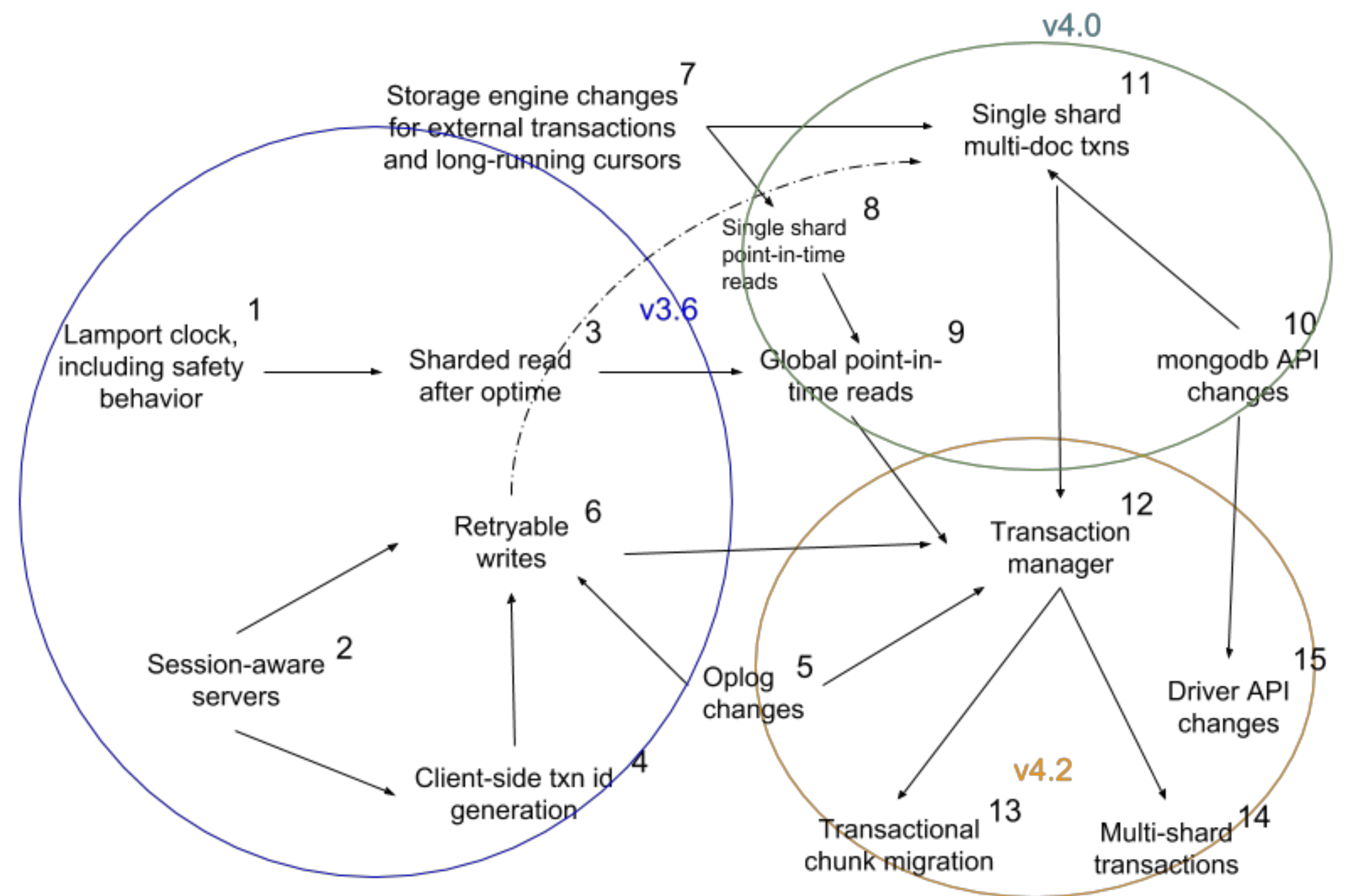# Path to MongoDB Distributed ACID Transactions

Delivering Features Along the Way
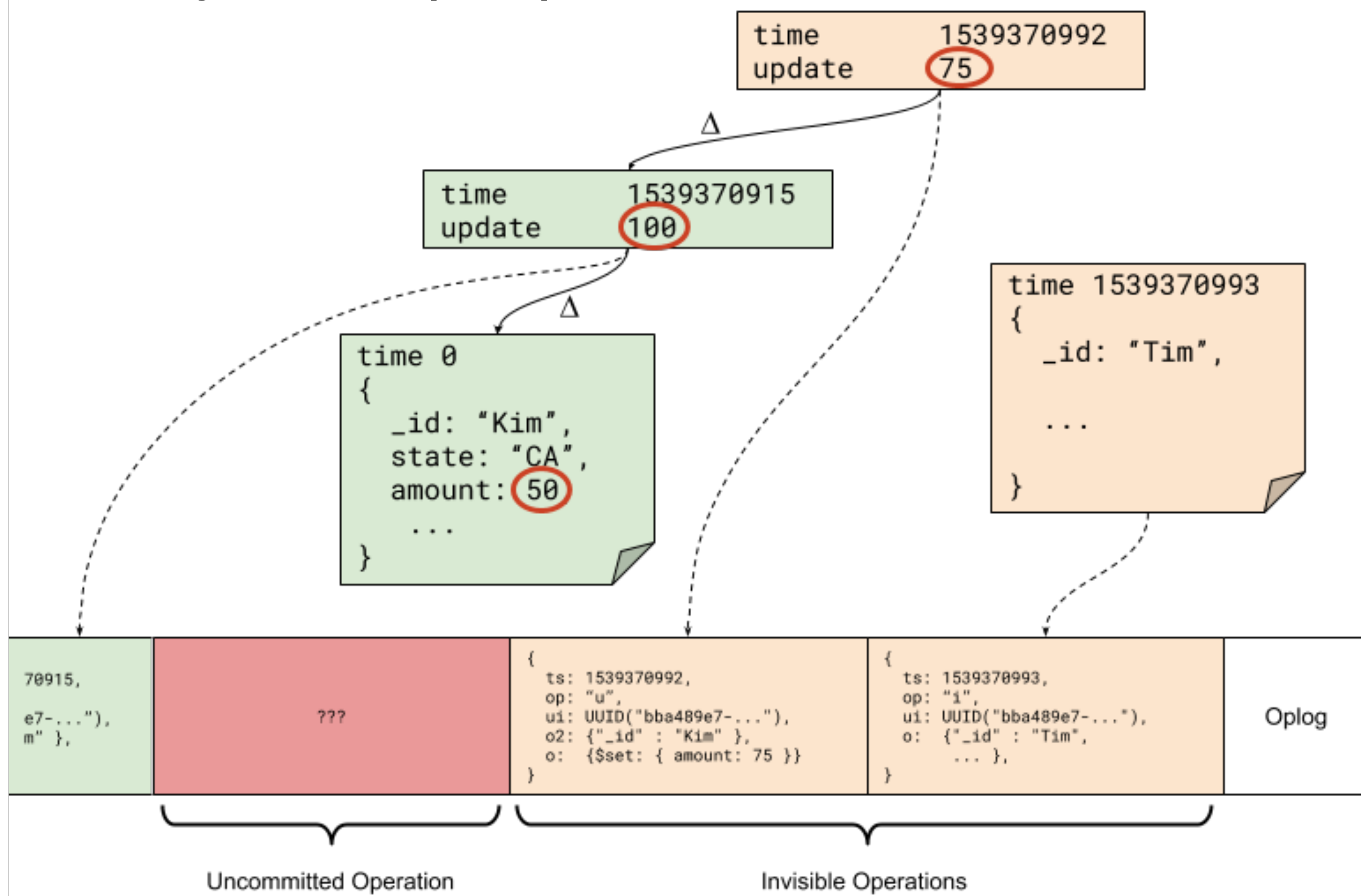
MongoDB 3.0	MongoDB 3.2	MongoDB 3.4	MongoDB 3.6	MongoDB 4.0	MongoDB 4.2
New Storage engine (WiredTiger)	Enhanced Replication protocol: stricter consistency & durability	Shard membership awareness	Consistent secondary reads in sharded clusters	Replica Set Transactions	Sharded Transactions
	Config server manageability improvements	WiredTiger default storage engine	Logical sessions	Snapshot reads	More extensive WT repair
			Retryable Writes	Recoverable Rollback via WT checkpoints	Transaction Manager
			Causal Consistency	Recover to a timestamp	Oplog applier prepare support for transactions
			Cluster-wide logical clock	Sharded Catalog Improvements	
			Storage API changes to use timestamps	Global point in time reads	
			Read concern majority always available		
			UUIDs in sharding		
			Fast in-place updates to large docs in WT		

Legend: ■ Done ■ In Progress

Project Dependencies

In-Memory Time-stamped Update Chains

Uncommitted Operation — Invisible Operations

Causal Consistency

Cluster time ordering: Ty < Tx < T0 < T1 < T2
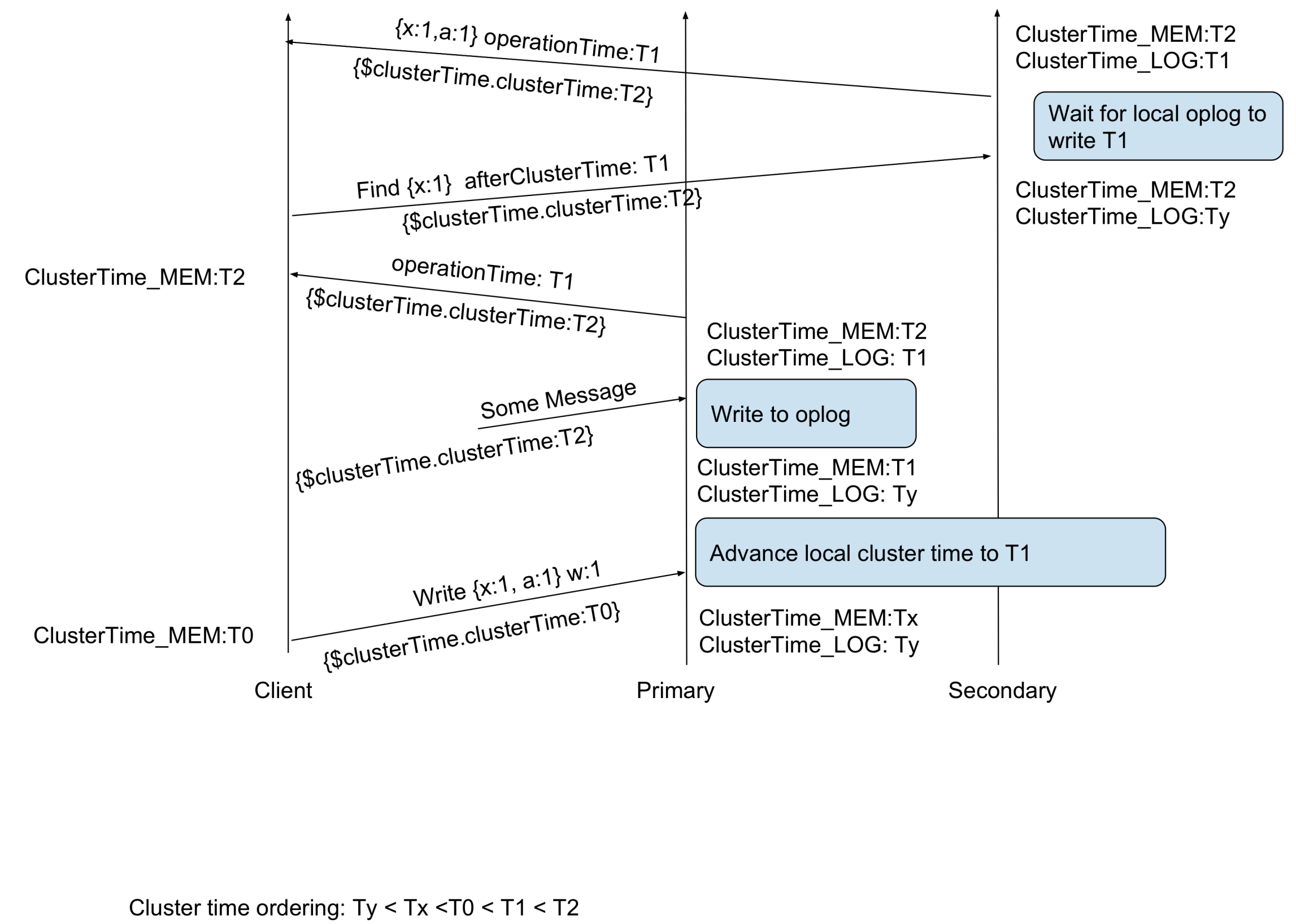
Distributed Commit Protocol

Legend:
■ Can release transaction resources
■ Can forget about transaction decision
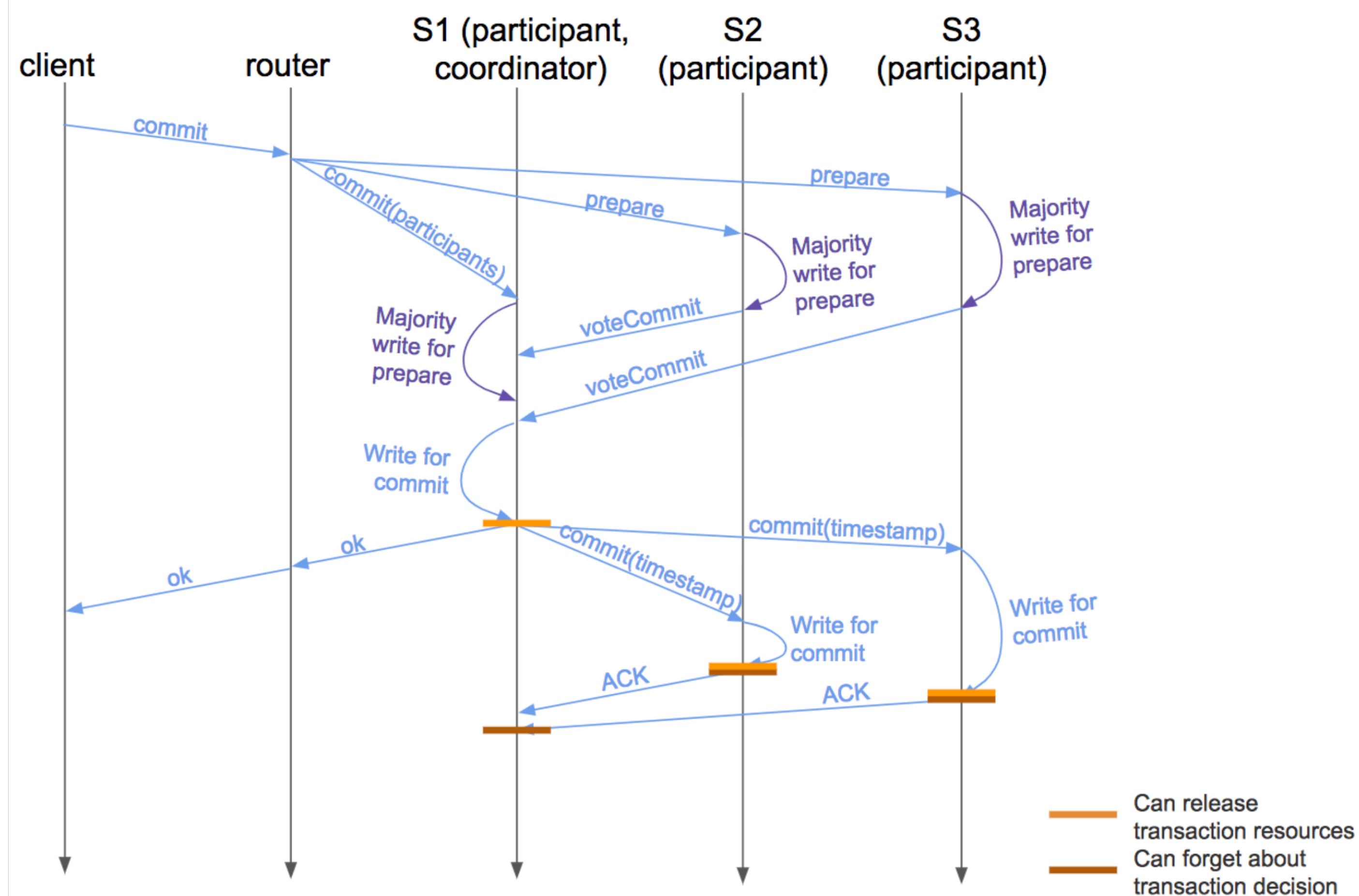
Distributed Transaction


```
update({ _id: "Kim", state: "CA" },
       { $dec: { amount: 25 } });
```

```
update({ _id: "Lia", state: "NY" },
       { $inc: { amount: 25 } });
```

```
prepareTransaction();
```

```
commitTransaction();
```