1. 서버 컨테이너 변경

서버 프로그램에서 lock으로 임계영역을 생성하는 것은 성능에 매우 큰 영향을 미치기 때문에 이를 제거하기 위해서 공용 데이터를 wait\_free 하거나 lock\_free 해야 한다.



따라서 지금 클라이언트를 관리하는 컨테이너인 ‘remoteClients’를 unordered\_map에서 concurrent\_unordered\_map으로 변경하면 lock없이 무잠금으로 동작할 수 있어 성능 향상에 큰 도움을 줄 것이다.

다만, 이 컨테이너는 erase가 unsafe 하기 때문에 컨테이너의 객체를 제거하기 힘들다.

그래서 클라이언트가 접속을 해제할 때 객체 자체는 남겨두고, 클라이언트가 사용할 수 없다고 표시만 할 것이다. (bool 형태의 변수)

그리고 Player도 Scene에 남아 있기 때문에 Player는 Scene의 gameObjectList에 넣지 않고 클라이언트를 루프 돌아 작업하게 할 것이다.

텍스트이(가) 표시된 사진

자동 생성된 설명

메모리 낭비라 볼 수 있지만 서버는 주기적으로 재부팅 시켜주며 unsafe를 safe하게 만들어주는 작업도 쉽지 않은 과정이기 때문에 시간 효율을 생각하여 이렇게 작업했다.

그 결과 프레임 속도가 2배 정도 늘어난 것을 확인할 수 있었다.

텍스트이(가) 표시된 사진

자동 생성된 설명

플레이어가 떠날 때 다른 클라이언트들에게 알려주고, SC\_REMOVE\_PLAYER 패킷을 받은 클라이언트는 마찬가지로 Player를 비활성화 한다.

1. 서버 Terrain 생성

텍스트이(가) 표시된 사진

자동 생성된 설명

Server에서 사용할 Terrain.h와 Terrain.cpp을 생성하고,

Client에서 사용하던 HeightMap Class를 사용했다.



Server Scene 생성 때 Terrain도 같이 생성해 저장하고,





플레이어 컨텍스트에 연결해준다.

텍스트이(가) 표시된 사진

자동 생성된 설명

1. Player Movement 변경

텍스트이(가) 표시된 사진

자동 생성된 설명

Server의 패킷 처리 일부

텍스트이(가) 표시된 사진

자동 생성된 설명

Scene을 Update하는 Thread의 일부

클라이언트에서 받은 Move방향을 받으면 예전엔 바로 다른 클라이언트들에게도 결과를 보냈지만, 지금은 Concurrent\_unordered\_map을 이용했기 때문에 Scene Update Thread를 문제 없이 이용할 수 있다. 처리할 땐 플레이어의 속도가 있다면 플레이어의 위치를 보내줌으로 위치 정보를 보내준다.

1. Jump / Dash 입력 처리

텍스트이(가) 표시된 사진

자동 생성된 설명

클라이언트로부터 키 입력이 들어오면 PlayerMovementComponent의 Jump/Dash 함수를 호출하여 플레이어의 속도를 변경한다.

1. 클라이언트 Text (UILayer) 추가

이전의 개인 프로젝트에서 작업했던 ChatMGR을 본 프로젝트로 옮겨왔다.

텍스트이(가) 표시된 사진

자동 생성된 설명

DirectX 2D를 이용하기 위한 lib를 미리 추가했다.



ChatMGR 초기화하는 장면

텍스트이(가) 표시된 사진

자동 생성된 설명

정렬 스타일과 글씨체를 새로 정의했다.

텍스트, 스크린샷, 화면이(가) 표시된 사진

자동 생성된 설명

클라이언트 로그인 입력 창에 맞게 크기와 색상을 조정해주었다.

를 GameFramework::FrameAdvance()에 넣어 업데이트한다.

텍스트이(가) 표시된 사진

자동 생성된 설명

텍스트이(가) 표시된 사진

자동 생성된 설명

‘T’를 입력하면 채팅입력 모드로 전환되고

입력한 뒤 ENTER키를 누를 때 LoginScene이라면 그 이름으로 서버에 접속한다.

1. 버그 해결

카메라 고정 해결

텍스트이(가) 표시된 사진

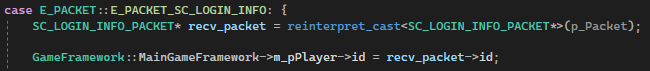
자동 생성된 설명

* 텍스트이(가) 표시된 사진

  자동 생성된 설명

카메라 움직임에서 플레이어의 방향만 바라보고 있어 위아래로는(Pitch) 카메라를 회전하지 못하는 오류가 있었다. 따라서 클라이언트에서 Pitch값을 저장하고 서버로부터 플레이어의 방향으로 카메라를 조정한 다음, 저장한 Pitch값 만큼 카메라를 회전시켜 구현했다.

플레이어 객체 id 오류 해결





클라이언트에서 Scene이 바뀔 때 마다 객체가 삭제되고, 다시 빌드하는 것으로 변경됐기 때문에 클라이언트의 id를 Player가 갖고 있는 것이 아닌 NetworkMGR의 멤버변수로 갖도록 해서 삭제되는 일이 없도록 했다.