What voting gives us

- You might get system-wide agreement or you might get stuck
- Can’t vote directly on consensus question (i.e., log entry)
- What can we vote on without jeopardizing liveness?
  1. Statements that never get stuck (irrefutable), and
  2. Statements whose hold on consensus question can be broken if stuck (neutralizable)
• A ballot is a pair \( \langle n, x \rangle \)
  - \( n \) – a counter to ensure arbitrarily many ballots exist
  - \( x \) – a candidate output value for the consensus protocol

• Conceptually vote to commit and abort ballots
  - If a quorum votes to commit \( \langle n, x \rangle \) for any \( n \), it is safe to output \( x \)

• Invariant: all committed and stuck ballots must have same \( x \)

• To preserve: can’t vote to commit a ballot before preparing it
  - Prepare \( \langle n, x \rangle \) by aborting all \( \langle n', x' \rangle \) with \( n' \leq n \) and \( x' \neq x \).
  - PREPARED message votes to abort all lower ballots not containing \( x \) (or all lower ballots period if previous is NULL)

• If ballot \( \langle n, x \rangle \) stuck, neutralize by restarting with \( \langle n + 1, x \rangle \)
  - Can prepare \( \langle n + 1, x \rangle \) even if \( \langle n, x \rangle \) is stuck
### Paxos example

#### candidate values

<table>
<thead>
<tr>
<th></th>
<th>a</th>
<th>b</th>
<th>c</th>
<th>d</th>
<th>e</th>
<th>f</th>
<th>g</th>
<th>h</th>
</tr>
</thead>
</table>

#### counter

- 1
- 2
- 3
- 4

0. Initially, all ballots are bivalent

1. Agree that $\langle 1, g \rangle$ is prepared and vote to commit it

2. Lose vote on $\langle 1, g \rangle$; agree $\langle 2, f \rangle$ prepared and vote to commit it

3. $\langle 2, f \rangle$ is stuck, so agree $\langle 3, f \rangle$ prepared and vote to commit it

4. See $T$ votes to commit $\langle 3, f \rangle$ (commit-valent) and externalize $f$
   - At this point nobody cares about $\langle 2, f \rangle$—neutralized

5. Node failure makes $\langle 3, f \rangle$ stuck, prepare and commit $\langle 4, f \rangle$
Paxos example

0. Initially, all ballots are bivalent

1. Agree that $\langle 1, g \rangle$ is prepared and vote to commit it

2. Lose vote on $\langle 1, g \rangle$; agree $\langle 2, f \rangle$ prepared and vote to commit it

3. $\langle 2, f \rangle$ is stuck, so agree $\langle 3, f \rangle$ prepared and vote to commit it

4. See $T$ votes to commit $\langle 3, f \rangle$ (commit-valent) and externalize $f$
   - At this point nobody cares about $\langle 2, f \rangle$—neutralized

5. Node failure makes $\langle 3, f \rangle$ stuck, prepare and commit $\langle 4, f \rangle$
0. Initially, all ballots are bivalent

1. Agree that ⟨1, g⟩ is prepared and vote to commit it

2. Lose vote on ⟨1, g⟩; agree ⟨2, f⟩ prepared and vote to commit it

3. ⟨2, f⟩ is stuck, so agree ⟨3, f⟩ prepared and vote to commit it

4. See T votes to commit ⟨3, f⟩ (commit-valent) and externalize f
   - At this point nobody cares about ⟨2, f⟩—neutralized

5. Node failure makes ⟨3, f⟩ stuck, prepare and commit ⟨4, f⟩
Paxos example

<table>
<thead>
<tr>
<th></th>
<th>a</th>
<th>b</th>
<th>c</th>
<th>d</th>
<th>e</th>
<th>f</th>
<th>g</th>
<th>h</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>x</td>
<td>x</td>
<td>x</td>
<td>x</td>
<td>x</td>
<td>x</td>
<td>x</td>
<td>x</td>
</tr>
<tr>
<td>2</td>
<td>x</td>
<td>x</td>
<td>x</td>
<td>x</td>
<td>x</td>
<td>x</td>
<td>?</td>
<td>x</td>
</tr>
<tr>
<td>3</td>
<td>x</td>
<td>x</td>
<td>x</td>
<td>x</td>
<td>x</td>
<td>x</td>
<td>?</td>
<td>x</td>
</tr>
</tbody>
</table>

0. Initially, all ballots are bivalent

1. Agree that \( \langle 1, g \rangle \) is prepared and vote to commit it

2. Lose vote on \( \langle 1, g \rangle \); agree \( \langle 2, f \rangle \) prepared and vote to commit it

3. \( \langle 2, f \rangle \) is stuck, so agree \( \langle 3, f \rangle \) prepared and vote to commit it

4. See \( T \) votes to commit \( \langle 3, f \rangle \) (commit-valent) and externalize \( f \)
   - At this point nobody cares about \( \langle 2, f \rangle \)—neutralized

5. Node failure makes \( \langle 3, f \rangle \) stuck, prepare and commit \( \langle 4, f \rangle \)
0. Initially, all ballots are bivalent

1. Agree that \( \langle 1, g \rangle \) is prepared and vote to commit it

2. Lose vote on \( \langle 1, g \rangle \); agree \( \langle 2, f \rangle \) prepared and vote to commit it

3. \( \langle 2, f \rangle \) is stuck, so agree \( \langle 3, f \rangle \) prepared and vote to commit it

4. See \( T \) votes to commit \( \langle 3, f \rangle \) (commit-valent) and externalize \( f \)

   - At this point nobody cares about \( \langle 2, f \rangle \)—neutralized

5. Node failure makes \( \langle 3, f \rangle \) stuck, prepare and commit \( \langle 4, f \rangle \)
Paxos example

0. Initially, all ballots are bivalent

1. Agree that \( \langle 1, g \rangle \) is prepared and vote to commit it

2. Lose vote on \( \langle 1, g \rangle \); agree \( \langle 2, f \rangle \) prepared and vote to commit it

3. \( \langle 2, f \rangle \) is stuck, so agree \( \langle 3, f \rangle \) prepared and vote to commit it

4. See \( T \) votes to commit \( \langle 3, f \rangle \) (commit-valent) and externalize \( f \)
   - At this point nobody cares about \( \langle 2, f \rangle \)—neutralized

5. Node failure makes \( \langle 3, f \rangle \) stuck, prepare and commit \( \langle 4, f \rangle \)
Instead of voting on \( \text{op}_1, \ldots \) directly, vote on \( \langle \text{view 1}, \text{op}_1 \rangle, \ldots \)
- Each \( \langle \text{view}, \text{op} \rangle \) selected by a single leader for view, so irrefutable
- E.g., chose leader by round-robin using \( \text{view}\# \mod N \)
- Really, a vote is a promise to include \( \langle \text{view 1}, \text{op}_1 \rangle \) in future views

What if votes on \( \text{op}_4 \) and \( \text{op}_5 \) are stuck (e.g., leader fails)?
- Neutralize by agreeing view 1 had only 3 meaningful operations
- Vote to form view 2 that immediately follows \( \langle \text{view 1}, \text{op}_3 \rangle \)

Failed to form view 2 (e.g., a node wants \( \langle \text{view 1}, \text{op}_4 \rangle \))? 
- Just go on to form view 3 after \( \langle \text{view 1}, \text{op}_4 \rangle \)
Instead of voting on $\text{op}_1, \ldots$ directly, vote on $\langle \text{view 1}, \text{op}_1 \rangle, \ldots$

- Each $\langle \text{view}, \text{op} \rangle$ selected by a single leader for view, so irrefutable
- E.g., chose leader by round-robin using $\text{view}\# \mod N$
- Really, a vote is a promise to include $\langle \text{view 1}, \text{op}_1 \rangle$ in future views

What if votes on $\text{op}_4$ and $\text{op}_5$ are stuck (e.g., leader fails)?

- Neutralize by agreeing view 1 had only 3 meaningful operations
- Vote to form view 2 that immediately follows $\langle \text{view 1}, \text{op}_3 \rangle$

Failed to form view 2 (e.g., a node wants $\langle \text{view 1}, \text{op}_4 \rangle$)?

- Just go on to form view 3 after $\langle \text{view 1}, \text{op}_4 \rangle$
Instead of voting on $op_1$, ... directly, vote on $\langle view\ 1,\ op_1 \rangle$, ...
- Each $\langle view,\ op \rangle$ selected by a single leader for view, so irrefutable
- E.g., chose leader by round-robin using $view\# \mod N$
- Really, a vote is a promise to include $\langle view\ 1,\ op_1 \rangle$ in future views

What if votes on $op_4$ and $op_5$ are stuck (e.g., leader fails)?
- Neutralize by agreeing view 1 had only 3 meaningful operations
- Vote to form view 2 that immediately follows $\langle view\ 1,\ op_3 \rangle$

Failed to form view 2 (e.g., a node wants $\langle view\ 1,\ op_4 \rangle$)?
- Just go on to form view 3 after $\langle view\ 1,\ op_4 \rangle$