

# Agent Augmented Game Development

Zhiqi Shen

Information Communication Institute  
Nanyang Technological University  
Nanyang Avenue, Singapore 679798  
65-98576794

zqshen@ntu.edu.sg

Chunyan Miao

School of Computer Engineering  
Nanyang Technological University  
Nanyang Avenue, Singapore 679798  
65-67906197

ascymiao@ntu.edu.sg

Yundong Cai

School of Computer Engineering  
Nanyang Technological University  
Nanyang Avenue, Singapore 679798  
65-67906197

caiy0004@ntu.edu.sg

## ABSTRACT

Computer Science students are not only interested in playing games but also highly motivated to learn how games are developed. This paper proposes a novel agent augmented game development framework in a 3D virtualized environment that allows students to become the players, the situated learners and the designers, who create games/stories in an immersive 3D environment, simultaneously.

Unlike most of existing games in which agents are designed to play some roles or to execute specific tasks, we augment the whole game world as an interactive multi-agent system. The agent intractability and autonomy enhance the user interactions and enable the dynamic story creation in a situated learning/playing environment. The proposed approach significantly increases the interests of students in learning to design the game from playing.

An interactive game for science learning in secondary schools is presented to illustrate our approach. The proposed game framework as well as the 3D virtualized game environment has been successfully used in a number of capstone projects in recent two years. Students enjoyed learning and designing games through playing.

## Categories and Subject Descriptors

K.3.1 [Computer Uses in Education]: Collaborative learning – *game-like learning, games, virtual world.*

## General Terms

Design.

## Keywords

Game development, Agent augmentation, Multi-agent systems.

## 1. INTRODUCTION

Computer Science students are not only interested in playing games but also highly motivated to learn how games are developed. However, most of the computer science students think

game development is challenging. Traditionally, students need to learn game development in the class first before they start to practice game development. Therefore, there is a big gap between the learning and real development of the games. In this paper, we present a novel approach that bridges this gap through an agent augmented 3-D virtualized environment for situated game design learning and game development. Each entity in the game world, whether it is a graphic object, an avatar or a user player, could be augmented with an agent. The whole game world will become an interactive multi-agent system. Such environment is supported by our agent augmented game development framework which acts not only as a game development teaching and learning tool but also a real game development environment.

Software agents as autonomous entities have many important characteristics, such as goal oriented, proactive, communicative, and intelligent. Agent augmentation facilitates the interactions between characters, players and dynamic stories. It also increases the involvements of players and interests of players to play and learn in a situated game environment. To enable students to learn game design in a situated environment and to develop games that support interactions and dynamic storytelling, an enabling agent augmented game development framework is critical.

Research efforts on game development with agent technology have been reported [1, 2, 3, 4]. However, in those game development environments, agents either act as specific roles or are designed to execute specific tasks. It is far from the expectation for the dynamic game design/storytelling and the interaction between game and player. In this paper, we present a novel game development framework whereby the story sequences could be decided by the players on the fly. We regard a game as a multi-agent system so that agent interactions and autonomy can be used to increase interactions and support dynamic storytelling/game design. Goal Net [5], a modeling tool, is used for story/game design and agent design. With this framework students can learn not only the conventional game development but also dynamic storytelling.

The paper is organized as follows. Following this section, the related work is studied. In Section 3, the theoretical background of the proposed game development framework is introduced. The agent augmented game development framework is described in Section 4. Section 5 presents a game for science learning in secondary schools developed using this framework by our students to illustrate our approach. Finally, the conclusion and the future work are discussed in Section 6.

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.

Conference '04, Month 1–2, 2004, City, State, Country.  
Copyright 2004 ACM 1-58113-000-0/00/0004...\$5.00.

## 2. RELATED WORK

Using agents in games has improved storytelling significantly. VISTA (Virtual Interactive Story Telling Agents) [1] project introduced an agent based architecture whereby players can interact with agents with questions they want to ask and therefore the players can understand the contents of the story better. SAGE (Storytelling Agent Generation Environment) [2] project constructed storytellers using agents to encourage players' emotional engagement. Mark Riedl, et. al. [3] proposed a narrative mediation to leverage the player involvement and the overall narrative. Marc Cavazza, et. al. [4] used HTN (Hierarchical Task Networks) for agents behavior planning.

In those projects, agents are used to increase interactions between characters in a story and players. On the other hand, techniques are used to plan the agents' behaviors and to control the interactions between agents and players so that the storyline is still maintained.

Unlike most of existing games in which agents are designed to play some roles or to execute specific tasks, we augment the whole game world as an interactive multi-agent system. More specifically, we use Goal Net, a goal oriented model to design both characters and plots in storytelling. Each entity in the game world, whether it is a graphic object, an avatar or a user player, could be augmented with an agent who carries a goal net. The whole game world will become an interactive multi-agent system. More over, we use Fuzzy Cognitive Map (FCM) [6] to model the emotions of characters or agents to simulate the human-like roles/characters in real life. The objective is to create a 3D virtualized immersive environment that allows students to become simultaneously the players, the situated learners and the designers who create games/stories in such an environment.

## 3. THEORITICAL BACKGROUND

### 3.1 Goal Net

Goal Net is a tool for modeling goals of an agent in a multi-agent system. It has been successfully used in the multi-agent system modeling for business forecasting, grid computing and E-learning. Goal Net model is composed of *Goals* and *Transitions*. *Goals*, represented by circles, are used to represent the goals that an agent needs to pursue. *Transitions*, represented by arcs and rectangles or vertical bars, connect from the input goal to the output goal, specifying the relationship between the two goals. Each transition is associated with a task list which defines the possible tasks that the agent needs to perform in order to transit from the input goal to the output goal. A goal net example is shown in Figure 1. There are two kinds of goals in Goal Net, *atomic goals* and *composite goals*. An *atomic goal* is a primitive goal which cannot be further decomposed, while a *composite goal* can be split into goals connected via transitions. Therefore, a complex goal can be recursively decomposed into sub-goals and sub-goal nets. The hierarchical structure simplifies the goal modeling process with different levels of abstraction.

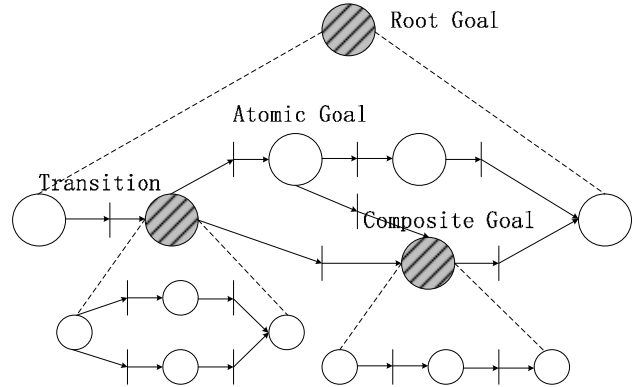


Figure 1. A Goal Net example.

In Goal Net, there are four types of temporal relations of goals represented by *transitions* connected the input goals and output goals: *sequence*, *choice*, *concurrency* and *synchronization*, as shown in Figure 2.

- **Sequence:** A direct sequential causal relationship between input goal  $i$  and output goal  $j$ .
- **Choice:** A selective connection from input goal  $i$  to possible output goals  $j$  and  $k$ , and only one output goal can be selected based on selection criteria.
- **Concurrency:** Input goal  $i$  at completing the tasks, all the output goals  $j$  and  $k$  can be achieved simultaneously.
- **Synchronization:** A synchronization point from different input goals  $i$  and  $j$  to a single output goal  $k$ , and transition to the output goal can only be started when all its input goals are achieved.

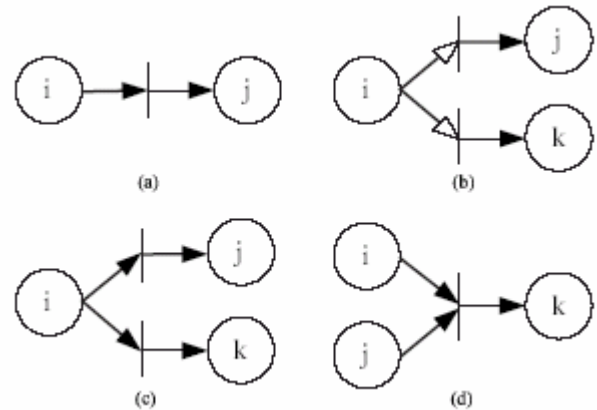


Figure 2. Temporal relationships among goals.

### 3.2 FCM

Fuzzy Cognitive Maps (FCMs) is a kind of causal relationship modelling tool. It provides a simple and straightforward way to model the relationships among different factors. FCMs include two elements: *concepts* and *causal relationships*. As shown in Figure 3, *concepts* are represented by circles, which represent the related causes and effects in the model. The *causal relationships* are represented by directed arcs, each of which has a sign and a weight. The '+' sign means *positive* causal relation, such that the

increase of the starting concept value may cause the increase of the ending concept value. Conversely, the '-' sign means *negative* causal relation, such that the increase of starting concept value will cause the decrease of the ending concept value. The weights differentiate the important levels from the starting concept to certain ending concept. Each concept is represented with a state value whose range is in [0, 1] or [-1, 1], while the causal relation is represented as a weight, whose range is in [-1, 1].

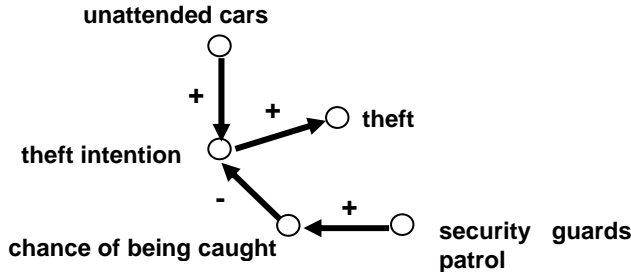


Figure 3. A FCM example.

### 3.3 Dynamic Storytelling

Goal Net is used in the framework to model the goals of an agent – a character in the game and the dynamic plans of the story – plots in the game for interactive storytelling. To model a character as an agent, goals and the temporal relationships of each character are modeled as a goal net. The four kinds of temporal relationships of Goal Net make the characters/agents behave according to the real situation during playing. To model plots of a story, a plot is regarded as a goal. A plot can be split to sub plots. The whole set of plots of a story can be designed as a goal net (plot net). The plots are organized according to their temporal relationships. The real storyline will be generated according to the goal net (plot net) during playing through interactions between the players and the characters.

### 3.4 Interactions

In the framework, FCMs are used as the reasoning tool for goal selection as well as action selection. The interaction from the player, context variables and the possible goals and actions are encapsulated as concepts. The causal relationships among different concepts are determined according to the expert knowledge or predefined rules in the knowledge base. High weights are assigned to those more assertive causal relationships.

Each character make decision for its behavior according to its goal net and FCMs. For example, the time a character *visitor* spent to talk with another character *patient* in a hospital is longer than a threshold, the character *visitor* may become sick. Hence its behavior may also change. The player may interact with the character for resolutions. Then the storyline may be also changed according to the dynamic plan of the goal net (plot net).

## 4. FRAMEWORK

### 4.1 System Architecture

The game system consists of four components as shown in Figure 4: *client application, game server, agent server, 3D object server.*

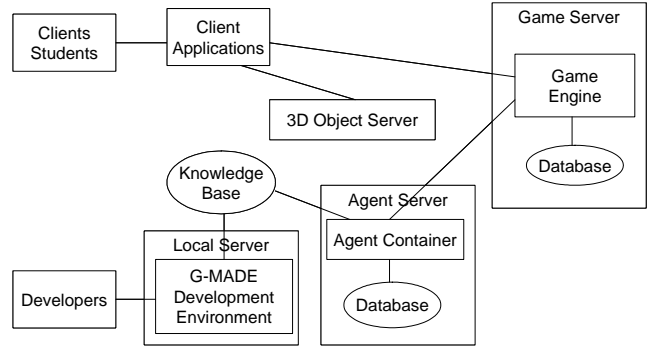


Figure 4. System architecture.

- Client application is user interface to players, where a player plays the game on his/her computer.
- Game server is the engine to run the game programs. It is also the collaborative space to synchronize different players in the game.
- Agent server is the container of the agents - the characters in the game. Each character has a 3D object in the game and an agent on the agent server. The behavior of the character is controlled by the agent.
- Object server is the container of the 3D objects. It contains the 3D objects used for the virtual world of the game. Each character has a 3D object on the object server.

When a game is started, the virtual world of the game will be created at the game server. The 3D objects are loaded to the client application to build the game environment. The characters will be loaded to the client application as well. The agents corresponding to the characters of the game including the avatar that represents the player are created on the agent server. The commands for the behaviors of the characters are controlled by the agents. Only the avatars can be controlled by both the player and the corresponding agents. Usually commands from the avatar agent are notifications or emotional behaviors reflecting the situation changes in the games. So the unreasonable conflict between the commands from the agent and those from the player is minimized. If another player joins the game, the whole set of the game environment will be loaded to his/her client applications.

### 4.2 Development Framework

The development framework consists of 3D object development, agent development, and game development. The game development and 3D object development can be done using commercial products in the market. In the paper, we only focus on the agent development. The agent development, namely G-MADE (Multi-Agent Development Environment for Game), consists of the Goal Net Designer, FCM designer, Agent Creator and Goal Net Loader and SDK (Software Development Kit) of game engine on the game server. Goal Net Designer, Agent Creator and Goal Net Loader are the key components of the MADE (Multi-Agent Development Environment) used for multi-agent system development [7]. G-MADE extends MADE by integration with FCM Designer and SDK of a game engine used in the framework. By this extension, agents can make decision on the actions using FCM inference and manipulate the characters in

the game through APIs (Application Programming Interfaces) of the SDK.

### 4.3 Game Development

In this paper, we regard a game as a multi-agent system. So we map the game development to the agent development using the framework presented in this paper.

#### 4.3.1 Agents Development

Typically, there are four steps developing an agent using G-MADE. They are:

- Function development: Functions are the actions a character may perform. In this step, all the functions will be developed using the SDK of the game engine.
- Goal Net design: Goal Net is the “brain” of an agent. In another words, an agent will pursue the goals according to the temporal relationships among them designed in a goal net. The functions developed at the last step will be used in this step to construct transitions of the goal net so that the agent can control the character represented by this agent.
- FCM design: Within a goal net, the agent needs to make decision to select suitable goal to pursue at the next step or select the next task to perform to pursue a goal according to the current situation. FCM reasoning is used to make decisions for the agent based on the selected context variables.
- Agent creation: With G-MADE, an agent is Goal Net enabled. Every agent created by the Agent Creator has the same structure. It does nothing after it is created. It becomes an active agent only after a goal net is loaded by the Goal Net Loader. So different agents/characters can be identified by different goal nets.

#### 4.3.2 Characters

Each character in the game has at least one goal net associated with. When the character starts to appear in the game, a goal net will be loaded to the corresponding agent to control its behavior. In a game as a multi-agent system, agents/characters need to cooperate or coordinate with each other according to the story. This can be done through goal nets of the plots.

#### 4.3.3 Plots

Plots design is important for storytelling. With Goal Net, a story can be decomposed to many smaller plots. Based on the temporal relationships and dynamics of the story, a goal net (plot net) can be constructed. Each atomic plot is a goal net containing all the goals of characters that should appear in the plot. Each goal represents a goal net of a character. So in this plot, all the goals of characters will be delegated to the corresponding agents to perform.

When a game is started, a special agent, namely director agent or instructor agent will be created who will take the goal net of plots to behave. The whole storytelling or the game will be controlled or directed by this agent. However the real sequence of plots will be generated dynamically according to the interactions with the players.

## 5. GAME FOR SCIENCE LEARNING

A game, namely “Mystery Illness Investigation at Nanyang Town”, was developed using the framework by the capstone project students in the school of computer engineering. The purpose of the game is to teach students in secondary schools about illnesses in a 3-D immersive environment as if in the real life.

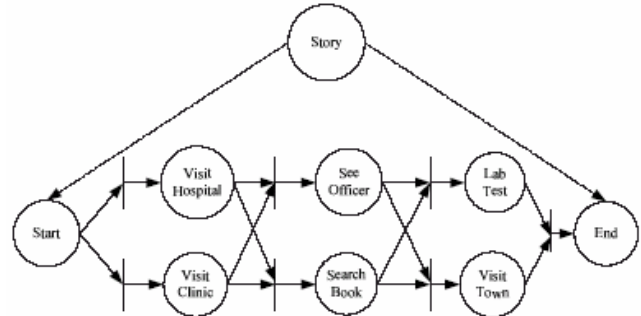


Figure 5. The goal net of the story.

The stories in the game guide players to explore the virtual town and investigate the mystery illness. By talking to different characters at different places in the game, or conducting lab experiments, the players need to find the symptoms of the mystery illness, study the differences among diseases, and conclude the thorough review over the mystery illness by the end of the game. The goal net (plot net) of the story is shown in Figure 5. As shown in the goal net (plot net) the players can go to either the hospital or the clinic to check the symptoms of the mystery illness and how widely the illness is spread. Depending on the availability of the officer in the ministry of health, the player can choose to ask for differences of different illnesses from the officer, or go to library to check them from the books. Moreover, the player can go to the town to verify the conclusion about the mystery illness, or he can do some further laboratory tests. The director agent selects a storyline as shown in Figure 6, in which the player should visit the hospital for illness symptoms, then go to meet officer to query about the differences of the illnesses, lastly go to visit the town to confirm the conclusion. Figure 7 shows a detailed design of plot “visit the hospital”. The story involves three characters: a doctor, a nurse and the avatar representing a player. As shown in Figure 8, the director agent delegates the tasks of visiting different places to the player, and the player is able to interact in the first-person view and third-person view.

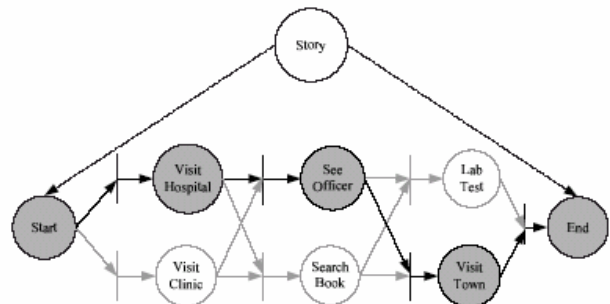


Figure 6. The selected storyline.

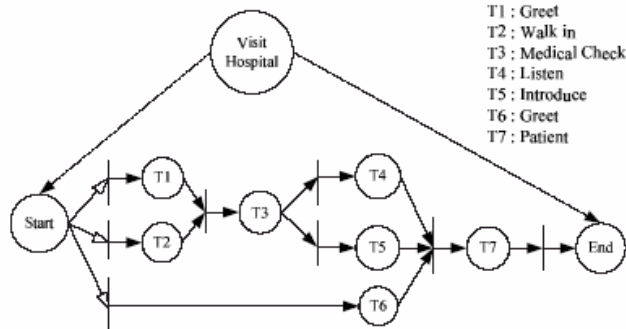


Figure 7. Goal net for visiting the hospital.

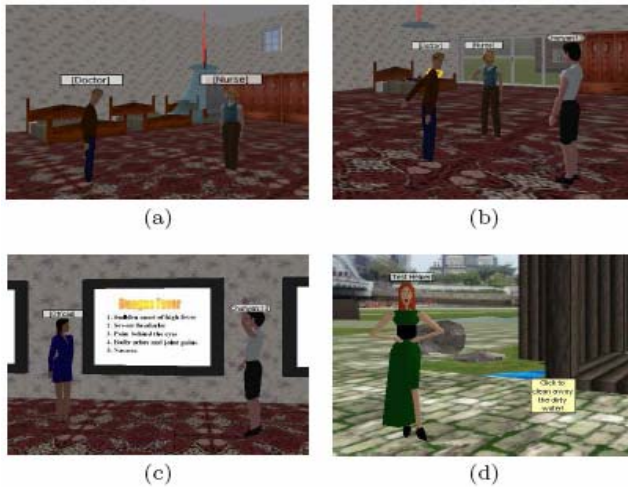


Figure 8. The screen shots of the game.

The whole game development infuses playing, learning, and development into an integrated process. In the beginning, students login as user avatars and start playing in a preliminary simple world. Then the students start to enrich the world or create their own new world with various virtual entities, i.e. graphic objects, avatars. After that students start to design game strategies, story scenes using Goal Net Designer for the virtual entities they created. Finally the students create the dummy agents, load goal nets into the agents and then augment the agents into the virtual entities to make their game world an interactive multi-agent system in a 3D immersive environment.

## 6. CONCLUSION AND FUTURE WORK

This paper proposes a novel game development framework based on Goal Net and FCM. Goal Net is used to design both characters and plots in storytelling so that character-based story design and plots-based story design can be combined and therefore a hybrid game system can be constructed. FCM is used to design the emotions of characters or agents so that interactions between agents and players and story dynamics are increased. The demonstration of game for science learning illustrates the proposed framework is practical and easy to use. With this framework students can learn game development in a situated

learning environment and develop games with intelligence/agent augmentation in a 3-D immersive environment.

The novelty of our approach is that such agent augmented 3D virtualized game development environment allows students to become simultaneously the players, the situated learners and the designers who create games/stories in an immersive 3D environment. It bridges the gap between the learning content of game development and real game development. It also transfers the challenging of learning and development into playing. More over, such environment supports collaborative learning and game design in a virtual team.

Our on-going work focuses on extending our framework with Microsoft XNA game engine. Our future work will focus on the situation awareness and intelligent question answer so that players will experience different story plots with the same game during playing.

## 7. REFERENCES

- [1] Figa, E. and Tarau, P. The VISTA Project: An Agent Architecture for Virtual Interactive Storytelling. *CSCW 2002 Workshop on Storytelling and Collaborative Activities* (New Orleans, Louisiana, USA, November 16, 2002).
- [2] Umaschi, M., and Cassell, J. Storytelling systems: constructing the innerface of the interface. In *Proceedings of the 2nd International Conference on Cognitive Technology (CT '97)* (Aizu, Japan, August 25-28, 1997). IEEE Computer Society Press, 1997, 98-108.
- [3] Fröhlich, B. and Plate, J. The cubic mouse: a new device for three-dimensional input. In *Proceedings of the second international joint conference on Autonomous agents and multiagent systems (AAMAS2003)* (Melbourne, Australia, July 14-18, 2003). ACM Press, New York, NY, 2003, 741-748.
- [4] Cavazza, M., Charles, F., and Mead, S. Agents' Interaction in Virtual Storytelling. In De Antonio, Aylett, and Ballin (eds.), *Intelligent Virtual Agents, Springer Lecture Notes in Artificial Intelligence*, vol. 2190, (2001), 156-170.
- [5] Shen, Z. Q., Miao, C. Y., and Gay, R. Goal Oriented Modeling for Intelligent Software Agents. In *Proceedings of the 2004 IEEE/WIC/ACM International Conference on Intelligent Agent Technology (IAT'04)* (Beijing, China, September 20 - 24, 2004).
- [6] Kosko, B. Fuzzy cognitive maps. *International Journal of Man Machine Studies*, 24, (1986), 66-75.
- [7] Shen, Z. Q., Miao, C. Y., and Gay, R. Goal-oriented Methodology for Agent-oriented Software Engineering. *IEICE Transactions on Information and Systems, Special Issue on Knowledge-based Software Engineering*, Vol. E89-D, No. 4, (Apr. 2006).