

## 3차원 게임 개발 사례: Dream3D 시범 콘텐츠

이 헌 주, 박 태 준, 김 현 빈  
한국전자통신연구원 컴퓨터소프트웨어연구소 가상현실연구부  
이메일 : {hjoo, ttjpark, hbkim}@etri.re.kr

### A Development Case of 3D Game: Dream3D Prototype Game Contents

Hun-Joo Lee, Tae-Joon Park, Hyun-Bin Kim  
Virtual Reality Research Department, ETRI  
E-mail : {hjoo, ttjpark, hbkim}@etri.re.kr

#### Abstract

Computer game has become the core part of multimedia and entertainment in our knowledge and information based society. We are competitive in developing 2D games. However, we have difficulties in maintaining the competitive edge in the area of 3D game technologies owing to the limited funds and technologies. Game engines are very useful to develop high-quality 3D games. In this paper, we discuss game engines and development of contents, and also present a development case of 3D games using Dream3D.

#### I. 서론

정보통신 분야의 하드웨어와 소프트웨어 기술의 빠른 발전으로 컴퓨터가 초고속, 대용량화하면서 가상현실이나 3차원 응용기술이 보편화되어 가고 있다. 뿐만 아니라 엔터테인먼트 산업에 대한 관심이 커지고 가정용 PC의 보급이 확대됨에 따라 컴퓨터 게임은 일반인들이 손쉽게 접할 수 있는 주요 엔터테인먼트 문화 가운데 하나로 자리잡아 가고 있는 추세이다. 최근에 컴퓨터 게임 개발사들은 현실감 있는 게임을 만들기 위하여 컴퓨터 그래픽 기술을 최대한 활용하여 화려하고 사실적인 그래픽을 제공하려 하고 있다. 한편, 기존의

게임은 2차원 형태의 게임들이 주류를 이루었으나, 사용자들의 요구가 2차원에서 3차원으로 변함에 따라 3차원 그래픽 기술이 적용된 게임들이 많이 개발되고 있는 추세이다. 3차원 컴퓨터 게임 제작을 위해서는 빠르고 사실적인 렌더링 기술과 자연스러운 애니메이션 기술이 필수적이며 3차원 음향 기술을 적용하여 실감나는 게임을 즐길 수 있도록 하고 있다[1].

3차원 게임의 유행과 더불어, 인터넷 기술의 발달로 인하여 컴퓨터 게임에 네트워크 기능을 추가한 온라인 게임이 주목을 받고 있으며 PlayStation2나 Xbox 등과 같은 비디오 게임도 온라인이 지원되고 있는 추세이다 [2]. 또한 무선 인터넷이 등장함으로써 모바일 게임이라는 무선 환경의 새로운 영역이 빠른 속도로 급성장하고 있다.

본 논문에서는 최근 기술을 적용한 3차원 게임의 개발 사례에 대하여 기술한다. 개발된 게임은 게임엔진 및 다양한 게임 에디터를 이용하여 제작되었으며, 게임엔진의 기능 및 성능을 검증하기 위한 시범 콘텐츠로 제작되었다. 2장에서는 게임을 개발하기 위한 일반적인 방법론에 관하여 기술하고, 3장에서는 제작된 3차원 게임의 개발 방법 및 내용을 알아본다. 끝으로 4장에서 결론을 맺도록 한다.

#### II. 게임엔진과 콘텐츠 개발

초기의 게임은 한 사람이 프로그램, 그래픽, 사운드

등 모든 것을 혼자 개발하기도 하였다. 그러나 사용자의 요구사항이 증가하고 어려워졌으며, 하드웨어의 발전과 더불어 새로운 게임 제작 기술이 발달하게 되었다. 이러한 변화에 의하여 게임 제작은 점차 세분화되었고, 게임엔진의 필요성이 중요시되게 되었다.

각 게임 개발사가 보유하고 있는 게임 제작을 위한 각종 노하우(knowhow)와 기술, 그리고 하드웨어를 제어하기 위한 각종 API의 사용법 등은 게임 하나만을 위해서만 사용되고 폐기되는 것이 아니라, 같은 게임 개발사의 다른 게임의 제작에도 계속적으로 보완 활용된다. 이러한 기술 재사용의 편의성을 위하여 각 게임 개발사는 자사의 노하우 및 기술을 언제나라도 수정 및 보완하고 재사용할 수 있는 소프트웨어 엔진의 형태로 관리하고 있다. 다시 말하여 게임 엔진이란 게임 제작에 자주 활용되는 개발 노하우나 기술, 하드웨어 API 사용법 등을 쉽게 활용할 수 있는 소프트웨어 라이브러리나 모듈의 형태로 구현해 놓은 것을 말한다.

게임엔진을 이용하여 게임을 개발할 경우에 개발기간의 단축 및 개발비용을 절감할 수 있다는 장점이 있으며 양질의 콘텐츠를 제작할 수 있다[3]. 자체적으로 게임엔진을 만들고 이를 이용하여 게임을 개발하거나, 상용화된 게임엔진을 도입하여 개발할 수도 있다. 표 1은 현재 상용화되어 있는 게임엔진 가운데 대표적인 것들을 비교한 것이다.

표 5. 대표적인 상용 게임엔진의 비교

게임엔진	Quake	Unreal	Genesis3D
기본 기능	실내	실내	실내
애니메이션	Mesh, Skeletal	Mesh, Skeletal	Mesh, Skeletal
스크립트 언어	Quake-C	Unreal Script	×
다중참여자	~10	~10	×
상용게임	Quake, Half-Life 등	Unreal, Deus-Ex 등	×
장르	액션	액션	액션
에디터	Q3Radiant	UnrealED	World

게임 콘텐츠의 개발은 크게 게임 기획, 상세 기획, 프로그래밍의 단계를 거친다. 게임엔진의 프로그래밍 단계에서 가장 중요한 역할을 담당하지만, 기본적인 게임의 기능을 제공하기 때문에 게임의 기획 단계에서부터 게임엔진의 선택이 게임 전체의 질을 결정하는 중요한 요소로 작용된다[5].

좋은 성능을 가진 엔진이 있다고 하더라도 그 엔진 내의 각 세부 모듈을 통합하여 하나의 콘텐츠로 만들기 위해서는 단순히 엔진을 조합하기만 하면 되는 것이 아니라 게임 개발 프로젝트 전체에 대한 이해와 세부 엔진에 대한 이해가 필요하다. 세부 엔진의 기능 중 개발하고자하는 콘텐츠에서 필요한 기능을 알아내고, 그것을 사용하고 관리할 모듈을 만들어내야 한다. 자신이 직접 엔진을 개발하였을 경우에는 상관없지만 외부에서 가져온 경우 반드시 필요한 작업이다. 이 경우 자신에게 필요한 부분을 가려낼 뿐만 아니라 부족한 부분을 알아내어 직접 작성을 해야 하기도 한다. 또한 최근의 게임 개발 경향에서 볼 수 있듯이 각 엔진을 여러 명의 개발자가 나누어 개발하는 경우에는 서로간에 개발한 엔진의 기능과 사용법에 대하여 충분히 숙지하여야 한다. 즉, 개발자는 게임의 기획을 충분히 이해하고 숙지한 후 설계에 들어가는 것이 바람직하다. 프로그래머는 실제 프로그래밍 작업에 들어가기 전에 최대한 상세한 부분까지 설계하여야 한다.

게임 프로그래밍의 기본 골격은 게임 진행모듈에 의하여 구성된다. 게임 진행모듈이란 세부 엔진을 유기적으로 통합하고, 각종 데이터를 효율적으로 관리하며, 기획을 충실히 반영하여 콘텐츠를 개발하는 데에 필요한 모듈을 말한다. 렌더링, 애니메이션 엔진 등 세부 엔진과 기획서가 충실히 준비되었다고 하더라도 그것을 이용하여 성공적인 게임 콘텐츠를 만들기 위해서는 많은 추가 작업이 필요하다. 일반적으로 게임이 진행될 때에 처리되는 과정은 아래와 같다.

```

while(!End)
processInput();// 입력 처리
updateGameData();// 게임 객체/상태 처리
updateAnimation();// 애니메이션 처리
updateSound();// 사운드 처리
render();// 화면 그리기
    
```

위와 같은 과정이 계속 반복된다고 해서 진행 루프라고 부르며, 매번 새로운 화면을 프레임(frame) 단위로 갱신하므로 프레임이라고 부르기도 한다.

### III. Dream3D 시범 콘텐츠 개발

#### 3.1 Dream3D

Dream3D는 3차원 온라인 게임엔진으로 게임 그래

픽 데이터를 게임진행 상황에 맞게 화면에 출력해 실시간 처리가 가능하게 해주는 렌더링엔진, 모션 캡처 데이터를 활용할 수 있는 애니메이션엔진, 좌우 사운드 출력만으로도 3차원 음향효과를 생성할 수 있는 사운드엔진으로 구성되어 있다. 또한 동시 접속자의 증감과 네트워크 트래픽 상황을 고려하여 자동으로 분산 처리 기능을 지원하는 서버엔진, 그리고 게임 기획자나 디자이너가 시나리오를 기준으로 게임공간의 구성, 이벤트 설정 등 게임진행에 필요한 모든 요소를 통합

된 환경에서 테스트하고 피드백 할 수 있는 게임 에디터로 구성되어 있다. 표 2는 세부 엔진별로 제공되고 있는 기능을 나타낸다.

### 3.2 시범 컨텐츠 개발

#### (1) 3차원 게임을 위한 데이터 및 객체 제작

게임에서 사용되어질 게임 환경을 만들기 위해서는 맵(map) 에디터라는 게임 에디터를 사용하였다. 게임의 지형 편집은 실외 지형의 기반이 되는 하이트맵의 높이 정보를 편집하고, 이에 따라 지형 메쉬(mesh)를 새로 구성하는 과정을 거치게 된다. 게임 지형의 경우 다른 게임 객체와 같이 3D MAX나 MAYA와 같은 전문 저작 도구를 이용하여 만들어 쓸 수도 있지만, 게임 지형에는 객체와는 달리 게임을 진행할 때 꼭 필요로 하며 게임에 특성화된 정보들을 많이 포함되어 있기 때문에 맵 에디터를 이용하여 만드는 것이 효과적이다. 그림 1은 에디터를 이용하여 게임 지형을 편집하는 모습을 보여주고 있다.

표 2. Dream3D의 기능

세부 엔진	기능
렌더링	<ul style="list-style-type: none"> <li>- 3차원 객체 출력</li> <li>- 게임 환경 출력</li> <li>- 광원 및 그림자 출력</li> <li>- 카메라 제어</li> <li>- 특수효과 출력</li> <li>- 텍스처 기반 특수 기능</li> </ul>
애니메이션	<ul style="list-style-type: none"> <li>- 캐릭터 애니메이션</li> <li>- 고급 동작 제어 기능</li> <li>- 다양한 객체 애니메이션</li> <li>- 물리엔진</li> </ul>
사운드	<ul style="list-style-type: none"> <li>- 사운드 재생, 정지, 믹싱 등 사운드 객체 출력 기능</li> <li>- 2채널 기반의 3차원 사운드 기능 (음상/음장 제어)</li> <li>- 사운드 특수효과 제공</li> <li>- 게임용 사운드 모델에 의한 사실적인 음원 발생 및 감쇄 기능 제공</li> </ul>
게임 서버	<ul style="list-style-type: none"> <li>- IOCP 기반의 thread pooling 기능</li> <li>- 네트워크 부하의 최소화를 위한 데드 레커닝(dead reckoning) 기능</li> <li>- 셀 기반의 AOI(Area of Interest) 관리 기능</li> <li>- 사용자의 확장성을 위한 다중/분산 서버 기능</li> <li>- 서버간 seamless 이동</li> </ul>
게임 에디터	<ul style="list-style-type: none"> <li>- 하이트맵(height map) 기반의 실외 지형 관리 기능</li> <li>- BSP/CSG 기반의 실내 관리 기능</li> <li>- 게임 객체 관리 기능</li> <li>- 게임 스크립트 관리 기능</li> <li>- 게임 환경 설정 기능</li> <li>- 게임 시뮬레이션 기능</li> <li>- 특수효과 편집 기능</li> </ul>

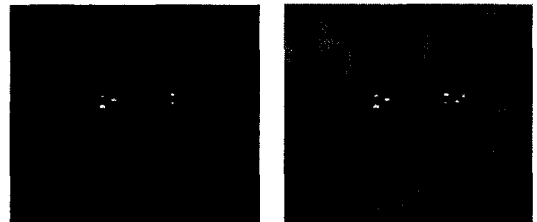


그림 1. 게임 에디터를 이용한 지형 편집

게임의 캐릭터 제작은 에디터를 이용하여 캐릭터를 디자인하고, 모션 데이터를 리타겟팅(retargetting)하여 캐릭터의 관절길이 크기에 맞게 변형하고, 모션 데이터를 직접 키프레임(key frame) 형태로 편집하였다. 그림 2는 게임에서 사용된 캐릭터의 제작 화면을 보여주고 있다.

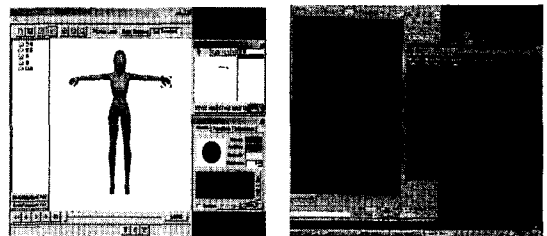


그림 2. 캐릭터 및 애니메이션 편집

캐릭터의 움직임은 모션 캡처(motion capture) 시스템을 이용하여 캡처된 데이터를 편집할 수도 있다. 모션을 자르거나 이어서 붙이거나 역운동학을 적용하는 등의 편집이 가능하다. 에디터를 이용하여 제작된 캐릭터 및 움직임 데이터는 애니메이션 엔진으로 로딩하여 애니메이션을 처리하게 된다.

3D MAX나 MAYA와 같은 사용 저작도구에서 개발된 객체는 Dream3D의 프리뷰어(previewer)를 이용하여 실제 게임과 동일한 환경으로 볼 수 있으며 상용 저작도구에서 조절하지 못하는 게임 속성들을 편집할 수 있게 해준다. 프리뷰어는 Dream3D의 렌더링 모듈과 애니메이션 모듈, 사운드 모듈을 이용하여 동작하기 때문에 게임 화면과 동일한 렌더링 품질을 보장한다. 그림 3은 제작된 캐릭터를 프리뷰어에서 제어하는 모습을 보여주고 있다.



그림 3. 프리뷰어에서의 렌더링 방식  
(왼쪽부터 Solid, Wire, Bone)

(2) 콘텐츠 결과

Dream3D의 기능 및 성능을 검증하고 실제 게임 제작시의 문제점을 발견하여 이를 보완하기 위하여 Dream3D를 기반으로 하여 3차원 게임을 시험 제작하였다. 제작된 게임은 렌더링 엔진 및 애니메이션 엔진을 이용하여 최적화된 화면을 구성하였고, 데이터 추출기 및 프리뷰어를 이용하여 애니메이션 데이터와 사운드의 동기화 기능을 구현하였다. 또한, 캐릭터의 거리에 따르는 카메라의 이동으로 게임 화면의 분위기가 최대한 살아나도록 하였다. 그림 4는 구현된 시범 콘텐츠의 실행 모습을 보여주고 있다.

IV. 결론

본 논문에서는 게임을 개발하기 위한 방법과 게임엔진 기술에 대하여 알아보았다. 그리고 최근 기술을 적용하여 제작된 3차원 게임의 개발 사례에 대하여 기술하였다. 개발된 게임은 게임엔진 및 다양한 게임 에디

터를 이용하여 제작되었으며, Dream3D 게임엔진의 기능 및 성능을 검증하기 위한 시범 콘텐츠로의 역할을 할 수 있었다.

향후에는 많은 사용자가 동시에 참여하여 게임을 할 수 있는 다중 참여자 온라인 게임으로 발전시켜서, 대규모 온라인 게임을 위한 서버의 안정성 검증이 가능하도록 하는 연구가 이루어져야 할 것이다.

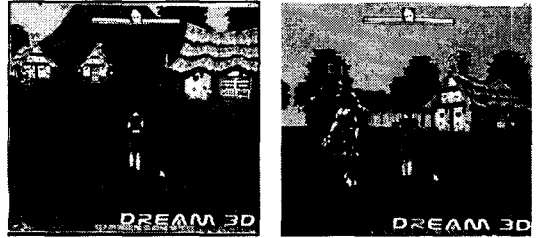


그림 4. 시범 콘텐츠 실행 화면

참고문헌

- [1] A. Watt, F. Policarpo, "3D Games: Real-time Rendering and Software Technology", Addison-Wesley, 2001.
- [2] D. H. Eberly, "3D Game Engine Design: A Practical Approach to Real-Time Computer Graphics", Morgan Kaufmann Publishers, 2001.
- [3] 이현주, 김준애, 임충규, 김현빈, "온라인 3D 게임 엔진 표준화", 한국정보처리학회지, 2002년 5월호.
- [4] A. Watt, "3D Computer Graphics, 3rd edition", Addison-Wesley, 2000.
- [5] Tae-Joon Park, Soon Hyoung Pyo, Chang Woo Chu, and Byoung Tae Choi, "Design and Implementation of a Rendering Engine for Developing Computer Games", Japan Korea Computer Graphics Conference, 2001.
- [6] Mark DeLoura, "Game Programming Gems 3", Charles Rivermedia, 2002.
- [7] 양광호 외 4인, "온라인 게임 서버의 기술 동향," 전자통신동향분석 제 16권 제4호, 2001년 8월호.
- [8] Mason McCuskey, "Special Effects Game Programming with DirectX", Premier Press, 2002.
- [9] Andrew Rollings, Dave Morris, "Game Architecture and Design," Coriolis, 2000.