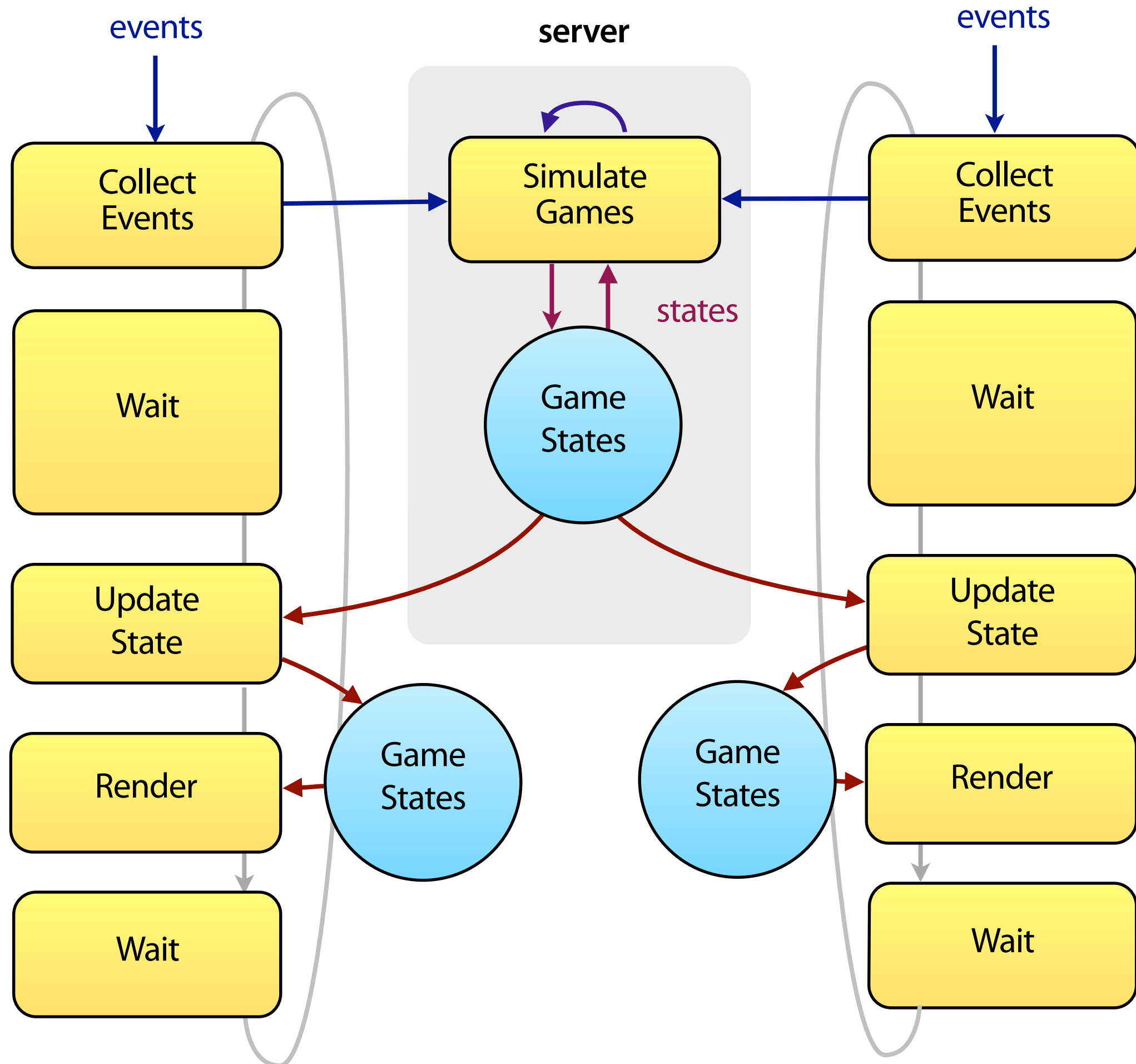


# Lecture 2

Lag



# Demo:

# Two-player Pong

# Event Messages

```
{  
  type: x,  
  key1: value1,  
  key2: value2..  
}
```

```
{  
  type: "move",  
  x: 30  
}
```

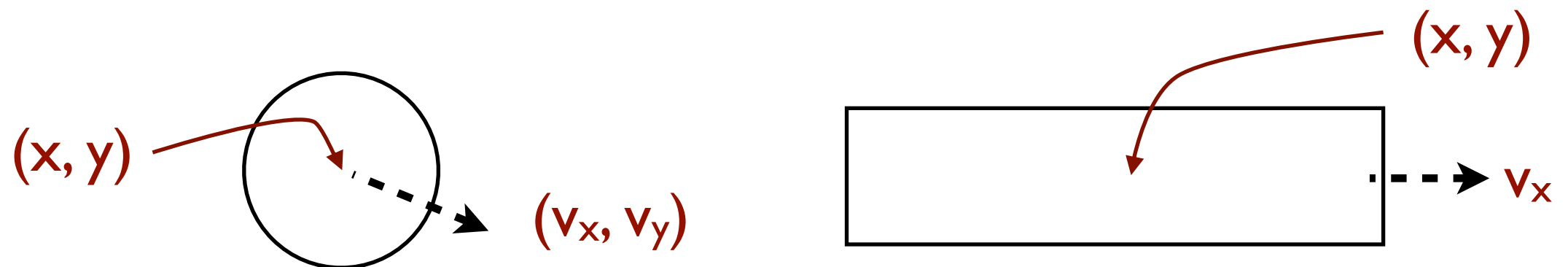
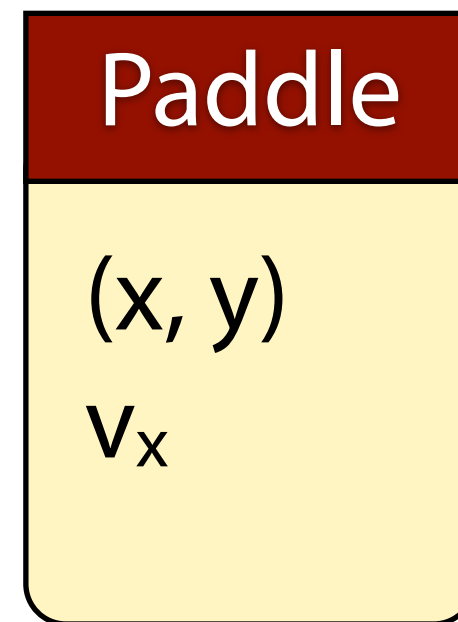
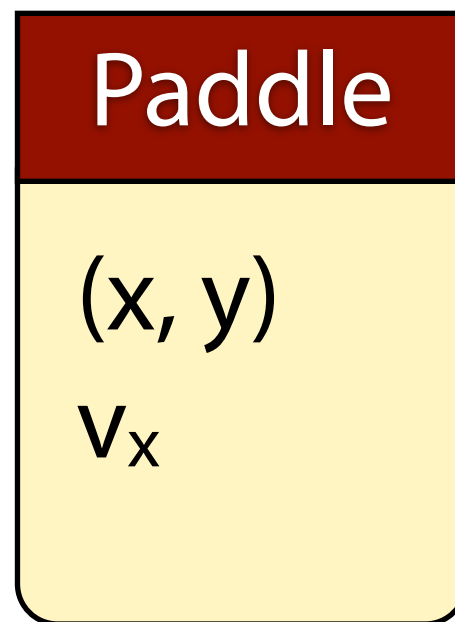
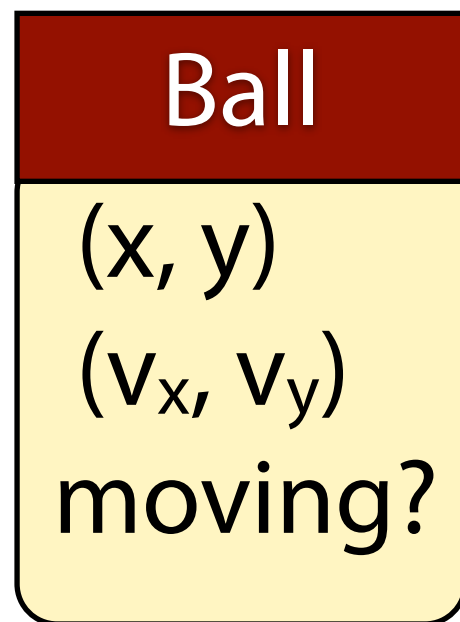
```
{  
  type: "start",  
}
```

```
{  
  type: "accelerate",  
  vx: 30  
}
```

```
{  
  type: "delay",  
  delay: 100  
}
```

```
{  
  type: "update",  
  ballX: 10,  
  ballY: 10,  
  myPaddleX: 10,  
  myPaddleY: 400,  
  oppPaddleX: 100,  
  oppPaddleY: 0  
}
```

# Game States





# Game Simulation:

move the paddles

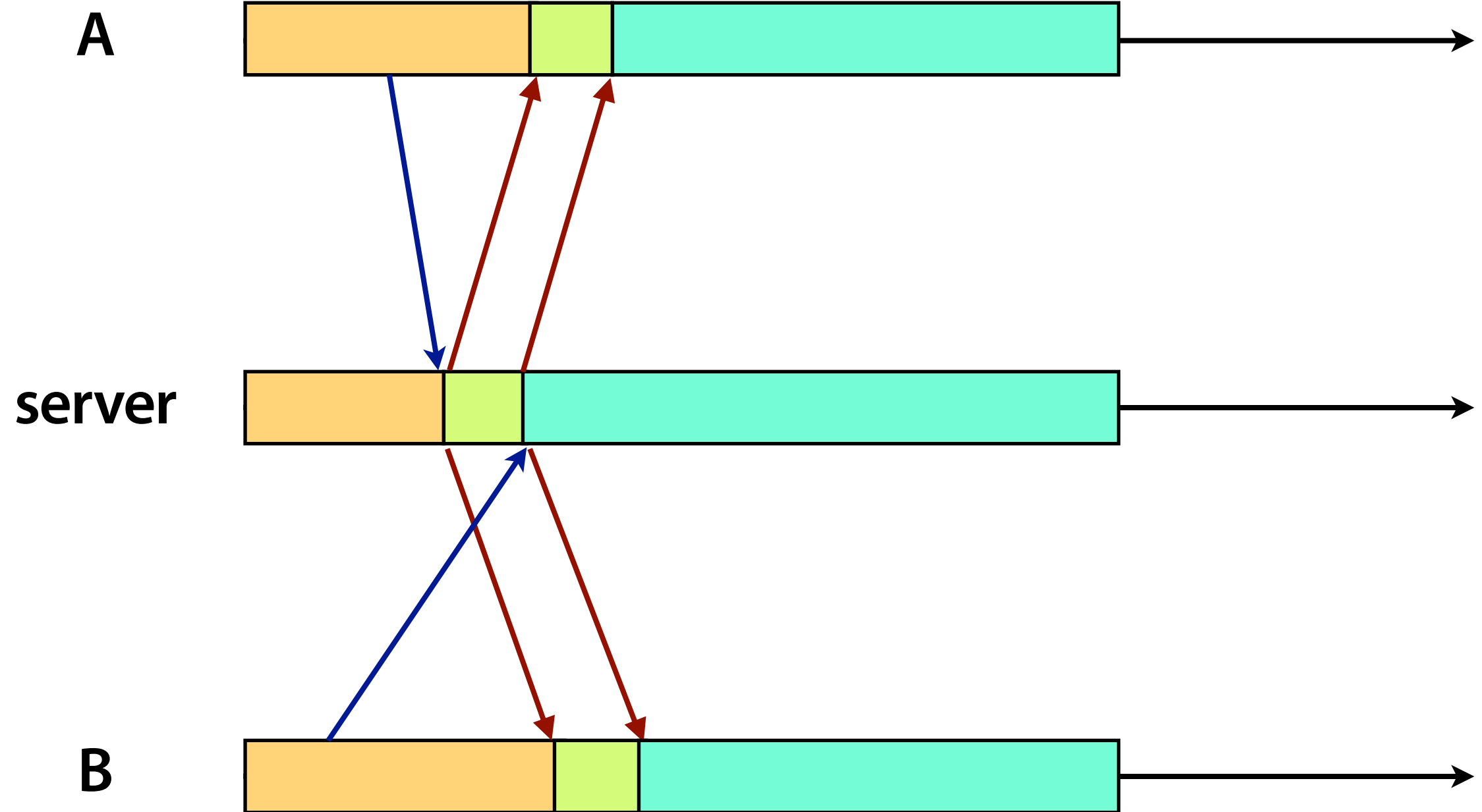
move the ball

if hit walls or paddles, bounce

if miss, restart game

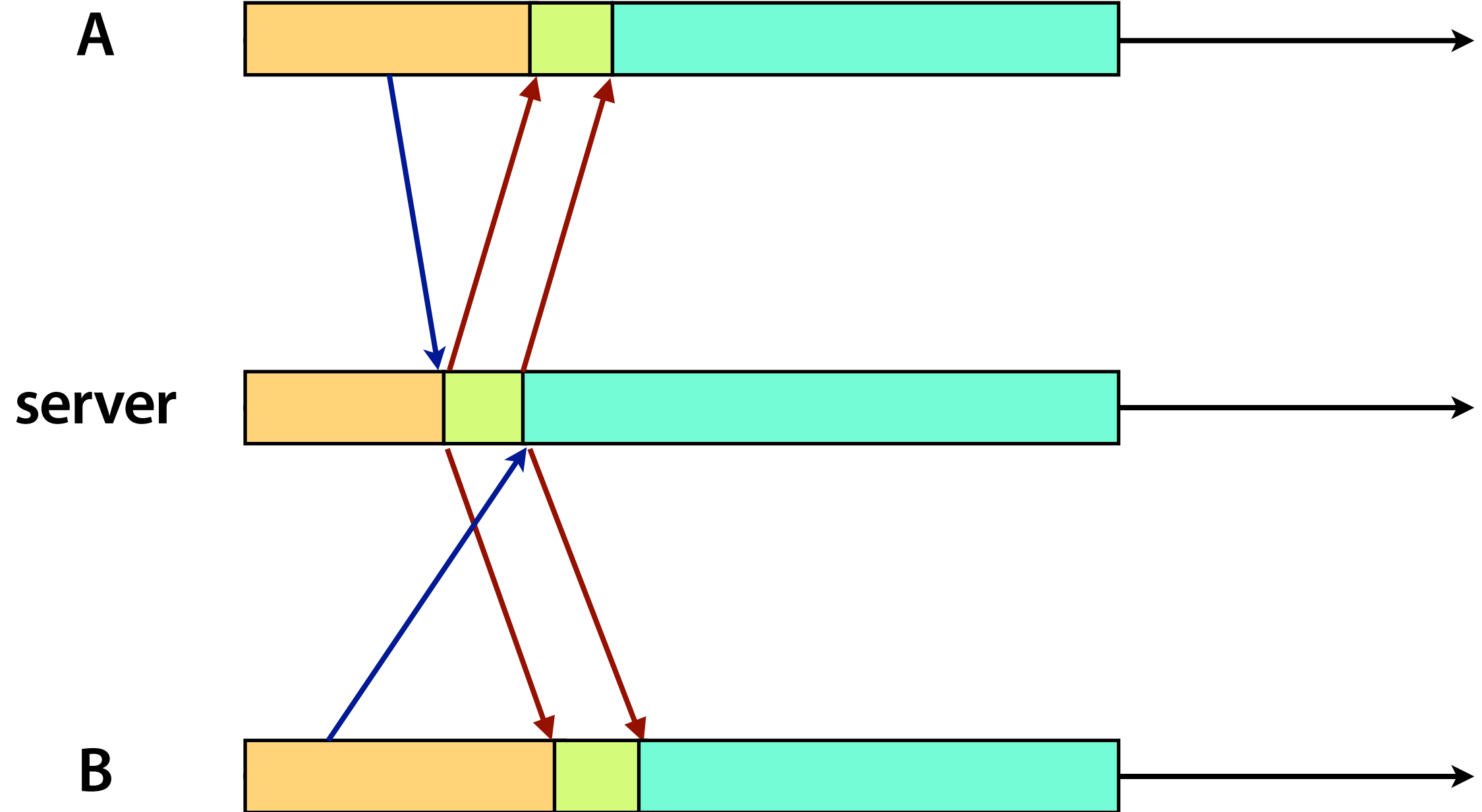
# Received-Order Delivery

Server executes the events as  
they are received.



# Unfair

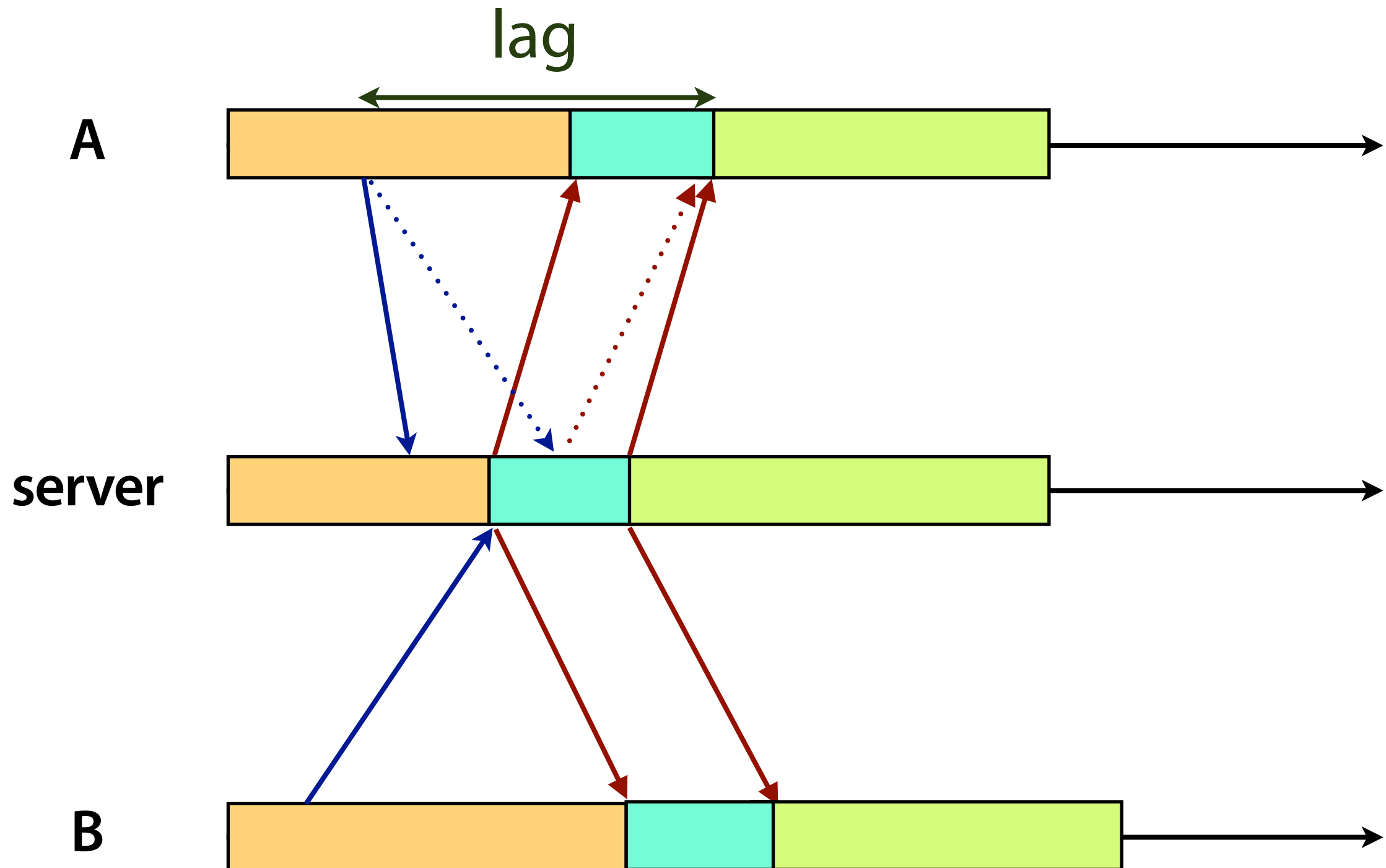
Different users experience  
different response time (aka lag)



# Idea:

# Artificial Server Delay

Equalize response time for all players by delaying the processing of events from players.



# Responsiveness

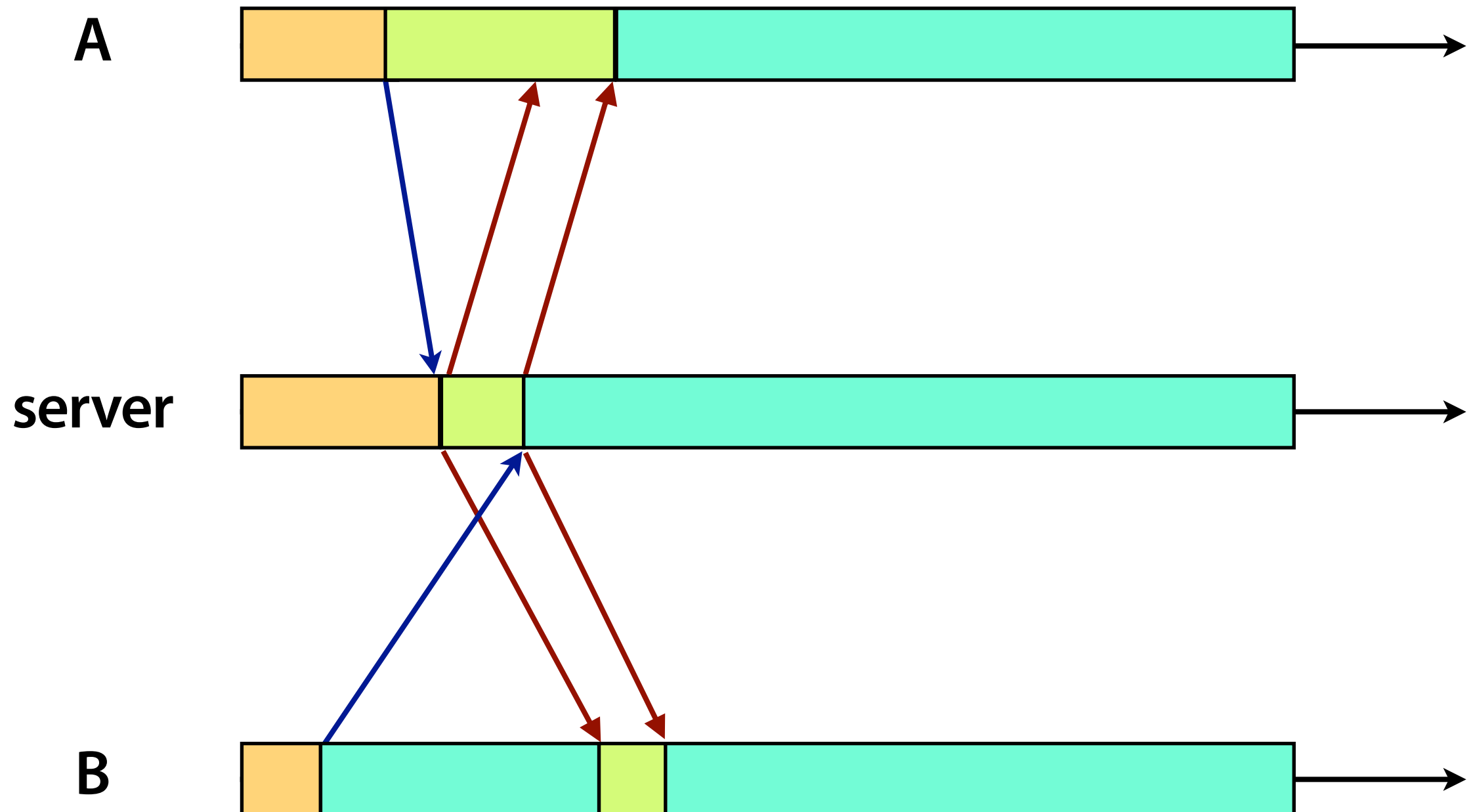
laggy game play annoys player

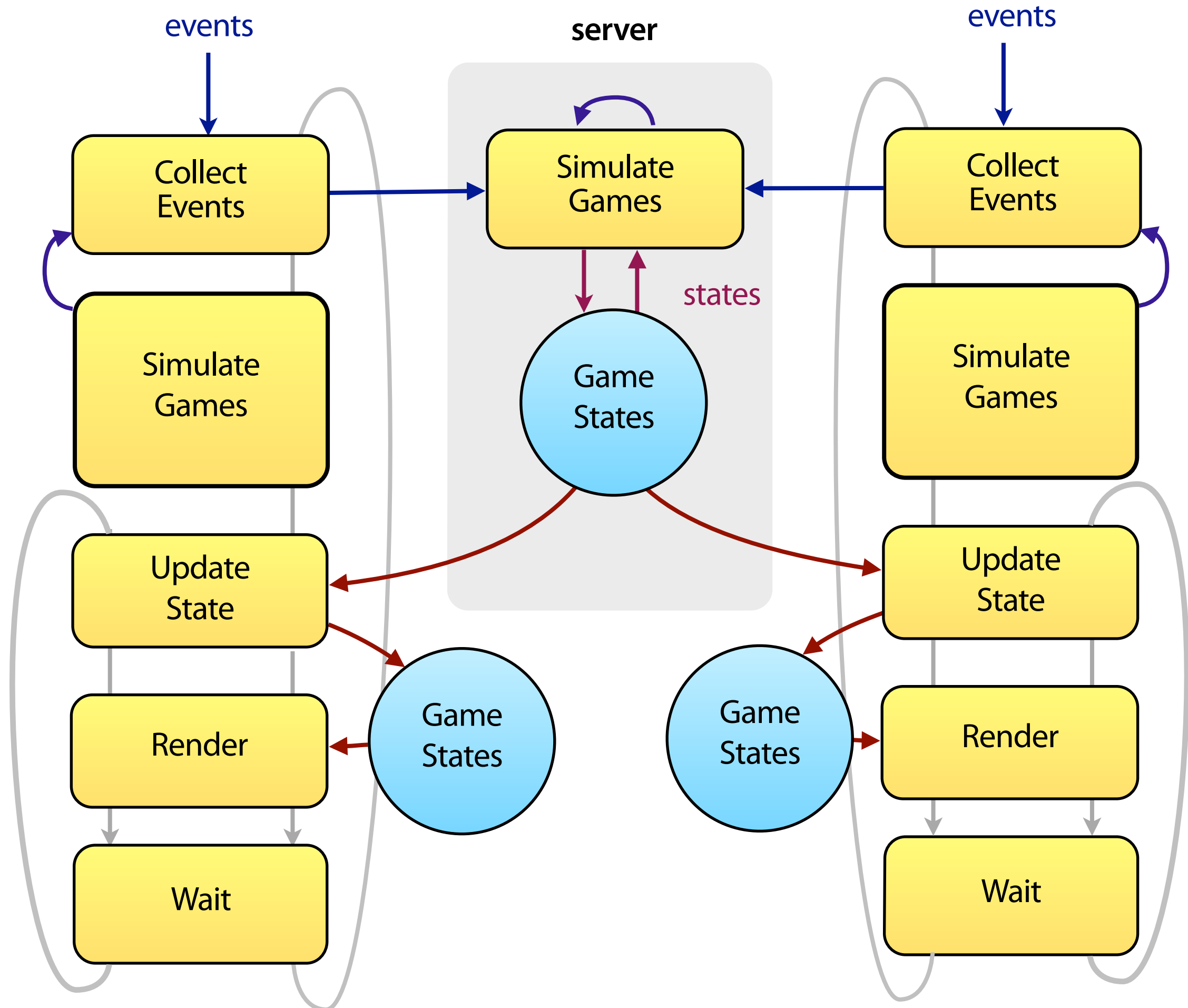


# Idea:

# Short Circuiting

Update states locally first, consolidate with server later.

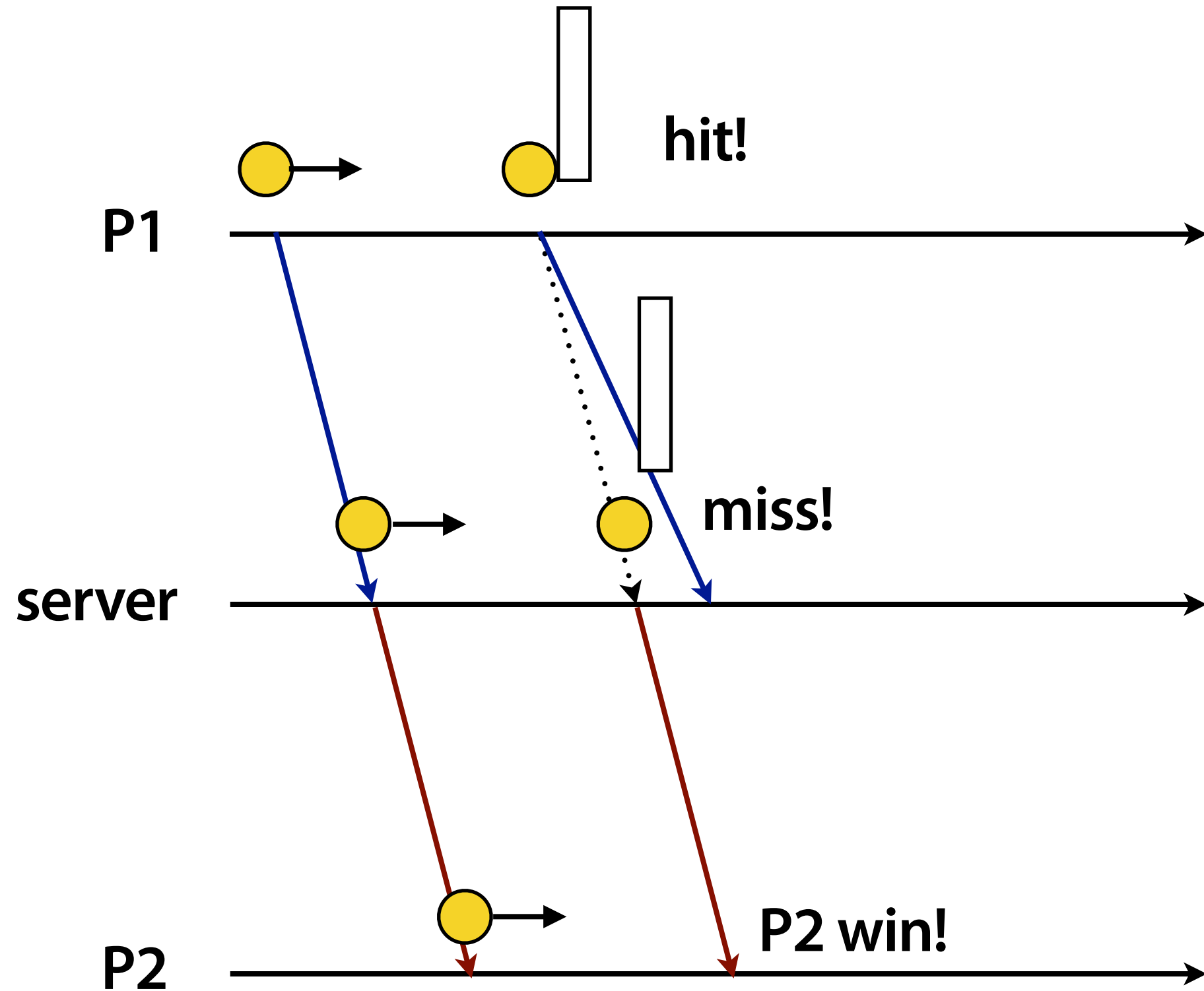




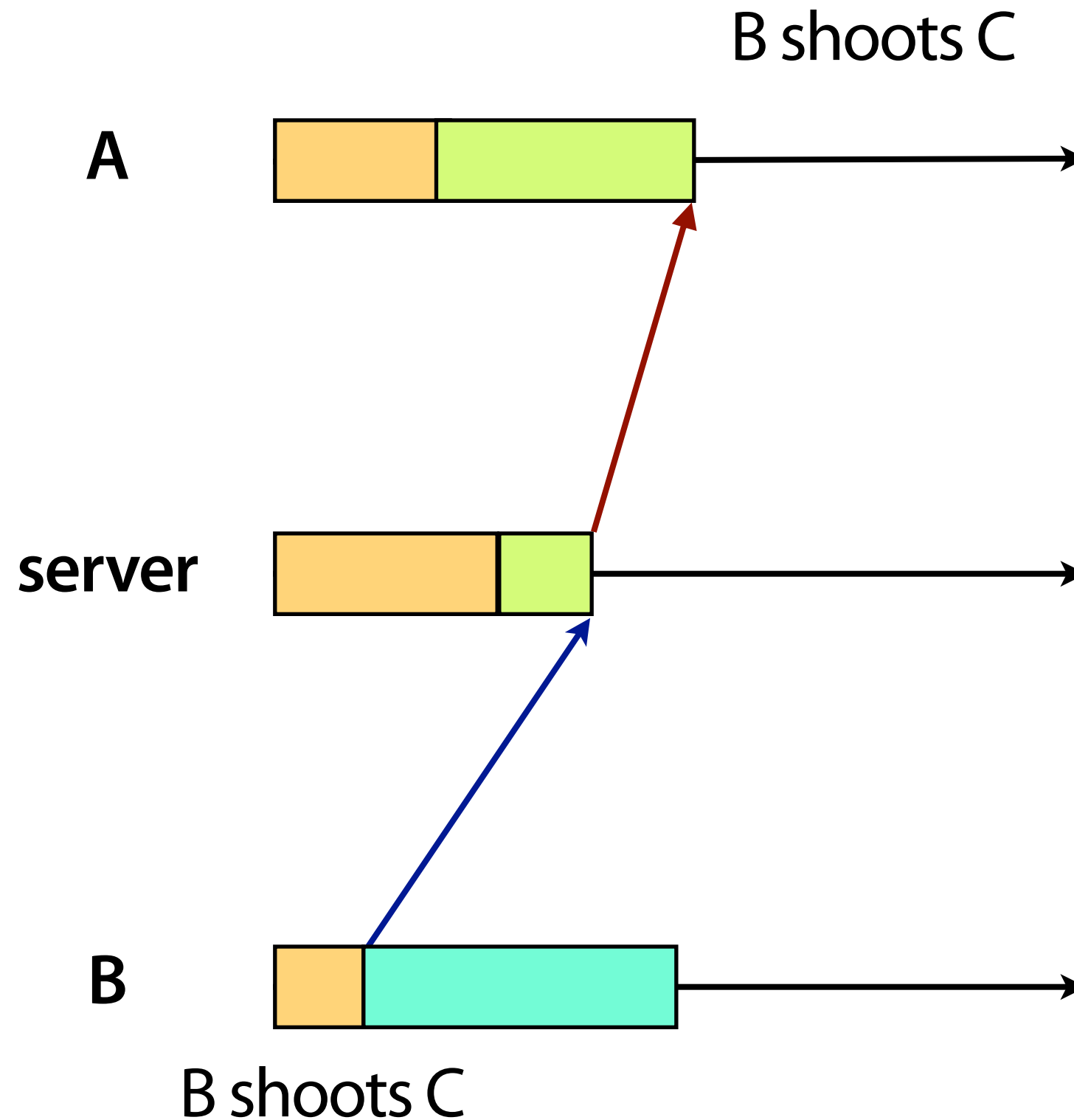
# Demo:

# Two-player Pong

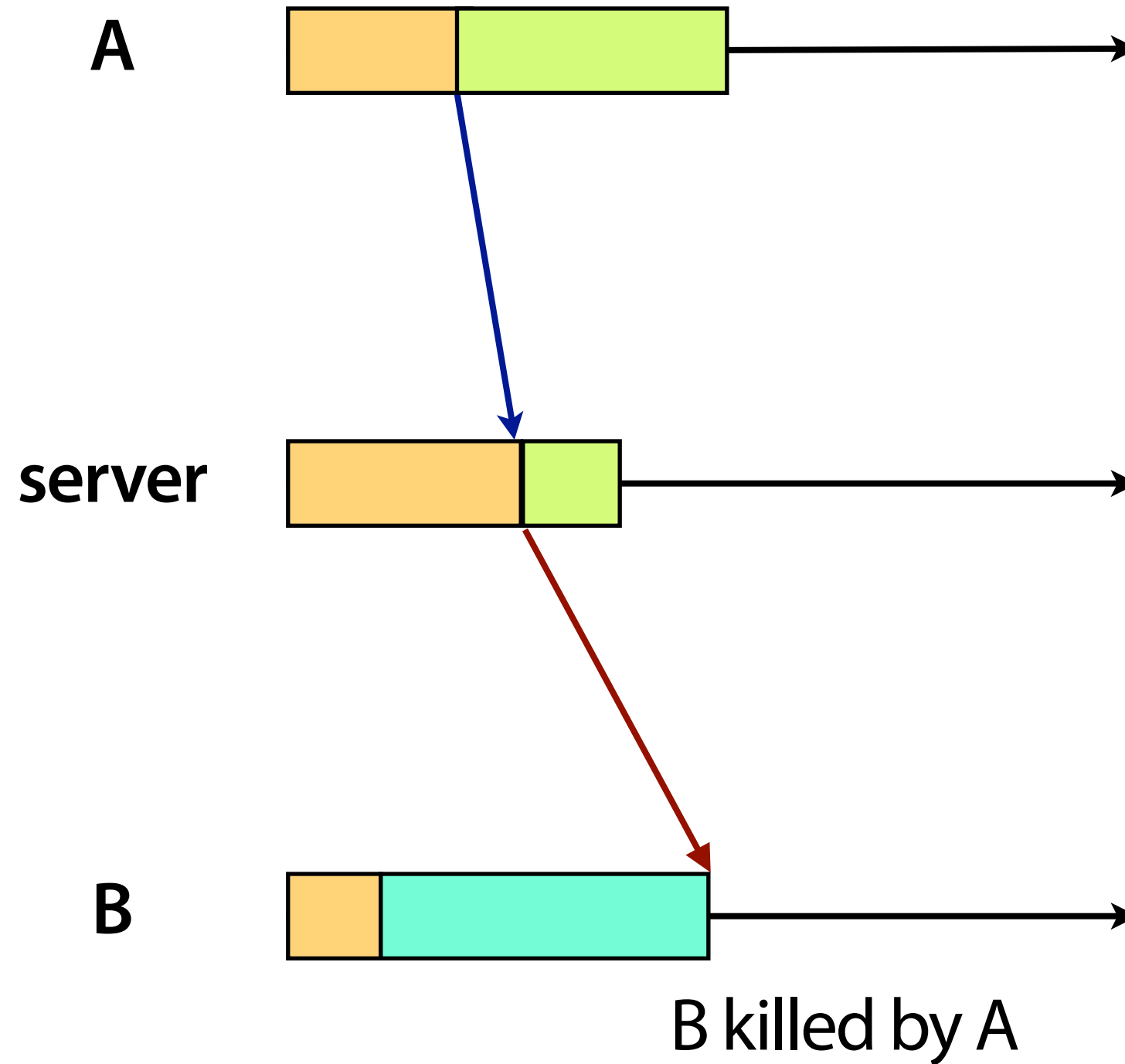
**What could go  
wrong?**



Consider a FPS..



A shoots B, B killed

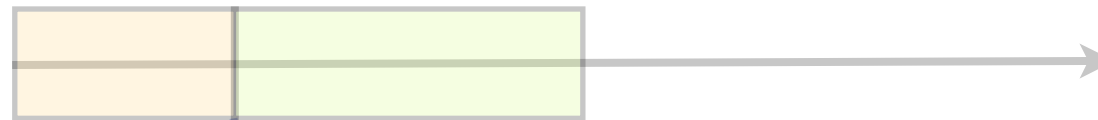




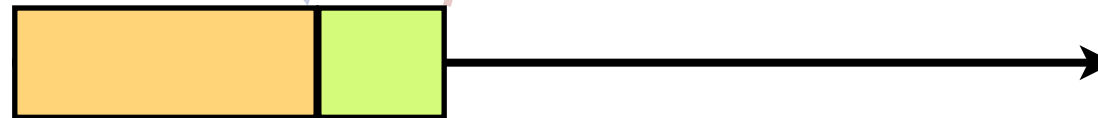
A shoots B, B killed

B shoots C

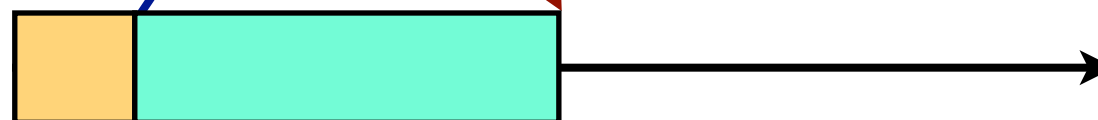
A



server



B



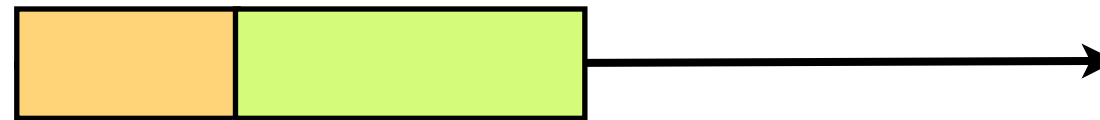
B shoots C

B killed by A

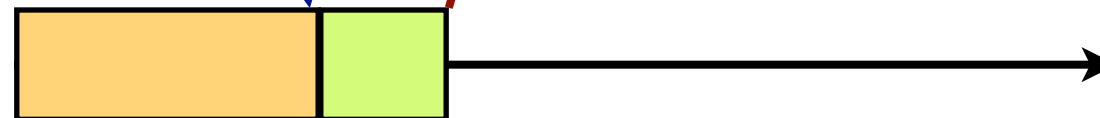
A shoots B, B killed

B shoots C

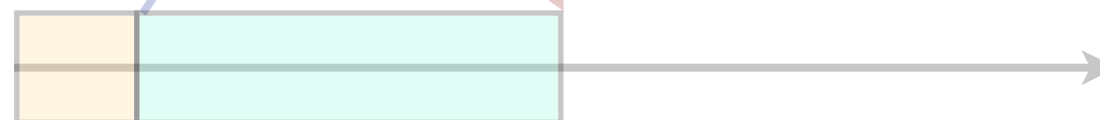
A



server



B



B shoots C

B killed by A

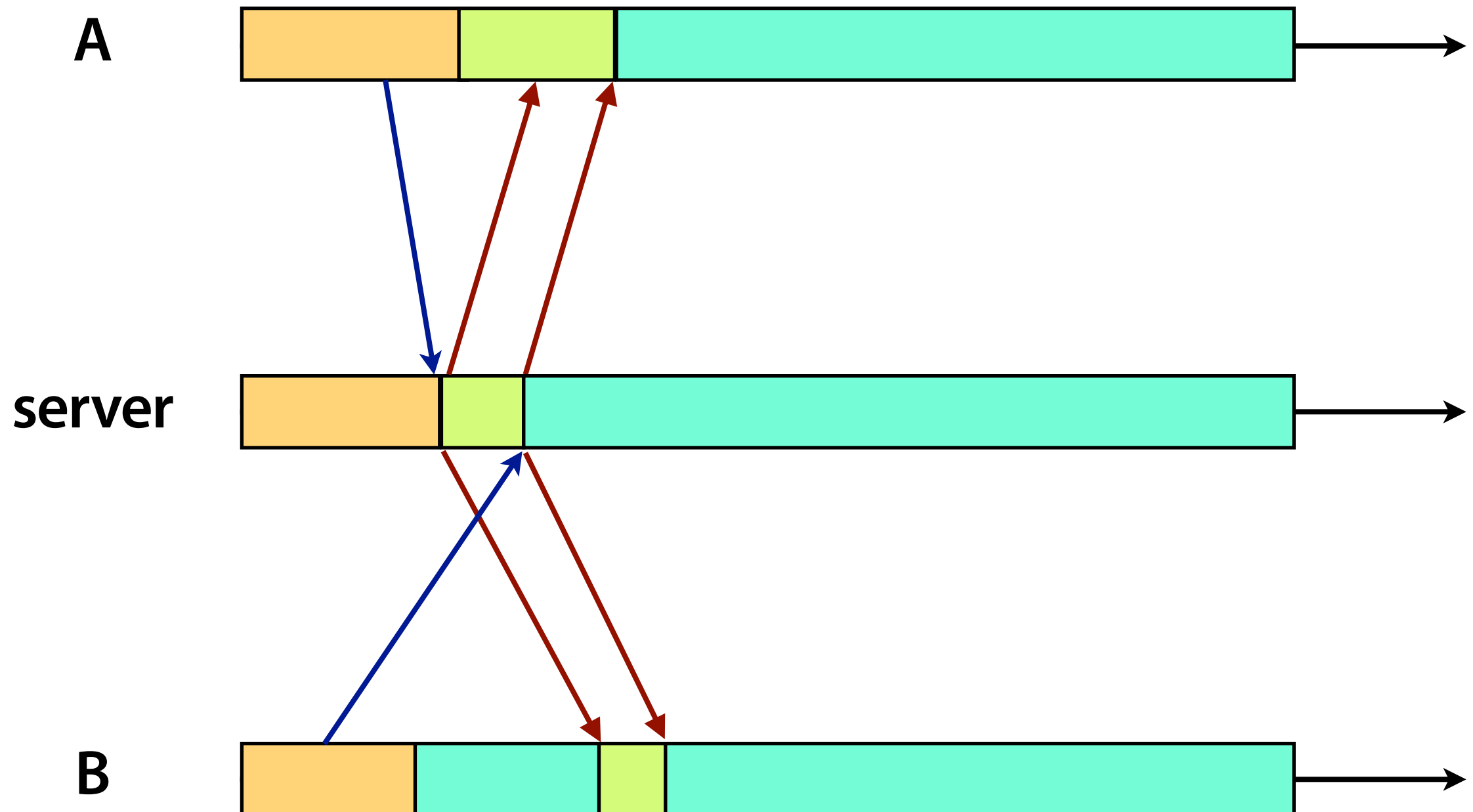
“A dead man that shoots”

# How to mitigate?

# Idea:

# Local Lag

# Update local state after some acceptable lag



Games can use audio/visual tricks to hide the lag.

What is acceptable lag?

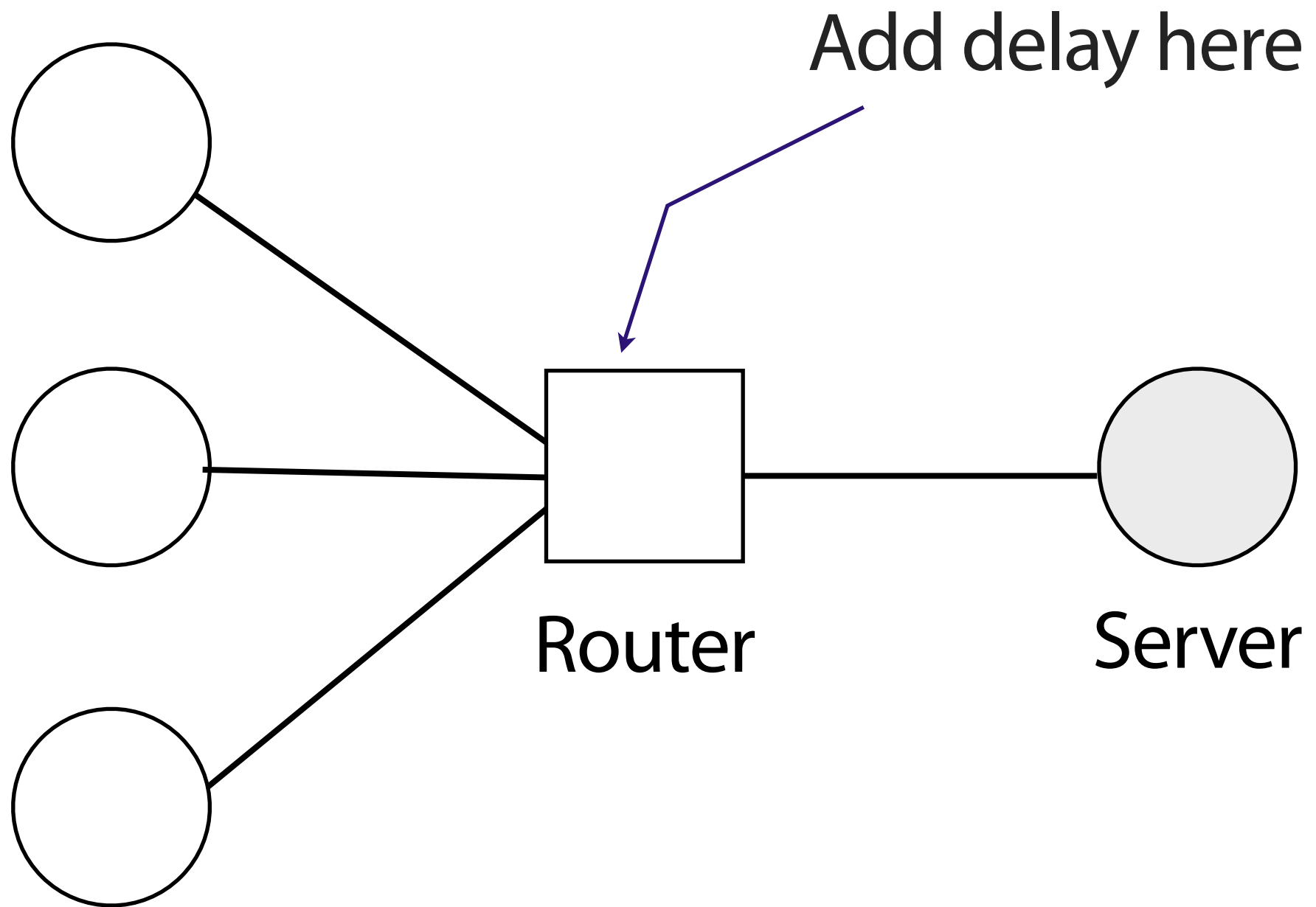


# User Studies on Acceptable Lag

**Goal:** How much lag is tolerable?

# **Method: User studies using Unreal Tournament 2003**

Clients



# **Game Activity:** move and shoot

# **Movement Test:**

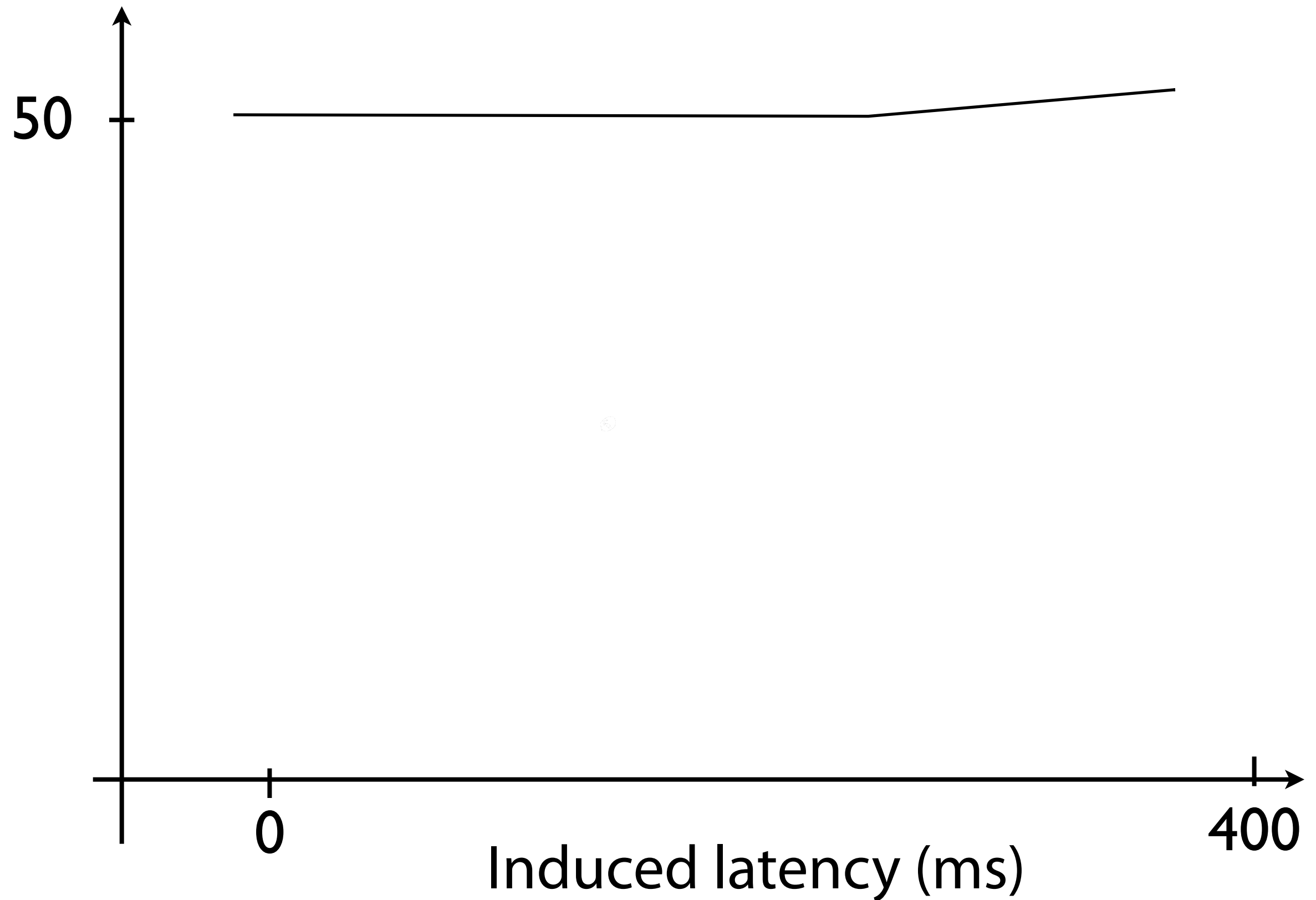
## Construct obstacle course



Over 200 users



# Time to complete course (s)



# **Shooting Test:**

Two players shooting at  
each other using  
precision weapon


1 1 2



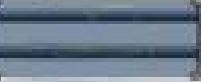
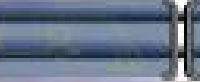



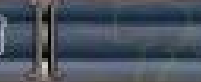


10



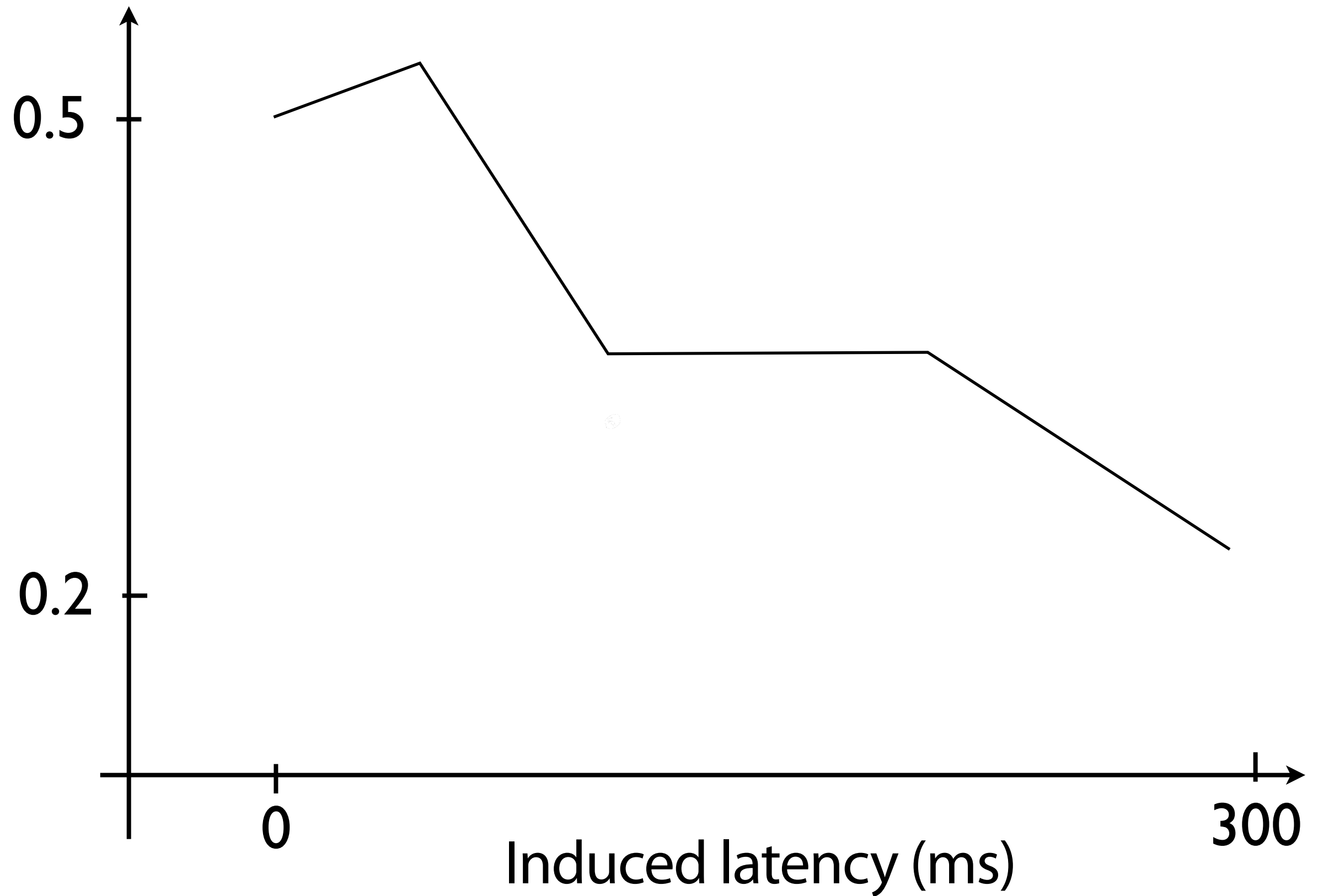
100

12





# Hit Fraction



“latency as low as **100 ms** were noticeable and latencies around **200 ms** were annoying”

Read the paper for complete results.

Other conclusion: loss rate up to 5% has no measurable effects.

# **Method: User Studies using Warcraft III**

**Game Activity:** build,  
explore, fight!



**Finding:** Players with larger delays see exactly the same events as players with smaller delays, only at a later time.

**Finding:** Latency of up to  
**800 ms** has negligible effect  
on the outcome of Warcraft  
III.

**Finding:** Latency of up to  
**500 ms** can be  
compensated by the players

**Finding:** Latencies  
between **500** and **800** ms  
degrade game experience.

Strategy is more important  
in RTS games, not reaction  
time.

Q: What is the acceptable lag?

**A: Depends on the characteristics of game.**

# Assignment 2

## Task 1

Find the acceptable lag for Pong.