# Last Week’s Accomplishments

Nat: Finished Permission Integration, Worked on design doc

Dan: Auto switch nodes to anonymous protocol to enable adding new devices easily.

Ryan: Skeleton of permissions layer created, launching particles

Alex: Worked on Design Doc and Project Plan and final doc

# Notes

No Advisor meeting next week. We will still meet. We need a poster for next week by Monday so we can get feedback from Somani by Monday.

Testing, give to Nat and Alex so its tested by a user, not a developer.

Non-functional requirement is that failure of use moves on, not crash

Find out final report format

# Next Week Goals

Alex + Nat: Poster Done Monday Night, send to Somani

Ryan + Dan: Particles working, testing adding multiple atoms

Dan: Sync start working, finish by following week

# Distribute Work Load

1. Run and config binary
   1. Two atoms working (maybe three)
2. Synchronization
   1. Version History of state: stack pushed to redis
      1. Version is key, high level key under that for Group and UUID
   2. Group Sync

# Post Advisor

## Device

Three IDs unique to each device.

ID of device

ID of atom

ID of network layer

## State

State is a key value pair. Value and key are strings. Value shall be desterilized into actual data. It’s a tree like structure stored as a flat view. AKA, file directory.

Format is:

Self: Defines what is in atom need to know things

Group: defines things like what particles are running on what devices. Combination of permissions and service discover.

Every Particle:

Every uuid particle, store shared state, name spaced to each particle (each part can only access to their own state)

## REdis

Setting up Redis

Configure Atom + other: \_\_\_\_.toml (address to the DB) \_\_\_\_.toml (init config) init\_config -> create / overwrite redis

Run Atom + other: \_\_\_.toml (address to the DB) -> read / write to redis

The initial config will say there is a command line god particle at start, can be deleted at a later point. This helps with getting initial communication to atom.

## Running Device

1. Creates an atom
2. Starts networking layer
3. Launches particles as defined from redis state, who’s address is defined in arguments