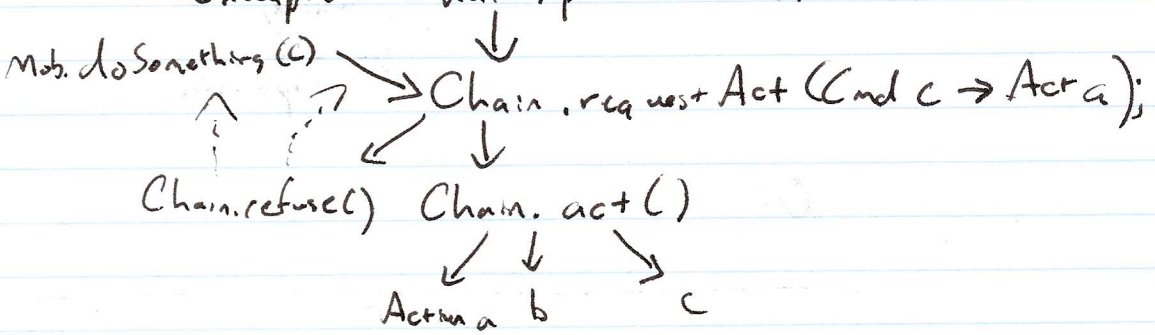
11/24/09

Immediate

2, Work Item Categories:

① Fix Actions
- Chain.requestAct(Action a);
- Basically need to add a level of abstraction between actions and actors "requesters". Everything should ask Chain to do something, not do it themselves.
  - Example: Chain.process(cmd C);



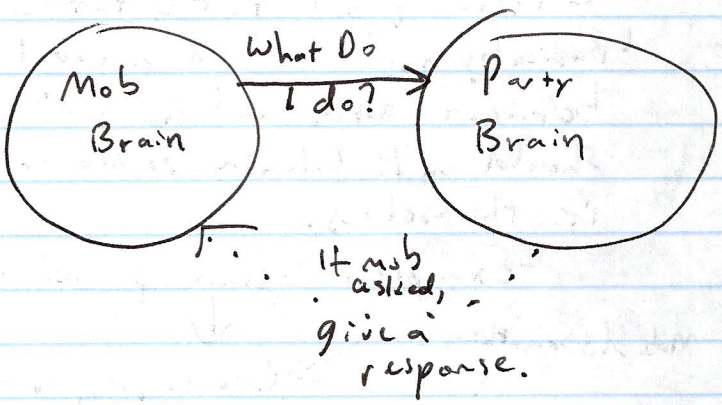
② Update/Fix Mobs
- Stat additions
- isAlive boolean, fix death/mob removal
- Fix attack command
- Move mob location from Mob to Chain.
- Make Mobs/Players/World more robust, interesting

Next steps (Research):

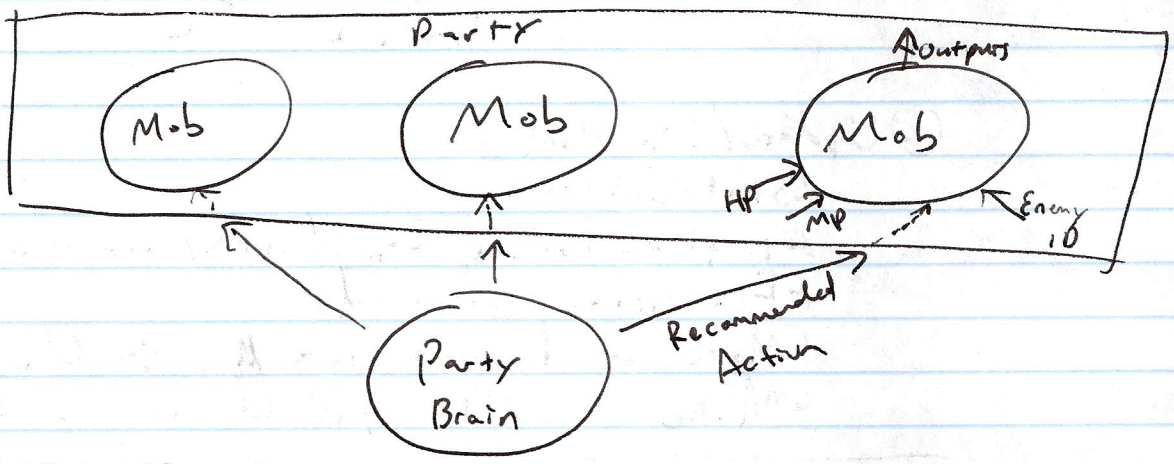
① Implement mob classes, followed by mob parties with interactions.
- How will mobs interact?
  - One idea is to have a brain that controls the mobs' party mentality that will be shared among all mobs.

So, when a mob makes a decision in battle, it will always have an input called "party recommended decision".

2 possibilities:



Or,

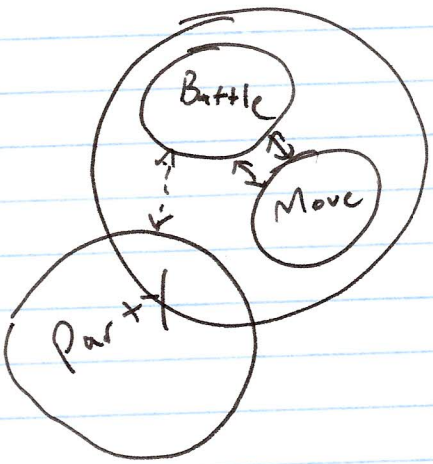


So here, a mob can take in the party brain's recommended action, then decide to ignore it, or to use it.

Want to see if this will cause mobs to only listen to Cmdr of party, do stuff on their own, etc.

② Brains behind mob movement.

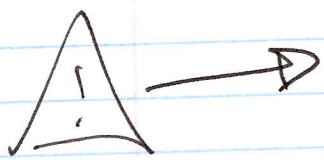
So now a mob's brain has 3 "sub" brains...



How is fitness handled here? A mob's fitness should include all of these, but be based on battle success

The move brain can have all the inputs as the battle brain, plus maybe an "in-battle" boolean. We want to see if the mob will learn to run if it sees it is under attack and has low HP.

↳ Maybe it will learn to run and cure?
↳ If fitness is based on party damage done, and healers can res, will certain mob classes learn to die?



③ Automatic formation of classes. If parties are in place, will certain mobs learn to become healers, others dps, etc. to reward the party brain?

#3 #1

Homogeneous vs Heterogeneous teams and evolution therein.