# CPRE 492 WEEKLY REPORT 15

*Project Molecule*

18 – 24 January 2017

May1739
may1739@iastate.edu

Dr. Arun Somani

Ryan Wade – Team Leader

Nathan Volkert – Communications Lead

Daniel Griffen – Key Concept Holder

Alex Berns – Webmaster & Scribe

## 1 CONTENTS

# 2 WEEKLY SUMMARY

This week we worked on fixing the issues discussed with our advisor about over complication of the synchronization algorithm. We learned we were trying to solve byzantine faults which is impossible. We fixed the over complication by stating that the user has more power and if the network is divided certain activities cannot be performed by the nodes.

We also considered more use cases during our advisor meeting this week and decided that we will focus on the more common issues with the system. Divided network is an uncommon case so we will minimize the amount of work on that in order to find more solutions to common problems. These include all nodes disconnect or part of network goes down (but does not operate); so only attempting to sync out-of-date nodes.

# 3 PAST WEEK ACCOMPLISHMENTS

All Members:

- Discussed UI communication design
- Discussed limits of fault tolerance

Ryan Wade:

- Brainstormed UI Architecture
- Worked on UI Manager Design

Nathan Volkert:

- Worked on synchronization - researched algorithms
- Started work on a console for reading packaging and getting information for the atomic layer

Daniel Griffen:

- Read documentation on Tokio framework

Alex Berns:

- Did research on Mocha Testing and preparing for creation of Mockup form builder

# 4 INDIVIDUAL CONTRIBUTIONS

| NAME | Hours | Semester Total | Cumulative |
|---|---|---|---|
| Ryan Wade | 10 | 21 | 141 |
| Nathan Volkert | 7 | 14 | 116 |
| Daniel Griffen | 7 | 21 | 155 |
| Alex Berns | 11 | 19 | 118 |

# 5 COMMENTS AND EXTENDED DISCUSSION

## 5.1 SYNCHRONIZATION

The following activities require full consensus:

- Adding, updating, removing apps
- Adding, updating, removing nodes
- Modifying permissions

If nodes are not present and full consensus is not reached the following exception applies

- All not present may be simultaneously removed.

## 5.2 All state changes are signed by both the particle and the associated atom from which they originate.

## 5.3 PARTICLE PACKAGING

Particle Package Format:

Manifest : TOML
    Name (Human Readable)
    App Signiture (Unique public signature)
    List Actions Supported (interface)
    Permissions Needed
Resources
Binary
Icon

Node Package Storage. Can store full and partial

App List: TOML
    <App Sig>
        particle package
    <App Sig>
        particle package
    <App Sig>
        Particle Manifest (no package)

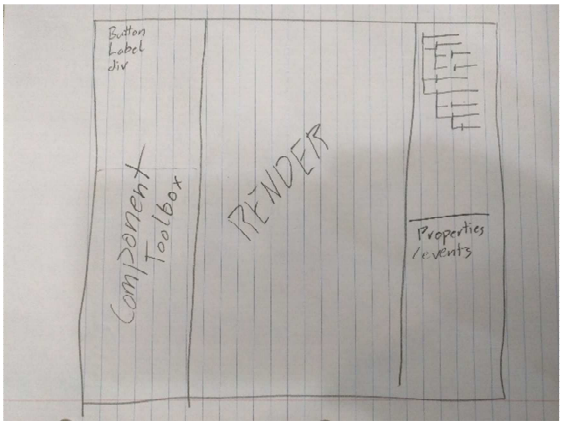
## 5.4 UI

How to handle new or reconnecting UI Client

1. Client Requests connection
2. Connection Manager sends client javascript and static assets (“binary”)
3. Connection Manager identifies client, if new then messages START app else resumes LAST app run by client
4. app responds with UI stage

5. Connection Manager messages client with stage change
6. Client renders stage

UI Builder




# 6 PLAN FOR COMING WEEK

Alex: Form Builder

Ryan: Working Client Manager implementation in Rust

Dan: Bidirectional stream... rooms, finish simple message api

Nat: Console App for reading app manifest info

# 7 SUMMARY OF WEEKLY ADVISOR MEETING

Discuss last week's work on information propagation. By only allowing major changes during full consensus. Upon reconnect of separated networks, attempt to find last point of consensus. Don't worry about bad actors since other part of system (permissions) handles that.

Main hub goes down – Everyone disconnected

Part of house loses power – Only part of network disconnects

Two Router Case: Consider later

We should focus on more common errors