

호서 3D 온라인 게임엔진을 이용한 3D 네트워크 게임개발에 관한 연구

오정현*, 박우승, 강중호, 최삼하, 김경식
호서대대학원 게임공학전공

A Study on the Development of a 3D Network Game using Hoseo 3D Online Game Engine

Jeong-Hun Oh*, Woo-Seung Park, Jong-Ho Kang, Sam-Ha Choi, Kyung-Sik Kim
Dept. of Game Engineering, Graduate school of Hoseo University

요 약

호서 3D 온라인 게임 엔진은 렌더링등의 기능을 담당하는 3D 엔진(클라이언트 부분)과 네트워크 기능을 담당하는 서버 엔진(서버 부분)으로 구성되어 있다. 본 논문은 호서 3D 온라인 게임 엔진을 이용한 3D Tank 데모 게임을 제작하기 위해 일반적인 게임 진행 루프를 분석한 후, 분석 결과를 게임의 목적에 따라 설계하였다. 또한, 설계에 따른 실제 3D Tank 데모 게임을 개발하였으며 이를 바탕으로 3D 온라인 게임 엔진 요소의 효율적인 통합에 대해 고찰한다.

1. 서론

일반적으로 게임 엔진이란 컴퓨터상에서 게임을 실행시키는 프로그램들의 모음 혹은 핵심 코드라고 볼 수 있으며, 온라인 3D 게임 엔진이란 온라인 3D 게임을 실행시키는 프로그램들의 모음으로써 일반적인 게임 엔진의 기능에서 네트워크를 통하여 사용자들을 연결시키는 네트워크 인터페이스를 포함한다[1,2].

좋은 성능을 가진 인정받은 게임 엔진이 있다고 하더라도 그 엔진내의 각 세부 모듈들을 통합하여 하나의 콘텐츠로 만들기 위해서는 단순히 엔진을 조합하기만 하면 되는 것이 아니라 게임 개발 프로젝트 전체에 대한 이해가 필요하며, 이를 위해서는 엔진에 대해 이해하고 정리하며 기획에 맞게 설계한 후 게임 진행 루프를 목적에 따라서 설계하는 방법이 필요하다[3]. 본 논문에서는 '호서 3D 온라인 게임 엔진'을 이용한 3D Tank 데모 게임을 제작하기 위해 일반적인 게임 진행 루프를 분석한 후, 분석 결과를 게임의 목적에 따라 설계한다. 그리고 설계에 따른 실제 3D

Tank 데모 게임을 개발에 대해 논의하고 이를 바탕으로 3D 온라인 게임 엔진 요소의 효율적인 통합에 대해 고찰한다. 본 논문의 구성은 다음과 같다. 먼저, 2장에서는 3D 게임 엔진 및 네트워크 게임 제작에 대한 관련연구를 살펴보고 3장에서는 3D Tank 게임 제작에 대해 설명하고, 4장에서는 게임 엔진 요소들의 효율적인 통합에 대해 고찰한다. 마지막으로 5장은 결론과 향후의 연구 과제를 제시한다.

2. 관련연구

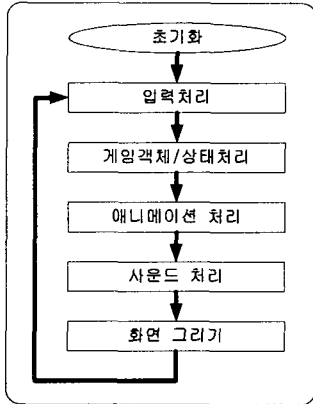
2.1 게임 진행 모듈

게임 진행 모듈은 개별 엔진을 유기적으로 통합하고, 각종 데이터를 효율적으로 관리하며 기획을 충실히 반영하여 콘텐츠를 개발하는데 필요한 모듈을 말한다[3]. 게임 진행 모듈은 크게 엔진의 통합과 게임 객체의 관리로 나누어서 설명할 수 있는데, 엔진의 통합은 각 세부 엔진을 전체적으로 통합하는 방법을 의미하며, 게임 객체 관리는 게임 내의 객체를 효율적으로 관리하는 것을 의미한다.

1 본 연구는 한국과학재단 목적기초연구(과제명: 3D 온라인 게임엔진 요소기술 연구, 과제번호: R05-2000-000-00288-0) 연구비에 의해 연구되었음

2.2 일반적인 게임 진행 루프(Loop)

게임 진행 루프는 게임을 진행할 때에 처리되는 과정으로 계속 반복된다고 해서 진행 루프라고 부르거나, 매번 새로운 프레임 단위로 갱신하므로 프레임(Frame)이라고 부르기도 한다[3]. [그림 1]은 게임 진행 루프 과정이다.



[그림 1] 일반적인 게임진행루프

2.3 일반적인 네트워크 게임 제작

클라이언트와 서버는 서로 지속적으로 통신하면서 플레이어(클라이언트)가 게임의 상태에 변화가 있을 때마다 메시지를 서버로 전송하며, 서버는 프레임 시간 동기화(synchronization)를 유지하기 위해 게임의 상태를 가져와 게임 세계를 갱신(update)한 뒤, 클라이언트에게 게임 갱신 정보를 전송한다[4,8].

2.3.1 서버(Server)

서버는 실행과 동시에 루프에 들어가 수신되는 네트워크 메시지를 지속적으로 처리하고 연결된 모든 플레이어들의 데이터를 갱신한 뒤 갱신된 정보를 전송한다.

2.3.2 클라이언트(Client)

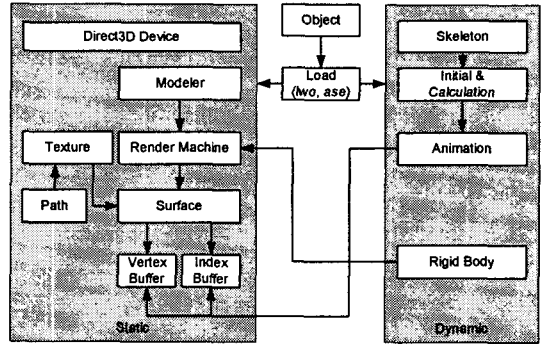
서버에 접속한 클라이언트는 로컬(local) 플레이어 컨트롤을 모아 서버에 전송하고, 서버로부터 갱신되어 전송된 메시지를 수신하여 처리한다.

3. 3D TANK 게임의 제작

본 연구에서 3D Tank 데모 게임은 호서 온라인 3D 엔진을 이용하였으며, 계층적 구조 애니메이션[6]과 물리, 역학 등을 포함한 애니메이션 부분을 중점

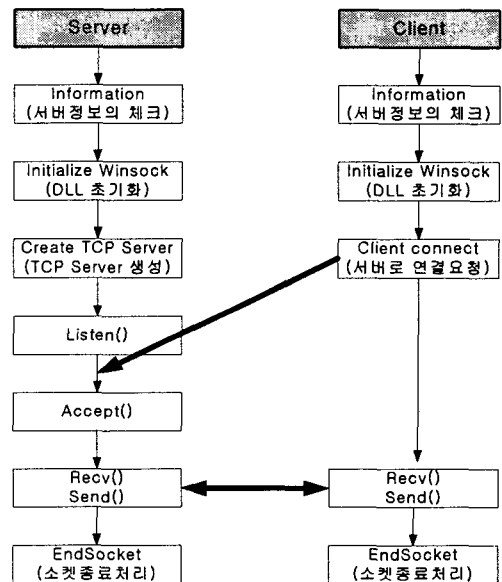
로 개발되었다. 본 엔진은 기본적인 3D 렌더링 엔진을 기반으로 캐주얼한 3D 액션 게임이나 간단한 시뮬레이션 게임 개발이 가능한 엔진이다[5].

3.1 호서 3D 온라인 게임 엔진



[그림 2] 호서 3D 엔진의 구조

[그림 2]는 호서 3D 엔진을 나타낸 그림으로 3D 객체(object)를 읽어 들여 정적 객체 부분과 동적 객체 부분으로 분리하여 처리한다. 정적 객체는 렌더링을 위한 데이터만을 가지는 데이터이며, 동적 객체는 애니메이션에 관련된 데이터만을 가지는 객체이다. 이 두 객체를 결합하여 애니메이션이 가능하다.



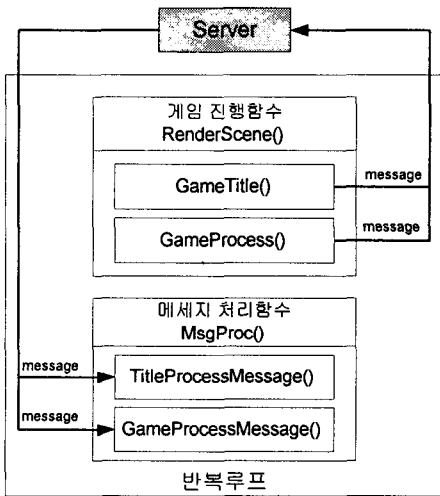
[그림 3] 호서 서버/클라이언트 엔진의 기능도

Initialize Winsock은 Winsock과 관련되어 있는

DLL파일을 초기화하는 역할을 하고 클라이언트에서 접속을 하면 Client connect에서는 서버로의 연결 요청을 하고 서버의 Accept부분에서 연결요청을 처리한다. Recv()/Send() 함수는 메시지의 송수신을 담당한다.

3.2 3D Tank 게임의 게임 진행 루프

3D Tank 게임은 소규모 네트워크 게임[7]이며, [그림 4]와 같이 메시지를 처리하는 부분인 MsgProc() 부분과 게임 진행 함수인 RenderScene() 부분으로 나누어서 게임진행 루프를 구성하였다.



[그림 4] 3D Tank 게임의 게임 진행 루프

반복적으로 게임 진행 함수인 RenderScene() 함수가 호출되면서 함수안의 각 세부 함수에서 메시지가 있으면 그것을 서버로 보내고 서버에는 이것을 다시 메시지 처리함수인 MsgProc()함수로 보내서 세부 함수별로 처리하게 된다.

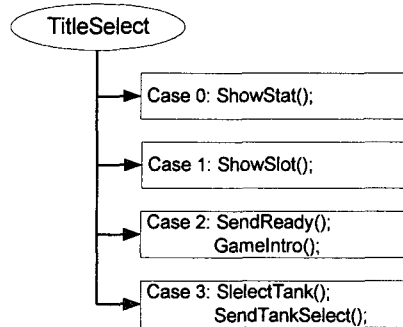
3.2.1 RenderScene() 함수

RenderScene() 함수의 구조를 보면 [그림 4]와 같이 GameTitle() 함수와 GameProcess() 함수 부분으로 나누어지며, 각 함수의 기능은 [표 1]과 같다.

함수명	기능
GameTitle()	기본적인 로그인 부분과 게임 시작/종료, 게임옵션을 선택하는 부분을 처리
GameProcess()	실질적인 게임 부분을 처리

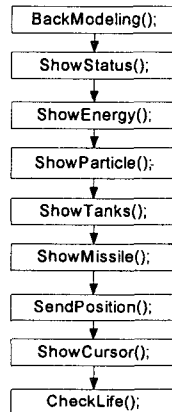
[표 1] 세부 함수들의 기능

GameTitle() 함수의 구성은 [그림 5]와 같으며, TitleSelect값에 따라 네 부분으로 나누어진다. Title Select값이 0 또는 1일 경우 플레이어의 능력치를 선택하는 부분과 보여주는 부분이며, 2일 경우 게임으로 들어가는 부분이다. 3일 경우에는 탱크유닛을 선택하는 부분이다.



[그림 5] GameTitle() 함수의 구성

실제 게임 부분을 담당하는 GameProcess() 함수의 구성은 [그림 6]과 같으며, 주요 함수의 기능은 [표 2]와 같다.



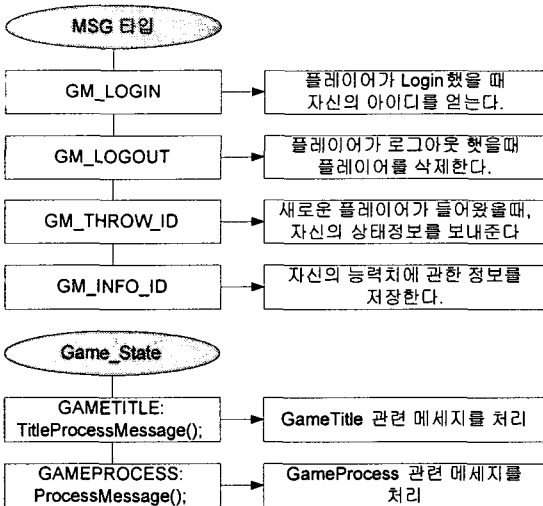
[그림 6] GameProcess() 함수의 구성도

함수명	기능
BackModeling()	배경맵의 출력을 담당, 순서대로 각각의 기능을 수행
SendPosition()	매 순간 자신의 위치정보를 서버에게 보내주는 역할
CheckLife()	플레이어의 탱크 life 변수를 체크하여 life 변수가 0일 경우, GameTitle()부분으로 돌아가게됨

[표 2] 세부 함수들의 기능

3.2.2 MsgProc() 함수

3D Tank 게임의 경우 갱신된 정보를 서버에게 보내고, 서버는 단순히 그 정보를 다시 모든 클라이언트에게 보내서 각 클라이언트가 갱신하는 단순한 구조로 되어있다. 모든 메시지 처리는 MsgProc() 함수에서 담당하며 구조는 [그림 7]과 같다.



[그림 7] MsgProc() 함수의 구성

MsgProc()함수는 두 부분으로 구성되어 있다. 기본적인 로그인/로그아웃 관련 및 자신의 정보를 보내주는 메시지 처리 부분과 GAME_STATE 값에 따라 GAMETITLE 부분과 GAMEPROCESS 부분의 메시지를 각각 처리하는 부분이다. 이때 클라이언트는 모든 정보를 갱신하게 된다.

4. 효율적인 엔진 요소의 통합에 관한 고찰

본 3D Tank 데모 게임은 호서 3D 엔진을 이용하여 제작한 간단한 네트워크 슈팅 게임이다. 호서 3D 엔진은 어떤 특정한 장르에 맞춰 특화된 엔진이 아닌 범용적인 엔진이라 할 수 있다. 따라서 실제 게임을 제작할 때는 엔진의 어떠한 요소를 어떻게 통합하여 쓸 것인가가 중요한 문제가 되었고, 실제 게임 제작에서 게임 진행 모듈의 구성은 이 데모 게임 제작의 핵심이 되었던 것이다. 본 논문에서 활용한 게임 진행 루프의 구성은 단순하며, 제작의 편의성을 고려한 구성으로 이루어져 있다. 실제로 대규모 사용자 3차원 대규모 멀티플레이어 온라인 게임(3D MMOG) 같은 큰 게임에서는 더욱 복잡한 구조가 필요하다[9]. 또한, 엔진을 활용하여 게임을 만들 때 장르별로 필요한 엔

진 구성 요소들을 모아서 각각의 게임 진행 모듈을 구성한다면 보다 효율적인 특화된 엔진으로서의 활용이 가능하다고 본다.

5. 결론 및 향후 연구과제

본 논문에서는 간단한 네트워크 3D Tank 게임 제작을 통해 게임 진행 모듈을 구성해 보았다. [그림 2]와 같은 일반적이고, 기능적인 구성하고는 달리 [그림 4]와 같이 게임 진행 루프를 서버/클라이언트간의 통신을 고려하여 게임 진행 부분과 메시지 처리부분으로 나누어서 구성하였다. 또한, 모든 게임에 필요한 엔진 기능들을 모듈화하여 효율적으로 관리할 수 있게 하였다. 그러나 실제 게임 제작에서는 장르와 엔진 구조방식에 따라 다르게 적용 가능하며, 특히 3D MMOG에서는 네트워크 메시지 전송의 효율성과 편리한 온라인 게임 월드 구성을 고려한 게임 진행 모듈 구성에 관한 연구가 필요하다.

향후에는 본 논문에서는 다루지 않은 게임 객체 관리 및 장르별 특화된 게임 진행 모듈 구성에 대해 연구가 진행되어야 할 것이다.

[참고문헌]

- [1] 김병철, "3D 게임 엔진이란 무엇인가?", 2001 추계 한국 게임 개발자 협의회 Conference Proceeding pp.252-278, 2001.11.
- [2] 김경식 외, 온라인 3D 게임엔진 표준화연구 최종 보고서, 한국전자통신연구원, 2001.11.
- [3] 정보통신부, GAME 엔진 개발론, 홍릉과학출판사, 2003
- [4] Jim Adams, Programming Role Playing Games with DirectX, Premier, 2002
- [5] 김경식 외, "3D 온라인 게임 엔진 요소 기술 연구", 한국게임학회 하계 학술대회 논문집, pp.207-213, 2002.7.
- [6] 김경식, "3D 온라인 게임을 위한 3D 렌더링 엔진 연구", 2002 한국게임개발자협의회 Conference Proceeding, 2002.11.
- [7] 최진성 외, "온라인 3D 게임 기술", 한국정보과학회지 제19권 제5호, pp.59-68, 2001.5.
- [8] 이만재, "온라인 게임엔진기술동향", 한국정보과학회지 제20권 제1호, pp.12-18, 2002.1.
- [9] 양광호, 심광현, 고동일, 박일규, 김종성, "온라인 게임 서버의 기술동향", 한국전자통신 동향 분석 제16권, 제4호, pp.14-22, 2001.8.