DEVELOPING AND EVALUATING JAVA-BASED EDUCATIONAL TOOLS

M. J. Jackson, D. I. Laurenson, and B. Mulgrew

Department of Electronics and Electrical Engineering, University of Edinburgh, United Kingdom

ABSTRACT

This paper describes the authors' research into using Java to develop tools to support the teaching of Digital Signal Processing. Focusing on the application of concepts of Human-Computer Interaction allows the formulation of design requirements which facilitate the development of tools that are both usable and useful. Developing tools meeting these requirements yields teaching aids endowed with a novel characteristic lacking from comparable Java tools - flexibility. An exemplar of this flexibility is that educators with no Java experience can configure the tools to meet course requirements. A user-centred evaluation has been performed to identify any educational gains that result from the use of the tools by educators and students. The evaluation was interpretive in nature and focused on the collection of subjective and qualitative data from both students and educators. One characteristic of the evaluation was that students and educators assessed each others behaviour when using the tools.

1. INTRODUCTION

When considering the development of Computer-Assisted Learning (CAL) tools to support the teaching of Digital Signal Processing (DSP) - or engineering in general - an important issue is the choice of software environment within which the tools are to be provided. Various factors such as students' familiarity with the enviroment, the accessibility of the environment to students, economic overheads incurred in providing the environment, ease of developing the tools under the environment, and support for modifying the tools in light of changing course requirements must all be considered.

Deciding upon Java as a software environment allows the provision of educational tools to students and other interested parties in a more flexible and accessible way than might be achievable using other programming languages or environments, especially those of a proprietary nature. Java-based tools can be accessed by students wherever and whenever they wish - conditional only on an Internet connection and compatible browser, or even off-line, via disk - providing less locational and temporal constraints on their study behaviour, removing one source of work-related pressure from students. As Picone *et al.* [1] highlight, the need to download esoteric tools is also removed since leading browsers provide support for Java. Furthermore, to use such tools students need only be familiar with the use of a web browser - a not unreasonable expectation in today's world. In addition, Java is straightforward to learn and compilers are freely available, which can keep development costs down.

However, when supporting education via computer-based tools, the development of the tools, under Java or any other language or environment, is not the only issue that must be considered. Once educational tools have been developed further work is then necessary to determine the effectiveness of the tools in achieving their primary aim of facilitating an enhanced learning experience for students and enabling or supporting their development of understanding of a concept or subject. Only by analysing the tools in use in a teaching environment can information relating to the educational effects of any educational tools be identified. Only then can decisions be made as to whether the tool development has been successful or whether further development is required.

Investigating the support of Digital Signal Processing education using Java is one concern of the WebEng Project (Web-based Procedures, Tools and Strategies for Internet-based Engineering Education). The WebEng project is being undertaken jointly by the Departments of Electronics and Electrical Engineering at the Universities of Strathclyde and Edinburgh and focuses on the development of a unified, on-line, web-based learning and teaching environment and evaluation of the effectiveness of this environment from a teaching perspective.

This paper discusses the issues that arose during the development and evaluation of a suite of Java tools for DSP education by the Edinburgh members of the WebEng project. In section 2 an overview of the tools and their characteristics is given. It is argued that the authors' tools are unique - in comparison to similar web-based tools - in terms of the flexibility of use supported for both students and educators. The paper then describes, in section 3, a user-centred evaluation designed to provide information to guide further development of the tools and to lay the groundwork for investigating issues relating to educational gains resulting from the use of these tools. The chosen approach to evaluation - interpretive evaluation - is described and justified and the evaluation plan is outlined. Finally, section 4 summarises the work described in this paper and enumerates the parties who may be interested in the results of the evaluation.

2. DEVELOPING JAVA TOOLS FOR DSP EDUCATION

The area of Human-Computer Interaction (HCI) is concerned with the construction of systems that are both usable - allow users to accomplish their tasks in a straightforward manner - and useful - allow users to accomplish tasks meaningful to them. To achieve the construction of usable and useful systems HCI emphasises an understanding of four key areas [2]: the intended users of a computer-based system, the tasks the system is intended to help the users accomplish, the technology that can support the user in the accomplishment of these tasks, and the geographical, physical and social environment in which the tasks are performed. In analysing the application of Java to the problem of developing tools to support DSP education, focusing on these four key areas facilitated the development of a set of requirements that such Java tools should satisfy. These requirements - fully motivated and described in [3] - are as follows:

- The tools should support the activities of two contrasting classes of user, acting as teaching aids for educators and learning and study aids for students.
- The tools should support dynamic visualisation and user interaction to provide lecturers with tools offering functionality not supported by conventional teaching materials and to allow students to actively engage with the tool and enjoy an enhanced learning experience.
- The tools should be operable on all major computing platforms under the main Internet browsers and under varying display sizes, so that as wide an audience as possible can access the tools from a wide a range of locations.
- The tools should minimise the requirement for the users to install additional software or plug-ins on their computing platforms, so that students especially can concentrate on achieving tasks important to them - learning DSP concepts.
- The tools should be customisable by educators to their own requirements, making the tools more attractive by allowing educators to adapt the tools to their (possibly tried-and-tested) courses (in terms of concepts or examples presented), rather than requiring that a course be modified to accomodate the use of a tool.

By observing these design requirements the authors developed a suite of Java tools to support the teaching of elementary DSP concepts. The tools are designed to be used by lecturers in lecturers as a teaching aid in addition to overhead slides; tutors in one-on-one or small group explanations to students to address any problems in understanding they may have; and students as a learning aid for use in their own study time in conjunction with their course notes, text books and other sources of information related to their course of study.

Concepts addressed by the existing tools include Fourier transforms, Fourier series, discrete Fourier transforms, windowing functions, power spectral density, convolution (continuous and discrete), correlation, FIR filters, discrete wavelet transforms and rotating phasors. The initial choice of concepts to address was guided by teaching requirements for third and fourth year undergraduate Electronics and Electrical Engineering courses at the University of Edinburgh and some tools are associated with the third year course text, Mulgrew *et al.* [4].

It is *not* this choice of concepts that makes the tools unique, nor is it their ability to be used by students as study aids, their exploitation of animation and their support for a high degree of user interaction. Indeed, other researchers and educators in the DSP, engineering and mathematics communities have produced similar tools (see, for example, [5, 6, 7, 8, 9, 10]). Instead, what *does* make the WebEng tools unique is their flexibility - the fact that they can be used on the three prevalent computing platforms (Macintosh, PC and UNIX) and under the major internet browsers, for many categories of user without additional plug-ins, unlike, say, the applets of Yeng *et al.* [9] which require a browser plug-in; are not tied to a specific display size; do not have to be embedded in a web page; can be run as stand-alone applications; and support user customisation of the display features, in particular their colour schemes. Most importantly, however, what sets these tools apart is the novel characteristic of providing straightforward support for customisability by other educators, a quality lacking in the other Java DSP tools surveyed by the authors. This provides the mechanics required to allow the tools to be configured by educators at any institution to their own ends without requiring the undertaking of Java programming.

3. ASSESSING EDUCATIONAL IMPACT

HCI emphasises the activity of user-centred evaluation as vital to the development of computer-based tools that are both usable and useful. User evaluation can range from showing the potential users prototypes and collecting their comments through to comprehensive studies in which the users' task performance, and the effect of a computer-based tool on this, is analysed in detail.

Having designed and developed some educational tools it is now necessary to see whether the benefits realised by their flexibility arise in practice, to assess whether they are a useful learning aid for students and tutoring aid for educators, to ask the question "are the tools both usable and useful?" In particular the following questions are of interest:

- Does the use of the tools lead to an increase in understanding of DSP concepts by students? When used

by the students as a learning and study aid? When used by the lecturers as a teaching and demonstrating aid?
- If such an increase in understanding arises then, why, and if not, why not?
- How easy is it for lecturers to integrate the tools into their own courses?
- How might the tools be more effectively integrated into a course as a whole?
- In what way can the tools be improved: to support students; to support lecturers?

It was intended that addressing these questions would not only provide information to assist the tool developers and drive forward the development of the tools, but would also provide information of interest to lecturers and tutors relating to how the tools can impart DSP concepts more effectively and can best be deployed to support the teaching of DSP. In the conclusion other parties who might also be interested in the answers to these questions are enumerated.

The authors' tools were deployed in the October to December 2000 term at the University of Edinburgh as aids for both lecturers and tutors and students and an interpretive evaluation was used as the means to address the above questions.

3.1. Interpretive Evaluation

HCI researchers including Monk *et al.* [11] and Whiteside *et al.* [12] have identified numerous problems with traditional evaluation techniques in which participants are monitored performing tasks chosen by evaluators and designed to test various aspects of some human-computer system. These problems relate to the artificiality of such evaluations in terms of the tasks that participants are usually set and the environment in which such evaluations are performed. The argument is that the use of some tool in a real-world environment can be a substantially different experience for the user than using the tool to perform some artificial and possibly unrepresentative task chosen by an evaluator in a controlled environment. This therefore casts doubt on the reliability or relevance of results produced by such methods in all but a few cases. This argument is particularly relevant when considering evaluation of the Java tools developed by the authors which are designed to be used in a wide range of environments by two contrasting classes of user each undertaking the related, yet contrasting, tasks of teaching and learning.

In addition, educational software evaluation incurs ethical considerations which, for example, exclude comparative studies involving the partitioning of a class into two groups with only one group being able to access the tools - if using the tools does yield an increase in understanding of DSP then this may cause resentment on the part of the students who were not provided with access to the tools. Should the tools in fact hamper understanding then the converse can also be imagined. Similarly, comparisons of examination results produced by a group of students who had access to the tools with those of previous years who did not could be contaminated by a number of other factors, not least the variance in the abilities of students or difficulty levels of examinations set from year to year.

These problems motivated an interpretive approach to evaluation focusing on the collection of information consisting of comments, views, complaints and descriptions of users' experiences and behaviour. While not yielding data that can be analysed to produce concrete objective evidence relating to improvements in student understanding or lecturer and tutor effectiveness, such an evaluation nevertheless provides subjective data from students themselves as to how they feel the tools improve, or, indeed, hinder, their understanding. This data can then be analysed in detail and lecturers and tutors can assess whether any reasons given by students for their responses are credible based on their understandings of the subject, course, and student behaviour and learning. In addition, allowing for the collection of qualitative data can provide information as to why any problems arise in the use of the tools and whether these derive from low-level interface design problems or more fundamental problems relating to the way the tool imparts information about the associated concept. Knowing why problems arise allows the appropriate re-direction of design efforts [13]. The use of studies based around the collection of qualitative data is common in the Human-Computer Interaction field (see, for example, [14, 15]).

User opinions can also provide information on the acceptability of a system [16]. Acceptability is a much more important issue than usability since a system or approach to interaction may be usable without necessarily being acceptable - users often make do with poorly designed systems that nevertheless provide some assistance with their task, and, conversely, neglect usable systems if they provide little effective assistance. Acceptability is important for educational systems since lecturers and tutors will avoid unacceptable tools and stick with more tried-and-tested methods, defending the interests of the students whose educational experiences can determine their future career path.

Collecting subjective and qualitative data also allows the elicitation of information relating to the context in which educational tools are used as well as the impact of other related artefacts, highlighting what users could perceive as contradictions between the way a tool imparts or visualises a concept and the approach taken by course texts, or differences in terminology between the tools and lecture notes. Such information can contribute to determining how various teaching aids can best be designed and improved to support and augment each other to achieve an effective, consistent and well-integrated educational experience for the students.

Since the tools were being deployed irrespective of any evaluation and the evaluation exploited the fact - rather than the tools being deployed purely for the evaluation - this resulted in an evaluation that is, in some respects, almost ethnographic in nature in that the evaluator had no control over the use of the tools, the participants or any variables but, rather, passively collected information on the use of the tools through observation and interrogation of the participants. In effect the participants - who have the most to gain or lose - were actually evaluating the tools - a major contrast to traditional approaches to evaluation.

3.2. Evaluation Data Sources, Collection Techniques and Motivations

Since there are two classes of intended user of the Java tools there were two classess of participant in the evaluation: lecturers (and tutors) and students. Each of these classes provided information to address the contrasting use of the tools as learning or teaching aids. One characteristic of the data collection and analysis was that students were asked for information concerning the use of the tools by lecturers and tutors - assessing the utility of the tools as a teaching aid. Conversely, lecturers and tutors were asked to comment upon student responses as to how they view the tools as improving their understanding to determine whether the students' comments were credible given the lecturers' knowledge of teaching and the domain of DSP. This was a consequence of the direct relationship between students as learners, lecturers and tutors as educators and the tools as a medium of communication from educators to learners. As a consequence of this the lecturers, tutors and students evaluated the following aspects of the tools.

Lecturers and tutors assessed the:

• views of students as to the effect of the tools on their understanding;
• utility of the tools as a teaching aid;
• usability of the tools.

Students assessed the:

• use of the tools as a teaching aid by lecturers and tutors;
• utility of the tools as learning aid;
• usability of the tools.

In addition, the evaluator assessed the use of the tools by lecturers and tutors in terms of usability.

Lecturers and Tutors

Lecturers and tutors provided information relating to the use of the tools as teaching aids for use in lectures and as tutoring aids supporting explanations one-on-one or in small groups of students. This information was gathered via semi-structured interviews with the lecturers and tutors who used the tools, the questions focusing on both the usability and utility of the tools as teaching aids. These interviews were recorded on cassette, the lecturers and tutors being encouraged to make any observations, comments, compliments, complaints and suggestions they deemed relevant. Use of a semi-structured interview allowed the evaluator to follow up on any interesting or unexpected issues as they arose.

Some of the interview questions addressed the ease with which educators were able to customise the tools to their course requirements since this was viewed by the authors as an important characteristic of the tools in terms of facilitating their use by educators. Another important group of questions investigated observations from the educators relating to changes in student behaviour arising from the use of the tools as study aids. This exploited the lecturers and tutors as a source of knowledge about education in general and DSP teaching in particular, and as a source of observations on how students react to explanations of DSP concepts when the tools are used to support these explanations, in contrast to previous years when such tools were not used. Such questions are particularly important since they relate directly to the primary goal of the development and deployment of the tools - to enhance students' educational experiences and increase their understanding of DSP.

The experiences of lecturers recorded in the interviews was supported by the use of video recording. The video recording provided direct examples of usability issues relating to the use of the tools and also evidence as to changes in the style and delivery of lectures when such tools are deployed. While there was a risk that video recording could prove off-putting and distracting to the students and would introduce an unnatural element into the lecture environment, its purpose was explained to reassure the students that it was the lecturer and their performance with the tools that was being recorded and not the students, and disruption to the lectures was kept to an absolute minimum.

In addition, an SUS questionnaire [17] was used to collect a quantitative measure of usability. This is a standard questionnaire and yields usability scores for a system, ranging from 0 to 100.

Students

Students provided data relating to the tools as learning aids. Again, information relating to both the utility and usability of the tools was collected and the provision of any relevant comments, criticisms, observations and suggestions was encouraged. The collection of data from students was via the use of questionnaires and the use of group sessions where student responses could be followed up. The questions were a combination of the SUS scale, ranking scales - which allowed subjective information to be summarised quantitatively - and open-ended questions following up on these to allow students to explain and justify their responses. Such a data collection method is less time-consuming for students than interviews. In particular, the use of rating scales can prove more acceptable for time-pressed students to complete than a questionnaire full of blank boxes requiring copious amounts of writing.

In keeping with the aims of tool accessibility the questionnaires were also provided on-line. However, to address concerns relating to any individual student fears that any responses they give may be attributed to them - perhaps by logging their I.D.s. - all questionnaires were also provided in Adobe PDF and PostScript so that the students could print them off and send or leave them with the evaluator. In addition, some students may just have preferred being able to complete paper-based questionnaires.

The most important class of question presented to students were those in which the students are asked to comment on any increases in understanding that the use of certain tools might have facilitated, or any confusion that might have instead arisen. These questions related to the use of the tools both by the students themselves and by educators and were specific to the concepts imparted by each tool. These questions directly addressed the main aim of the tools - to facilitate increases in students' understanding of DSP concepts. The responses to these questions were analysed in conjunction with domain experts (lecturers and tutors) who could judge whether the reasons given were credible given their knowledge of teaching and student learning

Furthermore, students were asked to comment on the use of the tools by lecturers which provides information relating to how the tools can be more effectively utilised as teaching aids by educators. In addition, since the tools are not the sole source of information on DSP concepts available to the students, issues relating to the integration of the tools with other course materials and practices were also addressed.

As the accessibility of the tools was an important design aim, some questions were designed to provide information relating to the accessibility of the tools by students, any problems they may have and how these may be addressed. This was intended to provide information to improve the accessibility of the tools.

In addition to the questionnaires, discussion sessions were held. These allowed any interested students to come along and discuss their use of the tools with a lecturer (who could follow up, elicit or clarify domain-related issues) and a developer (who could follow up on usability-related concerns). Students were assured that it was the tools and their use that the lecturers were concerned with and that they should feel free to raise any concerns they may have without fear of action being taken against them. Such discussion sessions allowed students to raise additional concerns and criticisms as well as allowing the evaluator to follow up on or clarify any interesting points raised in the questionnaires. In addition, work in progress and future plans for the tools could be discussed with the students to guide the tool development process. These sessions were recorded on cassette.

3.3. Other Institutions

Since it is intended that DSP educators at other institutions will make use of the tools it is planned that all on-line evaluation materials and an evaluation framework be made available with the tools as part of any distribution. Encouraging other educators to undertake this - or similar evaluations - provides the scope to generate a wide-ranging set of experiences from educators in different establishments as to the effectiveness of the tools and to contribute to the drawing of more accurate conclusions about what educational benefits occur, why these occur, and how these might be enhanced. In addition, this would also allow a more effective determination of suitable areas for re-development and the formation of more accurate guidelines for the development of Java tools to support DSP education in general.

4. CONCLUSION

In this paper the development and evaluation of educational software in Java has been discussed. The authors' development has yielded a set of tools for the teaching of elementary DSP concepts that are arguably unique in terms of their flexibility - supporting their use from a wide range of platforms under a wide range of browsers and screen displays, and supporting customisability of their content by educators. This flexibility has been achieved via the application of HCI principles during the design process, focusing not only on the available technology but also on characteristics of the users, their tasks and their environments.

An interpretive evaluation - focusing on the collection of subjective and qualitative information from both educators and students - was then described and justified, the intention of this evaluation being to assess of the educational effects that the deployment of the tools has on students' understanding of DSP concepts. This evaluation was performed during October to December 2000. The data from this evaluation is currently under analysis and it is intended that this will yield information relevant to a range of interested parties including:

the tool developers - providing information as to how the tools should be modified and extended to improve their usability and utility as teaching and learning aids;

lecturers and tutors - providing information as to how the existing tools should best be applied as both teaching aids for use by lecturers and tutors and as learning aids for use by students;

students - as to any positive contribution to their education that the use of such tools can make;

other developers of DSP teaching and learning tools - demonstrating the issues that must be considered when developing such teaching and learning tools;

other researchers in Computer-based Teaching and Learning (CBT) - demonstrating the pros and cons of using Java-based tools as teaching and learning aids and the issues that must be considered to ensure that such tools make a positive contribution to students' educational experiences. In addition, it will hopefully demonstrate the necessity and benefits of undertaking user-centred evaluation of CBT tools and an example of one approach to this;

funding bodies - as empirical evidence as to why the development of such CBT tools should be encouraged and funded.

The authors look forward to reporting the results of the evaluation.

REFERENCES

1. Picone, J., Hamaker, J.E., Brown, R., Hansen, J.L. and Cole, R.A., 1999, Modern DSP Education: The Story of Three Greek Philosophers, IEEE Signal Processing Magazine 9, 48-56.

2. Preece, J., Rogers, Y., Sharp. H., Benyon, D., Holland, S., and Carey, T., 1994, "Human-Computer Interaction", Addison-Wesley, Wokingham, England.

3. Jackson, M., Laurenson, D.I., and Mulgrew, B., 2000, Supporting DSP Education Using Java, "First IEEE Signal Processing Education Workshop", Waldemar Ranch Resort, Hunt, Texas.

4. Mulgrew, B., Grant, P. and Thompson, J., 1999, "Digital Signal Processing Concepts and Applications", Macmillan Press Limited, London.

5. Chen Lee, H., Crutchfield, S., Hocker, C., Kahn, S., Ma, L., Nesky, M., Rosenbaum, K., Rugh, J, W., Venkataramani, R. and Woo, B., 2000, "Signals Systems Control Demonstrations", http://www.jhu.edu/~signals/, The Johns Hopkins University, U.S.A.

6. Belkhouja, A., Rabenstein, R., and McLachlan, A., 2000, "Systool", http://www.nt.e-technik.uni-erlangen.de/~anis/systool.html, Telecommunications Institute I, University of Erlangen-Nürnberg, Germany.

7. Huber, T., 2000, "Fourier Synthesis", http://www.gac.edu/~huber/fourier/, Physics Department Gustavus Adolphus College, Saint Peter, Minnesota.

8. Hwang, F-K., 2000, "Fourier Synthesis", http://web-physics.davidson.edu/Applets/TaiwanUniv/sound/sound.html, Department of Physics, National Taiwan Normal University, Taiwan.

9. Yeng, Y., Brown, .R, Hamaker, J., Weilundemo, T., Duncan, R., and Wheeler, E., 2000, "Demos", http://www.isip.msstate.edu/projects/speech/education/demos/, Institute for Signal and Information Processing, Mississippi State University, U.S.A.

10. Tong Zhou, G. and Hong-Jing, L., 2000, "Applets for Random Signals and Noise", http://users.ece.gatech.edu/users/gtz/ee3340/java/, School of Electrical and Computer Engineering, Georgia Institute of Technology, U.S.A.

11. Monk, A., Wright, P., Haber, J., and Davenport, L., 1993, "Improving Your Human-Computer Interface: A Practical Technique". Prentice-Hall International: New York.

12. Whiteside, J., Bennett, J., and Holtzblatt, K., 1987, Usability Engineering: our experience and evolution. In Helander, M. (Ed.), "Handbook of Human-Computer Interaction", 791-817. North-Holland: Amsterdam.

13. Whitefield, A., Wilson, F., and Dowell, J., 1991, A framework for human factors evaluations, Behaviour and Information Technology 10(1), 65-80.

14. Buckley, P. and Long, J., 1990, Using videotex for shopping-a qualitative analysis, Behaviour and Information Technology, 9(1), 47-62.

15. Howard, S., 1997, Trade-off decision making in user interface design, Behaviour and Information Technology 16(2), 98-109.

16. Sweeney, M., Maguire, M., and Shackel, B., 1993, Evaluating user-computer interaction: a framework, IJMMS 38(4), 689-711.

17. Brooke, J., 1996, SUS - A quick and dirty usability scale, In Jordan, P.W., Thomas,B., Weerdmeester, B.A., and McClelland, I. (Eds.), "Usability Evaluation in Industry", Taylor and Francis Publishers Limited: London.