Chapter XVI
Developing Relationships Between Autonomous Agents: Promoting Pro-Social Behaviour Through Virtual Learning Environments Part II

Scott Watson
University of Hertfordshire, UK

Kerstin Dautenhahn
University of Hertfordshire, UK

Wan Ching (Steve) Ho
University of Hertfordshire, UK

Rafal Dawidowicz
University of Hertfordshire, UK

ABSTRACT

This chapter is a continuation from Part I, which has described contemporary psychological descriptions of bullying in primary schools and two Virtual Learning Environments (VLEs) designed as anti-bullying interventions. The necessary requirements for believable, autonomous agents used in virtual learning environments are now outlined. In particular, we will describe the technical and engagement-oriented considerations that need to be made. The chapter concludes with recommendations of how to meet these needs and how to design a VLE by including potential users in the development process.

INTRODUCTION

Part I of this chapter has described how human social networks operate, and have focused specifically on the issue of childhood bullying within primary school classes. We also introduced two VLEs (FearNot! And C-SoNeS), which have been created as anti-bullying interventions.

In developing these VLEs, the authors have uncovered a number of issues which needed to be resolved successfully in order that the software was developed to be functionally and pedagogically sound. Many of these issues seem generalisable and pertinent to developers of other VLEs, and so this chapter aims to impart some hard-learned lessons to allow the greater community to prosper from our endeavours.

This chapter intends to show what considerations are necessary when designing an engaging VLE and will outline a number of ways in which these requirements can be met. In this way, it is hoped that future development teams can benefit by drawing from our experiences in the design and implementation of VLEs.

LESSONS TO BE LEARNED FOR THE DEVELOPMENT OF VLES

As part of the development team of FearNot! and C-SoNeS, the authors have learned a number of lessons which are hoped to be of use to future developers of VLEs for young people. Broadly speaking there are two main considerations to make: technical considerations and engagement-oriented considerations.

Technical Considerations

Technical considerations concern the design and implementation of a VLE from a software developer's point of view. The most important aspect here is to ensure that the VLE works and is stable enough when installed on school computers – children will not learn anything from software which does not work! This may not be so much of a concern for VLEs which are not designed for the classroom, such as the NICE system (Johnson et al, 1998) which uses a CAVE environment, for the simple reason that researchers often have more control over, and access to, the available hardware for development and testing. However, this concern should be carefully considered by developers of systems designed to be used in the classroom, for a number of reasons.

Firstly, programmers will not usually have access to a school's computers during the development of software. This is an issue because schools (at least in the UK), while often possessing relatively new PCs, do not equip their computers to as high a specification as those usually used in a developer's laboratory. It is a simple fact that primary schools do not need as powerful machines as those used in software development, and so prefer to purchase lower specification computers in order to keep their expenditure to a minimum. For example, in installing FearNot! in local primary schools for evaluation, the authors found that most school computers have rather low specifications in terms of slow processors, a lack of RAM, and do not usually have separate graphic cards (rather they share on-board memory between graphics and RAM). While a general survey of the current state of computers in UK primary schools was conducted by the e-CIRCUS project's educational experts, it was almost impossible to determine the precise specification of equipment available in schools. This has caused some problems since the FearNot! application runs in the OGRE 3D[1] graphical engine, and has led to the undesirable situation where some schools who want to take part in the project are unable to because their systems simply do not meet FearNot!'s minimum system requirements.

It is very difficult to determine a general level of computer sophistication in primary schools because there is large variability. Schools are responsible for the purchase of their own computers in the UK, meaning that there is no consistency in terms of the systems installed. There is a wide range of choice available for schools; from laptops to desktops, different suppliers, and different combinations of hardware. This compounds the issue of reduced performance further – not only do developers have to 'scale-down' their applications, but they must also be stable enough to run across many different configurations of low specification machines.