

A Demonstration of ScriptEase Ambient and PC-interactive Behavior Generation for Computer Role-Playing Games

Maria Cutumisu, Duane Szafron, Jonathan Schaeffer, Kevin Waugh, Curtis Onuczko, Jeff Siegel, Allan Schumacher

Department of Computing Science, University of Alberta
Edmonton, Canada
{meric, duane, jonathan, waugh, onuczko, siegel, schumach}@cs.ualberta.ca

Abstract

ScriptEase is a visual tool that enables game designers to create complex interactive stories for computer role-playing games, without programming. In particular, ScriptEase automatically generates the scripting code for ambient and PC-interactive non-player character (NPC) behaviors from a set of *behavior patterns*. Without ScriptEase, a game designer would have to write scripting code manually to specify NPC behaviors. This demonstration describes the steps of generating complex and non-repetitive *ambient* and *PC-interactive* behavior scripts using generative *behavior patterns* with ScriptEase. We show how *ambient* behavior patterns are used to re-generate and improve the behaviors of all ambient NPCs in the Prelude module of the BioWare Corp.'s Neverwinter Nights official campaign. We also demonstrate *PC-interactive* behaviors for a guard NPC in a custom Neverwinter Nights game module. With ScriptEase behavior patterns, game designers can easily and quickly populate a story with an engaging group of NPCs.

Interactive Character Behaviors

Many non-player characters (NPCs) that could potentially enrich a game adventure display repetitive behaviors due to the difficulties that occur when trying to script interactions between them [3]. For example, in games such as Fable and Morrowind, the state-of-the-art character behaviors are repetitive, with characters that rarely interact with each other [1]. NPCs walk predefined paths, make random comments about the player character (PC), stand still or perform a simple animation. In Sims 2, ambient behaviors are more developed, but they are very dependent on the design model integral to the game. The most novel and challenging ambient behaviors are the ones that use behaviors collaboratively (interacting NPCs) and these are rarely seen in computer role-playing games (CRPGs). We have developed a mechanism that generates engaging NPC behaviors without writing code. Our *ambient* and *PC-interactive* behavior patterns generate complex and non-repetitive NPC behaviors. A behavior pattern is defined by a set of behaviors and a three-fold control model that selects the most appropriate behavior at any given time. A

behavior can be used *proactively* in a spontaneous manner or *reactively* in response to another behavior. This demonstration illustrates the coverage of our *proactive/reactive* models. We also support *PC-interactive* behaviors, where the PC's actions affect NPC behaviors and we demonstrate a Guard to highlight our *PC-interactive* model. These behaviors are sufficiently interesting to assess the use of ScriptEase [5] behavior patterns for generating scripting code for NPC behaviors. An *ambient behavior pattern* generates two types of NPC behaviors. An *independent behavior* occurs when the NPC acts alone. If the NPC interacts with another NPC, its behavior is *collaborative*. We conducted a case study for the Prelude of Bioware Corp.'s Neverwinter Nights (NWN) campaign story directed at both *independent* and *collaborative* behaviors. Collaborative behaviors are rare in CRPGs because their existence complicates event synchronization. Halo 2 [2] has support for "joint behaviors" but the game designer still has to write custom code for these behaviors. We developed a concurrency control mechanism that solves the inherent synchronization problems [1]. An *eye-contact* mechanism prevents new ambient behaviors from being initiated while an NPC is still executing behaviors in progress or when a PC-interactive behavior occurs. The synchronization model used by ScriptEase behavior patterns is scalable to more complex character interactions. Our behavior pattern set can be generalized to cover the behaviors of NPCs in other settings, such as a tavern, commonly occurring in CRPGs.

Ambient Behavior Patterns

Each behavior pattern is composed of basic behaviors that are re-usable and easy to assemble together. The *Duet* ambient behavior pattern is used to simulate the behaviors of all interacting pairs of NPCs that take turns in accomplishing a collaborative task, such as the spellcaster-spellcaster, spawner-destroyer, and speaker-speaker. To demonstrate the utility of our approach, we removed all of the manually scripted NPC behaviors in the Prelude of the NWN official campaign and we show a demonstration of a revised Prelude, where all NPC behaviors are generated from our patterns. We show that the resulting scene is much more entertaining than similar scenes in the original game module. As a result of this study, six new ambient

behavior patterns were identified: *Poser*, *Bystander*, *Speaker*, *Duet*, *Striker*, and *Expert*. These patterns were sufficient to generate the ambient behaviors for all the NPCs in the Prelude of the NWN campaign story. In this demonstration, we will show a new Prelude module with ambient behavior scripts generated from these patterns [4].

PC-interactive Behavior Patterns

A *PC-interactive* behavior pattern is a high-level description of an NPC behavior with respect to the PC. These behaviors are not considered ambient, since the player character is controlled by the player and the NPCs are influenced by the PC's actions when the PC is perceived. We constructed a *Guard PC-interactive behavior pattern* to illustrate our PC-interactive model [4]. The Guard has the following *proactive* behaviors: patrol near the guarded object (e.g., a chest) in a non-deterministic manner, rest on a bench, check the guarded chest, and pose (perform an animation). The guard has the following *PC-interactive* behaviors: watch for the PC, warn the PC when the PC is close to the guarded object, and attack the PC when the PC is very close to the guarded object. The guard performs its proactive behaviors and is interrupted by PC-interactive behaviors.

Demonstration Overview

This demonstration shows the steps of generating complex *ambient* and *PC-interactive* behaviors with ScriptEase, in the context of two NWN modules. The first module, NWN Prelude, represents an original game module from the official NWN campaign story that we refactored to verify the functionality and scalability of ScriptEase ambient behavior patterns. The second module, Guard, represents a custom module that we created to illustrate PC-interactive behaviors. With a few behavior patterns and the exploitation of their underlying character interaction concurrency mechanisms, rich and non-repetitive behaviors can be easily created with ScriptEase. Game designers can easily and quickly populate a story with an engaging group of NPCs. There are three steps in creating an interactive CRPG story with complex ambient behaviors using ScriptEase.

Step 1: Module Creation

Create a module with several NPCs using BioWare Corp.'s Aurora toolset provided with the NWN game.

Step 2: Scripting with ScriptEase

Use ScriptEase to specify the behavior patterns. The game designer opens the module in ScriptEase and instantiates the behavior patterns. To re-create a scene from the NWN Prelude game module, appropriate ScriptEase behavior patterns are instantiated. For example, Figure 1 illustrates the *Duet spawner-destroyer* pattern selected to express a spawn-destroy collaborative behavior. This pattern's options are set to an actor (the spawner), a partner (the

destroyer), and a skeleton (the creature that is spawned and then destroyed). Depending on the author's intent, the pattern can be further adapted. Finally, the module is saved and compiled with ScriptEase that automatically generates standard NWScript code for the game module.

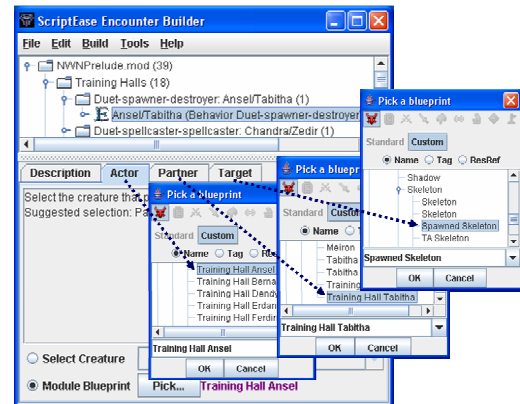


Figure 1. An instance of the Duet-spawner-destroyer pattern in the NWN Prelude.

Step 3: Play!

Play the module in the NWN game. This demonstration shows more entertaining and engaging ambient behaviors for NPCs in the refactored Prelude.

Conclusion

This demonstration illustrates how *ambient* and *PC-interactive* behaviors can be inserted into BioWare Corp.'s Neverwinter Nights game, improving the overall game experience. *Ambient* behavior patterns are used with ScriptEase to easily and quickly re-generate all of the ambient behaviors of NPCs in the NWN Prelude and improve them. *PC-interactive* behaviors are used to script a more engaging guard that is perceptive to the actions of both the PC and other NPCs. This demonstration shows how patterns hide the level of complexity necessary to create a realistic interactive story and how patterns can be re-used for several NPCs.

References

- [1] Cutumisu, M., Szafron, D., Schaeffer, J., McNaughton, M., Roy, T., Onuczko, C., Carbonaro, M.: Generating Ambient Behaviors in Computer Role-Playing Games. LNAI 3814, Intetain 05, Italy, 2005, pp. 34-43
- [2] Isla, D.: Handling Complexity in the Halo 2 AI. Game Developers Conference, 2005
- [3] McNaughton, M., Cutumisu, M., Szafron, D., Schaeffer, J., Redford, J., Parker, D.: ScriptEase: Generative Design Patterns for Computer Role-Playing Games, Proc. 19th ASE '04, Austria, 2004, pp. 88-99
- [4] Prelude and Guard Movies.
<http://www.cs.ualberta.ca/~script/aiide06>
- [5] ScriptEase. <http://www.cs.ualberta.ca/~script/scriptease.html>