Reading Report: Automatic Customization of Non-Player Characters Using Players Temperament

Kyle Li

School of Computer Science, McGill University, Montreal Qc H3A 2T5, Canada
yifan.li@mail.mcgill.ca

Abstract. Non-Player Chracters(NPCs) are becoming more important a part in the nowadays video games, especially in Role-Playing Games(RPGs), in which game designers are trying to make the bahavior of the NPCs believable and interesting. This report is going to review a model proposed by Hector Gomez-Gaucha and Federico Peinado[1], who try to customize the NPC behavior according to the players temperament. Ontologies[2] and Case-Based Reasoning[3] are used as the reasoning model for adaption. I will use a fantasy RPG - Never Winter Nights(NWN) as an example to illustrate this process.

1 Introduction

One of the major goals of video games is evoking player's emotion through various interactions in the game. Current game design is much focused on making the interaction challenging and interesting to keep the player wanting to explore more. However, players have different temperaments, thus will be affected by interactions in a different way. It is naive to assume that one particular NPC behavior is interesting to everyone.

I will first explain what game customization is, and current approaches for it. Most current approaches will result in NPCs with various appearances and skill levels, but static behaviors.

To make the behavior more complicated, I will introduce Keirsey's theory[4] as a simplified model to describe human temperament. By Keirsey's theory, a person's temperament is a combination of four basic temperaments - Artisan, Guardian, Idealist, and Rational. This information for a particular player is retrieved before the game session.

Once the player's information is known, I will use Ontologies and Cased Based Reasoning(CBR) to reason about that information, and calculate the most appropriate customization. There are three steps involved. I will first retrieve a case similar to the current player from the case base. Then I will use Ontologies to adapt that case to make it more appropriate for the current player. After the game session, the player's feedback is going to decide if the new case is usefull and can be stored for the future.

This model has its limit too. Information about a player has to be collected in a static way for one thing. I will point out some of the future works for this model in the conclusion.

2 Current Approaches for Game Customization

Most RPGs nowadays employ a conversation tree and the interaction are state-controled. Some games offer NPC customization but require manual configuration. Blade Runner is an RPG that is replayable by employing different NPC behavior every time the game is played, even though the auto-customization is not based on the particular player. Whereas some action game such as Max Payne, can adjust the NPC skills automatically based on the player's skill level. Fable, another RPG, rates the player's action as good or bad, and change NPC behavior based on player's reputation. This is truly dynamic, but rather simple to judge everything as either good or bad.

It is becoming common that intelligent components of games are developed as a piece of software by itself, and get linked to game engines through middleware. Some of these intelligent systems include: Zocalo, I-Storytelling, and KIIDS which uses CBR and Ontologies.

This report focuses on CBR and Ontologies, but is not subject to any particular platform. In the following section I will use a particular scene in NWN to illustrate the process.

3 The example Scenario

The game I will use is Neverwinter Nights. The player's avatar in the game is called Drax. The scenario is that after Drax defeat evil monsters and returned to his castle, his servant is trying to welcome and comfort the master.

in figure 1, there are a number of things that could be customized. By default, the servant will fall to the ground and the default text will show. The servant will then start fast worshiping animation. This could be a funny animation to some players. But might not be appropriate to players with a serious temperament. It could be even offensive to players from a certain culture. There is no single NPC behavior that is good for every player, but rather various behavior for various player types. To do that, we need models to describe the player types.

4 CBR and Ontologies

Ontology is a term from philosophy that describes entities in a particular domain. One can specify individuals, classes, properties, roles as Ontology entities. I will use Ontologies to describe all the facts in our domain such as player tempraments, customization actions, and possible adaptions.

CBR defines a case base. Each case is a problem-solution pair. The problem describes what temprament the player has. The solutioin is the command to

Fig. 1. Servant: Welcome home, sir. [Repeating fast worship movements] Drax: Hum, I feel so tired... Please, servant, prepare the bath. Servant: Immediately, sir. [Running to the bathroom] Example dialogue at the 13th level of politeness

change NPC behavior so it looks more interesting to that player. This could be done either in game native commands or some scripts developed for the game. CBR represents the knowledge that we have about customization.

I will integrate the knowledge and facts together by defining both problems and solutions as entities of Ontologiy and build knowledge intensive CBR(KI-CBR). After we build the case base, the whole customization process follows like this:

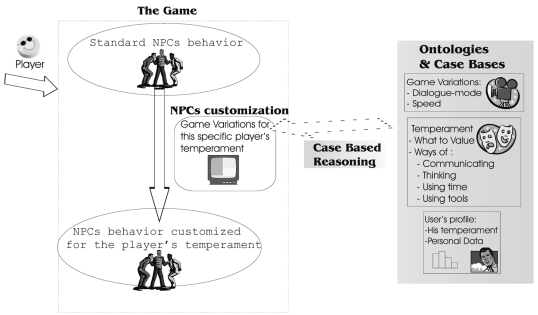

Fig. 2. Elements of the automatic NPC customization model

The model starts with a player playing the game which has a standard NPC behavior. The CBR cycle will retrieve a case that is similar to that of the player. The customization is performed to make NPC more similar to the player. Then if the distance between the retrieved case and the player is more than the thresh-

old, the case will be adapted to make the NPC behavior more suitable. The Ontologies of different entities are discribed next.

5 Temperament Ontology

Based on David Keirsey's theory, which is very widely applied in psychology, each person has a unique proportional combination of the four basic temperaments. In this report I will use a unique combination as the new case: Artisan 10%, Guardian 10%, Idealist 30%, and Rational 50%.

Usually one of the temperaments is predominant, which means the person will behave like that type most of the time. Rational is predominant in our case.

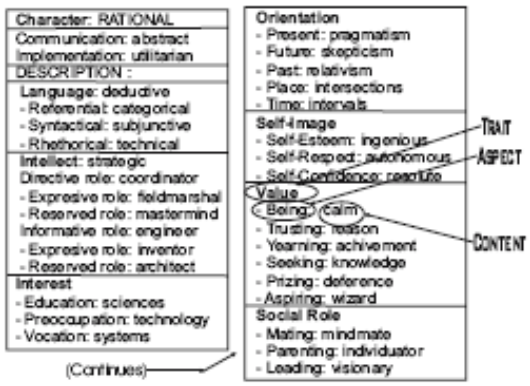

Fig. 3. The basic temperament Rational and its traits

Above is the description of Rational basic type. The description has several traits. Each trait has several aspect. Each aspect has a value. Each trait defines how people behave in each aspects of that trait. For example, the Value trait has an aspect Being, which has value calm. That means Rational people value most or like to be calm.

6 Variation Ontology

Variations are all the possible changes for customizing NPC behavior. For example if we want to implement one assertion: "a polite NPC with slow movements makes you feel calm", then there are two aspects in this customization: an NPC's politeness and speed of movements.

The customization is carried out either by game native commands such as changing the value of a varialbe which decides the level of politeness of NPCs, or by performing scripts which are developed for the game. Both of these are described in hasActivationCommand and hasActivationParameters traits.

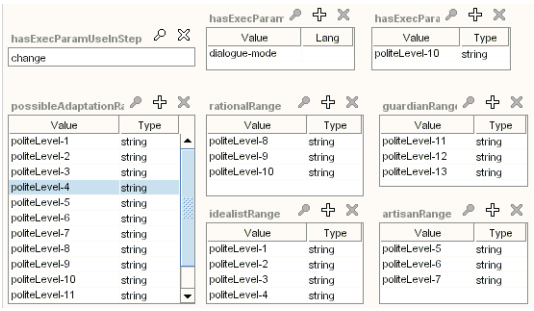

Fig. 4. ExecParam of a variation concept and ranges to be adapted

7 Mapping Temperaments into Variations

Each variation may affect some traits of a specific temprament. This is captured by relating affectToTraits trait of Variation to a particular trait in a Tempra-ment. In the figure below, affectTotraits has value ValueBeingCalmX, which means it is related to the Value trait of Rational temperament. Now we can prmote the calmness by performing the changes defined in the Variation.

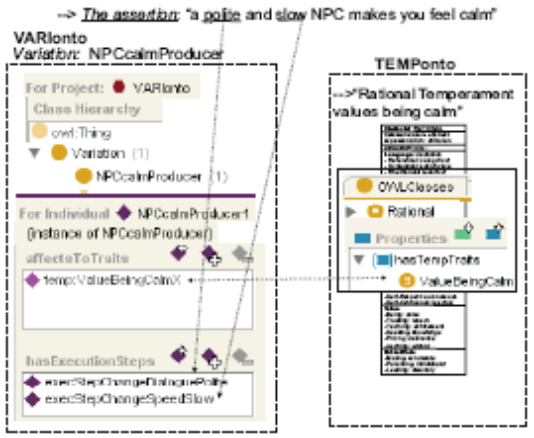

Fig. 5. An assertion as a variation related to a specific trait of a temperament

Also notice that the same variation affects different temperaments in a to-tally different way. For example, the same dialogue change affects Value trait of Rational temperament in a different way to the same trait of Artisan tem-perament. This is becase Rational people appreciate to be calm while Artisan people appreciate to be excited. This is tackled by having different parameters for different temperaments.

8 Player Ontology

This is where the player profile is stored. Player Ontology has 3 traits. hasTemperamentProportions defines what combination a player has. hasUserType defines the best match from the case base. hasUserAdaptions defines the adaption needed to adapt the user type to a specific player.

9 CBR Process

First we get the description of the player by Keisey's questionnaire before the game session. Suppose we get a player description like this: Artisan 10%, Guardian 10%, Idealist 30% and Rational 50%. From the case base, we get the most similar case e.g. Artisan 10%, Guardian 30%, Idealist 30% and Rational 30%.

To measure the similarity between the two cases, the sum of the differences between each components is calculated:

$$\sum_{T=Artisan}^{Rational} (\%NewCase_T - \%RetrievedCase_T) * Correction_T$$

A CorrectionFactor is used here to make sure the sum is large as long as the predominant temperament is very different, even if the rest of the temperaments are similar.

The model reuses the retrieved case by adapting it. In our example, Atisan stays the same, Guardian must be increased by 20%, Idealist stays the same, Rational must be 20% less. The model uses the possibleAdaptionRange trait of Variation to perform the adaption. For example, to make the case 20% more Rational, we look up the list of values in the adaption range of Rational. This list of values goes from 0% to 100%. We use the new proportion of Rational as an index to get the corresponding element in the list. After the adaption, the modified variation is the new case solution and will be executed. The following figure shows the same scenario after customization. The text is adjusted to be more polite and the animation of the servant is different. This is because the game knows the player is Rational predominant, so the game politeness has increased from 13th level to 8th level.

The last step is to get the feedback from the player after the game session to decide if the new case is a useful one, and store usefel ones in the case base for future reuse.

10 Conclusions

I introduced a model that integrates CBR and Ontology to customize NPC behavior according to player's temperament. I used Never Winter Nights as an example to illustrate the process.

Fig. 6. Servant: Welcome to the castle, sir! Everybody has heard about your courage in the battlefield. Drax: I am tired because of the battle, servant. Prepare my bath. Servant: At your command, my lord. [Saluting him] A perfumed bath is ready for you. Example dialogue at the 8th level of politeness

First of all, this model is independent of any particular game. Once we build the case base and the player's profile is known, it is possible to use them customizing multiple games by changing the activation commands only. The model can also be used to reason about group behaviors such as families, troops or even the whole cast of the game by using a temperaments compatibility table.

The model has its limit too. Building the case base is time consuming because it has to be continuously refined based on the player's feedback. The model is not truly dynamic because the player has to fill out a questionnaire before the first game session. This is annoying and not feasible for most of the commercial games. The future work would be to use some heuristics to get this information according to the player's game action.

References

1. Gomez-Gaucha, H., Peinado, F.: Automatic Customization of Non-Player Characters Using Players Temperament. Depto. Ingeniera del Software e Inteligencia Artificial Universidad Complutense de Madrid, Spain
2. : http://ontology.buffalo.edu/smith/articles/ontologies.htm
3. Aamodt, A., Plaza, E.: Case-Based Reasoning: Foundational Issues, Methodological Variations, and System Approaches. (1994)
4. Keirsey, D.: Please Understand Me II: Temperament, Character, Intelligence, 1st Ed.,. Prometheus Nemesis Book Co.