Server

```
receive msg {
  case ClientReq:
    this.Data = message.Val;
    // Replicate data to all nodes
    foreach (sn in this.Nodes)
      sn.send(ReplReq, this.Data);
  case Sync:
    Node node = message.Id;
    Log log = message.Log;
    doSync(node, log);
}
```

```
doSync (Node sn, Log log) {
    // If the storage log is not
    // up-to-date, replicate
    if (!isUpToDate(log))
      sn.send(ReplReq, this.Data);
    else {
      this.NumReplicas++;
      if (this.NumReplicas == 3)
        this.Client.send(Ack);
    }
}
```

Storage Node

```
receive msg {
  case ReplReq:
    // Store received data
    store(message.Val);
  case Timeout:
    // Send server the log
    // upon timeout
    this.Server.send(Sync,
      this.Id, this.Log);
}
```

Client

```
while (hasNextRequest()) {
  this.Server.send(ClientReq,
    this.DataToReplicate);
  receive(Ack); // Wait for ack
}
```

Timer

```
// Send timeout to node when
// countdown reaches 0
if (this.Countdown == 0)
  this.SN.send(Timeout);
```