WebGPSS on the newest Internet Explorer, provided the Java plug-in had been installed. During the last years there have, from time to time, been problems in this regard.

Due to these three types of problems we felt forced to develop a stand-alone WebGPSS system, i.e. a system that in its entirety is installed on the hard disk and hence does not require any net access. Since WebGPSS was written in Java, it was a fairly straightforward task to turn this Web-based Java-Applet into a stand-alone Java Application. This was done in 2001. The total cost was in this case roughly ten percent of the original cost of the system. It should be noted that with this stand-alone version of WebGPSS, the three advantages mentioned above are lost.

When installed on a PC running under Windows NT, Windows 2000, Windows XP or on a central server with these operating systems, it will run exactly as on the Web. For a PC running under Windows 98 or 95, only the block diagram GUI part ran, however, to our surprise in the same way as on the Web. The same problem also occurred when running this stand-alone version on central installations, where one for safety reasons do not allow the students to save programs on certain parts of the disk. We hence had to also a smaller complementary run module, partly in collaboration with the Otto-von-Guericke University of Magdeburg. We have also learnt the hard way that even this module runs differently under different Windows versions.

Need for physical meetings

I shall here bring up another point, namely to what extent one can have distance learning without any physical meetings at all. Our experience from the course in Bergen has been that such physical meetings are necessary for a number of reasons. As mentioned above, we have most years had three meetings. At the first meeting with the students, I get them introduced to the WebGPSS system. I can then ascertain that every student can continue with his/her studies using the WebGPSS system. A second meeting is regarded as important in particular to get the students to decide on projects and to get them organized into groups. A third meeting gives me a chance to correct programs and to ascertain that every member in each group has worked in the group projects.

The first two meetings have in fact proved to be necessary. It is my strong conviction that it would have been very difficult for **all** of the students to use the system in an efficient manner without having been shown how to do this at our first meeting. At this meeting I had the chance, by giving simple exercises to the students, to really find out that nobody was left in the dark. I am also convinced that the general discussions with the students on the project topics could not have been so efficiently carried out over the net as in personal conversations. For example, it is more difficult to convince students in an e-mail that a project plan is unrealistic than to get them to modify their plans in a discussion with eye contact, by e.g. gradually asking the students about detailed time plans.

I also found out that it often took a lot of E-mail correspondence to get a program correct. With more than 20 students, I found it timesaving, both for me and the students, to fly an additional time to Bergen. I have measured the time I spent with the students in personal sessions and the time I spent on doing the same over E-mail. Getting the students to write their programs correctly took more than three times longer time for me when doing it over E-mail than doing it in personal conversations. Sitting together in front of the same computer, we could interactively debug and correct the program, step after step, without having to close down the systems each time. When doing the corresponding thing over E-mail, I would, when a student had done one correction, have to download the new version of the program from the E-mail system to the GPSS system, start the GPSS system, run the program, find another error, report this new error and ask the student a new question about this in a new E-mail, etc.

Some other problems

Another problem experienced is that students will have not only different operating systems, but also screens with different sizes and also be accustomed to different screen resolutions. Some of our students had, for example, only 10 inch screens. Their view of the software was very different from the views of the software developer, using e.g. a 17 inch screen. It is with such a large screen tempting to include details that are not so well visible on the 10 inch screen. Avoiding too many small details on the screen is also important for the introductory lessons, when the software is demonstrated using a computer projector. The software pictures must be readable also to the student sitting in the back of the room. Only by being very

attentive to student feed-back during the development process, we hope to avoid mistakes obviously made by many others.

As mentioned above, we have run into a great amount of unforeseen technical problems, sometimes leading to high costs. It also appears very questionable that there are any substantial savings to be done by avoiding paper, as some authors claim. In fact, both R. Born, my collaborator on the Powerpoint CD textbook, and I have experienced that students are very interested in paper copies of things that are available electronically. Reading from paper has advantages from a convenience point of view; it is e.g. easier to read from a paper in a bus. Many students also have the desire to underline or make comments on the paper copy.

My general conclusion is that one must combine sound pedagogical principles with a great attention to technical details, great patience and, preferably also access to more cash than at first estimated, to be able to have any chance of success in E-learning projects.

Ingolf Ståhl
Stockholm School of Economics
ingolf.stahl@hhs.se