

# Is Two-Phase Commit Evil?

Evan Jones, Sam Madden. MIT CSAIL. {evanj, madden}@csail.mit.edu

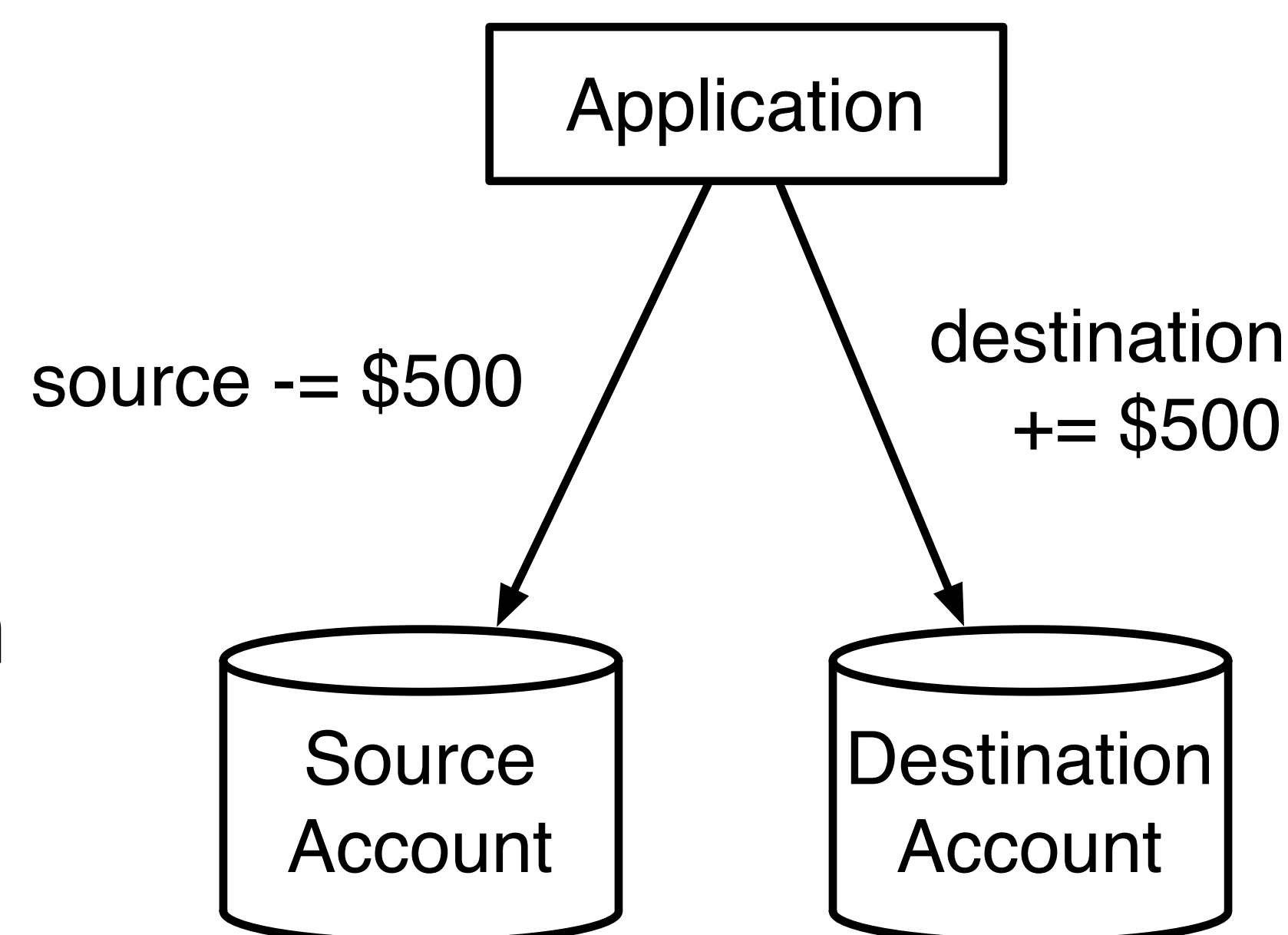
**Conventional Wisdom:** 2PC should never be used: it is slow and error prone.

**Hypothesis:** When applied properly, it is fast and useful.

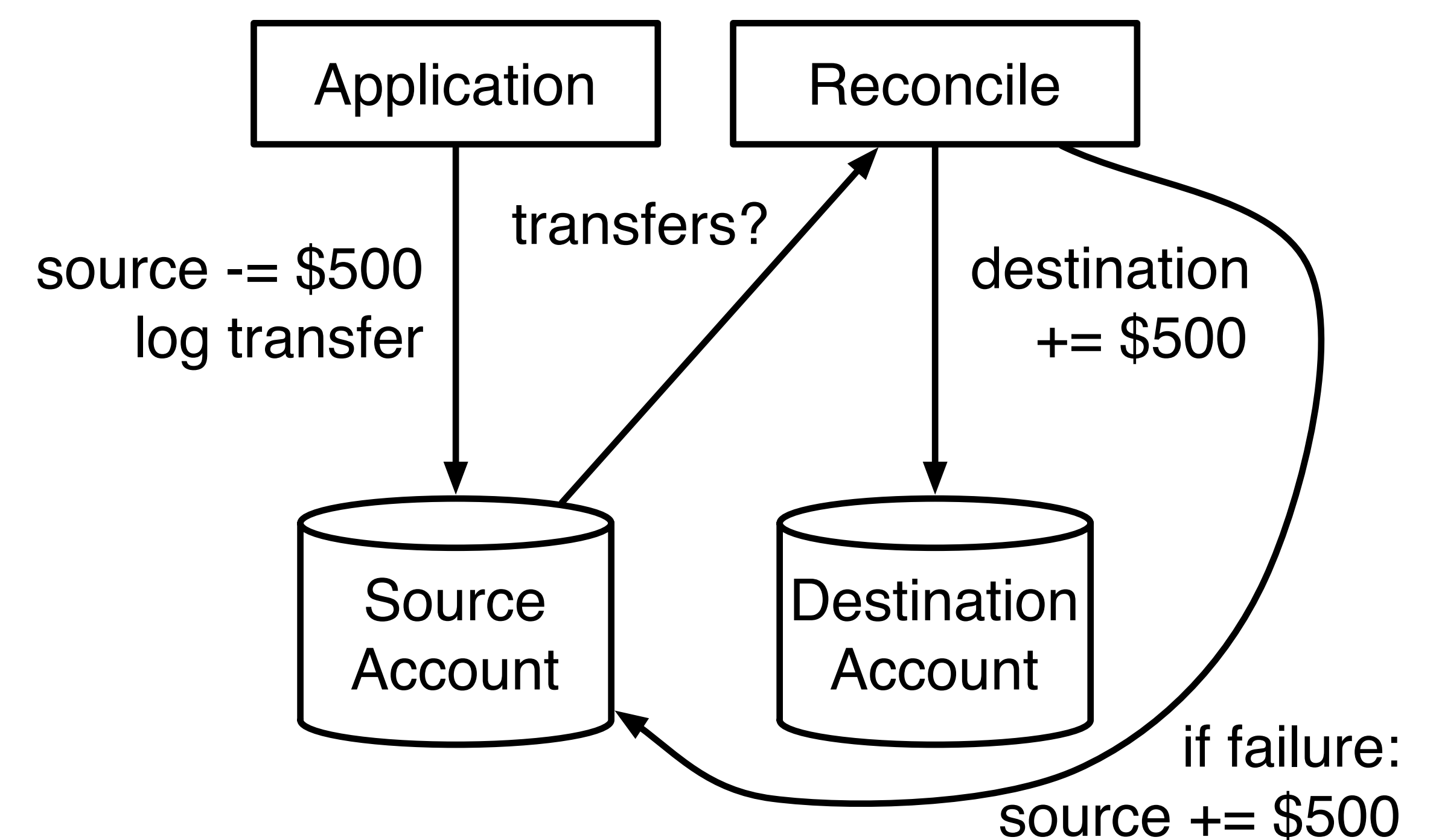
## Pros

- + Clean, understandable semantics
- + Faster development
- + Fewer errors
- + Application unaware of data distribution

### Balance Transfer With 2PC



### Balance Transfer Without 2PC



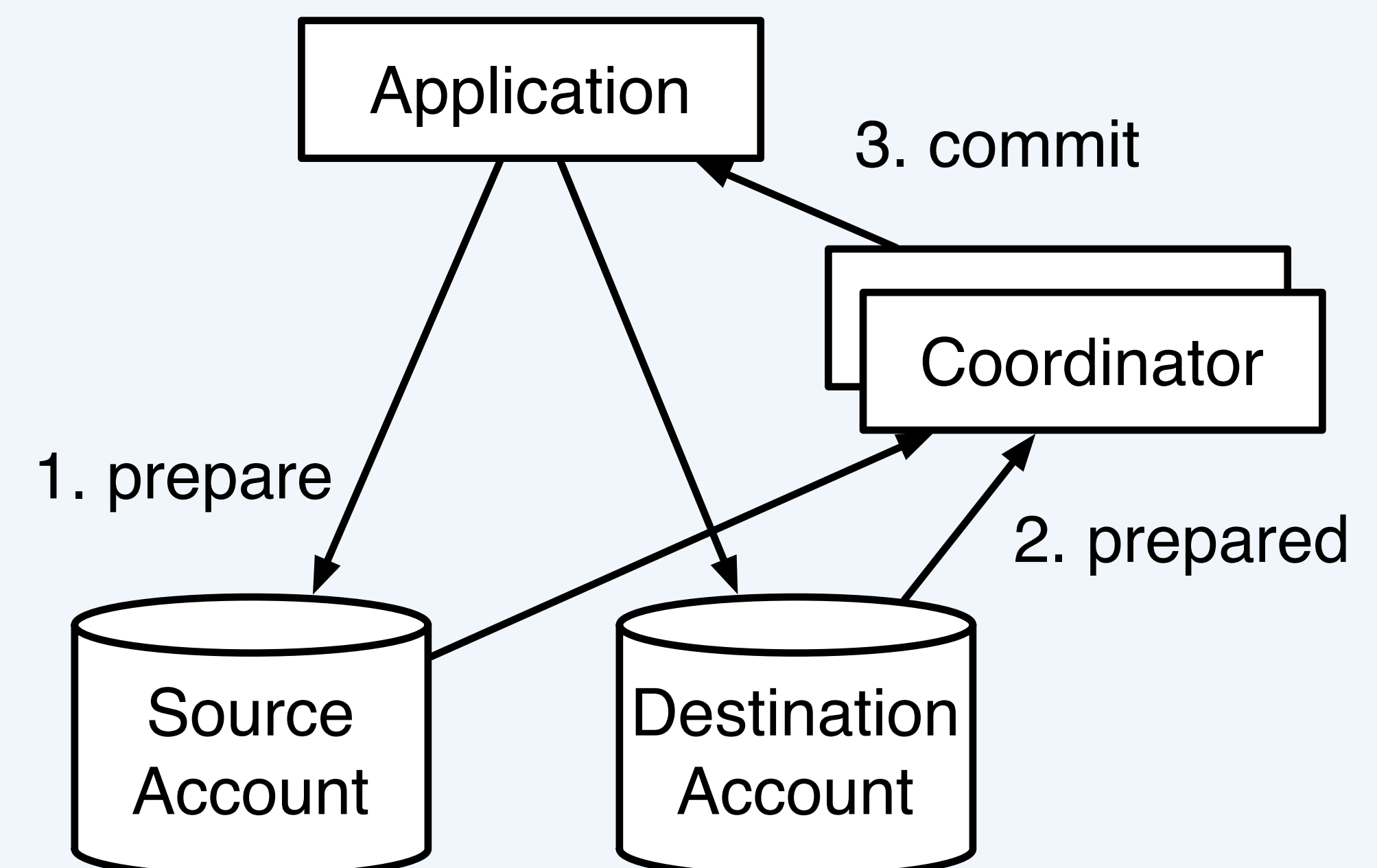
## Cons

- Higher Latency: 1 extra disk write at coordinator  
~8 ms magnetic disk, ~100  $\mu$ s flash
- Lower Throughput: More messages to handle
- Lower availability: Total success or total failure  
Any participant fails, operation fails
- In doubt transactions:  
Transaction is prepared, coordinator unreachable,  
Data items are inaccessible

## Solutions?

### Replicated Coordinator:

- + Can reduce latency (~1 ms for Gigabit Ethernet)
- + Reduces probability of in-doubt txns



### Efficient Protocol:

- + Combine messages where possible

### Asynchronous Transaction Fragments:

- + Queue part of transaction for later
- + Exception handling for partial failure

### Tools/Patterns for Avoiding 2PC:

- + Ordering updates carefully
- + Reducing the "critical section"

**Agree? Disagree?** Tell me why 2PC does (not?) work for you.