

goto;
chicago'

The soul of Erlang and Elixir

Saša Jurić
@sasajuric

BEAM languages

Erlang

Elixir

LFE

Alpaca

Gleam

...

process

• • •

do_something(...)

• • •

...
spawn (fn -> ... end)
...

process A

• • •

foo(. . .)

bar(. . .)

• • •

process B

• • •

baz(. . .)

qux(. . .)

• • •

```
graph LR; A[process a] --> B[process b]
```

process a

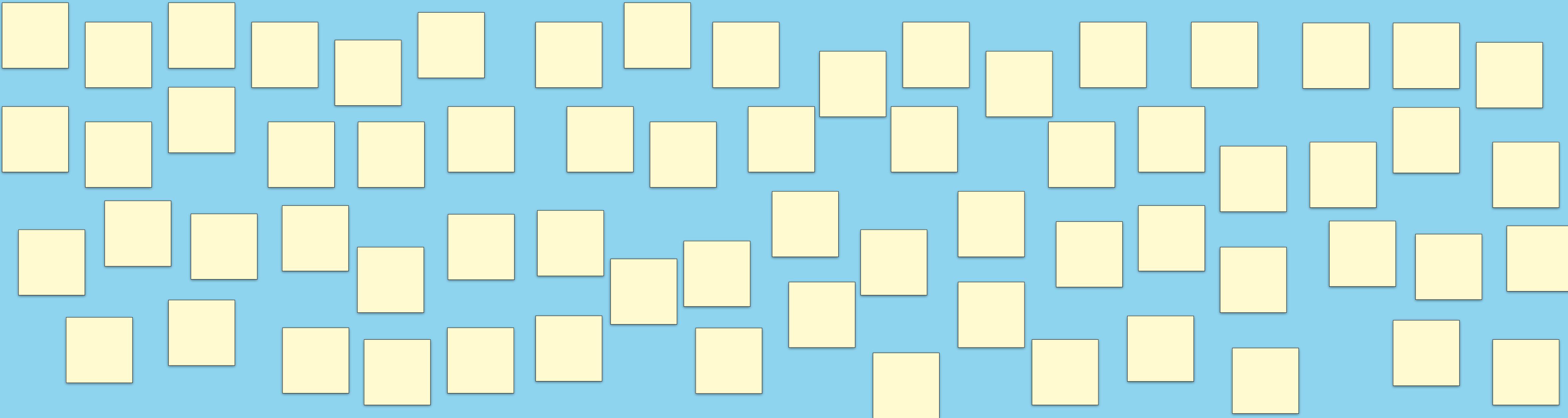
send(
 process_b,
 some_message
)

process b

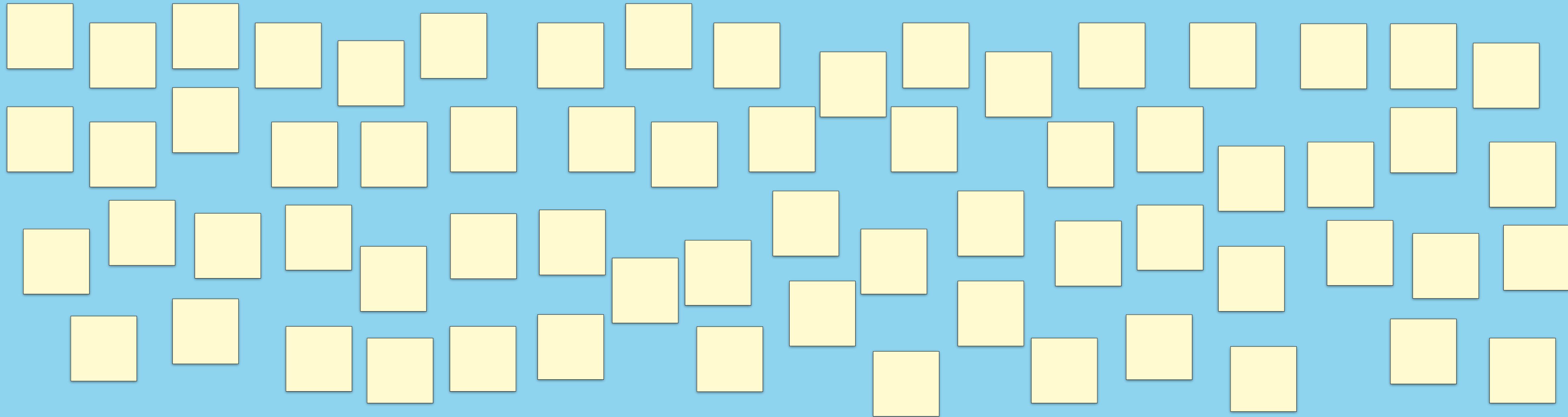
receive do
message ->
handle(message)
end

BEAM (Erlang VM)

BEAM (Erlang VM)



BEAM (Erlang VM)



scheduler

scheduler

scheduler

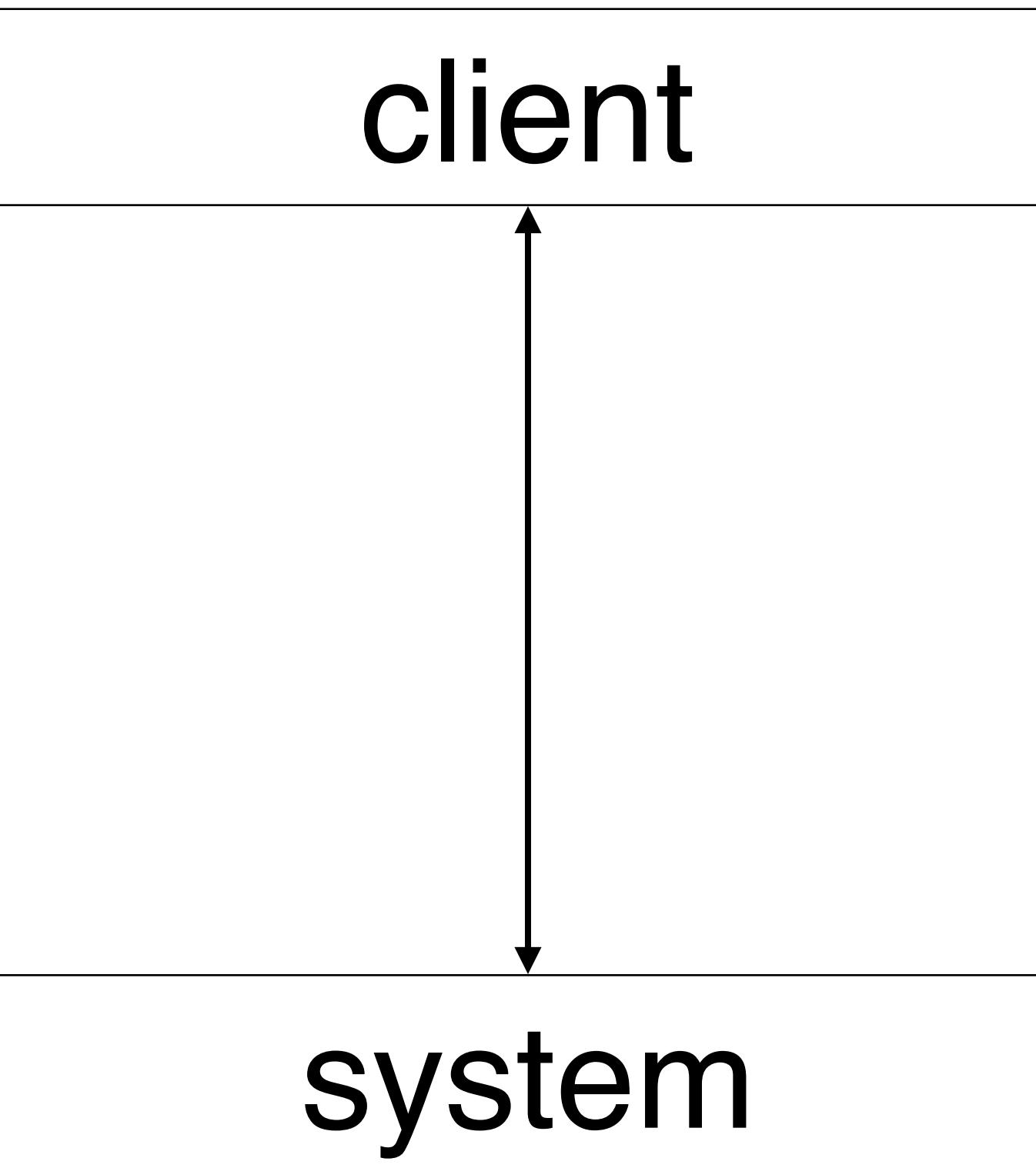
scheduler

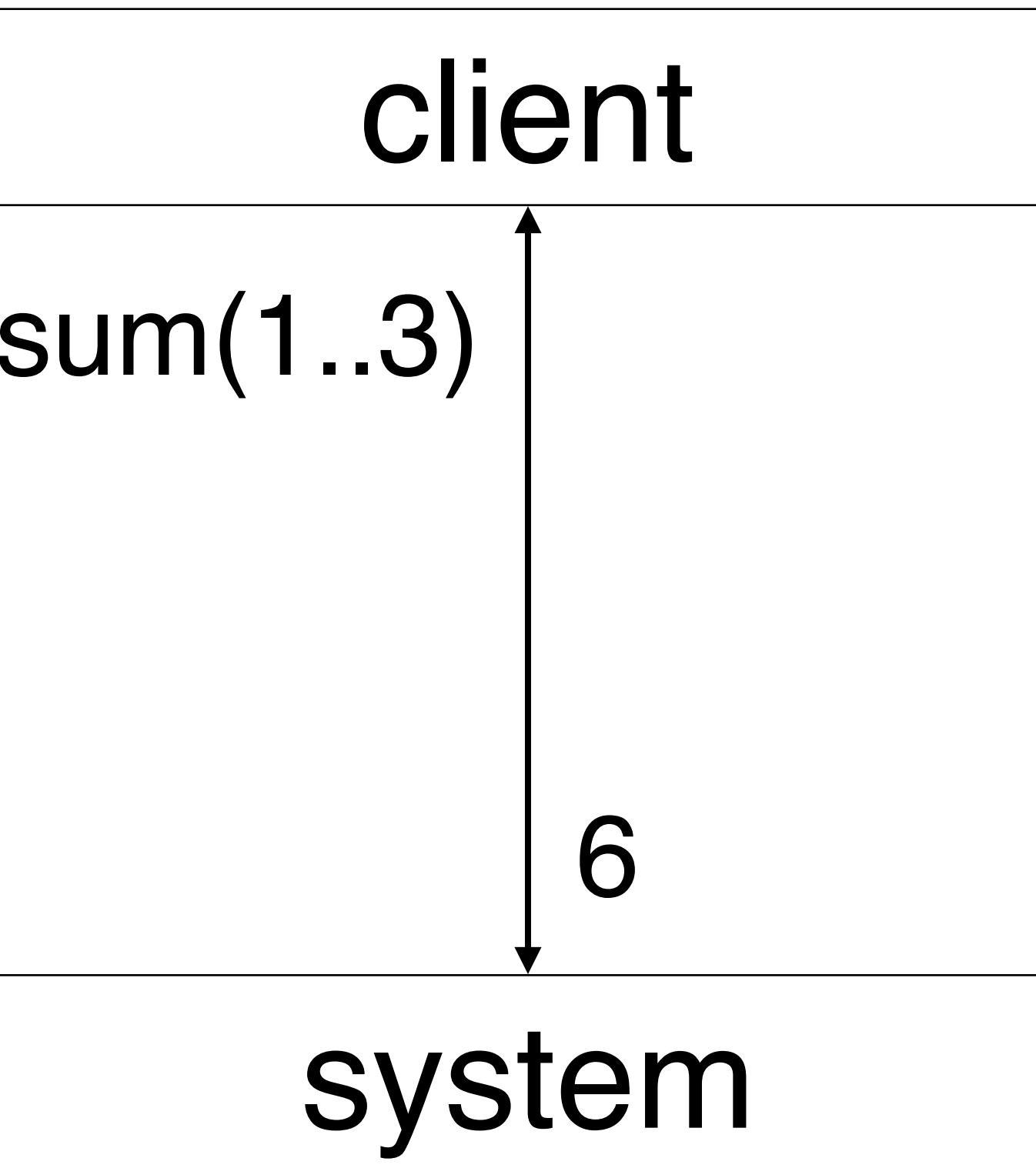
CPU

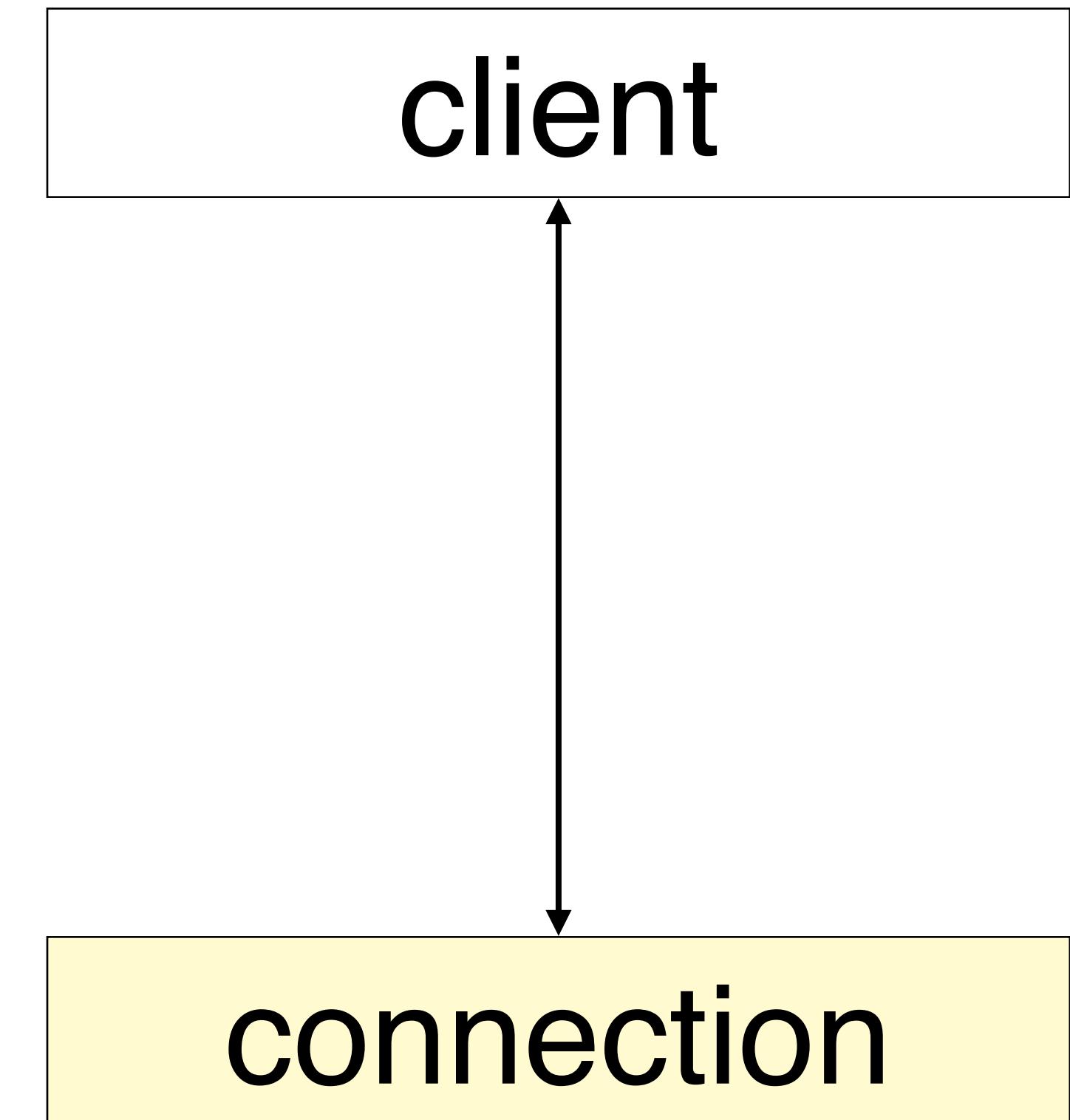
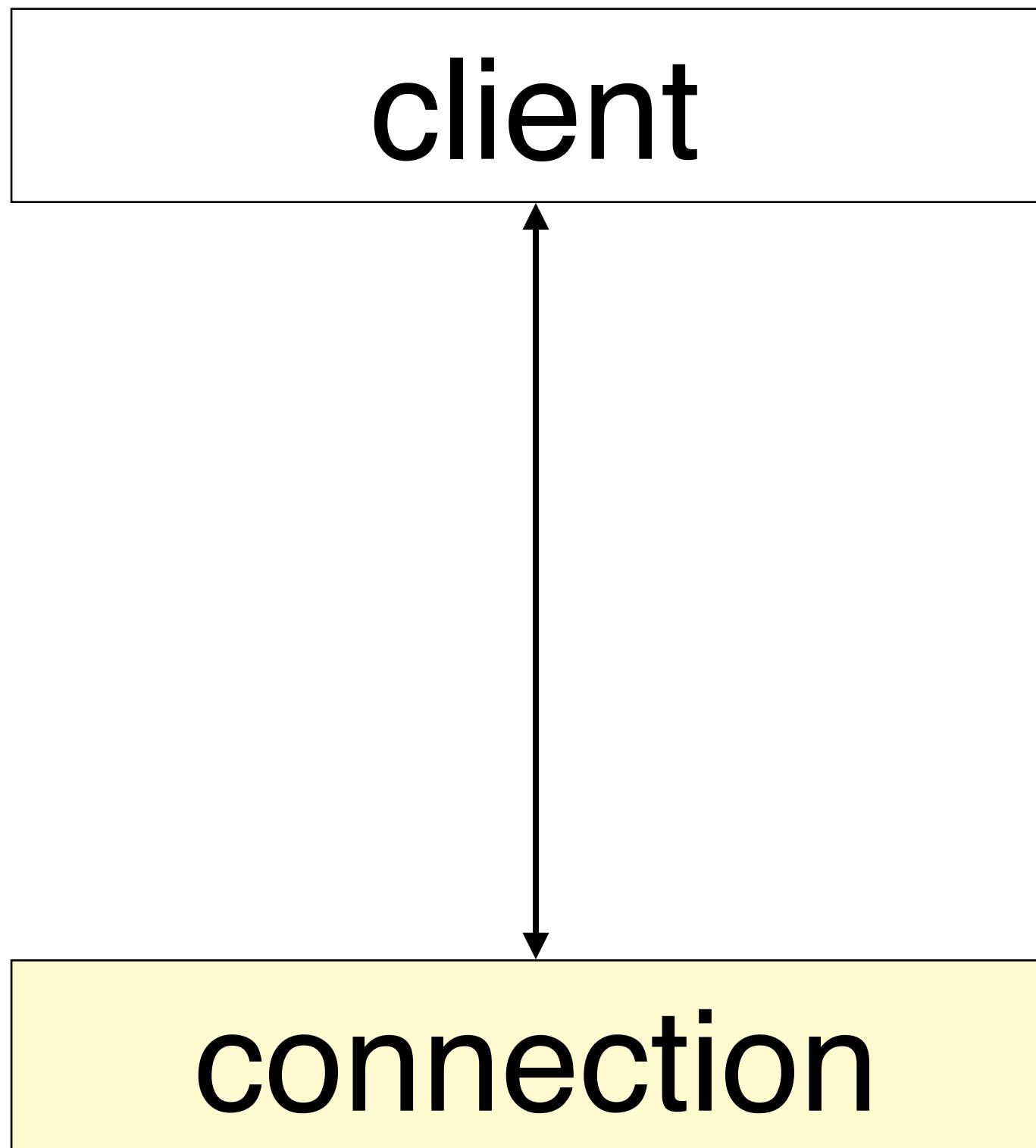
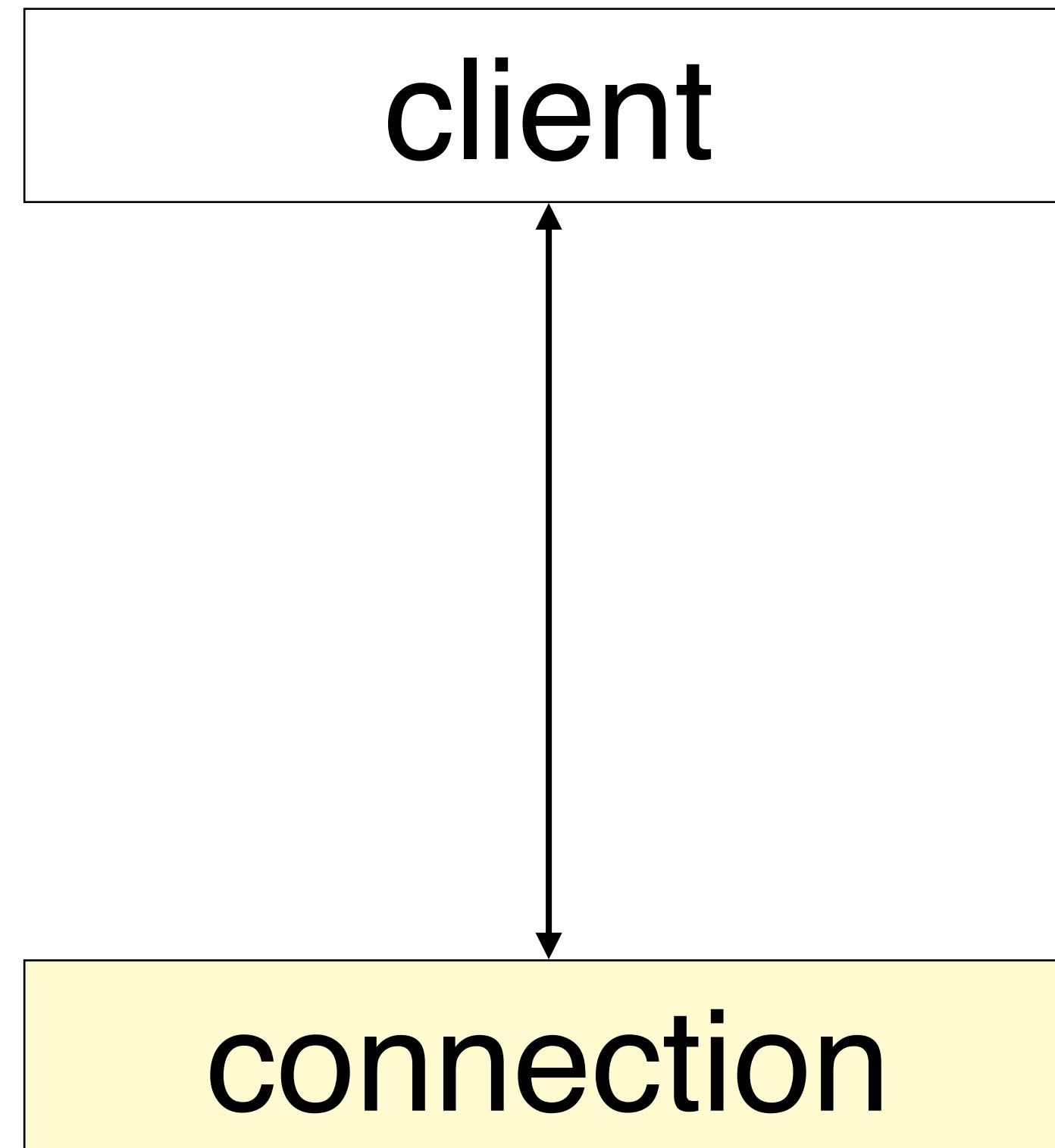
CPU

CPU

CPU







connection

spawn

calculation

send result & stop

Technical requirement	Server A	Server B
HTTP server	Nginx and Phusion Passenger	Erlang
Request processing	Ruby on Rails	Erlang
Long-running requests	Go	Erlang
Server-wide state	Redis	Erlang
Persistable data	Redis and MongoDB	Erlang
Background jobs	Cron, Bash scripts, and Ruby	Erlang
Service crash recovery	Upstart	Erlang

uniformity

simpler development
simpler testing
simpler deployment
simpler maintenance
improved collaboration

start simple
go far

Elixir IN ACTION

SECOND EDITION

Saša Jurić

MANNING



50% off using code
wm042919au

40% off using code
ctwgotochicago19

at manning.com

goto;
chicago'



Please

**Remember to
rate this session**

Thank you!