

1. Introduction

Chapter 1

Introduction

“Research is what I’m doing when I don’t know what I’m doing.”

Wernher Von Braun

The subject matter of this thesis falls in the area commonly described as Virtual Reality (VR). Ask anybody to describe what VR is and you will get a different answer. The term was originally coined by Jaron Lanier to describe a system using immersive technology, such as Head-Mounted Displays (HMDs) and data gloves (Pimentel and Teixeira, 1993). Since then the perception of what VR is has changed, for better or worse, to encompass many different combinations of novel (and not so novel) input and output devices. The common factor between all of these is the use of three-dimensional (3D) computer graphics. The layman would therefore be forgiven for thinking that anything that uses 3D graphics is VR - a connection often reinforced by the media.

The term Virtual Environment (VE) is used to describe the environment that one enters when using a VR system. This has also become popular but it is an inaccurate description because there is nothing virtual about the environment (this topic is dealt with later). Essentially VR is used to refer to the whole subject area, its hardware, software, applications, etc., and a VE is the thing being partly or wholly simulated by the VR system.

This brief introduction describes the author’s motivation behind the work presented in this thesis. The next section outlines the author’s perspective on why VEs are modeled the way they are presently, what should change and why distribution is necessary. This chapter concludes with a preview of the contents of this thesis.

1.1 Motivation

The author first became interested in the field of VR in 1991 whilst working for a company that built real-time 3D Computer Image Generators (CIGs). There were two stages required to build an application using these CIGs: modeling and coding. First of all the geometrical and surface properties, i.e. colour, texture, etc., of the 3D objects that would populate the simulated environment were described in a special 3D modeling package. These were then converted into the CIG’s native model format and their behaviour coded into the main body of the application. The variables needed to describe the objects’ behaviours depended on the

nature of the simulation. Some objects would be under user control and thus behave as the user wanted. The behaviours of the computer controlled objects were often choreographed to obtain the best visual effect. This was usually achieved by breaking down the movements into a series of parameterised actions which were called in sequence to effect the desired behaviour.

Each time a simulation was developed, as many existing models as possible were recycled and organised with new purpose-built models in the standalone modeling package. Traditionally the application code was written again from scratch, except for a few key routines. After having used this process a couple of times, the author designed a core extensible application framework that could be specialised for each simulation. Although the properties of objects could be encapsulated and reused where possible, they were still held and manipulated separately to their geometrical representations.

After joining the Virtual Environment Laboratory (VEL) at the University of Edinburgh a year later, the need for a flexible system to model and support VEs became even more apparent. The visual perception experiments undertaken in the laboratory required many varied environments. Often these were modified slightly for various trials to provide a basis for comparison of the user's performance. Both immersive and non-immersive displays were used, complemented with appropriate input devices. The target platform for the system was a network of IBM Personal Computer (PC) compatibles. Due to the large number of devices and tasks that were required to simulate the VEs, the devices and simulation workload had to be distributed amongst these machines.

The design of the system was constrained by the technology used and it was at this point that the concepts underlying a more ideal architecture began to form. This thesis represents the development of these initial ideas into a coherent design and a prototype implementation for a system capable of modeling and executing VEs on different types of machines connected over varying distances.

1.2 A Modeling Time-Line

To understand the VE modeling techniques used today, it is beneficial to look at the heritage that has influenced the current process. With this knowledge we may reflect on existing approaches and speculate on how these will (or should) change in the future.

1.2.1 Past

The strong relationship that has been established between VR and visuals is not an accident. Pictures drawn by computers have fascinated people for the past three decades and, shortly after this ability was recognised, they were applied to a real world task. Computer-Aided Design (CAD) started its life as a two-dimensional electronic technical drawing bench and has, over the years, naturally progressed into the third dimension. Initially models were pure geometry, but as the applications of CAD increased in step with processing power, so other attributes were added such as material properties. Amongst the properties described were the material's visual appearance, e.g. colour, texture, reflectivity, etc. Nowadays high quality renderings representing realistic materials can be produced from CAD models which are used to design everything from bolts to skyscrapers.

Whilst one branch of computer graphics worked on attaining realism, another concentrated on speeding the process up so that interaction with the images was possible. The military were one of the first institutions to recognise the possible applications for real-time 3D graphics and they had the money required to fund the development of the necessary hardware and software. The resulting spectrum of solutions covered the high-end, high quality flight simulators (Schachter, 1981) down to the (relatively) low-cost SIMNET networked tank simulators (Kanarick, 1991). These simulators were built around a fast visual display but now there was also a requirement to model additional information. Not just material properties, but the attributes of the actual thing being simulated which, by necessity, also included its environment. This extra information was typically specified separately from the visual model of the simulation and both were managed simultaneously by the simulator software.

1.2.2 Present

The birth of VR signalled the start of a reintegration of the various areas of computer graphics. Technology was sufficiently advanced and at a price which meant that such systems were affordable by more people. One of the earliest applications was architectural walk-throughs which presented CAD models at interactive rates (Airey *et al.*, 1990). The line between the VR and low-end real-time simulation markets has also become blurred and, for the most part, has meant absorbing the complexity of the simulations.

Audio is now rated favourably with visuals and sound effects are not limited to plain stereo but may be positioned and oriented in 3D space (Wheeler *et al.*, 1993). Single projection displays have been joined by many types of stereoscopic displays which present a pseudo-3D view on the VE (Rushton and Wann, 1993). There is active research into tactile displays which are dependent on surface textures and their properties for the technology's success (Minsky *et al.*, 1990), e.g. softness, apparent temperature, etc. Force-feedback devices have also been used in applications, the most cited of which is molecular docking (Ouh-young *et al.*, 1988). Subsequently, there is a need for Physically-Based Modeling (PBM) of the VE which can rely on a considerable number of variables and equations. Of course there is no requirement to develop VEs that closely model our own environment, which means the structure and content of the information accompanying the seemingly obligatory visuals can vary a great deal. Indeed, it may be beneficial to model information that is not part of the VE *per se* but affects how objects interact within it, e.g. medium, aura, focus, nimbus and adapters (Benford and Fahlén, 1993).

Attempting to meet this sudden increase in information, existing visual modelers have been retrofitted with new features to accommodate some of the non-visual information that designers want to model, e.g. audio links, behavioural information, physically-based modeling parameters, etc. The result is often unwieldy and inflexible with modeling still centred around visuals instead of approaching the modeling task without bias. This is, in fact, the best case; it is still common to find integration of data within the application rather than at a higher-level. This is partially due to the fact that each VE system uses its own modeling system with a proprietary structure and format. Certainly any exchange of information requires an explicit conversion process which can often lead to a loss of detail and/or a sub-standard content.

1.2.3 Future

The amount and type of information that needs to be modeled will inevitably increase and, unless a suitable flexible framework is adopted, the VE model may collapse under its own weight. Standardisation of any area is generally a bad idea when that area is not well understood, but if each proposed solution is sufficiently flexible then there is the possibility of a gradual merging until, eventually, only one form exists. This approach can be applied to VE modeling which can take advantage of the benefits of standardisation to aid high-level tools development and ease data exchange. A good starting point for the development of such a model would be the elimination of the emphasis on any one type/medium of information used to build a VE, e.g. visuals, audio, etc.

1.3 Distributing Simulations

The more complex a model becomes, the more computing power a system will need to execute it. Only so much computational power can be squeezed into a single machine and, for anything other than small models, it will be necessary to distribute the simulation between a number of machines to cope with the extra load. In this way more efficient use of each machine's resources can be made and the possibility of multiple user interaction is introduced.

The problems of distributing a simulation over a number of machines are many and are compounded by increasing the distance between machines. These problems are slightly different depending on the combination of hardware used and the geographical distance covered. There is no one technique that can be applied at all levels of distribution that will address all of the problems posed. Therefore a suitable multi-level solution is needed that applies the right technique in the right place.

Ideally, the modeling technique should influence the architecture of the simulation system but it is not uncommon for this situation to be reversed (DIS, 1994). If improvements are to be made to the modeling process, it is essential that the underlying system provides the comprehensive support necessary.

1.4 Interactivity

The work presented in this thesis first takes a broad look at VE systems and then concentrates on a specific aspect: *interactivity*. The adjective “interactive” is commonly used to indicate that the thing it is applied to runs at a fast enough rate to form some relationship with the human user. Many of the observations and techniques described in this work are valid regardless of the applications such a system is applied to, but, in light of the primary concern, the emphasis has been placed on two factors: *consistency* and *real-time*.

Consistency refers to the problem of ensuring that the VE each participant is interacting with appears the same in spite of the fact that it may be distributed over a number machines covering a certain geographical distance. It also deals with the issues regarding multiple users and the problems they bring, e.g. two users may not simultaneously manipulate the same properties of a given object.

Whilst interactivity is a goal, “real-time” identifies a set of techniques that may be used to realise that goal. The latter term is often confusingly used to describe interactive systems as the author will find in chapter 2 when current VE systems are reviewed. However, the author has attempted to distinguish the two starting in chapters 3 and 4. Consequently, real-time has been applied in two ways to the original work in this thesis. Firstly, to describe real-time displays that produce a fast, constant update rate to enable effective interaction; and secondly, to describe the fundamental nature of the system that permits these types of displays to be realised and support consistency.

Real-time displays are a requirement of the ideal VE system considered here, but are not essential for all the applications that such a system may be used for. For example, somebody visualising a complex data set may be happy to tolerate a few display updates a second, whereas a pilot in a flight simulator may find his job very difficult if the display is updated less than 60 times a second. These examples may also be used to scope the importance of consistency. Modification of one part of the data set whilst another person views a different portion may be perfectly acceptable if it does not affect that person’s task. On the other hand, suddenly introducing another plane into a networked flight simulation or perhaps removing part of the terrain could have quite profound consequences.

It is very important to understand that a real-time display is *not* a physical display that is just updated fast, e.g. a monitor, it is a display of the VE which itself is updated at a fast rate with a constant duration in between updates. Possible types of displays include visual, aural and tactile.

1.5 Thesis Preview

Chapter 2 presents a method of classifying the issues involved in the design of a system capable of distributing VEs. Existing solutions to this problem are described, including their approaches to modeling, and then compared using the classification scheme.

Chapter 3 looks at the whole concept of environment modeling, reassesses what we are trying to accomplish and presents a new approach to the task. During this process, the structure of our natural environment is examined in the hope that it will provide enlightenment about modeling in general. This section concludes by deriving a suitable definition and abstract model for a VE. Finally, an aspect of human-computer interaction is highlighted which has implications on how VEs are simulated. Many systems today have variable-rate displays that distort some of the information a human uses to make decisions. A visual perception theory is used to demonstrate how a constant-rate display can resolve this problem.

Based upon the knowledge gained in the previous chapters, the design of a new distributed VE system is presented in chapter 4. First of all, a flexible modeling language is described that is integral to the system architecture. Rather than targeting a specific set of hardware or geographical distance, the system solution is structured in such a way that the correct techniques are applied at the right time, so that all configurations may be supported.

The implementation of a prototype system is described in chapter 5. Not all of the design’s elements are fully implemented, but it is sufficiently represented to verify the viability of the ideas used in the proposed solution. Each of the core system components are dealt with in

turn, addressing the key decisions taken during implementation and the major data structures used.

Chapter 6 is an evaluation of the prototype which was implemented on a number of test platforms. Performance of the building-block components is established before dealing with system performance as a whole. The chapter concludes by outlining a number of enhancements that could be made to the design and implementation in order to improve the prototype's performance.

The thesis concludes in chapter 7 with the application of the classification scheme to the proposed system, a summary of its most important features and suggestions for further work.

1.6 Summary

This chapter began with a cursory introduction to VR and VEs which is significantly expanded in the next two chapters. The author's motivations for this work were based purely on practical experience, combined with the wish to make the development of and interaction with VEs less painful. The reasons for the current state of VE modeling were outlined and their weaknesses exposed. At the centre of any solution to this problem is the modeling system. A more flexible approach is required, as well as the underlying framework to support this process and an integrated modeling/simulation system capable of handling the result. The road to a new software architecture begins with an examination of existing system solutions.