Faster-than-light coordination for networked systems with quantum non-local games

Venkat Arun, Vijay Chidambadam, Scott Aaronson

UT Austin

Should we find applications before a useful quantum computer exists?

A potential application in networked systems

- ✓ That run at room temperature and are inexpensive
- ✓ That uses mature technology from decades ago
- ✓ Where systems creativity is the missing ingredient

Sources of quantum advantage

Exponential speedups over classical computers

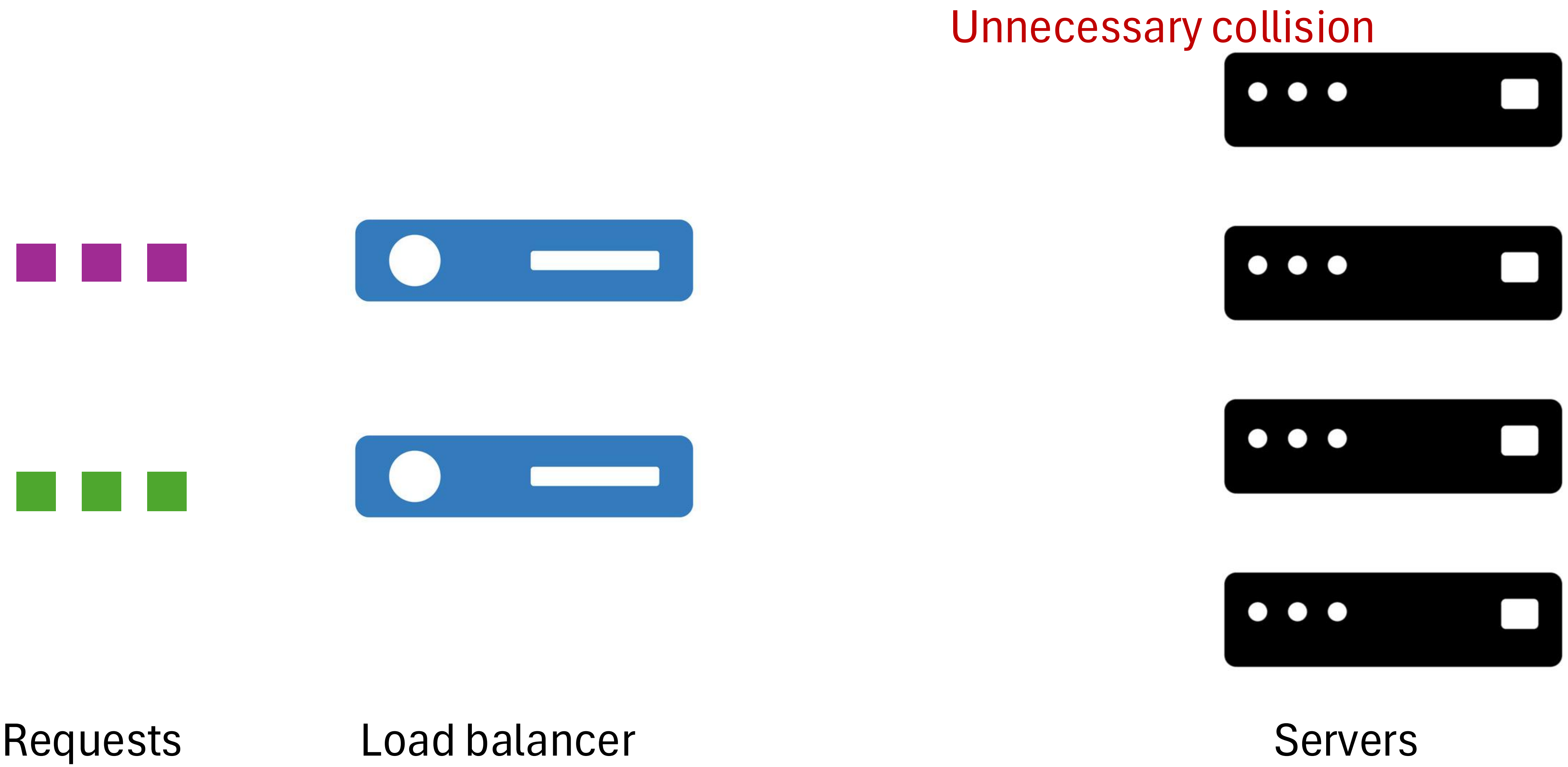
Quantum systems can make randomized decisions at distant locations, that are nevertheless correlated!

Communication with unconditional security

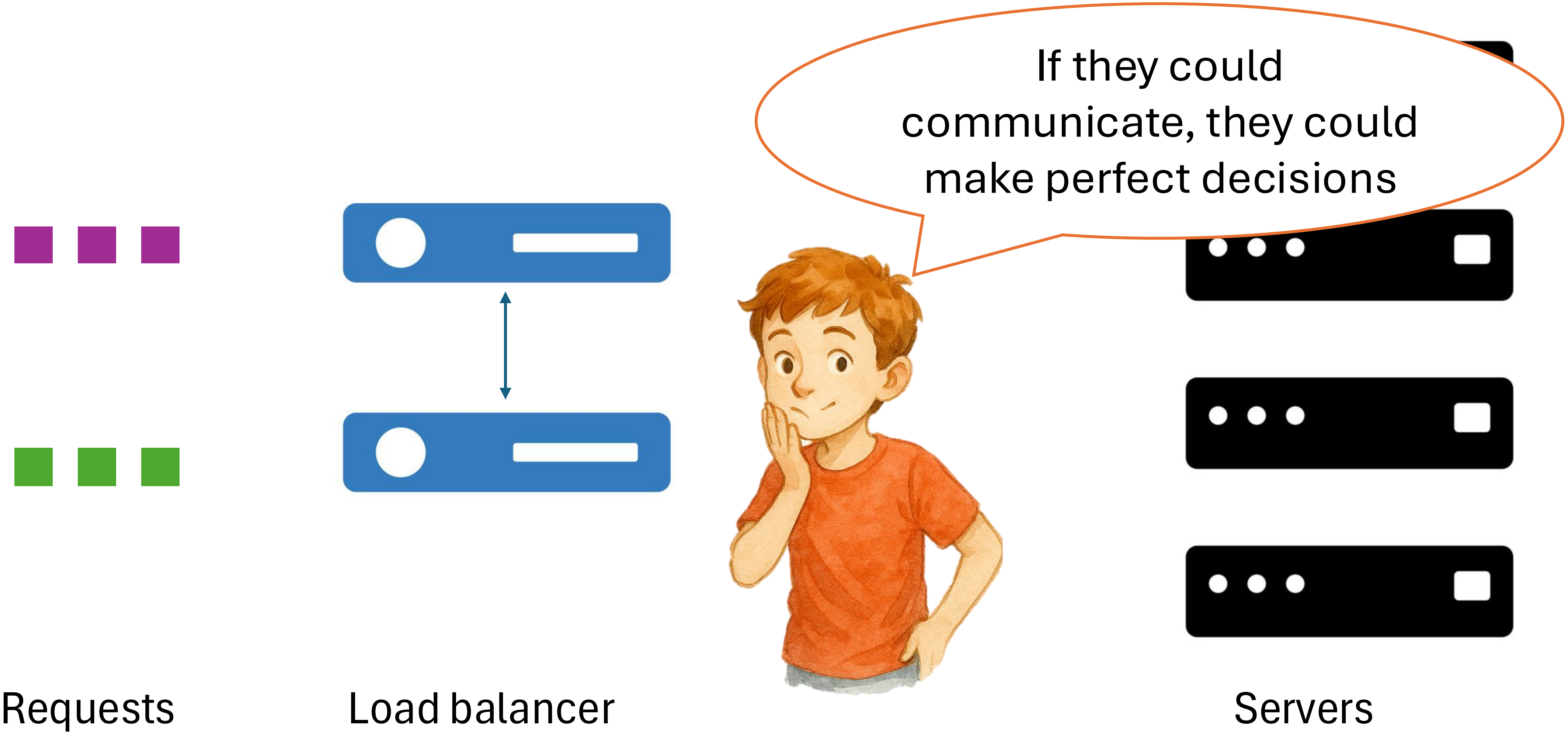
Spooky action at a distance New!

Networked systems make randomized decisions all the time!

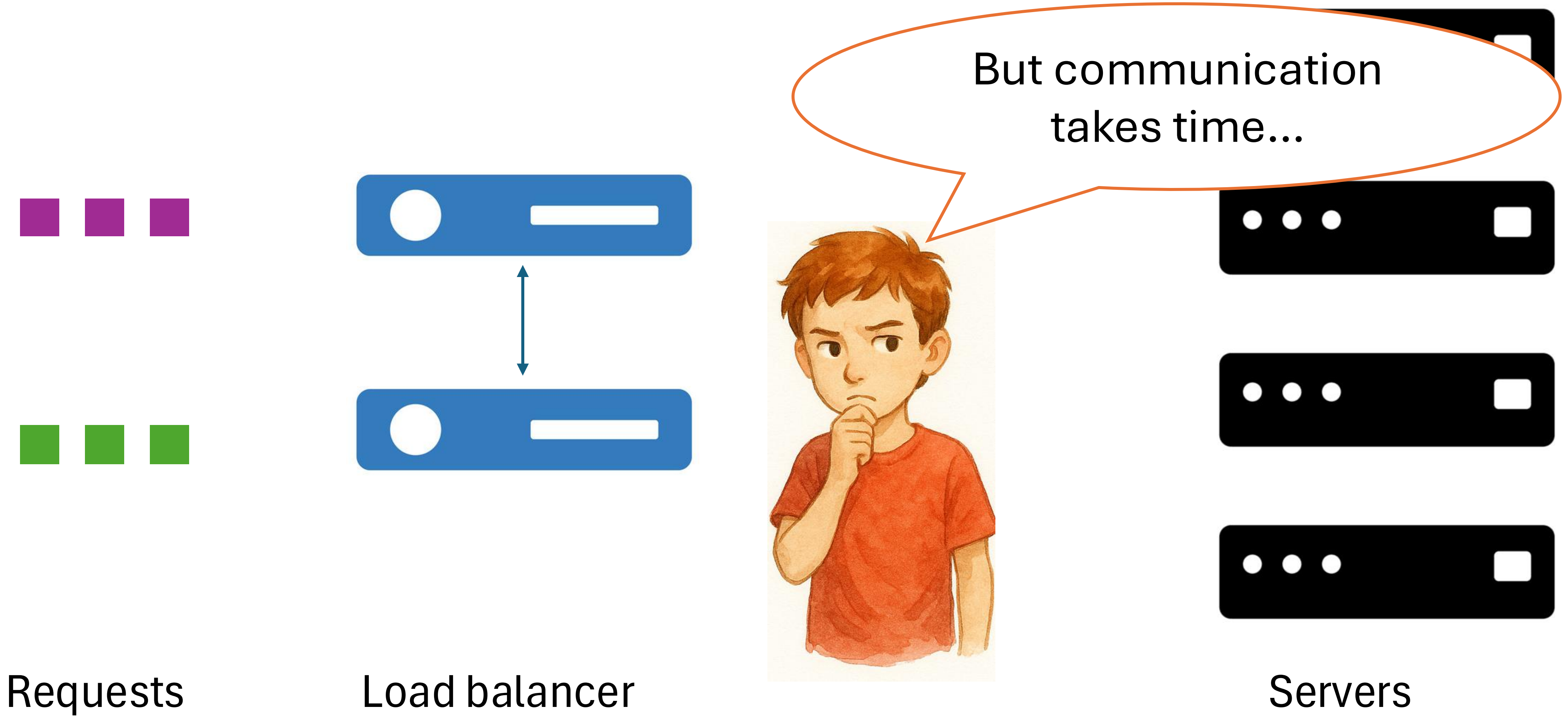
Networked systems make randomized decisions all the time!



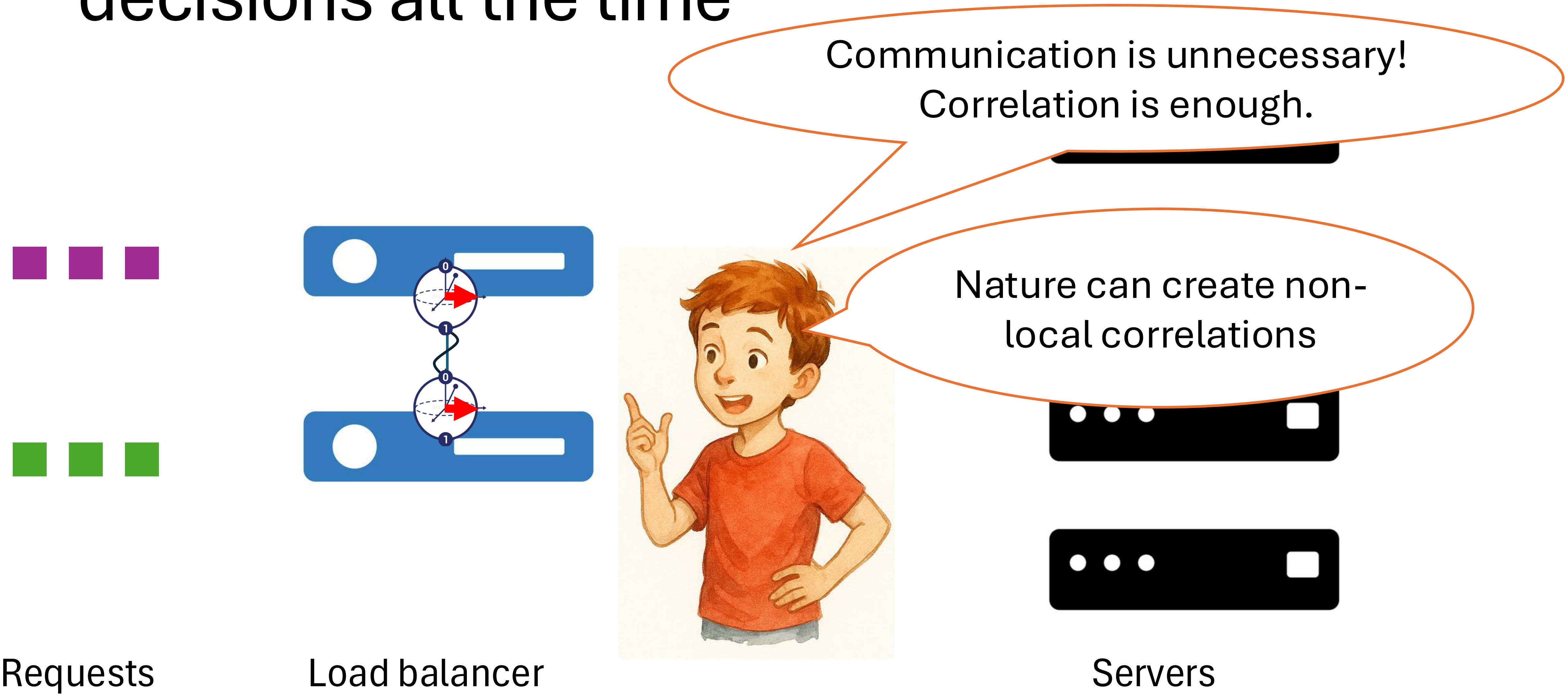
Networked systems make randomized decisions all the time



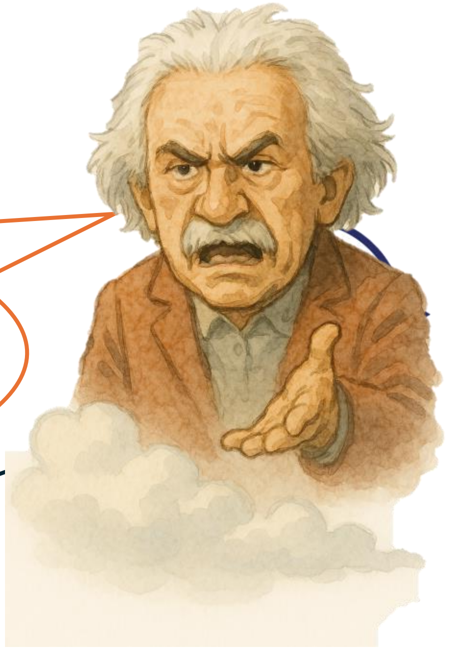
Networked systems make randomized decisions all the time



Networked systems make randomized decisions all the time

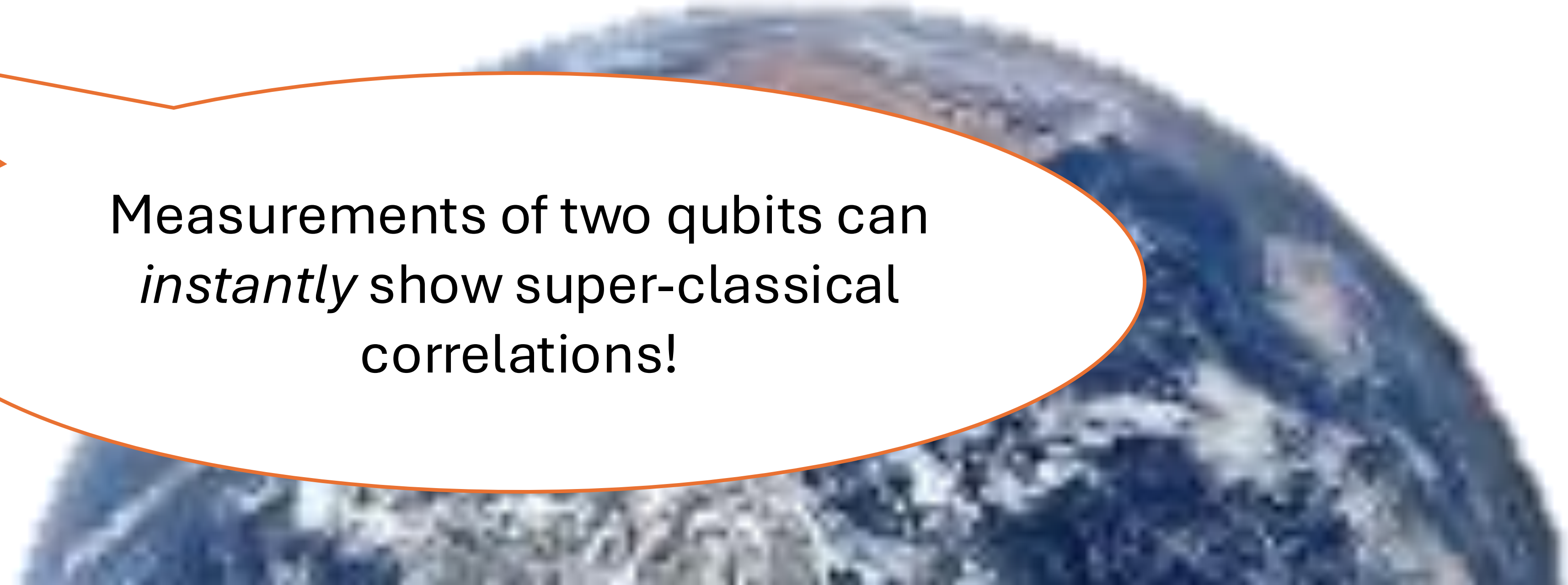


Spooky action at a distance

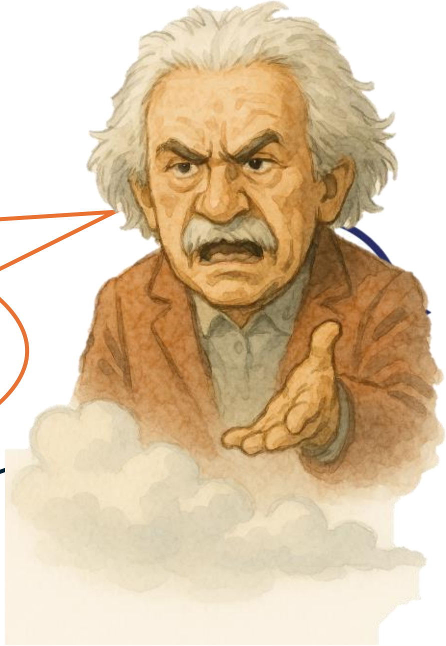


This goes faster than the speed of light? I explicitly prohibited such behavior!

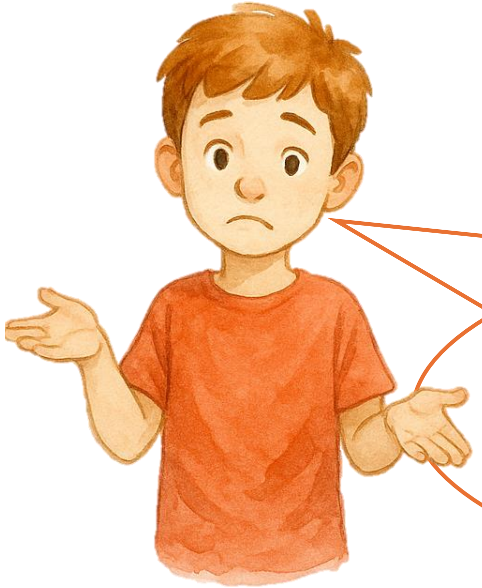
Measurements of two qubits can *instantly* show super-classical correlations!



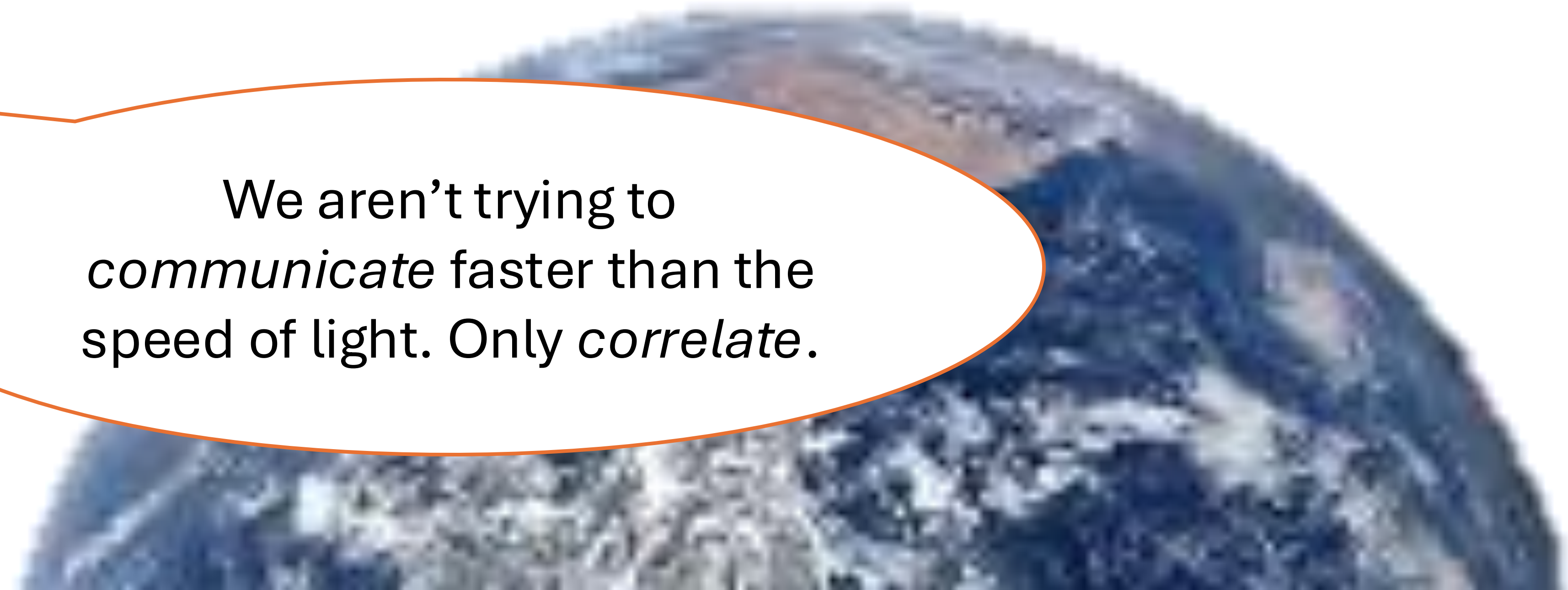
Spooky action at a distance



This goes faster than the speed of light? I explicitly prohibited such behavior!

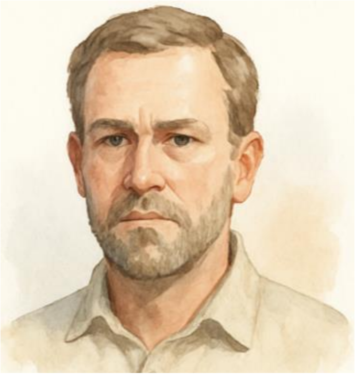


We aren’t trying to *communicate* faster than the speed of light. Only *correlate*.



Formalizing nonlocal correlations

Load balancer 1



Load balancer 2

Input A

Output A

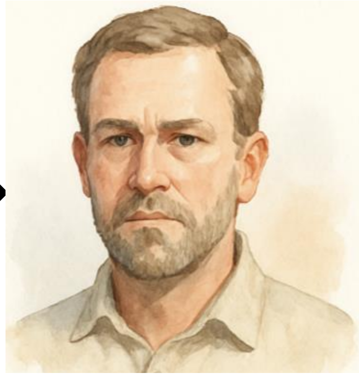
Output B

Input B

Formalizing nonlocal correlations

Load balancer 1

No communication



Load balancer 2

Output A

Output B

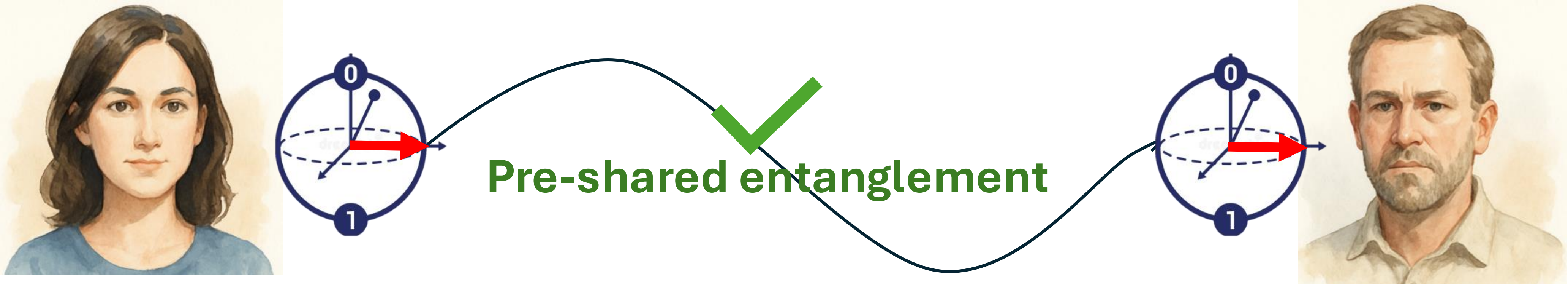
Judge(Input A , Output A , Output B , Input B) = Correlated/Uncorrelated

Input A

Input B

Goal: Maximize win probability

Formalizing nonlocal correlations



Load balancer 1

Load balancer 2

Output A

Output B

Judge(Input A , Output A , Output B , Input B) = Correlated/Uncorrelated

Input A

Goal: Maximize win probability

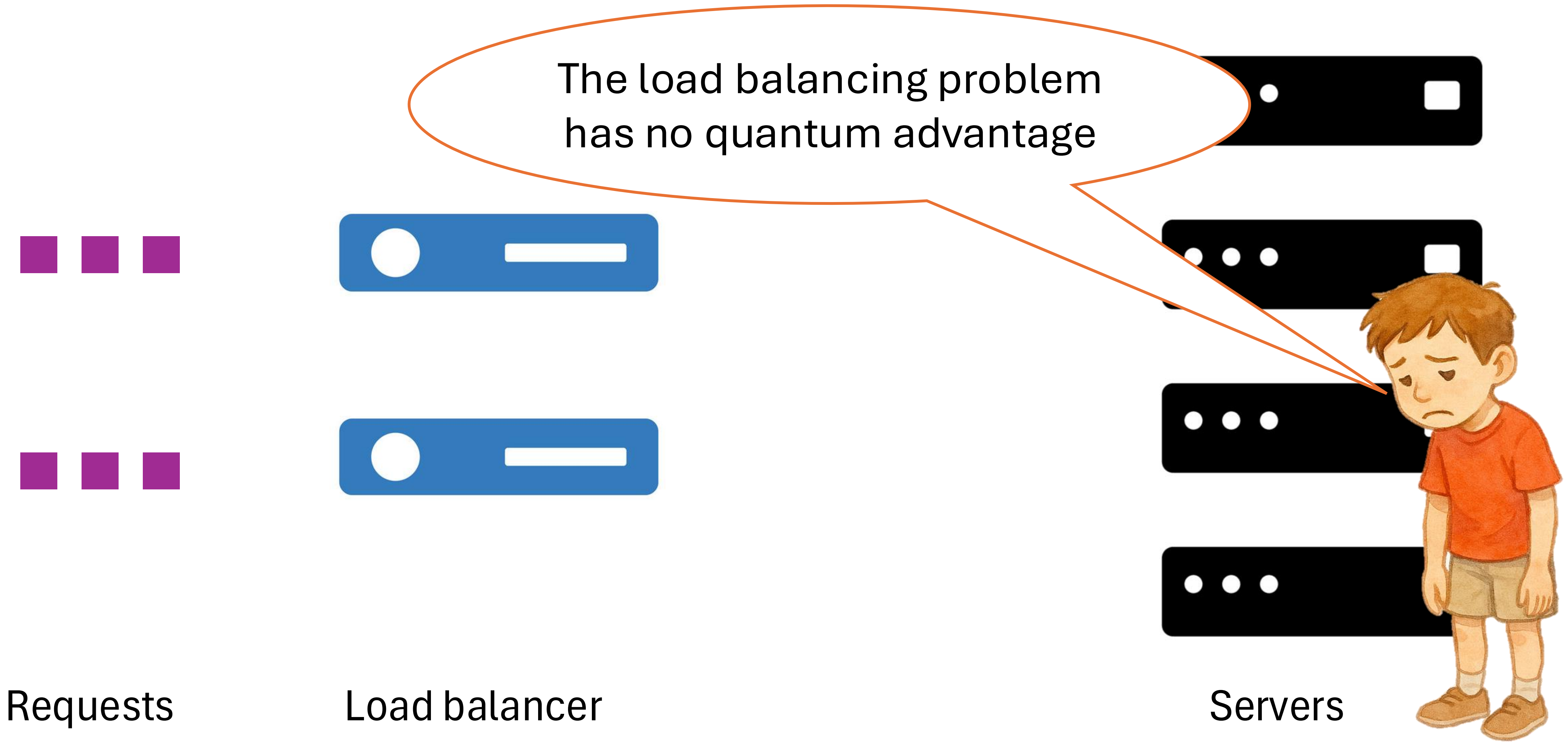
Input B

Looking for non-local games

There are many Judge functions where a quantum advantage exists

We need to find one that maps to a problem we care about

But such problems are hard to find…

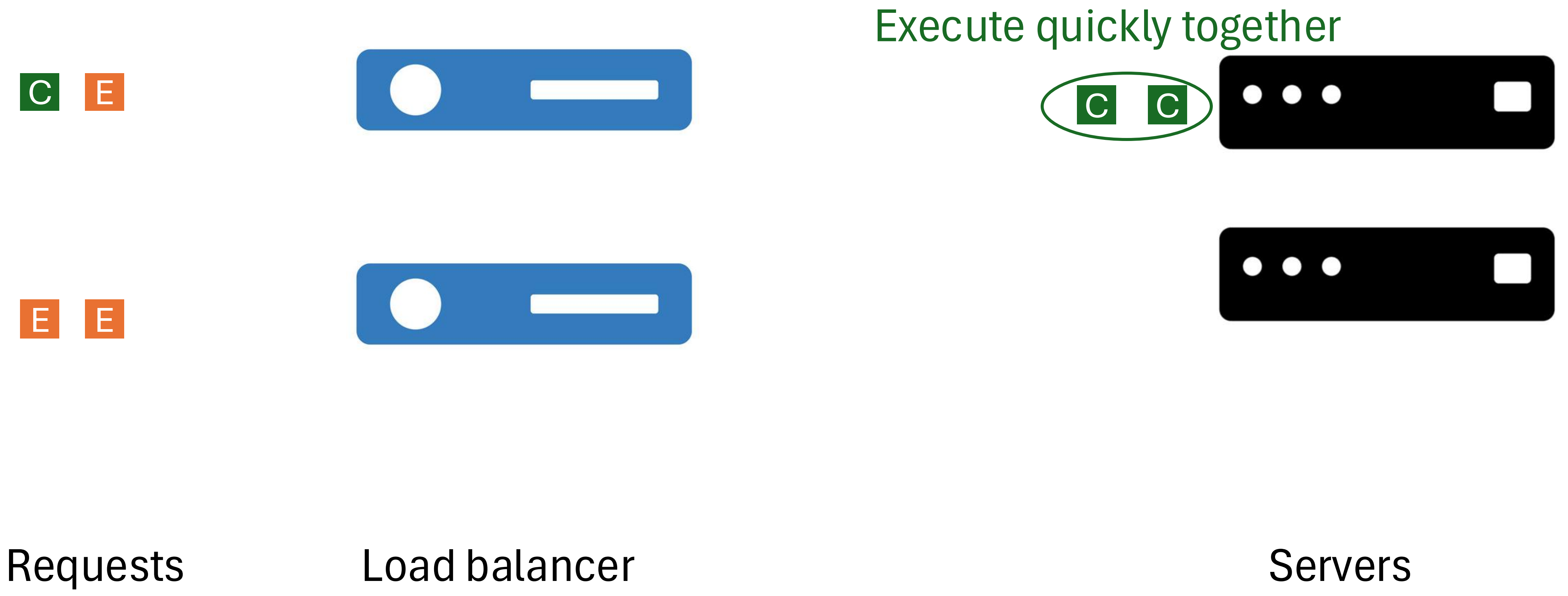

The load balancing problem
has no quantum advantage
Requests
Load balancer
Servers

Finally, a quantum advantage!

- C Two type-C tasks can be co-located for efficiency
- E Type-E tasks want exclusive access to a server

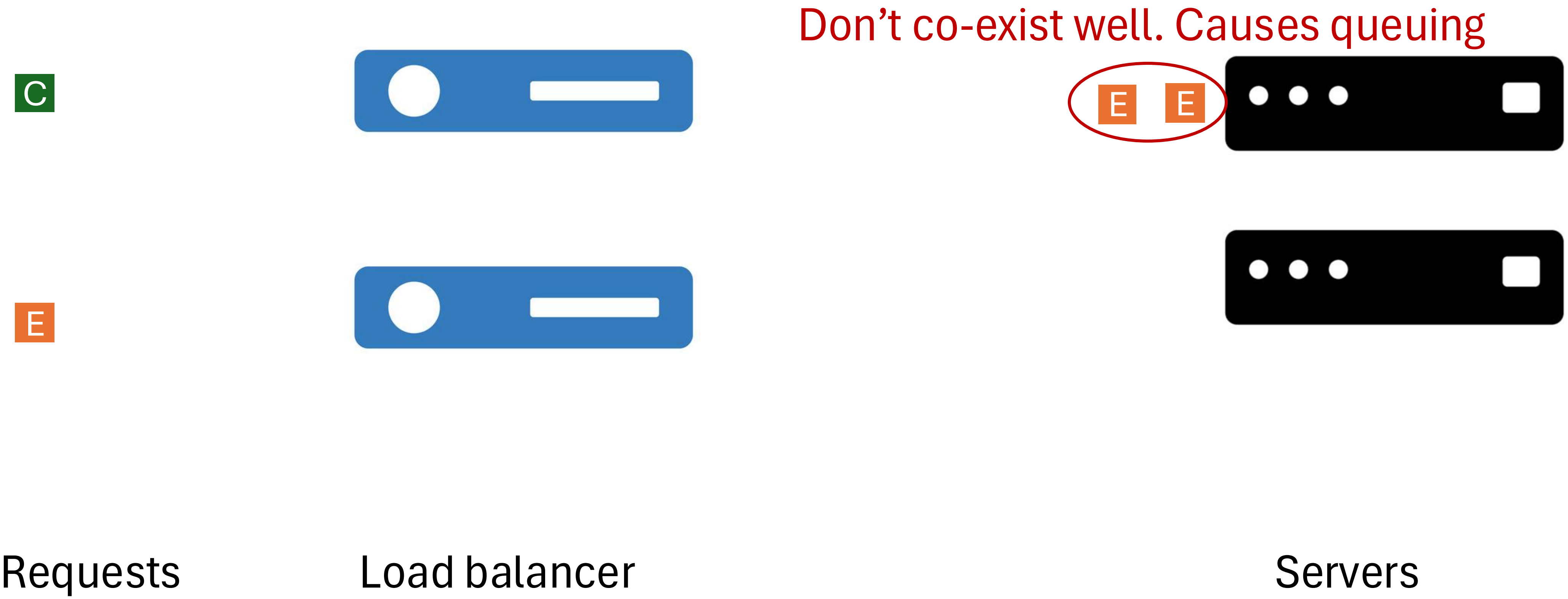
Finally, a quantum advantage!

- C Two type-C tasks can be co-located for efficiency
- E Type-E tasks want exclusive access to a server



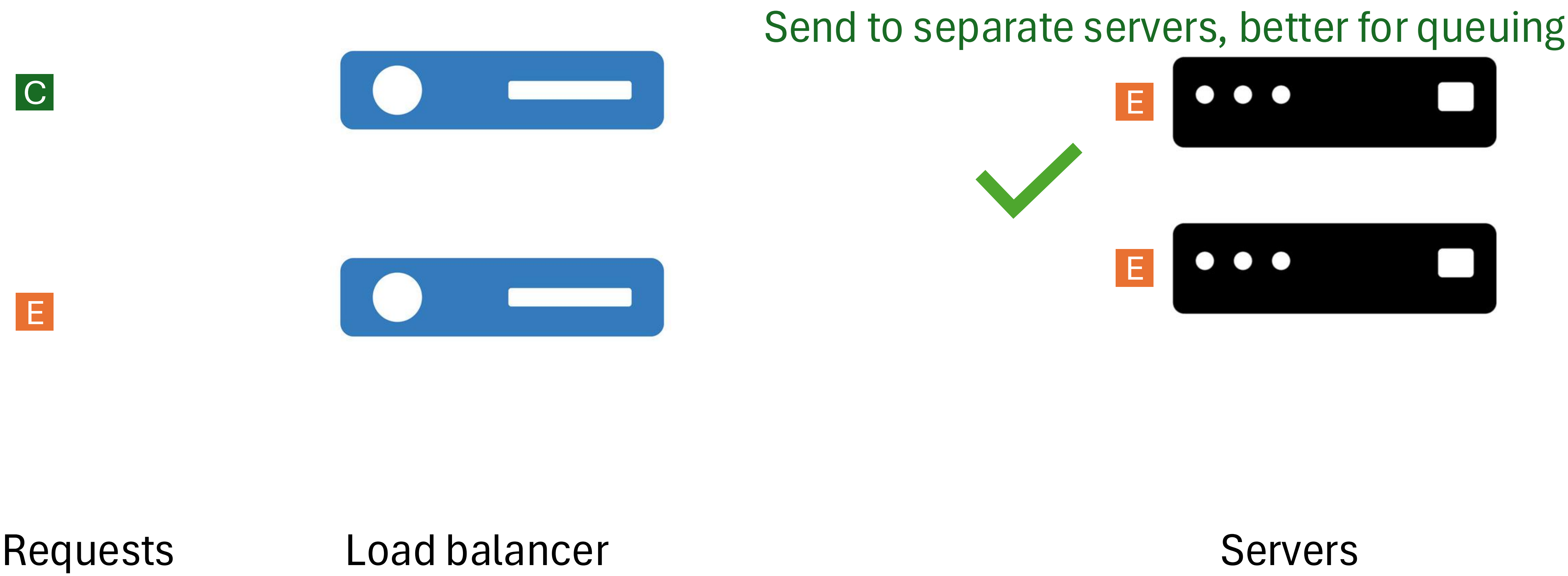
Finally, a quantum advantage!

- C Two type-C tasks can be co-located for efficiency
- E Type-E tasks want exclusive access to a server



Finally, a quantum advantage!

- C Two type-C tasks can be co-located for efficiency
- E Type-E tasks want exclusive access to a server



Finally, a quantum advantage

C Two type-C tasks

E Type-E tasks want exclusive

Send to separate better for queuing

This is the same as the CHSH game!

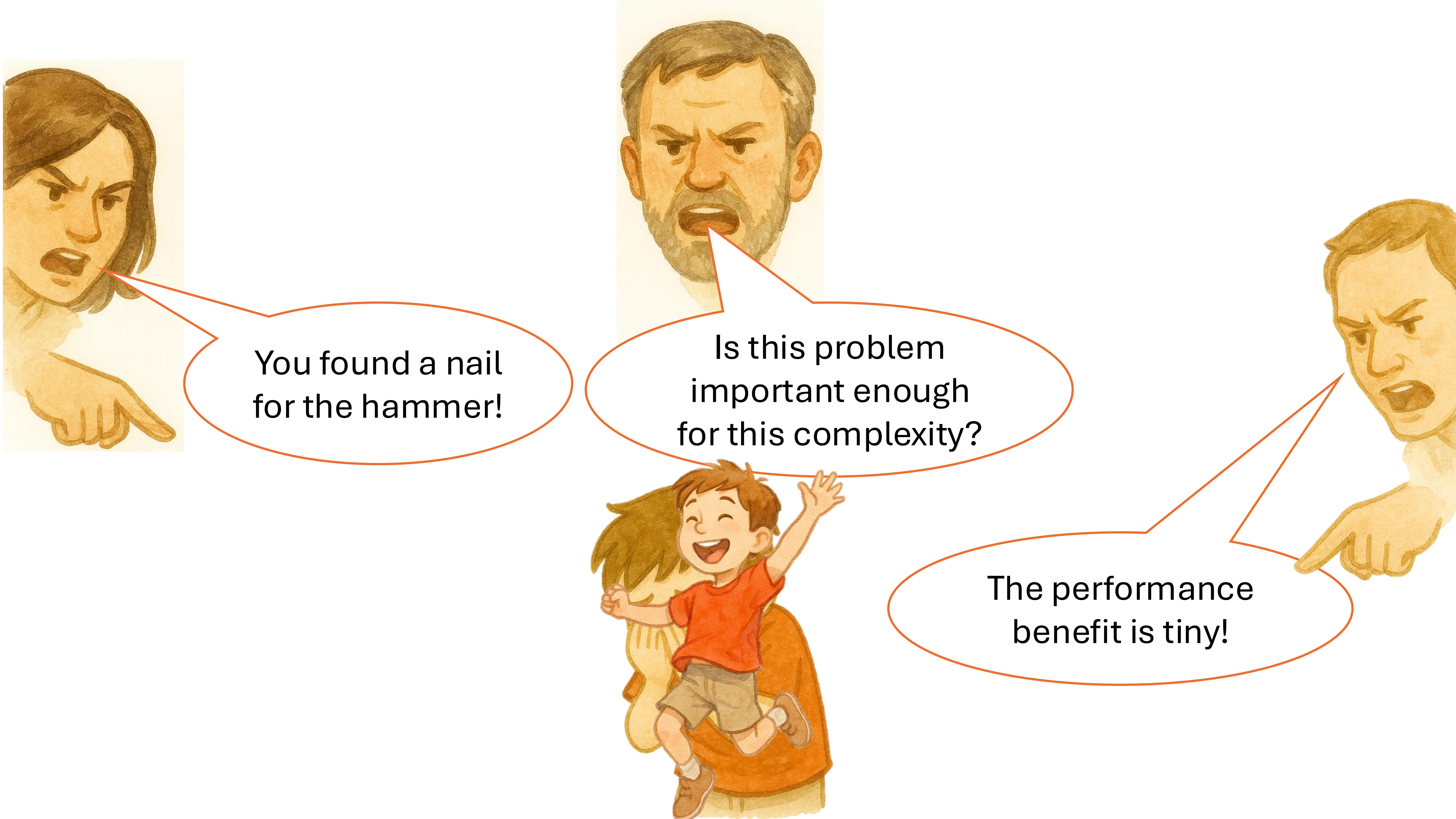
Needs only one entangled pair per decision

Extends to arbitrarily many load balancers and servers

Extends to arbitrarily many types of tasks

Requests

Servers


You found a nail for the hammer!
Is this problem important enough for this complexity?
The performance benefit is tiny!


This is HotNets. I'm just the ideas guy

Hope 1: We find an application where quantum systems offer such a compelling advantage that we *have* to deploy it.

Hope 2: This serves as an early success for quantum computers.

We found a niche application. Hopefully, you can do better!