**네트워크 연동 사항**

**공통**

파티 슬롯 0번이 여러 로직을 결정하는Master, 그 외가 Slave가 된다.

네트워크 기능을(동기화/스폰) 사용하기 위해서는 해당 액터에NetworkBaseCP 컴포넌트를 넣어야한다.

네트워크 위치 동기화 기능을 사용하기 위해서는 해당 액터에 NetworkTransformCP 컴포넌트를 넣어야한다.

네트워크 애니메이션 동기화 기능을 사용하기 위해서는 NetworkAnimationCP 컴포넌트를 넣어야한다.

**스폰**

네트워크 스폰을 사용하기 위해서는 InGame/InGameNetworkProcessor.h의 ENetSpawnType과 GetClassByType 함수에 추가하여야한다.

High Level RPC/SyncVar

**RPC(원격 함수 호출 : 다른 컴퓨터 함수 호출)**

#include “InGame/Network/Component/NetworkBaseCP.h”

RPC 함수는 리턴 값이 항상 Void이여야 하며, 인자는 다음 중 하나여야 한다.

(void, float, char, EMessageType, bool, FVector, FSerializableString, uint64) RPC로 등록할 함수 선언부의 상단에

RPC\_FUNCTION(클래스 이름, 함수 이름) 을 표기한다.

EX)

RPC\_FUNCTION(AHACKED\_ESPER, Esper\_Attack, int)

void Esper\_Attack(int damage);

1. RPC로 등록한 함수 구현부의 상단에

RPC(NetBaseCP 포인터, 클래스 이름, 함수 이름, RPC 타입,신뢰성 여부, 인자…); 를 표기한다.

EX)

void AHACKED\_ESPER::Esper\_Attack(int damage)

{

RPC(NetBaseCP, AHACKED\_ESPER, Esper\_Attack, ENetRPCType::MULTICAST, true, damage);

}

1. BeginPlay에서 다음을 작성해야한다.

BindRPCFunction(NetBaseCP 포인터, 클래스 이름, 함수 이름);

EX)

void AHACKED\_ESPER::BeginPlay(){

…

BindRPCFunction(NetBaseCP, AHACKED\_ESPER, Esper\_Attack);

…

}

**변수 동기화**

#include “InGame/Network/Component/ReplicatableVar.h”

CReplicatableVar<타입> 변수명; 으로 선언을 한다음, BeginPlay에서 변수명.Init(&NetBaseCP 포인터);를 시행하면 된다.

PrimitiveType 만 가능. (int, float, char, bool) (나머지는 동작을 확신할 수 없음.)

Operator\*을 사용하면 실제 값을 받아 올 수 있다.

Low Level RPC/SyncVar (응용부에서 사용하지 않음.)

**RPC (원격 함수 호출 : 다른 컴퓨터 함수 호출)**

RPC Type의 함수는 void (int32 len, char\* buf) 타입으로 필수된다.

Len, Buf로 인자가 넘어가며, 직접 Parsing해서 사용하여야한다.

다음과 같이 int를 핸들로 BeginPlay에서 함수를 바인딩하고,

auto newDelegate = networkBaseCP->CreateBindableDelegateFunction((int32)ETestFunc);

if (newDelegate) {

newDelegate ->BindUObject(this, &AMyActor::TestFunc);

}

다음과 같이 호출 가능하며,

networkBaseCP->ExecuteNetFuncByVoid((int32)ETestFunc, ENetRPCType::MULTICAST);

void ExecuteNetFunc(int32 handle, ENetRPCType type, int len, char\* arg, bool isReliable = true, bool excuteSelf = true);

제일 law한 함수는 다음과 같다.

RPC 타입은 다음과 같다.

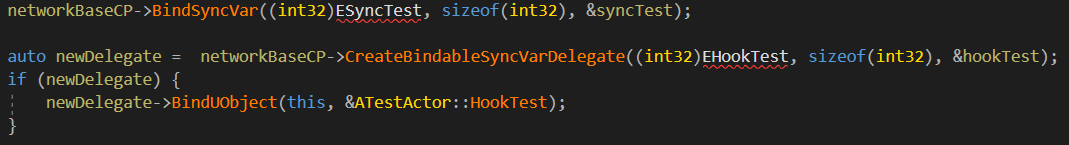
MULTICAST : 자신을 포함한 모든 대상에게 함수 호출 명령

MASTER : 마스터에게 함수 호출 명령

TARGET : 동기화 권리를 Slave가 소유하는 액터안에서, 해당 Slave에게 함수 호출을 명령한다.

**변수 동기화 + 변수 동기화 감지 콜백 함수**

변수동기화는 오직 어쏘리티를 가진자만 가능하다.



다음과 같이 SyncVar 혹은 HookVar를 Bind 한뒤,



CP->SetSyncVar 함수로 인자를 넘기면 된다.

기본 인자는 len, buf[len]이며, 기본 변수를 랩핑한 함수가 몇 개 준비되어 있다.

Hook 함수는 RPC와 같이 int len, buf[len] 의 형식으로, 변수의 변경값이 넘어오며, 직접 Serialize해서 사용하도록 해야한다.