

This Team's Current Debt

The current largest pain point is that bugs cross between code from several teams. Bugs bounce back and forth and get blocked on other teams.

Last Meeting's Hypothesis

- Bugs aren't prioritized equally for all teams.
- Cross team communication and prioritization needs improved.

This Meeting's Goal

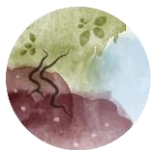
- **Team** is sharing results of this line of inquiry.
- **Leader** wants to improve team's ability to own tech debt.

Beliefs

The problem is clearly that bugs that bounce between teams and then sit there with no action for days.

Remember Your Stance!

- Ownership is more important than the team finding the right solution immediately.
- The team has more info than you, always.
- Learning happens when there are unexpected outcomes. You don't want everything to happen as planned or hoped!
- If the team asks for ideas, help them create options rather than give direction.



This Team's Current Debt

The current largest pain point is that bugs cross between code from several teams. Bugs bounce back and forth and get blocked on other teams.

Last Meeting's Hypothesis

- Bugs aren't prioritized equally for all teams.
- Cross team communication and prioritization needs improved.

This Meeting's Goal

- **Team** is sharing results of this line of inquiry.
- **Leader** wants to improve team's ability to own tech debt.

Beliefs

Tracking and communication experiments are simply not fixing it.

Diagnosing these bugs is hard because 1) it requires deep knowledge, 2) the information is arcane, and 3) only a few people understand the code well.

Key individuals lose time thrashing between tasks, and bugs block on them.

Everybody on both teams are trying to do the right thing.

Remember Your Stance!

- You own this. You probably know more than the leader.
- Don't try to convince; just try to show what is true.
- If you need a resource, ask for it. But labor is not a resource – your labor is your own and you don't have to ask for it.

