

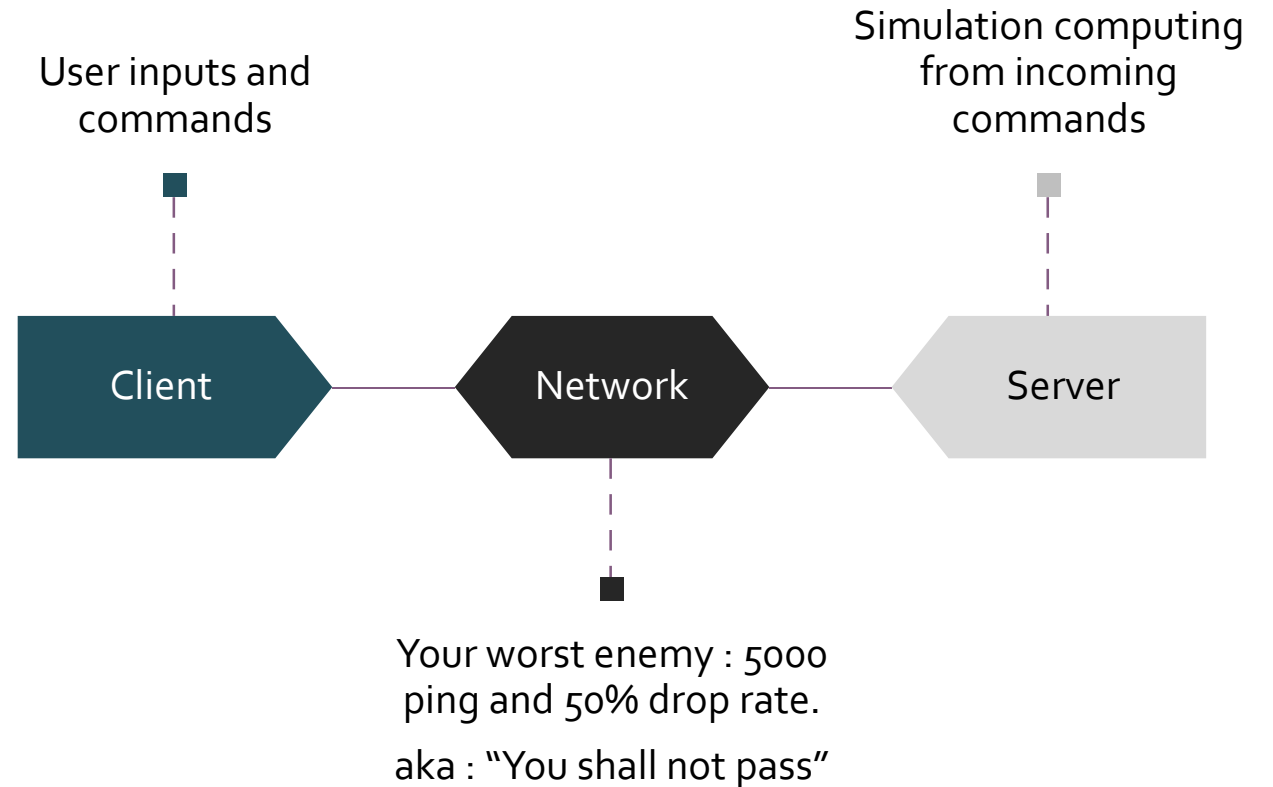


*Also known as :
How do we calculate and fix client
position after server-side computing,
with a network with 800 ping and still
keep going ?*

Lag Compensation in Games

Client / Server Architecture

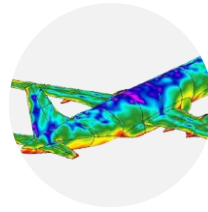
Quick overview and reminder of
client/server architecture in video
games



Workarounds and theres Drawbacks

« There is no problems, there is only solutions. The Human mind then invent the problem. It sees problems everywhere. »

[André Gide](#)



Client and Server
Simulation



Hardware
Infrastructure



Design Choice



Security



Cost

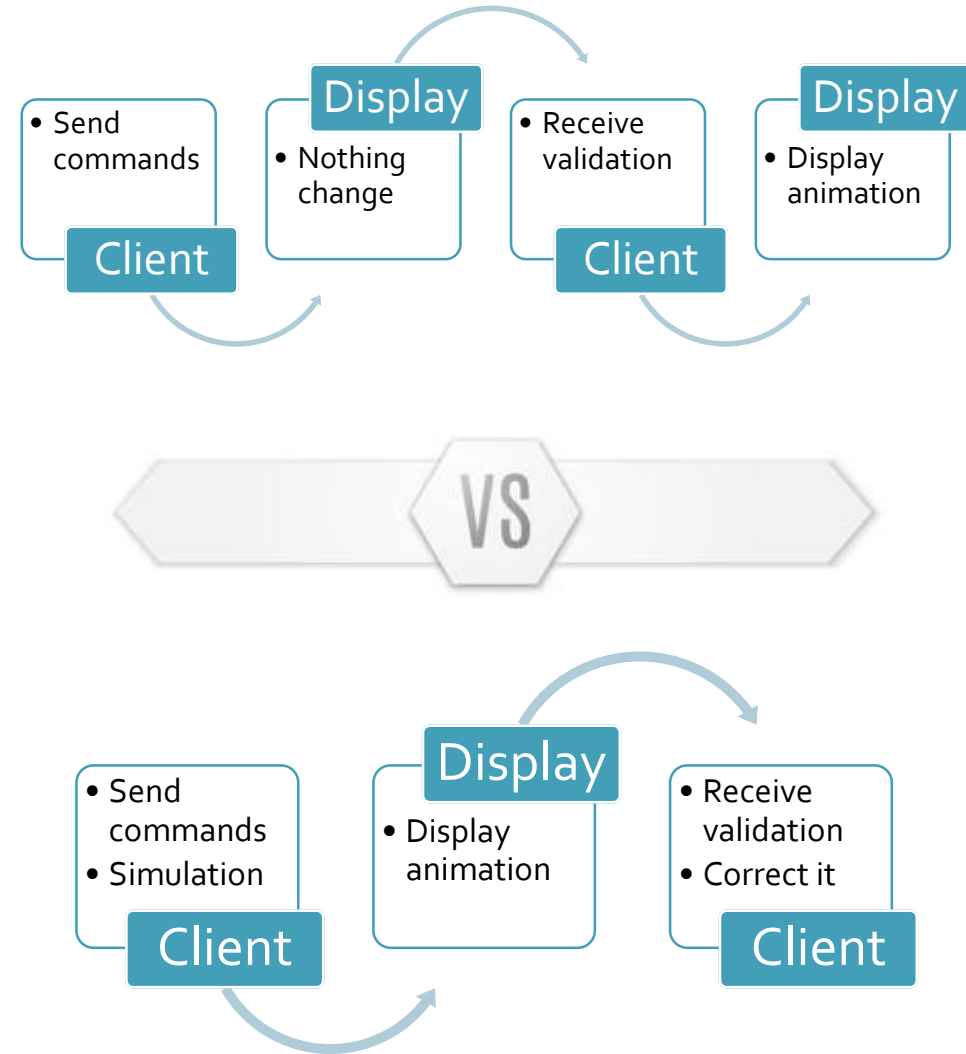


Bug or Feature ?

Client / Server Simulation

“My business is to paint what I see,
not what I know is there.”

— J.M.W. Turner



Client and Server Simulation

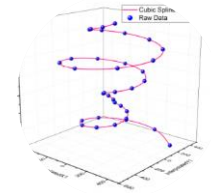
Few algorithms and methods about
simulation on client and/or server



Prediction



Reconciliation



Interpolation

Prediction

Foresee server answer, and act before it.

1

Store and send actions to the server, then compute a simulation with it.

2

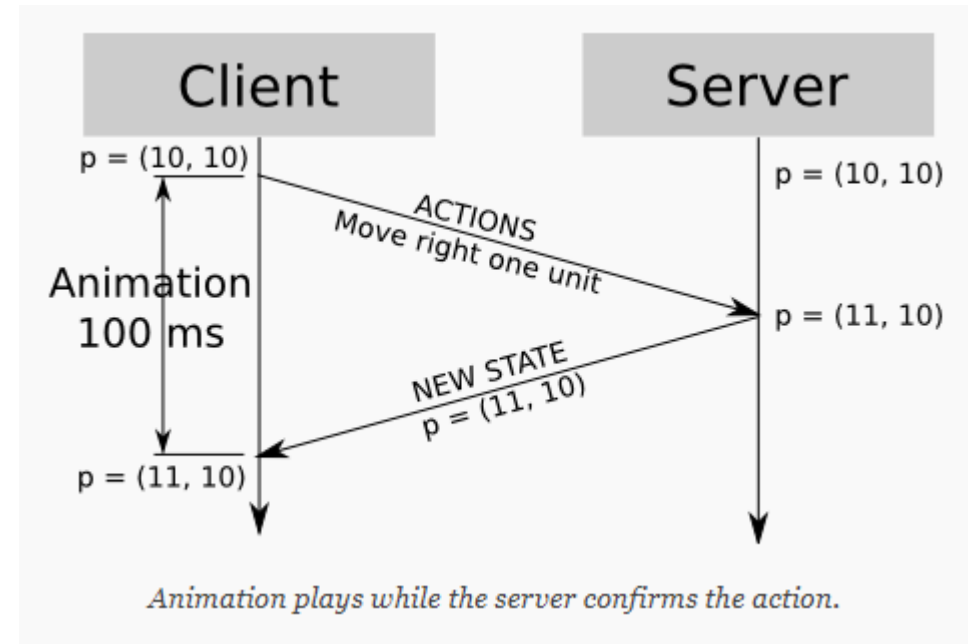
Display the result of your local simulation

3

On server validation reception, check your output, and correct it if needed.

Prediction

The goal is to play animations while waiting for the validation from the server.



Server Reconciliation

Conciliate the expectation of the client and the server answer.



Store and send actions to the server, then compute a simulation with it.



Display the result of your local simulation



On server validation reception...
PROBLEM

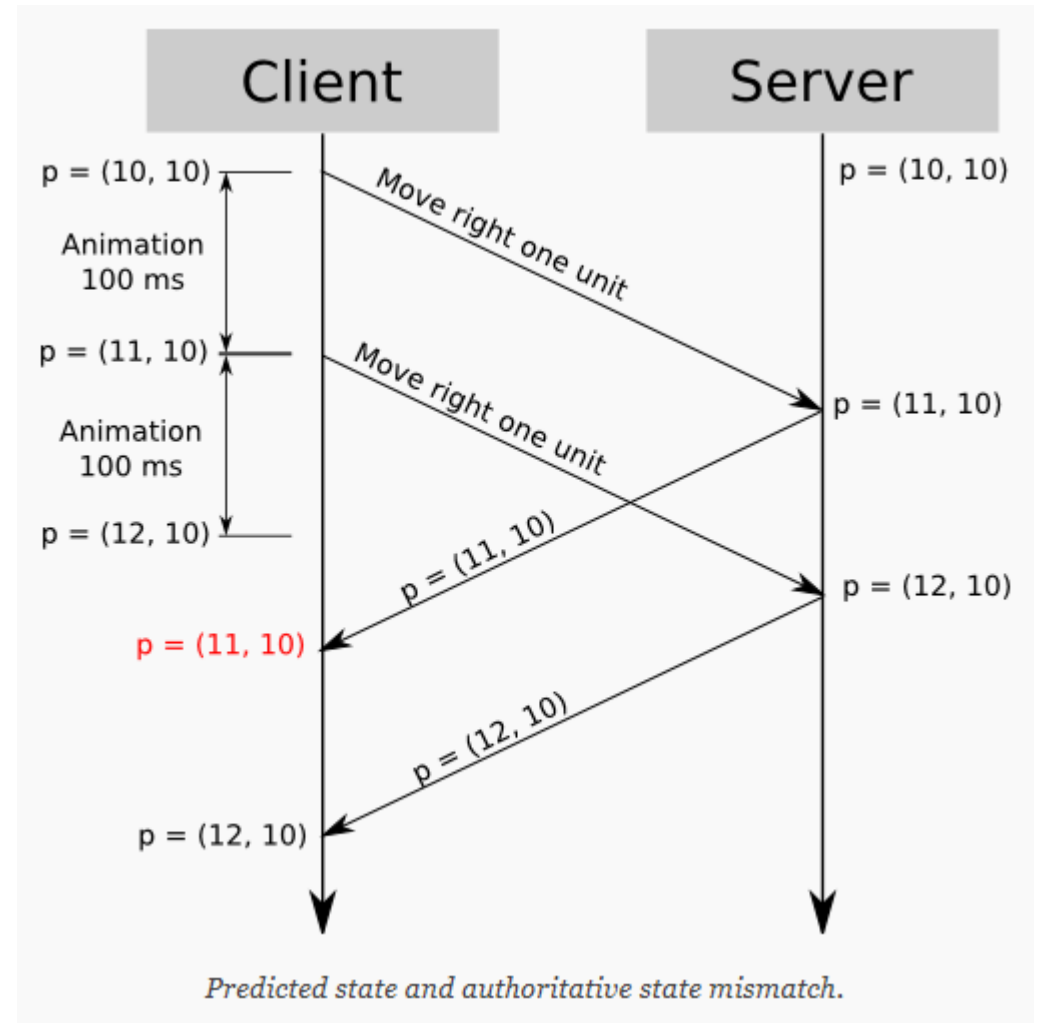


Solution :

Store state and roll back only if needed

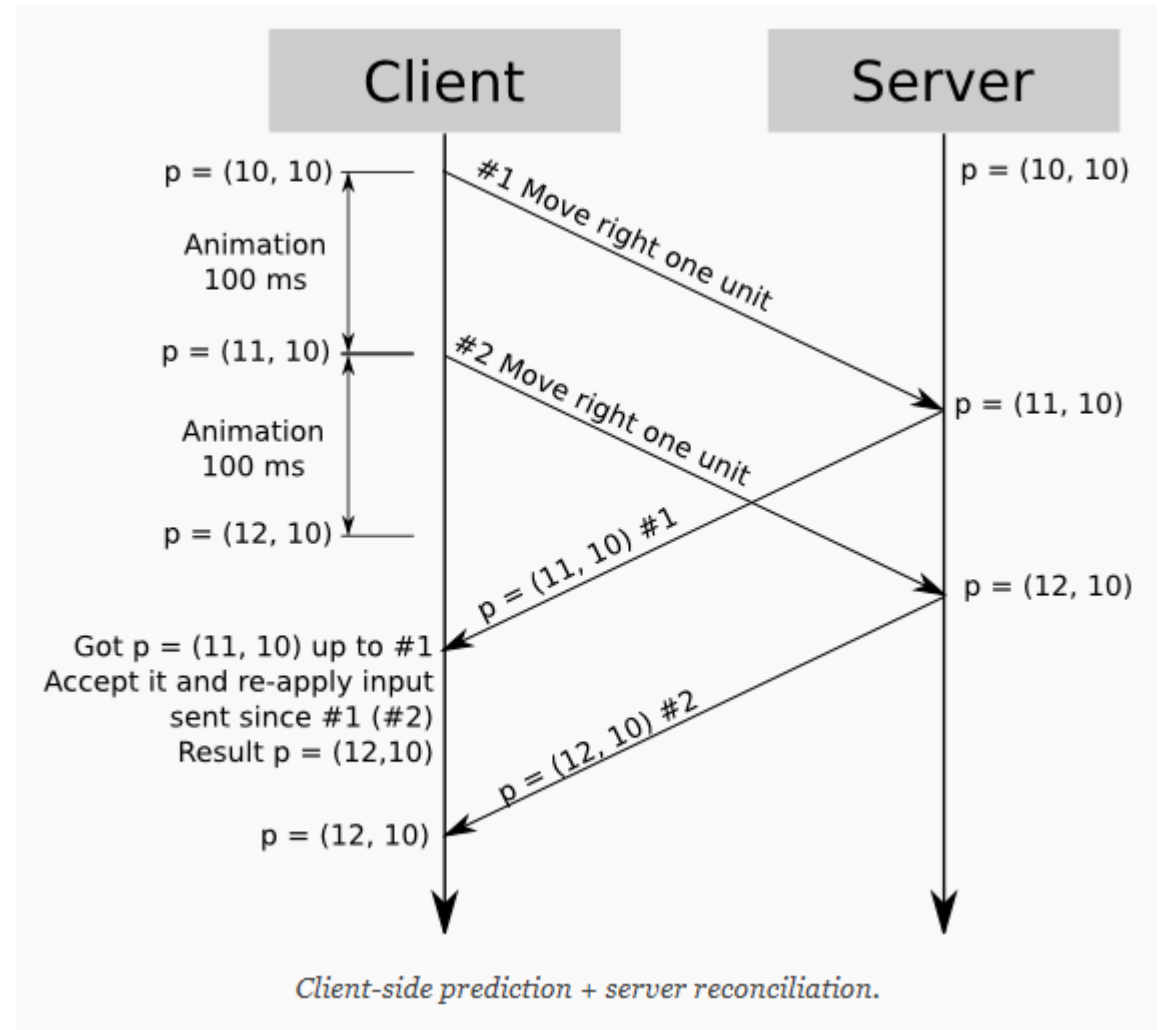
Server Reconciliation

We have conflict of state, we receive validation for state 11 while being in state 12



Server Reconciliation

In order to avoid to rollback from 12 to 11 when we receive validation, we store unvalidated states. Then, we recomputed from save point, and rollback only if output is different.



Interpolation

Avoid teleportation and sudden update on multi-client application



Store and send actions to the server, then compute a simulation with it.



Display the result of your local simulation



Receive server validation and conciliate simulation with it.



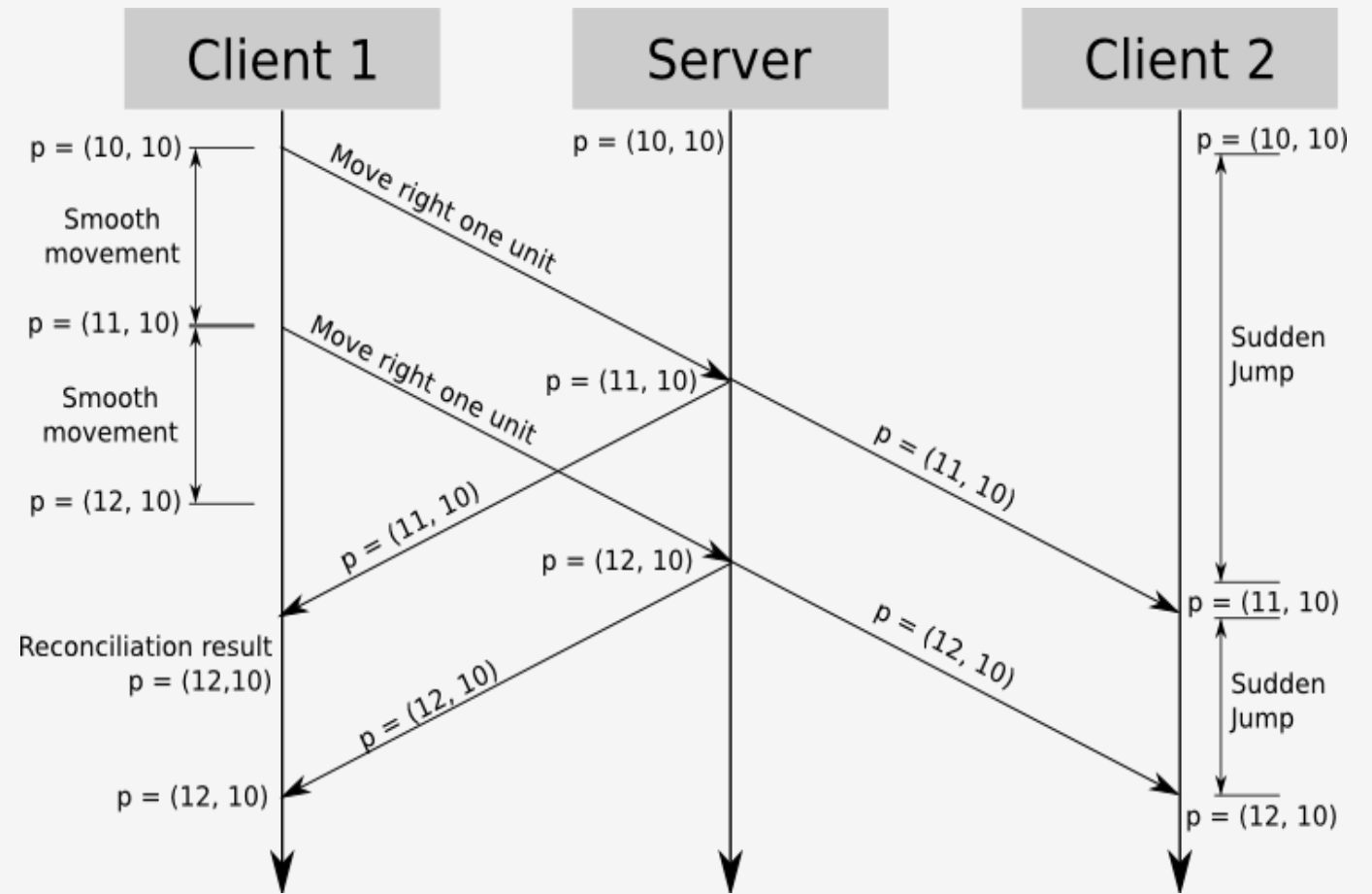
On another client validation reception...
PROBLEM



Solution :
Apply prediction on other objects, known as interpolation

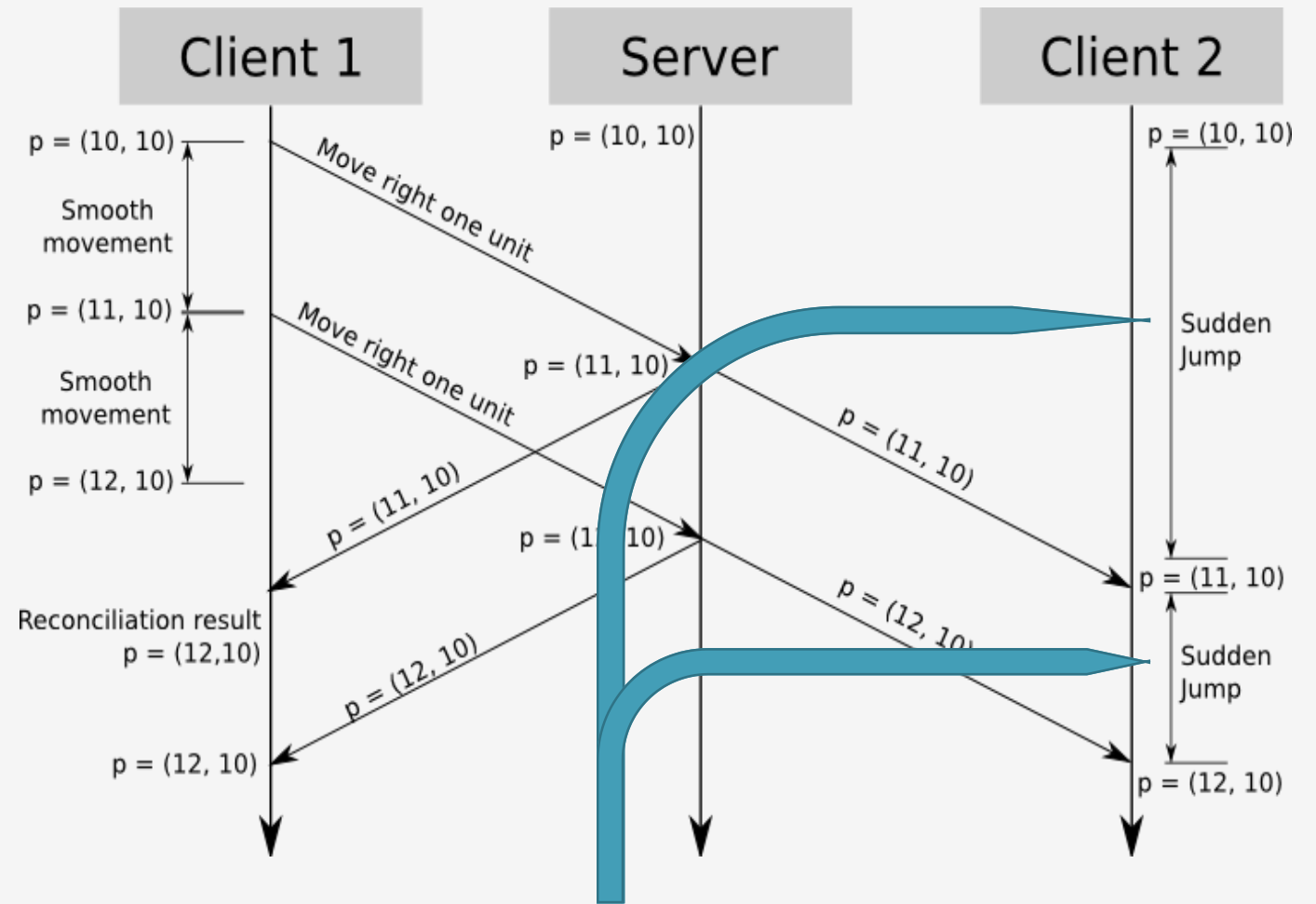
Interpolation

"A wild update appeared!"



Interpolation

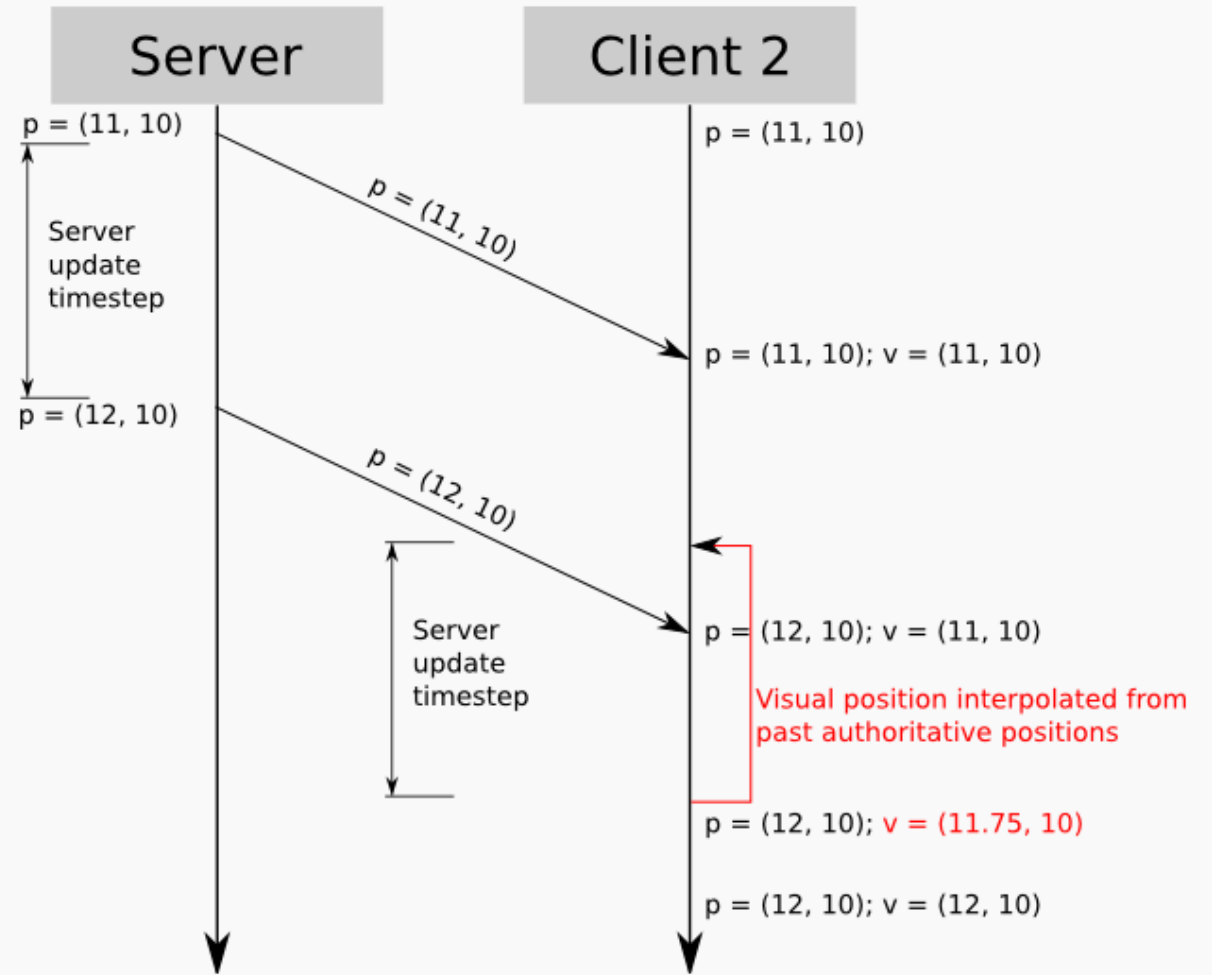
"Dead Reckoning" Strategy



Solution : Just keep going this way !

Interpolation

“Entity Interpolation” Strategy



Client 2 renders Client 1 “in the past”, interpolating last known positions.

Sum-up

- Client simulation helps to reduce delay between computation and display
- Client have to deals with the server validation that validates past event.
- In case of multiple clients, you have to run a simulation for each clients.
- Despite all of theses simulations, some rollback may happens, depending of the server simulation.
- All in all, the best solution is still to have low ping !

Questions ?

Link & Resources

- Git Half-Life :
<https://github.com/ValveSoftware/halflife>
- Half-Life documentation :
https://developer.valvesoftware.com/wiki/Latency_Compensating_Methods_in_Client/Server_In-game_Protocol_Design_and_Optimization
- Online documentation and demo :
<http://www.gabrielgambetta.com/client-side-prediction-live-demo.html>