네트워크 메시지 정리

맨앞의 C와 S는 Client에서 발송, Sever에서 발송의 의미를 담고 있으며,

발송자\_메시지사용처\_메시지설명 으로 작명 되었음.

본 문서의 순서는 메시지 타입 ENUM의 순서와 관계 없음.

1. COMMON\_ECHO

+ String

단순 에코로, 서버와 클라이언트의 구분이 없다. FSerializableString으로 보내며, 수신측은 로그나 화면에 표시하도록 한다.

1. COMMON\_PING

연결을 유지시키기 위해 핑을 주고 받는 메시지. 서버에서 PING을 보내면 똑같이 클라이언트에서 PING을 보내는 형식으로 Heartbeat를 감지한다.

1. S\_Common\_RequestId

스팀 아이디 전송을 요청한다.

C\_Common\_AnswerId에서 연동

1. C\_Common\_AnswerId

ㄴ UINT64 steam ID

스팀 아이디를 담아 답장한다.

1. C\_Lobby\_InviteFriend\_Request

ㄴ UINT64 targetID

스팀 아이디를 담아, 초대를 요청한다.

S\_Lobby\_InviteFriend\_Request에서 연동

1. S\_Lobby\_InviteFriend\_Request

ㄴ UINT64 SenderID

요청자의 아이디를 담아, 대상에게 초대를 요청한다.

C\_Lobby\_InviteFriend\_Answer에서 연동

1. C\_Lobby\_InviteFriend\_Answer

ㄴ bool isAccept

ㄴ UINT64 targetID

요청 받은 타겟의 ID와, 응답을 담아 전송.

S\_Lobby\_InviteFriend\_Answer에서 연동

1. S\_Room\_Info

ㄴ UINT64 slot[MAX\_PLAYER]

방 정보를 모두 전송.

1. S\_Room\_ChangeState

ㄴ int slot

ㄴ UINT64 steamID

해당 슬롯의 변경을 통지

1. S\_Lobby\_InviteFriend\_Failed

ㄴ FSerializableString msg

방이동 실패 통보.

1. C\_Lobby\_SetPartyKing

ㄴ int slot

0번 슬롯 클라이언트에서 해당 슬롯에 파티장 위임을 요청.

통보는 S\_Room\_Info로 온다.

1. C\_Lobby\_FriendKick\_Request

ㄴ int slot

0번 슬롯 클라이언트에서 해당 슬롯에 파티원 강퇴를 요청.

이후 S\_Room\_Info로 온다.

1. C\_Lobby\_MatchRequest

ㄴ bool isOn

Bool의 상태로 매치 상태를 변경한다.

1. S\_Lobby\_MatchAnswer

ㄴ bool isOn

ㄴ int slot

해당 술롯이 매치 상태를 변경했음을 통보한다.

1. S\_Lobby\_GameStart

게임 시작을 클라이언트들에게 통보한다.

1. C\_Debug\_RoomStart

디버그 전용 메시지.

게임 진행과 상관없이 비어있는 룸을 찾아 들어간다.

1. C\_InGame\_RPC

ㄴ char callType (0: multicast, 1: server)

ㄴ int32 objectID

ㄴ int32 functionHandle

ㄴ int32 len

ㄴ char arg[len]  
RPC를 서버에게 요청한다.

1. S\_InGame\_RPC

ㄴ int32 objectID

ㄴ int32 functionHandle

ㄴ int32 len

ㄴ char arg[len]

RPC 를 클라이언트들에게 통보한다.

1. C\_InGame\_SyncVar

ㄴint32 objectID

ㄴint32 syncvarHandle

ㄴ int32 len

ㄴchar arg[len]

마스터(혹은 동기화 주도권을 가진 단말)에서 나머지 클라이언트들에게 변수 동기화를 요청한다.

1. S\_InGame\_SyncVar

ㄴint32 objectID

ㄴint32 syncvarHandle

ㄴint32 len

ㄴchar arg[len]

서버에서 나머지 Slave들에게 변수 동기화를 통보한다.

1. C\_UDP\_Reg

UDP 연결을 요청한다.

S\_UDP\_Response에서 계속.

1. S\_UDP\_Response

UDP 연결 응답을 보낸다.

1. S\_Disconnect\_Slot

ㄴ int32 slot

해당 슬롯이 disconnect 됬음을 알림.

1. S\_Reconnect\_Slot

ㄴ int32 slot

해당 슬롯이 Reconnect 됬음을 알림.

1. S\_Reconnect\_Slot

ㄴ UINT64 slot[MAX\_PLAYER]

방 정보와 함께 재접속한 사람에게 재접속임을 통보함 .

1. C\_End\_Game

게임 종료를 통보한다.

S\_END\_GAME에서 계속

1. S\_End\_Game

게임 종료를 해당 룸에 통보한다.

1. C\_Reconnect\_Server

ㄴ bool isGameRoom

ㄴ UINT64 myID

ㄴ UINT64 slot[MAX\_PLAYER]

클라이언트에 유지된 데이터를 토대로 서버에서 복구를 요청한다.