

# 객체 상호간의 행위 관계를 지원하는 저작 도구

이동훈<sup>U</sup> 이 건 김상욱  
경북대학교 컴퓨터학과  
{leedh, gal9627, swkim}@cs.knu.ac.kr

## Authoring Tool Supporting Behavior-Relation between Objects

Dong-Hoon Lee<sup>U</sup> Gun Lee Sang-Wook Kim  
Dept. of Computer Science, Kyungpook National University

### 요 약

본 논문에서는 기존의 저작 도구에서 볼 수 있는 시간 중심 저작의 단점을 극복하고자 규칙 기반의 저작을 가능하게 함으로써 객체 상호간의 행위 관계를 지원하는 Genesis 시스템이라는 저작도구를 소개한다. Genesis 시스템은 사용자가 구현하고자 하는 하나의 세계를 만들고 그 세계 속에서 행동하게 될 객체들과 그 객체들간의 행동 관계를 규칙을 통해서 정의할 수 있다. 변수, 모양, 행동 등이 정의된 하나의 객체는 시간의 흐름, 사용자의 이벤트 등에 의해서 자신에게 정의된 위치 검사와 변수 검사와 등의 규칙을 점검한 뒤 그 규칙을 만족을 하게 되면 정의된 행동을 수행한다. 이러한 객체에 대한 규칙의 정의, 객체가 존재할 공간에 대한 환경 설정 등을 통해서 사용자는 자신이 원하는 가상 세계, 시뮬레이션, 게임 등을 제작할 수 있다.

### 1. 서론

현재 일반인들이 널리 이용하고 있는 멀티미디어 저작도구는 마이크로소프트사<sup>[1]</sup>의 파워포인트나 매크로미디어<sup>[2]</sup>의 사의 디렉터, 플래시 등이 있다. 이러한 저작도구들로 사용자는 일반 텍스트나 이미지 이외에도 움직이는 애니메이션, 음악효과 등을 통하여 생동감 있는 저작물을 생성해 낼 수 있다. 하지만 이러한 기존의 저작도구에서는 사용자의 입장에서 생각해 볼 수 있는 한계점이 있다. 파워포인트와 같은 저작도구의 경우에는 글, 그림, 소리 등의 효과를 줄 수는 있지만 애니메이션 되는 내용이 사용자가 제시하고자 하는 프리젠테이션 내용과는 무관한 효과를 줄 수밖에 없다. 디렉터와 플래시의 경우에는 사용자가 고안한 애니메이션의 효과를 줄 수는 있지만 시간 중심에 기반한 저작도구이기 때문에 특정한 시간에 지정한 특정한 행동을 할 수밖에 없는 제한된 애니메이션으로 시뮬레이션과 같은 효과는 구현 할 수 없다.

본 논문에서는 이러한 기존의 저작도구의 한계점을 극복할 수 있는 새로운 개념의 저작도구인 Genesis 시스템을 소개한다. Genesis 시스템에서 사용자는 직접 어떠한 한 객체에 대하여 다양한 규칙(규칙)을 설정할 수 있다. 객체는 특정한 시간에 자신에게 설정된 다양한 규칙 가운데 조건을 만족하는 모든 행동을 할 수 있으므로 해서 객체 상호간의 행위 관계를

형성 할 수 있으며 이는 곧 시뮬레이션 효과를 가능하도록 지원하게 된다.

제 2 절에서는 Genesis 시스템의 구조 및 개요를 소개하고, 제 3 절에서는 Genesis 시스템의 내부 구현 원리를, 제 4 절에서는 구현 예를 보이고 제 5 절에서 향후 결론을 맺고자 한다.

### 2. Genesis 시스템의 구조 및 개요

시스템은 크게 객체 생성기, 물 생성기, 세계 및 환경 생성기, Genesis 파일 생성기, 애니메이션 프리젠테이션 툴, 이렇게 5개의 부분으로 나누어진다.

#### 2.1 객체 생성기

사용자는 객체 생성기를 통하여 자신이 원하는 객체를 생성할 수 있다. 사용자가 생성하는 객체에는 그 객체가 가지고 있어야 할 변수와 모양을 등록할 수 있고 어떤 규칙이 만족할 때 그 객체가 취할 행동을 정의 할 수 있다. 또한 상속의 개념을 도입해서 상위 객체의 특성을 상속받을 수 있는 기능을 제공한다.

#### 2.2 규칙 생성기

하나의 객체가 디자인되면 그 객체가 어떤 행동을 하기 위해서 검사해야 할 규칙을 규칙 생성기를 통해서 정의할 수

있다. 규칙은 자신 내부적으로 검사해야할 규칙, 다른 객체와의 관계에서 검사해야할 규칙, 혹은 자신을 둘러싸고 있는 환경과 관계에서 검사해야할 규칙 등을 정의할 수 있다. 하나의 객체에 대해서 규칙을 체크하는 경우는 시간에 따라서 주기적으로 수행을 하는 경우와 사용자와의 상호 작용을 지원하기 위해서 설정된 키보드나 마우스의 입력이 있을 때 수행할 수 있도록 구현하였다.

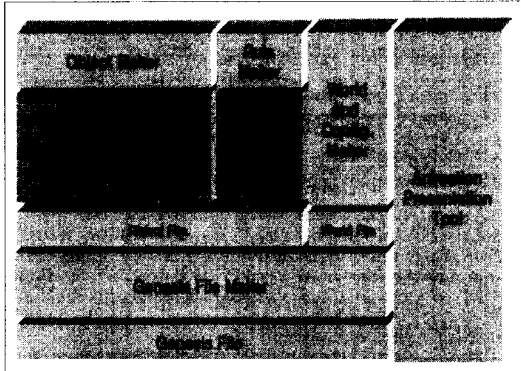


그림 1. Genesis 시스템 전체 구조도

### 2.3 세계 및 환경 생성기

위와 같이 정의된 객체가 활동을 하기 위한 환경인 세계(world)를 정의하고 기타 환경 설정을 하기 위한 환경설정 생성기가 있다. 세계에는 그 곳에서 활동할 객체들을 등록할 수 있는 기능을 제공하고 객체들과 세계와의 규칙 생성을 위한 세계 자체의 변수와 보드 등을 등록할 수 있다. 또한 네트워크 지원을 위한 사용자 등록기능, 제어판 설정 기능 등을 제공한다.

### 2.4 Genesis 파일 생성기 및 애니메이션 프리젠테이션 툴

이렇게 설정된 객체와 세계 및 환경 설정에 대한 정보는 각각 객체 파일과 세계 파일로 나뉘어 저장되며 이 파일은 Genesis 파일 생성기를 통하여 Genesis 파일을 생성해 내고, 이 Genesis 파일은 애니메이션 프리젠테이션 툴의 입력으로 들어가서 원하는 애니메이션을 만들어 낼 수 있게 된다.

## 3. Genesis 시스템 내부 구현 원리

### 3.1 객체

객체는 내부적으로 변수(variable), 모양(appearance), 소리(sound), 행동(behavior)에 관한 정보를 가지고 있다. 변수는 객체의 특성을 나타낼 수 있는 값을 말한다. 예를 들어 "나무"라는 객체를 디자인을 할 경우 필요한 나이, 에너지 등은 변수로 정의할 수 있다. 모양은 하나의 객체를 시각적으로 표현하기 위한 이미지를 의미한다. 하나의 나무가 성장함을 표현하려면 나이라는 변수도 필요하지만 그 나이에 따라서 변화하는 나무의 이미지도 필요하다. 소리는 객체가 표현할 수 있

는 소리를 의미한다. 동물이면 우는 소리, 먹는 소리 등을 지정할 수 있다.

행동은 객체의 움직임을 나타낸다. 다음절에서 제시될 규칙이 만족하게 되면 객체는 규칙에서 지정한 행동을 하게 된다. 행동은 크게 시간에 따라 실행되는 행동과 사용자의 입력에 따라서 실행하는 행동으로 크게 2가지로 분류할 수 있다. 시간에 따라 시행하는 행동은 다시 시간에 따라 항상 수행하는 행동, 시간이 되면 임의의 하나를 수행하는 행동, 시간이 되면 우선 순위에 따라서 수행하는 행동으로 분류된다. 사용자의 입력에 따라서 수행하는 행동은 마우스 눌림, 마우스 떨어짐, 키 눌림, 키 떨어짐 이렇게 4가지 상호 작용에 의해서 수행된다. 사용자의 입력에 의해서 수행되는 행동은 사용자가 지정한 객체에 의해서만 반응할 수 있다.

객체가 실제로 수행하게 되는 행동은 특정위치에 새로운 객체를 생성하는 "만들기" 행동과 기존의 객체를 없애는 "지우기" 행동, 그리고 세계나 다른 객체의 변수 값을 변경하는 "바꾸기" 행동 이렇게 3가지의 행동을 수행할 수 있다.

### 3.2 규칙

객체에 정의할 수 있는 규칙은 특정한 위치에 특정한 객체가 있는지를 검사하는 위치체크와 특정한 위치 상에 존재하는 객체나 세계의 변수를 검사하는 변수체크 이렇게 두 가지가 있다. 위치체크는 현재 객체가 존재하는 좌표를 중심좌표(0,0)로 해서 상대적으로 지정된 사각형 좌표(left, top, right, bottom) 영역 안에 지정된 다른 객체가 존재하는 지 혹은 존재하지 않는지를 검사하는 것이다. 또한 변수 체크는 영역 안에서 지정된 다른 객체의 변수와 환경 변수의 값을 일반적인 비교연산자(=, >, <, !=, >=, <=)를 이용하여 비교할 수 있도록 하였다.

### 3.3 세계 및 환경설정

세계 또한 일종의 객체 개념으로 변수나 소리 등과 같은 특성을 가진다. 또한 객체의 모양처럼 보드라는 개념을 가지고 있어서 규칙에 따라서 배경을 변화할 수 있다. 세계가 가지고 있는 이러한 특성은 어떤 객체의 규칙을 정의할 때 설정할 수 있도록 되어 있다. 기본적으로 세계는 해상도를 나타내기 위한 Width, Height 값을 가지고 현재 사용자를 표시하기 위한 값(Current\_Player), 현재 보드를 나타내기 위한 값(Current\_Board), 현재 마우스의 위치를 나타내기 위한 값을(Current\_MouseX, Current\_MouseY)를 가지게 된다. 또한 이 세계상에 등록된 여러 객체의 ID값도 가지고 있다.

환경설정은 사용자에게 좀 더 나은 서비스를 제공하기 위한 것이다. 사용자가 각 객체에 대한 제어를 좀 더 쉽게 하기 위해서 Control Layer 개념을 도입하여 그 레이어 상에 고정 문자열, 고정 이미지, 변수의 상태 값을 확인하기 위한 Watch, 특정키의 입력이 특정한 이벤트를 발생하기 위한 KeyButton, 네트워크상의 채팅을 지원하기 위한 Chatline, 그리고 현재

World상의 객체의 위치를 표시하기 위한 미니 맵 등의 기능을 제공한다.

3.4 리소스

Genesis 시스템 상의 모든 객체뿐만 아니라 그 객체와 세계상의 모양을 위한 이미지 파일과 소리를 위한 웨브 파일 등은 모두 리소스의 개념으로 전체적으로 하나로 관리가 된다. 이들은 같은 형태를 가지는 리스트에 순차적으로 등록이 되며 등록된 리소스는 그 인덱스 값으로 다른 곳에서 참조를 가능하게 한다.

4. 시스템 구현 예

이번 장에서는 사용자가 탱크 게임을 제작하고자 할 경우 시스템이 구현하게 되는 예를 보인다. 먼저 사용자는 탱크 게임이 진행될 세계와 그 세계 안에 탱크 객체 및 세계에서 필요한 환경 변수 등을 설정하게 된다.

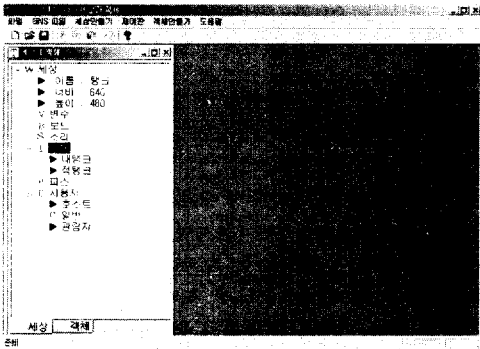


그림 2. 세계를 생성, 객체 등록

다음으로 탱크에 필요한 폭탄, 에너지 등의 변수를 설정하고 탱크가 가지게 모양과 그 모양에 대한 속성 등을 할당한다.

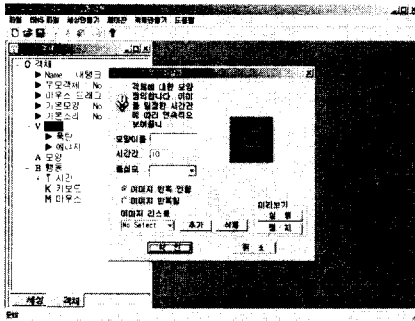


그림 3. 모양을 등록

이렇게 생성한 탱크라는 객체에 대해서 사용자는 규칙을 정할 수 있다. 마우스가 클릭된 지점으로 탱크의 위치를 이동시키고 스페이스를 누르면 폭탄이 발사된다. 다른 탱크의 폭탄이 자신의 일정영역 안으로 들어오게 되면 그 폭탄을 사라지게 하고 자신의 에너지를 감소시킨다. 이때 자신의 에너지가

0 이가 되면 자신도 사라지게 된다. 이와 같은 룰을 사용자 는 제공되는 인터페이스를 통하여 설정할 수 있게 된다.

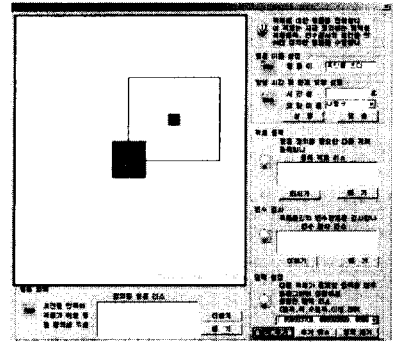


그림 4. 규칙 설정

이렇게 규칙을 생성하여 Gns 파일 만들기를 선택을 하면 사용자가 제작한 여러 객체와 규칙들은 하나의 프리젠테이션 파일로 만들어지게 된다. 사용자는 아래와 같은 실행화면을 통해서 자신이 제작한 탱크 게임을 즐길 수 있다.

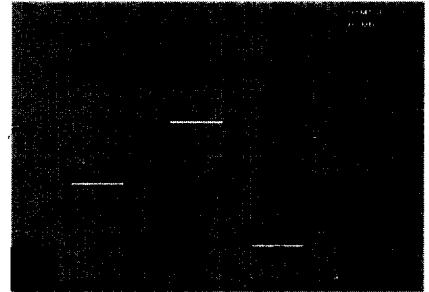


그림 5. 실행화면

5. 결론

본 논문에서는 기존의 시간 중심의 저작 도구에서 볼 수 있는 한계성을 극복할 수 있는 규칙 기반의 저작도구인 Genesis 시스템을 소개하였다. 사용자는 이 시스템을 통하여 생태계나 자연 현상과 같은 것을 가상적으로 구현해 볼 수 있고, 뉴턴의 역학계와 같은 과학 현상을 직접 검증해 볼 수도 있다. 또한 사용자 이벤트를 강화하여 네트워크를 제공하는 게임이나 가상 수업에 적용할 수 있는 교제도 만들어 낼 수 있다.

앞으로 이 시스템을 웹 기반으로 개발하여 전세계 인터넷 상에서 구현이 되었다면 훨씬 더 많은 적용분야가 있으리라 기대된다.

참고문헌

[1] <http://www.microsoft.com/>  
 [2] <http://www.macromedia.com/>  
 [3] Apple Cocoa DR Team "Cocoa Tutorial" Apple Computer, Inc., 1997  
 [4] M. Jourdan, N.Layaida, C.Roisin, L.Sabry-Ismail, and L.Tardif, "Madeus, an Authoring Environment for Interactive Multimedia Documents," Proceedings of ACM MULTIMEDIA '98, September 12 16, 1998