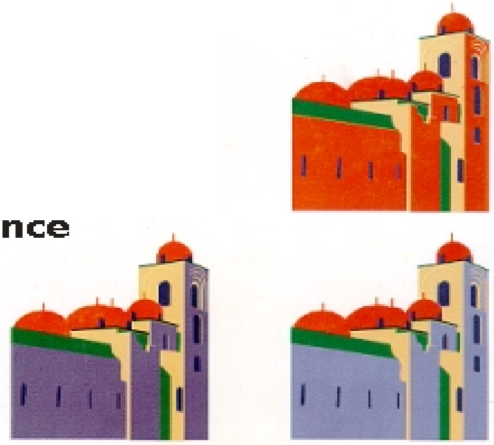
Multimedia and Architectural Disciplines

The 13th European Conference on Education in Computer Aided Architectural Design in Europe



MULTIMEDIA, A MULTI PURPOSE PROGRAMMING ENVIRONMENT

Michael Grant and Inga Paterson

ABACUS,
Department of Architecture and Building Science
University of Strathclyde,
Glasgow, G4 ONG, UK

Abstract

In the few short years since the emergence of multimedia programming tools this activity has moved from the periphery of the Information Technologies to the mainstream of computing applications. This is due not only to the progressive development in hardware and software technologies but also to the escalating set of desires of authors and users of multimedia products. Perhaps the most interesting theme within this strand of development is in the progression of the capabilities of the scripting and programming capabilities now on offer. The purpose of this paper is to trace the development of this aspect and speculate on the future of multimedia authoring tools as a new generation programming environment where the distinction between multimedia and CAD becomes less well defined.

Introduction

Over the past five years we have seen the exponential growth of that fusion of hardware and software" products that comprise the technology known as multimedia. Taken individually these products are not exceptional however what is revolutionary is the way that when they are applied in conjunction the combination can open up new horizons. Approached at a simplistic level multimedia can be compartmentalised into three categories, hardware, software and the human element of design. The hardware provides the platform on which to integrate different media, the software the tools to control it, and functionality and purpose come from within the design. As with all aspects of the Information Technologies (IT) nothing stands in the way of progress and it is all too easy to see the rapid advances in hardware and software. However it is in the progress in the design and application of these new generation products that perhaps the most interesting advances are seen and hence form the focus of this paper.

MultiMedia

Looking at the scope of applications that come under the heading of multimedia it is apparent that this terminology means "all things to all people". It is commonplace to find the most humble electronic slide show alongside a production featuring fully integrated video, audio and animation, both residing quite comfortably under the title of "multimedia". Obviously each serves it’s own purpose to good effect and if we look at some of the interim stages between these extremes it is easy to chart the development of all three of the essential ingredients of multimedia, the hardware, software and the element of design, then use this progression to extrapolate beyond the current state of the art and forecast what the future holds.

Progress

The progress in multimedia applications can be mapped out through a series of landmarks. The first of which has its origins in sequencing media and is essentially the electronic slide show. The natural evolution of the product from this point is evidenced in the development of the ability to navigate, first in a linear sequential manner, then bi- directionally, leading on to hierarchical structures and their more complex composite offspring.

Given that the contents of the slide show remain unchanged during the above evolution it is apparent that the increased ability to navigate adds functionality, not least in the users ability to sequence data as the user desires. It can be seen that the development of this theme leads to the next milestone, one that is perhaps typified by the "Architectural Archive". Essentially this category is just the components of the slide show allied to a navigational structure that allows functional groupings of data which, given these groupings, gives added value to the content.

This strand of development is finite as there is a limit on what can be achieved by purely sequencing media on some time line. To understand this point consider the classic problem of creating a virtual space where the objective is to allow the user to roam the environment by moving through the space, changing position and direction of view at will. Traditionally this is achieved by creating two matrices of images. The first is the positional matrix which may contain 20 positions, say five in the East/West axis and four in the North/South axis. Associated with each index pointing into the positional matrix is a direction matrix allowing four views, say North, South, East and West. This gives a sum of eighty images which represent the total degree of freedom allowed to the viewer. In order to interact with this virtual space it would be necessary to provide two sets of controls allowing the user to move forward and back, left and right and to view North, South, East and West.

Logically there are potential 80x4x2 individual interactions that define the navigational structure of the problem. To sequence this interaction by tying explicit actions to 640 buttons imposes obvious limits to the scope of the technique. However if the authoring environment supports a scripting language then the entire navigational problem is reduced to four conditional case statements and a simple mathematical formula.

This then opens up a whole new horizon in the functionality of multimedia applications in that the constraints of purely sequencing media are overcome and the tools can be applied in a more creative manner.

The new generation

While there are many examples that illustrate the progression of this theme it may be useful to consider, as the next milestone, an application where multimedia tools are used to recreate existing software applications. As an example consider the need to provide user training on a complex, multi-user system containing critical data. It may be inappropriate to give a novice users access to such a system yet hands on experience is vital in order to train staff in it’s operation and maintenance. One solution is obviously to provide access on a duplicate system where mistakes would not corrupt data and the trainee could explore in safety. However in an expensive hardware environment the cost of providing a machine base purely for training may be prohibitive. A solution that is increasingly being applied is to represent this software environment by recreating the application using those tools commonly used for authoring multimedia. The visual characteristics of the application can very quickly be captured by utilising screen dumps from the original and interactivity can be added by using the functionality inherent in the multimedia authoring package. In this manner the target program can be "cloned" with little effort. The overall functionality of this clone will be restricted, yet with careful attention to the composition of the "media" full functionality for a limited number of routes through the program can be simulated. This strategy can result in the creation of applications that totally mimic the look and feel of the real thing and yet in terms of time, effort and productivity cost much less. Similar economies can be obviously be found in the ability to rapidly prototype and test the interface design for new applications.

In the above example the multimedia application was used as a vehicle for presenting pre-recorded material, in response to user interaction, to simulate the operation of some existing program. The next developmental milestone is an application that is not restricted to the play back of pre-ordained material but functions, in a limited capacity, in it’s own right. A typical example might be the provision of a graphical front end to an existing simulation program. Many simulation programs, particularly from an engineering background, still lack the user interfaces that are now taken for granted in most other disciplines. These programs operate either in batch mode or at a very low level of user interaction. The ease with which multimedia tools can create such a graphical interface is their greatest strength and it is easy to see the possibilities inherent in using multimedia applications to perform data preparation to assemble an input stream that can then be used as input to older, less friendly applications. In this instance the user's prime task is to select parameters and data sets that will be used to control the eventual simulation. The capabilities of even the most modest multimedia authoring are tailor made for this aspect of functionality in that they have been designed with the prime aim of providing the tools to facilitate this very task. The same argument is equally true at the other end of the process, in that the capabilities of programs created in multimedia authoring packages can be applied to tasks related to data recovery and presentation. Again this allies the functionality of the user interface with the computationally less demanding tasks of transferring data from file and presenting it on screen. So, although these examples are not ground breaking in terms of the functionality offered, they are genuinely useful and represent a major step in the progression from the earlier instances.

The future

The final stage in this progressive development is to dissolve the barrier between conventional programming and multimedia scripting. There is no longer such a wide divide between mainstream programming languages and the scripting capabilities of the top end multimedia packages and with the computing power offered by the average PC or Macintosh it is not inconceivable that future multimedia tools may not be regarded as just an authoring package but as a fully fledged programming environment. The next generation of multimedia tools will feature compiled rather than interpreted scripting languages thus unleashing much enhanced processing power. If this comes about the product will be a unique blend of a high level programming language closely bound to an integrated and capable graphical subsystem. This vision of the future is already close at hand, even current multimedia tools provide enough of the required functionality to program simple yet robust applications that do not rely on pre-recorded media for their operation and are capable of performing simple numerical simulations. The undeniable attraction of this scenario is in the productivity offered by the high level approach to programming both in numerical and graphical areas. Compared with the learning curve and skills required to create comparable programs in conventional programming environments, such as X Windows, the cost effectiveness of this approach is very attractive.

Conclusion

Several years ago, when users were first introduced to the then revolutionary capabilities of multimedia titles, even the most simple capabilities were impressive. Yet as the technology has moved from the sidelines and into the mainstream of IT it is apparent that in today's environment titles must be faster, more capable and offer much more functionality to capture the same level of interest. This leads to an ever greater reliance on the programmability of multimedia tools as designers look for greater control over the media that they employ and will undoubtedly force the technology towards the future as suggested above. Similarly we can expect advances in hardware to influence the future of multimedia production, already it is possible to see the trend towards networking capabilities and as these aspects of functionality come on stream they will then influence the capabilities of the multimedia environment.

It is not difficult to see the future of multimedia authoring tools as a new generation programming environment where the distinction between multimedia and conventional CAD becomes less well defined. This will come about because of progress in hardware technologies, integrating functionality into a single platform, the ever increasing ability of software tools to access this functionality and the creativity of designers to utilise these developments in the production of increasingly more sophisticated and capable multimedia products. Indeed, it is perhaps true to say that the future of CAD lies in multimedia