



## G-Blender 게임엔진

현 호 진  
(주니어블)

### 목 차

1. 3D 게임 엔진 G-Blender
2. G-Blender 소개
3. 제작 중인 게임 소개

### 1. 3D 게임 엔진 G-Blender

1995년도에 마이크로 소프트 Windows 95가 발표되면서 퍼스널 컴퓨터에서 다양한 게임 개발 환경이 제공되었다. 그 중 Real Time 3D Graphics 구현을 목적으로 SGI에서 제공한 OpenGL 라이브러리가 Windows에서 기본 제공된 것은 3차원 게임 개발을 지원하는 첫 단추였고, VOODOO칩이라는 3D 가속 전용 하드웨어의 출시는 퍼스널 컴퓨터에서 30프레임 이상의 3D 게임을 플레이 할 수 있는 환경을 제공해주었다. 이런 환경 하에서 차츰 3차원 영상을 실시간으로 플레이 할 수 있는 게임들이 발표되기 시작했고 대표적인 게임이 ID Soft의 Quake와 Core Design의 Tomb Raider였다.

3D 게임을 개발하기 위한 환경을 구축하는데 비용과 시간이 많이 소요되었다. 이러한 비용과 시간을 줄이기 위해서 이미 개발된 3D 게임의 라이브러리를 구입하여 사용하게 되었다. 이후 3D 게임 라이브러리는 급속하게 발전되어 3D 게임 엔진의 형태가 되었고 전문적으로 제작되

어 상품화 되었다. 1997년 지금의 G-Blender도 범용적인 렌더링 엔진과 애니메이션 엔진을 목표로 개발을 시작하였다.

초기 엔진제작의 목적은 게임 개발에서 3ds max 2.0 기반의 플러그인과 간단한 뷰어를 제공하여 게임 개발을 빠르게 하는 것이었고 C언어로 디자인 되었다. 그러나 점점 많은 기능을 담아야 했고 유저들에게 다양한 형태의 렌더링 함수를 제공해야 했기 때문에 1999년 이후에는 C++로 개발되었다.

2000년에는 Web 상에서 실시간으로 3D Animation을 볼 수 있는 “G-Power”라는 상품을 G-Blender의 초기 버전으로 제작하여 출시했고, 2001년에는 (재)한국게임산업개발원의 ‘범용 3D 게임 엔진 개발 사업’에 선정되어 OpenGL기반의 “GI3D 게임 엔진”을 출시했고 이를 이용해 아케이드 일인칭 슈팅 “폴트리거”, 조이온에서 서비스하는 온라인 슈팅 “팡팡테리블” 등이 상용화되었다.

2003년 Direct3D를 사용한 렌더러로 렌더링 API를 바꾸고 유연한 Shader기반의 Script를 제

〈표 1〉 연도별 3D 게임 엔진 개발 현황

연도	국내관련산업현황	국내엔진개발 현황	해외 현황
1990~1994	2D PC 패키지 게임이 주류	엔진개념 전무	3D엔진 연구 본격화
1994~1997	온라인게임개발시작 MUD게임서비스 MUG 게임태동기	엔진개념 습득 초보적	3D 엔진을 통한 콘텐츠 개발 시작 퀘이크 출시
1998~2000	3D Game에 대한 태동기 엔진에 대한 인식 태동기	3D 관련 기술 습득을 통한 엔진 개발 가이블 웹3D 출시 및 엔진개발 본격화	자체엔진의 판매 전문엔진개발사등장
2001~2002	범용엔진 개발 추진	3D Game 출시 시작 자체 엔진에 대한 투자 해외 라이선스 취득 가이블 GI3D 출시	엔진 개념 일반화 엔진 라이선스 일반화
2002~현재	대부분 업체 상용엔진 개발포기, Game사로 변환 엔진에 대한 인식 일반화 G-Blender상용화 엔진 라이선스	ETRI Dream3D 출시 G-Blender 라이선스 기술 서비스 체제 구축	지속적 업그레이드

공하는 엔진 G-Blender를 출시, 현재 ㈜COWON, 소프트온 엔터테인먼트, 이노디스, 사이닉소프트, 이젠 등 30여 개 게임 개발 업체와 성균관대학교, 동서대학교, 한국기술교육대학교 등 10여 개 대학교에 라이선스하였다. <표 1>은 게임엔진에 관한 간략한 역사적 흐름을 보여준다.

## 2. G-Blender 소개

G-Blender는 3D 객체의 애니메이션과 렌더링하는 과정을 최적화하여 믿을 수 있는 성능을 제공하고, 유연한 구조로 다양한 환경의 3D 게임 제작을 소화해 낼 수 있도록 제작된 3D 엔진이다. 앞으로 (그림 1)을 중심으로 기술할 것이다.

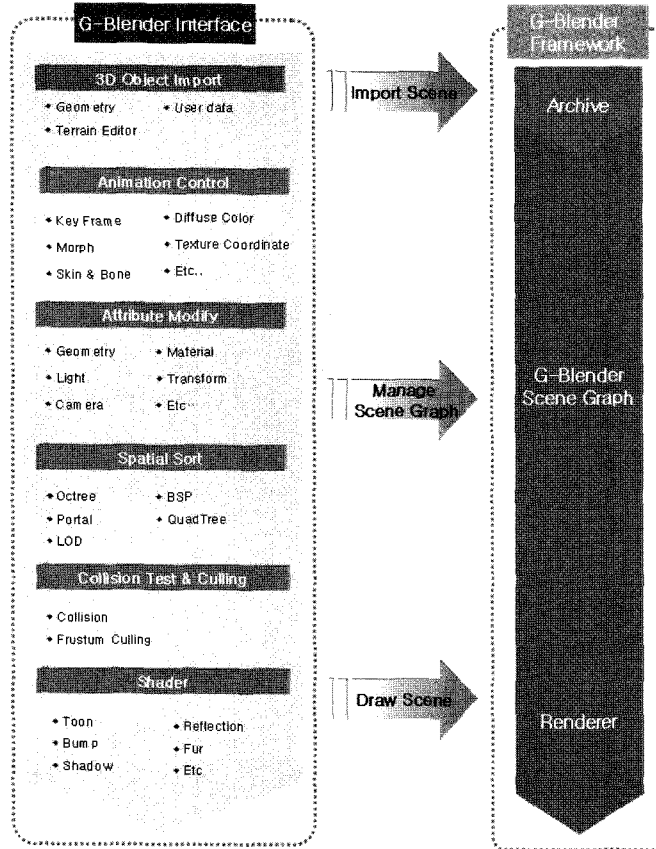
### 2.1 G-Blender Interface

G-Blender에서 제공하는 엔진 인터페이스는 크게 3가지의 종류로 나눌 수 있는데 첫째, 3ds max로 만들어진 3D 모델 파일과 텍스처와 같은 틀에서 만들어진 리소스들을 Import하고 이러한 리소스를 관리하는 인터페이스가 있다. 둘째, Import된 리소스들의 상태를 제어하고 각각의 상태를 유지하는 컨트롤러 들이 있고, 이러

한 각각의 객체들을 조합하여 장면 그래프 자료 구조로 관리하고 내부에서는 스마트 포인터와 핸들 시스템으로 각각의 리소스를 공유하기 위해서 클론인터페이스를 제공한다. 마지막으로 이러한 3차원 객체들을 화면에 그리기 위해서 Shader와 Render State을 관리하는 렌더링 인터페이스가 있다. 렌더링 인터페이스 내부에는 렌더링 속도를 향상시키기 위해서 각각의 상태들을 그룹으로 묶는 작업을 하고 배치 최적화 작업을 자동으로 수행하며 SIMD 등과 같은 CPU 최적화 작업을 거쳐 화면에 그려주게 된다.

### 2.2 G-Blender Framework

엔진에서 제공하는 인터페이스를 통하여 원하는 애니메이션과 충돌 처리 그리고 객체의 질감 등을 결정했다면 이것들을 엔진의 메인 프레임웍에 넣어주게 된다. 이 작업으로서 화면에 작업한 3D 객체를 화면에 그릴 준비가 된 것이다. G-Blender 프레임웍 안에서는 CPU와 GPU의 최적화 작업을 통해서 등록된 3D 객체를 그리고자 하는 렌더 타겟에 한번에 그려주게 된다. 여기서 알파 블렌드 정렬, 렌더 스테이트 정렬, 파티클 이펙트 등의 작업이 이루어지고 정



(그림 1) G-Blender 파이프라인

적 버퍼가 아닌 동적 버퍼(예를 들자면 소프트웨어 스키닝)를 사용하는 객체들도 생산된다.

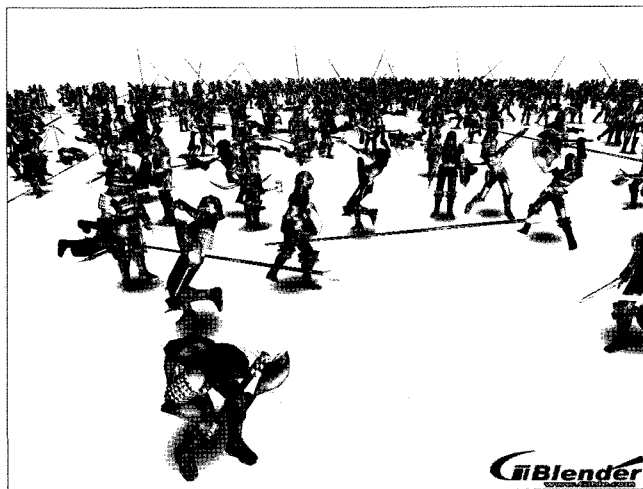
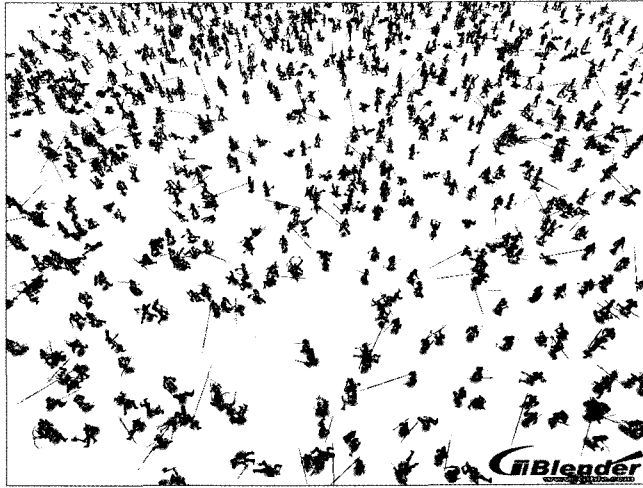
### 2.3 유연한 Shader 구조

DirectX8 이후에 소개된 Shader라는 프로그래머블 파이프 라인으로 게임 개발자는 GPU를 접근하여 좀 더 다양한 효과를 제공할 수 있게 되었다. G-Blender에서는 사용자가 쉽게 Shader를 제작하고 엔진에 적용할 수 있도록 G-Blender Annotation & Semantics를 제공한다. DirectX의 HLSL로 쉽게 제작할 수 있으며, G-Blender에서 읽어 들일 수 있도록 매뉴얼에서 제공하는 문법을 사용하여 Shader를 제작하면 다양한 파라미터의 애니메이션과 상태 변화를 손쉽게 할 수 있다. 또한 디자이너는 제공되는 G-Blender

Shader Plug-In을 사용하여 3ds max에서 간단한 인터페이스를 통해 재질의 질감 표현을 손쉽게 결정할 수 있다.

### 2.4 G-Blender의 배치 최적화

DirectX로 예를 들어 설명하면, DrawIndex Primitive 함수 같은 Draw Call이 한 번 호출 할 때를 1배치(Batch)라고 한다. 이 배치의 개수(Draw Call 개수)가 많아지면 CPU를 많이 사용하게 되고 따라서 게임의 초당 프레임이 저하되게 된다. 이러한 문제점 때문에 배치의 최적화는 게임의 성능을 올릴 수 있는 가장 쉬우면서도 중요한 요소이다. 엔진 내부에서는 배치의 최적화를 위해서 다양한 방법을 제공하고 있다. 이러한 최적화 방법을 통하여 게임의 속도를 적



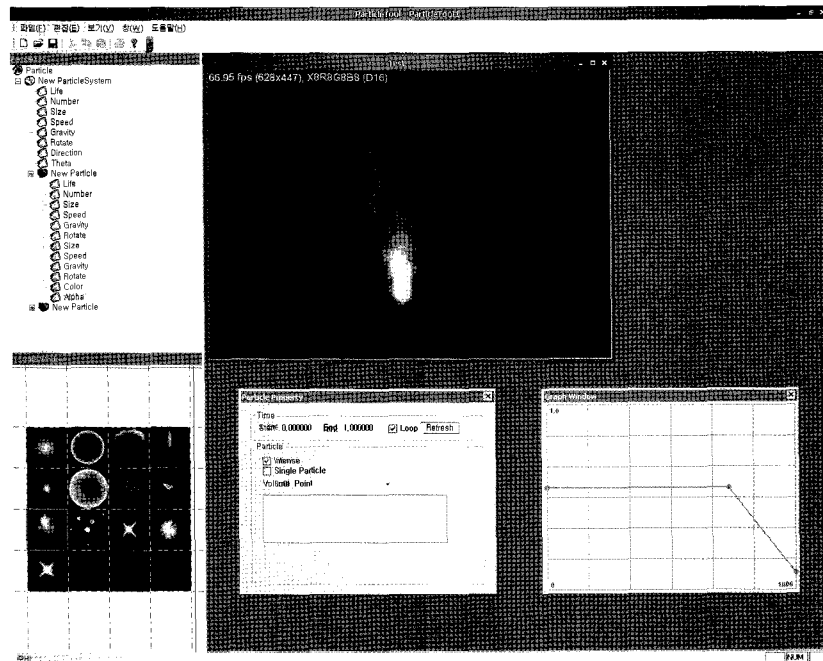
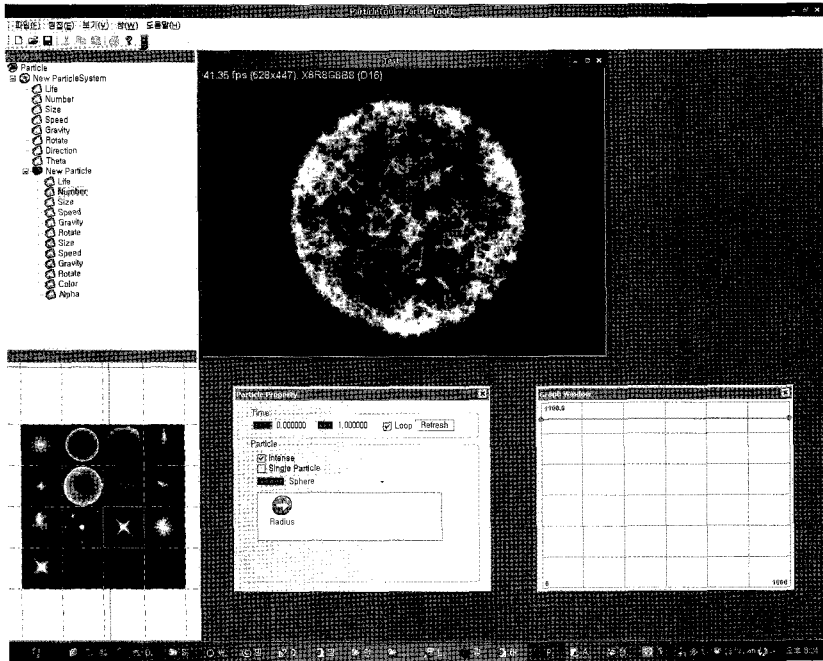
(그림 2) G-Blender 배치 최적화 데모  
(폴리곤 2400개, 16가지 애니메이션, 6가지 부분 조합된 캐릭터 총 2000명을 실시간으로 그려지는 데모)

계는 2배에서 많게는 4-5배까지 올릴 수 있다 (그림 2 참조).

### 2.5 이펙트 툴

3D 게임이 2D 게임보다 화려한 이유 중의 하나는 블렌드 효과를 사용한 다양한 이펙트를 만들어 낼 수 있다는 것이다. G-Blender에서는 특수 효과를 손쉽게 제작할 수 있는 다양한 툴들을 제공한다. DirectX의 Render State과 Texture Stage State을 손쉽게 조작할 수 있는 GMS Tool을 제공하고, 다양한 이펙트를 만들어내는

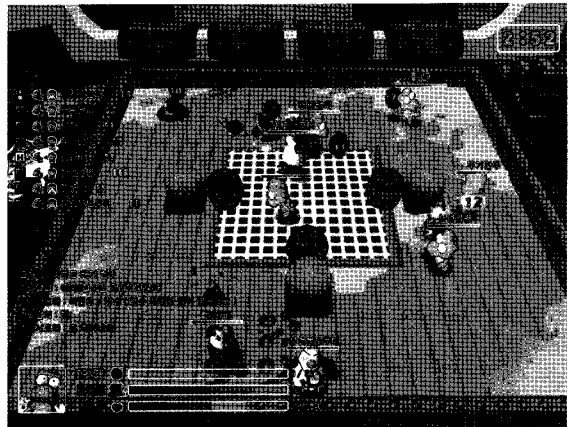
G-Blender Particle Tool을 제공한다. 파티클을 3D 화면에 렌더링할 때 가장 문제가 되는 것은 파티클 입자가 많아짐으로써 CPU연산이 많아지고 배치의 개수가 크게 증가하여 프레임이 많이 떨어진다는 것이다. 이러한 문제의 예는 일반적인 3D 게임에서 쉽게 찾아볼 수 있다. 자동차 경주 게임에서 배기가스 같은 부분을 파티클로 구현하는데, 게임 도중 배기가스가 많이 나올 때 프레임이 떨어지는 것을 느낄 수 있을 것이다. 이러한 문제를 G-Blender에서는 배치 최적화 작업으로 성능 최적화를 실현하였다. 또한,



(그림 3) G-Blender 이펙트 툴

파티클의 블렌딩 옵션 때문에 파티클 입자들의 정확한 가시거리 정렬이 필요하다. 파티클 입자들이 적게는 몇 백 개에서 많게는 몇 만개까지 실시간에 뿌려지므로 이들의 가시거리 정렬 또

한 성능 저하에 큰 요인이 된다. 이러한 문제점을 해결하기 위해 G-Blender는 최적화된 가시거리 정렬 알고리즘을 제공하여 프레임의 저하 없이 파티클을 원활하게 구현해 낼 수 있도록 했



(그림 4) G-Blender 1.6으로 제작 중인 사이닉 소프트의 '풍류공작소'

(그림 5) G-Blender 1.5로 제작된 이노디스의 '평이야'

다(그림 3 참조).

### 3. 제작 중인 게임 소개

G-Blender 1.0 또는 이전 버전으로 상용화된 게임들이 다양한 장르에 걸쳐 있고, 현재 버전인 1.6으로도 다양한 상용 게임들이 만들어지고 있으며 2006년 상반기 이후 많은 게임들이 출시될 것으로 기대된다. 그 중 몇 가지 작품을 소개하면 다음과 같다.

#### 3.1 사이닉 소프트 '풍류공작소'

풍류공작소는 실사 3D와 카툰 3D로 양분되어 있는 MMOPRG에 새로운 애니메이션 튠 3D

를 구현해 전혀 새로운 그래픽을 보여준다. 지금까지 카툰 렌더링 방식을 사용한 게임을 보면 SD 캐릭터의 모습을 띄거나 만화같은 이미지를 사용하고 있는데, 풍류공작소는 카툰 렌더링 방식으로 제작하지만 극장용 애니메이션 같은 느낌을 주고 있다. 이는 2005년 하반기 우수게임 공모전 대상을 수상한 바 있다(그림 4 참조).

#### 3.2 이노디스의 '평이야'

'평이야'는 폭탄을 던져서 상대를 맞으면 이기는 방식으로 간단한 조작과 진행을 채택하여 누구나 손쉽게 게임을 즐길 수 있는 캐주얼 온라인 게임이다. 캐주얼의 목적에 맞게 저 사양 컴퓨터에서도 잘 구동될 수 있도록 카툰 형식의

그래픽에 아기자기한 게임성을 접목시켰다. 현재 [www.pung2ya.co.kr](http://www.pung2ya.co.kr)에서 서비스 중이다(그림 5 참조).

위에 소개된 게임 외에도 ㈜이젠에서 제작 중인 온라인 뮤지컬 게임 ‘아프로 비스킷’과 캐주얼 축구게임 ‘레드카드’ 등이 2006년 상반기에 상용 서비스에 들어갈 예정이다. 2005년 현재 G-Blender를 사용하고 있는 업체가 40여 곳에 이르는 만큼 2006년에는 G-Blender로 제작된 많은 게임 타이틀이 출시될 것으로 보인다. 또한 2006년부터는 학생들을 대상으로 한 강의에 맞도록 엔진을 재구성하고 교육을 위한 체계적 샘플들을 제공하는 등 교육 지원 시스템을 갖춘 아카데미 전용 G-Blender를 발표할 예정이다.

## 지식약력



현 호 진

1993년~2003년 동국대학교 산업공학과(학사)  
 1999년~2000년 (주)유니아나 3D 엔진 개발팀장  
 2000년~현재 (주)가이블 개발 이사  
 2002년~현재 세종 사이버 대학교 게임 프로그램 3 강의  
 관심분야: 웹 3D, 3D 게임 엔진, 컴퓨터 그래픽스  
 디자인 패턴  
 e-mail : [hjhyun@gible.com](mailto:hjhyun@gible.com)