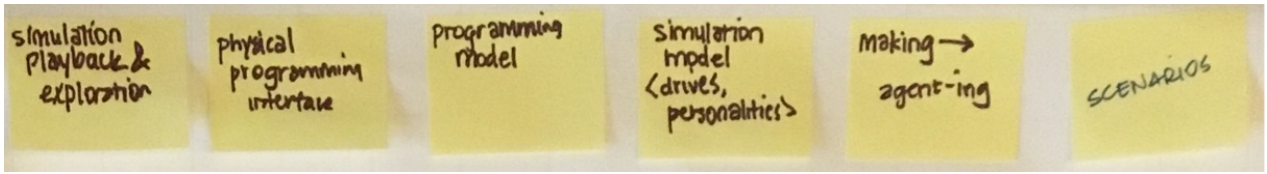
Journal

Monday, Feb 22: Jam

- After the jam on Monday, Nagle and I have been wrapping our brains around fleshing something out for our next Parts and Crafts activity (Mar 20 - 24).
  We started listing some realms of city building + simulation that we could dig deep into.



Tuesday, Feb 23: Boardgames

- After playing “Bay Area Regional Planner” a few weeks ago, we were inspired by the idea that part of the Parts & Crafts activity could be in the form of a board game. Nagle suggested we do some board game research, so last Tues, we played board games with Marko. Some rudimentary analyses:
  - Rules in the cards vs rules on the board
  - Turn state visible by virtue of the pieces and placement
  - Some games are about maintaining state in your head and about how effectively you can do it (Kingdom builder). The cards that provide rules are somewhat declarative.
  - 3d pieces are moving agents; flat tiles describe environmental state
  - Games have “win conditions” and “end conditions,” and they are not always the same

  We want to create a board game that allows the kids to be able to question the rules of the game and make their own.





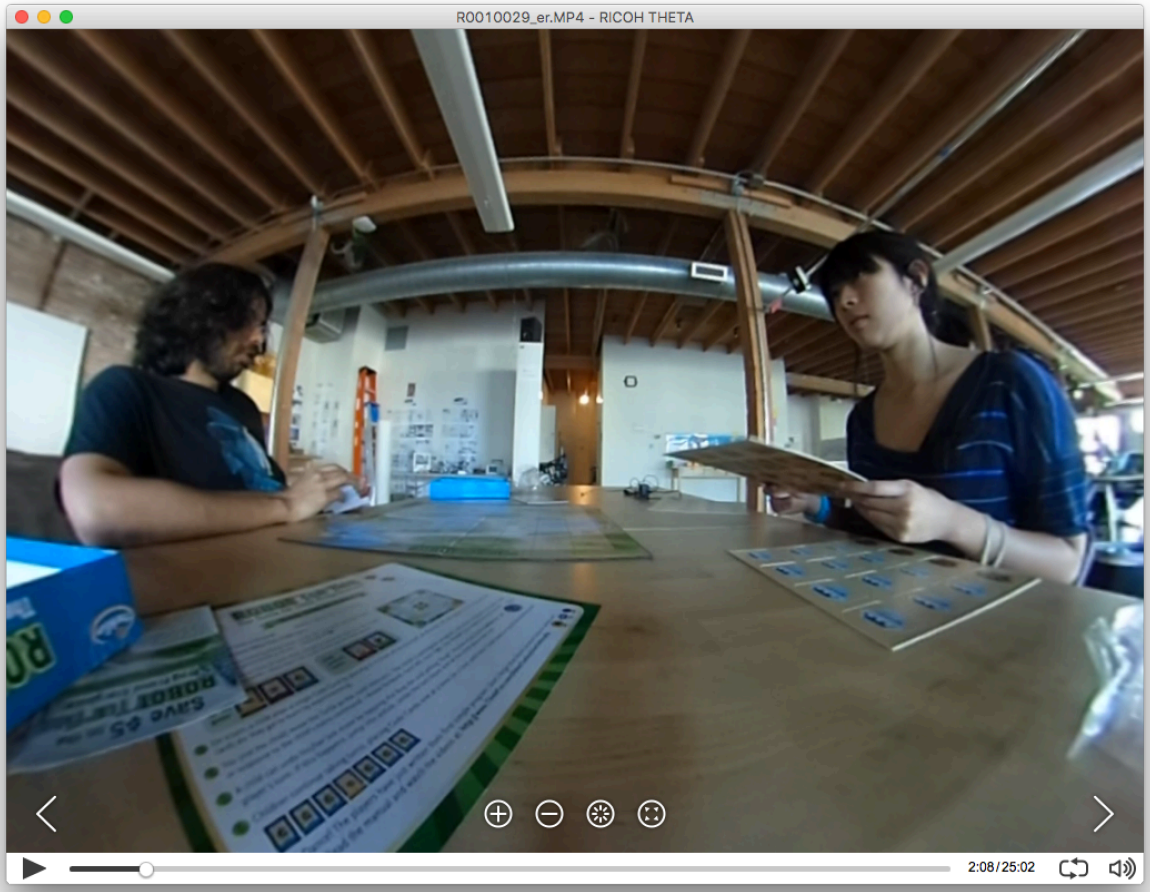




We played Agricola (or the baby version of it) all the way through. This game seemed more open to changing the rules, since the pieces are very simple and nondescriptive, unlike some other pieces. I did appreciate the combination figurine + text on the monster pieces, though.

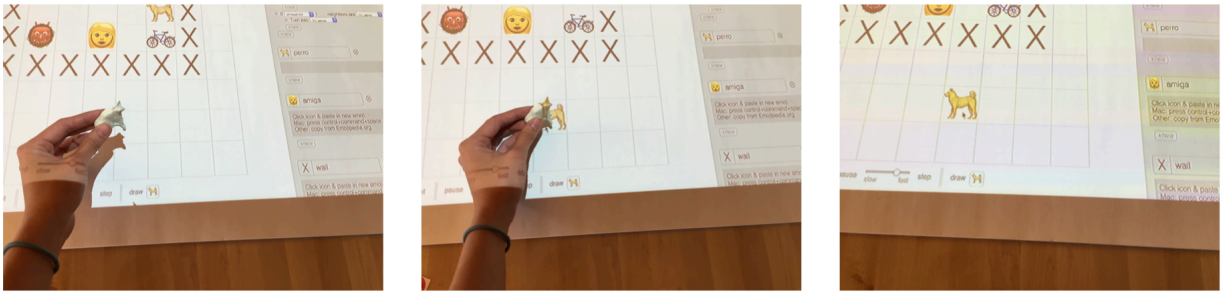
Wednesday Feb 24: robot turtles + nicky sim on paper

- After that, Nagle purchased the “programming” board game called Robot Turtles. (It’s lying around near the lunch table.) Comenzar a hablemos en espanol, tal vez porque el juego no era muy interesante. You basically have to get your turtle to a gem, and you do this using “direction” cards. The one neat aspect is that if you find a repeating pattern in the directions, you can use the “function frog” and turn the repeating pattern into a function. It teaches you real good how to write imperative code.



I also wanted to see what it would look like to be able to use Nicky's emoji sim in the world so we hooked up a projector back onto the lunch table mount that Toby set up.

- I wanted to get a feeling for an idea that you could "stamp" out the emoji into the simulation world.
- Like this:

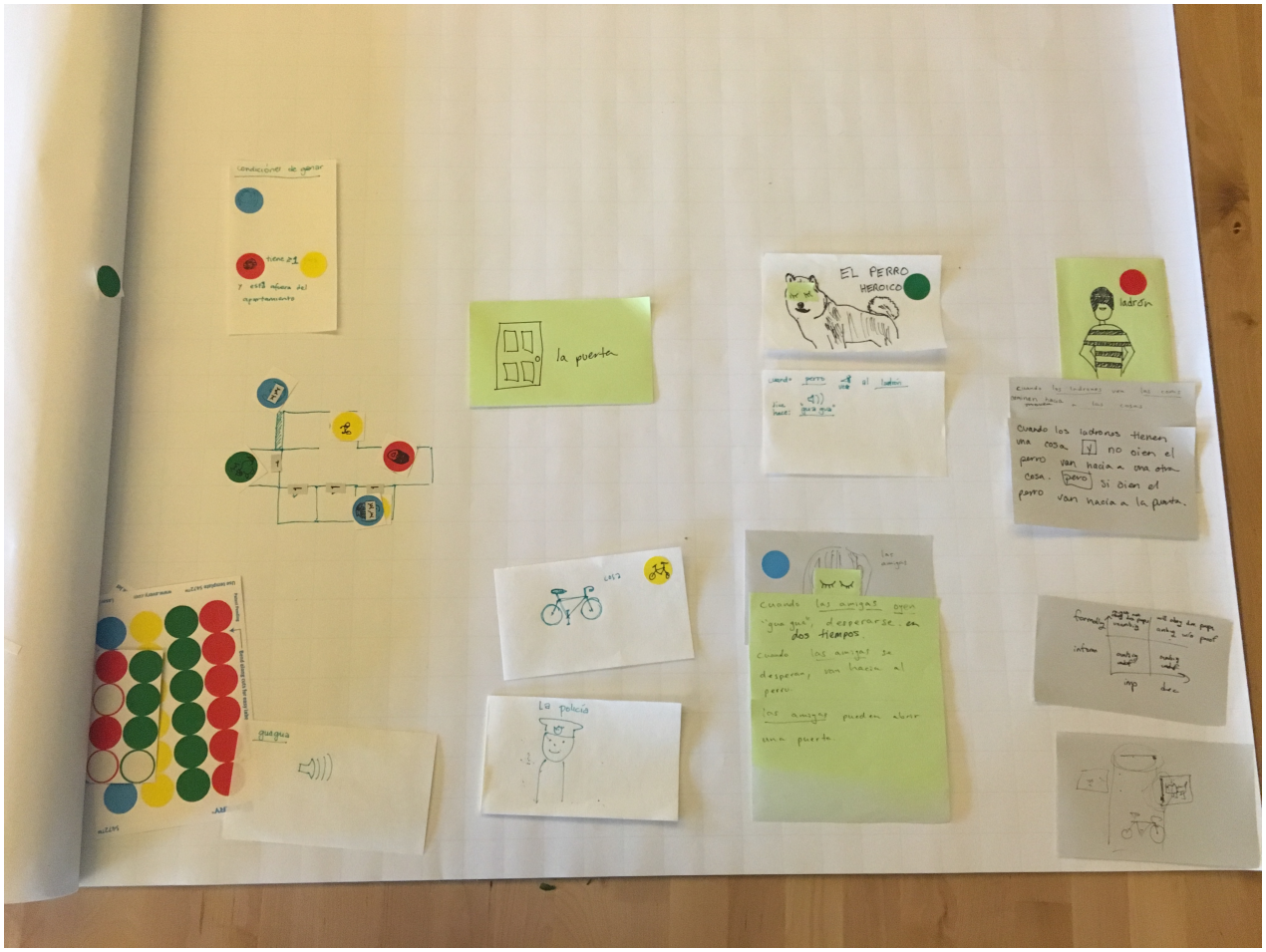


We immediately noticed that the conversation was much more collaborative on a projection on the table than on a screen, even if we can't control the medium by touching it.

- 

Thursday Feb 25: collaborative sim from scratch (no interpreter)

- On Thursday at 2am some thieves broke into my house and stole my flatmates' bicycles (the flatmates' shiba inu barked and woke my flatmate up, but obviously she was too frightened to open her door & confront the thieves mid-theft). I groggily went to work y otra vez hablemos en espanol; le dije al nagle, josue, y emily lo que ocurre, y fue muy gracioso. Nagle suggested we simulate the what happened, so we created a sim with some reglas.

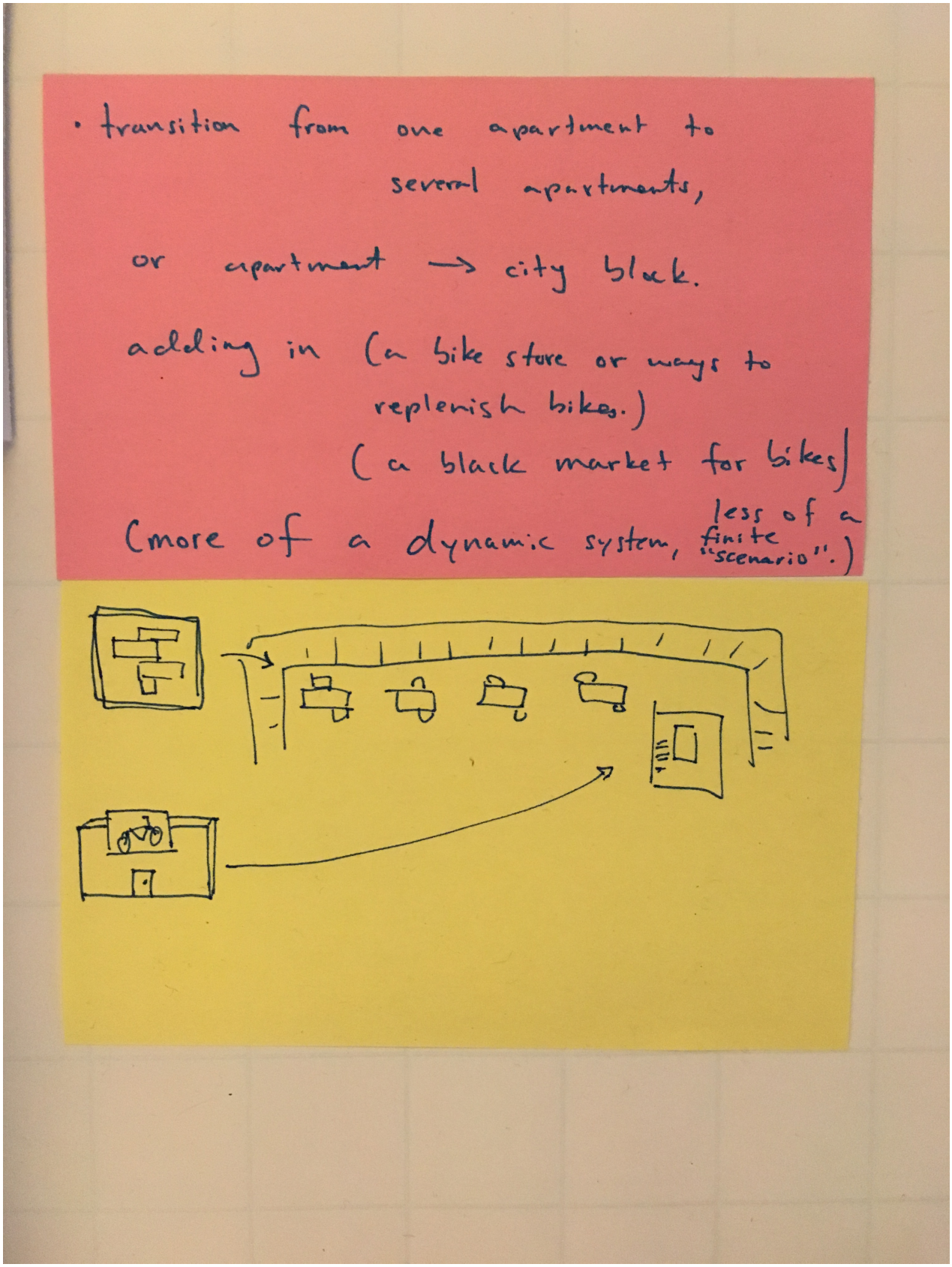


Analyzing this, it was really interesting to see what questions came up when we were creating this simulation entirely from scratch. We were both the programmers and the interpreters, so questions came up like, “Can dogs smell thieves multiple squares away?

Monday Feb 29: flesh out rules & questions of the thief sim

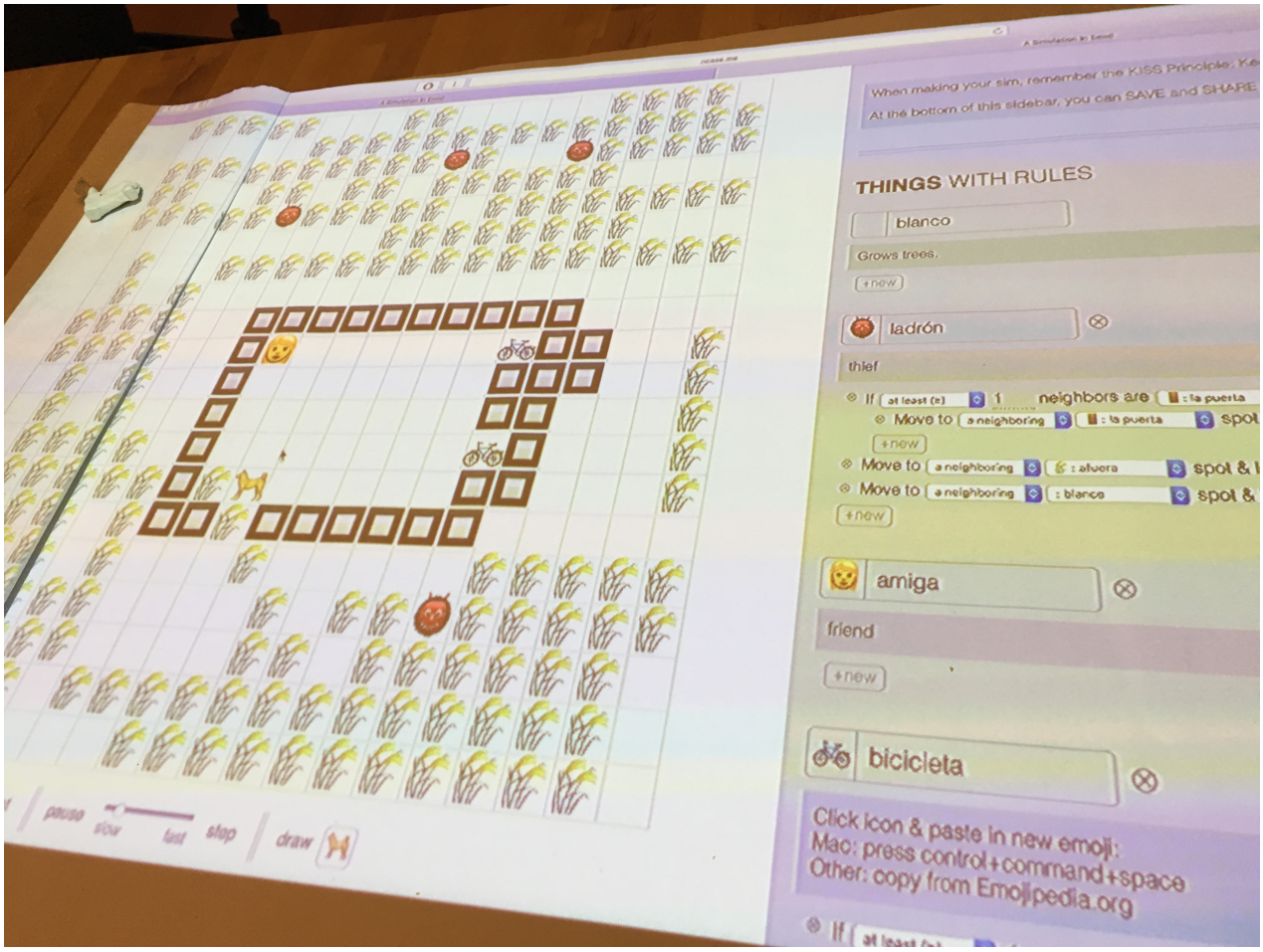
- We abstracted out some of the rules into goals/drives and wrote out all our questions we had about the sim. We think we’re on to a simple way of letting the kids program a few agents & inanimate objects, on a grid. We also talked about this becoming a hook into larger simulations; what if you could take this apartment and stamp out 5 more apartment scenarios on the field, then build a bike shop around the block, and start to

see a more complex system of the problems of city thefts?



- Simplified goals/drives
  - Ladron drives:
    - Steal bike
    - Run away
  - Amiga drives:
    - Wake up
    - Chase robber
  - Perro drives:
    - Bark at ladron

- **Tuesday/Wednesday Mar 1, Mar 2 : replicating sim in Nicky's emoji sim**



- We started to replicate the sim in the emoji sim, and took notes of what we'd like to do with our sim that this current implementation can't easily do. We finished this up on

Wednesday. Here are things we think would benefit our world that don't yet exist in emoji sim (in order of paula's perceived importance)

- pathfinding (knowing how to get from A to B)
- seeing radius (seeing objects more than one square away)
- Scenting (objects leaving a trail) (we think we can implement this with existing rules of emoji sim)
- environmental state (giving a property to the tile of the simulator, rather than an object)
- also related: allowing two agents to stack on a tile
- defining regions (wrapping up a region within a "function" and replicating it)
- laying down walls and doors on edges, rather than on spaces.