

# Application State Sync



draft-ralston-mimi-protocol

# Group State Agreement is Powerful

MLS confirms that the group agrees on the state of the group

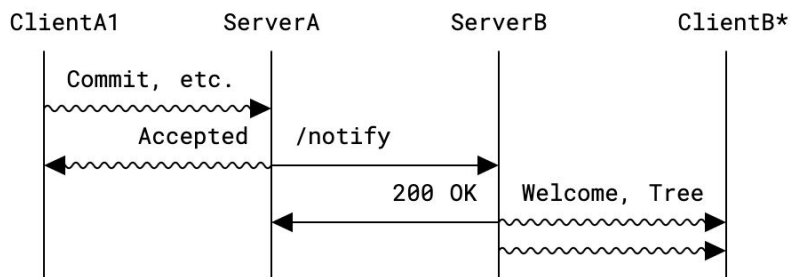
Applications frequently want all clients to agree on application state

Let's import application state into the state of the group!

# Envisioned Use in MIMI

Participant list is included in the MLS key schedule via AppState extension

Changes to the PList are communicated in AppSync proposals



The Commit includes regular *Adds* and an AppSync Proposal updates app state

```
ClientA1: Prepare Commit over AppSync(+Bob), Add*
ClientA1->>ServerA: [[ Commit, Welcome, GroupInfo?, RatchetTree? ]]
ServerA: Verify that AppSync, Adds are allowed by policy
ServerA: Identifies Welcome domains based on KP hash in Welcome
ServerA->>ServerB: POST /notify/a.example/r/clubhouse Intro{ Welcome, RatchetTree? }
ServerB: Recognizes that Welcome is adding Bob to room clubhouse
ServerB->>ClientB*: [[ Welcome, RatchetTree? ]]
```

# AppSync Extension + Proposal

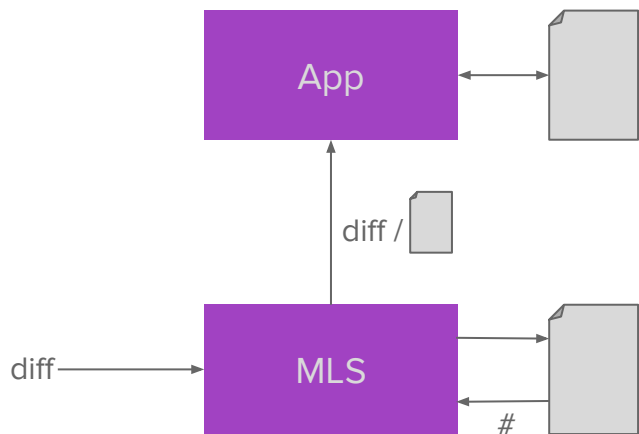
```
struct {
  uint32 applicationId;
  StateType stateType;
  select (stateType) {
    case irreducible:
      OpaqueElement state;
    case map:
      OpaqueMapElement mapEntries<V>;
    case unorderedList:
      OpaqueElement unorderedEntries<V>;
    case orderedArray:
      OpaqueElement orderedEntries<V>;
  };
} ApplicationState;
```

**GroupContext Extension**

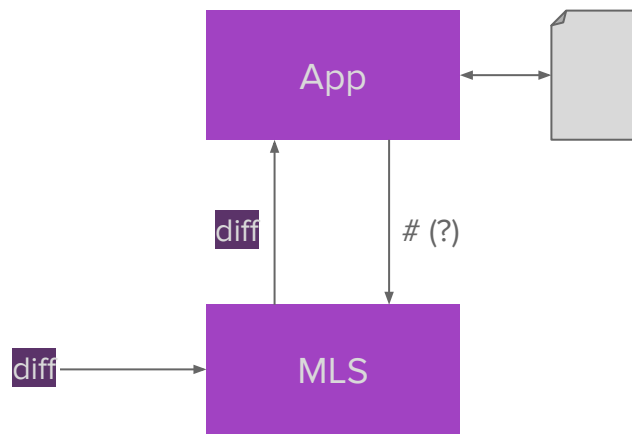
```
struct {
  uint32 applicationId;
  StateType stateType;
  select (stateType) {
    case irreducible:
      OpaqueElement newState;
    case map:
      OpaqueElement removedKeys<V>;
      OpaqueMapElement newOrUpdatedElements<V>;
    case unorderedList:
      uint32 removedIndices<V>;
      OpaqueElement addedEntries<V>;
    case orderedArray:
      ElementWithIndex replacedElements<V>;
      uint32 removedIndices<V>;
      ElementWithIndex insertedElements<V>;
      OpaqueElement appendedEntries<V>;
  };
} AppSync;
```

**Proposal**

# APIs and Design Choices



State **intelligible** to MLS  
(current)



State **opaque** to MLS

# Current Status

Described in draft-ralston-mimi-protocol

If MIMI adopts that draft, it will have a dependency on MLS WG doing something like this