struct ~~Login~~ Msg
{
char User
char Pass
}

1. AVS
set User
set Pass

Login Msg.Cpp AVS.cpp   Login exec

3.

↓ 2.
set MsgType Login
set User
set Pass

↓ 4.
Login Packet user
             pass

↓ 5.
Login Packet. Write Packet

siehe
(age
firstname)
Lastname)
Beispiel GME

↓ 6.
Login Packet. Destroy Packet

Loginpkt.cpp

Getpos

Setpos

ReadPacket
Get Type
Handle Packet (Type) Packet)
  Switch ~~Type~~
    Case Login
      Handle Packet Login (Packet)

Msg types ─┐
Handlers ──┼──────────────────────────┐
GNE: │ │

Login Msg   Login Msg Handler

            uint32. GetMsgLength → return origin.Length() + text.Length()+2;
Virtual   void — Serialize → buf.WriteString(origin)
          bool — Deserialize ↘   (text)
                          ↘ if(!buf.ReadString(origin)) return false;
csString origin,text;   (text))
                          return true;   ''

          virtual int GetSerialType() const {return PacketID;}
               static int PacketID;

                    Serial Buffer :: Read...
                              write...

                              Init
                              InitBuffer

Parsing

World + Models in Sector

Player ⟶ is Model

Player → PC

Player → NPC

Player → Monster

PC ↓ is Controlable

NPC ↓ is AI

Monster ↙ is Enemy

Monster ↓ is Enemy