

# **L20: Replicated state machines**

Nickolai Zeldovich  
6.033 Spring 2012

# **6.033 for Big Screw**

**\$1 donation for each quiz 2 taken (\$220 total)**

**→ donuts for everyone at the final exam**

**Charity: Greater Boston Food Bank**

# Paxos

## **Propose(V):**

choose unique  $N$ , preferably  $N > N_p$

send **Prepare(N)** to all nodes

if **Prepare\_OK(N<sub>a</sub>, V<sub>a</sub>)** from majority:

$V' = V_a$  with highest  $N_a$ , or  $V$  if none

send **Accept(N, V')** to all nodes

if **Accept\_OK(N)** from majority:

send **Decided(V')** to all

Proposer

## **Prepare(N):**

if  $N > N_p$ :

$N_p = N$

reply **Prepare\_OK(N<sub>a</sub>, V<sub>a</sub>)**

Acceptor

## **Accept(N, V):**

if  $N \geq N_p$ :

$N_a = N, V_a = V$

reply **Accept\_OK(N<sub>a</sub>, V<sub>a</sub>)**

# Summary

- Pessimistic replication: single-copy semantics
- Replicated state machines provide single-copy
  - Key issue: agreeing on order of operations
  - Hard case: network partition
- Paxos allows replicas to reach consensus, in presence of machine and network failures