

HOW OUR
MOBBING SYSTEM
ADAPTED TO
REMOTE

The Remote Experiment

Firstly, who am I?

- Born and raised in San Diego, CA
- Bachelor's degree in Computer Science from California State University San Marcos
- Two years working at Hunter Industries as a Software Engineer

What I work on



- Hunter Industries is an irrigation manufacturing company
- Customers range from domestic and international homeowners to large complexes to golf courses
- Software talking to the controller makes it all happen

How do we mob?

7 hours a day with 1
hour of learning

10 people on my team –
3 mobs all working on
the same project at the
same time throughout
all layers of the stack

All job levels working
together under constant
rotation

Have an Agile
Retrospective once a
week with whole team

How do we retro?



Radical feedback



Assume good intent



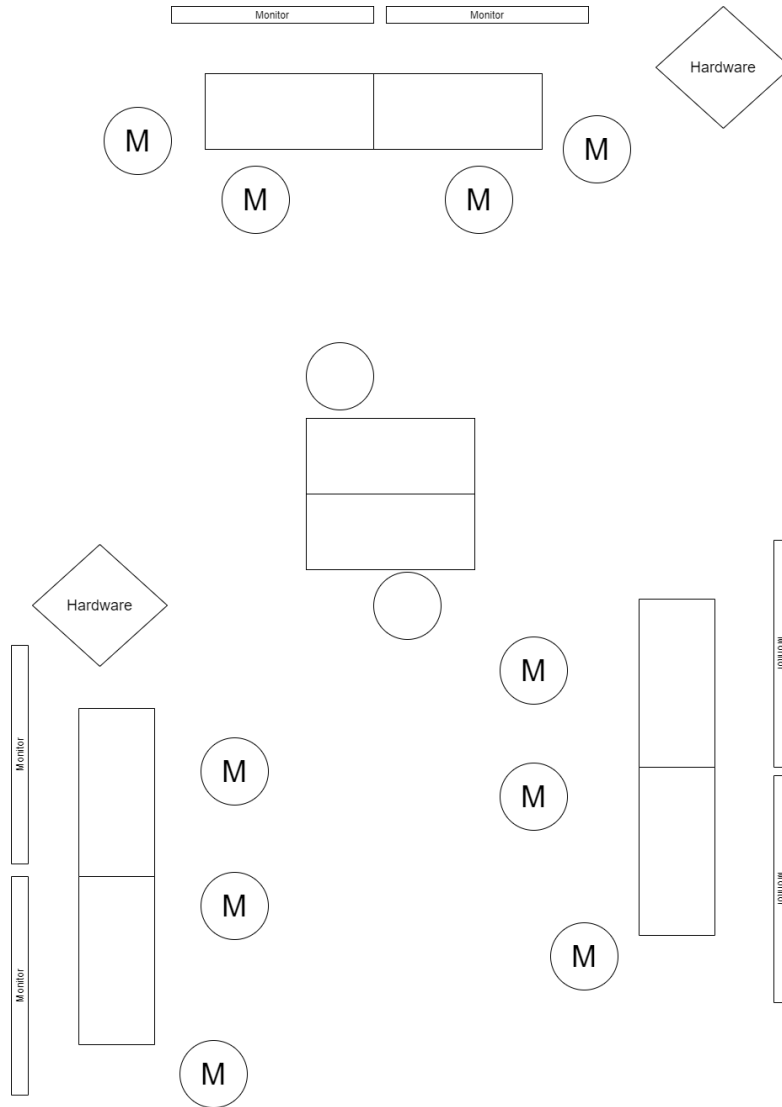
SMART action items



Everyone gets a chance to facilitate

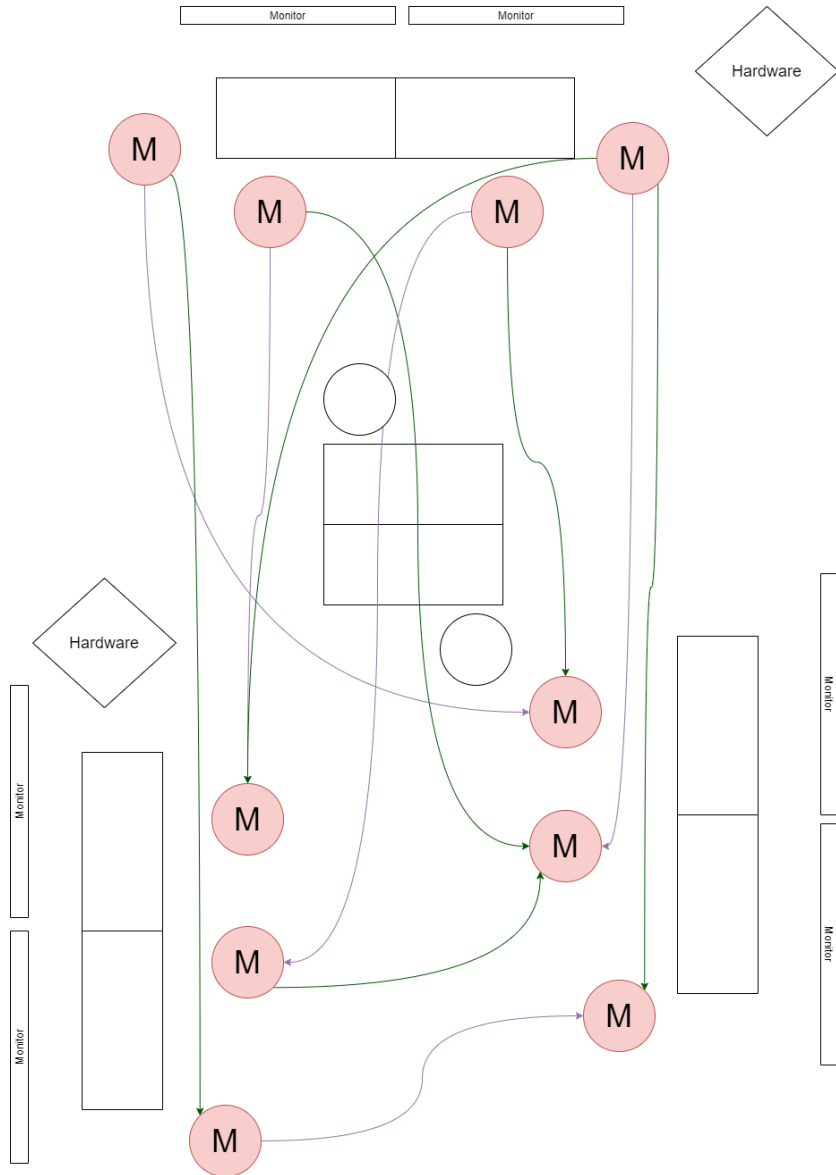


Format is up to the facilitator that week

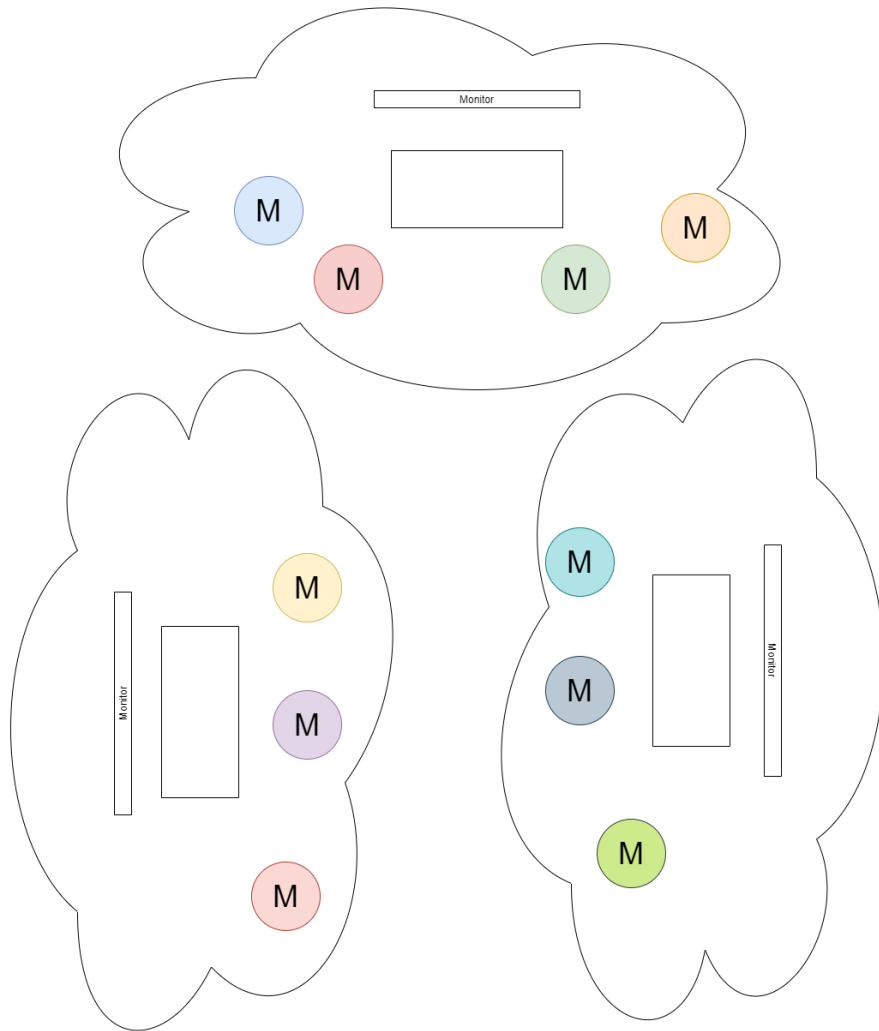


Where did we mob?

- All mobs in the same open room
 - self organized free-form concept
- Everything on wheels
- Three other project mobs on either side with sound blocking walls



What happens when we take this complex system home?



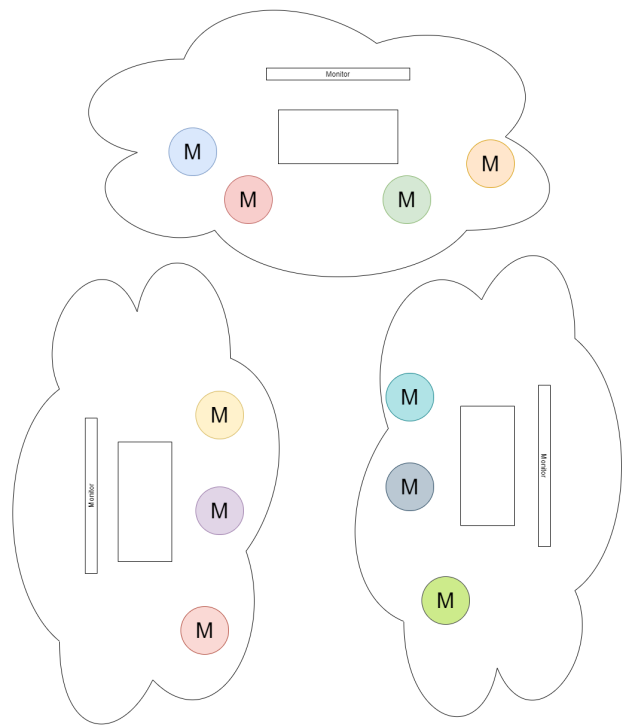
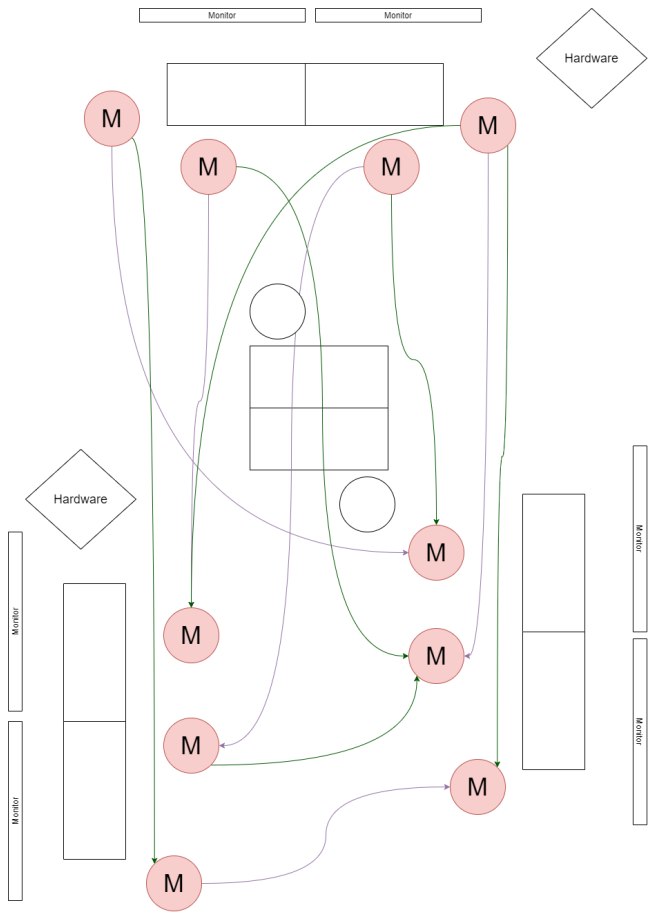
The New System

- Technical access was literally cut in half
- Now dependent on personal systems for performance and ability to mob
- New technologies
 - AnyDesk
 - Teamviewer
 - Microsoft Office365
- No easily shared space for large hardware

But the tech wasn't the problem.

- We eliminated any technical problems, but why were we failing to communicate properly?
 - Long silences in meetings
 - Long delays in answering questions
 - Would hesitate to ask questions of other teams
 - Less awareness of what other teams are doing
 - Less engagement while working in front of small screen

We weren't communicating effectively!



We had
overestimated
our ability to
communicate

There were gaps in communication within/without the group – silence silos



We had neglected aspects of the project and training

Face-to-face contact became more important than ever

But what did we discover?

Silence Silos

- We couldn't turn around and listen to other mobs anymore
- Isolating each mob led to overall less communication
- We only retro once a week, and during those retros we don't want to discuss technical matters
- Hurts us when it comes to talking to our product owner

Silence Silos

➤ Experiments

- Designated speakers in meetings – easy to forget in the moment ✖
- Have two daily standups everyday – became too many standups to be substantial, moved to 1 ✔
- **Switching more often to see more code – hard to really gain context in a week and work effectively** ☁

Code Neglect

- We have old parts of our code that no one has touched in a long time or ever
- People who joined the team were not trained effectively to work on the team
- Frequent switching made it obvious where knowledge gaps were in old and new coding practices

Code Neglect

➤ Experiments

- Refactoring day to work on upgrading old code – too hard to figure out how to refactor these obelisk pieces in a day ❌
- Try doing smaller isolated refactoring for smaller victories – in progress 🧱
- Work on training/resources for the team to use to get familiar with project and languages ✅
- **Is everyone engaged or even interested in the code?** 🤔

Face-to-Face Connection

- We were talking to a circles on a screen, so engagement was low
- In the office, we were surrounded with people and hardware
- It became hard to communicate effectively without seeing a person's expressions

Face-to-Face Connection

➤ Experiments

➤ **Cameras on during all retros**

➤ Conducting a survey to measure levels of engagement within the smaller mobs 

➤ New retro formats were introduced

➤ More group learning sessions

Silence Silos -
experiment with
switching

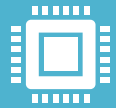


Low
Engagement –
turn on cameras

What did we
discover in the
end?

Failures lead
to the
underlying
problem in a
team

What are three things I have learned from going remote?



What can seem like a large and complex problem sometimes calls for the simplest steps towards a solution



Fail often



Mobbing will never stop being an experiment no matter the environment you are in



Questions
