



온라인 3D 게임엔진 표준화*

이헌주*, 김준애**, 임충규***, 김현빈****

● 목 차 ●

1. 서 론
2. 온라인 3D 게임엔진 개발 표준권고안
3. 결 론

1. 서 론

국내 온라인 게임 산업은 세계 최고 수준의 IT 인프라에 힘입어 최근 몇 년간 급속한 성장을 하였다. 또한 실시간 컴퓨터 그래픽 기술의 발전과 하드웨어의 성능 향상으로 몰입감을 증대시키는 3차원 게임의 구현이 가능하게 되어 3D 기반의 온라인 게임이 주류를 이루게 되었다.

미국, 일본 등 게임선진국은 축적된 컴퓨터 그래픽 기술, 3D 애니메이션 기술, 게임 관련 기술 등을 기반으로 하여 끊임없이 사용자들의 요구를 반영하며 게임엔진 기술을 발전시켜가고 있다. 또한 업체들은 개발된 게임엔진 기술을 도입하여 게임을 개발하고 있으므로 수준 높은 콘텐츠를 제작할 수 있으며 세계시장을 장악할 수 있는 바탕이 되고 있다. 국내의 경우는 게임엔진 개발에 대한 의욕은 높은 반면에 관련 전문 인력이 부족하고 업체 규모가 영세하다. 또한 축적된 기술에 대해서 공개를

꺼려하고 있어 기술공유가 이루어지고 있지 않으며 이러한 상황은 국내 게임 기술의 발전 저해로 이어지고 있다. 따라서 대부분의 업체에서는 많은 시행착오와 함께 게임엔진과 온라인게임에 필요한 기술력 보유를 위해 과중한 인력과 자금을 투입하거나 외국 엔진을 고가의 라이선스 비용과 로열티를 지불하고 있으며 이는 게임 콘텐츠의 질적 수준을 향상시키는데 걸림돌이 되고 있다.

이러한 고비용 저효율의 게임 개발 구조를 바꾸기 위하여 국내 게임업계가 공통으로 활용할 수 있는 온라인 3D 게임엔진의 표준권고안을 만들어 기술을 공유할 수 있는 기반을 제공하여야 한다. 본고는 기존의 온라인 3D 게임 및 게임 엔진을 조사하여 비교 분석함으로써 엔진의 표준 기능에 대해 연구한다. 이를 통해 제시된 표준 권고안은 게임업계에 활용할 수 있는 차세대 온라인 게임의 아키텍처를 제시하여 기술적 발전을 이루고 게임 개발의 효율성을 증대시켜 세계시장에서 경쟁력을 지닐 수 있는 완성도 높은 게임 개발의 기초가 될 것이다. 또한 가능성 있는 국내 온라인 게임 개발 업체들의 근본적인 문제점을 해결해 줌으로써, 중소기업체들의 중복 및 과잉투자를 막고 국내의 온라인 3D 게임 산업을 발전시키는 바탕이 될 것이다.

† 본 고의 내용은 “온라인 3D 게임엔진 표준화 연구”의 최종연구보고서에서 발췌하였음.

* 한국전자통신연구원 선임연구원

** 한국전자통신연구원

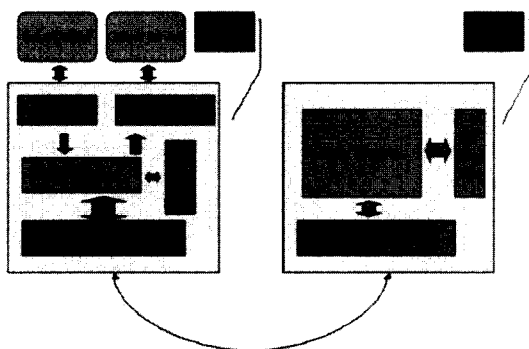
*** 한국전자통신연구원 팀장/선임연구원

**** 한국전자통신연구원 부장/책임연구원

2. 온라인 3D 게임엔진 개발 표준권고안

본 장에서는 온라인 3D 게임 엔진의 개발시 포함시켜야 할 기능과 내용에 대한 표준권고안을 기술한다. 엔진을 각 모듈별로 구분함에 있어서 물리적으로 크게 클라이언트에 관련된 부분과 서버에 관련된 부분으로 나눌 수 있다. 그러나 어떠한 엔진들은 클라이언트와 서버 양쪽에 모두 포함될 수도 있다. 이러한 요소들을 어떻게 구분짓는가 보다도 각 엔진의 역할이 명확하게 구분되어야 하고, 엔진간의 인터페이스 규격이 하나의 콘텐츠에 종속적이지 말아야 한다. 또한 가장 기본적인 인터페이스 규격을 제시하고, 그 엔진을 가지고 개발하는 개발자가 필요한 요소들을 쉽게 추가할 수 있는 인터페이스가 제공되어야 한다.

(그림 1)은 클라이언트와 서버의 엔진 구성도를 나타낸다. 사용자와의 인터페이스에 관련된 부분들은 인터페이스 장치와 실제 게임엔진 사이에 위치하고, 네트워크 엔진은 클라이언트와 서버 사이에서의 인터페이스를 제공한다. 인공지능 엔진은 클라이언트와 서버 양쪽 모두에 존재하게 된다.



(그림 1) 클라이언트와 서버의 게임엔진 구성도

2.1 네트워크 엔진

네트워크 엔진의 설계 시에 고려되어야 할 사항들은 다음과 같다.

가. 안정적인 데이터 전송 보장과 완벽한 예외처리

- (1) TCP/IP 프로토콜 슈트에서 제공하지 못하는 예외처리 루틴 포함
- (2) 모든 에러를 분류 처리하여 응용은 통보만 받음

나. 온라인 게임에 적합한 응용 계층 설계

- (1) 바이트 스트림을 메시지 스트림으로 변환
- (2) 메시지 스트림을 구현할 수 있는 응용계층을 설계

다. 네트워크 엔진의 모듈화

- (1) 핵심적인 루틴만으로 모듈을 구성하여 재사용성을 활용
- (2) 응용에서 직접적으로 네트워크 에러를 인지하지 않도록 한다.

라. 네트워크 엔진 성능 최적화

- (1) 데이터베이스에 대한 접근 전략 수립
- (2) 버퍼 복사의 최소화
- (3) 최소한의 파이프라인 사용
- (4) 1유저 1스레드 모델은 오버헤드가 큼
- (5) 멀티스레드 구조

마. 대칭 및 비대칭 서버구조에 적합한 인터페이스 모듈 설계

- (1) 각각의 호스트 간의 통신으로 인한 오버헤드를 최소화
- (2) 전체적인 서버 시스템의 물리적 구성

2.2 서버 엔진

게임을 위한 서버는 연결형 서비스이다. 많은 사용자와 지속적으로 연결이 되어 서비스를 제공하며 트래픽이 발생된다. 또한 사용자들의 공유데이터에 대한 접근이 빈번하며, 이러한 공유데이터들

은 대부분이 메모리 상주 DB와 같은 데이터들이다. 그리고 읽기만이 발생하는 것이 아니라 쓰기도 지속적으로 발생되므로 효율적인 접근방법을 이용하지 않으면 이로 인하여 발생하는 시스템 전체에 대한 병목현상으로 시스템의 효율을 저하시키게 될 것이다.

가. 서버 운영자 인터페이스 설계

- (1) 슬립(Sleep) 타임이 없도록 설계
- (2) 에러나 장애에 즉각 대응
- (3) 에러를 즉각적으로 운영자가 인지할 수 있도록 모니터링 시스템이 필요

나. 멀티 서버 시스템에서 네트워크 가상세계 구현을 위한 분산 시스템 전략 및 동기화 정책

- (1) 각 노드의 명확한 역할 분담과 공동작업에 대한 상세 설계가 필요
- (2) 가상 세계를 운영할 범용 동기화 전략 수립
- (3) 콘텐츠에 따른 기획부분의 제한요소와 기술적 제한요소를 고려한 적절한 동기화 전략 수립
- (4) 다양한 사용자의 네트워크 지연을 보상할 수 있는 알고리즘

다. 보안 정책

- (1) 방화벽 사용
- (2) 권한 관리와 시스템에 대한 물리적인 보안의 강화

2.3 3D 그래픽 엔진

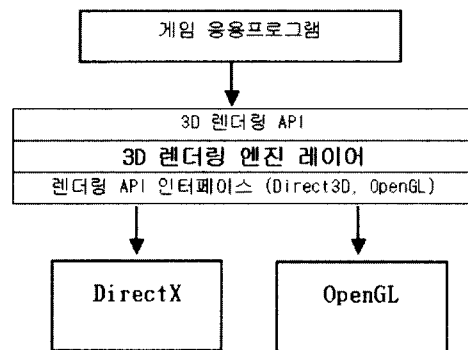
클라이언트 시스템에 포함되는 엔진으로 화면에 3D 그래픽을 연출한다. 캐릭터나 NPC 등 3차원으로 모델링한 물체와 배경의 3D 표현을 담당하는 3D 렌더링 엔진과 3D 물체의 이동 및 충격에 따른 변화 등의 움직임을 담당하는 3D 애니메이션 엔진으로 구별하여 포함될 수 있는 기능 및 내용을 기

술한다.

2.3.1 3D 렌더링 엔진

가. 3D 렌더링 엔진의 구조

본 고에서의 3D 렌더링 엔진에서는 3D 온라인 RPG 게임에 적합한 구성 요소들을 기본으로 하며, 구조는 (그림 2)와 같이 위쪽으로는 게임 응용프로그램에서 호출할 수 있는 3D 렌더링 API 셋을 포함하며, 아래쪽으로는 기존의 3D 렌더링 API 셋을 호출하는 구조로 이루어진다.



(그림 2) 3D 렌더링 엔진 레이어 구조

나. 3D 렌더링 엔진의 기능

다음은 렌더링 엔진이 포함할 수 있는 기능들을 나열하였다.

① 기본 자료 구조

- Point
- vector
- color
- Line, Line Strip, Line Loop.
- Triangle, Triangle Strip, Triangle Fan
- Quad, Quad Strip
- polygon

② math 관련 기능

- vector 관련 기능

- matrix 관련 기능
- quaternion
- ③ shading
 - Flat Shading
 - Gouraud Shading
 - Phong Shading
- ④ Texture Mapping
 - Texture Mapping
 - Multitexturing
 - Bump Mapping
 - Displacement Mapping
 - Environment Mapping
 - Sphere Mapping
 - Cube Environment Mapping
- ⑤ Lighting
 - Lighting
 - Point light
 - Directional Light
 - Spot Light
 - Ambient Light
 - Projective Texturing
 - Shadow 기법
- ⑥ 특수 효과
 - Fog
 - Particle System
 - Sprite
 - Billboard
 - vertex animation
- ⑦ LOD
 - static
 - dynamic (continuous)
- ⑧ Advanced DS
 - BSP
 - PVS
 - Height Field
- ⑨ 고급 shader

- vertex shader
- pixel shader
- ⑩ Blurring
 - Focal Blur
 - Motion Blur

2.3.2 3D 애니메이션 엔진

가. 개요

3D 애니메이션 엔진은 렌더링된 3D 객체들의 움직임이나, 충돌 등에 의해 변형되는 움직임 등과 관련된 기능을 담당한다.

나. 3D 애니메이션 엔진의 기능

다음은 애니메이션 엔진이 포함할 수 있는 기능들을 나열하였다

① 캐릭터 애니메이션

- 본 애니메이션(bone animation)
- 키프레임 애니메이션
- 스킨링 애니메이션 (skinning animation)
- 스켈리털 애니메이션(skeletal animation)
- 얼굴표정 애니메이션 (Facial expression)
- 충돌검사 (Collision Detection)

② 특수효과(플러그인)

- Character Studio
 - Rigid : Vertex의 가중치가 적용되지 않은 Bone Animation 데이터를 추출
 - Blend : Vertex의 가중치가 적용된 자연스러운 Bone Animation 데이터를 추출
- 3D Studio MAX
 - Sub-material : 한 Object에 여러 재질로 된 데이터를 추출
 - Camera : 카메라가 움직인 좌표를 추출
 - Smooth Group : 부드러운 면으로 구성된 그룹을 분리
- 3D Studio MAX Plug-in

- Map : BSP를 이용하기 위한 Map 데이터를 추출
- Object : Map 상에 반복적으로 사용될 Object 데이터를 추출
- Character : Character Studio를 통해 설정된 bone 및 vertex weight 데이터를 추출

2.4 3D 사운드 엔진

2.4.1 개요

3D 사운드 엔진이란 3D의 현장감있는 음향을 구현하기 위한 시스템이나 기술을 말한다. 과거의 3D 사운드는 다수의 스피커를 이용하여 각 스피커마다 고유의 소리를 출력하는 방법으로 입체음을 구현했지만, 최근에는 하드웨어적인 디지털 가속 기술로 2개의 스피커에서도 입체음을 낼 수 있다.

2.4.2 3D 사운드 엔진의 기능

가. Multi-Channel 지원

여러 웨이브 소스를 동시에 출력할 수 있음은 물론이고, 각각의 소스를 출력할 스피커를 따로따로 결정할 수 있는 '멀티클라이언트' 기능까지 지원한다.

나. 3D Sound 지원

3D 사운드는 소리의 주파수를 분리하거나 빠르게 좌우 스피커 출력을 바꾸는 방법으로 입체음을 구현한다.

다. Special Effect 구현

① Volume

일반적인 음량의 조절을 통한 Fade In, Fade Out을 구현할 수 있다.

② Panning

소리의 좌우구분을 정의하며 펜을 통하여 좌우의 밸런스를 맞출 수 있다. 역동적인 Panning의 제어로 입체감있는 스테레오 사운

드를 만든다.

③ Doppler

파원에 대하여 상대속도를 가진 관측자에게 파동의 주파수가 파원에서 나온 수치와는 다르게 관측되는 현상으로 3D게임내의 객체들의 이동에 따라 변화하는 사운드를 만들어낸다.

④ Reverb

어떤 공간안에서 음향적 성질을 만들어내는 반복되는 에코로 인공적인 잔향이 전자장치를 통한 신호처리로 만들어진다.

⑤ RealTime Filtering

객체가 되는 사운드를 실시간으로 변화시켜 출력하는 루틴

2.5 인공지능 엔진

2.5.1 요구사항

3D 렌더링 엔진과 달리 인공지능 기능은 장르에 따라 다른 요구사항을 갖고 있다. 따라서 장르별 요구사항을 정리하고 이들의 공통적인 기능을 확인하도록 한다. 이 엔진은 MMORPG 장르를 기본으로 하며 액션, RPG, RTS의 세가지 장르로 확대 사용할 수 있도록 한다. 각 장르에 따른 일반적인 사항을 정리하면 <표 1>과 같다.

<표 1> 장르에 따른 인공지능 대상

| 장르 | MMORPG | 액션 | 일반 RPG | RTS |
|--------|--------------------------------------|--|--|--|
| 대상 캐릭터 | Monster NPC | Enemy Monster | Partner Monster NPC | Enemy Group Unit |
| 처리 내용 | Behavior Dialog | Behavior | Behavior Dialog | Goal Plan Pathfinding Behavior |
| 게임의 예 | Ultima Online Ashron's Call | Quake Unreal Tournament | Baldur's Quest | Starcraft Age of Empire II |
| 구현 방법 | FSM Decision Tree Script 언어 | FSM Decision Tree Script 언 어(FSM 기능) | FSM Decision Tree Script 언어 (Rule Based System) | FSM Decision Tree Script 언어 (Planning) |

가. MMORPG

MMORPG를 제외한 다른 장르는 Enemy 역할을 인공지능이 담당하여야 하기 때문에 인공지능이 없이는 게임을 구현할 수 없다. 이에 비해 MMORPG의 경우에는 monster 는 quest의 대상일 경우가 많으며 따라서 RTS와 같은 plan 기능을 요구하지는 않는다. 그러나 이를 제외한 나머지 장르의 경우에는 인공지능이 Enemy 의 역할을 대신 수행하여야 한다. 일반적으로 MMORPG의 인공지능 기능은 RPG의 기능과 유사하게 처리할 수 있다.

나. 액션

Unreal Tournament나 Quake Arena와 같은 사용자 간의 액션게임의 경우 Monster 를 사용하지 않기 때문에 인공지능 기능은 불필요하다. 그러나 monster가 등장하는 게임의 경우에 이들 monster의 지능은 게임 엔진에서 담당하여야 한다. 다양한 monster를 등장시키기 위해서는 monster 간의 class 상속이 있어 인공지능 기능을 상속받을 수 있도록 한다. 또한 monster는 여러 장소를 이동함에 따라 월드에 대한 정보를 기록할 수 있는 메모리를 필요로 한다. 액션게임의 인공지능 요구사항을 정리하면 다음과 같다.

- FSM 지원
- 빠른 처리
- 캐릭터 상속 지원
- Pathfinding 지원
- Flocking 지원

다. RPG

RPG게임의 경우 게임 world에 등장하는 몬스터나 플레이어와 동행하는 Partner, 그리고 게임의 현실감을 제공하는 NPC의 행동은 인공지능 기능이 처리하여야 한다. RPG 게임의 인공지능 요구사항을 정리하면 다음과 같다.

- 스크립트 언어

- 캐릭터의 스크립트 상속
- 대화기능의 스크립트 처리

라. RTS

RTS의 경우에는 Enemy를 대신하는 전략적 목표를 설정하는 기능과 일시적으로 선택된 그룹의 전술적인 행동, 그리고 그룹에 속하는 개별 유닛의 단위 시간의 행동을 담당하는 기능의 인공지능이 필요하게 된다. 따라서 RTS 게임의 인공지능에서는 게임에서 플레이어가 활용할 수 있는 자원을 어떻게 할당을 결정하는 것이 중요한 문제이다. RTS 게임의 인공지능 요구사항을 정리하면 다음과 같다.

- 스크립트 언어
- Pathfinding
- Influence Mapping

2.5.2 인공지능 엔진 아키텍처

앞서 설명한 바와 같이 게임의 장르에 따라 인공지능 구현방법은 큰 차이가 난다. 따라서 엔진에는 FSM과 Rule-based System을 선택적으로 사용할 수 있다. FSM은 기본적인 C++ 코드를 사용하여 구현할 수 있으며 Rule-based System은 스크립트 언어를 사용할 수 있다.

가. FSM 사용

인공지능은 게임 플레이와 밀접한 관계를 갖고 있으며, 단순한 FSM 지원을 위해서는 switch 문장만으로 가능하기 때문에 FSM을 구현하기 위한 별도의 API는 필요하지 않다.

나. 스크립트 인터프리터 사용

스크립트 인터프리터를 사용하기 위해서는 C++ 언어에 스크립트 프로그램과 연결하는 장치를 마련하여야 한다. 이것은 기본적으로 스크립트 언어와 사용되는 변수의 값을 주고 받는 장치, 해당되

는 루틴을 call할 수 있는 장치를 만들면 가능하다.

3. 결론

본 고에서는 온라인 3D 게임엔진의 개발시 포함 시켜야 할 기능과 내용에 대한 표준권고안을 알아 보았다. 본 고에서 기술한 온라인 3D 게임엔진 표준권고안은 업계와 학계의 의견을 받아 계속 보완, 개선한다면 국내 온라인 3D 게임기술의 발전과 함께 다음과 같은 효과를 기대할 수 있을 것이다.

- 게임 개발기간 단축, 개발인력 절감 및 개발비 감소
- 게임 개발사의 콘텐츠 개발 노력 증대
- 국내 온라인 3D 게임산업의 개발 저변 확대
- 국내 온라인 3D 게임엔진의 세계 표준화
- 개방형 표준규격 연구

본 고에서 기술된 권고안은 앞으로도 온라인 게임기술의 여러 부분에서 지속적으로 표준화 작업이 진행되어야 한다. 특히 게임엔진의 표준화는 온라인 게임의 핵심이 되는 표준화이다. 효과적으로 표준화를 이루기 위해서는 국내 뿐만 아니라 국제적 표준화 포럼에 적극적으로 참가하여 국내에서 개발이 이루어진 결과를 국제표준으로 채택되도록 공동으로 노력해야 한다.

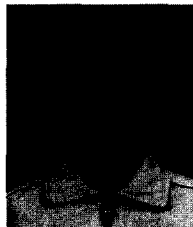
온라인 게임분야는 게임엔진의 요소를 근간으로 하는 콘텐츠 등 표준화 서브그룹을 구성하여 해당 분야의 전문가들이 모여서 전문적인 서브그룹 표준화 작업을 지속적으로 진행해야 한다. 또한, 표준화를 통하여 도출된 권고안에 따라서 개방형 표준규격을 개발하여 게임업체들의 다양한 제품간에 상호운용성을 갖도록 시험하여야 한다. 국책연구기관은 개방형 표준규격 개발을 주도하고 게임개발업체들과 표준화에 따라 공동개발하거나 제품의 상호운용성을 시험할 수 있는 시험환경을 제공하

는 것이 바람직하다. 또한 개발된 온라인게임의 상호운용성을 시험하고 인증해주는 기관이 필요하다. 국내 게임업체들도 표준권고안에 따라서 적극적으로 참여해야겠지만, 산업체에서 추진하기 어려운 표준화 작업은 국가연구소에서 개발되고 있는 게임엔진이 국제표준이 될 수 있도록 하는 체계적인 연구와 지원제도가 필요할 것이다.

참고문헌

- [1] 가상현실연구부, 온라인 3D 게임엔진 표준화 연구, 최종연구보고서, 2001.11, 한국전자통신연구원.
- [2] 김병철, "3D 엔진이란 무엇인가?", 2001 추계 한국 게임개발자 협의회 Conference Proceeding, pp. 252-278, 2001. 11.

저자약력



이 현 주

1991년 중앙대학교 컴퓨터공학과 (공학사)
1993년 중앙대학교 컴퓨터공학과 (공학석사)
1998년 중앙대학교 컴퓨터공학과 (공학박사)
2001년-2002년 Iowa State University (Post-Doc.)
1998년-현재 한국전자통신연구원 (선임연구원)
관심분야: 인공지능, 가상현실, 3D 게임엔진
e-mail: hjoo@etri.re.kr



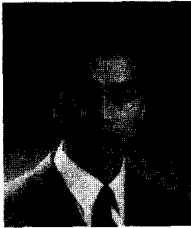
김 준 애

2000년 이화여자대학교 (컴퓨터학 전공)
2002년 포항공과대학교 대학원 공학석사 (컴퓨터공학, 가상현실 전공)
2002년-현재 한국전자통신연구원
관심분야: 가상현실, 3D 게임엔진
e-mail : ckjay@etri.re.kr



김 현 빈

1985년 중앙대학교 학사 (응용통계학 전공)
1988년 중앙대학교 대학원 석사 (응용통계학 전공)
1996년 Okayama Univ. 대학원 박사 (전산통계학 전공)
1991년-1993년 Nagoya Univ. (연구원)
1984년-현재 한국전자통신연구원 (부장/책임연구원)
관심분야: 가상현실, 게임엔진, HCI
e-mail : hbkim@etri.re.kr



임 충 규

1988년 서울대학교 수학교육과 (이학사)
1990년 서울대학교 수학교육과 (교육학석사)
1998년 미국 루이지애나 주립대 (Ph.D)
1990년-1992년 현대전자 (S/W엔지니어)
1999년-현재 한국전자통신연구원 (팀장/선임연구원)
관심분야: 게임엔진, 기하 모델링, 가상현실
e-mail : cglim@etri.re.kr