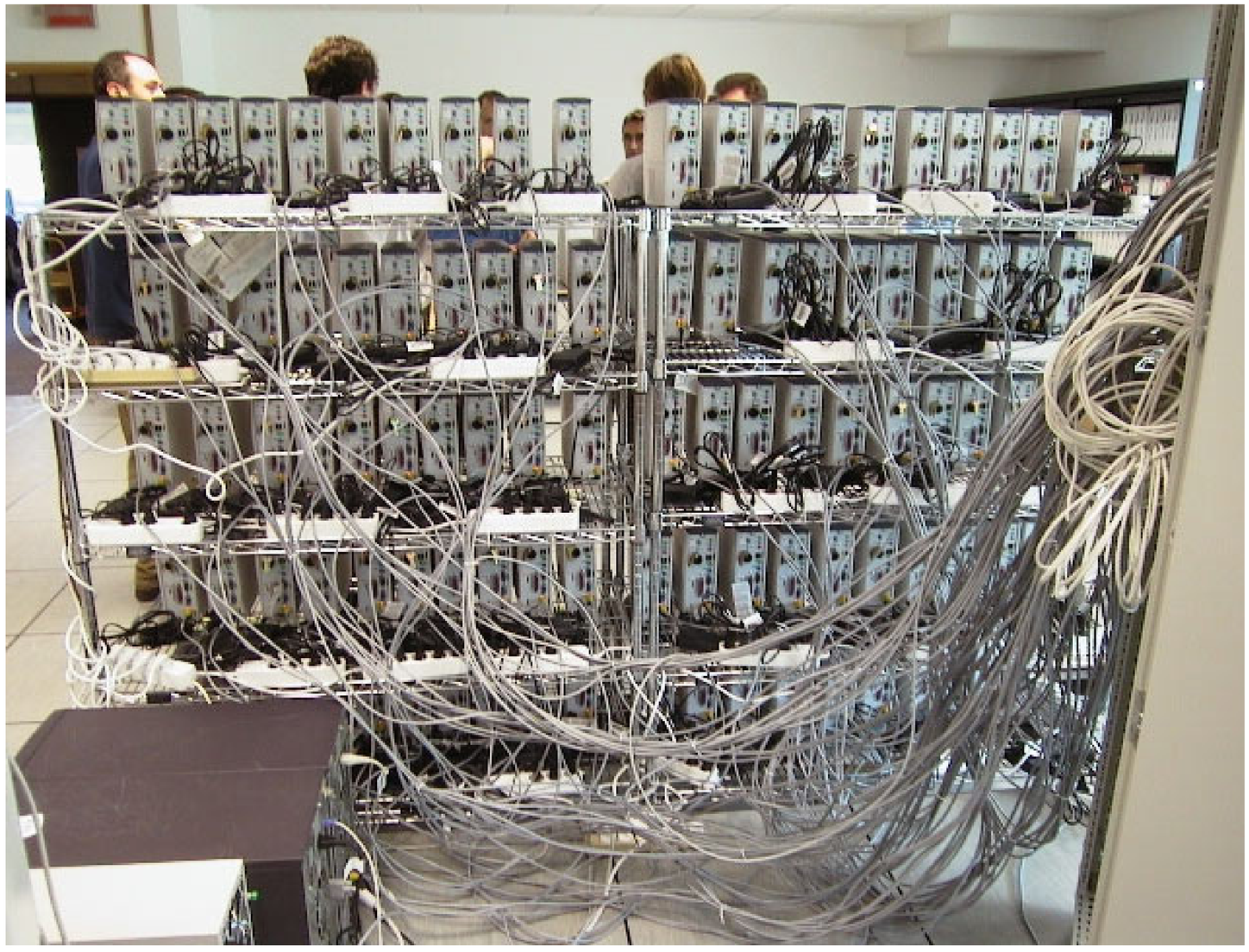
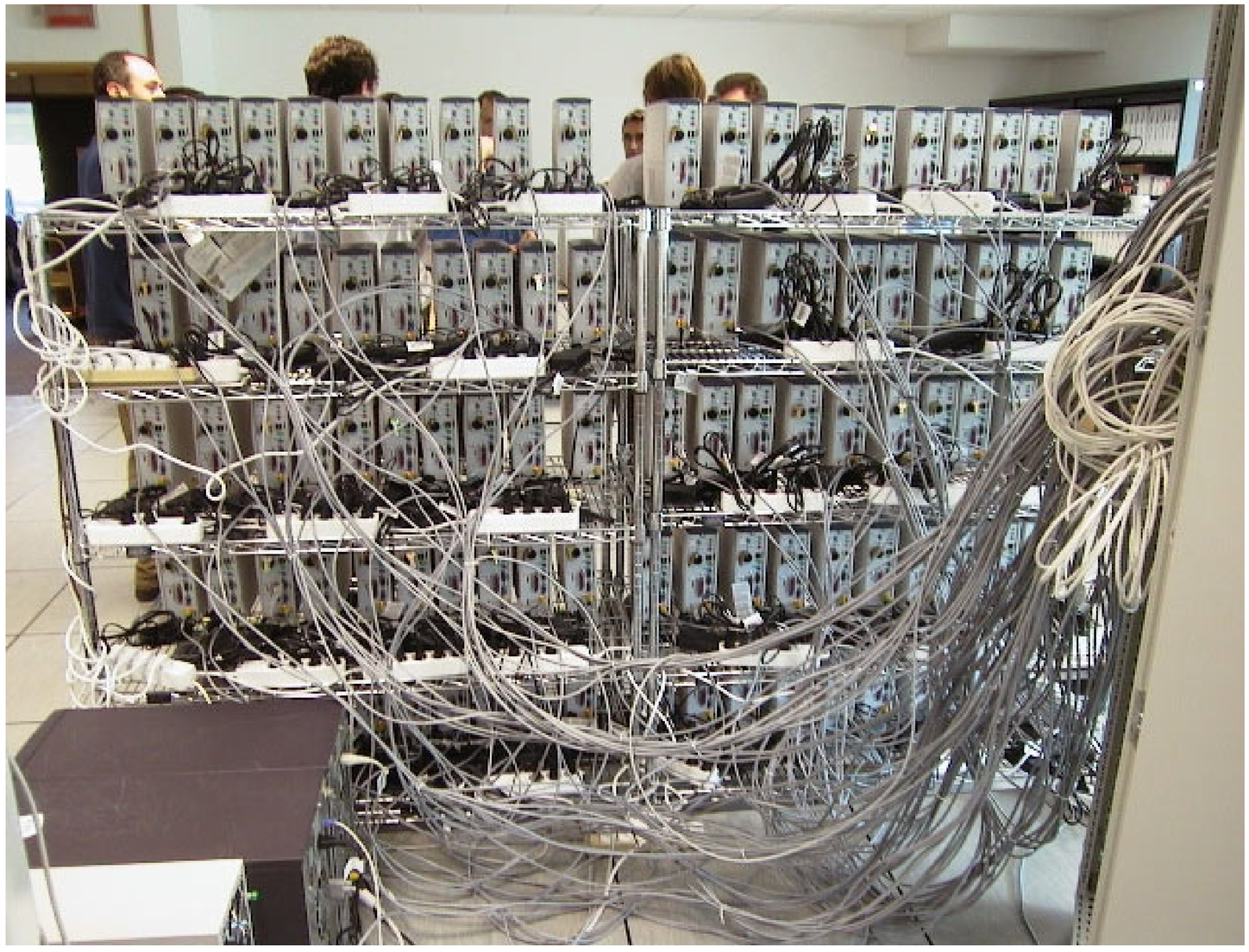
Yandex

Distributed Systems

Unreliable Components v2

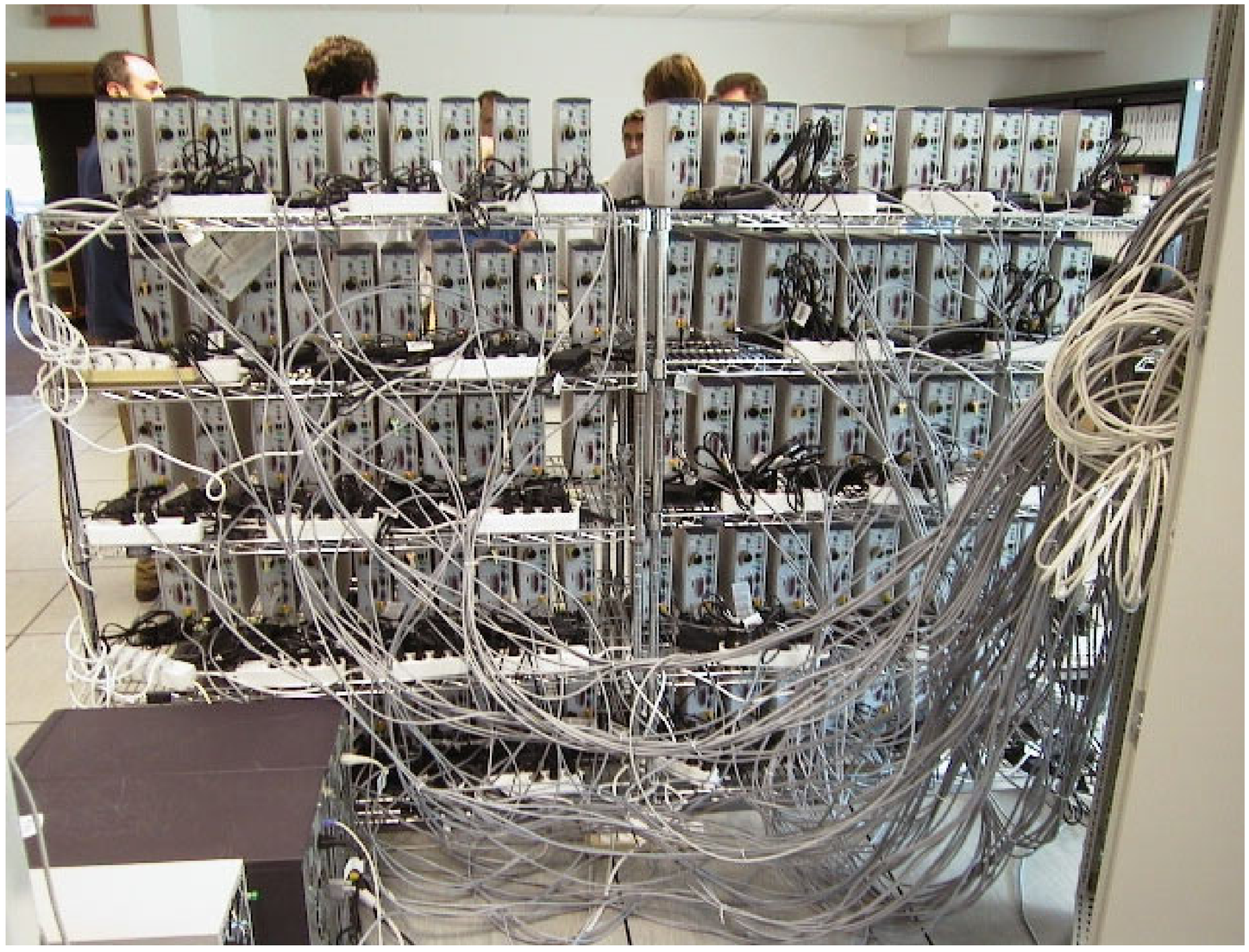


Link Failures:



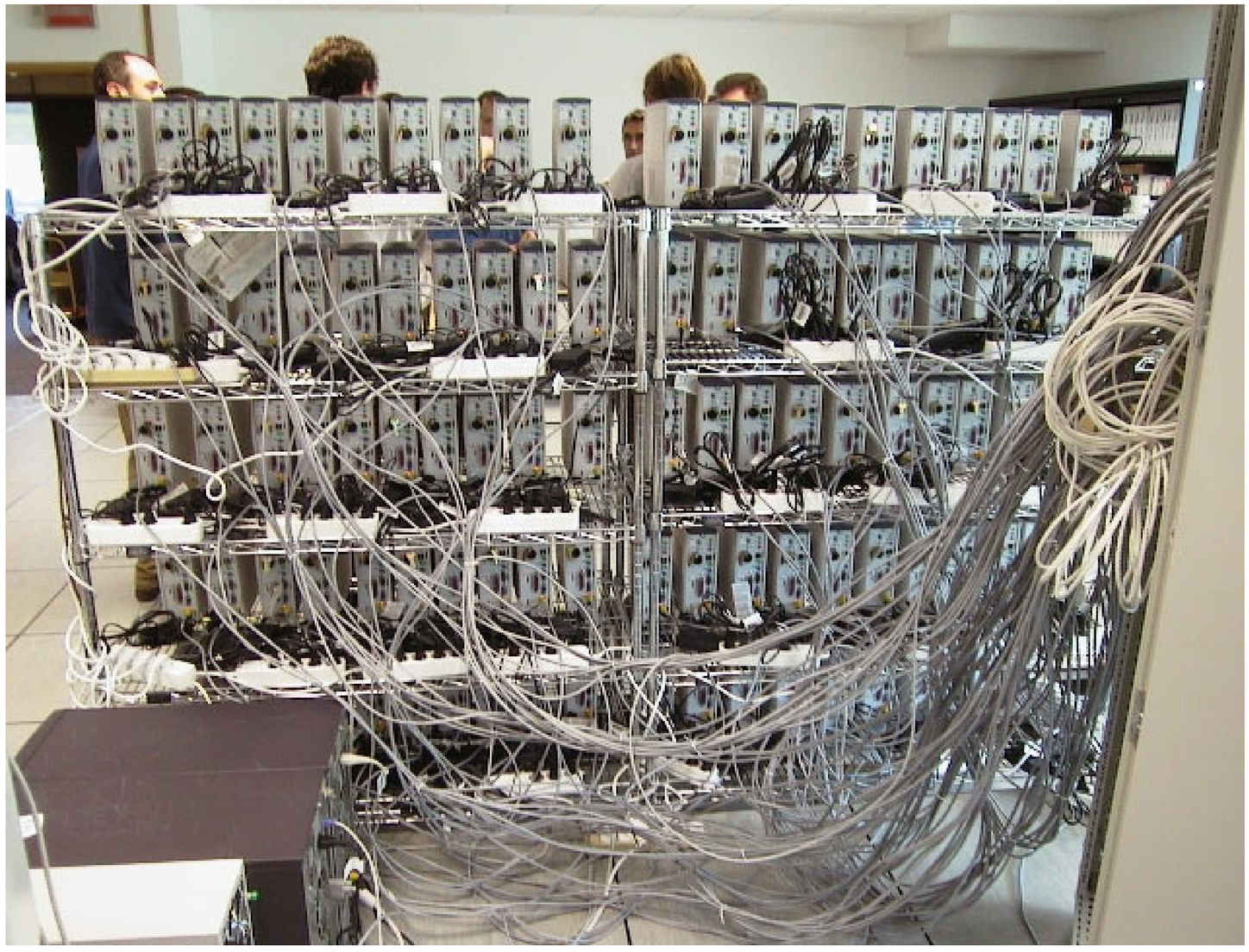
Link Failures:

- Perfect Link



Link Failures:

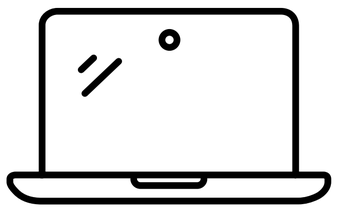
- Perfect Link
- Fail-loss Link



Link Failures:

- Perfect Link
- Fail-loss Link
- Byzantine

Perfect Link



send $(a_1, a_2, ..., a_k)$

receive $(b_1, b_2, ..., b_k)$

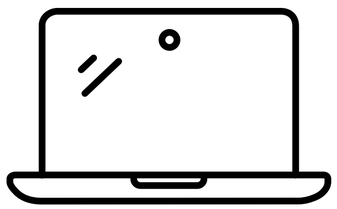
$b_1 = a_1$

$b_2 = a_2$

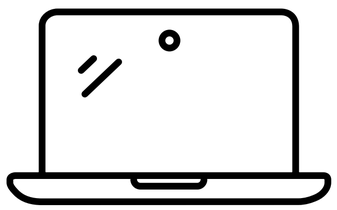
...

$b_k = a_k$

Fair-Loss Link



send $(a_1, a_2, ..., a_k)$

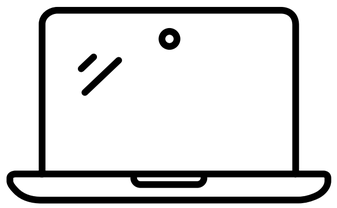


receive $(b_1, b_2, ..., b_l)$

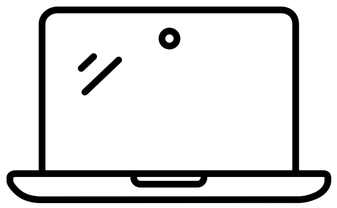
$l \leq k$

$\{b_i\} \in \{a_j\}$

Fair-Loss Link



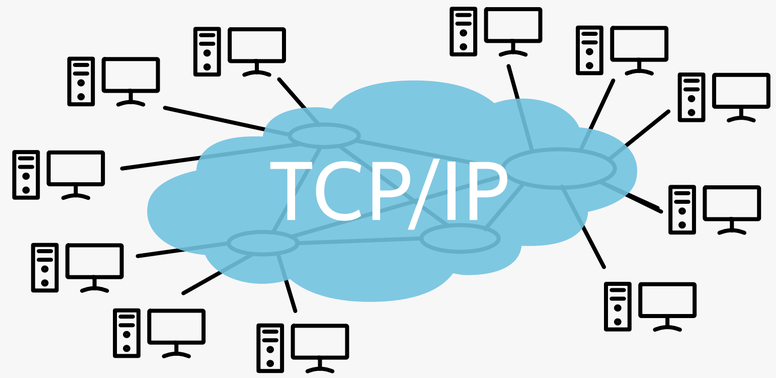
send $(a_1, a_2, ..., a_k)$

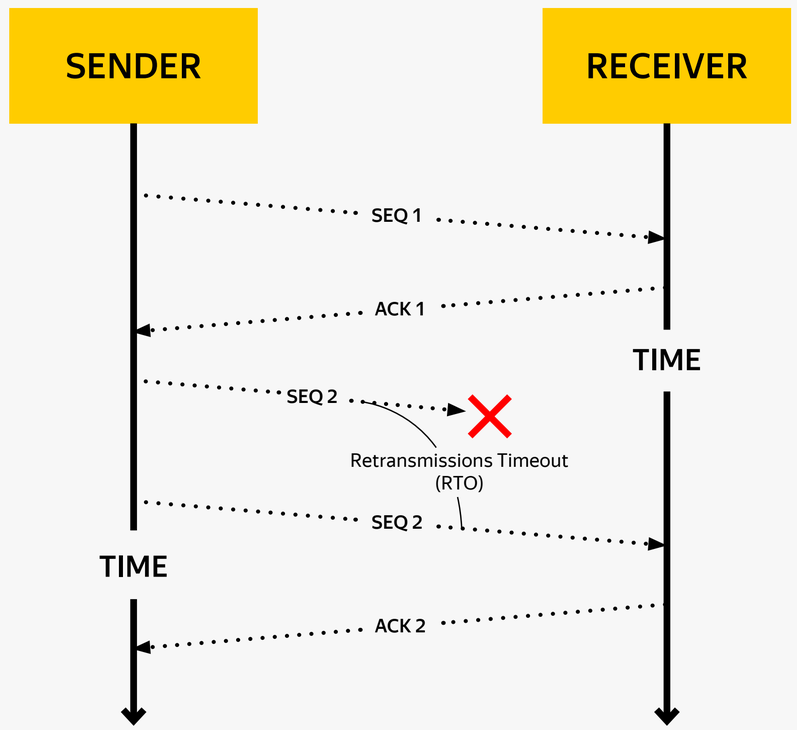


receive $(b_1, b_2, ..., b_l)$

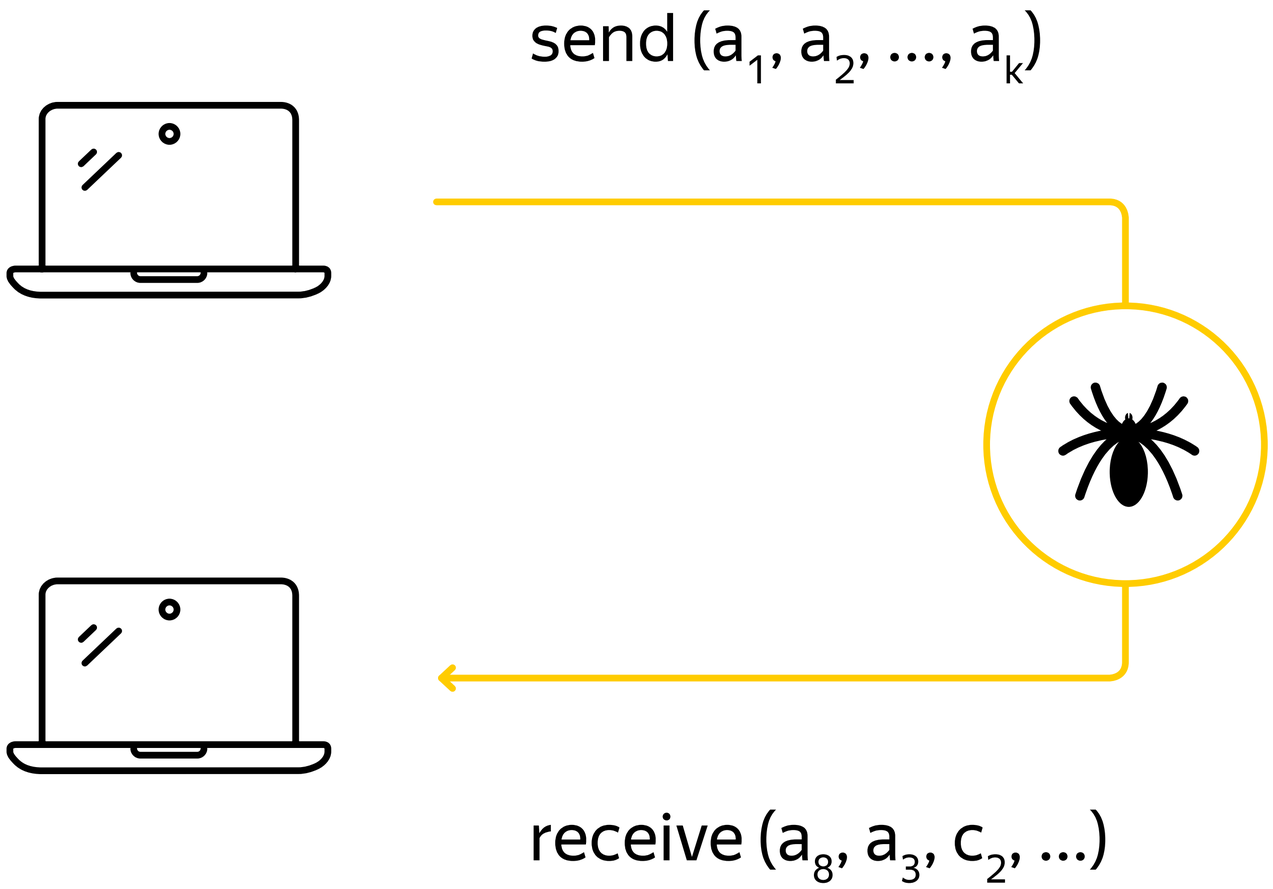
$l \leq k$

$\{b_i\} \in \{a_j\}$





Fair-Loss Link



The Two Generals Paradox

1975

SOME CONSTRAINTS AND TRADEOFFS
IN THE DESIGN OF
NETWORK COMMUNICATIONS*

E. A. Akkoyunlu
K. Ekanadham
R. V. Huber†
Department of Computer Science
State University of New York at Stony Brook

Notes on Data Base Operating Systems

Author: Jim Gray

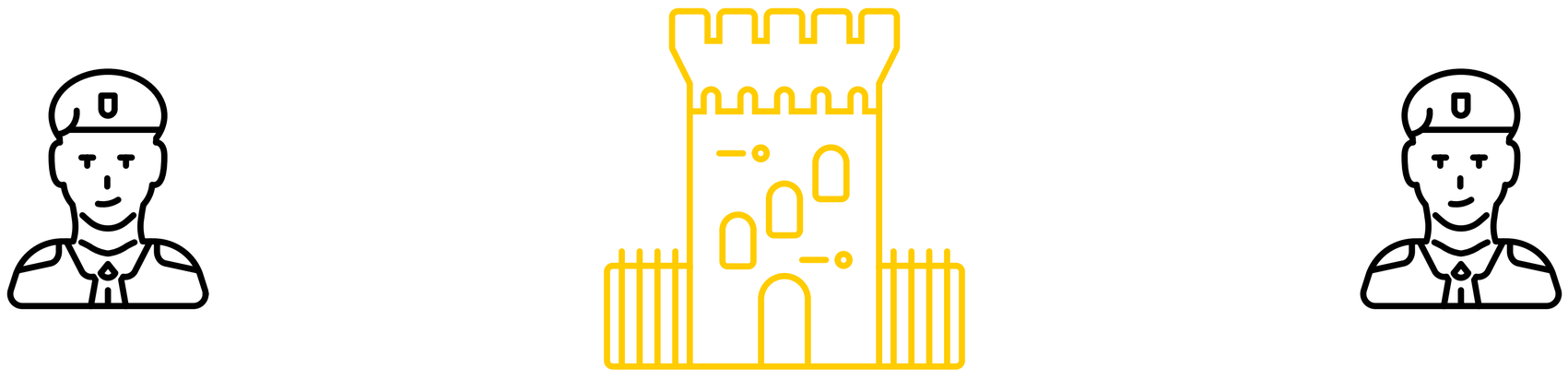
1978 Article

Published in:
· Proceeding
Operating Systems, An Advanced Course
Pages 393 - 481
Springer-Verlag London, UK ©1978
table of contents ISBN:3-540-08755-9

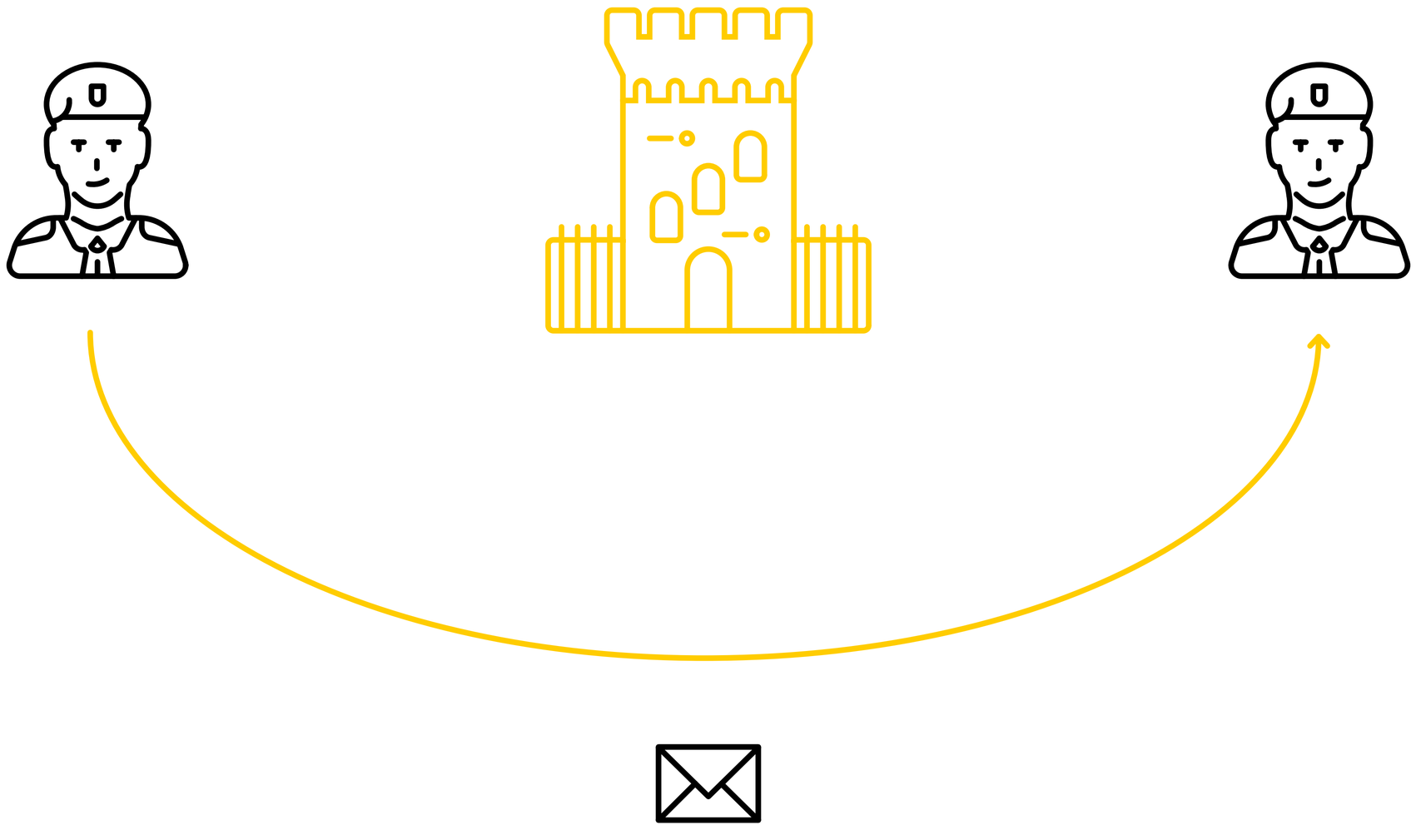
Bibliometrics

· Citation Count: 630
· Downloads (cumulative): n/a
· Downloads (12 Months): n/a
· Downloads (6 Weeks): n/a

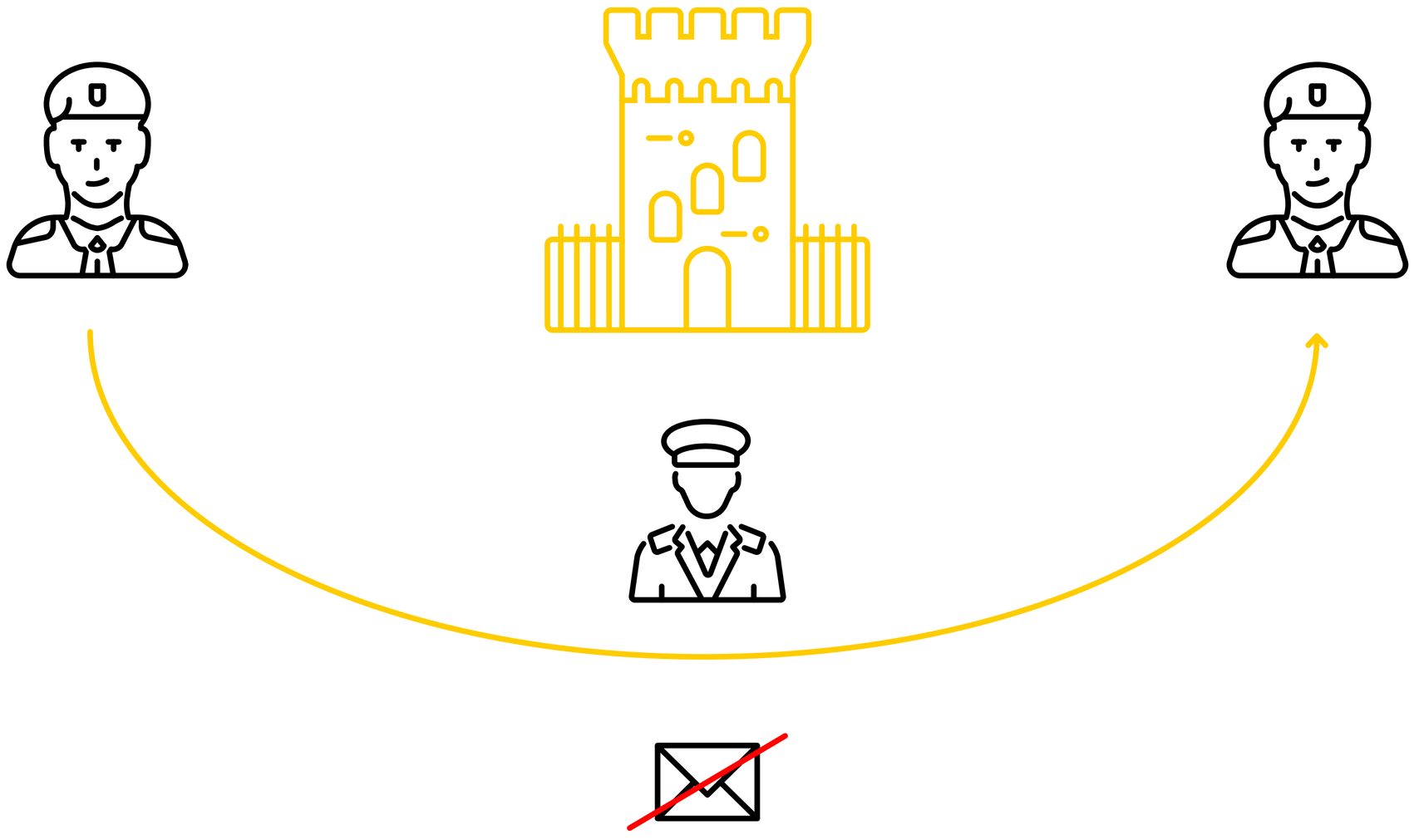
The Two Generals Problem



The Two Generals Problem



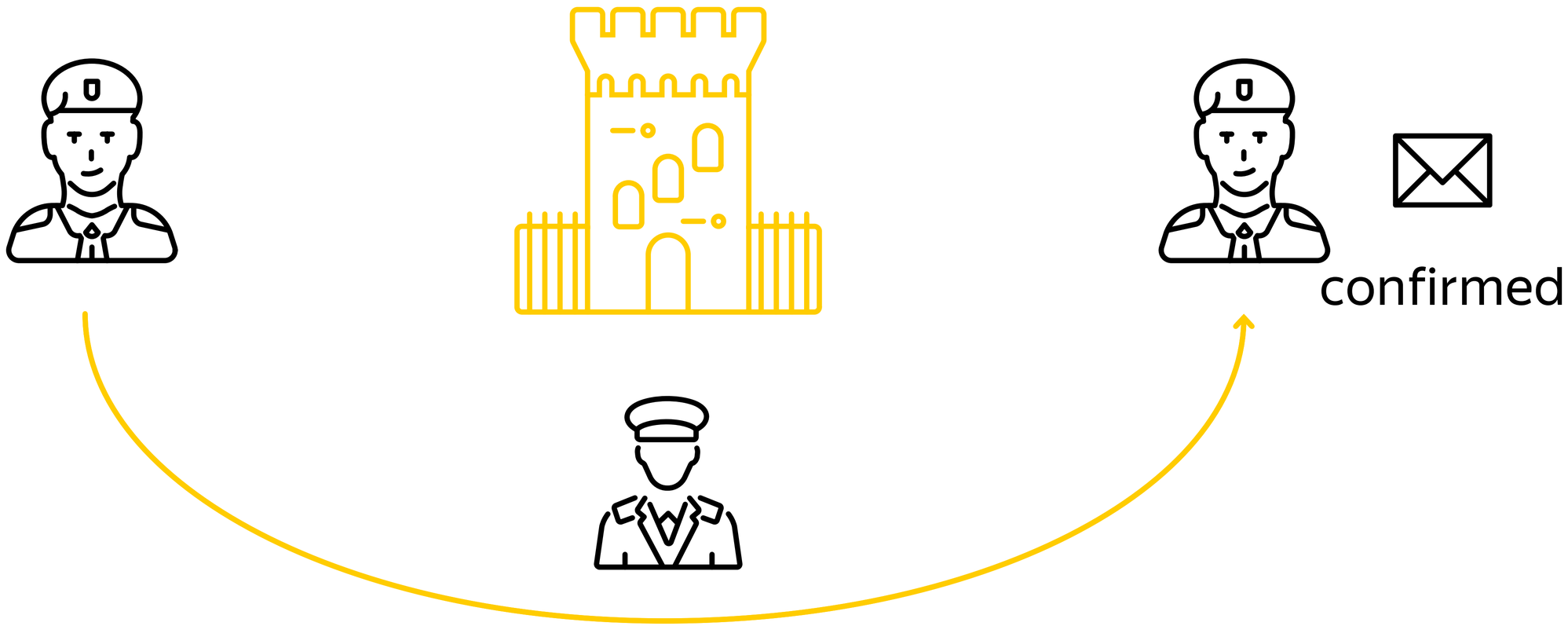
The Two Generals Problem



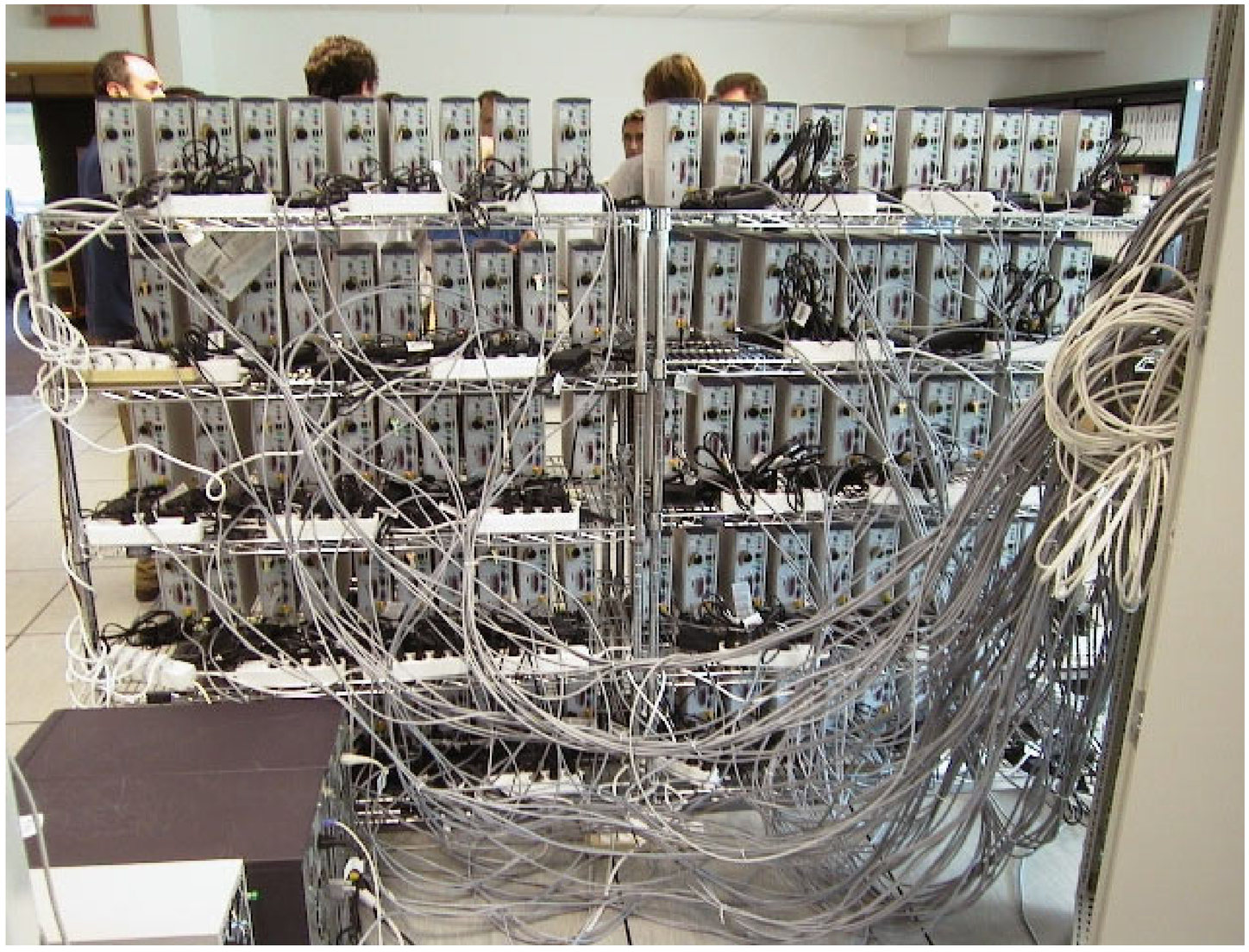
The Two Generals Problem

The Two Generals Problem

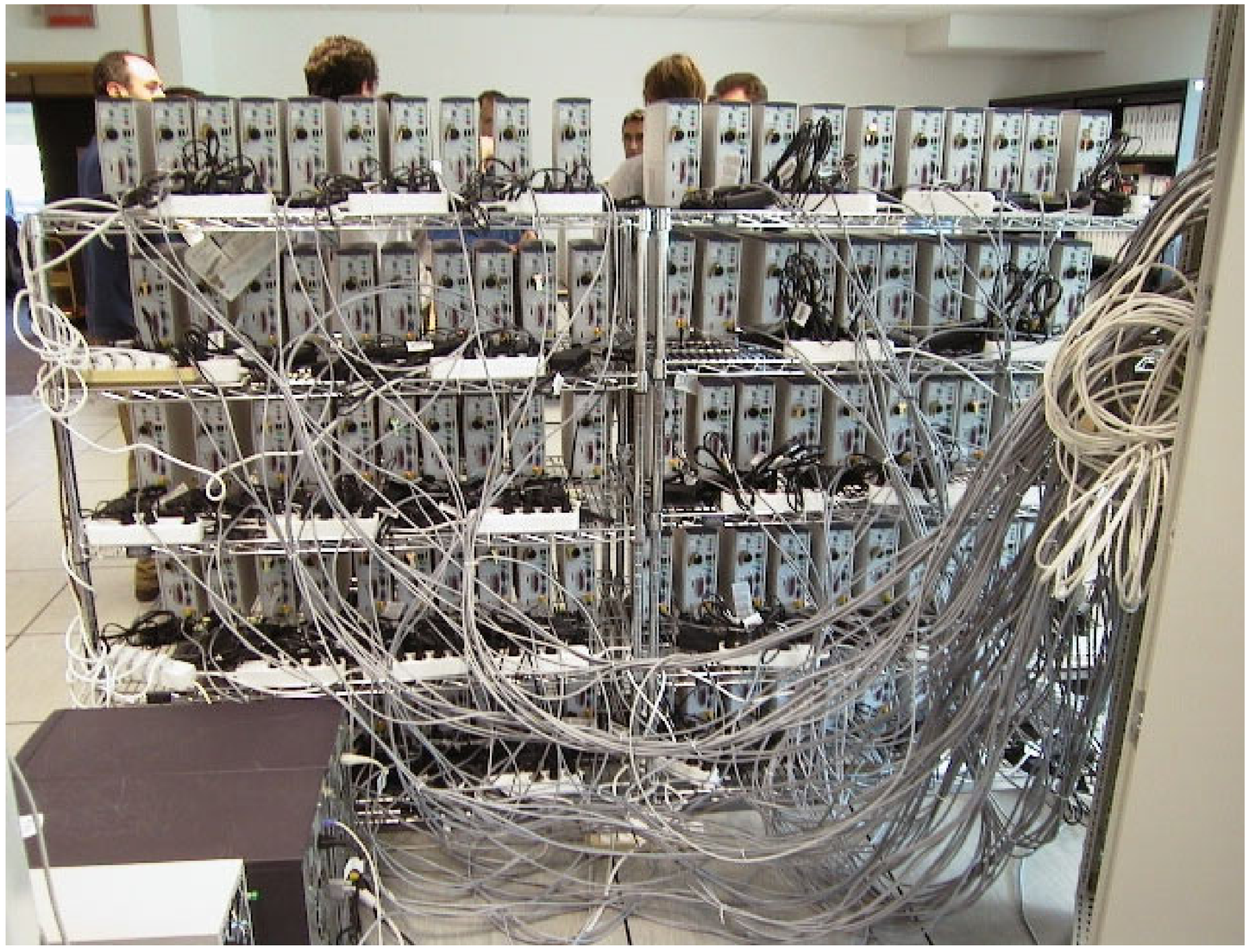
The Two Generals Problem



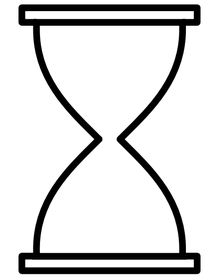
The Two Generals Problem



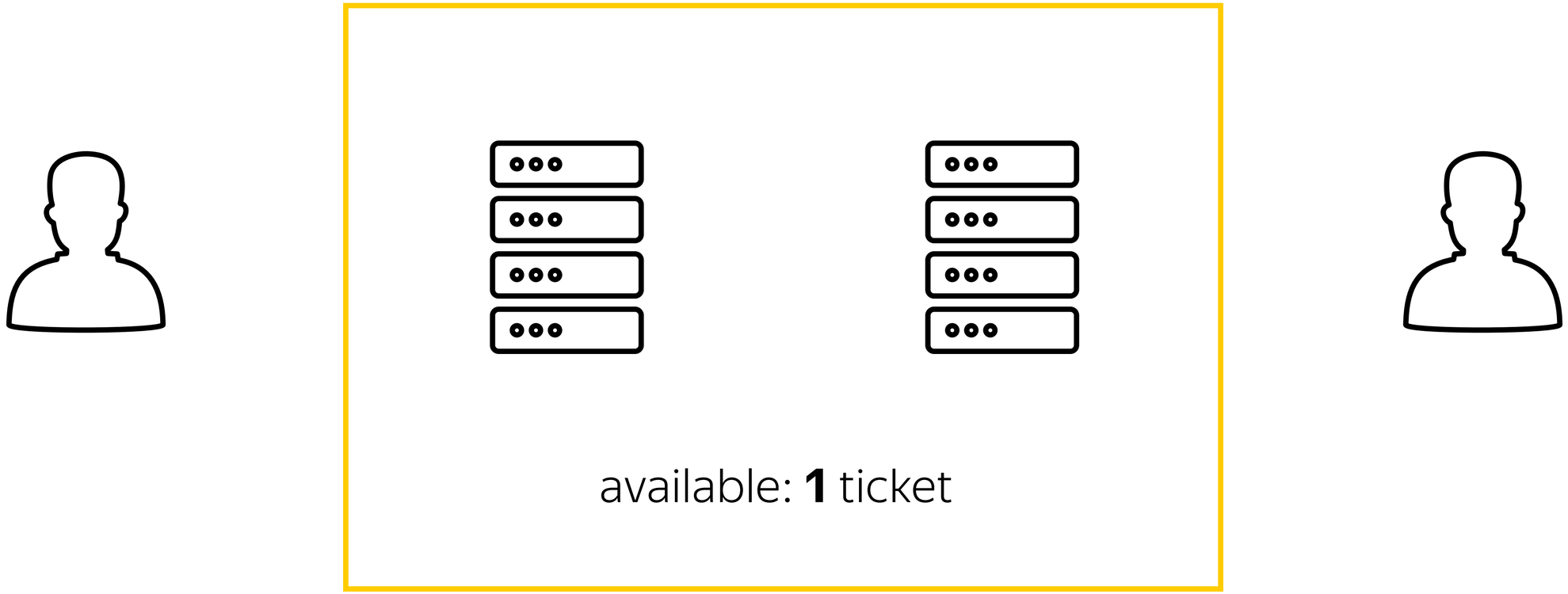
- Nodes
- Links
- ???



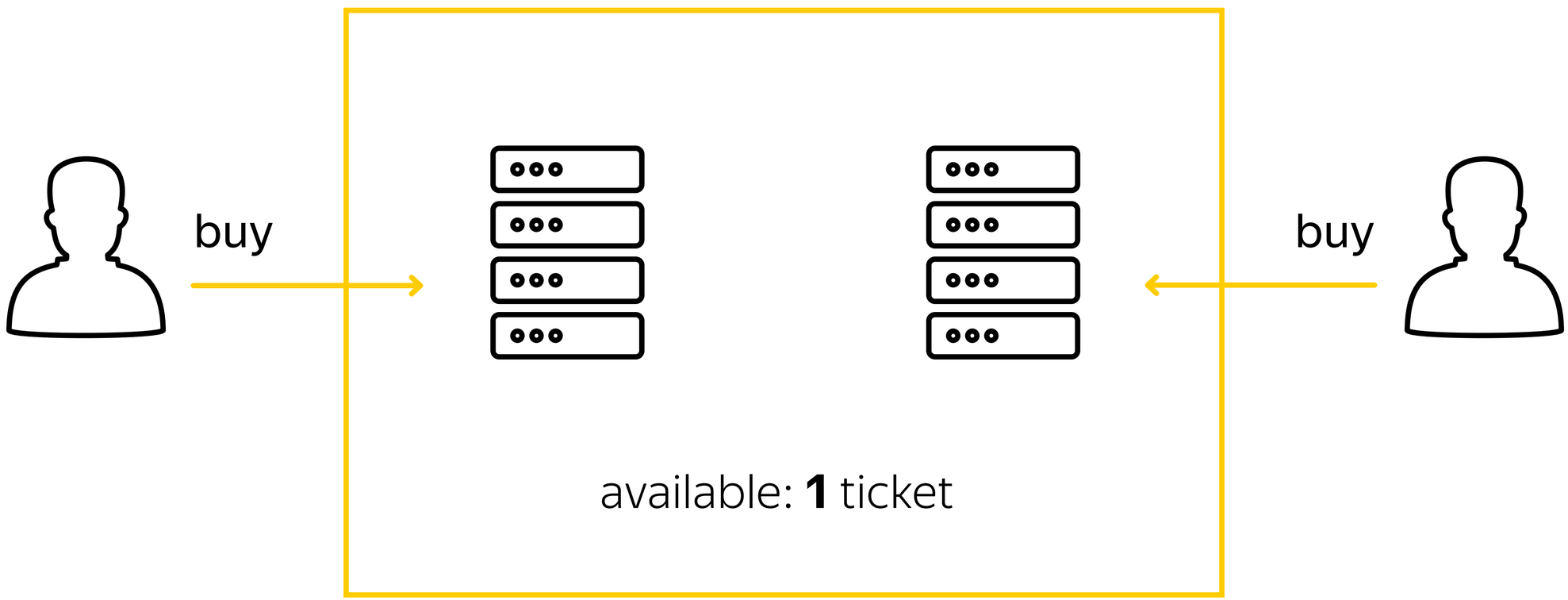
- Nodes
- Links
- Clock



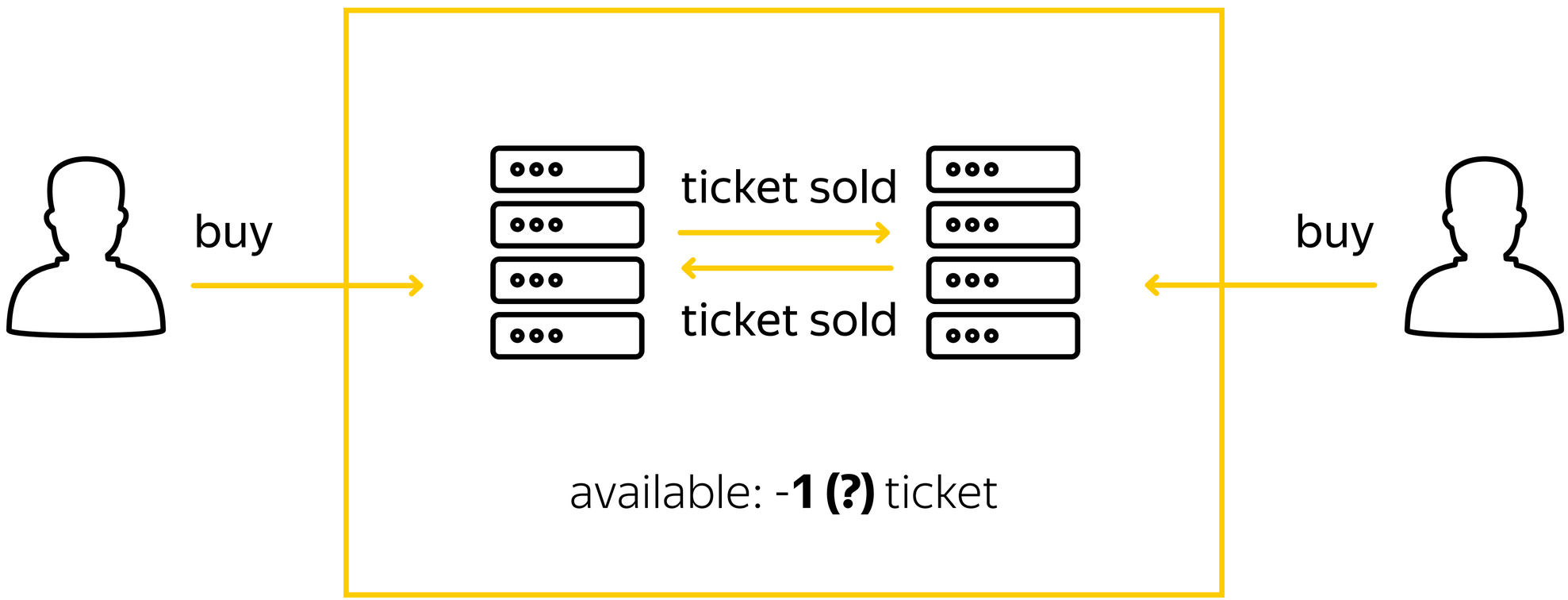
Distributed Booking System



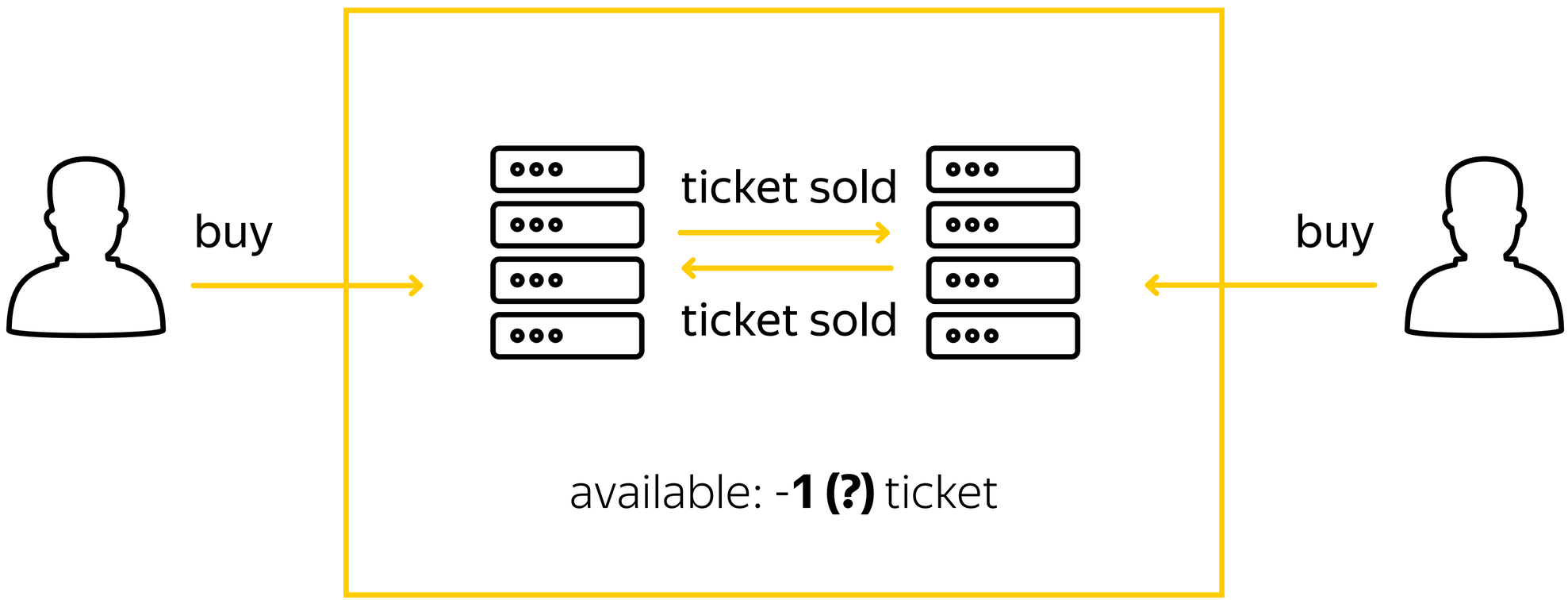
Distributed Booking System



Distributed Booking System

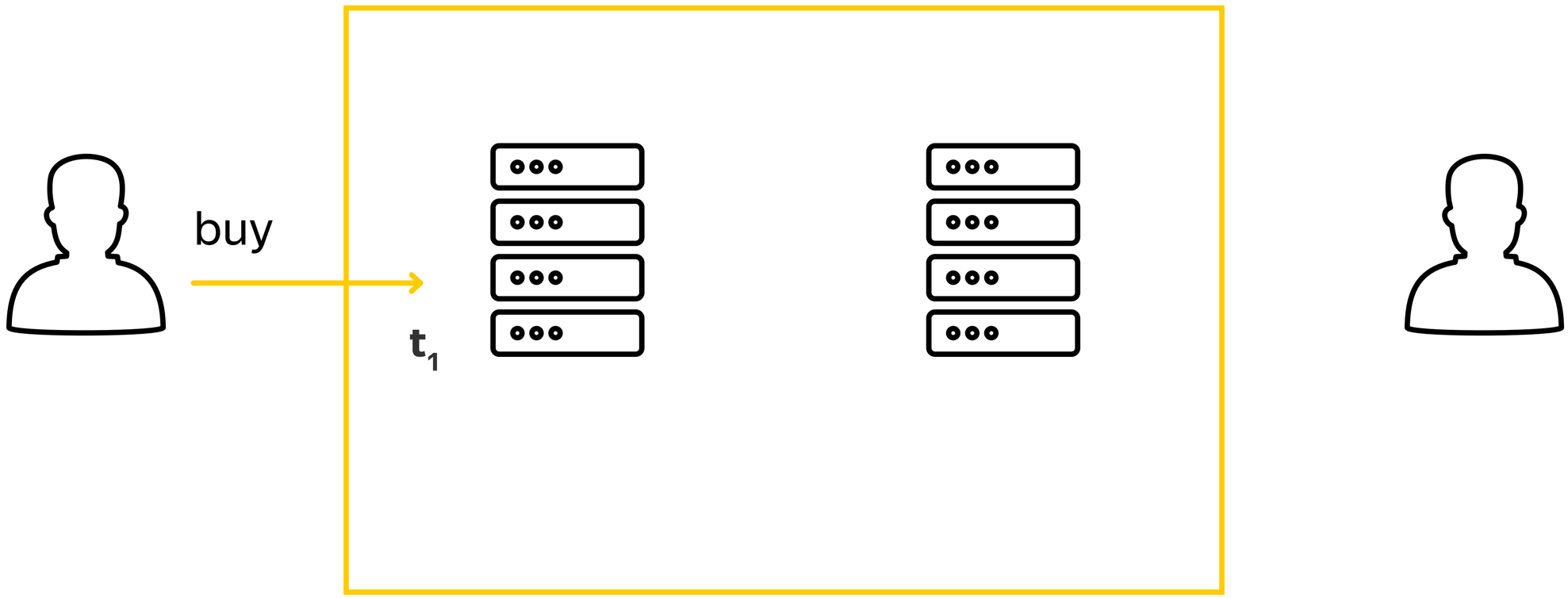


Distributed Booking System

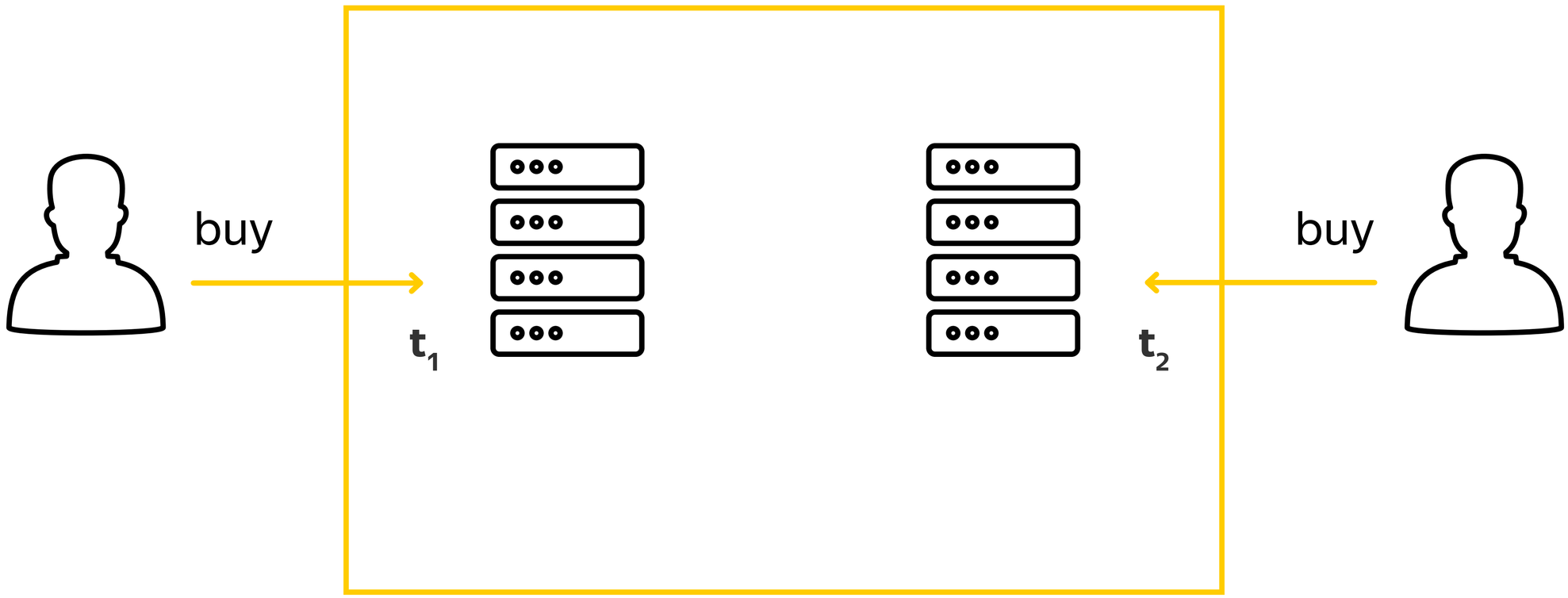


idea: add unix timestamp
to the query time

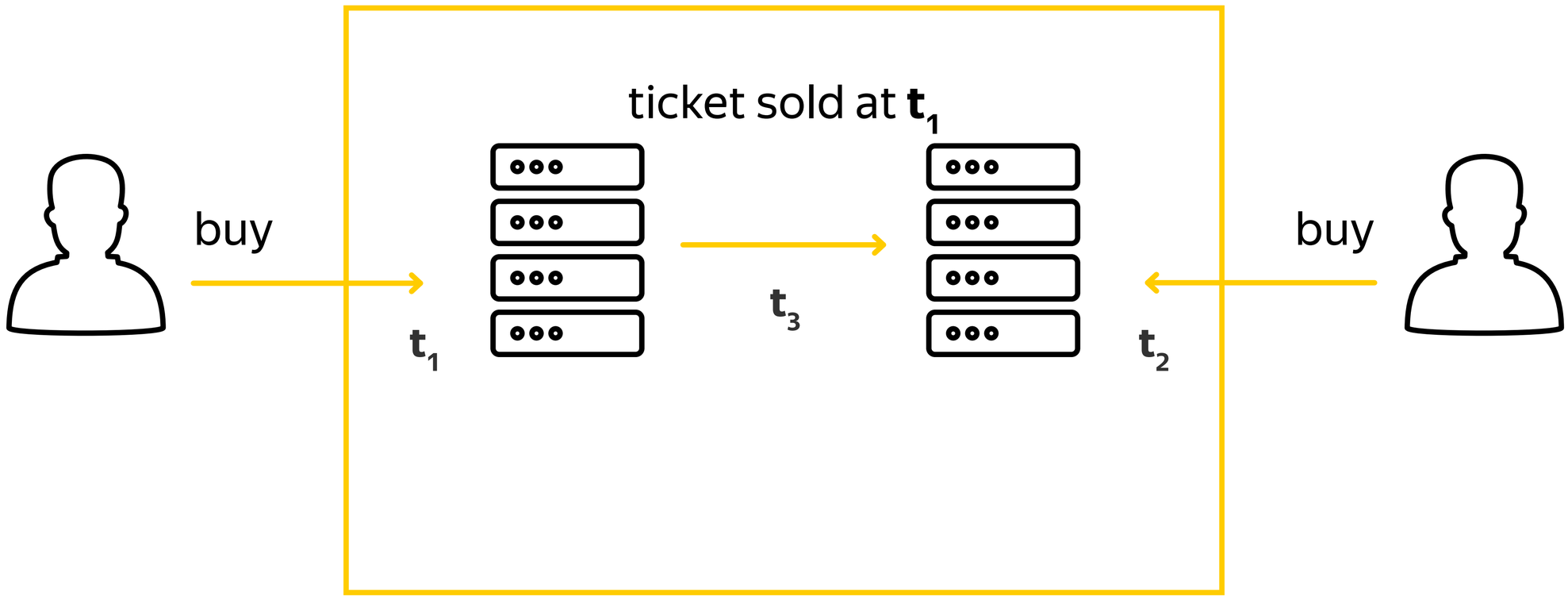
Distributed Booking System



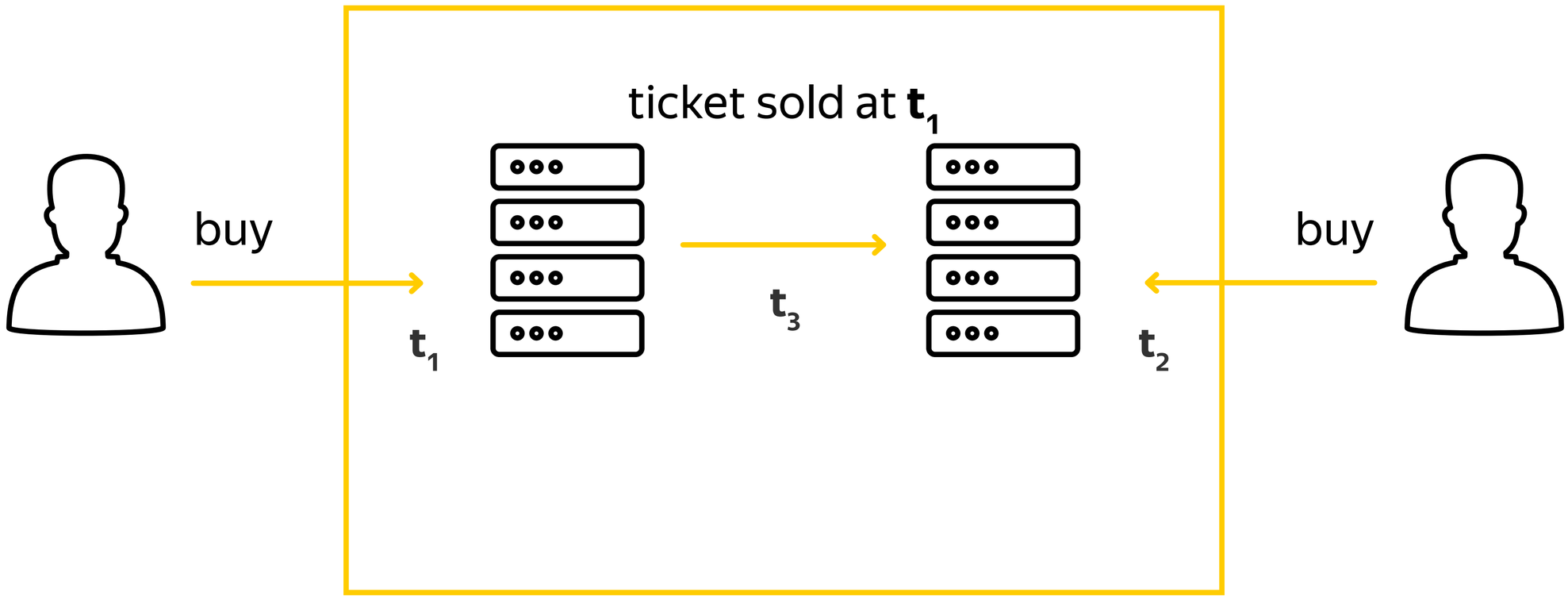
Distributed Booking System



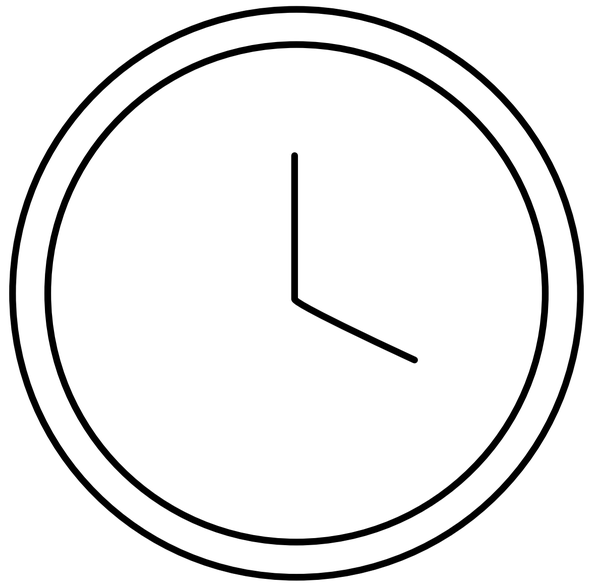
Distributed Booking System



Distributed Booking System

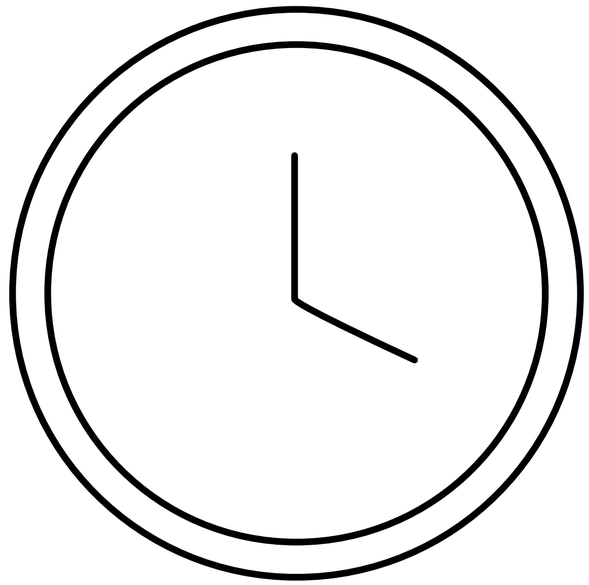


$t_2 < t_3 < t_1$



1 clock skew

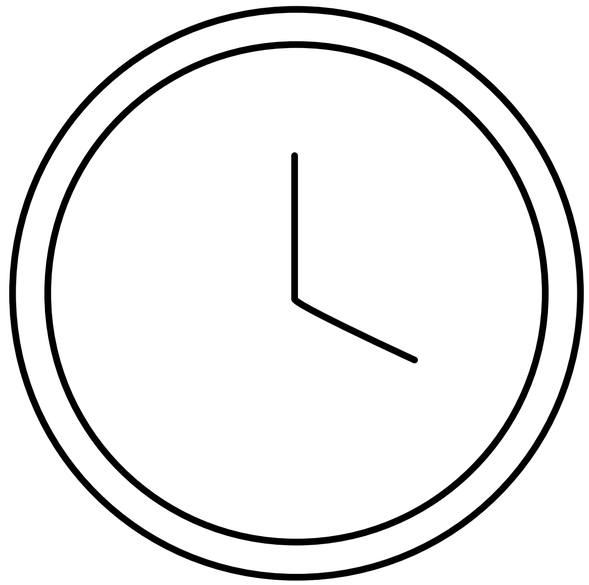
2 clock drift



1 clock skew

2 clock drift

idea: use logical clocks

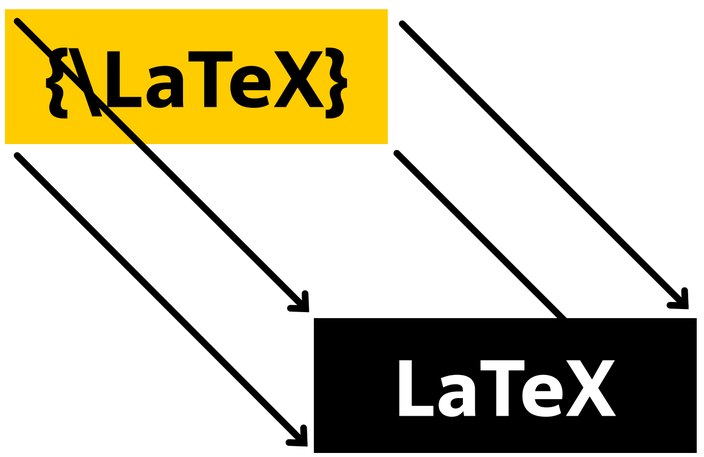


1 clock skew

2 clock drift

Lamport logical clocks

Leslie Lamport



[A] Synchronous Systems

[A] Synchronous Systems

- Every message between nodes is delivered within limited time;

[A] Synchronous Systems

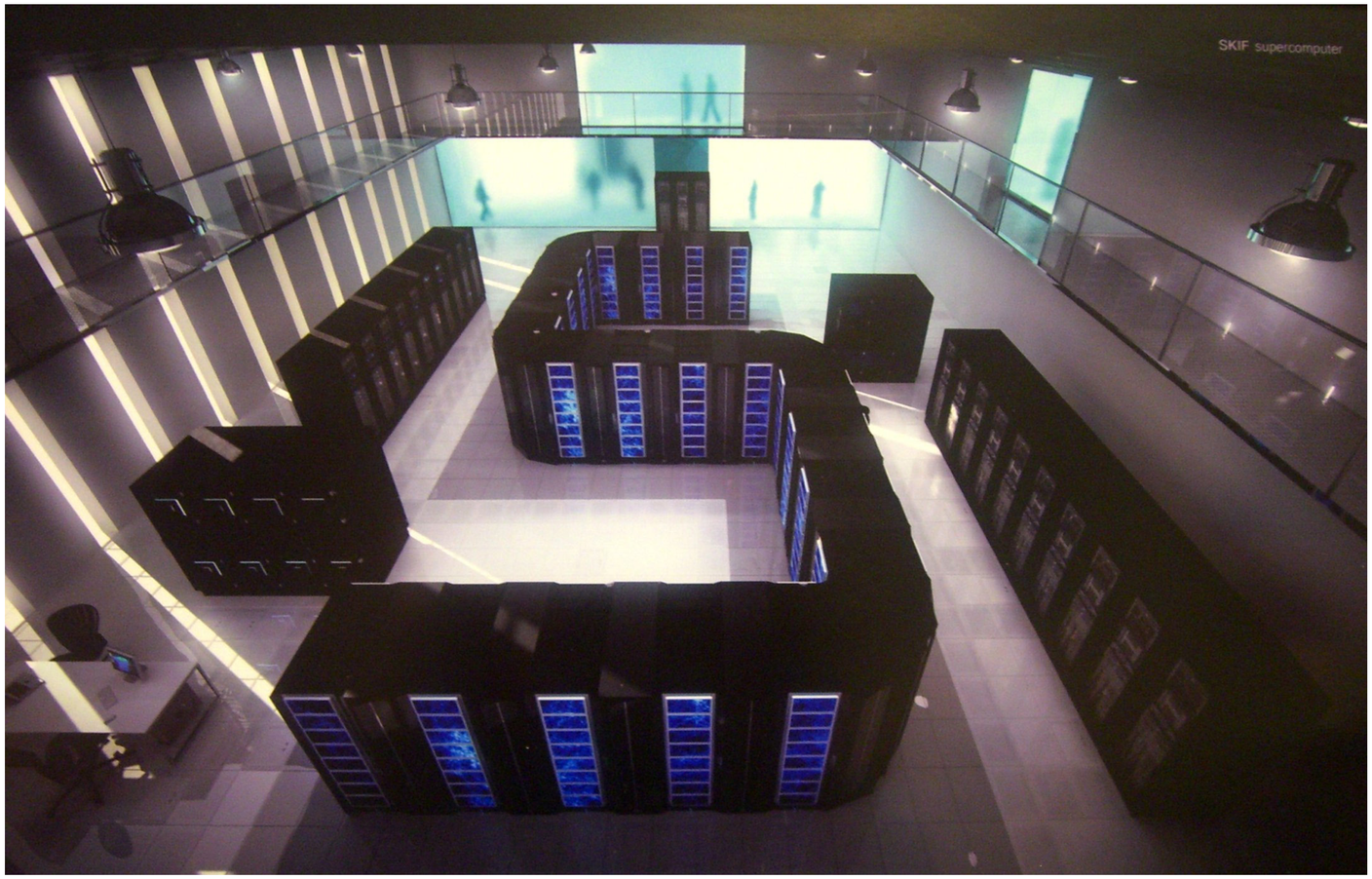
- Every message between nodes is delivered within limited time;
- Clock drift is limited;

[A] Synchronous Systems

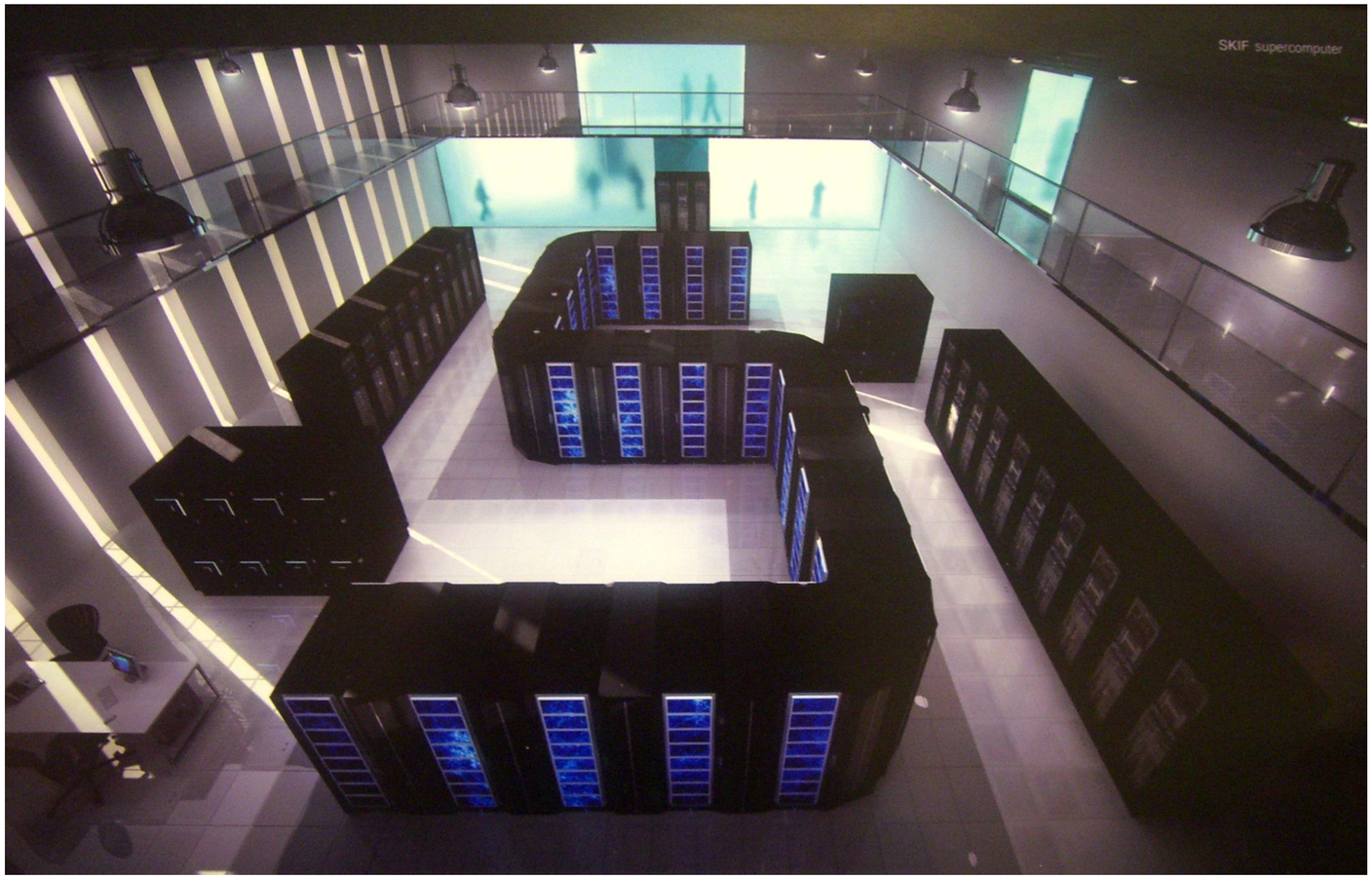
- Every message between nodes is delivered within limited time;
- Clock drift is limited;
- Each instruction execution is also limited.

Fail-Stop + Perfect Link + Synchronous

Fail-Stop + Perfect Link + Synchronous



Fail-Stop + Perfect Link + Synchronous

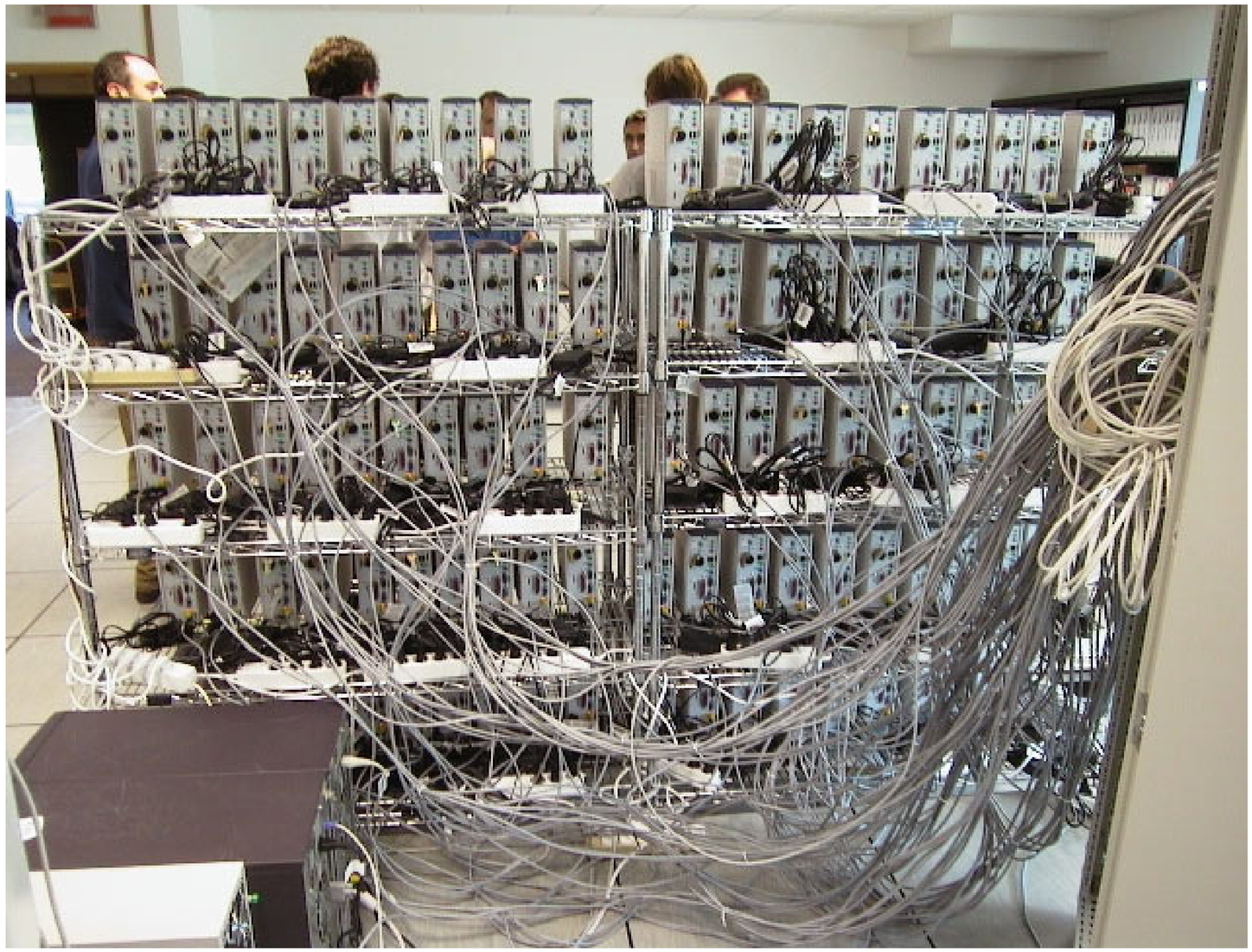


OpenMP

MPI

Fail-Recovery + Fair-Loss Link + Asynchronous

Fail-Recovery + Fair-Loss Link + Asynchronous

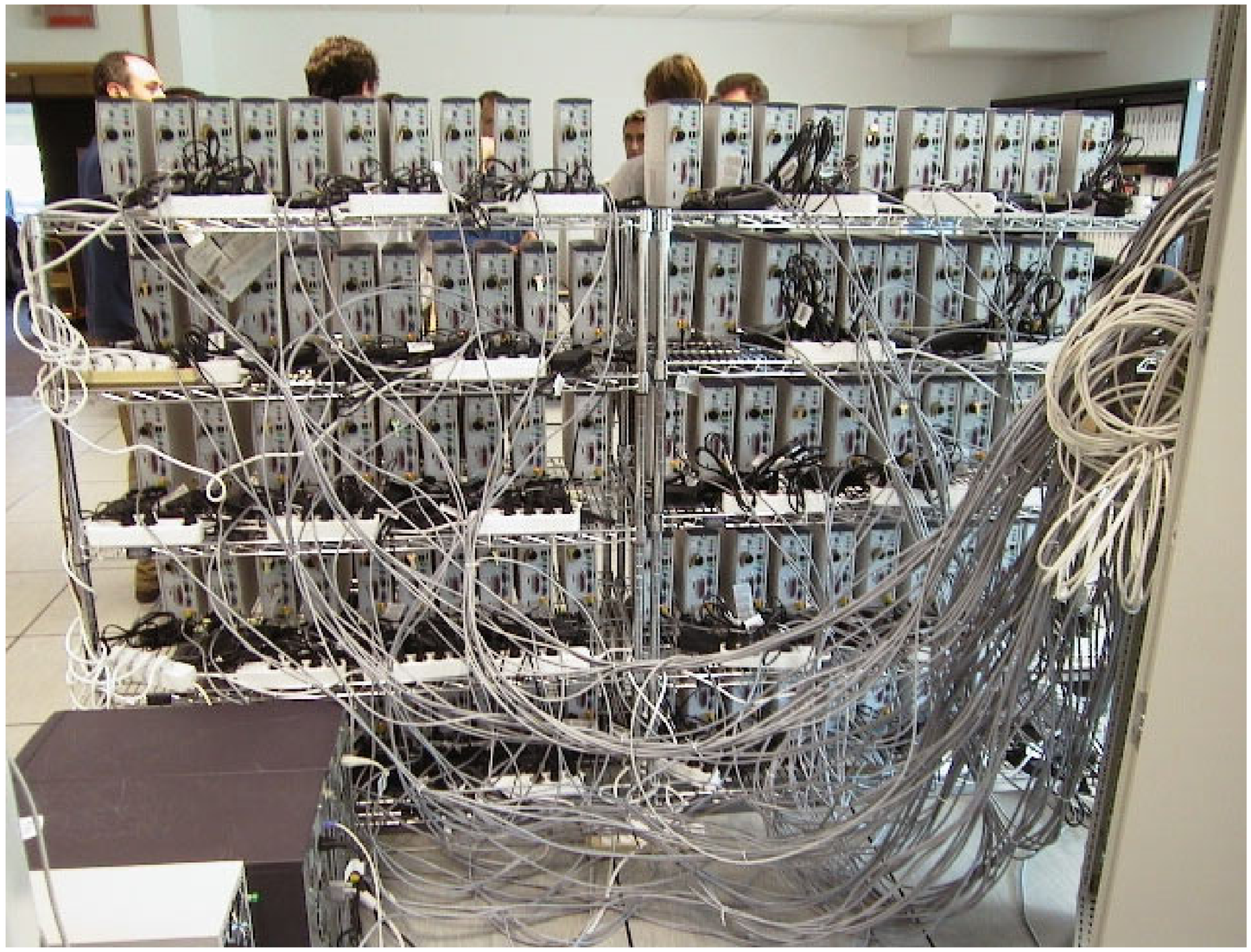


Byzantine-Failure + Byzantine Link
+ Asynchronous

Byzantine-Failure + Byzantine Link + Asynchronous

Grid Computing

Summary



You can provide a thorough description of a distributed system according to its robustness to node failures, link failures and clock synchronization model used in the system.

BigDATAteam