

Building Executive Information Systems - Visual Basic for Management Students?

Arthur Tatnall

Victoria University of Technology, Melbourne, Australia.

arthur@cougar.vut.edu.au

Abstract

One of the topics often introduced into courses in Management Information Systems for MBA students is how information systems are built. Students are introduced to concepts of process modelling, data modelling and IT project management, but not usually to programming as conventional wisdom says that it is not useful to try to teach computer programming to non-computing students. To really understand how information systems are built, however, it is desirable to have some idea at least of what is involved in programming. Executive Information Systems and client/server computing are two topics which should be introduced to MBA students, but ideally this should involve more than just talking about these concepts. This paper describes how the author introduced Visual Basic for the construction of small systems as part of an MBA course. Research findings into the success of this are discussed.

Keywords

Management Education, Executive Information Systems, Programming, Visual Basic.

1 INTRODUCTION

Like many such courses, the Master of Business Administration (MBA) at Victoria University of Technology (VUT) has a one-semester computing subject entitled Management Information Systems. As the MBA has no prerequisite for previous studies in computing, this subject attempts to introduce students, many of whom have little knowledge of the topic, to a wide range of concepts and issues in Information Technology. By the time they have completed the course the students have learned something of the various types of information systems used in organisations, issues of systems integration, data communications, how information systems are used for strategic advantage and how these systems are built. In discussing the building of information systems, students are introduced to concepts of process modelling, data modelling and IT project management. In the computing laboratory students use Excel, dBASE IV (or Access) and Microsoft Project to reinforce theoretical concepts discussed in seminars. Until 1995 however, the course involved no programming. All these topics are approached from the

viewpoint that these students will probably not, themselves, be building computer systems, but that an understanding of the concepts involved will make them better able to work with, and manager those who do.

2 EXECUTIVE INFORMATION SYSTEMS AND CLIENT/SERVER

An MIS course should consider various types of information systems, and how these systems can be of assistance to managers. Much of the literature on Executive Information Systems (EIS) suggests that it is important that these systems should be customised to the needs of the individual executive. It is thus difficult to give any meaningful demonstration of an EIS in the laboratory, and university budgets being as they are, the cost of these systems also becomes a problem. But is it possible that students could build their own small systems?

Many of the details of Client/Server computing are well beyond the scope of an MIS course for non-computing students, but the basic concepts are not. This, however, is another area where building something is a great deal better than just talking about it.

Clearly these two topics are linked and if it is possible for students to build a rudimentary EIS, this would involve understanding and using some client/server concepts. The problem, however, is that the construction of even small systems requires some programming and this has meant that it has been too difficult to include in an MIS course.

3 COMPUTER PROGRAMMING IN MANAGEMENT COURSES

Conventional wisdom says that it is not useful to try to teach computer programming to non-computing students as programming is too difficult, takes too much time to teach, and anyway, it is unnecessary for these students. To really understand how information systems are built, however, it is desirable that students have some idea of what is involved in programming. While languages like Pascal and Cobol certainly do take too long to teach to management students, and with too little gain for the time spent, there are other ways to give these students some idea of programming with tools that a typical university can afford.

3.1 End-user programming

Over the years there have been many attempts to teach programming to 'non-computing' students, but most of these attempts have been unsuccessful. Smith et al (1994) note that the 'end-user programming problem' is an unsolved one in computing. They discuss how the designers of programming languages including Basic, Logo, Smalltalk, Pascal and HyperTalk intended these languages to be easy for non-expert programmers to use, but that this has not turned out to be the case. They note that most people, however, do seem to be able to handle the editing functions in a computer package quite well, and suggest as a possible solution 'language-less' programming in which 'programming environments' that look more like editors, and make use of a good interface, are used. "Our approach is to apply the good user interface principles developed during the 1980s for personal computer applications to the *process* of programming." (Smith et al, 1994).

Attempts to involve these end-users in systems construction through teaching the use of application generators, such as the dBASE IV Application Generator, have also been largely unsuccessful. These packages are not as simple to use as their proponents would like us to believe, as anyone who has tried to use one of them with non-computing students will attest.

One area that has had some success however, is the use of macros. The first macros in Lotus 123 required a form of programming, but most current microcomputer packages allow the recording of macros, which is well within the capabilities of most people, if they are game to give it a try.

Following on from this, and the work of Smith et al (1994), some sort of 'select an option' programming seemed to be worth trying, and Visual Basic was an obvious choice. Visual Basic was chosen for the quite pragmatic reasons of availability and cost, but seems to have been a good choice for the intended purpose.

3.2 Systems development in Visual Basic

Visual Basic has been used in a third year undergraduate computing subject in the Bachelor of Business (Computing) course at VUT since 1994 and several of us have gained useful experience in what can be done easily and what cannot. Furthermore, a colleague and I have conducted research over the last two years both at VUT and at RMIT into how students cope with learning Visual Basic and what teaching techniques work best. (Tatnall and Davey, 1995) Admittedly these were information systems students, but we believed that the lessons we had learned could also be applied to MBA students, and in second semester 1996 I gave this a try. The database access aspects of Visual Basic seemed the most appropriate place to begin.

Visual Basic (VB) is an event-driven, object-based language, and constructing a system in this language involves the following steps which are quite different in sequence to those used in conventional procedural languages like Cobol:

1. Define the problem and develop an overall plan for the system.
2. Design a suitable user interface by adding appropriate controls (such as Text Boxes, and Command Buttons) to a Form, so giving the program its functionality.
3. Set properties for the Form and for each control to specify things such as its size, position, colour, caption and visibility.
4. Write code to cause the controls to respond to events like mouse clicks or changes in their contents.
5. Test the operation of the system.

The way that the 'programmer' works in designing the interface is to select controls (objects) from a toolbar and then drag and drop them at an appropriate place on the screen. The properties of these objects are then assigned using a properties form. This much is fairly straightforward and familiar to most students who have had some exposure to a Windows spreadsheet or database package. As writing code is almost the last step, it is possible to get quite a long way before coming to this, and in some cases to produce a complete system with no code at all. It is possible, for instance, to create a program to view the contents of a database, using almost no program code.

4 USING VISUAL BASIC TO BUILD A SMALL SYSTEM

In second semester 1995 the author designed an exercise where the MBA students would create a Visual Basic 'front end' to view the data in a dBASE IV database. The students had, in previous lectures, discussed the different types of information systems and the advantages of systems integration. They had also learned something of the main concepts of data modelling, having discussed the need for normalisation and the use of Entity-Relationship diagrams as a modelling tool. They then spent three or four hours in the computing laboratory using dBASE IV (Microsoft Access in 1996) and so had some idea of using microcomputer database management systems. In dBASE IV the students created a three table personnel system for the twenty five employees of a mythical cake manufacturer and retail shop - Wotan's Wonders. As programming in dBASE is well beyond the scope of this course, the exercise just involved creating tables, queries and reports. (In 1996 we used Microsoft Access in place of dBASE IV and were able to go a little further.)

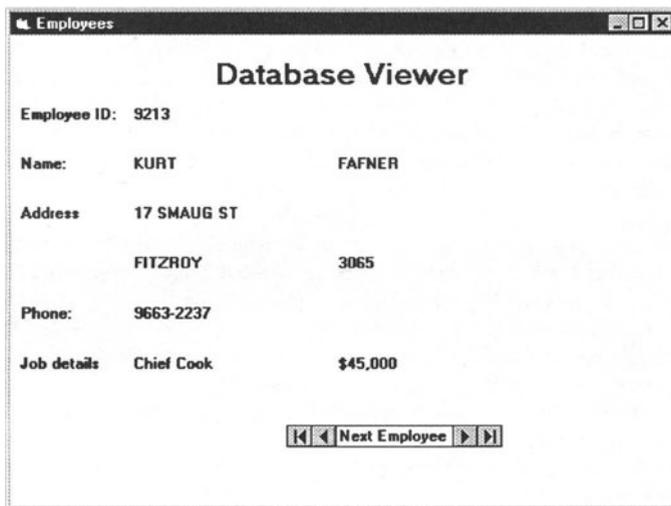


Figure 1 Visual Basic front-end (1st stage) to Wotan's Wonders personnel database.

The amount of theory necessary to make use of Visual Basic to create a database front-end can be taught in an hour or so, and two hours were then spent in the laboratory building the application. The VB front-end was intended simply to enable a user to view the data in the database, using the data control to scroll through the records from the three linked tables. The first stage of the exercise involved creating a Form with a number of Text Boxes and a Data Control to display the data. Setting the properties for each of the controls came next, followed by writing a single SQL statement to link the tables. The SQL statement was the only code that was required at this time, no VB code being necessary to get this far. All the students were able to complete this first stage. In the second stage, attempted by about 80% of the students, pictures of each employee were displayed in an Image Control on the screen. The

pictures were supplied on a data disk as .BMP files, and a few lines of VB code were necessary to display them in connection with the appropriate database record.

The exercises did *not* have as an aim that students would, unaided, be able to build systems of their own, without further study, by the end of the course. They did aim to guide the students through an example of how small systems, which illustrated client/server functionality, could be built. Although on a very small scale the system did show client/server concepts, as it extracted data from a dBASE IV server, and displayed the results in a Visual Basic client, even if both were on the same computer. Ideally the server would have been a remote one, such as an Oracle database on a mini computer, but this presented too many difficulties for an exercise such as this, and would have detracted from it. We discussed, and the students did appreciate, that if you could extract data from one local server, in principle it would be little different to extract data from other remote servers. (Davey and Tatnall, 1996)

5 EXPERIENCES USING VISUAL BASIC IN AN MIS COURSE

Although it is sometimes difficult to believe that anyone could have managed to avoid it for so long some of these students, particularly the part-time evening students, had little experience using Windows, and when it came to using the mouse to draw boxes in Visual Basic, they did have some difficulty. This was the main problem that they experienced in using Visual Basic, and could have been avoided if these students had been encouraged to attend one of the introductory computing courses run for them.

5.1 The students

The students themselves were all undertaking a general MBA course which has a single compulsory MIS subject. In semester two 1995 there were two groups comprising ten part-time and fifteen full-time students. In first semester 1996 we had three groups, with a total of fifteen part-time and thirty five full-time students. In each case the students came from quite varied backgrounds with some having studied engineering, science, law, business and the humanities. A small number of students had some knowledge of programming, but most had used a computer only for word processing and spreadsheets.

Apart from some experience in using a computer, and particularly in using Windows, students required little background before completing this exercise. Some idea of object concepts was provided in a lecture so that the students could appreciate why VB objects were assigned properties, and some elementary SQL was discussed so that they could understand how the tables were linked. It was unnecessary to say much about programming beyond the basic concepts of sequence, conditions and looping, which presented no problems to the students.

When surveyed, all students said that they found using VB a worthwhile experience, and 90% also found it quite easy to use at this level, while 10% reported some difficulty. Most were somewhat frightened off by the look of the VB code and relieved not to have to proceed much further with this.

5.2 Problems using Visual Basic

The first stages in the construction of a Visual Basic program; designing the interface and assigning properties to the objects, are quite straightforward, but the next stage is not. It would be neither worthwhile nor probably even possible, to attempt teaching these students any VB programming to the level of writing much code. Even computing students find some event-driven programming concepts to be quite difficult, and understanding what is going on a little daunting. I would certainly not advocate going this far in a single MBA subject.

As previously discussed, other research has shown that non-computer people have more success with a 'select the required operation' type of programming, and are usually able to create simple macros. One of the problems with application package macros is that it is very difficult for end-users to test them to see if they give correct results, and in many cases it is not even apparent to these users that testing is necessary. Visual Basic does present similar problems and students can easily get out of their depth without realising it. The power of this environment is readily apparent to the students, but not always the problems and the possibilities for disaster when accessing a corporate database. There is also the potential for inadvertently teaching the 'wrong things' about systems development, as conventional design tools such as structure charts, pseudocode etc are not wholly appropriate for use with Visual Basic. Finally, it is hard to get the students to stop!

5.3 Going to the next stage - construction of a simple EIS

In the time available in this single subject I have not been able to get further with my MBA students, and have had to just demonstrate the complete EIS. In another week, however, it would be possible to get quite a bit further, still with the writing of very little of code.

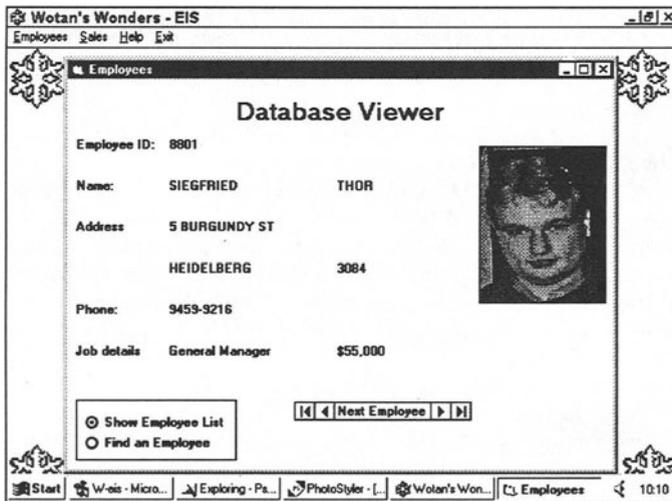


Figure 2 Wotan's Wonders EIS showing the 2nd stage of the database viewer.

A simple EIS simulation (Tatnall and Davey, 1996) could show more than just database access, by obtaining data from a variety of sources. Obtaining Excel data using Object Linking and Embedding (OLE) again requires no programming, just the setting of object properties, and is thus quite easily possible. This also gives some idea of drill down as the user can, with a double click, get right into the spreadsheet data. Dynamic Data Exchange (DDE), while offering many more possibilities, requires too much programming to be worth attempting with these students

The only other thing necessary is the creation of a menu operated system, and this can be constructed very easily using Visual Basic.

6 VISUAL SYSTEMS DESIGN

It seems clear that Visual Systems design is an important way ahead in information systems development. As well as Visual Basic, other visual programming environments like Dephi, Visual Age and Power Builder are also available, and any of these could probably be used in a similar manner to introduce design concepts to MBA students. Before we proceed much further in this direction however, some thought should be given to the building of appropriate graphic tools for designing visual, event-driven systems as such tools do not appear to presently exist.

7 CONCLUSION

From my experience, it is certainly possible to teach some 'programming' in a single subject to management students if a language like Visual Basic is employed. Making use of a 'select an option' form of relatively 'languageless' programming makes this possible. I suggest that this is a worthwhile activity in an MBA course as it can really reinforce the concepts of building an information system, and give practical examples of important topics like EISs and client/server computing.

8 REFERENCES

- Davey, Bill and Tatnall, Arthur (1996); *Tools for Client-Server Computing*, presented at Software Engineering: Education and Practice - SEEP'96, Dunedin, NZ. To be published in 1996 by IEEE Computer Society Press, Los Alamitos.
- Smith, David Canfield; Cypher, Allen and Spohrer, Jim (1994); *KIDSIM: Programming Agents Without a Programming Language*, in *Communications of the ACM*, July 1994, Volume 37, Number 7.
- Tatnall, Arthur; Davey, Bill and Burgess, Stephen (1996); *Management Information Systems - an introduction (3rd edition)*, Data Publishing, Melbourne.
- Tatnall, Arthur; Davey, Bill and McConville, Doug (1996); *Information Systems - Design and Implementation (3rd edition)*, Data Publishing, Melbourne.
- Tatnall, Arthur and Davey, Bill (1995); *Concept Development in an Event-driven Environment*, Working paper, VUT Department of Business Computing.

9 BIOGRAPHY

Arthur Tatnall is a senior lecturer in Information Systems at Victoria University of Technology. His research interests include information systems curriculum development, visual programming and project management. He lectures in systems implementation, management information systems and visual programming to computing and management students.