You are not alone

Scaling Multiplayer Games

Jesper Richter-Reichhelm - @jrirei
Knut Nesheim - @knutin

Wooga


481.150
27
DIAMOND
DASH
Play


481.150
27
DIAMOND DASH
Play
Monster World


481.150
27
DIAMOND
DASH
Play


Monster World


7
Level 2
Stage 2
35
Level Score


481.150
27
DIAMOND
DASH
Play


Monster World


7
Level 2
Stage 2
35
Level Score

> 1,000,000 daily users

> 1,000,000 daily users

> 50,000 concurrent users

> 1,000,000 daily users

> 50,000 concurrent users

> 10,000 requests / second

Ok, you can handle that.

Ok, you can handle that.

But why only single player?

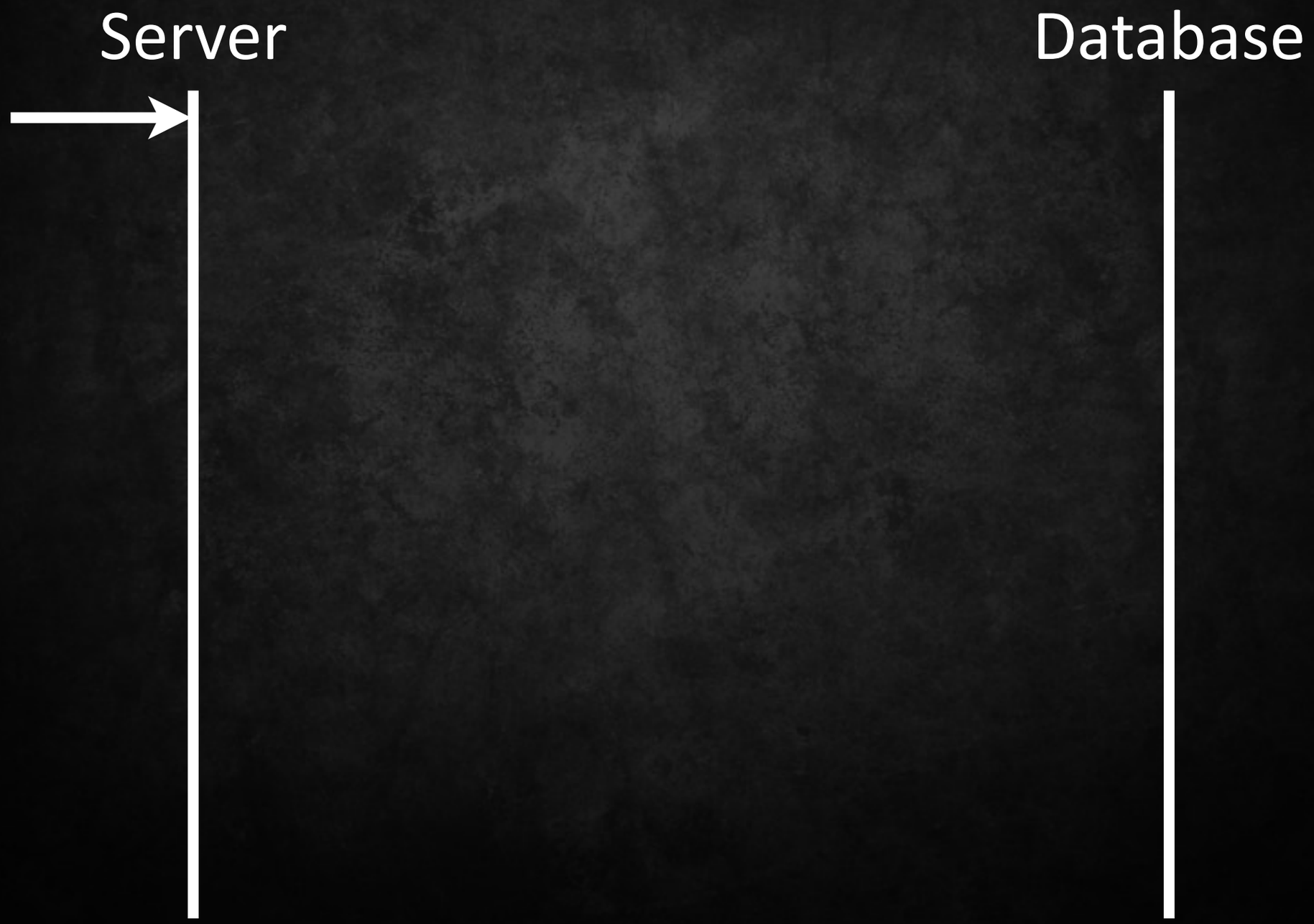
The Games

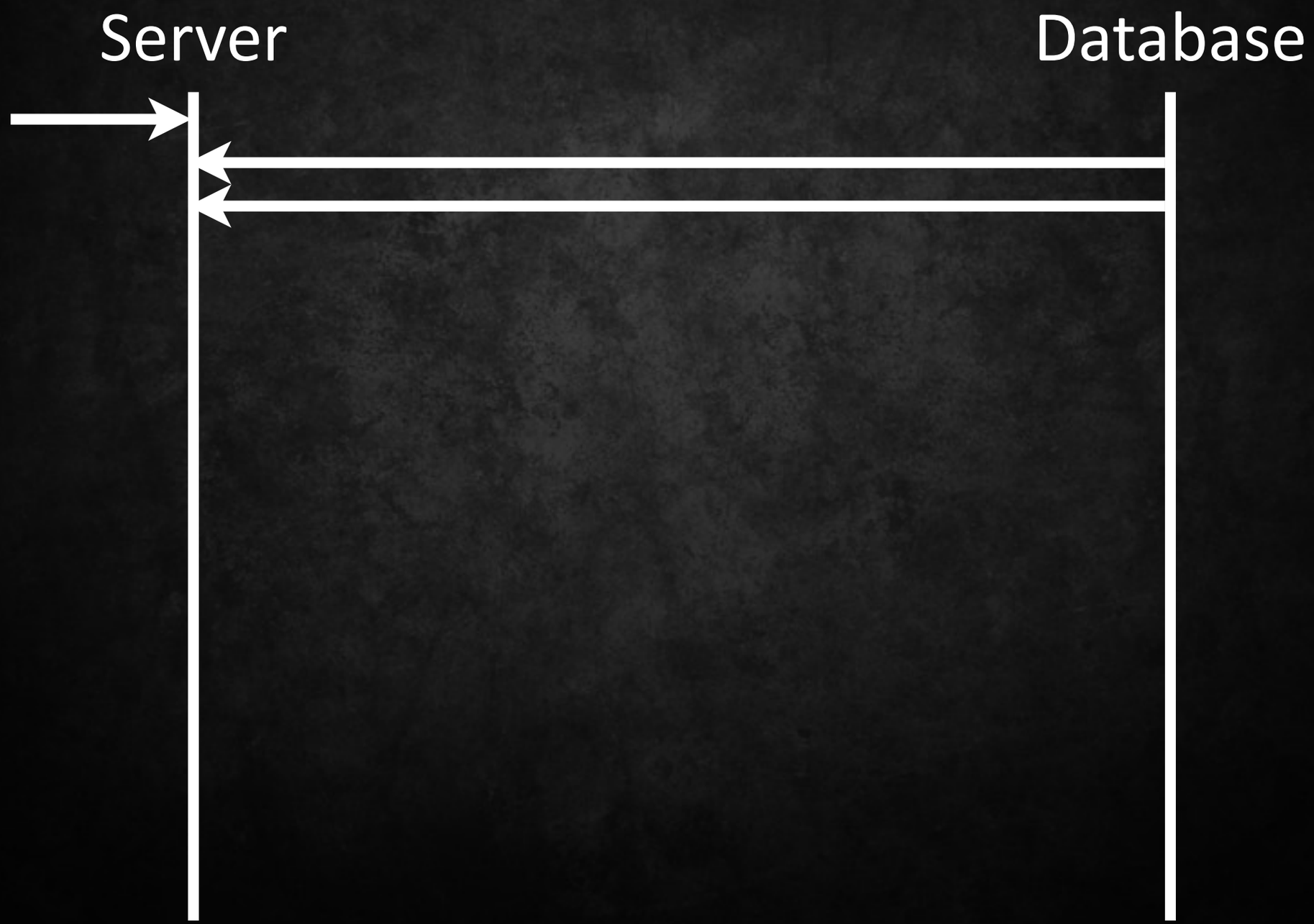
The Past

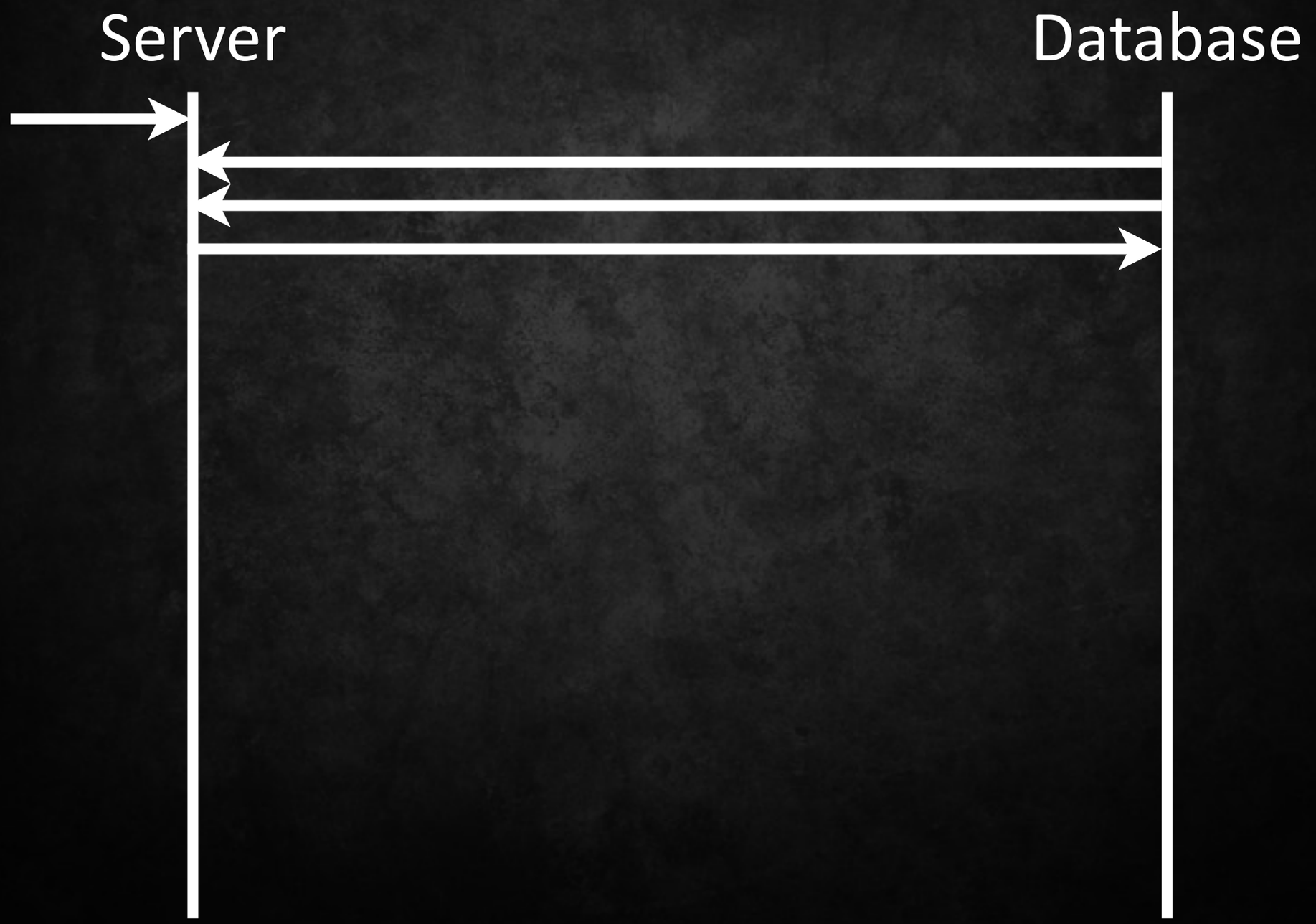
The Present

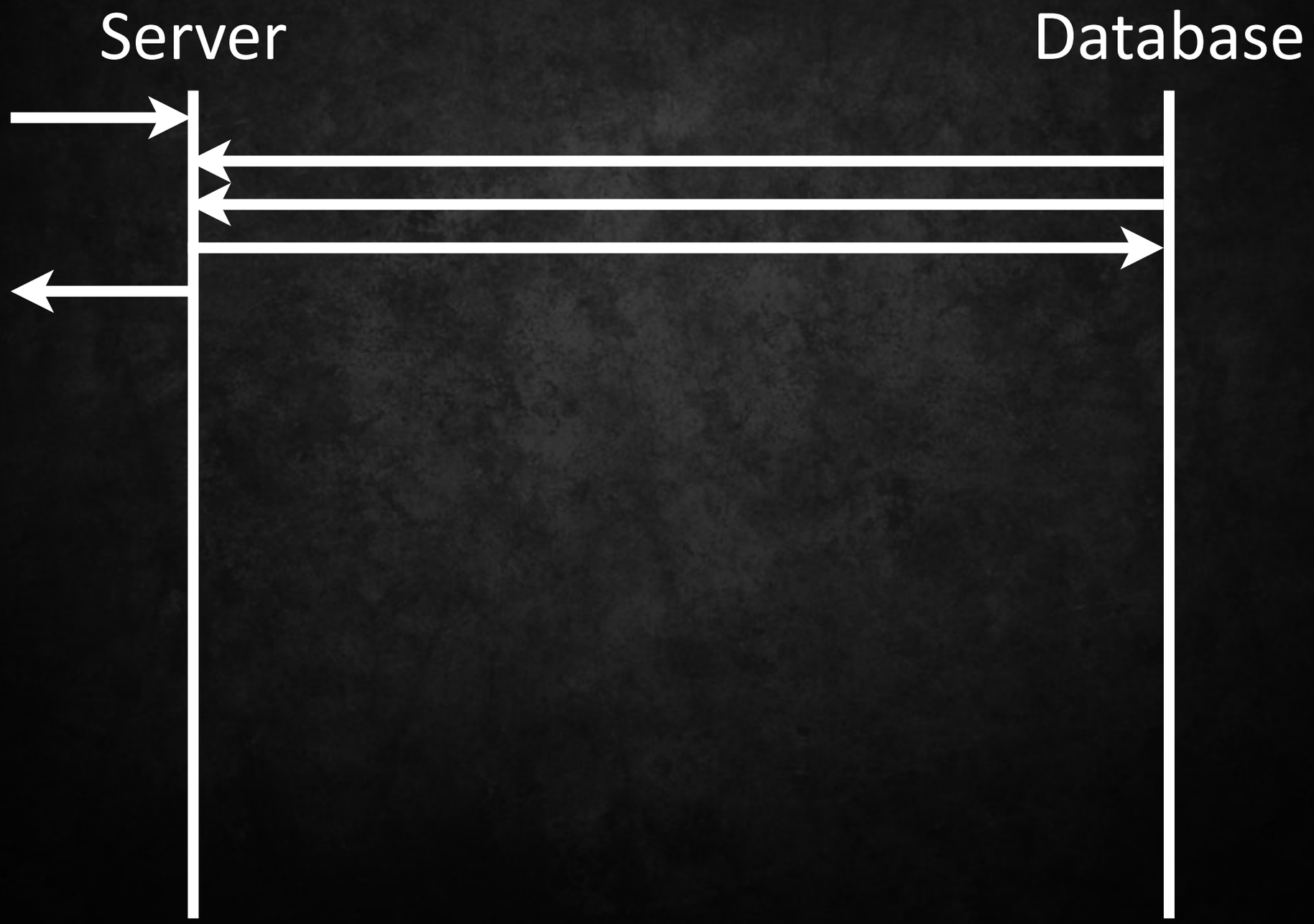
The Future

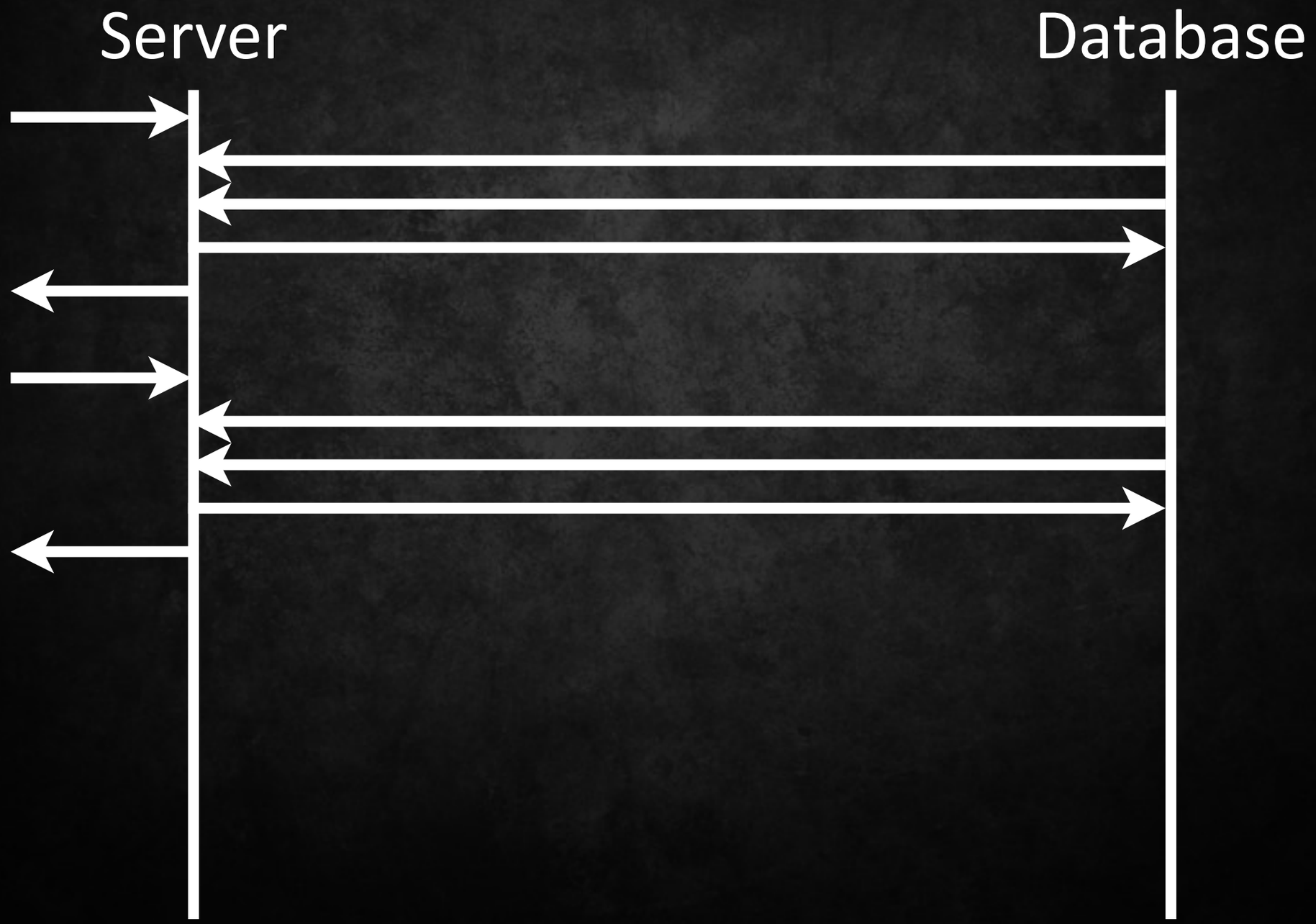
The Foundation

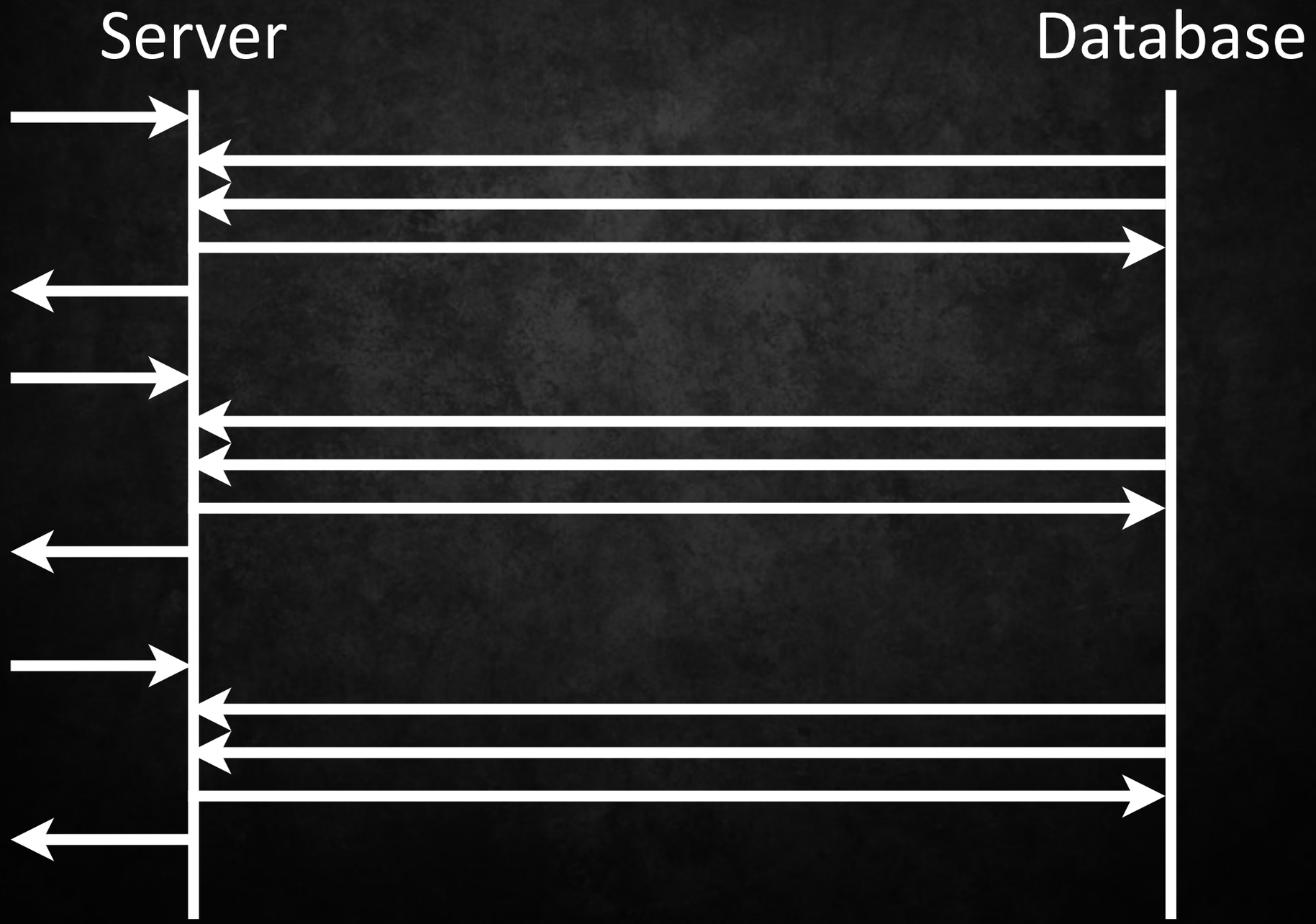

Server
Database


Server
Database


Server
Database

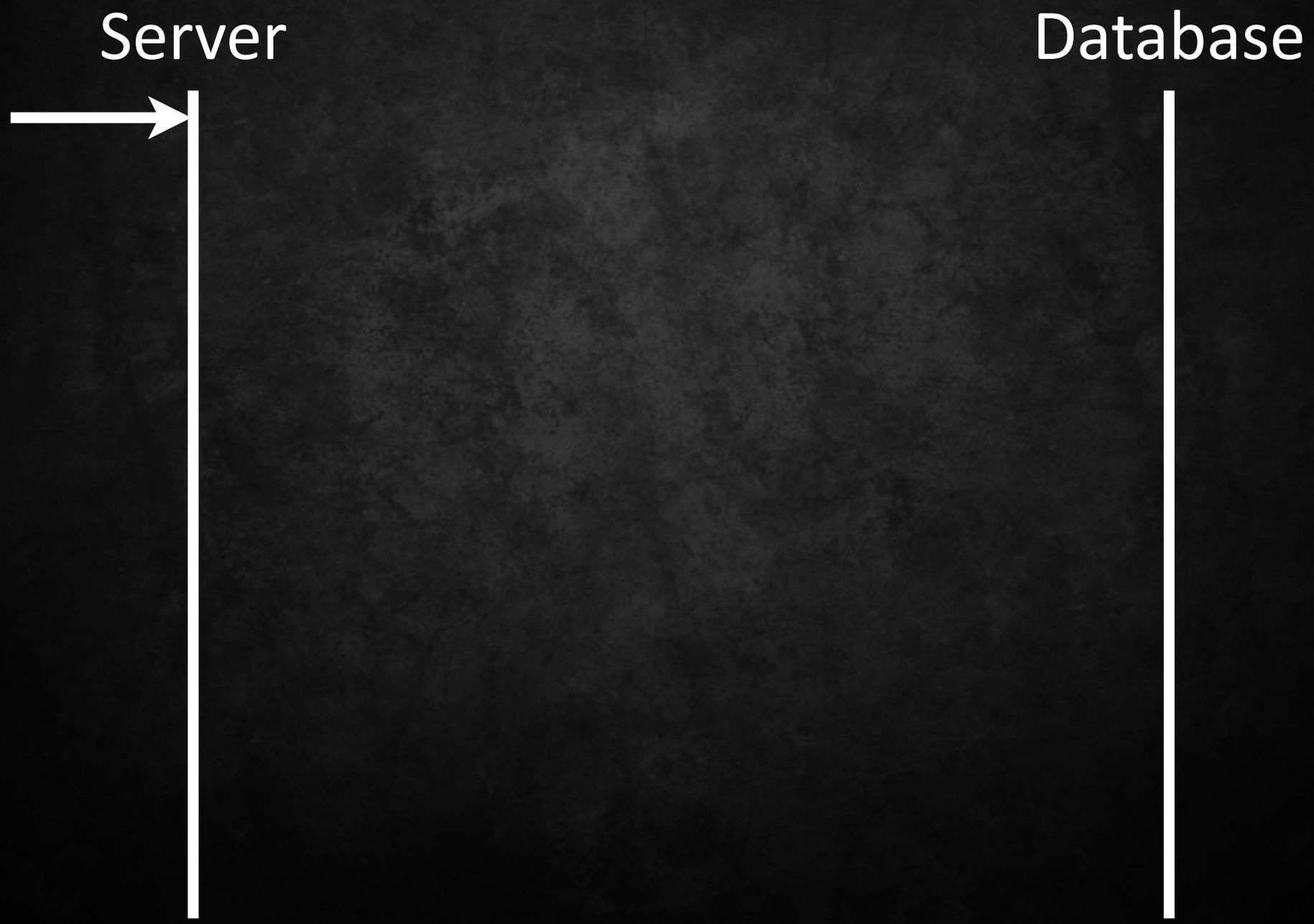

Server
Database

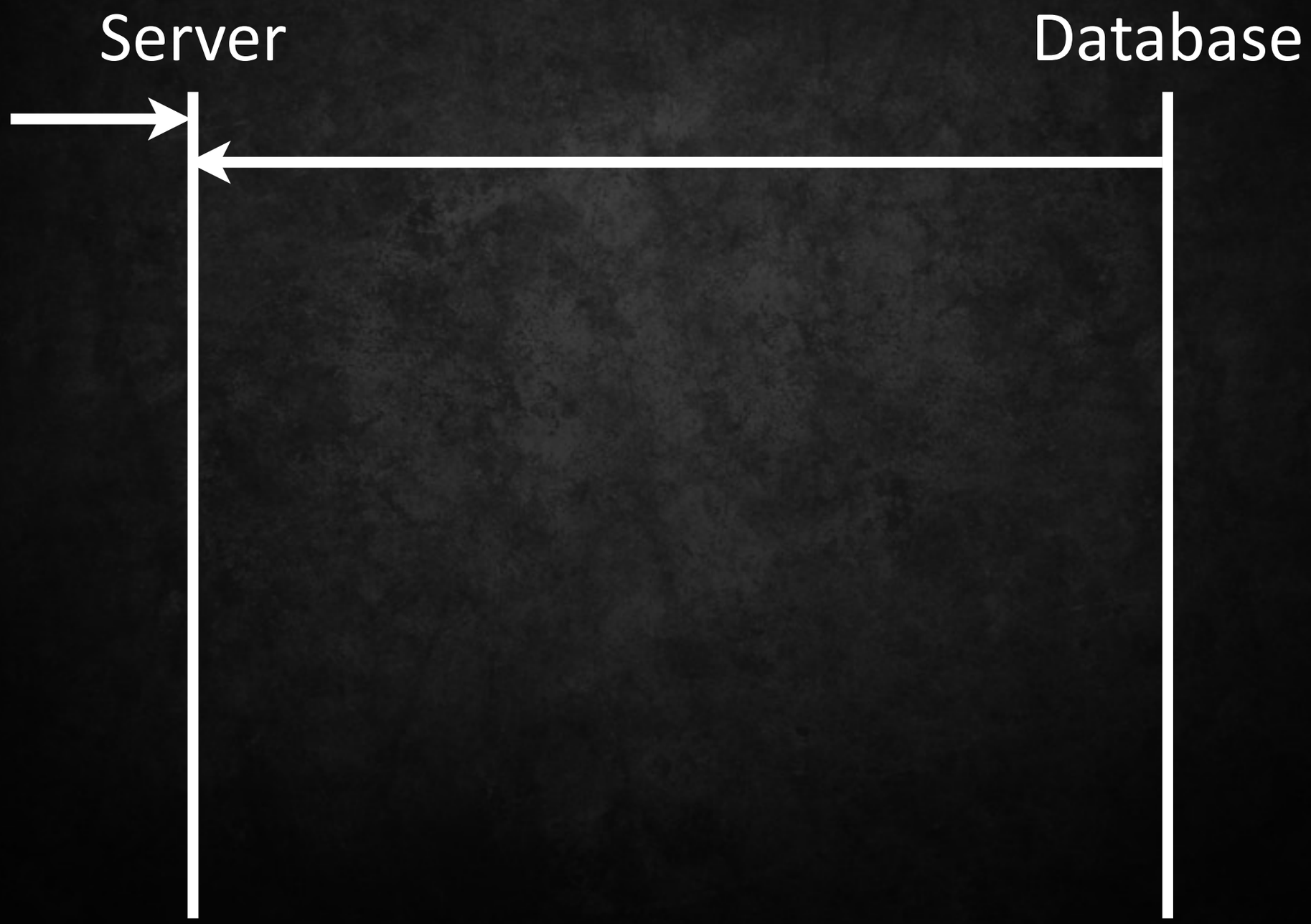

Server
Database

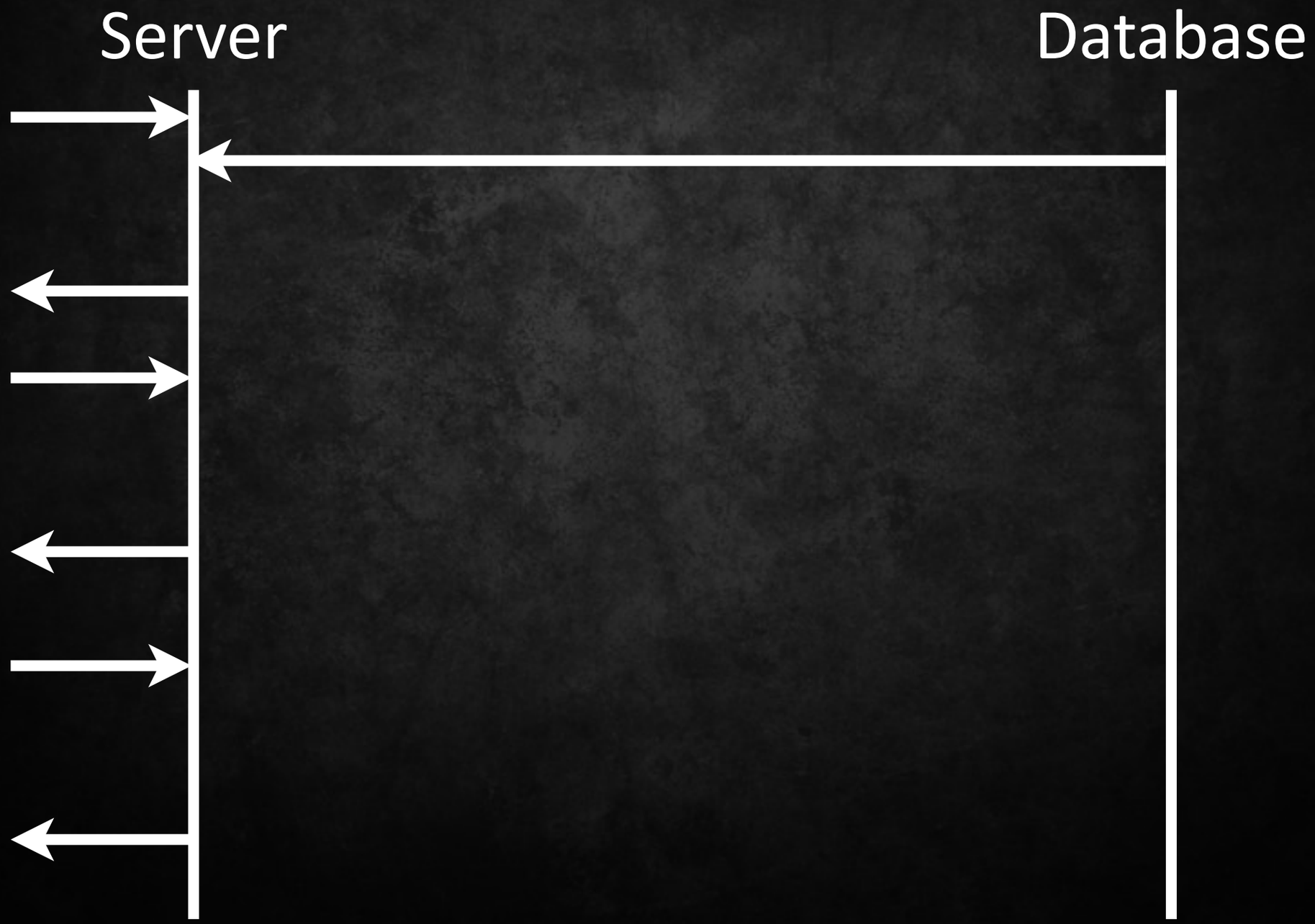

Server
Database

DB is the bottleneck

Magic Land


Server
Database


Server
Database

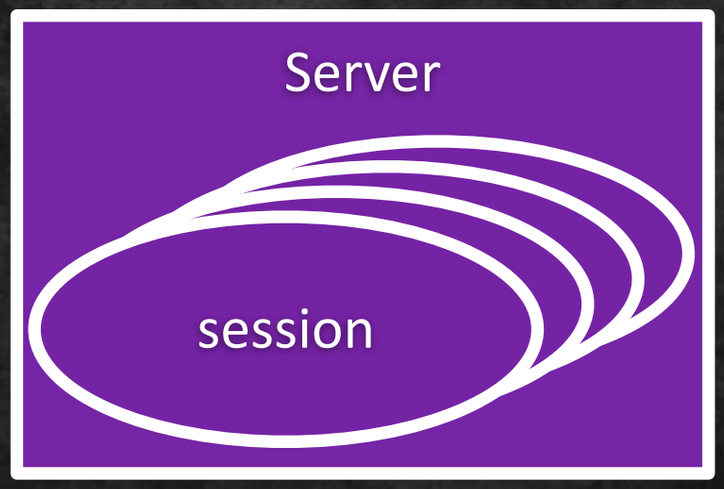

Server
Database

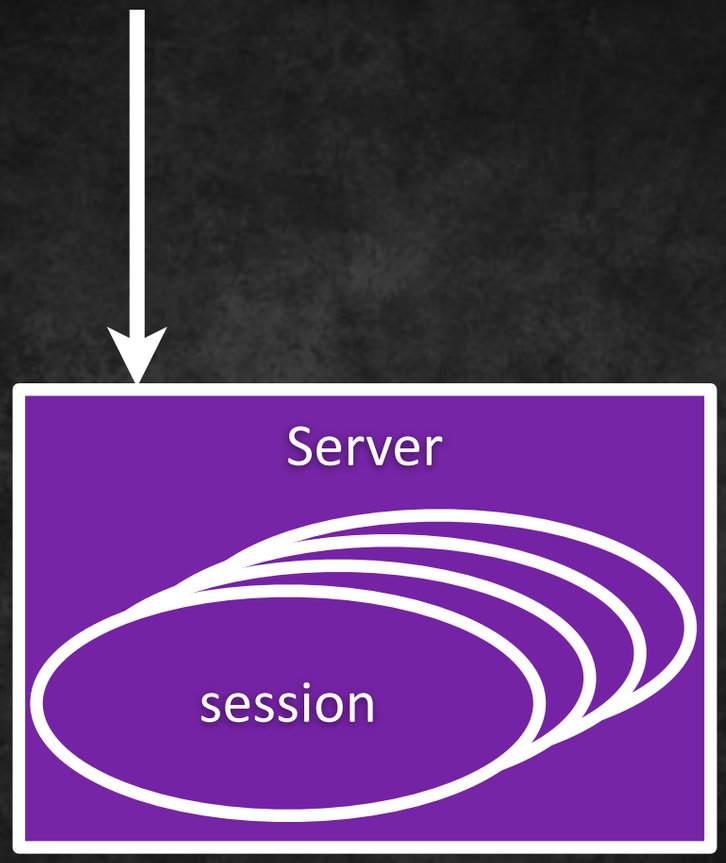

Server
Database
One Game Session

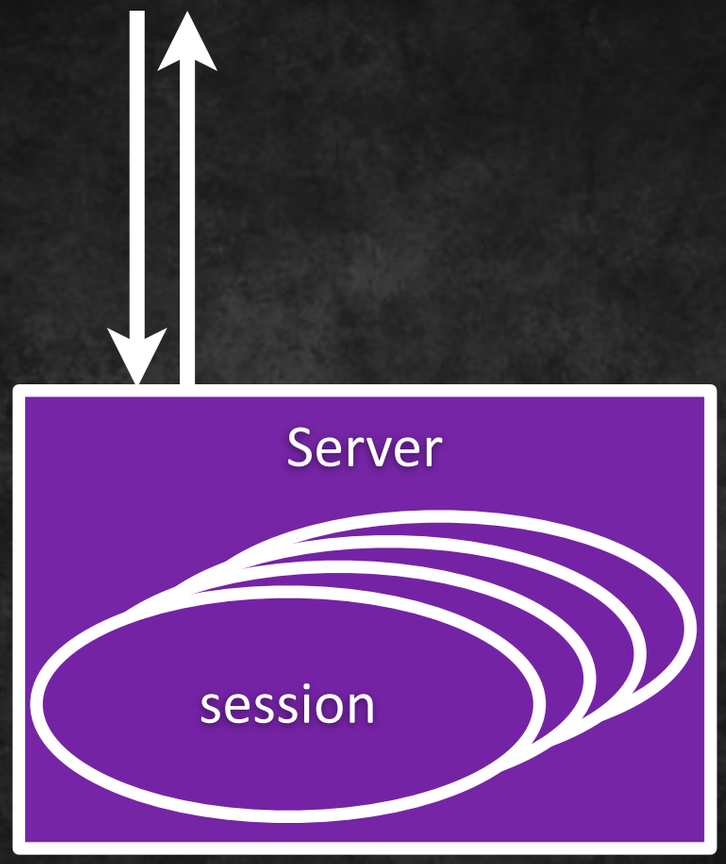

Server
Database
One Game Session

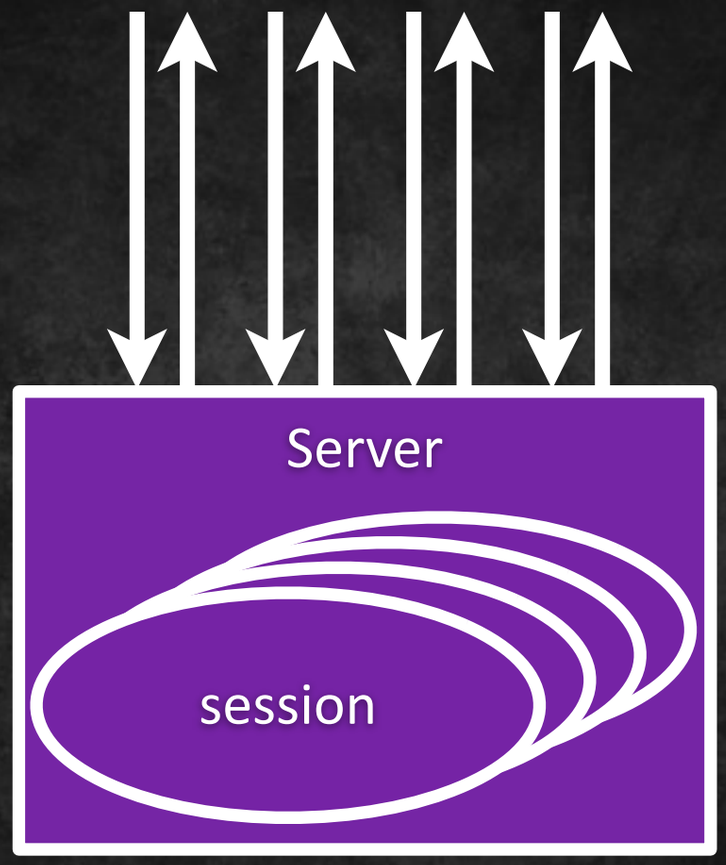

session

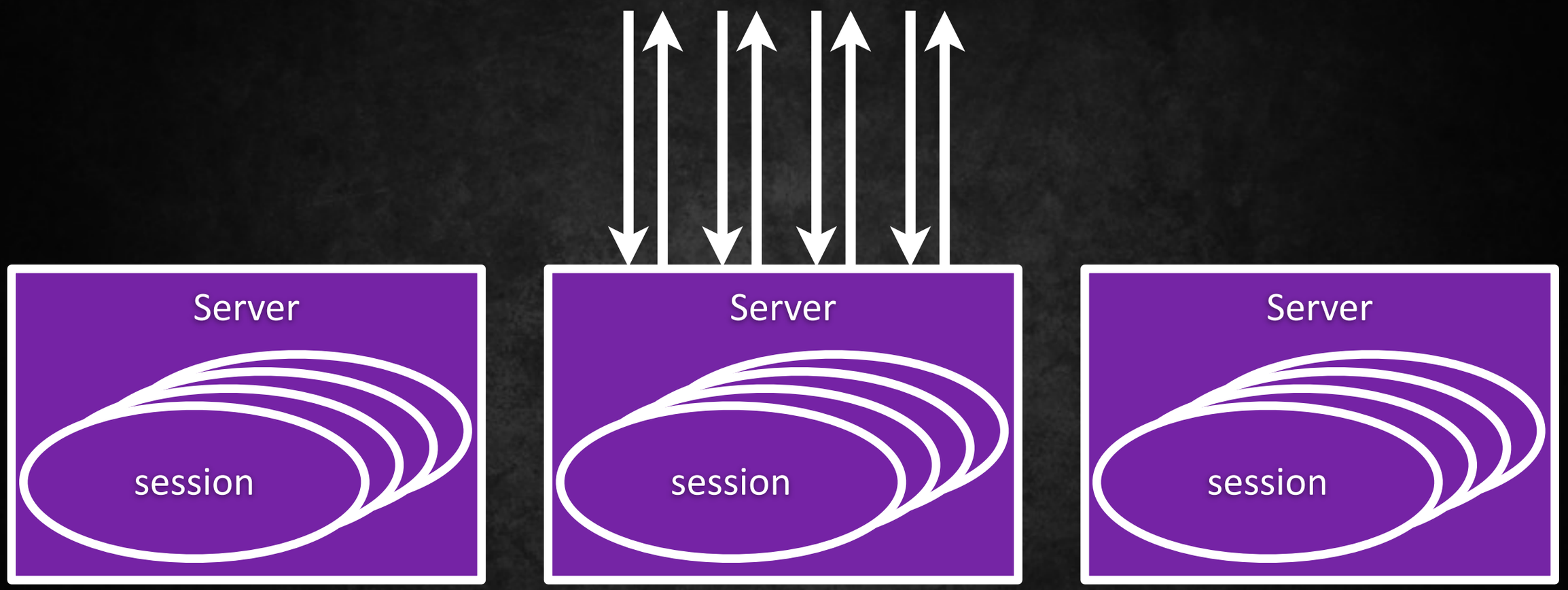

session


Server
session


Server
session


Server
session


Server
session


Server
session
Server
session
Server
session

The Games

The Past

The Present

The Future

The Foundation

es

t

nt

re

tion

Thoughts

Client

Server

Operation

es

t

nt

re

tion

Thoughts

Client

Server

Operation

Game

eight

You are not alone

Pessimistic

Pessimistic

Schizophrenic

Pessimistic

Schizophrenic

Optimistic

Pessimistic

Client
A

Server
A

Client
B

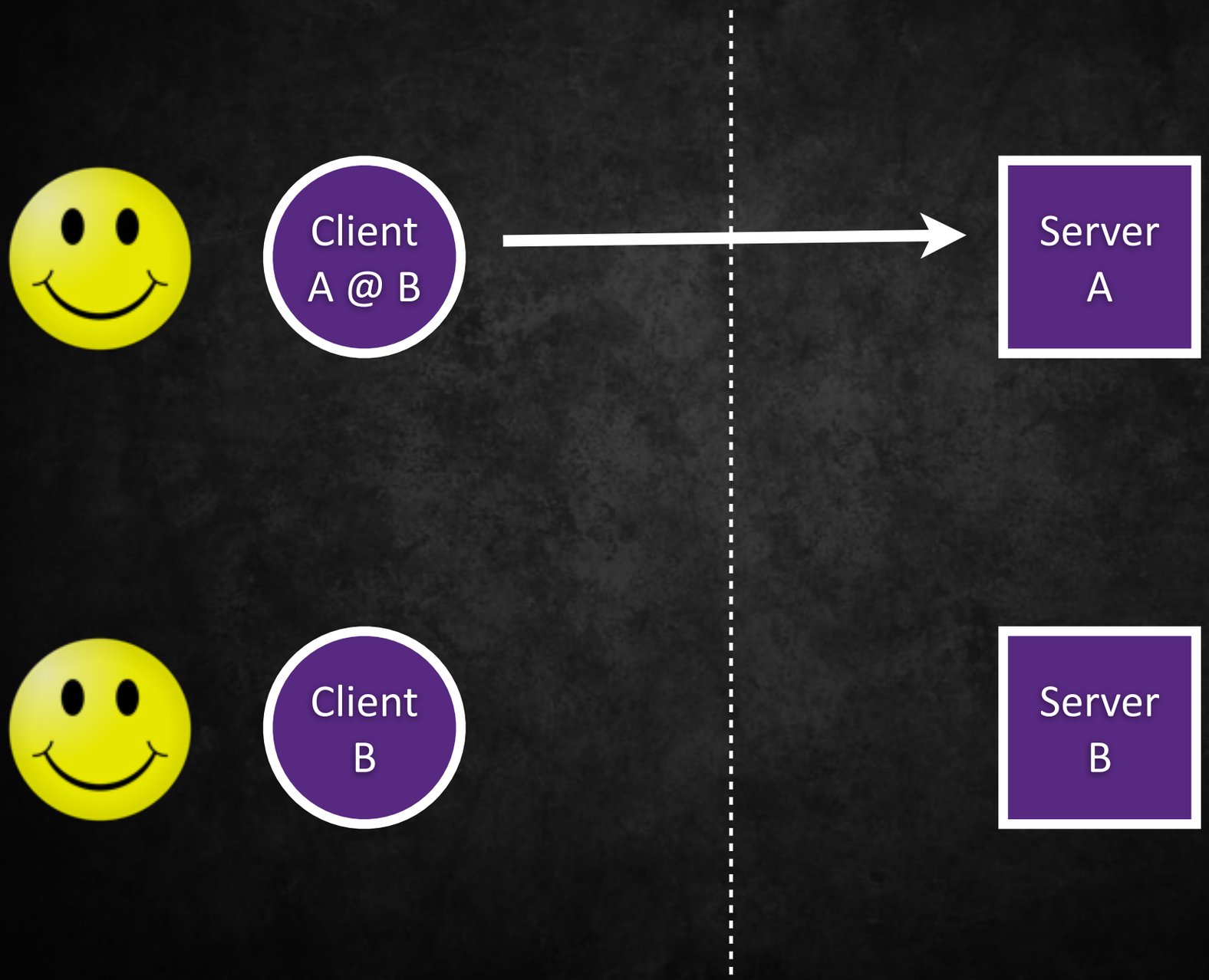
Server
B

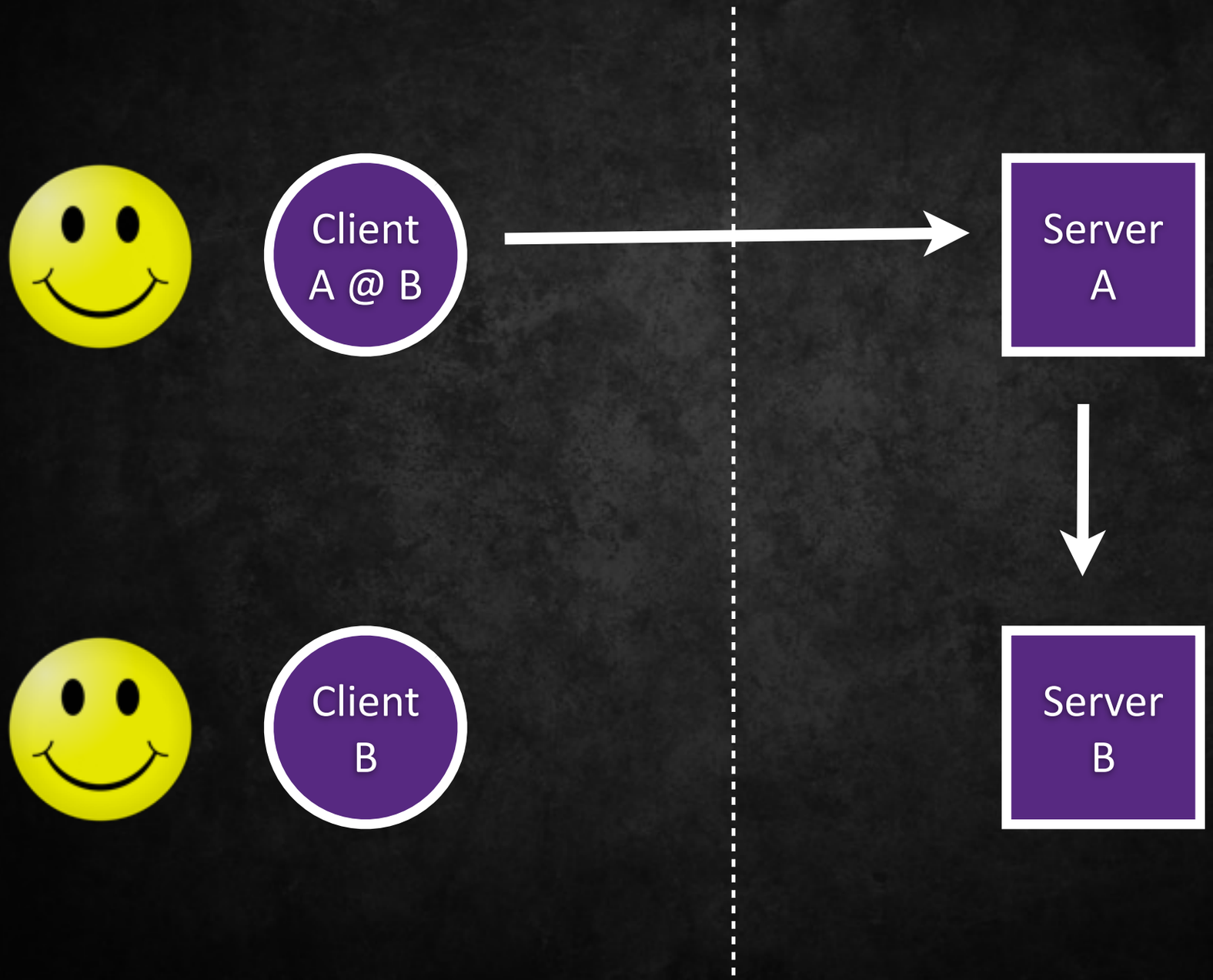
Client
A @ B

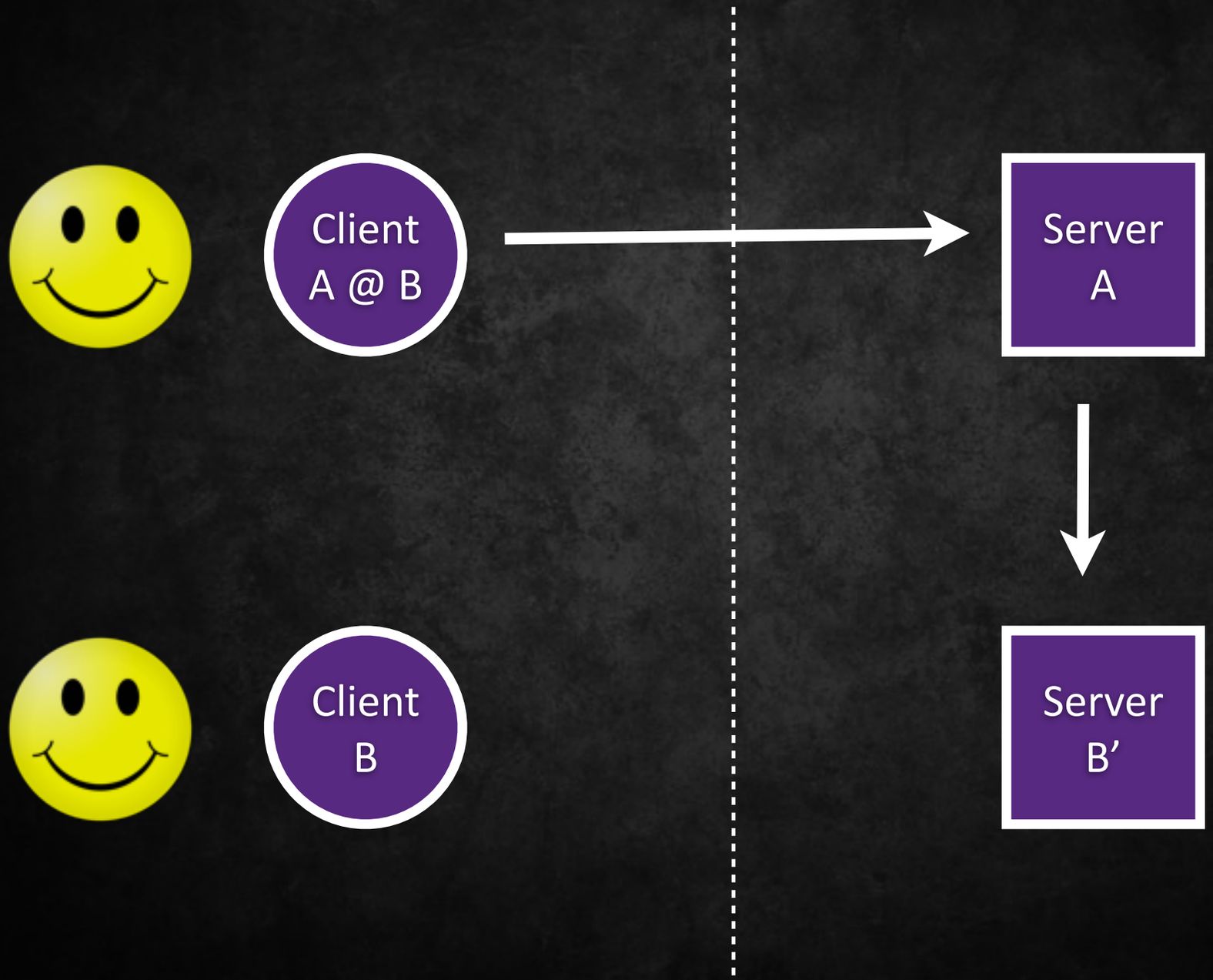
Server
A

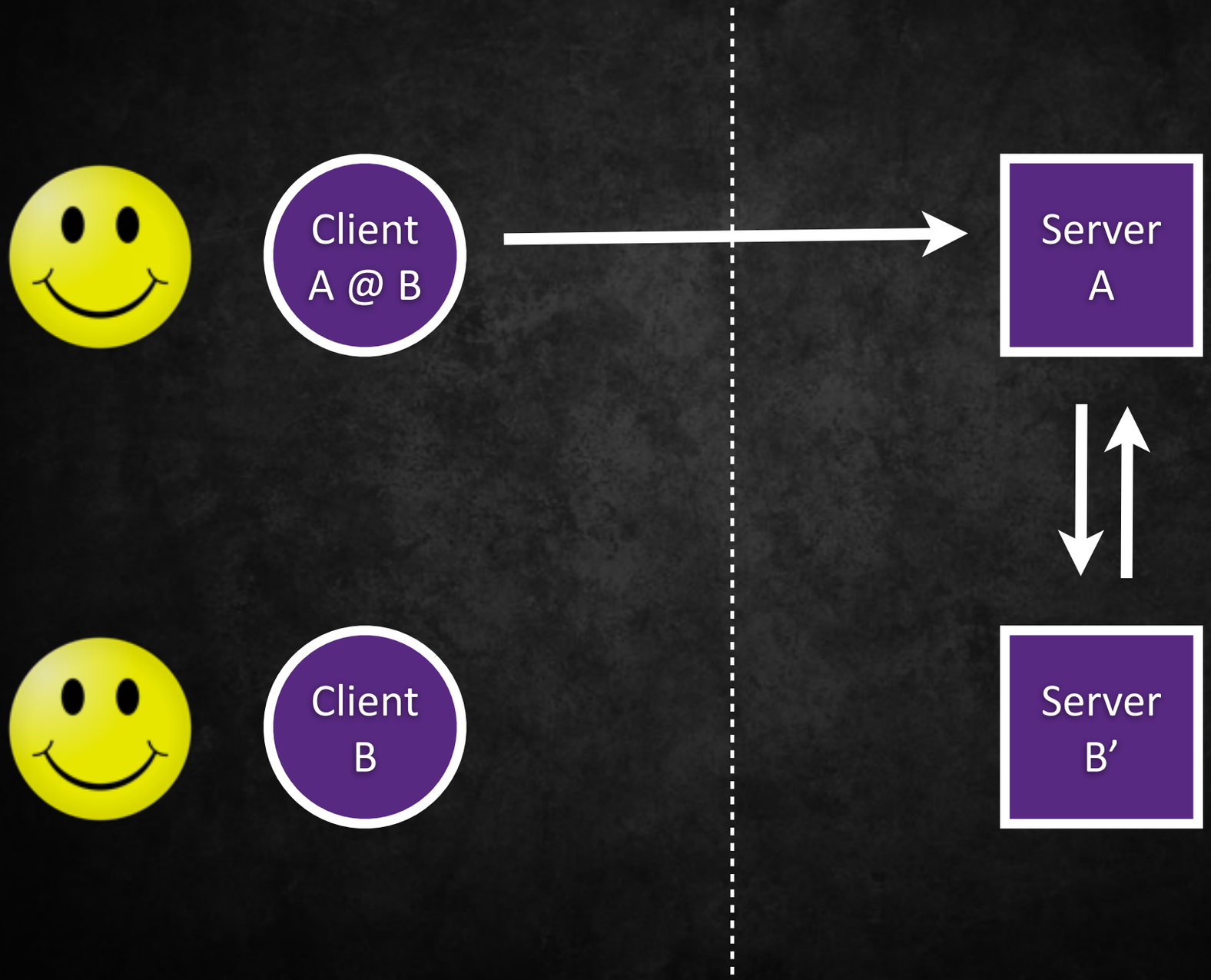
Client
B

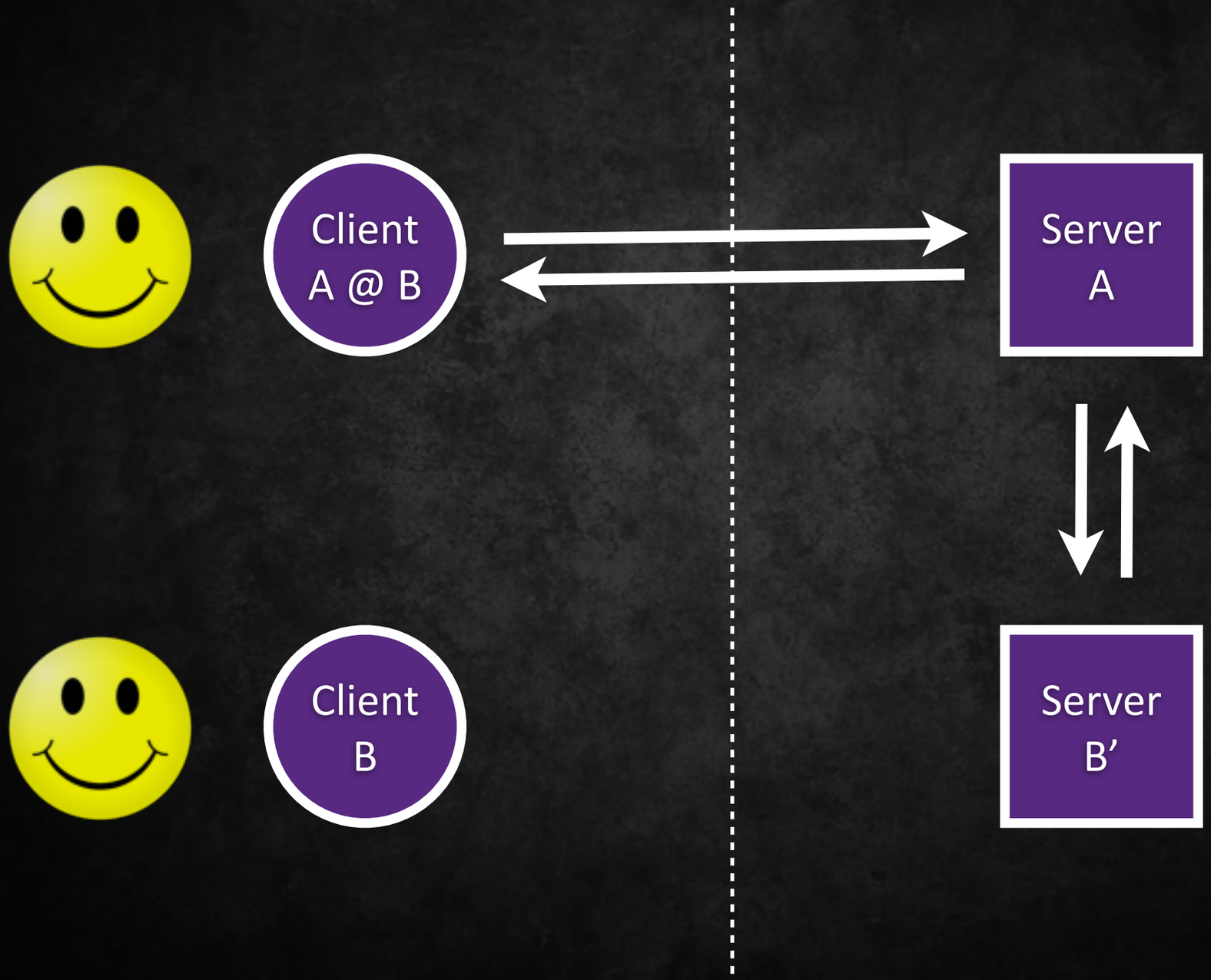
Server
B

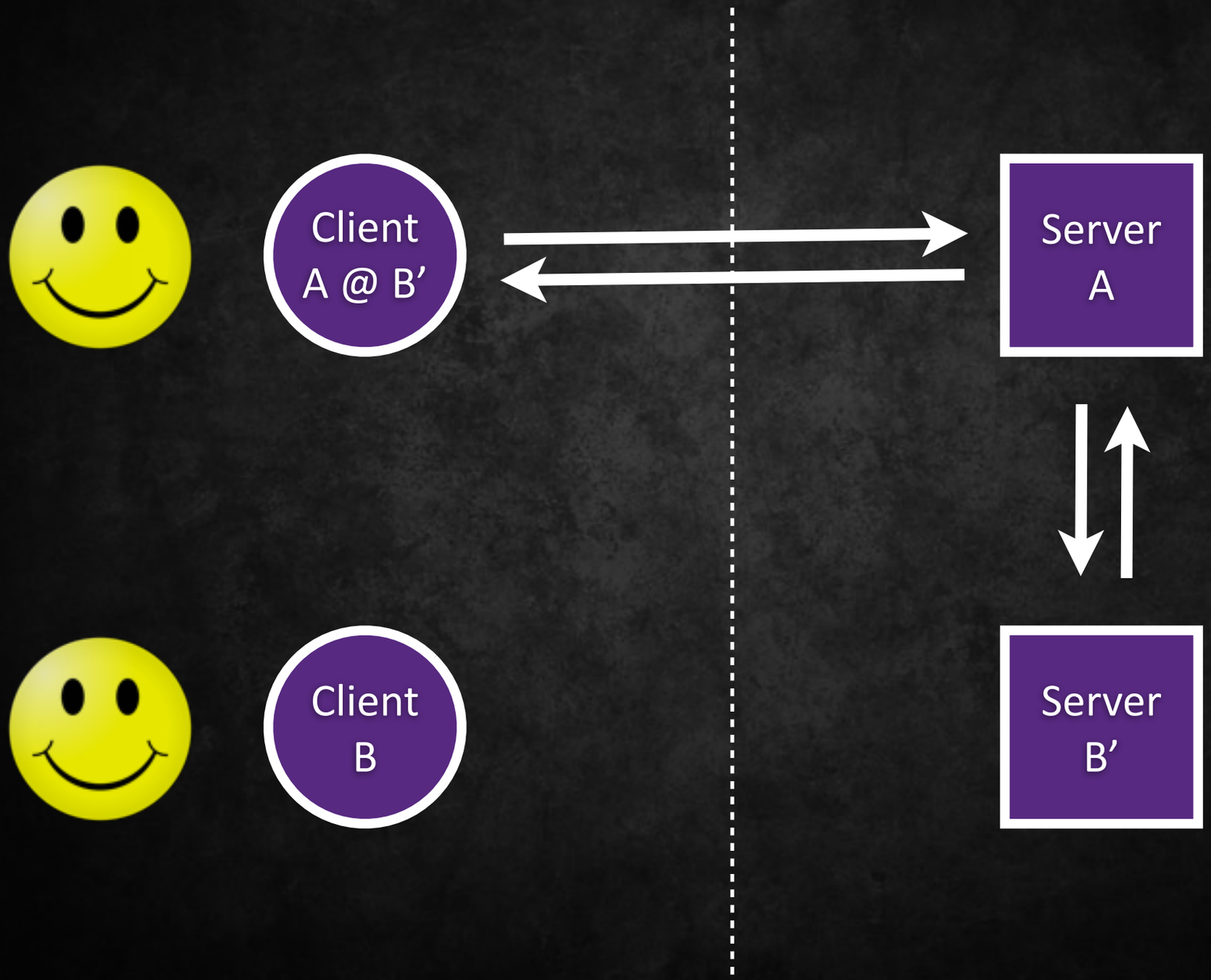

Client
A @ B
Server
A
Client
B
Server
B

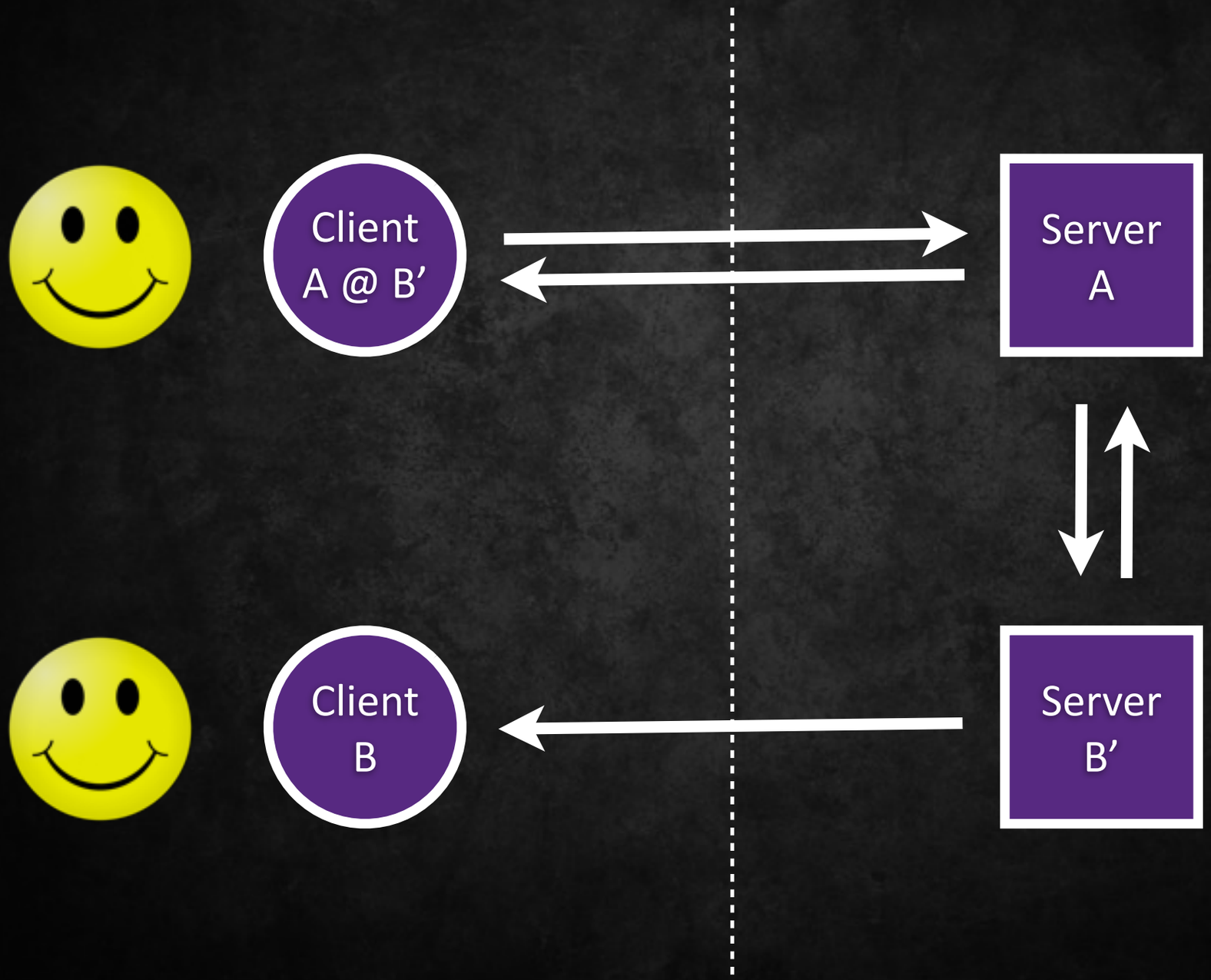

Client
A @ B
Server
A
Client
B
Server
B

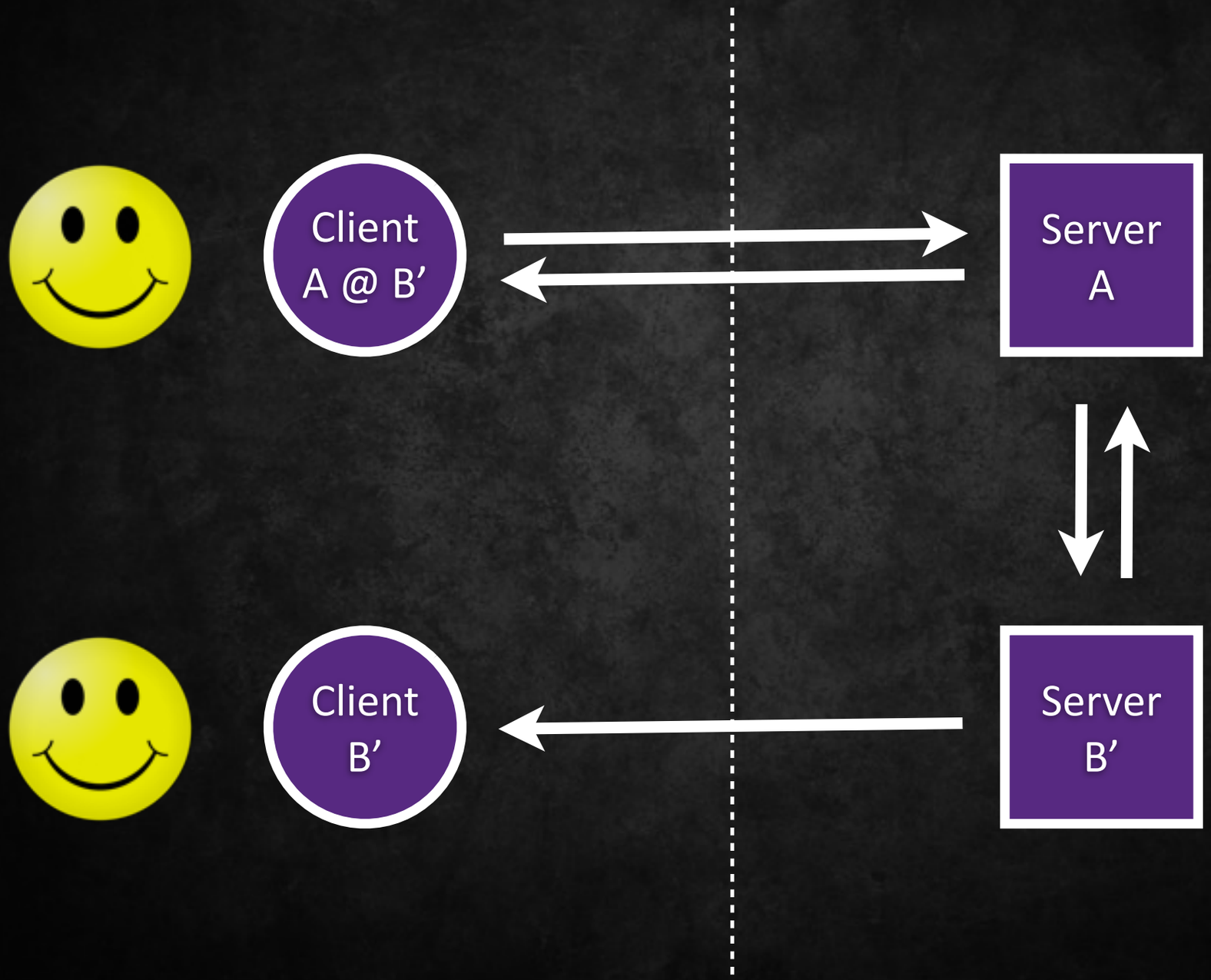

Client
A @ B
Server
A
Client
B
Server
B’


Client
A @ B
Server
A
Client
B
Server
B’


Client
A @ B
Server
A
Client
B
Server
B’


Client
A @ B’
Server
A
Client
B
Server
B’


Client
A @ B’
Server
A
Server
B’
Client
B


Client
A @ B’
Server
A
Client
B’
Server
B’


Waiting on server responses?


Waiting on server responses?
Are you out of your mind?

Schizophrenic

Client
A

Server
A

Client
B

Server
B

Client
A @ B

Server
A

Client
B

Server
B

Client
A @ B

Server
A + B

Client
B

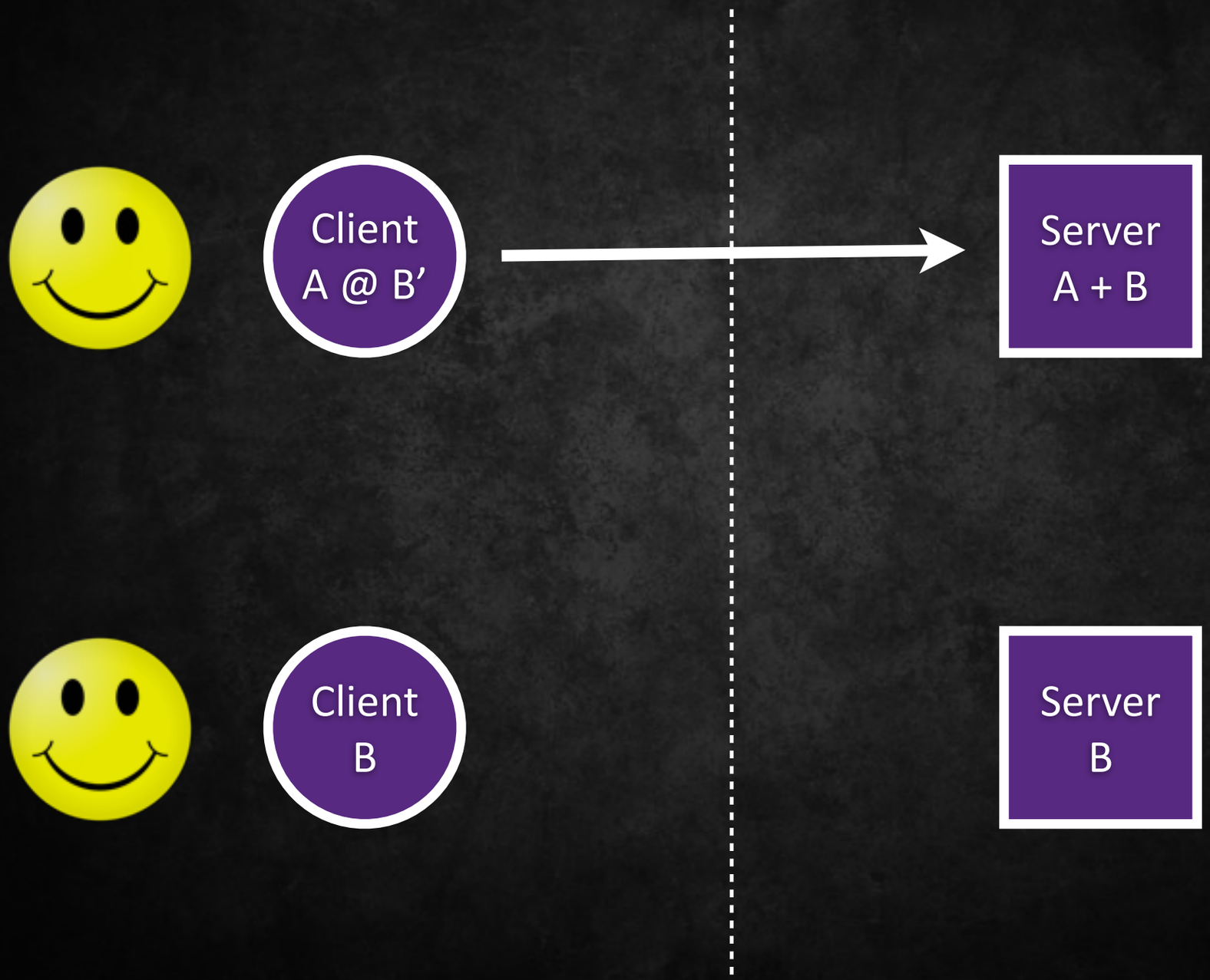
Server
B

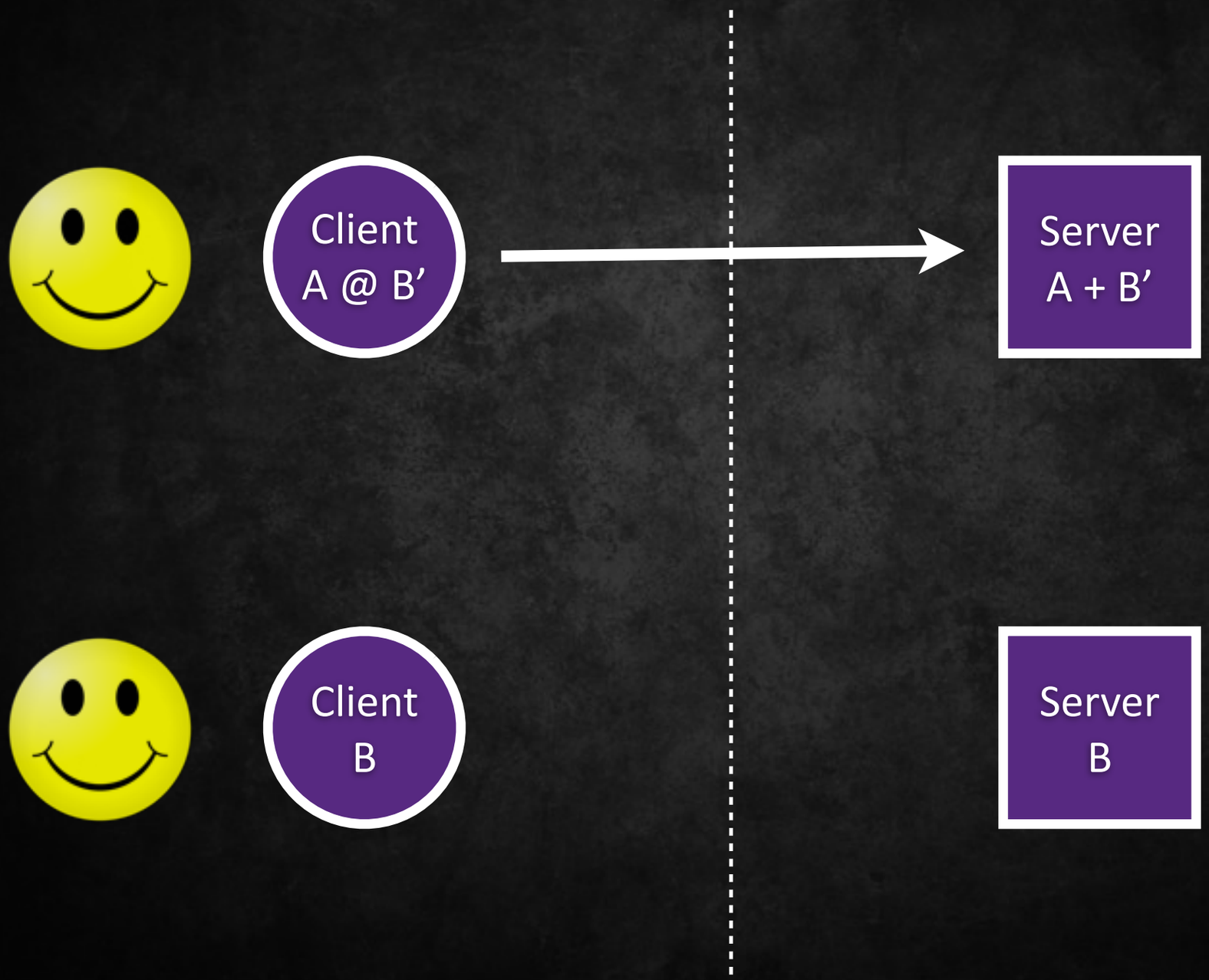
Client
A @ B’

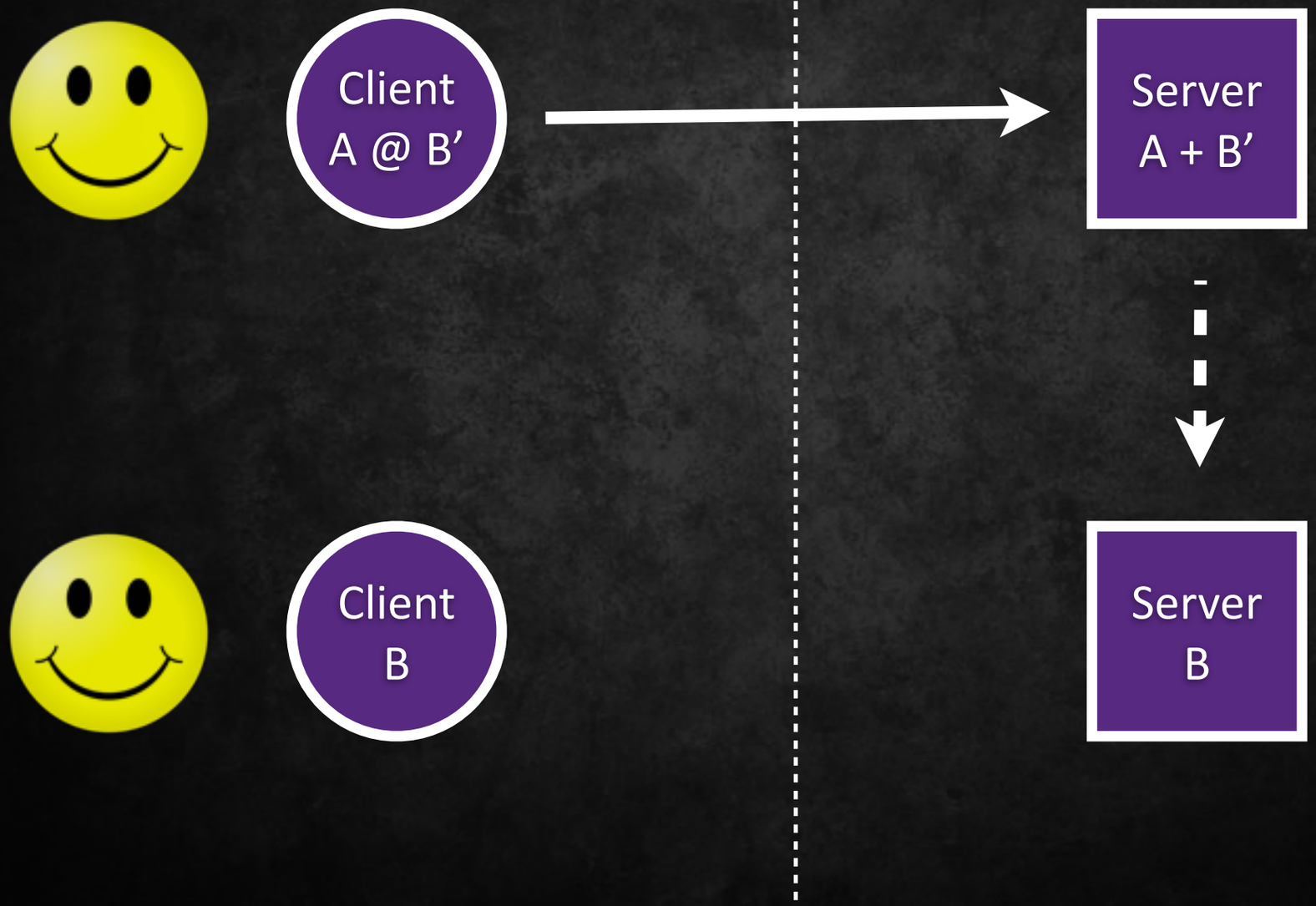
Server
A + B

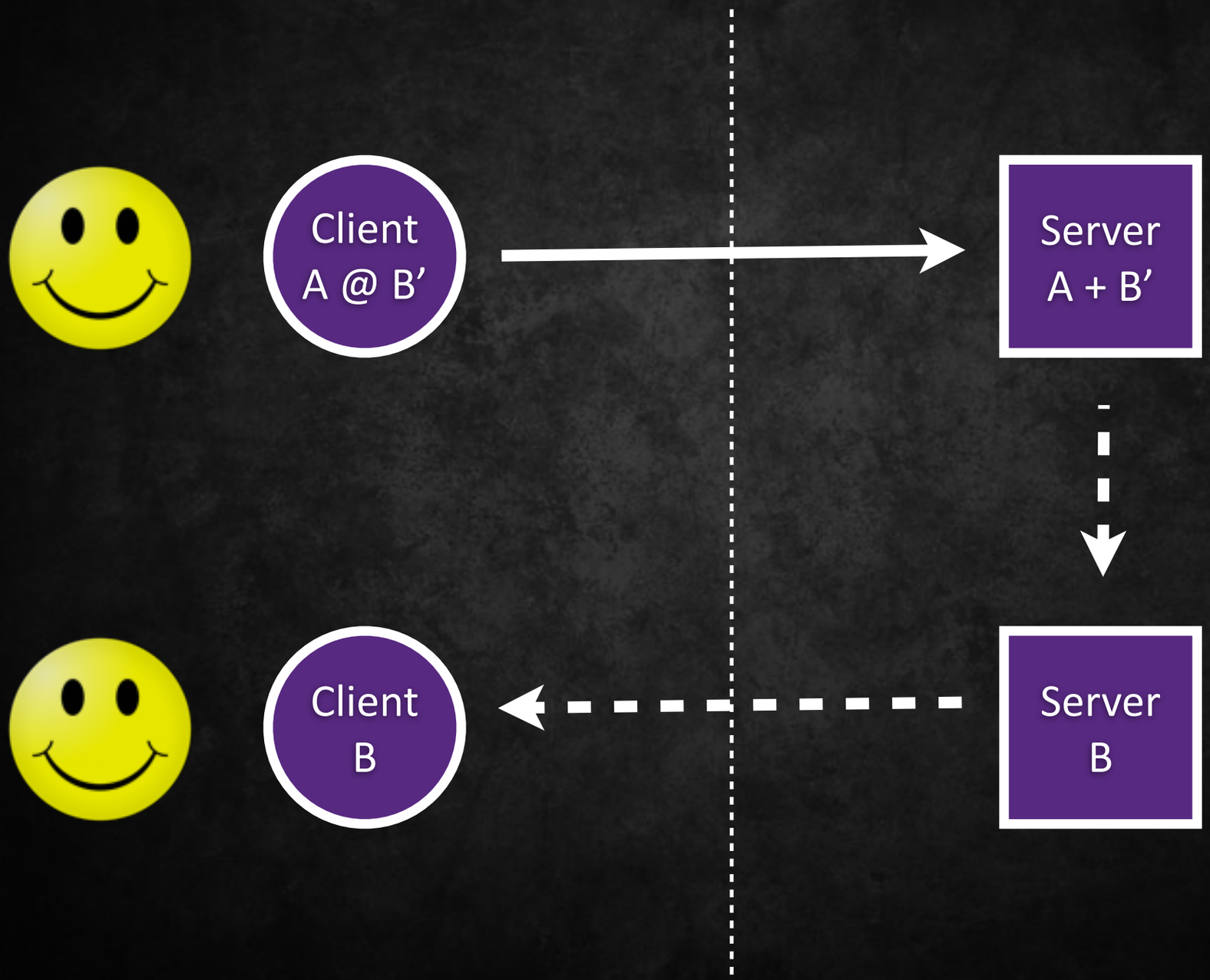
Client
B

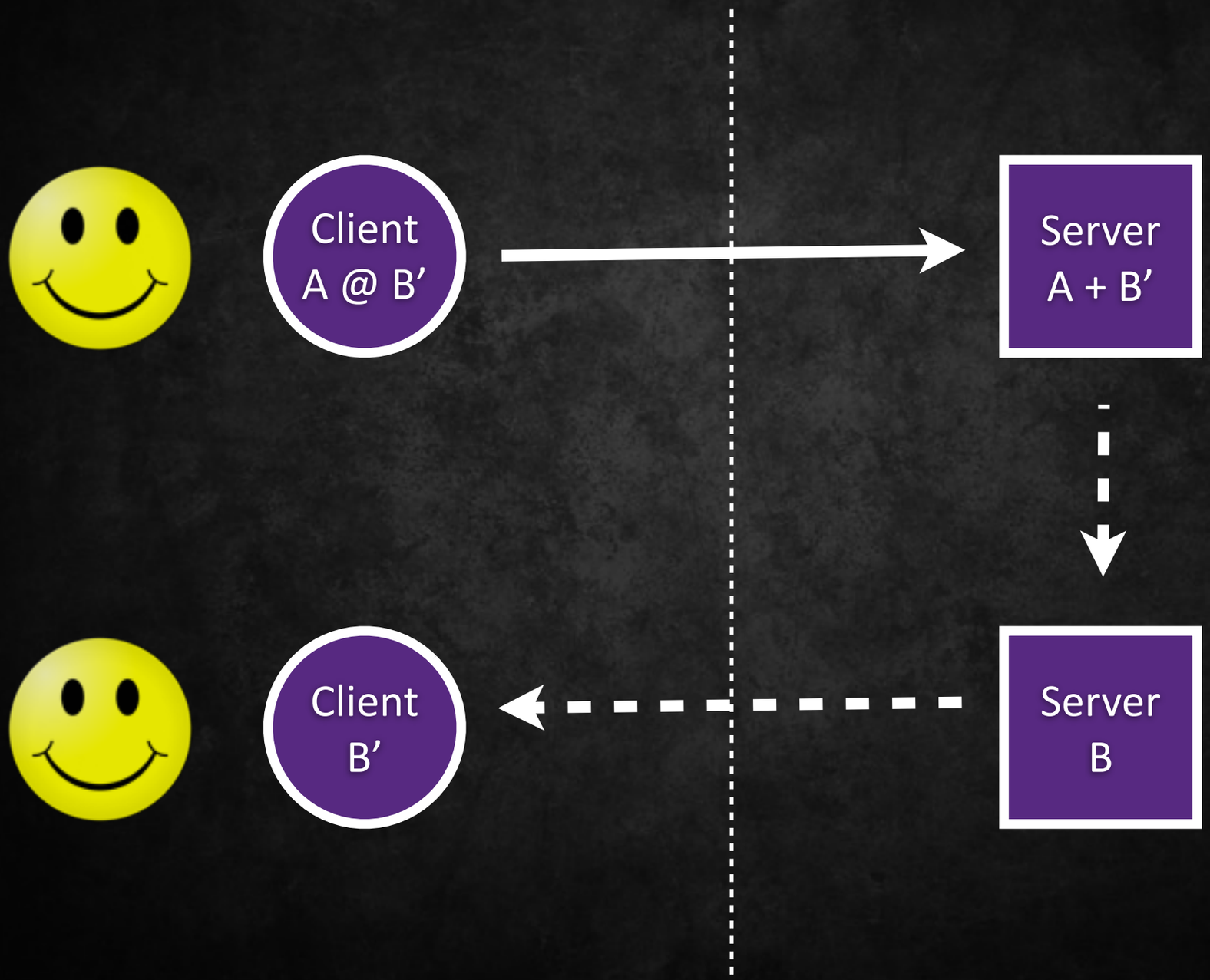
Server
B

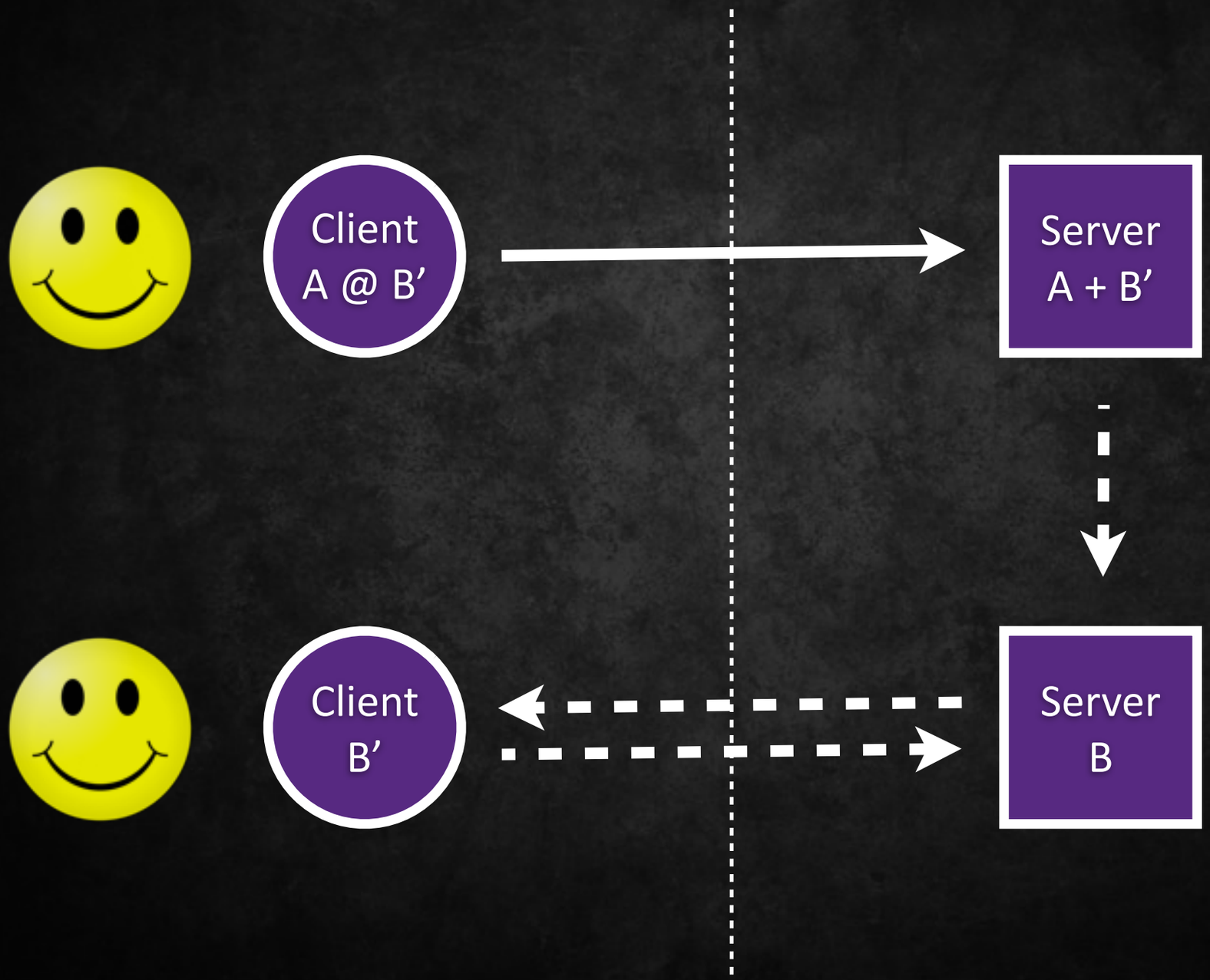

Client
A @ B’
Server
A + B
Client
B
Server
B

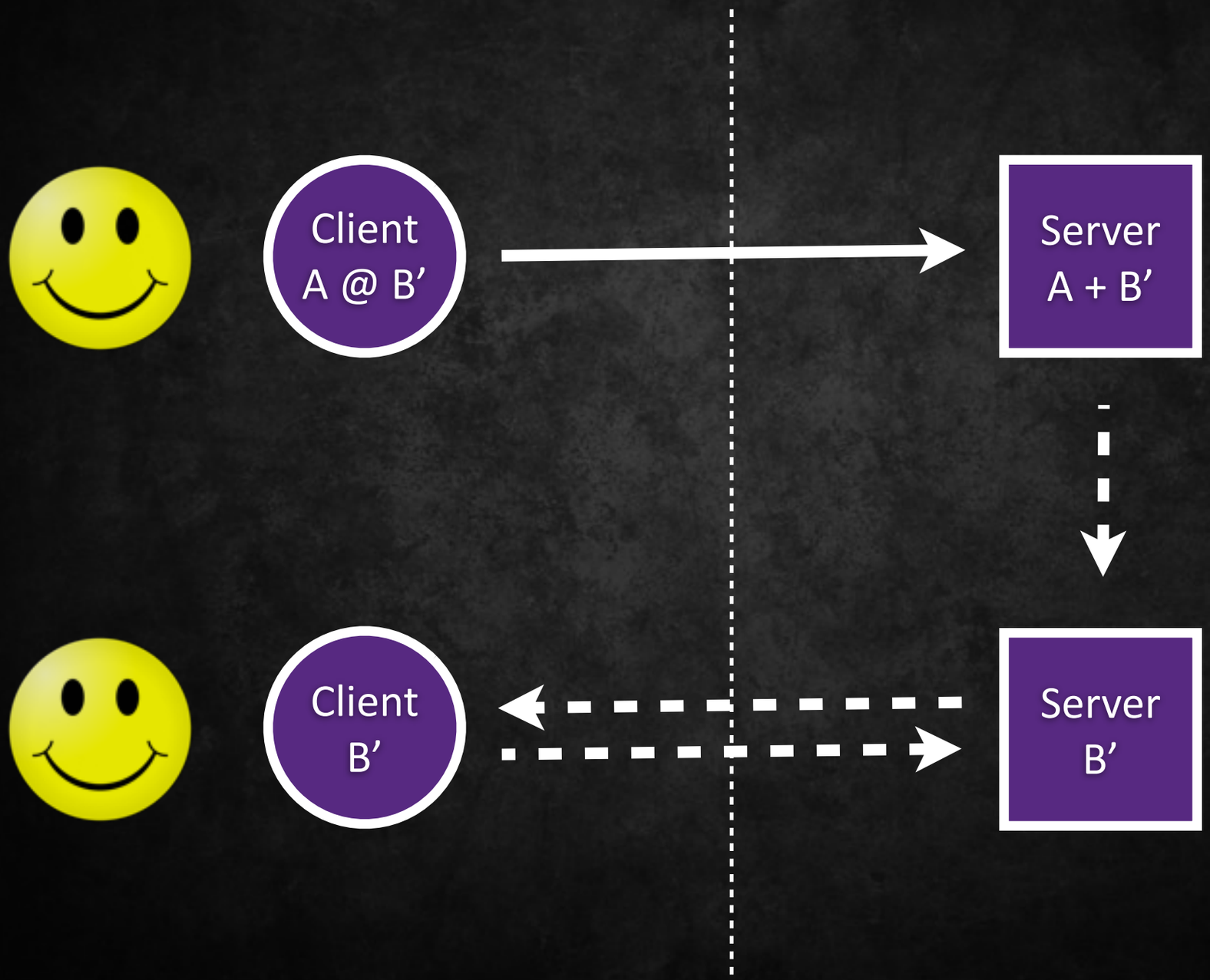

Client
A @ B’
Server
A + B’
Client
B
Server
B


Client
A @ B’
Server
A + B’
Client
B
Server
B


Client
A @ B’
Server
A + B’
Client
B
Server
B


Client
A @ B’
Server
A + B’
Client
B’
Server
B


Client
A @ B’
Server
A + B’
Client
B’
Server
B


Client
A @ B’
Server
A + B’
Client
B’
Server
B’

What about real interactions?

What about real interactions?

And what about complexity in the backend?

Optimistic

Client
A

Server
A

Client
B

Server
B

Client
A @ B

Server
A

Client
B

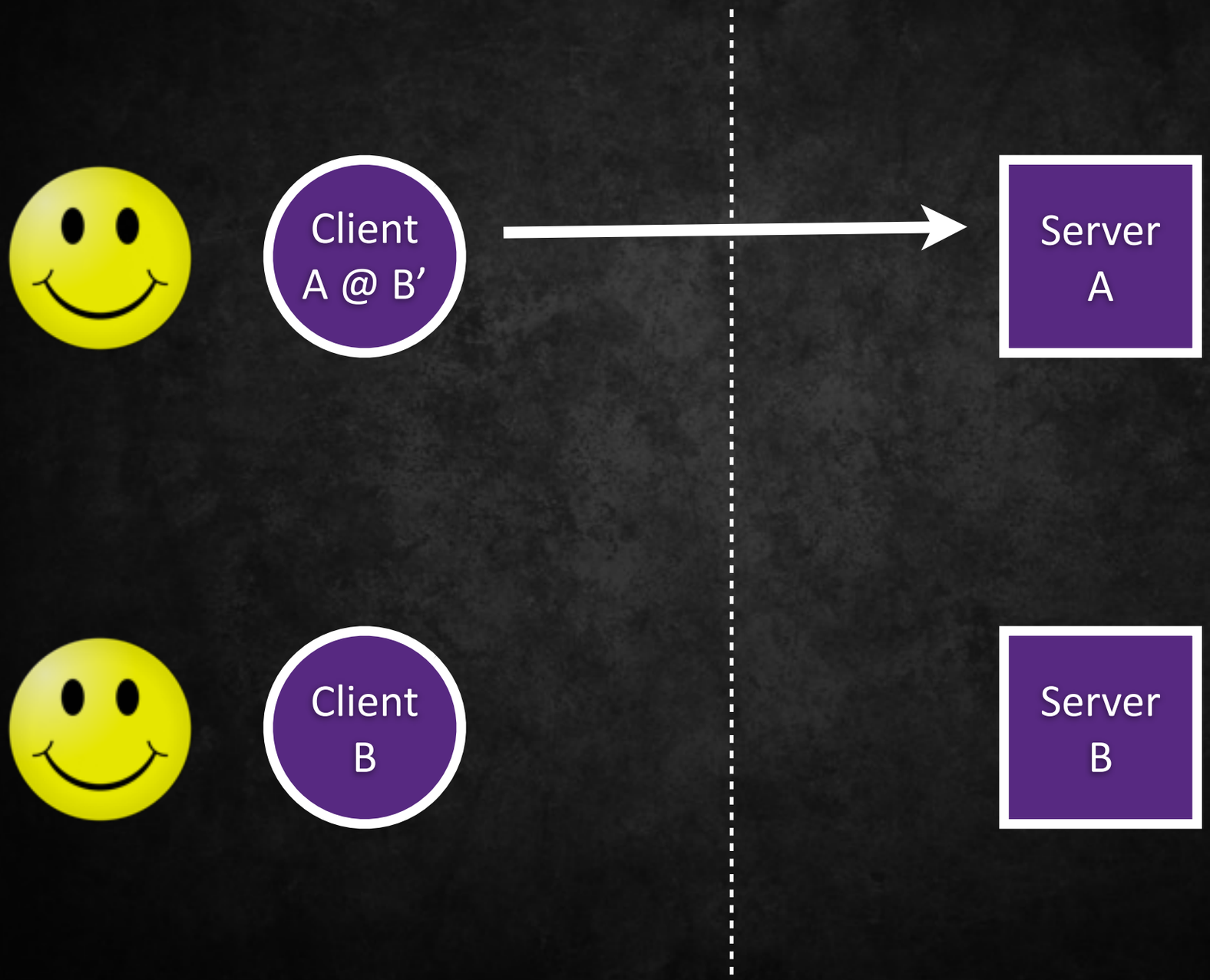
Server
B

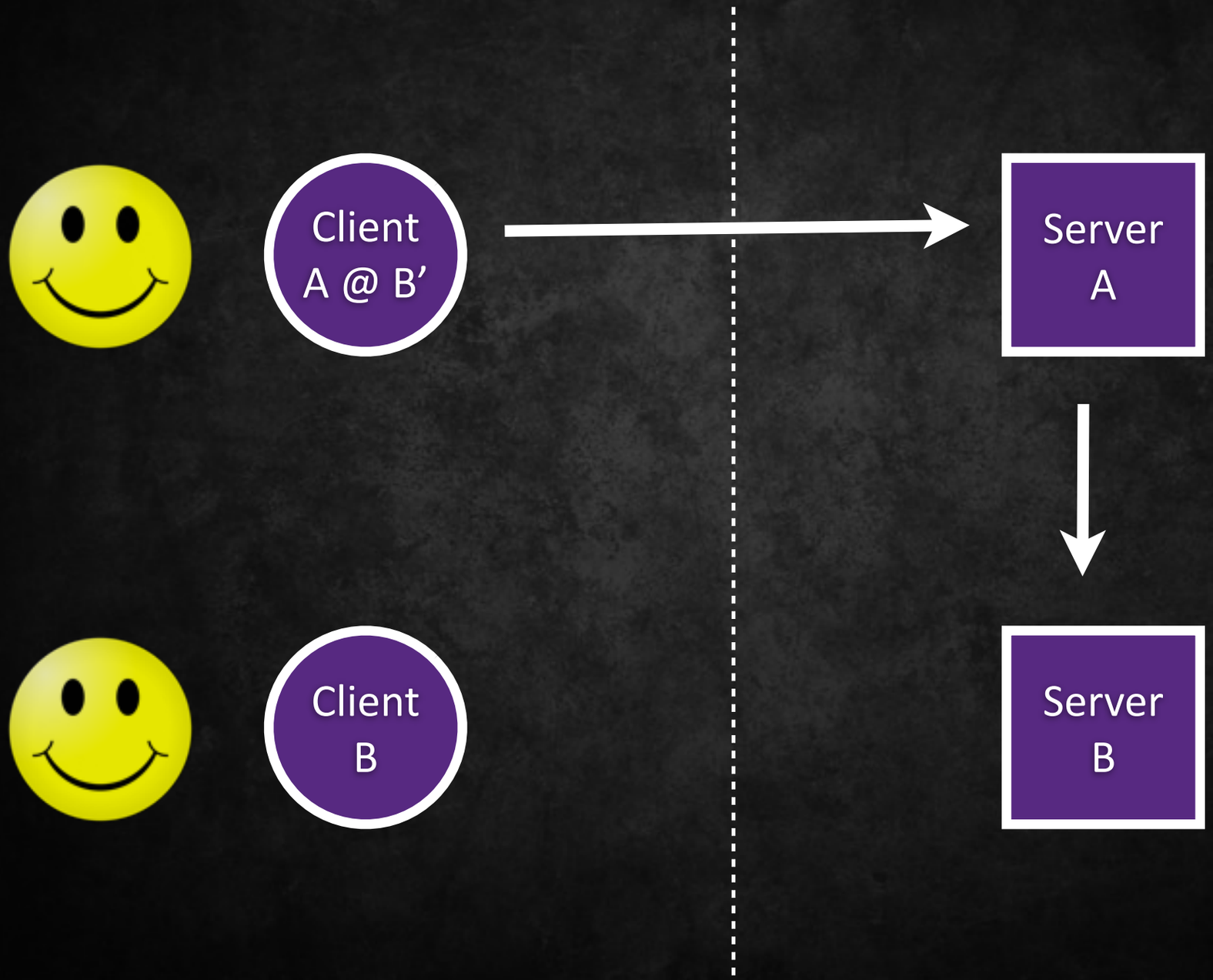
Client
A @ B’

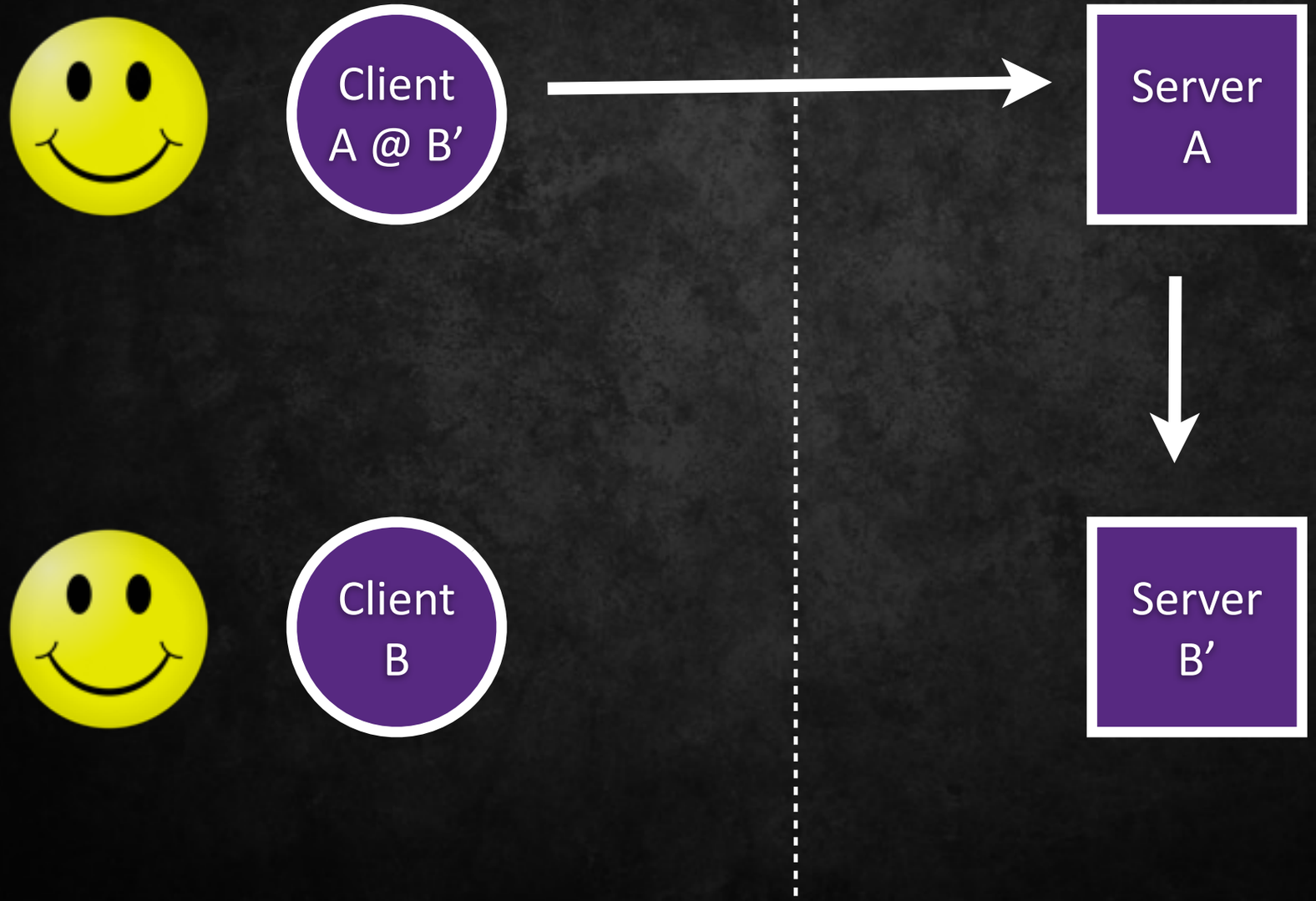
Server
A

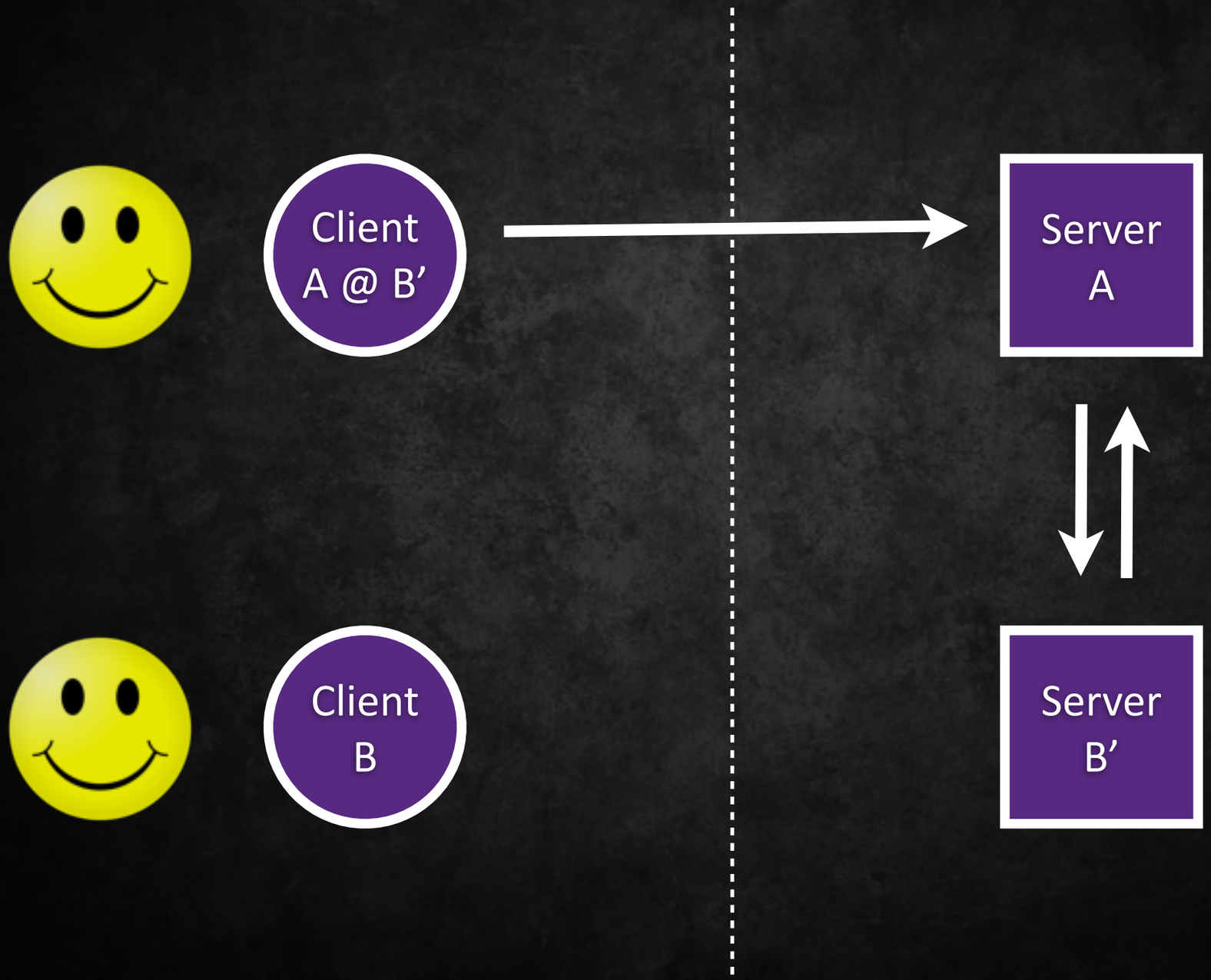
Client
B

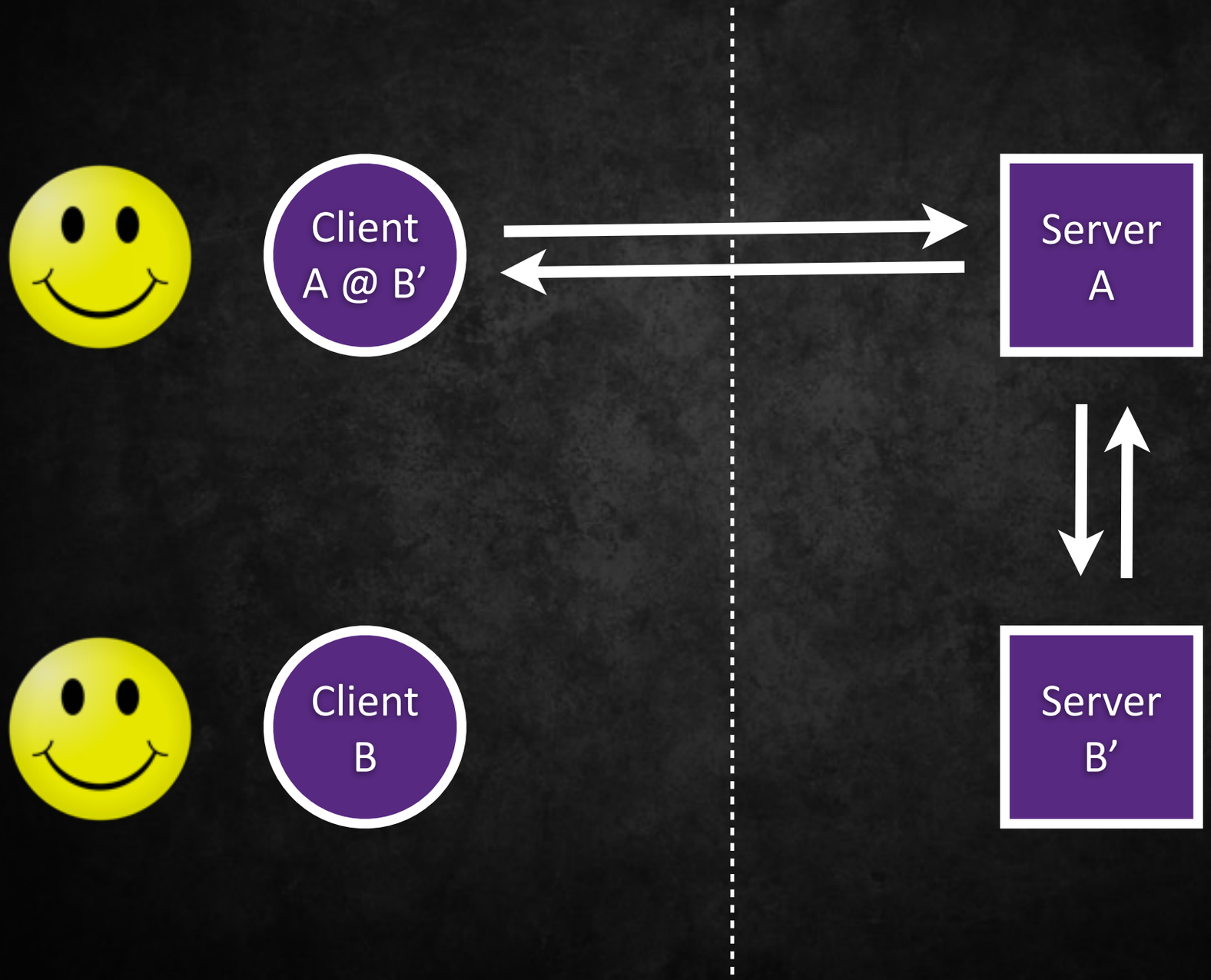
Server
B

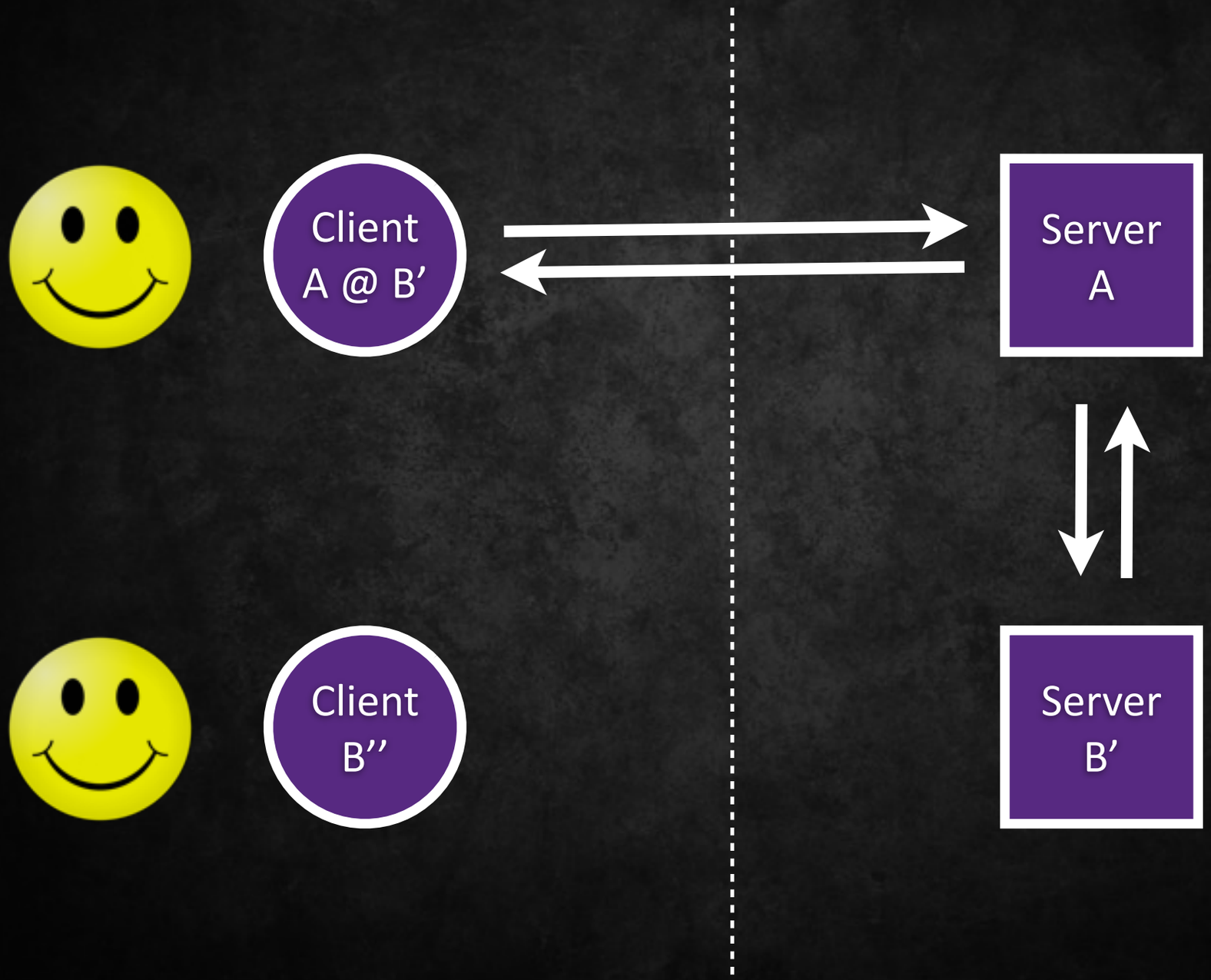

Client
A @ B’
Server
A
Client
B
Server
B

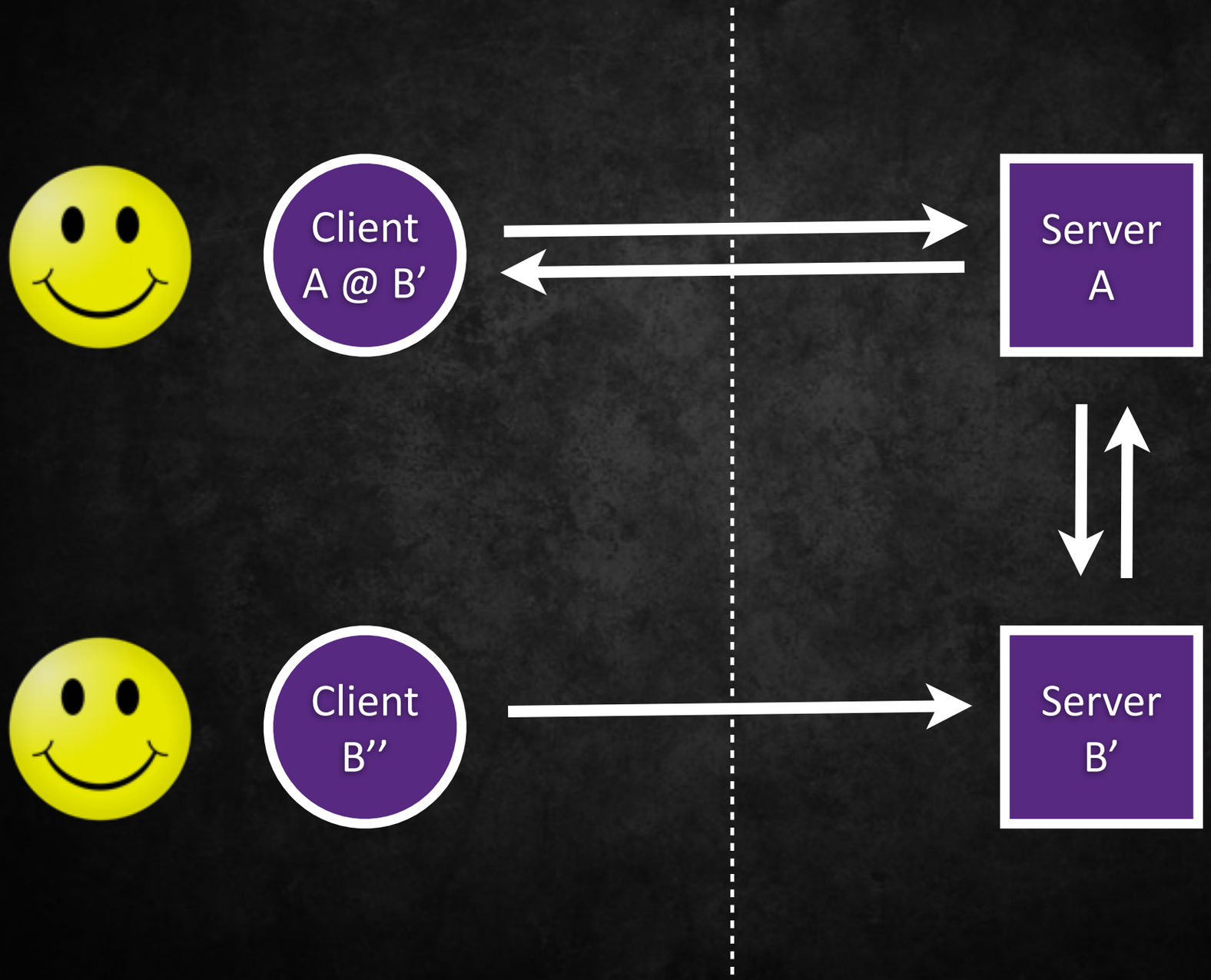

Client
A @ B’
Server
A
Client
B
Server
B

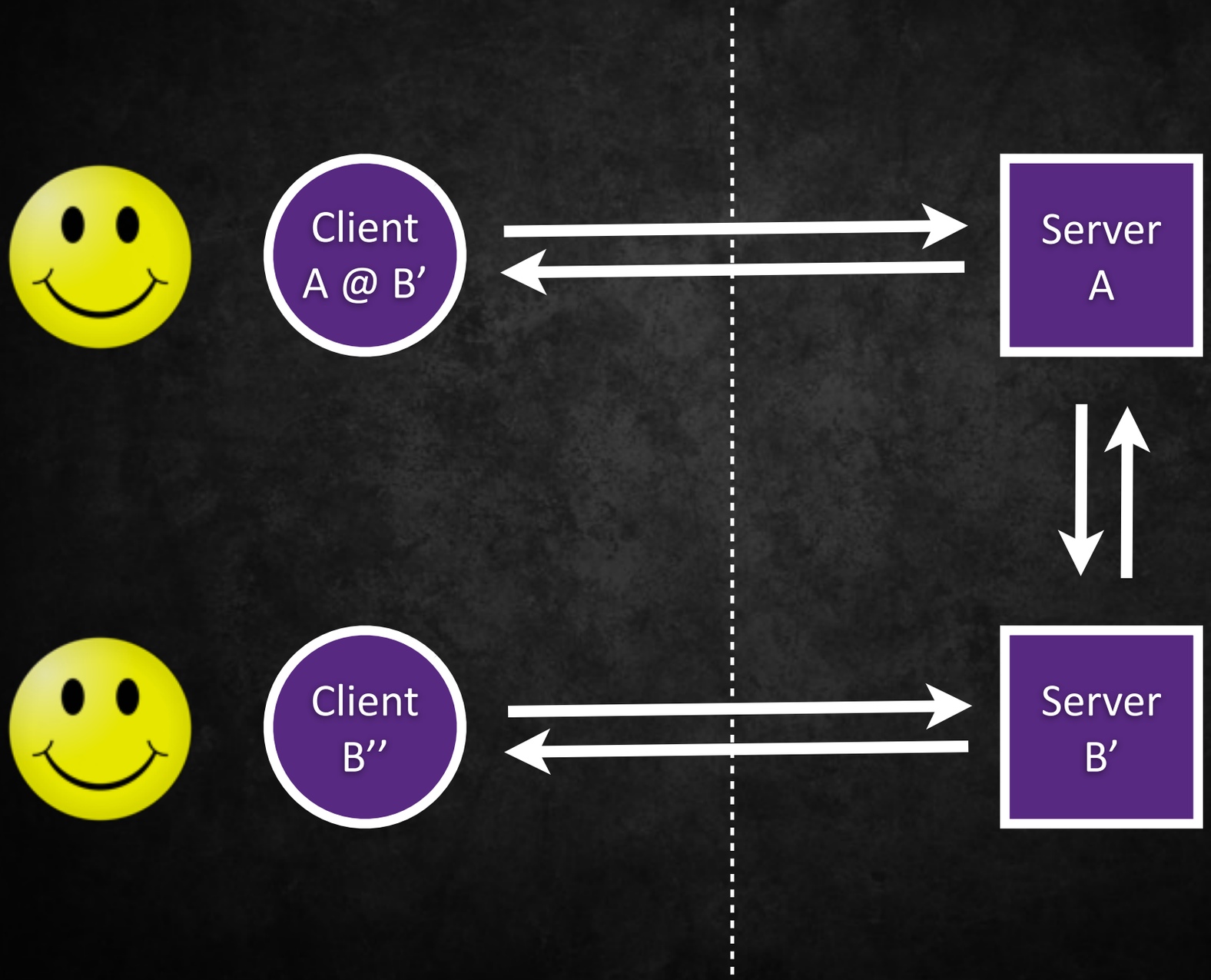

Client
A @ B’
Server
A
Client
B
Server
B’

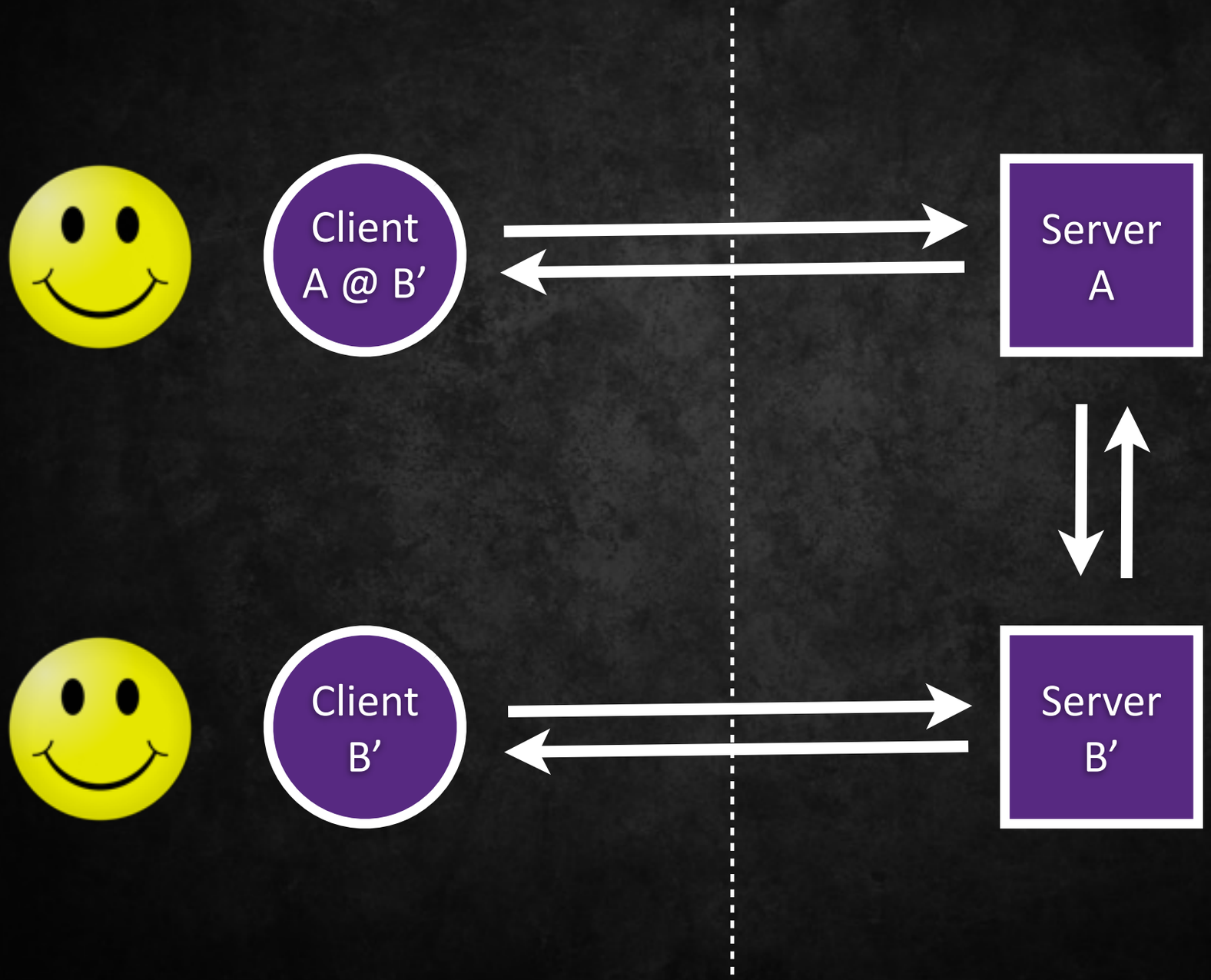

Client
A @ B’
Server
A
Client
B
Server
B’


Client
A @ B’
Server
A
Client
B
Server
B’


Client
A @ B’
Server
A
Client
B’’
Server
B’


Client
A @ B’
Server
A
Client
B’’
Server
B’


Client
A @ B’
Server
A
Client
B’’
Server
B’


Client
A @ B’
Server
A
Client
B’
Server
B’

Hmm, client rollbacks?

Hmm, client rollbacks?

Are you sure you can build that?

es

t

nt

re

tion

es

t

nt

re

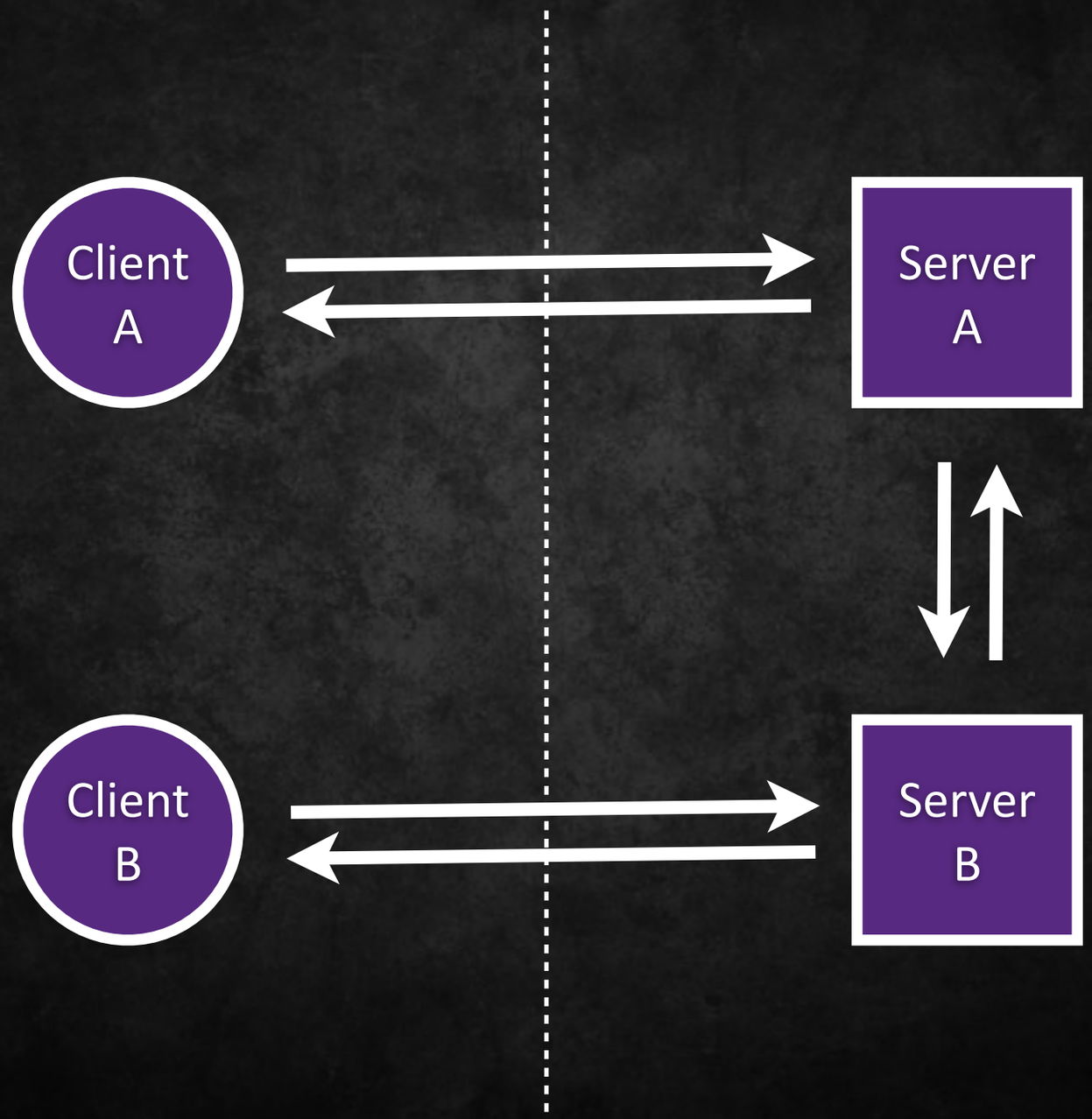
tion

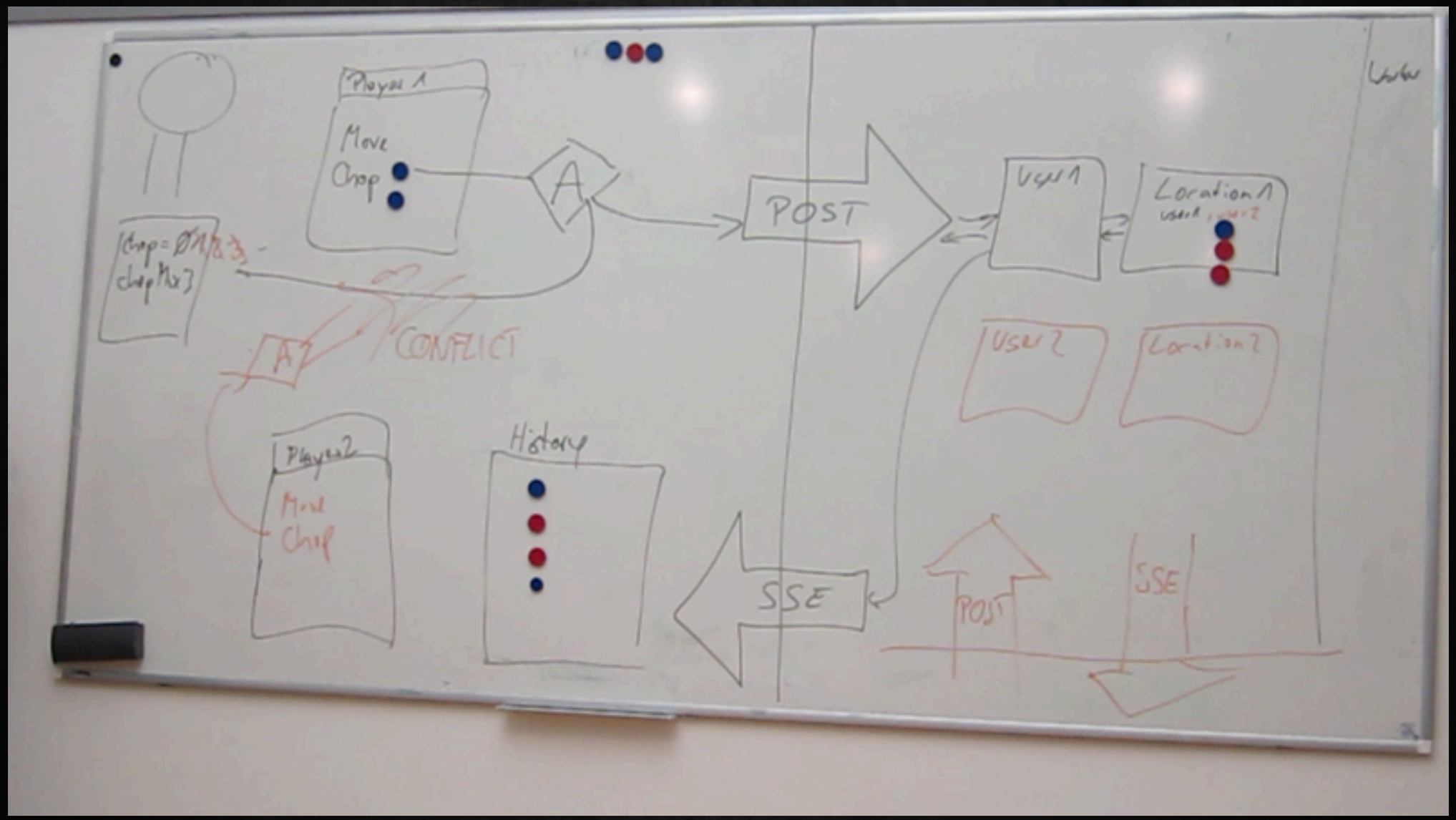
Thoughts

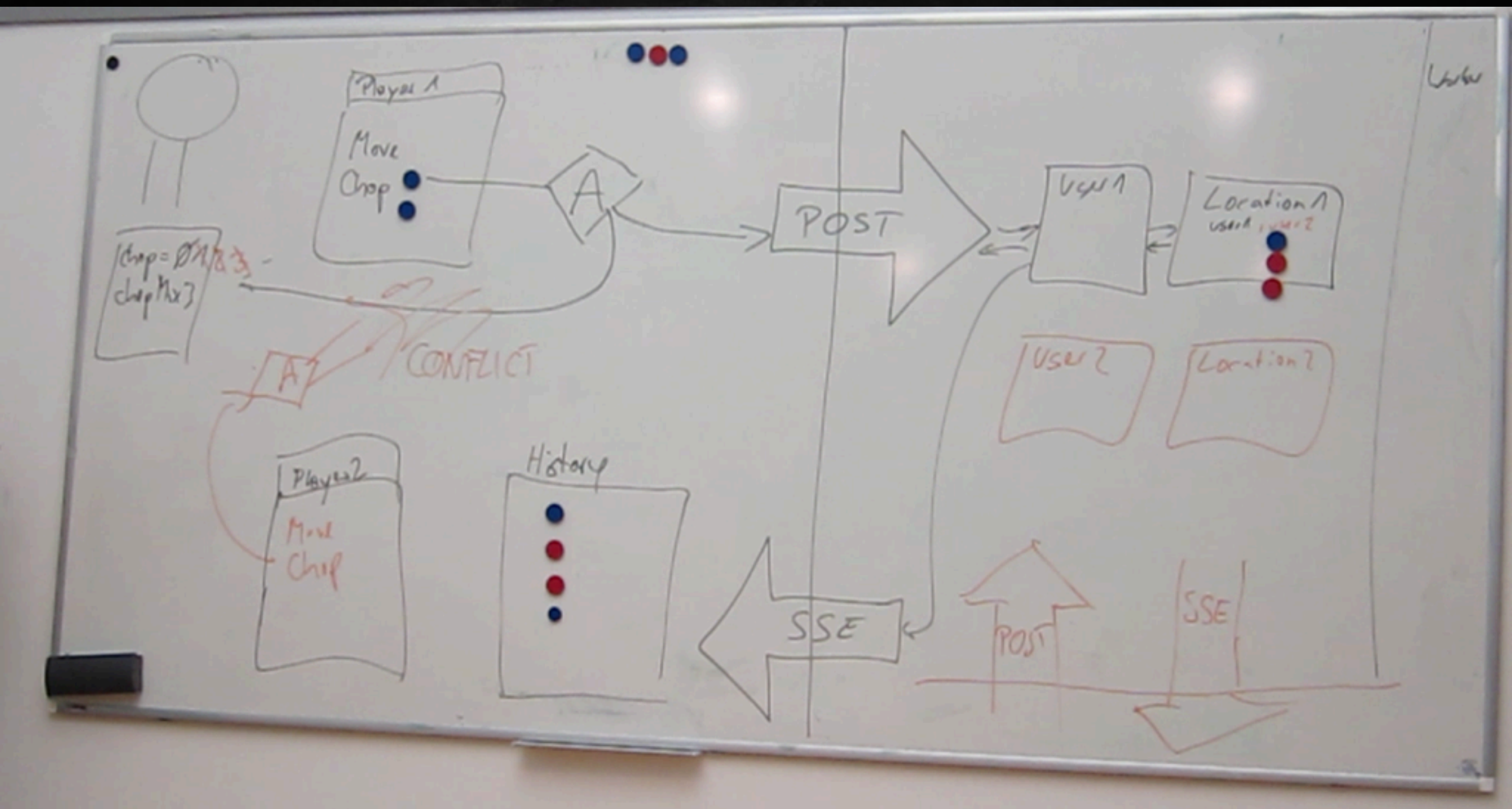
Client

Server

Operation


Client
A
Server
A
Client
B
Server
B


Player 1
Move
Chop
A
POST
CONFLICT
A?
Player2
Move
Chop
History
SSE
POST
SSE


Player 1
Move
Chop
A
POST
User 1
Location 1
CONFLICT
A
Player 2
Move
Chop
History
SSE
POST
SSE

Client A

state

count 0

max 3

Client A

user A

move

state

count 0

max 3

Client A

user A

move

pick

state

count 0

max 3

Client A

user A

move
pick

state

count 1
max 3

Client A

user A

move
pick

state

count 1
max 3

Client A

Server

state

count 1

max 3

user A

move

pick

user A

Client A

Server

state

count 1
max 3

user A

move
pick

user A

location

count 0

Client A

Server

state

count 1

max 3

user A

move

pick

user A

location

count 1

Client A

Server

state

count 1

max 3

user A

move

pick

user A

location

count 1

Client A

Server

state

count 1

max 3

user A

move

pick

user A

location

count 1

history

move A

pick A

Client A

Server

state

count 1

max 3

user A

move

pick

user A

location

count 1

history

move A

pick A

Client A

Server

user A

move
pick

state

count 1
max 3

user A

location

count 1

history

move A
pick A

Client A

Server

user A

move
pick

state

count 1
max 3

user A

location

count 1

user B

history

move A
pick A

Client A

Server

user A

move
pick

state

count 1
max 3

user A

location

count 1

user B

location

history

move A
pick A

Client A

Server

state

count 1

max 3

user A

move

pick

history

move A

pick A

user A

location

count 2

user B

location

Client A

Server

user A

move
pick

state

count 1
max 3

history

move A
pick A

user A

location

count 2

user B

location

Client A

Server

user A

move
pick

state

count 1
max 3

user A

location

count 2

user B

location

history

move A
pick A
pick B

Client A

Server

state

count 1

max 3

user A

move

pick

user B

pick

history

move A

pick A

pick B

Client A

user A

location

count 2

user B

location

Server

state

count 2
max 3

user A

move
pick

user B

pick

history

move A
pick A
pick B

Client A

user A

location

count 2

user B

location

Server

state

count 2
max 3

user A

move
pick

user B

pick

history

move A
pick A
pick B

Client A

user A

location

count 2

user B

location

Server

state

count 2

max 3

user A

move

pick

pick

user B

pick

history

move A

pick A

pick B

Client A

user A

location

count 2

user B

location

Server

user A

move
pick
pick

state

count 3
max 3

user B

pick

history

move A
pick A
pick B

Client A

user A

location

count 2

user B

location

Server

user A

move
pick
pick

state

count 3
max 3

user A

location

count 2

user B

pick

history

move A
pick A
pick B

user B

location

Client A

Server

state

count 3

max 3

user A

move

pick

pick

user B

pick

history

move A

pick A

pick B

user A

location

count 2

user B

location

Client A

Server

state

count 3

max 3

user A

move

pick

pick

user B

pick

history

move A

pick A

pick B

user A

location

count 3

user B

location

Client A

Server

state

count 3

max 3

user A

move

pick

pick

user B

pick

history

move A

pick A

pick B

Client A

user A

location

count 3

user B

location

Server

state

count 3

max 3

user A

move

pick

pick

user B

pick

history

move A

pick A

pick B

user A

location

count 3

user B

location

Client A

Server

state

count 3

max 3

user A

move

pick

pick

user B

pick

history

move A

pick A

pick B

pick B

user A

location

count 3

user B

location

Client A

Server

state

count 3

max 3

user A

move

pick

pick

user A

location

count 3

user B

pick

pick

history

move A

pick A

pick B

pick B

user B

location

Client A

Server

user A

move
pick
pick

state

count 3
max 3

user B

pick
pick

history

move A
pick A
pick B
pick B

Client A

user A

location

count 3

user B

location

Server

state

count 3

max 3

user A

move

pick

pick

user B

pick

pick

history

move A

pick A

pick B

pick B

user A

location

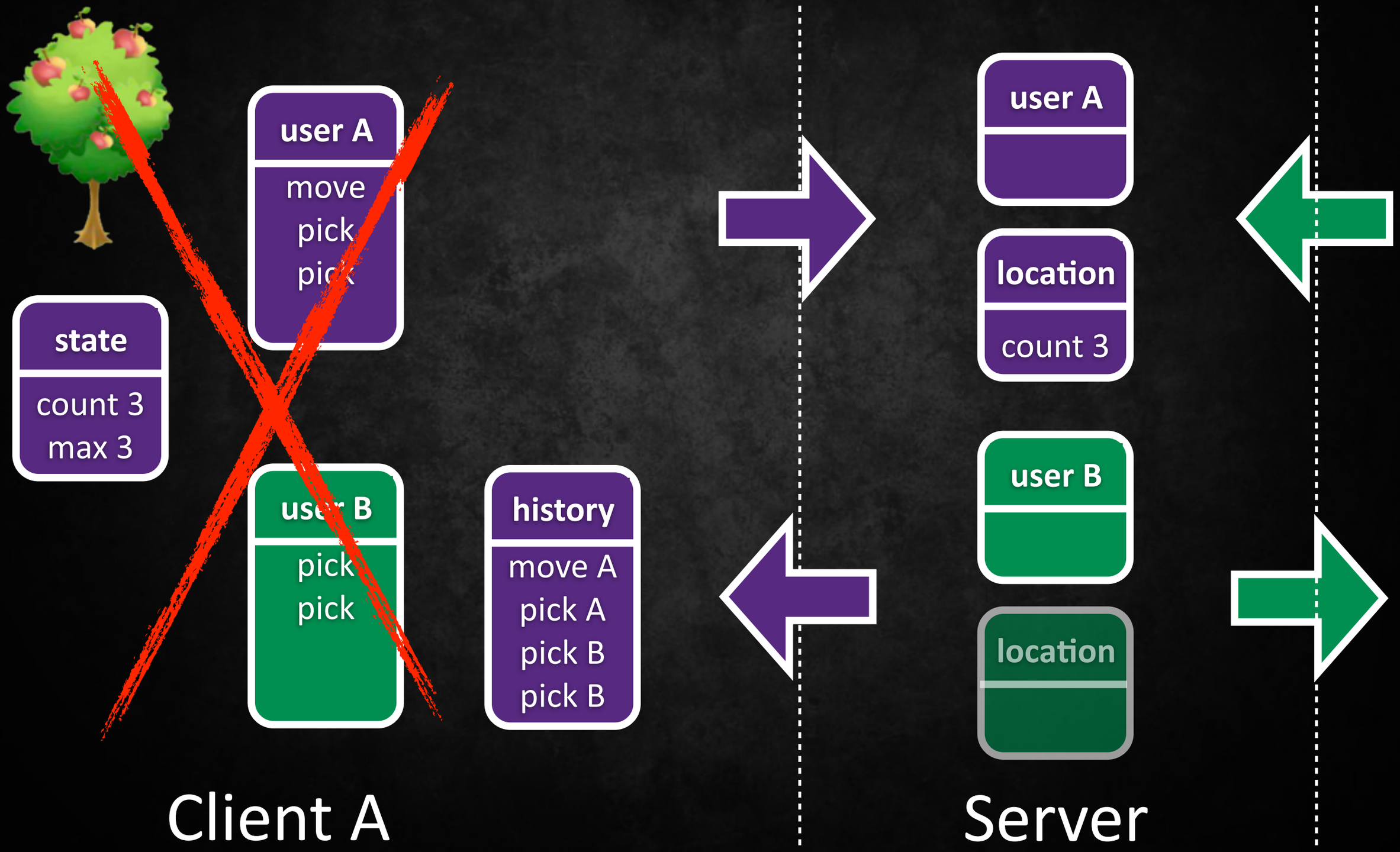
count 3

user B

location

Client A

Server


user A
move
pick
pick
state
count 3
max 3
user B
pick
pick
history
move A
pick A
pick B
pick B
user A
location
count 3
user B
location
Client A
Server

state
count 0
max 3

user A

user B

history
move A
pick A
pick B
pick B

Client A

user A

location
count 3

user B

location

Server

state

count 0

max 3

user A

move

user A

location

count 3

user B

history

move A

pick A

pick B

pick B

user B

location

Client A

Server

state

count 1

max 3

user A

move

pick

user B

history

move A

pick A

pick B

pick B

user A

location

count 3

user B

location

Client A

Server

state

count 2

max 3

user A

move

pick

user B

pick

history

move A

pick A

pick B

pick B

user A

location

count 3

user B

location

Client A

Server

state

count 3

max 3

user A

move

pick

user B

pick

pick

history

move A

pick A

pick B

pick B

Client A

user A

location

count 3

user B

location

Server

...

(Let's look inside)

es

t

nt

re

tion

es

t

nt

re

tion

Thoughts

Client

Server

Operation

Erlang

Erlang

Process isolation

Erlang

Process isolation

Networking

Erlang

Process isolation

Networking

Distribution

User process

User process

Stores user state

User process

Stores user state

Real-time push to client

World process

World process

Serializes updates

World process

Serializes updates

Concurrency “control”

World process

Serializes updates

Concurrency “control”

Runs game logic

Game logic

Game logic

Pure

Game logic

Pure

```
f(State, Cmd) -> {NewState, Reply}
```

Game logic

Pure

```
f(State, Cmd) -> {NewState, Reply}
```

Transactions

Real-time push

Real-time push

Browser support

Real-time push

Browser support

“Transfer-Encoding: chunked”

Real-time push

Browser support

“Transfer-Encoding: chunked”

State updates as JSON

es

t

nt

re

tion

es

t

nt

re

tion

Thoughts

Client

Server

Operation

Elli

Elli

Efficient, robust

Elli

Efficient, robust

High-throughput, low latency

Elli

Efficient, robust

High-throughput, low latency

Request-response model

Elli

Efficient, robust

High-throughput, low latency

Request-response model

github.com/knutin/elli

Distribution

Distribution

Idea: Identical nodes, Riak

Distribution

Idea: Identical nodes, Riak

No centralization/serialization

Distribution

Idea: Identical nodes, Riak

No centralization/serialization

No SPOF

Distribution needs

Distribution needs

Locking

Distribution needs

Locking

Dynamic cluster

Meet “locker”

Meet “locker”

Distributed lock table

Meet “locker”

Distributed lock table

CAS

Meet “locker”

Distributed lock table

CAS

Write quorum, 2PC

Meet “locker”

Distributed lock table

CAS

Write quorum, 2PC

Asynchronous replication

Meet “locker”

Distributed lock table

CAS

Write quorum, 2PC

Asynchronous replication

330 SLOC, one gen_server

The Games

The Past

The Present

The Future

The Foundation

Research

Research

Aka “hacking”

Everything is possible

Everything is possible

First person

Everything is possible

First person

Real-time strategy

Everything is possible

First person

Real-time strategy

Turn-based

Everything is possible

First person

Real-time strategy

Turn-based

Etc

Nothing is possible

Nothing is possible

Network latency kills gameplay

Lag compensation

Lag compensation

Many solutions

Lag compensation

Many solutions

Depends on game

FPS

FPS

Speed is king

FPS

Speed is king

Divergent simulations

FPS

Speed is king

Divergent simulations

Server keeps history

FPS

Speed is king

Divergent simulations

Server keeps history

Validates delayed commands

RTS

RTS

Synchronization is king

RTS

Synchronization is king

Identical simulations

RTS

Synchronization is king

Identical simulations

Server buffers commands

RTS

Synchronization is king

Identical simulations

Server buffers commands

Broadcasts, lockstep

The Games

The Past

The Present

The Future

The Foundation

Small independent teams

Each team adds something

Based on previous learnings

Standing on the shoulders of giants

Slides: http://woo.ga/backend

Slides: http://woo.ga/backend

Code: http://github.com/wooga

Questions?

Questions?

@jrirei @knutin

http://wooga.com/jobs