Apps have finger prints fo changes

# Algorithm

## 1) Call for Meeting

A Node calls for meeting.

Freeze changes for App that meeting is being called for. Locking the App State.

Rollcall with last meeting tag sent to everyone.

## 2) Sub Committee Priority Calculation

Determine by

1. Number of Nodes in Sub Committee : Majority Minority
2. Sum of Priority of Nodes in Sub Committee : Quorum Priority Comparison (REMOVED)
3. Greatest Priority of Nodes in Sub Committee : Last Ditch Effort

## 3) Resolution Phase

Allow Meeting Dropouts.

Priority Number determines order of merges.

### 3.1) Sub Committee Meetings

Send "Last Meeting to Top" to sub-committee members. Priority is based on each nodes priority

Sub-Committee Leader checks for errors.

### 3.2) Committee Meeting

Sub Committee Leaders send "Last Consensus to Last Meeting" to everyone. Priority is based on group priority (see 2.2 and 2.3). Priority Sub-Committee Leader handles error.

Sub-committee leaders send "Last Meeting to Top" to everyone.

## 4) Tagging and Exit

Tag Last Meeting

Tag Last Consensus

Confirm Last Meeting and Last Consensus with everyone.

If Error, rollback meeting changes.

Can leave when you receive confirmation from everyone.

# Cases and Extra Ideas

## Each App has its own history hash

Freeze application specific change queue. Each application handles critical merge conflicts. Send notification to user about that and handle it in that Apps UI

Each App handle

## Divided Network

Node A and B are separated from Node C and D

Nodes need to identify that nodes are divided

Make sure to not throw away anything bc on reconnect it might become relavent

## Divided Network – Add/Remove Node

## Out of Date Hash

## Squashing for Checkpoints

Squash after T time and full meeting. Merge first and second into first and renumber thus keeping X checkpoints.

# OLD

## 2) Resolution

=======

Priority Minority of each group sends the state from Last Consensus to Last Meeting to all in the Majority Minority.

Majority sends to all minorities what the last meeting should be.

=========

Minority Nodes send to chief Nodes their diff from last meeting. Chief Nodes handle updating of minority nodes.

Even division of winner nodes updates the out-of-date nodes (Sort [Pidgeon hole] by priority of nodes and then mod). Slow corruption?

* A updates C and E
* B updates D

