

# 액션 스크립트 기반의 소셜 네트워크 게임엔진의 개발

## Development of Social Network Game Engine based on ActionScript

우 종 우\*  
Chong-Woo Woo

김 대 령\*\*  
Daeryung Kim

### 요 약

페이스북, 싸이월드 등의 소셜 네트워크 서비스(Social Network Service: SNS)가 발전하면서, 이를 기반으로 하는 소셜 네트워크 게임, 소셜 커머스 등의 서비스도 활성화되고 있다. 특히, 소셜 네트워크 게임(Social Network Game: NG)은 기존 온라인 게임보다는 규모가 가볍고, 친구들과리 쉽게 즐길 수 있다는 점에서 사람들에게 많은 호응을 얻고 있다. 이러한 소셜 네트워크 게임 시장 규모는 해가 지날수록 점점 커지고 있지만 게임 개발 측면에서는 아직 미흡한 부분이 많다. 특히, 현재 나온 대부분의 개발엔진들이 온라인 게임이나 콘솔 게임에 초점이 맞춰져 있고, 소셜 네트워크 게임의 특징에 적합한 전용 엔진이 거의 없는 상황이라서 소셜 네트워크 게임을 개발하는데 많은 시간이 소요된다. 본 논문에서 제시한 엔진은 기존의 게임 엔진이 가지는 요소 및 특징뿐만 아니라 소셜 네트워크 게임의 특징을 고려하여 소셜 네트워크 게임 개발에 최적화하여 설계하였다. 또한 소셜 게임 장르 중 가장 많은 비율을 차지하는 시뮬레이션 게임을 구현하기 위해 필요한 맵과 캐릭터 애니메이션을 쉽게 구현할 수 있는 모듈과 툴을 제공한다. 본 논문에서 구현된 엔진은 이미지, 텍스트 및 캐릭터 출력 속도를 기준으로 성능을 평가하였고, 실험 결과 이미지, 텍스트 및 캐릭터 모두 게임을 구현하기에 충분히 빠른 출력 속도를 보여주었다.

### ABSTRACT

As the social networking service (SNS), Facebook, and Cyworld, is developing, the social network game and social business commerce based on this service is activated. Especially, the Social Network Game (SNG) is getting explosive interests and it becomes popular, because it is small scale and user can enjoy the game among close friends. The market for this game is getting larger every year, but still it has some limitations in developing the game. Especially, the current game engine is aiming for developing online or console game, and there is no exclusive game engine for developing SNG. Therefore, it takes lots of time for developing SNG with this game engine. In this paper, we described a design and development of the game engine optimized for developing SNG, which not only adapts the main characteristics of the previous game engine, but also considers the specific characteristics of the SNG. The engine also supports map for the simulation game that is the most popular game in SNG, and also provides modules and tools for developing character animation easily. The evaluation standard for the performance of the game engine is the output generation speed of image, text and character. And the results showed reasonable output speed for developing the SNG in generation of image, text, and character.

☞ keyword : Social Network, Social Network Game, Game Engine, Authoring Tool.

## 1. 서 론

개인 홈페이지에서 블로그, 그리고 트위터와 같은 마이크로 블로그까지 우리는 인터넷 세상에서 개인을 표현하는 공간을 만들고 그곳으로 자신의 인맥을 초대하고 관계를 유지하는 것에 이미 익숙해져 있다. 사람과의 관

계는 이제 더 이상 오프라인에만 한정되지 않으며 기존의 인맥이나 관계를 지원하던 온라인의 소셜 네트워크 서비스(SNS)는 이제 현대인을 연결하는 가장 창의적이고도 활발한 소통의 수단으로 활용되고 있다. SNS의 발전은 소셜 네트워크를 기반으로 하는 게임, 커머스 등의 확장된 서비스를 형성하였다. 그 중 소셜 네트워크 게임(SNG)은 기존 온라인 게임보다는 게임의 규모가 가벼우면서 친구들과리 게임을 즐길 수 있다는 점에서 사람들에게 많은 호응과 주목을 얻고 있다 [1][2][3].

처음 SNG가 주목을 받기 시작한 것은 미국의 소셜 네트워크서비스인 페이스북에 연동된 농장 타이쿤 게임인 팜빌[4]이 크게 성공하면서 부터이다. 이미 전 세계적으로 5,000만명이 넘는 페이스북 이용자가 팜빌을 즐기고

\* 종신회원 : 국민대학교 컴퓨터공학부 교수  
cwwoo@kookmin.ac.kr(교신저자)

\*\* 정회원 : (주) Linyworks 연구원  
drforgod@gmail.com (주저자)

[2011/07/04 투고 - 2011/07/11 심사 - 2011/11/17 심사완료]

☆ 본 논문은 국민대학교 2011년도 교내연구비 지원을 받아 수행되었음.

있다. 우리 나라의 중소기업사들 역시 팜빌의 성공을 모델로 SNG 개발에 집중하기 시작했고 팜빌과 비슷한 형식의 소셜 네트워크 기반 농장게임을 비롯하여 다양한 형식의 소셜 게임을 선보이고 있다 [5][6].

이러한 SNG 시장은 최근 급속히 발전하고 있고, 시장이 점점 커지고 있지만 게임 개발 환경 측면에서는 아직 초기 단계라고 할 수 있다. 소셜 게임의 특징에 맞게 구현된 엔진이 없는 상황이고, 있다고 하더라도 공개된 버전이 없어 게임 개발의 효율성이 많이 떨어진다. [7][8]

본 논문에서는 SNS의 특성을 고려한 소셜 게임 개발 엔진을 설계하고 구현하고자 한다. 현재 대부분의 소셜 게임들이 플래시로 제작이 되고 있기 때문에 본 논문에서 구현한 엔진도 액션 스크립트 기반으로 작성하였으며 이미지 및 텍스트 출력 성능 실험을 통해 엔진을 평가하고자 한다.

## 2. 관련연구

### 2.1 SNS기반의 SNG

SNG는 SNS와 같은 맥락으로 볼 수 있지만, 형식이 서비스가 아닌 게임이라는 것이다. 즉, 게임도 하나의 서비스를 감안할 때 SNG는 SNS에 콘텐츠적 형태를 가진 것이라 할 수 있으며, 일반적인 온라인 게임처럼 모르는 사람들과의 관계가 아니라, 이미 형성된 지인들과의 관계로 게임이 이루어지게 된다. 이러한 SNG의 특성은 게임이 가지는 재미와 사회적 커뮤니케이션의 결합이라고 할 수 있다 [9][10]. 자신의 홈페이지에서 소셜 라이징 (socializing)을 목적으로 하는 SNG의 특성은 다음 표1과 같다.

이러한 특성을 요약하면, SNG는 난이도가 쉽고, 다른 사용자들과의 커뮤니케이션의 재미와 유용성을 제공하며, 게임 내 사용자 네트워킹이 현실 사용자를 재 생산하여 발전하는 게임 서비스로 정의할 수 있다. SNG는 기존 온라인 게임에서도 경험할 수 있는 “이야기성” 이 다른 사용자들과의 커뮤니케이션을 통해 다양화되고 확장되어, 더욱 자유롭고 자발적인 게임플레이를 유도할 수 있다[11].

이러한 특성들과 게임엔진과의 연관성은 다음과 같이 구분된다. 예를 들면, 자발성은 연결 지향적인 게임보다는 웹과 같이 비 연결 지향 기능을 제공 함으로서, 사용자가 원하는 시간에 접속 할 수 있어야 한다. 사교성과 이야기성은 게임도중에 접속된 친구의 리스트 및 친구와

(표 1) SNG의 특성

특 성	내 용
자발성 (Spontaneity)	- 연결감(Always on): 사용자가 순간적으로 접속하고 싶을 때 자유롭게 접속 한다. - 원 클릭 플레이(One-Click Play): 여러 키들이 조합이 아닌 한 번의 클릭으로 게임을 플레이 한다.
상징적 현실성 (symbolic-physicality)	- 게임내 보상/아이템을 다른 사람과 나누거나, 선물로 줄 수 있다. - 현실적인 게임 표현: 감정 표현을 포함하는 현실적 행동들을 상징적으로 전달한다. - 친구/커뮤니티의 유용성: 게임 내 친구/커뮤니티를 형성이 유용 하다는 인식을 가질 수 있게 한다.
사교성 (sociability)	- 프로필의 정보 및 기타 게임 내의 정보들이 플레이 또는 네트워킹을 통해 추가적인 가치를 얻게 된다. - 그룹 플레이의 기반은 온라인/오프라인에서의 연관성을 바탕으로 한다.
이야기성 (narrativity)	- 다른 플레이어들의 점수, 도전 성취 등을 보여 줄 수 있다. - 게임 내 행동들이 어떠한 이야기화 될 수 있다. - 게임 내 일부 데이터를 통해 사용자들이 추가적인 가치를 가지는 부가데이터로 재생산이 가능하다.

의 상호작용에 대한 기능이며, 상징적 현실성은 게임이 실생활과 밀접한 내용을 제공 한다는 의미이다. 따라서 이러한 SNG의 요소들을 이해해서 게임엔진 설계의 중요 요소로 반영하여야 한다.

### 2.2 대표적인 SNG게임

온라인 게임의 대표적인 MMORPG(Massively Multiplayer Online Role-Playing Game) 사용자들은 채팅을 통하여 친구를 만들어 오프라인의 인맥관계를 맺기도 하지만, 기본적으로 이러한 기능을 퍼포먼스의 보조수단으로 이용하고 있는 반면에, SNG의 경우에는 일차적인 목표가 소셜 라이징에 있다고 볼 수 있다. 즉 MMORPG는 퍼포먼스를 위해 소셜라이징을 하고, SNG는 소셜라이징을 위해 퍼포먼스를 이용한다고 말할 수 있다.

이러한 SNG의 대표적인 게임으로는 ‘팜빌’과 ‘레스토랑시티’가 있는데, 팜빌은 페이스북 기반의 SNG게임이며, 페이스북 이용자끼리 협동을 하면서 농장을 경영하는 시뮬레이션 게임으로 페이스북의 이용자가 5억명을

넘으면서 팜빌과 같은 소셜 게임의 인기도 함께 상승하고 있다. 팜빌은 액션 스크립트 기반으로 개발 되었고, SNG의 특성에 맞게 게임하단의 UI를 통하여 이용자들의 목록을 보여주며, 다양한 이벤트를 통해 상호작용을 할 수 있도록 구성되어 있다.

레스토랑시티 역시 페이스북 기반의 경영 시뮬레이션 게임으로, SNS라는 플랫폼의 특성을 극대화 한 게임이다 [12]. 자신의 페이스북 친구들을 고용해 다양한 임무를 수행킬수 있기 때문에, 이웃이 많을수록 게임이 보다 원활히 진행되며, 자연스럽게 경쟁과 흥미를 극대화 할 수 있다. 레스토랑시티 역시 팜빌과 같은 UI를 제공하며 식당이라는 실생활과 밀접한 요소를 게임에 추가하였다.

### 2.3 액션 스크립트 기반의 게임엔진

현재 개발된 액션 스크립트 기반의 게임엔진에는 다음과 같은 엔진들이 있으며 특징은 다음과 같다[13][14].

#### (1) PushButton 엔진

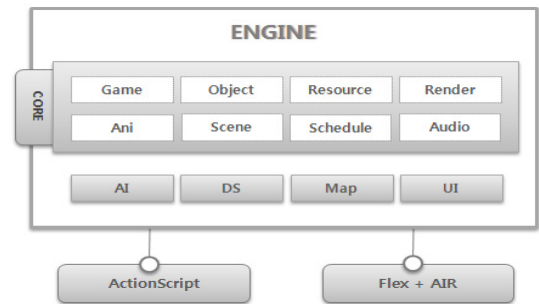
PushButton 엔진은 ActionScript 기반의 오픈 소스 Flash 게임 프레임워크이다. 이 엔진은 기본적으로 리소스 관리, 로그 시스템, 디버그 모니터링, 직렬화, 시간 관리 등을 제공하여 게임 개발을 쉽게 해주며, 컴포넌트 기반의 게임 구조를 갖고 있다. 컴포넌트들은, "Box2D 기반의 물리 컴포넌트, 2D 렌더링 기반의 스프라이트 컴포넌트, 타일맵 시스템 컴포넌트, 길찾기 컴포넌트, 그리고 네트워크 처리와 관련된 컴포넌트" 등으로 구성되어 있다.

#### (2) Flixel 엔진

Flixel 엔진은 Adam Atomic에 의해 만들어진 2D Flash ActionScript 라이브러리이며, 2D 비트맵을 이용한 2D 게임 개발에 적합하다[8]. 래스터방식으로, 벡터 방식에 비해 속도 면에서는 장점이 있지만, 렌더링 기능 면에서는 많은 제약이 생긴다. 제공되는 기능은 "애니메이션, 충돌 처리, 타일 맵, 파티클 효과, 기본 물리 처리" 등이다.

#### (3) FlashPunk 엔진

FlashPunk 엔진은 2D Flash 게임 개발을 위한 ActionScript 라이브러리이며, 엔진의 특징은 "충돌 처리 시스템, 애니메이션과 텍스트 등을 위한 헬퍼 클래스, 실시간 디버깅과 정보 트래킹을 위한 디버그 콘솔, 사운드 이펙트, 모션 트위닝, 파티클 이펙트" 등의 기능을 제공한다.



(그림 1) 게임엔진의 전체 구조도

## 3. 게임엔진 설계

본 연구에서 설계한 엔진은 다음과 같은 주안점으로 설계하였다. 첫째, SNG의 특성을 파악하여 SNG개발을 위한 효율적이고 최적화 된 기능을 제공한다. 둘째, MVC (Model-View-Control)모델과 Scene구조를 지원하여 효율성과 확장성을 높인다. 셋째, Flex를 활용하여 GUI제작 및 게임개발 속도를 향상시킨다. 넷째, 실시간으로 처리하기 힘들거나 부하가 많은 작업에 대해 선 처리가 가능하게 한다.

### 3.1 게임 엔진의 구조

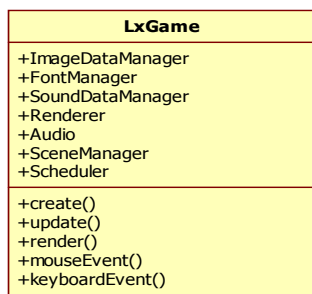
다양한 장르의 게임을 완벽히 지원하는 엔진을 구현하기는 상당히 어렵기 때문에, 본 연구에서는 SNG개발에 공통으로 필요한 부분을 기본 패키지로 제공하고, 장르에 따라 필요한 모듈들을 선택적으로 사용할 수 있도록 하였다. 본 연구의 게임엔진은 다음 (그림 1)에서처럼 엔진의 핵심적 기능을 하는 Game, Object, Resource, Render, Ani, Scene, Schedule, Audio등의 모듈들이Core 패키지를 구성하고 있으며, AI, DS, MAP, UI 등의 추가 모듈들과 함께 엔진을 구성하고 있다.

### 3.2 세부 모듈의 기능

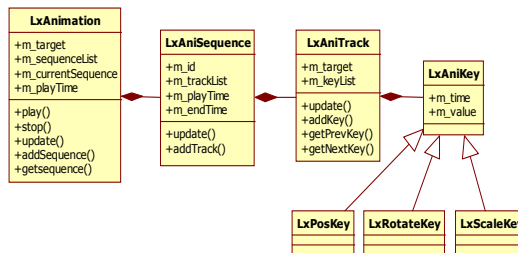
게임엔진의 각 부분별 모듈의 세부적인 기능 및 구조는 다음과 같다.

#### (1) Game 모듈

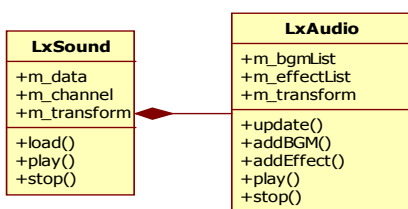
엔진을 대표하는 모듈로 다른 모듈들을 포함하거나 사용하여 원하는 형태로 게임을 실행하고 관리할 수 있



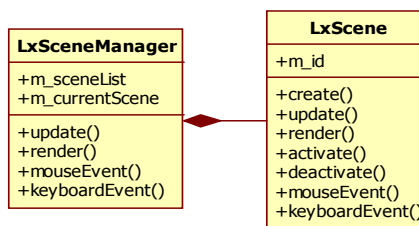
(그림 2) Game 모듈의 클래스 설계



(그림 4) Animation 모듈의 기본 설계



(그림 3) Audio 모듈의 기본설계



(그림 5) Scene모듈의 기본 설계

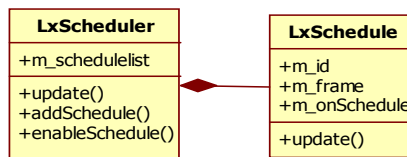
게 해주며, 시스템 이벤트들을 처리한다. 관련 클래스의 기본 설계는 다음 (그림 2)와 같다.

(2) Audio 모듈

Audio 모듈은 사운드 파일을 로드하고, 출력하는 기능을 하는 모듈이다. ActionScript에서는 기본적으로 MP3 파일만 지원된다. 본 엔진에서는 MP3파일을 LxSoundData 클래스를 통해 관리하고 LxSound 클래스를 통해 사운드의 재생, 볼륨 조절 등을 관리한다. LxAudio 클래스는 LxSound객체들을 배경음과 효과음으로 구분하여 관리하게 된다.

(3) Ani 모듈

Ani 모듈은 일정 시간 동안 렌더링 오브젝트에 Matrix Transform이나 Color Transform을 적용하여 원하는 모습으로 변형하여 애니메이션 시키기 위해 사용된다. LxAniKey는 특정 시간에 적용될 애니메이션 키 값을 나타낸다. LxAniTrack은 여러 개의 연결된 LxAniKey들로 이루어지며, 특정 오브젝트에 대해 일정 시간 동안 적용되는 애니메이션 키들의 모음이다. LxAniSequence는 여러 개의 LxAniTrack들로 이루어지며, 하나의 애니메이션 동작을



(그림 6) Schedule 모듈의 기본 설계

나타낸다. LxAnimation은 여러 개의 LxAni Sequence들로 이루어지며, 특정 게임 오브젝트의 애니메이션 모음을 의미한다. 본 엔진에서는 위치, 회전, 확대, 행렬 변환, 등의 애니메이션을 제공한다.

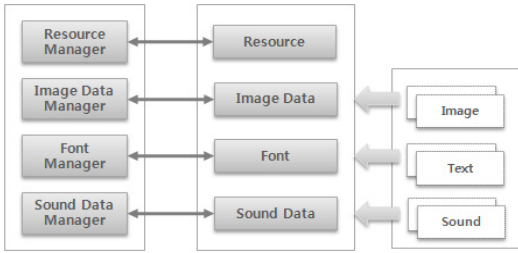
(4) Scene 모듈

Scene 모듈은 특정 기능을 하는 화면 단위로 게임 로직을 구분하여 효율적으로 게임 로직을 관리하기 위함과 Flex를 이용한 쉽고 편리한 GUI 작업을 하기 위해 만들어진 모듈이다.

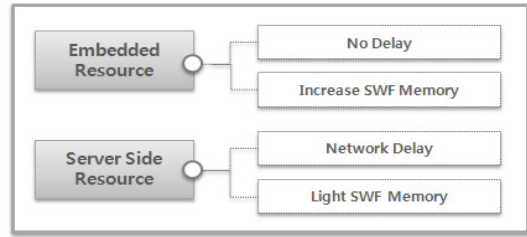
LxScene 클래스가 하나의 Scene을 나타내며 LxSceneManager에 의해 관리된다.

(5) Schedule 모듈

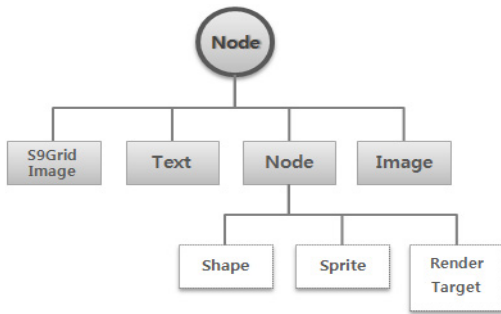
Schedule 모듈은 게임 로직의 업데이트 부분과 화면



(그림 7) 리소스 관리 구조



(그림 9) 엔진의 리소스 처리방식



(그림 8) 렌더링 모듈의 구조

출력 부분 등을 구분하여, 일정 간격으로 정해진 스케줄의 원활한 처리를 지원하는 모듈이다. LxSchedule 클래스를 생성하여 Schedule을 만들 수 있으며, LxScheduler에 등록하면, 정해진 스케줄대로 이벤트를 보내준다.

#### (6) Resource 모듈

본 엔진에서 사용되는 리소스에는 이미지, 폰트, 및 사운드가 있다. 리소스는 다음 (그림 7)과 같이 각각의 리소스 매니저에 의해 관리 되고 그 리소스를 이미지, 텍스트, 사운드 클래스에서 가져가 사용하는 구조로 되어있다. 이러한 리소스 관리 구조는 각각의 리소스마다 가지는 특징들을 효율적으로 처리하게 해주며, 리소스의 중복을 최소화 한다.

#### (7) Render모듈

이미지, 텍스트, 도형등의 시각적 객체들을 화면에 출력해주는 역할을 한다. 본 엔진에서는 트리 계층구조를 렌더링구조로 사용하여, 이러한 시각적 객체들을 효율적으로 처리할 수 있다.

### 3.3 SNG 엔진의 특성 및 타 엔진과의 차별성

#### (1) SNG에 최적화된 리소스 사용

리소스는 크게 다음 두 가지 방식으로 입력이 가능하다. 첫 번째는 리소스를 플래시 파일에 포함시키는 방식이고, 다른 하나는 네트워크를 통해 서버에 있는 리소스를 다운받는 방식이다. 리소스를 포함시킨 경우는 리소스가 로딩되는 즉시 사용 가능하지만 원래 실행 파일로써 필요한 용량 외에 리소스의 용량까지 포함되어 실행 파일의 크기가 커진다는 단점이 있다. 둘째, 서버를 통해 리소스를 다운받는 방식은 실행 파일의 크기는 작아지나 네트워크 통신에 의한 입력 지연이 발생한다. 이 방식은 전자와는 달리, 업데이트가 되어야 하거나 상황에 따라서 사용여부가 결정되는 리소스에 적합하다.

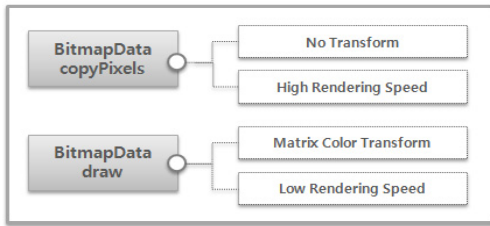
Actionscript 기반의 네트워크 게임을 만들게 되면, 이러한 두 가지 방식의 리소스가 모두 필요하게 되며, 따라서 본 연구의 엔진에서는 이 두 가지 방식을 모두 지원하여 엔진의 효율성을 높였다.

#### (2) Flex를 이용한 간편하고 효율적인 GUI구성

ActionScript는 게임을 만들기에 효율적인 언어이지만, GUI를 구성하기에는 부족한 점이 많다. 따라서 본 엔진에서는 ActionScript를 이용한 게임 화면의 렌더링과 Flex를 이용한 GUI 구성을 함께 사용할 수 있는 엔진 구조를 지원한다. 즉, 엔진에 포함된 Scene 모듈을 사용함으로써, 개발 과정에서 GUI 적용 모습을 확인하며 작업이 가능하기 때문에 간편하고 효율적인 GUI 구성과 게임 개발이 가능하다.

#### (3) 효율적이고 확장성 높은 렌더링 구조

ActionScript에서는 크게 두 가지 방식으로 렌더링이



(그림 10) 렌더링 방식



(그림 11) 캐릭터 파츠의 구성

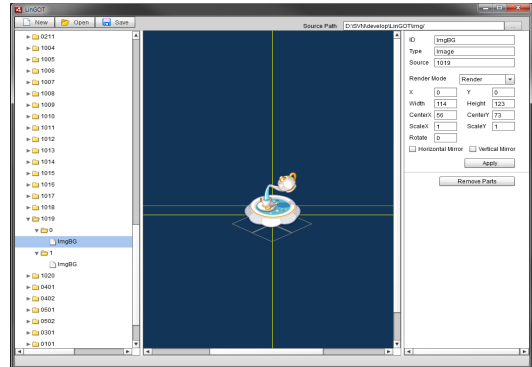
가능하다. 한가지는 비트맵 이미지를 대상 버퍼에 그대로 복사하여 사용하는 방식으로, 이미지를 변형하여 렌더링 할 수 없지만 빠른 속도로 렌더링 할 수 있다. 이 방식은 `BitmapData.copyPixels` 함수를 사용한다.

또 다른 방식은 `BitmapData.draw` 함수를 사용하는 방식으로 `Matrix`와 `ColorTransform` 등을 이용하여 변형된 렌더링이 가능하지만 속도가 느리다는 단점이 있다. 본 엔진에서는 렌더링 객체들에 두 가지 방식을 모두 지원하였으며, 엔진의 사용자가 사용 용도에 맞게 선택하여 게임의 성능을 높이고, 게임의 구조는 확장성을 높게 하였다.

#### (4) 캐릭터 애니메이션

캐릭터 애니메이션은 시뮬레이션을 기본으로 하는 SNG 게임에서 꼭 필요로 하는 기능이다. 그러나 SNG의 특성상 3D의 사용이 쉽지 않고, 2D 스프라이트 방식은 많은 메모리를 필요로 하기 때문에 스프라이트 방식의 캐릭터 애니메이션은 사용하기가 쉽지 않다. 그래서 본 엔진에서는 두 가지 방식을 혼합하여 작은 메모리 사용으로 효율적인 성능을 내는 방식을 선택하였다.

캐릭터는 다음과 같이 파츠라고 하는 작은 2D 이미지들로 이루어지며, 각각의 파츠에 모션을 추가하여 캐릭터가 실제 애니메이션하는 것처럼 구현하였다.



(그림 12) 오브젝트 저작도구

## 4. SNG 엔진의 구현

### 4.1 개발환경

본 연구의 SNG 게임 엔진 개발 환경은 다음과 같다. 엔진의 제작 도구로 FlexBuilder4.0을 사용하였으며, 개발 언어는 액션 스크립트 기반으로 구현하였다. 액션 스크립트 기반으로 작성된 이유는 기존의 SNG 서비스가 대부분 플래시를 기반으로 작성되었기 때문이다. 또한 리소스 및 게임 데이터 관리를 위해 XML을 사용하였다. UI툴은 Flex AIR를 사용하여 데스크탑에서 설치하여 사용할 수 있도록 하였다. 성능 평가는 CPU 펜티엄 듀얼 코어 CPU E6300 2.8GHZ과 메모리 4GB의 컴퓨터에서 테스트하였다.

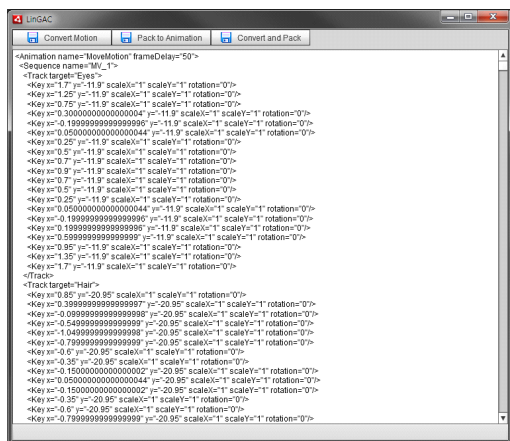
### 4.2 모듈별 저작도구

본 연구의 엔진 개발을 보다 간편히 할 수 있도록 몇 가지 객체생성을 위한 저작도구를 개발하였다. 각 도구들의 상세 기능은 다음과 같다.

#### (1) 오브젝트 저작도구

오브젝트 저작도구는 시뮬레이션 게임에서 사용할 수 있는 오브젝트들을 쉽게 제작하고 사용할 수 있게 해주는 툴이다. 툴의 왼쪽에는 오브젝트 목록, 가운데에는 렌더링 화면, 그리고 오른쪽에는 속성 창으로 구성되어 있다.

오브젝트 목록에서 특정 오브젝트와 오브젝트의 방향, 파츠들을 선택하게 되면, 선택된 오브젝트의 속성들이 속성 창에 바로 나타나고, 가운데 있는 렌더링 화면으로



(그림 13) 애니메이션 데이터 변환 저작도구

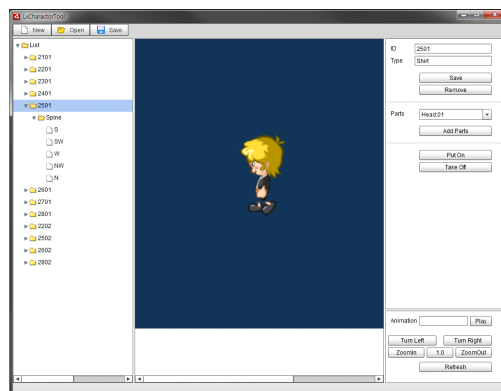
출력되게 된다. 따라서 원하는 오브젝트를 쉽게 수정 가능 하며, 수정된 결과를 바로 확인하면서 작업할 수 있게 된다.

### (2) 애니메이션 데이터 변환 저작도구

(그림 13)의 애니메이션 데이터 변환 툴은 Flash에서 뽑아낸 애니메이션 데이터를 본 엔진에서 사용 가능한 형식으로 변환해주는 툴이다. 세가지 메뉴로 구성되어 있으며, 가운데 변환결과를 확인할 수 있는 편집 창이 붙어 있다. 첫 번째 메뉴는 하나의 모션 파일을 변환하는 메뉴이고, 두 번째 메뉴는 변환된 모션 파일들을 하나의 애니메이션 파일로 묶어주는 기능이다. 마지막으로, 세 번째 메뉴는 두 개의 기능을 한번에 처리하게 해주는 기능을 한다.

### (3) 캐릭터 저작도구

(그림 14)의 캐릭터 저작툴은 시뮬레이션 게임에서 자주 사용되는 캐릭터 모듈을 사용하기 위해 필요한 데이터를 추출하기 위한 툴이다. 캐릭터 저작 툴에는 크게 두 가지 기능이 존재한다. 첫 번째 기능은 캐릭터를 꾸미는데 사용되는 캐릭터 아이템을 관리하는 기능이고, 두 번째 기능은 캐릭터에 애니메이션을 적용하여 확인해 볼 수 있는 기능이다. 툴의 왼쪽에는 캐릭터 아이템 목록이 있고, 툴의 가운데에는 렌더링 화면, 그리고 오른쪽에는 아이템 속성 창과 애니메이션 컨트롤러가 있다. 캐릭터 아이템 목록을 선택한 후에 오른쪽의 속성 창을 통해 값을 수정하고 캐릭터에 적용하거나 해지할 수 있다. 또한



(그림 14) 캐릭터 저작도구



(그림 15) 맵 저작도구

애니메이션 컨트롤러를 통해 원하는 애니메이션을 실행 하거나, 캐릭터의 방향 및 확대, 축소를 할 수 있다.

### (4) 맵 저작도구

(그림 15)의 맵 저작 툴은 오브젝트 툴과 애니메이션 변환 툴, 그리고 캐릭터 저작 툴에서 추출된 데이터들로 맵에 설정해보고 시뮬레이션 해볼 수 있게 해주는 툴이다.

왼쪽의 오브젝트 목록과, 가운데 렌더링 화면으로 구성되어 있다. 오브젝트 목록을 클릭하면 해당 오브젝트가 생성되고 맵의 특정 위치에 위치시킬 수 있다. 렌더링 화면으로는 시뮬레이션 결과를 확인해 볼 수 있다.



(표 2) 이미지 출력 테스트 실험

이미지 크기 (px)	그리기 회수 (회)	소요 시간(ms)	
		Blit	Render
92x94	100	2	15
	1000	25	140
	10000	250	1410
138x41	100	7	20
	1000	51	185
	10000	520	1900

(표 3) 텍스트 출력 테스트 실험

그리기 회수 (회)	소요 시간(ms)	
	Blit	Render
100	3	20
1000	20	110
10000	70	1110

(표 4) 캐릭터 출력 테스트 실험

그리기 회수(회)	소요 시간(ms)
20	29
30	39
50	53
100	91

### 4.3 실험결과

#### (1) 이미지 출력테스트

2D 게임은 화면의 대부분을 이미지로 연출하게 된다. 그러므로 프레임당 많은 이미지들을 출력해야 되고, 이미지 출력 시간이 게임 업데이트 루프에서 대부분의 시간을 차지하게 된다. 결국 각각의 이미지들을 얼마나 효율적으로 출력하느냐가 게임을 얼마나 빠르고 가볍게 실행하느냐를 결정하게 된다.

이미지 출력 테스트는 특정 크기의 이미지를 일정 회수 반복해서 그린 후 소요 시간을 측정하는 방식으로 진행하였으며, 본 엔진에서 제공하는 두 가지 그리기 방식에 대해 테스트하여 서로 비교하였다.

우선, Blit(copyPixels) 방식은 출력하고자 하는 이미지를 스크린 백퍼피에 단순히 픽셀단위로 복사를 하는 방식으로 이미지의 회전, 확대, 색상 연산 등의 기능을 사용할 수 없지만, 빠른 속도로 출력할 수 있다는 장점이 있다. 반대로 Render(draw) 방식은 출력하고자 하는 이미지에 회전, 확대, 색상 연산 등의 기능을 사용할 수 있지만, Blit 방식에 비해 속도가 느린 단점이 있다.

테스트 결과, 초당 30프레임을 기준으로 하였을 때, 92x94크기를 갖는 이미지의 경우 Blit 방식으로는 프레임당 1000회 이상, Render 방식으로는 프레임당 160회 이상 출력할 수 있었다.

#### (2) 텍스트 출력 테스트

게임에서 이미지 다음으로 화면 연출에 많이 사용되는 것이 문자열이다. 문자열 출력은 설정된 폰트로부터 출력 정보를 얻어와 화면에 출력하게 된다. 본 엔진에서는 폰트의 실시간 처리로 인한 속도 감소를 막기 위해 문자열의 변경이 있을 때에만 문자열을 이미지화하여 저장

한 후 화면에 출력하는 방식을 사용하였다.

테스트 결과, 초당 30프레임을 기준으로 하였을 때, 7글자의 문자열의 경우 Blit 방식으로는 프레임당 1500회 이상, Render 방식으로는 프레임당 150회 이상 출력할 수 있었다.

#### (3) 캐릭터 출력 테스트

본 엔진에서는 시뮬레이션 장르의 SNG 개발을 돕기 위해 캐릭터 모듈을 추가하였다. 캐릭터는 14개의 파츠로 구성되어 있고 모든 파츠가 Render 방식으로 출력되기 때문에, 출력 소요 시간이 오래 걸린다.

테스트 결과, 초당 30프레임을 기준으로 하였을 때, 프레임당 20회 이상 출력할 수 있었다.

## 5. 결 론

본 논문에서는 액션 스크립트 기반의 소셜 게임 엔진을 설계하고 구현하였다. 엔진 내부적으로는 기존 게임 엔진과 다르게 소셜 게임 서비스에 특화된 모듈을 제안하였고, 게임 제작의 효율성을 높이기 위해 각종 툴들을 제작 하였다. 본 논문에서 제시한 엔진은 SNG 서비스 환경의 특징을 고려하여 설계하였다. 특히, SNG 서비스 환경이 웹 환경인 것을 고려하여 네트워크 환경에서 리소스를 효율적으로 관리할 수 있도록 하였으며, 효율적인 렌더링 구조를 지원하여 최대한의 성능을 얻고자 하였다. 또한, SNG 게임들 중에서 큰 규모라고 할 수 있는 시뮬레이션 게임 제작을 위해 필요한 오브젝트 툴, 캐릭터



툴, 애니메이션 툴, 맵 툴 등을 제공하였다.

본 논문에서 구현한 엔진의 성능은 이미지, 텍스트, 캐릭터 애니메이션의 출력 회수로 평가하였다. 그 결과는 세 부분 모두 게임을 구현하기에 충분히 빠른 처리 속도를 보여주었다.

향후 과제로는 모바일 시장이 확대됨에 따라 모바일 기반에서도 사용할 수 있는 버전을 구현할 예정이다. 또한, 구현된 엔진을 이용하여 다양한 게임에 적용하면서 설계를 보완 개선할 예정이다.

### 참 고 문 헌

- [1] N. Ellison, and D. Boyd, "Social Network Sites: Definition, History, and Scholarship", *Journal of Computer-Mediated Communication*, vol. 13, issue 11, pp210-230, 2008.
- [2] R. Baden, A. Bender, N.I Spring, and B. Bhattacharjee, "Persona: an online social network with user-defined privacy", *Proceedings of SIGCOMM '09*, pp135-146, 2009.
- [3] F. Bonchi, C. Castillo, A. Gionis, and A. Jaimes, "Social Network Analysis and Mining for Business Applications", *Transactions on Intelligent Systems and Technology (TIST)*, Vol. 2 Issue 3, article 22. 2011.
- [4] FarmVille, <http://apps.facebook.com/onthefarm/>
- [5] J.Sung, T.Bjornrud, Y.Lee, and D.Y.Wohn, "Social network games: exploring audience traits", *Proceedings of the 28th of the international conference(CHIEA'10)*, pp.3649-3654, 2010
- [6] R. Ossmann and K. Miesenberger, "Accessibility of a social network game", *LNCS vol. 6179*, pp243-246, 2010
- [7] E. Lengyel, "Game Engine Gems", Jones and Bartlett Publishers, 2011.
- [8] J. Gregory, "Game Engine Architecture", A. K. Peters, 2009.
- [9] Aki Jarvinen, "Game Design for Social Networks: interaction design for playful dispositions", *Proceedings of the ACM SIGGRAPH Symposium on Video Games*, pp95-102, 2009.
- [10] Aki Jarvinen, "Workshop: game design for social networks", *Proceedings of 13<sup>th</sup> international MindTrek Conference(MindTrek'09)*, pp224-225, 2009.
- [11] K. Chen and C. Lei, "Network game design: hints and implications of player interaction", *Proceedings of 5<sup>th</sup> ACM SIGCOMM workshop on Network and system support for games*, pp1-9, 2006.
- [12] Restaurantcity, <http://apps.facebook.com/restaurantcity>
- [13] list of flash game engines, <http://www.flashrealtime.com/flash-game-library-engine-list/>
- [14] C. Moock, "Essential ActionScript 3.0", O'Reilly, 2007.

● 저 자 소 개 ●

**우 종 우 (Chong-Woo Woo)**

1988년 미국 Illinois Institute of Technology 전산학과(박사)  
1994년~현재 국민대학교 컴퓨터공학부 교수  
관심분야 : 지능형 에이전트, 상황인식, 모바일게임, 시뮬레이션  
E-mail : cwwoo@kookmin.ac.kr



**김 대 령 (Daeryung Kim)**

1998년 국민대학교 컴퓨터공학부 졸업(학사)  
2011년 국민대학교 대학원 컴퓨터공학 졸업(석사)  
2010~현재 (주) Linyworks 연구원  
관심분야 : 지능형 에이전트, 모바일게임, 시뮬레이션  
E-mail : drforges@gmail.com

