

# CMP 303 - Networking Presentation

By Joseph Roper

# Network Architecture

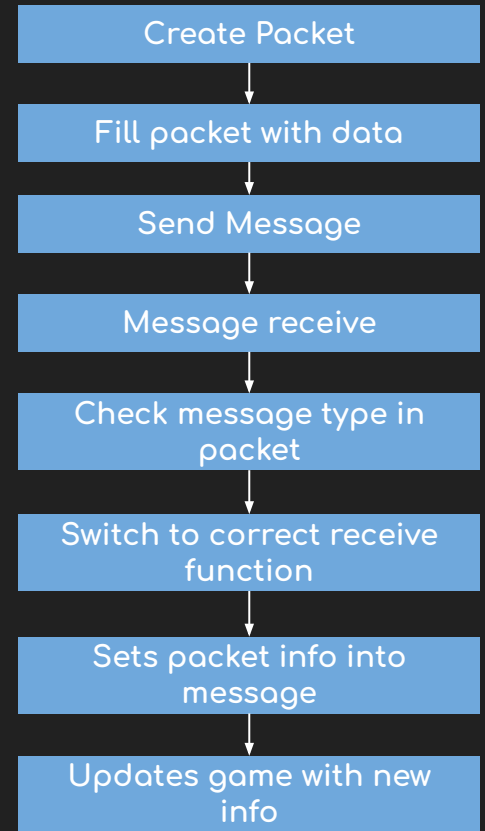
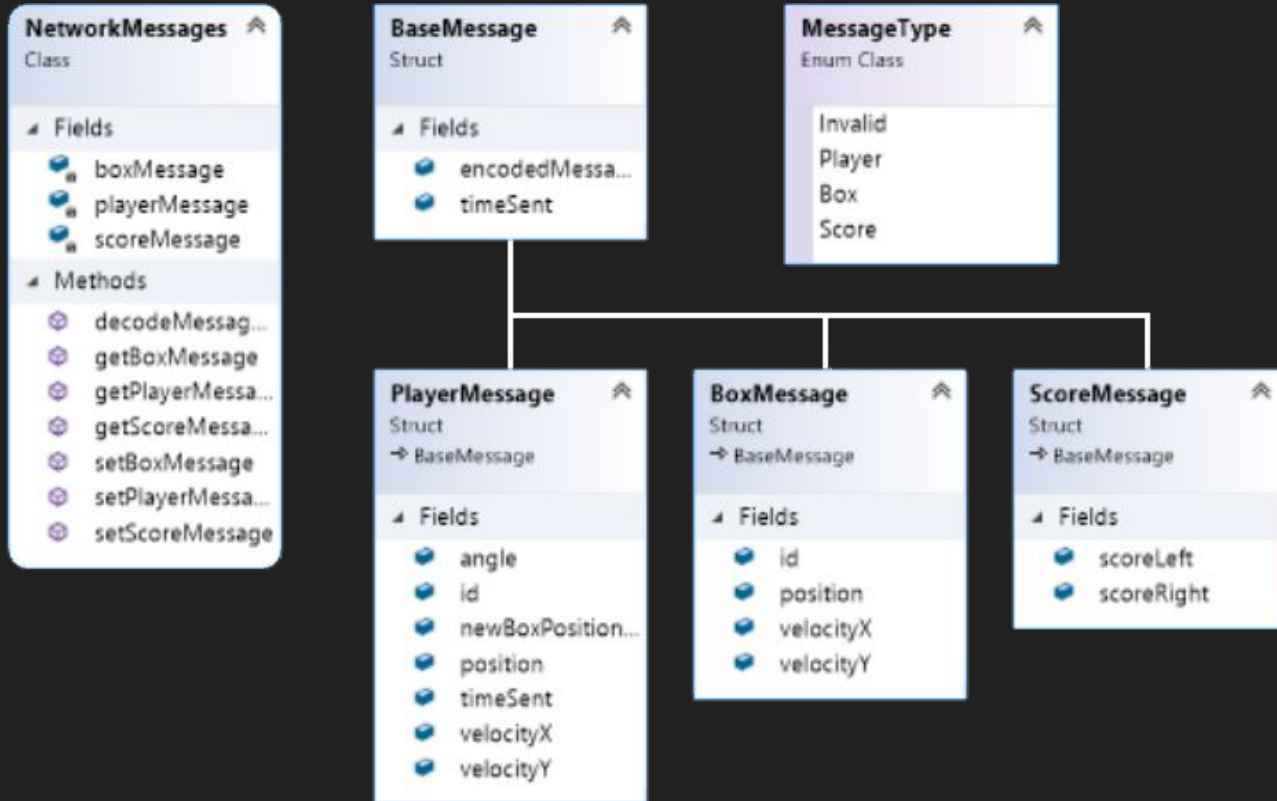
## Client - Server

There can only be 1 client and 1 server, as I wanted the game to only be 2 player.

This was chosen over peer-to-peer out of preference, I believe client & server is better during latency as it has a predefined source for valid data aka the server

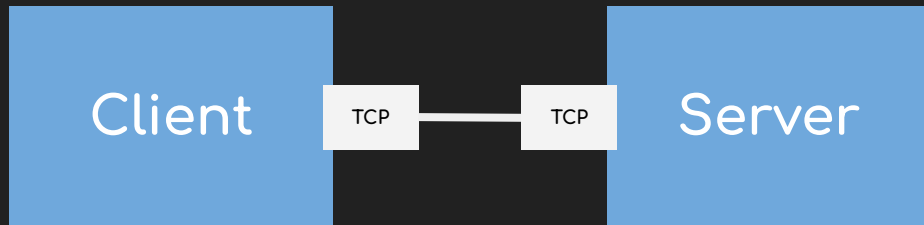


# Application Layer



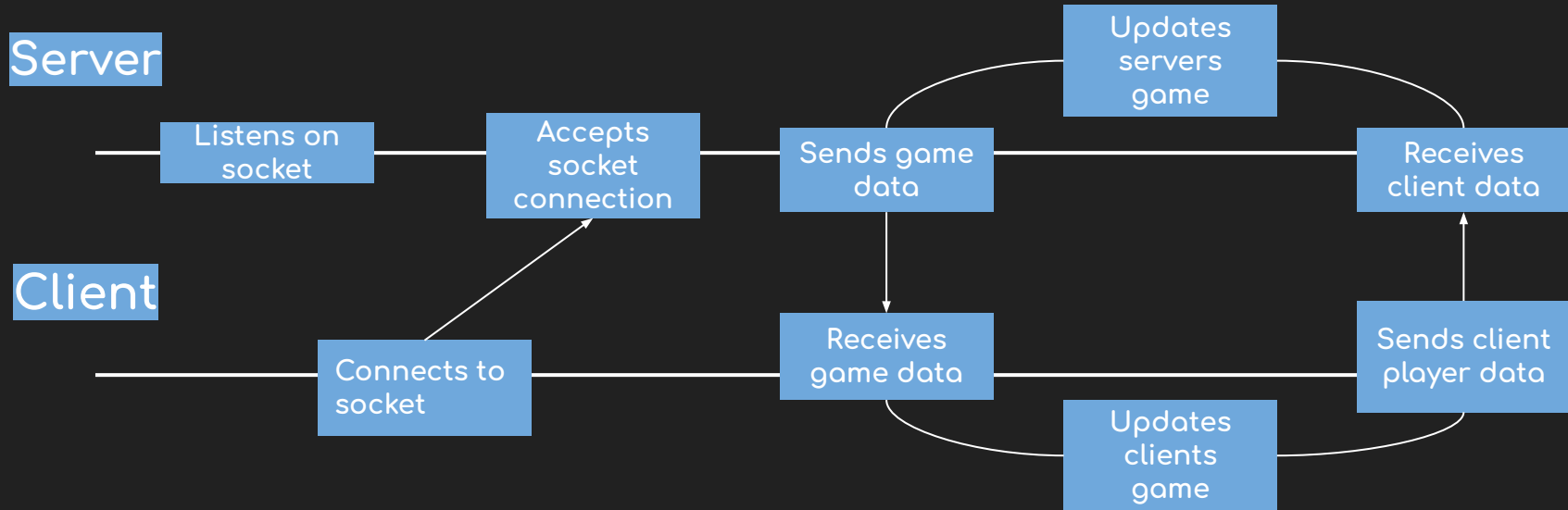
# Transport Layer

- I use TCP
- Good for small applications
- Reliable
- Message sorting not needed unlike UDP
- UDP transfer speeds are faster but unnecessary for my application



The socket is set to non-blocking so that the program is not blocked when a message is not received.

# Network Communication



# Network API

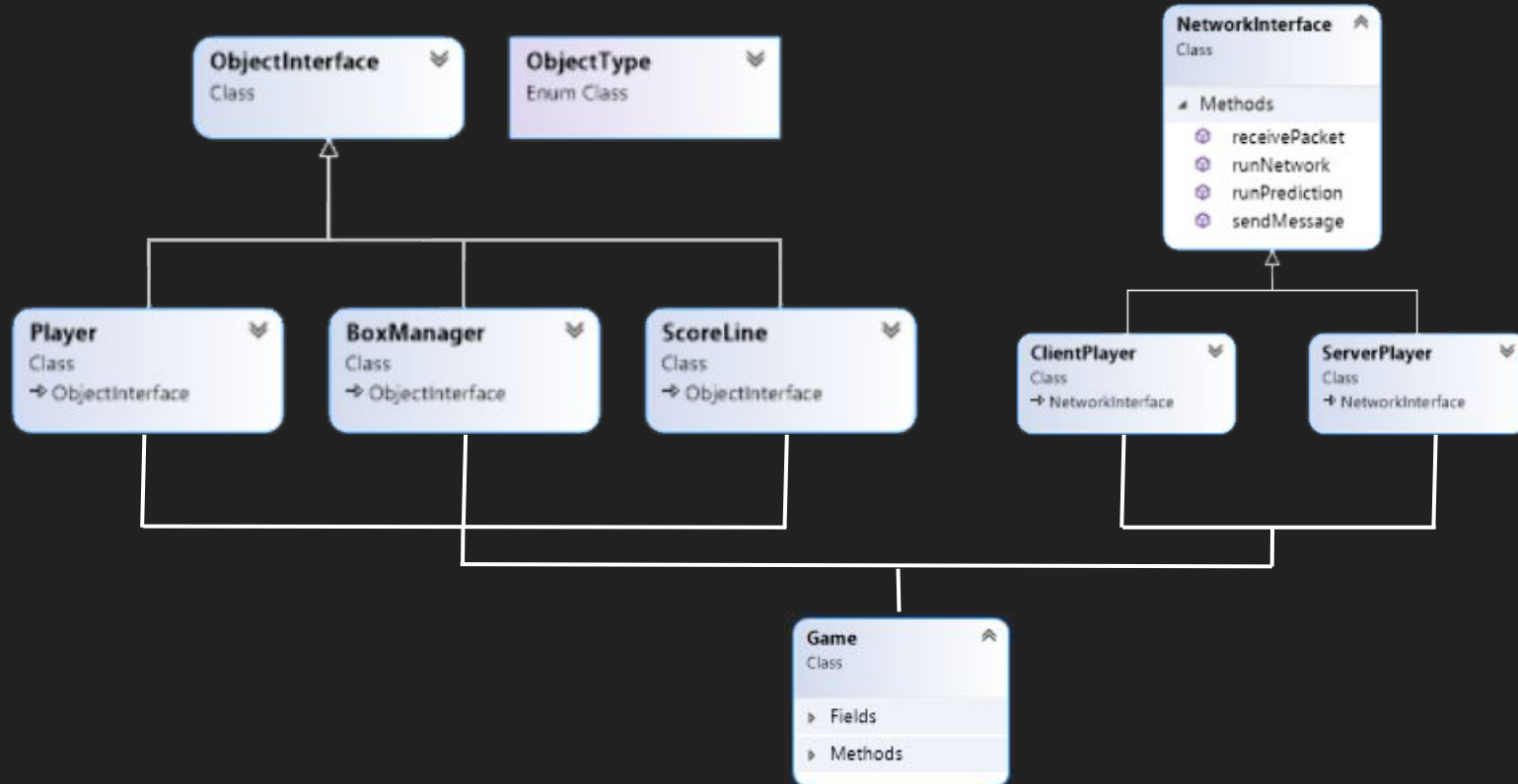
I have chosen to use the SFML c++ library to make my application and thus used its networking API. SFML is well written and allows for effective object-oriented programming. I mainly used SFML as I have experience from using it in another university module. Due to this, it is more compatible to use SFML's own Networking API in a SFML created project than trying to implement winsock.



SFML 2.5.1

Latest stable version

# Game Structure



# Prediction

I have used linear prediction for my game.

For this, I created a vector which holds 3 of the most recent player messages at a time.

When a player message is received, it stores the new message into the vector.

In the game loop, the calculate function is called to update the player's position.

Prediction calculation;

Next position = previous position + displacement

Displacement = speed \* time (since last message)

Speed = distance between last two positions / time between last two positions

Interpolation was not necessary for my program as the player movement and prediction already fulfil that role

## PlayerMessage

Struct

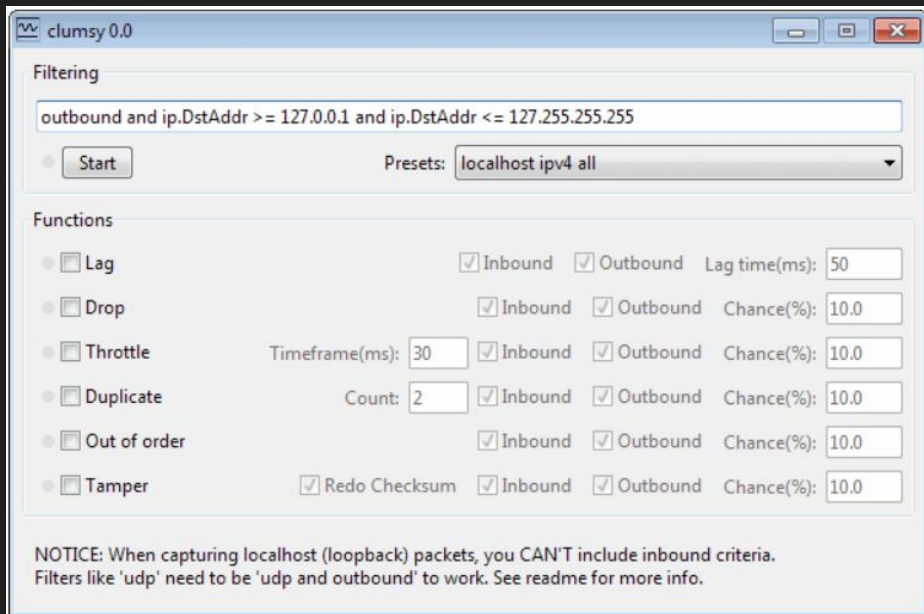
→ BaseMessage

### Fields

- angle
- id
- newBoxPosition...
- position
- timeSent
- velocityX
- velocityY



# Testing



## Functions Used When Testing

Lag (ms): 50, 100, 150, 200

Drop (%): 10%, 50%

Duplicate (%): 10%, 50% Count: 2

Out of Order (%): 10, 50

### Final Tests:

Lag (ms): 50 ms Drop (%): 5

Lag (ms): 50 ms Drop (%): 10

Lag and Drop had the most effect on my program. Especially with the box movement.

Duplicate and Out of Order had no effect due to the rate of messages that are sent and prediction.

# Critical Discussion

Whilst running my programs, the game windows flicker, but I believe this is a problem with SFML and would be fixed if I used a later version of the library.

In the future, I would like to add different states within the game such as a start screen and pause menu and improve on the gameplay such as a dash for the player. The first implementation I would add is multiple boxes, however it would require a lot more data to be transferred between the client and server.

Whilst testing with clumsy, I noticed the box behaviour to be strange, with lag and dropping of packets. I believe to fix this, I will have to add prediction to the box as well.

# References

Sfml-dev.org. (2019). SFML. [online] Available at: <https://www.sfml-dev.org/>.

www.youtube.com. (n.d.). SFML C++ Tutorial 01 | Setting up SFML. [online] Available at: [https://www.youtube.com/watch?v=yEiZalvDOj4&t=6s&ab\\_channel=SurajSharma](https://www.youtube.com/watch?v=yEiZalvDOj4&t=6s&ab_channel=SurajSharma) [Accessed 28 Mar. 2022].

Twitter. (n.d.). <https://twitter.com/davitmasia/status/1195262851516116992>. [online] Available at: <https://twitter.com/davitmasia/status/1195262851516116992?lang=ur> [Accessed 6 Apr. 2022].

Stack Overflow. (n.d.). c++ - error LNK2005, already defined? [online] Available at: <https://stackoverflow.com/questions/10046485/error-lnk2005-already-defined> [Accessed 6 Apr. 2022].

en.sfml-dev.org. (n.d.). sf::TcpSocket, sf::Packet and NonBlocking behavior... [online] Available at: <https://en.sfml-dev.org/forums/index.php?topic=16477.0> [Accessed 6 Apr. 2022].

en.sfml-dev.org. (n.d.). How to use non-blocking TCP? [online] Available at: <https://en.sfml-dev.org/forums/index.php?topic=4944.0> [Accessed 6 Apr. 2022].

www.sfml-dev.org. (n.d.). sf::Packet Class Reference (SFML / Learn / 2.5.1 Documentation). [online] Available at: [https://www.sfml-dev.org/documentation/2.5.1/classsf\\_1\\_1Packet.php](https://www.sfml-dev.org/documentation/2.5.1/classsf_1_1Packet.php).

www.youtube.com. (n.d.). SFML 2.4.x Tutorial 003 - Text and Fonts. [online] Available at: <https://www.youtube.com/watch?v=jRmF-rW1AtE> [Accessed 6 Apr. 2022].

www.youtube.com. (n.d.). SFML C++ Tutorial 11 | Text and Font quick and easy. [online] Available at: <https://www.youtube.com/watch?v=IEZkQvQ6ivU&t=261s> [Accessed 6 Apr. 2022].

www.youtube.com. (n.d.). SFML C++ Tutorial 17 | Collision detection with screen and objects! (PART 1). [online] Available at: <https://www.youtube.com/watch?v=onRDXy6WoVk&t=1s> [Accessed 6 Apr. 2022].

www.dafont.com. (n.d.). Retro Gaming Font | dafont.com. [online] Available at: <https://www.dafont.com/retro-gaming.font>.

jagt.github.io. (n.d.). clumsy, an utility for simulating broken network for Windows Vista / Windows 7 and above. [online] Available at: <https://jagt.github.io/clumsy/download.html> [Accessed 7 Apr. 2022].