
M M O 2 D

TCPFighter

N A M E : 권 성 호

T E L : 010-8874-6452

P A R T : PROGRAMING

Efforts make all doors open



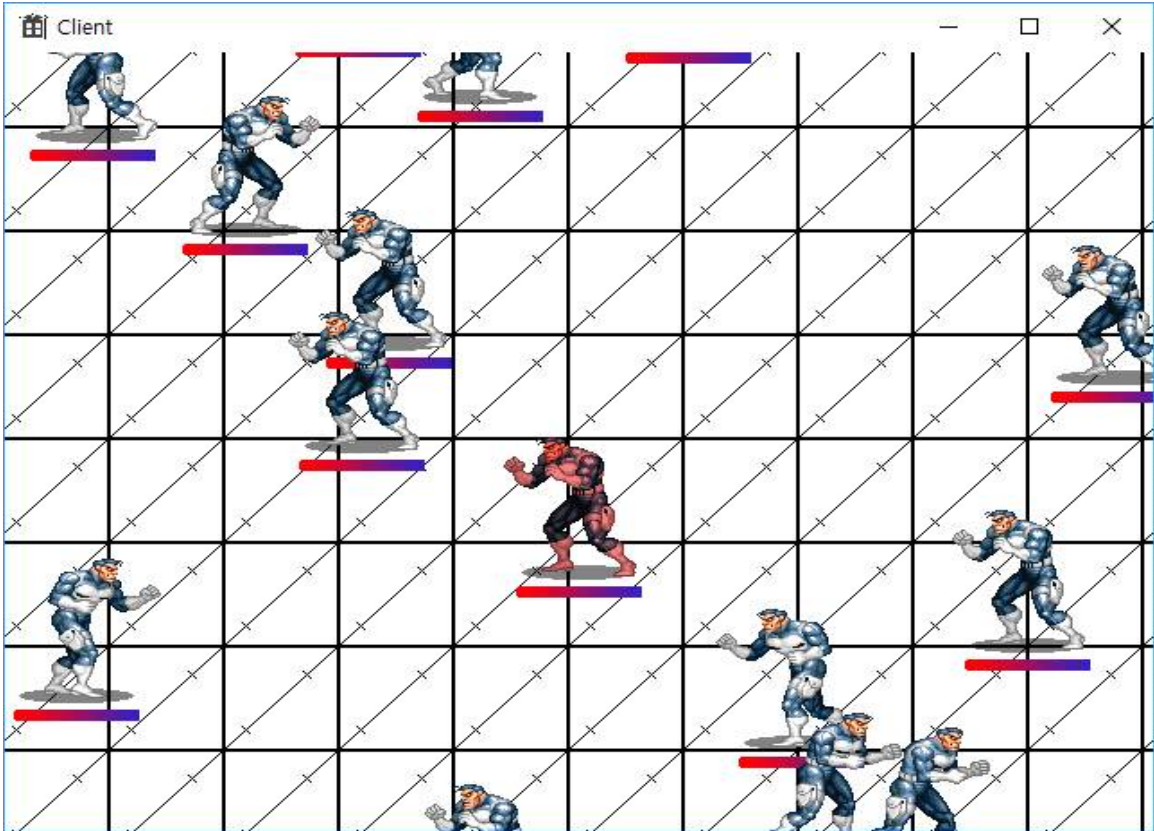
목 차



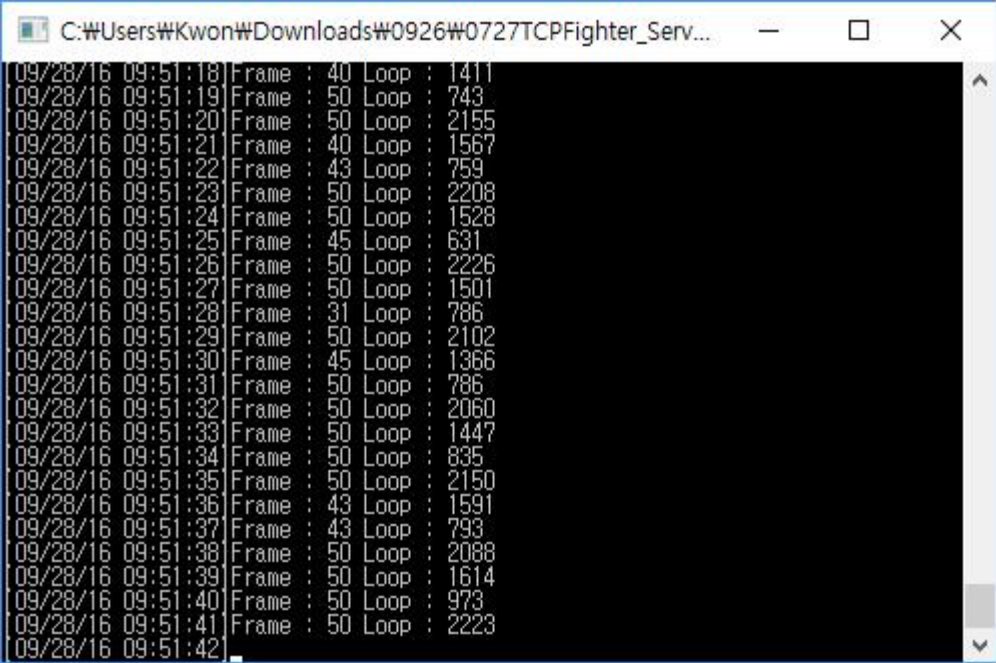
I. 프로그램 개요	1
1. TCPFighter Client	1
2. TCPFighter Server	2
3. Dummy Client	3
II. 기술요약 및 소스분석	4
1. Server 구조	4
2. Server 모듈설명	5
3. 링크	6

I. 포트폴리오 개요

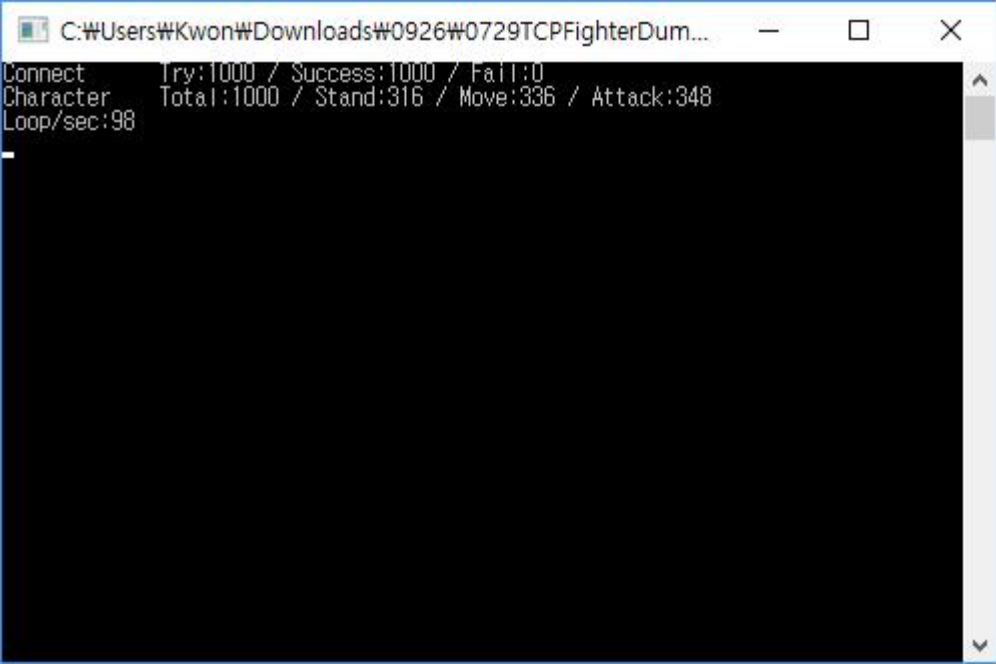
1. TCPFighter Client

프로그램명	TCPFighter Client
플랫폼	PC
게임화면	
	
소개	<ol style="list-style-type: none"> 1. WSAAsyncSelect로 제작 된 Client 2. Server의 응답에 따라 반응한다. 3. 클라이언트가 1000명 이상의 접속자의 패킷을 계속 주고받는 것은 매우 무겁기 때문에 플레이어의 중앙 섹터 포함 9개의 섹터영역의 클라이언트의 정보만 화면에 뿌려준다.(Server에서 관리)
개발 환경	WinAPI(C++)
소감	WinAPI로 제작된 Client이다. 엔진을 사용한 것이 아닌 게임의 로직과 렌더링부터 패킷 송수신 부까지 차근차근 만들었다. 프로그래밍에 자신감을 붙여준 좋은 녀석이다.

2. TCPFighter Server

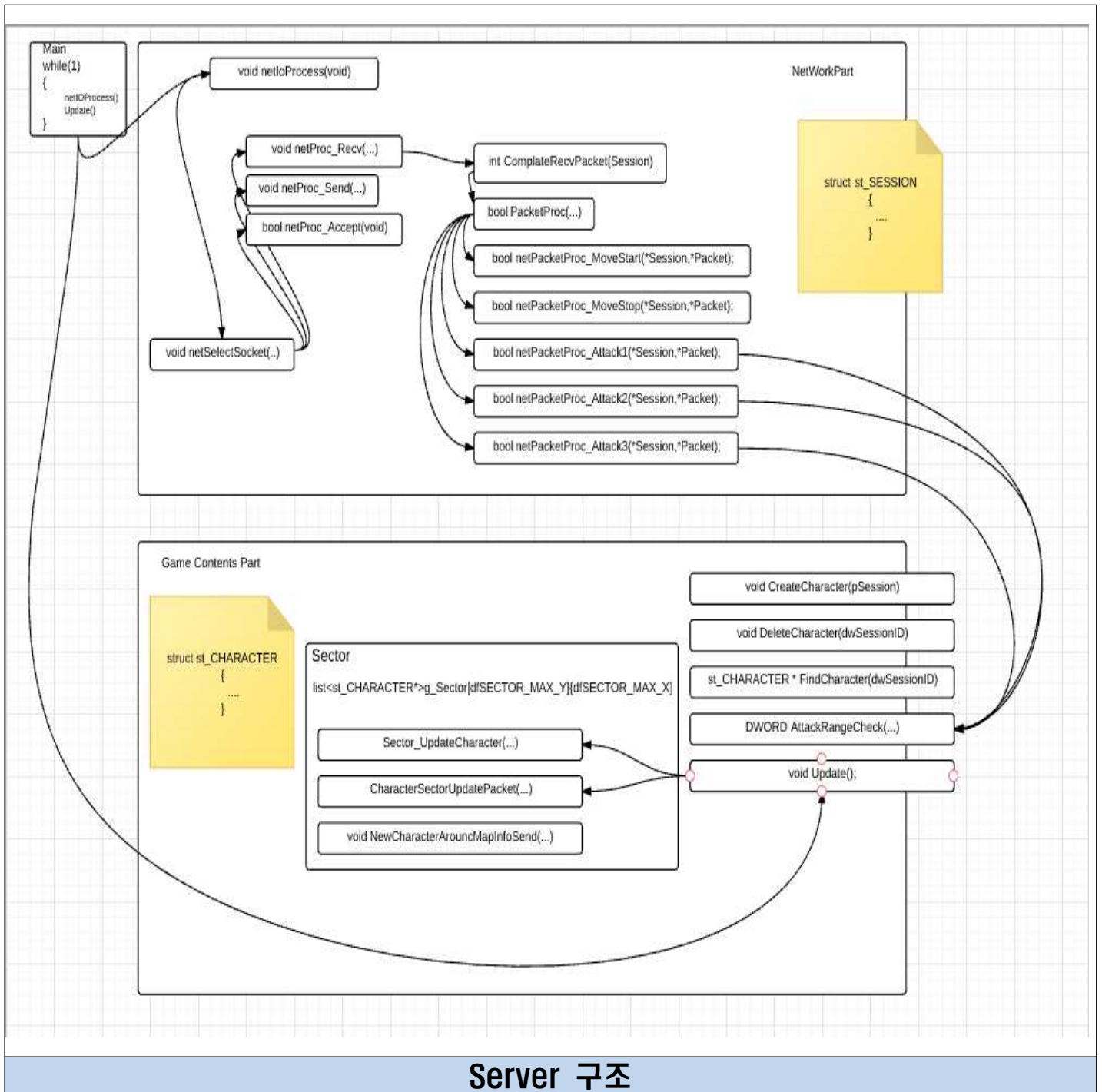
프로그램 명	TCPFighter Server
플랫폼	PC
게임화면	
 <p>The screenshot shows a Windows console window titled "C:\Users\Kwon\Downloads\0926\0727TCPFighter_Serv...". The console displays a log of network activity with timestamps and frame/loop counts. The log entries are as follows:</p> <pre> 09/28/16 09:51:18 Frame : 40 Loop : 1411 09/28/16 09:51:19 Frame : 50 Loop : 743 09/28/16 09:51:20 Frame : 50 Loop : 2155 09/28/16 09:51:21 Frame : 40 Loop : 1567 09/28/16 09:51:22 Frame : 43 Loop : 759 09/28/16 09:51:23 Frame : 50 Loop : 2208 09/28/16 09:51:24 Frame : 50 Loop : 1528 09/28/16 09:51:25 Frame : 45 Loop : 631 09/28/16 09:51:26 Frame : 50 Loop : 2226 09/28/16 09:51:27 Frame : 50 Loop : 1501 09/28/16 09:51:28 Frame : 31 Loop : 786 09/28/16 09:51:29 Frame : 50 Loop : 2102 09/28/16 09:51:30 Frame : 45 Loop : 1366 09/28/16 09:51:31 Frame : 50 Loop : 786 09/28/16 09:51:32 Frame : 50 Loop : 2060 09/28/16 09:51:33 Frame : 50 Loop : 1447 09/28/16 09:51:34 Frame : 50 Loop : 835 09/28/16 09:51:35 Frame : 50 Loop : 2150 09/28/16 09:51:36 Frame : 43 Loop : 1591 09/28/16 09:51:37 Frame : 43 Loop : 793 09/28/16 09:51:38 Frame : 50 Loop : 2088 09/28/16 09:51:39 Frame : 50 Loop : 1614 09/28/16 09:51:40 Frame : 50 Loop : 973 09/28/16 09:51:41 Frame : 50 Loop : 2223 09/28/16 09:51:42 </pre>	
소개	<ol style="list-style-type: none"> 1. SelectModel로 만들어진 TCPFighter 전용 서버 2. 패킷의 송수신을 섹터단위로 관리한다. 3. 섹터라는 개념으로 SelectModel을 이용하고도 1000명 그 이상을 수용할 수 있다.
개발 환경	WinConsole
소감	처음으로 만든 게임서버이다. IOCP과정을 배우기 전 단계에서 제작한 게임 서버이다. SelectModel이지만 충분히 상용화 가능한 게임서버라 자부심을 가지고 있다.

3. DummyClient

프로그램 명	DummyClient
플랫폼	PC
게임화면	
	
소개	<ol style="list-style-type: none"> 1. Server와 Client의 버그를 잡기 위한 DummyClient이다. 2. 사용자 테스트를 하기 전 Server의 문제점을 찾는 건 매우 어려운 일이다. 3. DummyClient라는 사용자가 할 수 있는 정해진 패턴의 패킷을 서버에 전송해서 Server의 예기치 못한 버그를 찾을 수 있다.
개발 환경	WinConsole
소감	<p>Server와 Client만 디버그를 했을 때 없었던 문제점을 찾을 수 있다. 많은 Client를 프로그래머가 실행하기에는 어려운 점이 많다. 그러므로 DummyClient를 제작하여 Server의 문제 점을 찾아야 한다. 나의 경우 DummyClient로 Server의 사용자 섹터 삭제, 생성 부분에서 문제점을 찾을 수 있었다.</p>

II. 기술요약 및 소스분석

1. Server 구조



Server 구조

2. Server 모듈설명

모듈 명	설명
netIOProcess	- SelectModle을 사용 한 송,수신 버퍼 검사
netSelectSocket	- 반응이 있는 Socket을 Recv Or Send 수행
netProc_Recv	- 섹션을 검사 후 RecvStreamQ에 삽입
netProc_Send	- SendStreamQ에 있는 패킷 같은 섹터에 있는 클라에게 전송
netProc_Accept	- Client접속처리
CompleteRecvPacket	- 정상적인 패킷인지 확인 후 PacketProc호출
PacketProc	- 패킷의 종류에 따른 netPacketProc_@ 호출
netPacketProc_@	- 직렬화 버퍼에서 뽑은 패킷을 처리 및 변경사항을 섹터 단위 패킷 전송(데드레커닝을 이용한 서버와 클라의 좌표 보정)
CreateCharacter	- 섹션에 따른 캐릭터 구조체 생성
DeleteCharacter	- 캐릭터 삭제
FindCharacter	- 캐릭터 검색
Sector_UpdateCharacter	- 캐릭터의 섹터변화를 확인
CharacterSectorUpdatePacket	- 섹터의 변경된 경우 클라에게 관련 정보를 전송
NewCharacterAroundMapInfoSend	- 섹터과 변화된 캐릭터의 정보를 현재 있는 섹터클라에게 보냄

- * sendStreamQ, RecvStreamQ를 이용하여 패킷을 바로 보내지 않고 직렬화 버퍼에 담거나, 받은 후 SelectModle을 이용하여 송,수신
- * 클라이언트, 더미 클라이언트 패킷 송수신부는 서버와 큰 차이가 없음
[더미 클라이언트는 게임로직을 처리하지 않고 데드레커닝으로 좌표계산]
- * Server오류 체크를 위해 캐릭터가 죽는 부를 주석처리

3. 링크

Server 실행파일:

<https://drive.google.com/file/d/0BzusZZDBAoL3TjA0RHRPV2JVWDg/view?usp=sharing>

Client 실행파일:

<https://drive.google.com/file/d/0BzusZZDBAoL3SldMbTkxdVdGZnc/view?usp=sharing>

DummyClient 실행파일:

<https://drive.google.com/file/d/0BzusZZDBAoL3QmpTUVdCU0RiZHM/view?usp=sharing>

Project파일:

<https://drive.google.com/file/d/0BzusZZDBAoL3ajAtY2w4emVHeEU/view?usp=sharing>