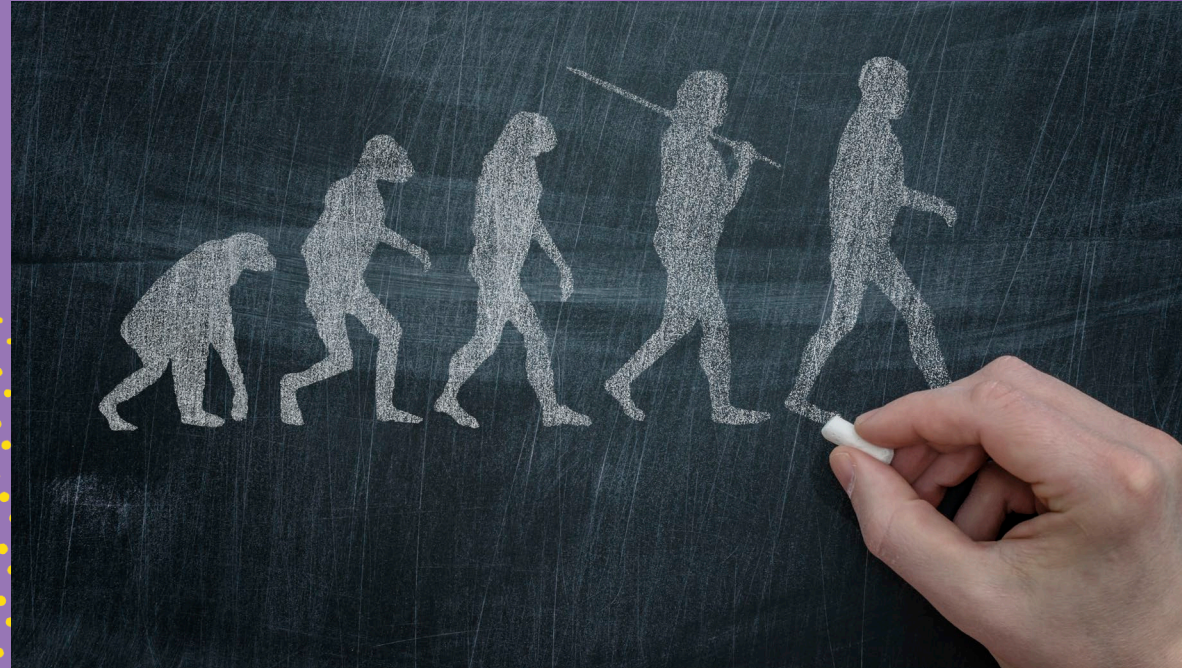




# Evolutionary Change Workshop

The Foundation of the Kanban Method



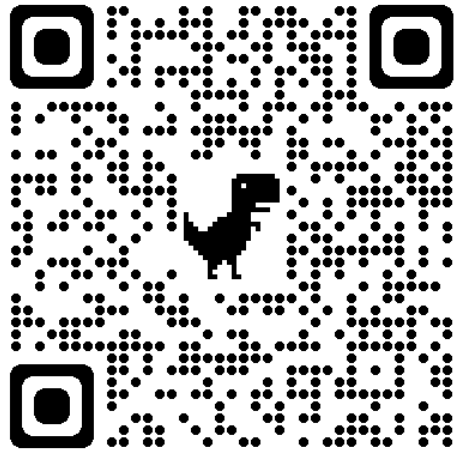
**Todd Little**  
Chairman, Kanban University







[linkedin.com/in/toddelittle/](https://www.linkedin.com/in/toddelittle/)

# Todd Little

## Chairman, Kanban University



# Outcomes from Agile Transformations

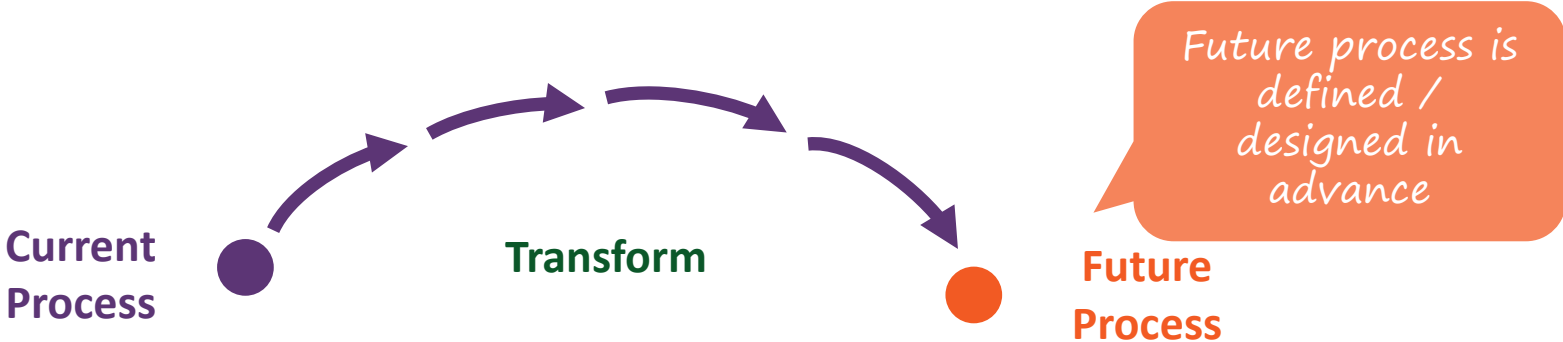
|  |  |
|--|--|
| <p>Agile worked for us, and we continue to get better.</p>  | <p>Agile helped a bit, but we haven't gotten much better lately.</p>  |
| <p>Agile has been a disaster.</p>                          | <p>Agile didn't really help, but didn't really hurt either.</p>      |



**"58% of Scrum implementations  
are late, over budget,  
with unhappy customers."**

**— Jeff Sutherland**

# Traditional Transformation is an A to B Process



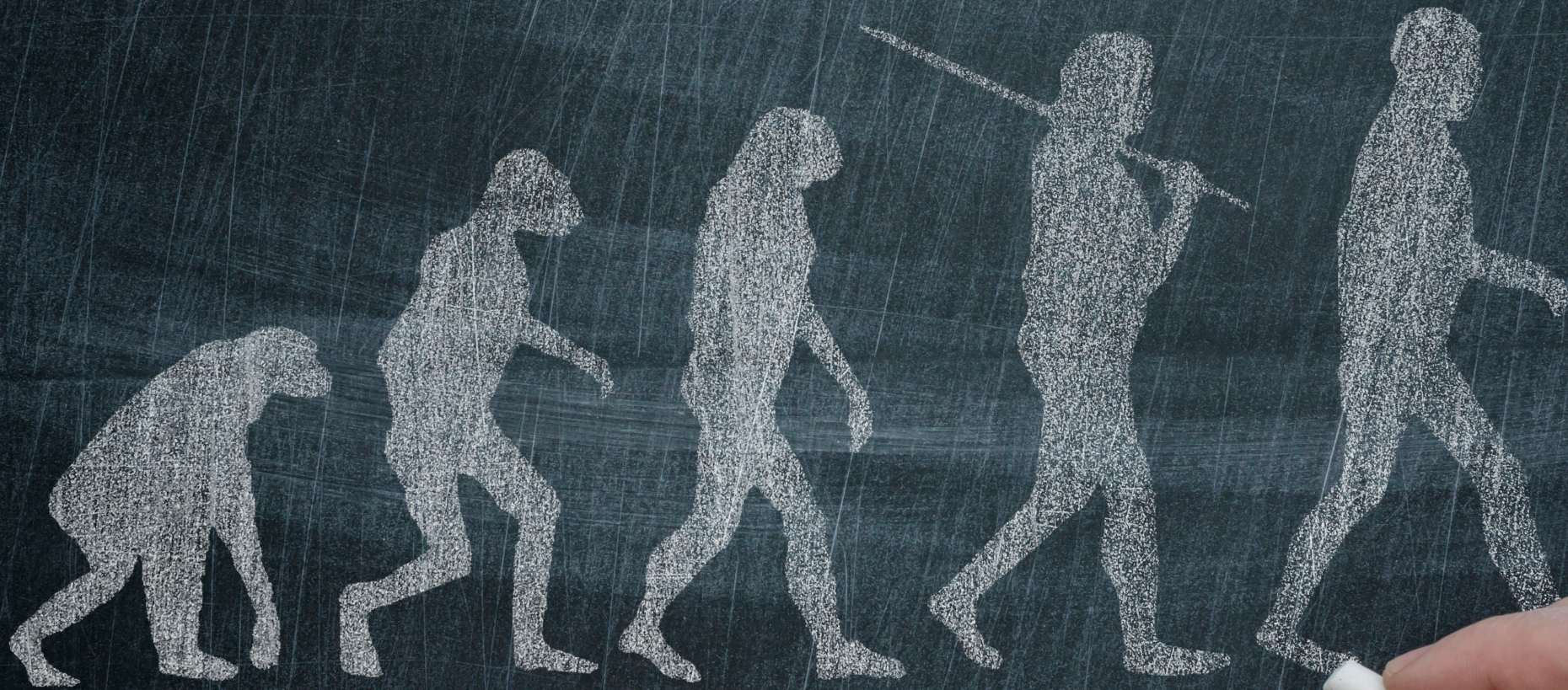


# Start with what you do now

- Add Kanban principles and practices on top of your existing process or framework

Existing Process or Framework

- Kanban is NOT a Process Framework!
- Kanban IS a method for managing and improving knowledge work service delivery through evolutionary change



Evolutionary Change is Humane



# Two types of Evolution

## Darwinism / Gradualism

- Evolution generally occurs uniformly and by the **steady and gradual transformation** of whole lineages.
- This is what most people associate with “evolution”.

## Punctuated Equilibrium

- In response to rapidly changing environmental conditions there is a **rapid burst** of evolution – it is changing a whole lot in a very short amount of time.

# Push vs. Pull





**TO CHANGE**

“

“People do not resist change,  
they resist being changed!”

Peter Senge

# Change Management Principles

1

Start where you are!

2

Agree to pursue evolutionary change.

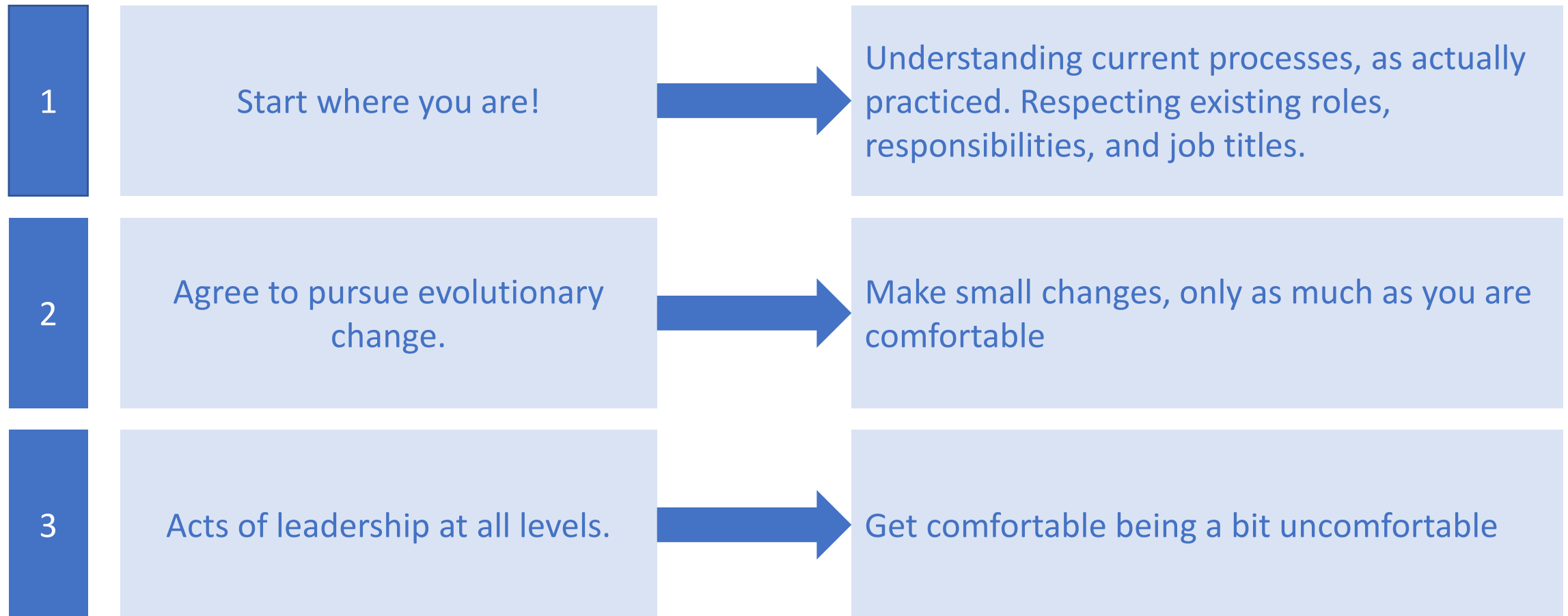
3

Acts of leadership at all levels.

With Kanban, we start by developing and visualizing a joint understanding of the current way of working.

Then, improvements are identified and implemented.

# Change Management Principles



# Evolutionary change formula

Areas to consider when planning out evolutionary change.

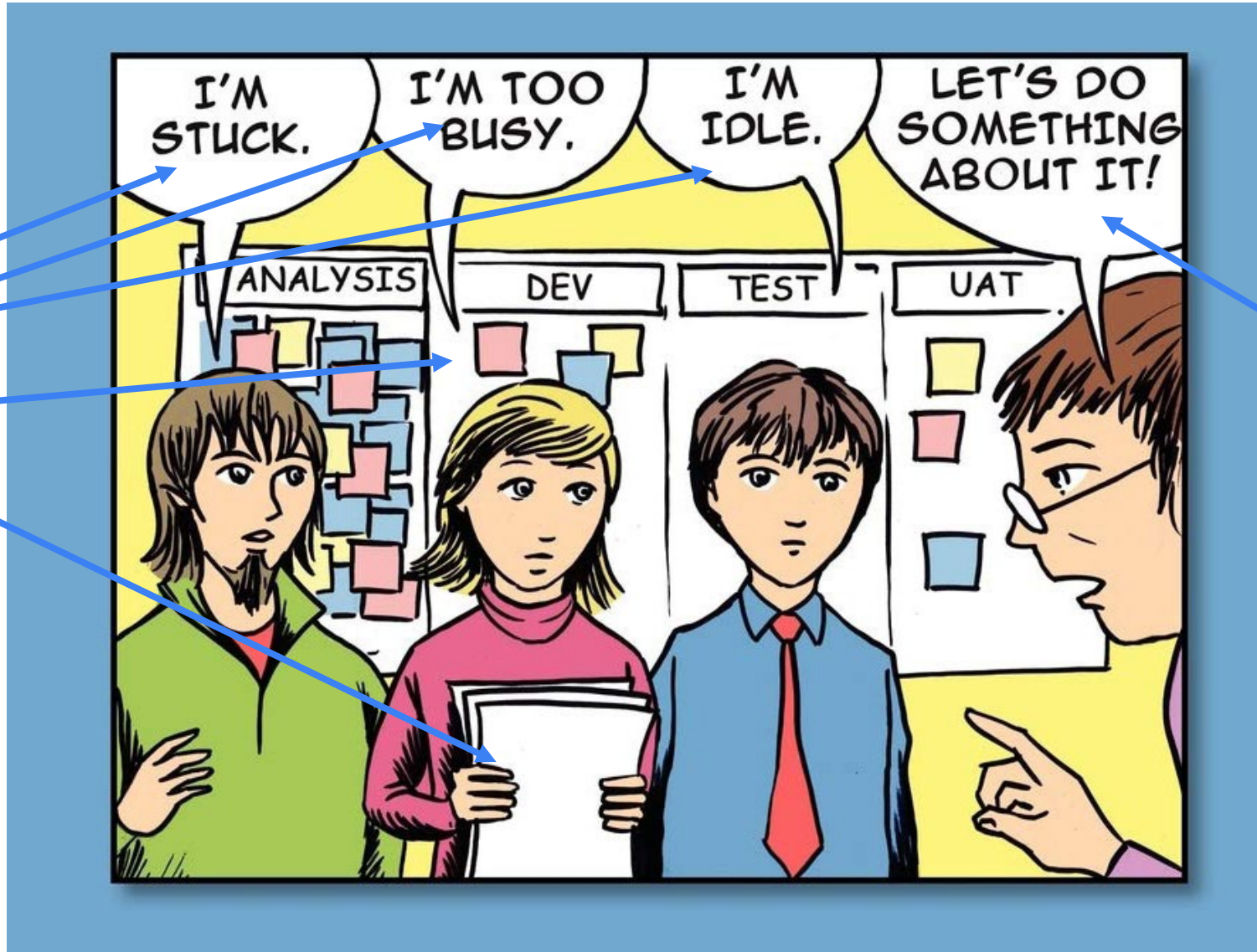


The Change Potion



# Feedback Loops - Reflection

Feedback Mechanisms  
Communication Meetings  
Board  
Reports and Metrics



Feedback Loop  
Acts of Leadership



“

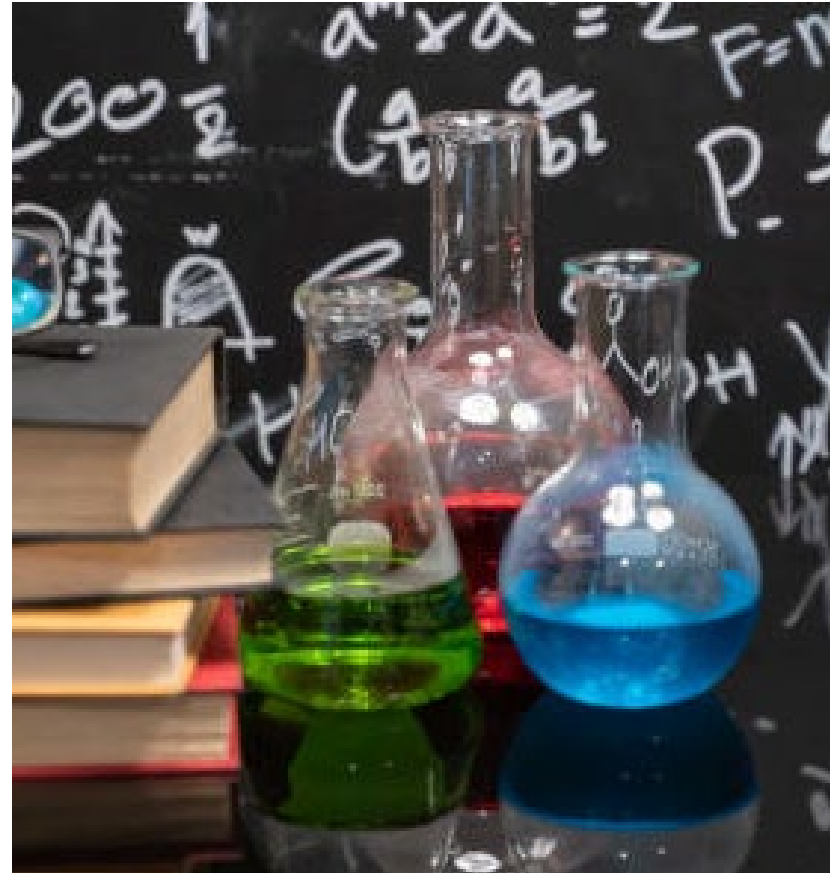
Feedback without action  
is not a feedback loop.

# Feedback Loop?

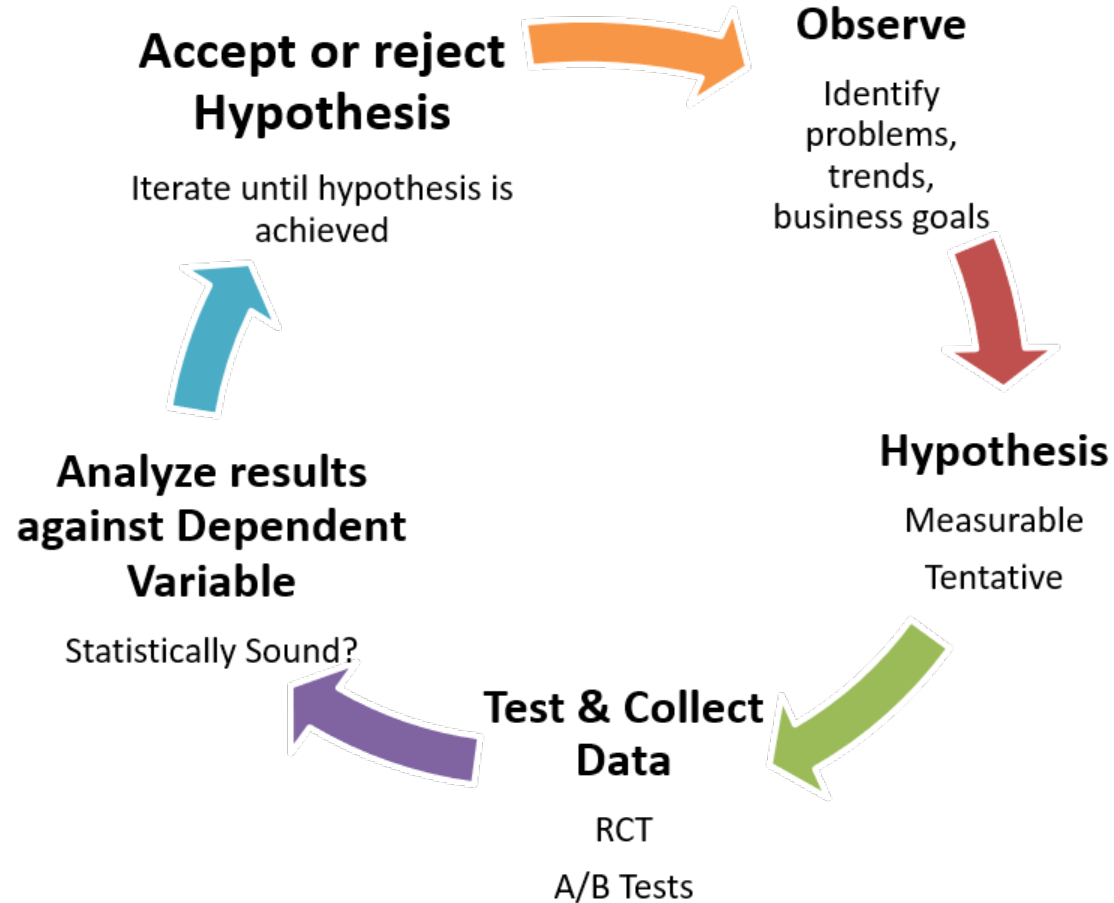


# Don't leave it to chance: **MANAGED** Evolution

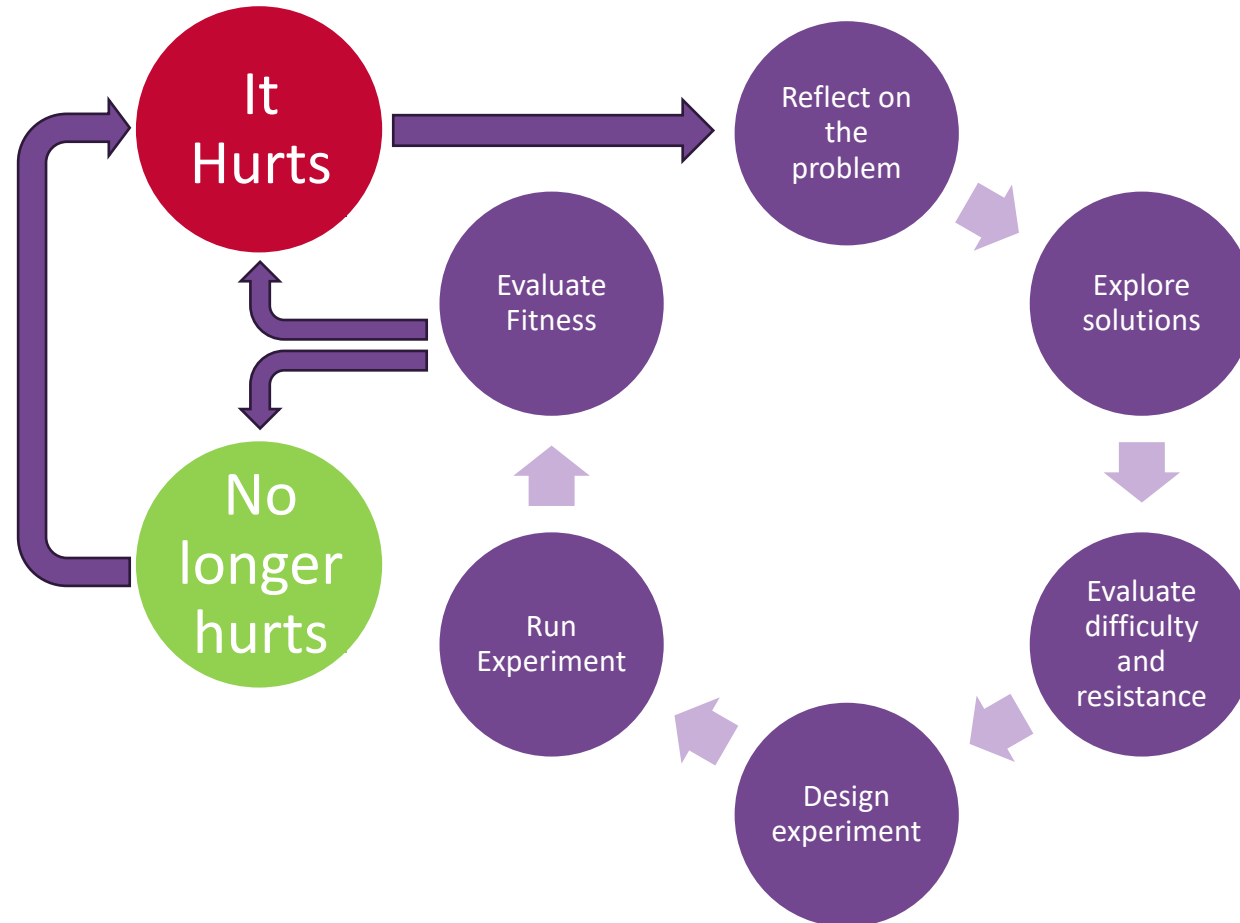
Only mutate that which isn't working well. Preserve that which is.




# The Scientific Method



# Improve collaboratively, evolve experimentally



# It Hurts



Here is what hurt for my team.

we receive mixed or conflicting messages from different team members

Feeling isolated in our individual tasks without a clear view of the bigger picture

We're lacking feedback

Environment is "down" too much and too long

Software engineers push work onto the testers and expect immediate feedback?

sprint scope keeps changing

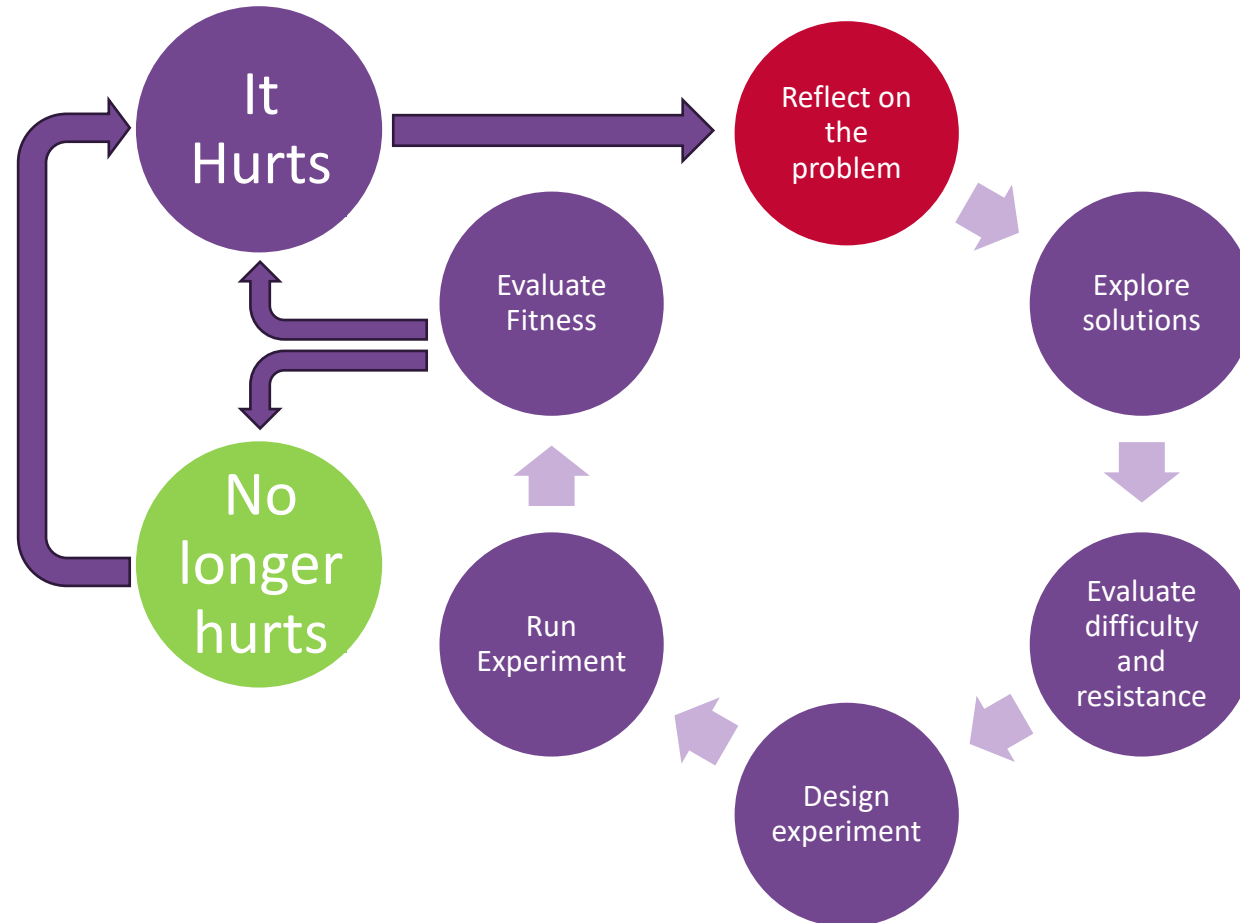
Voicing concerns are being largely ignored

we're constantly set up for failure with goals that are unrealistic

our sprint goals aren't clear, and it's like we're shooting in the dark

Testing takes way too long!

# Improve collaboratively, evolve experimentally



# Reflecting on the problems

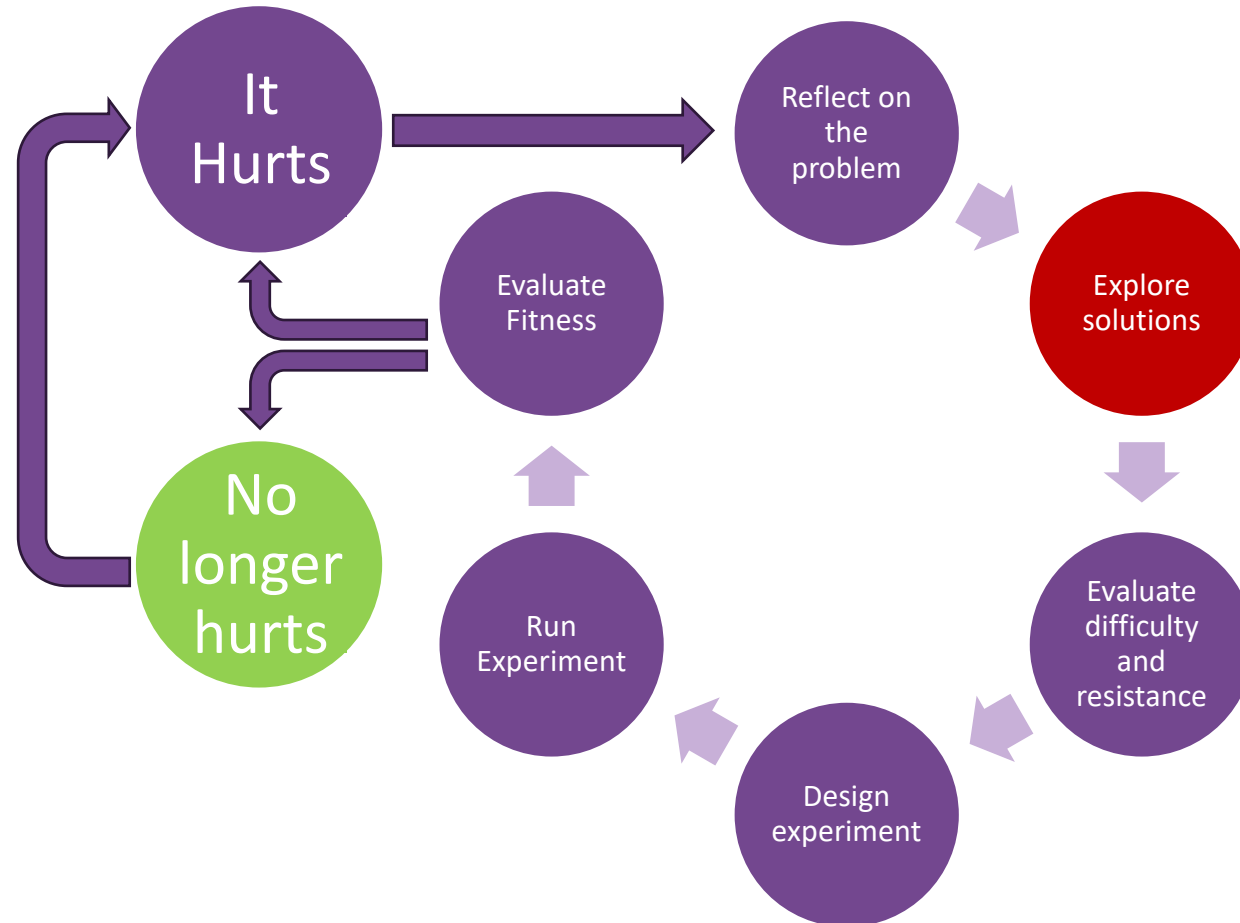


We dug deeper behind the pains and found the following causes.





# Improve collaboratively, evolve experimentally

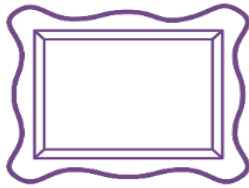


## Visualize

**Show work and its flow.**

**Visualize risks.**

**Build a visual model that reflects how you actually work.**



*no skinny mirrors!*

## Limit Work in Progress

**Stop starting, start finishing!**

**Left yields to right.**

**Limit work in the system to available capacity.**

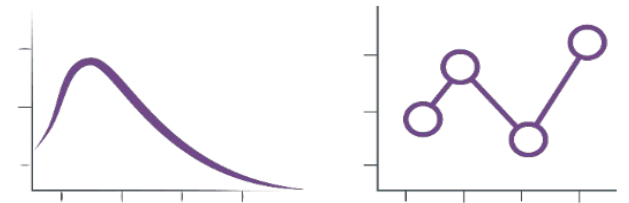


## Manage Flow

**Flow is the movement of work.**

**Manage flow to be smooth and predictable.**

**Use data.**



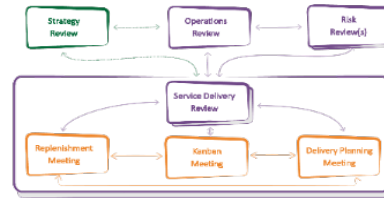
## Make policies explicit

**Have agreed policies, visible to everyone involved.**

- Pull Criteria
- WIP Limits
- Classes of Service
- Dependencies and blocker handling
- ....



## Establish Feedback loops



**Establish feedback loops at an appropriate cadence.**

**Foster collaboration, learning, and improvements.**

**Data-driven.**

## Improve collaboratively, evolve experimentally

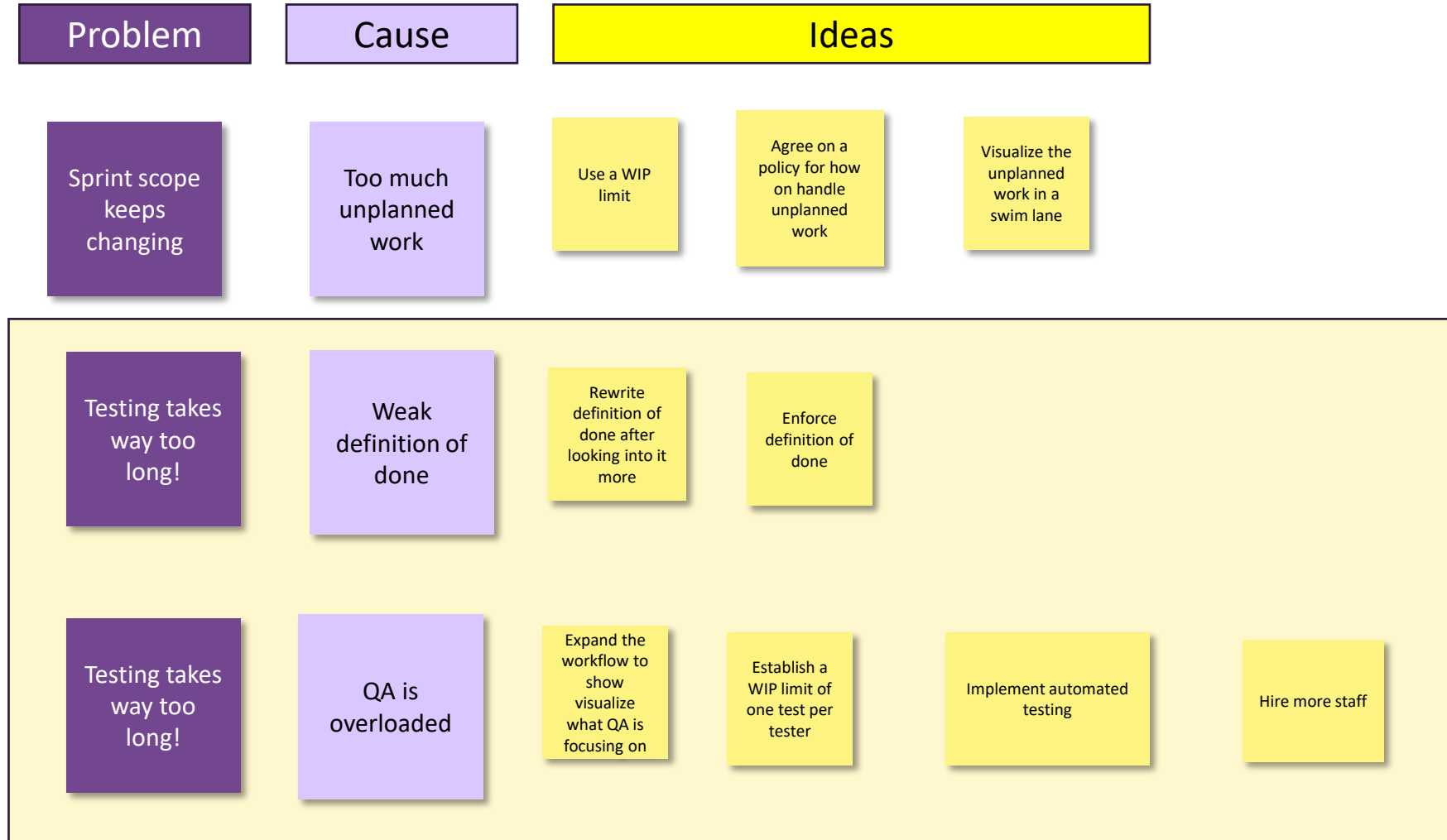
**Using the scientific method.**

**Hypothesis-driven change.**

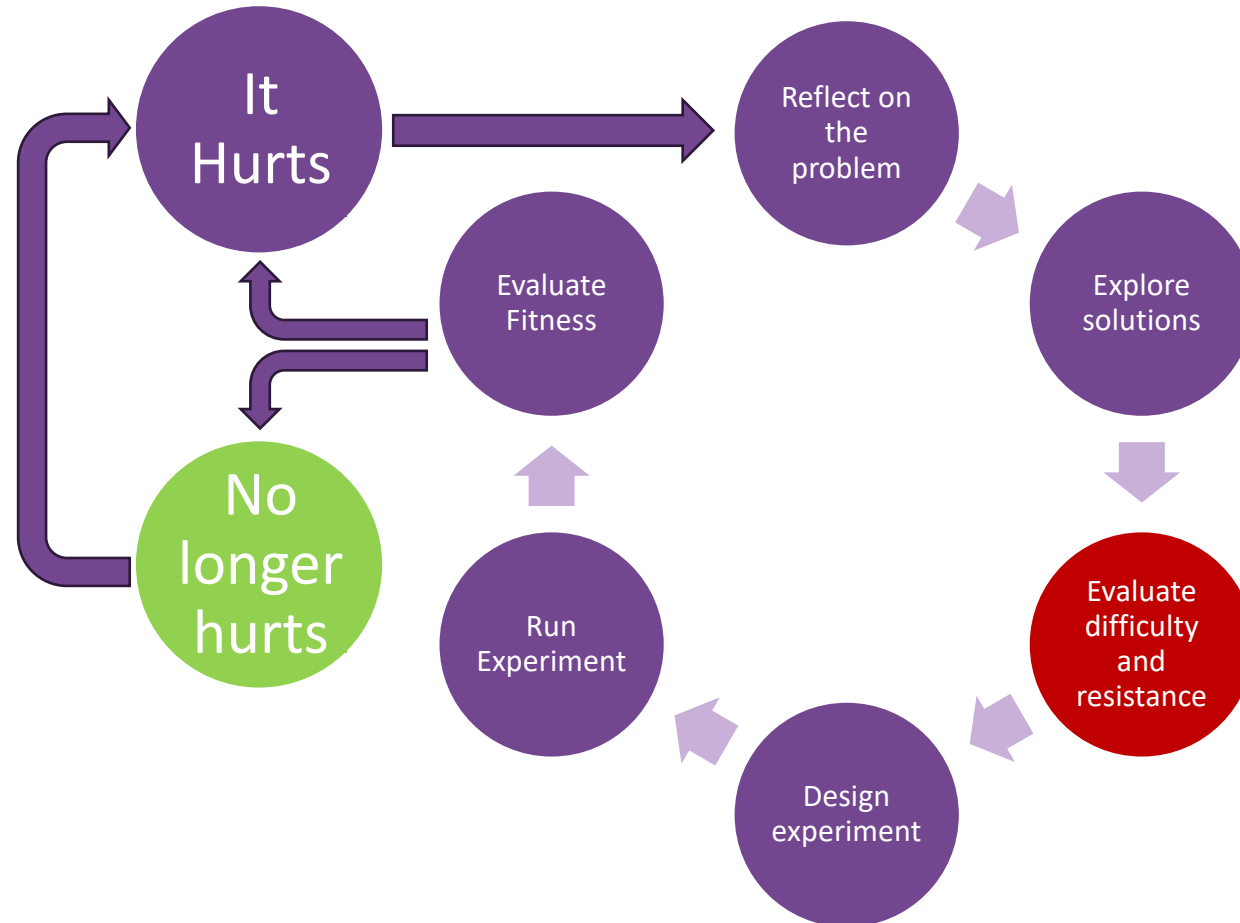
**Run safe-to-fail experiments.**



# The Scrum Teams' ideas



# Improve collaboratively, evolve experimentally





*“be like  
water”*

*the rock  
represents  
resistance*

Water flows around the rock

## Water becomes the cup



We can't be agile unless we change the culture!

Agile needs to adapt to culture as much as culture needs to adapt to agile.

# Why Around the Rocks?

**B.C.**

©John Hart Studios Inc. All rights reserved.



Dist. by Creators

# Around the rocks in real life

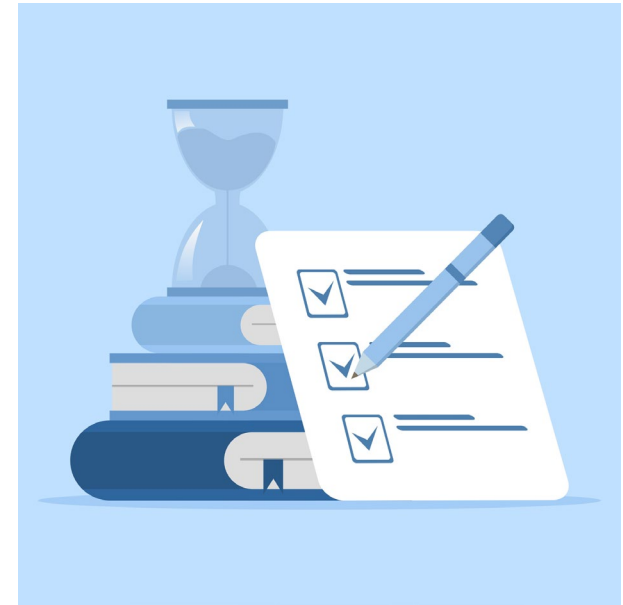
- Under the radar



- Break it down



- Run experiments





# Be like water

Attributes for an idea

## **Resistance from people**

Who will resist your idea? How much? A little or a lot?

## **Technical difficulty**

How hard is your idea from a technical perspective? Really easy?

Does it require iterations to learn and then put it into place (e.g., hard work)?

## **Ability to influence**

How much autonomy do you or your team have to put the idea into place without consulting others?

# Deeper Dive into Resistance

## Resistance to the Problem

- I don't agree that is a problem, or don't think it is important
- I actually benefit from the problem

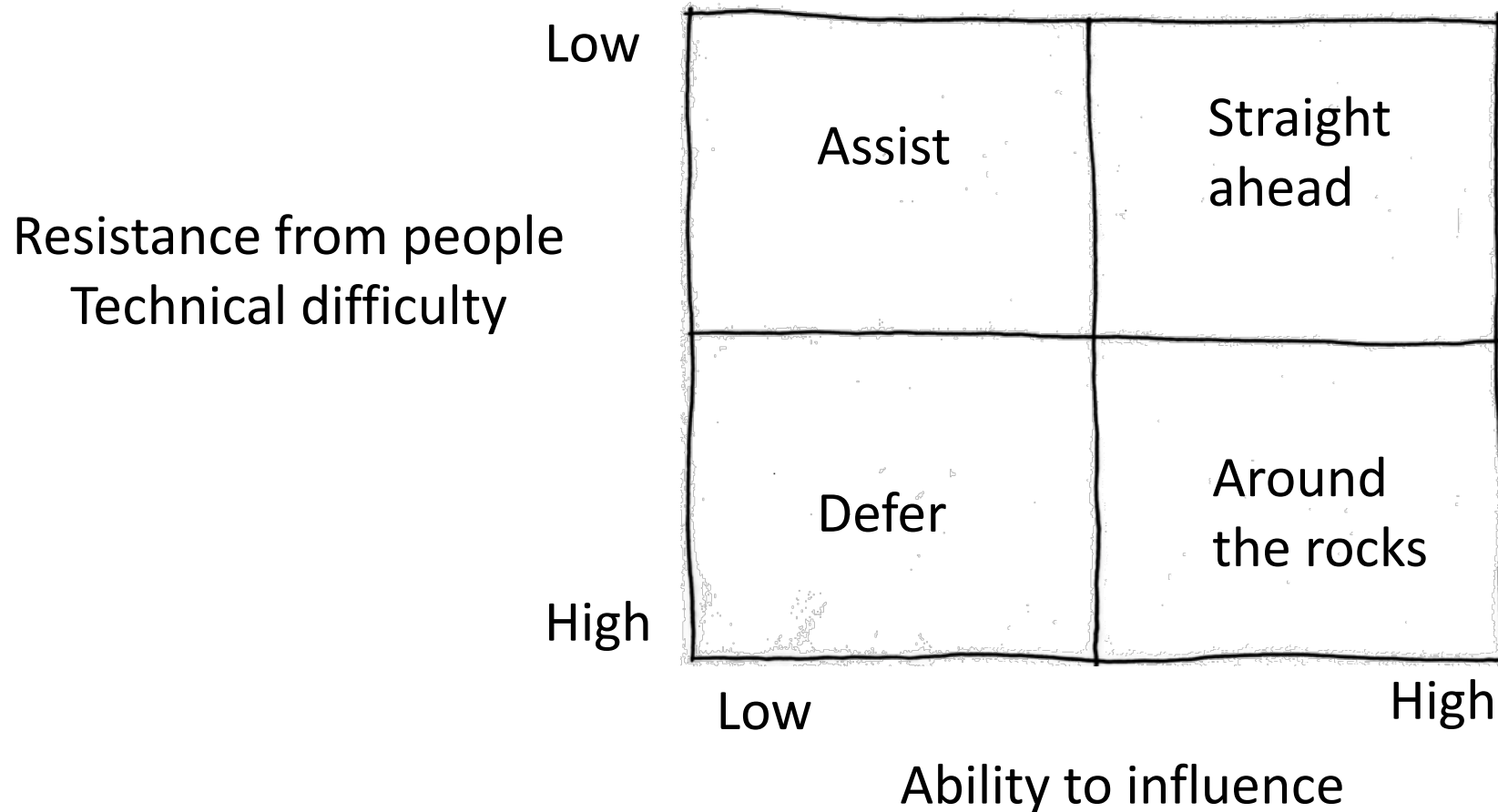
## Resistance to the Solution

- I don't agree that the proposed solution will work, or I have a better solution
- The solution harms me

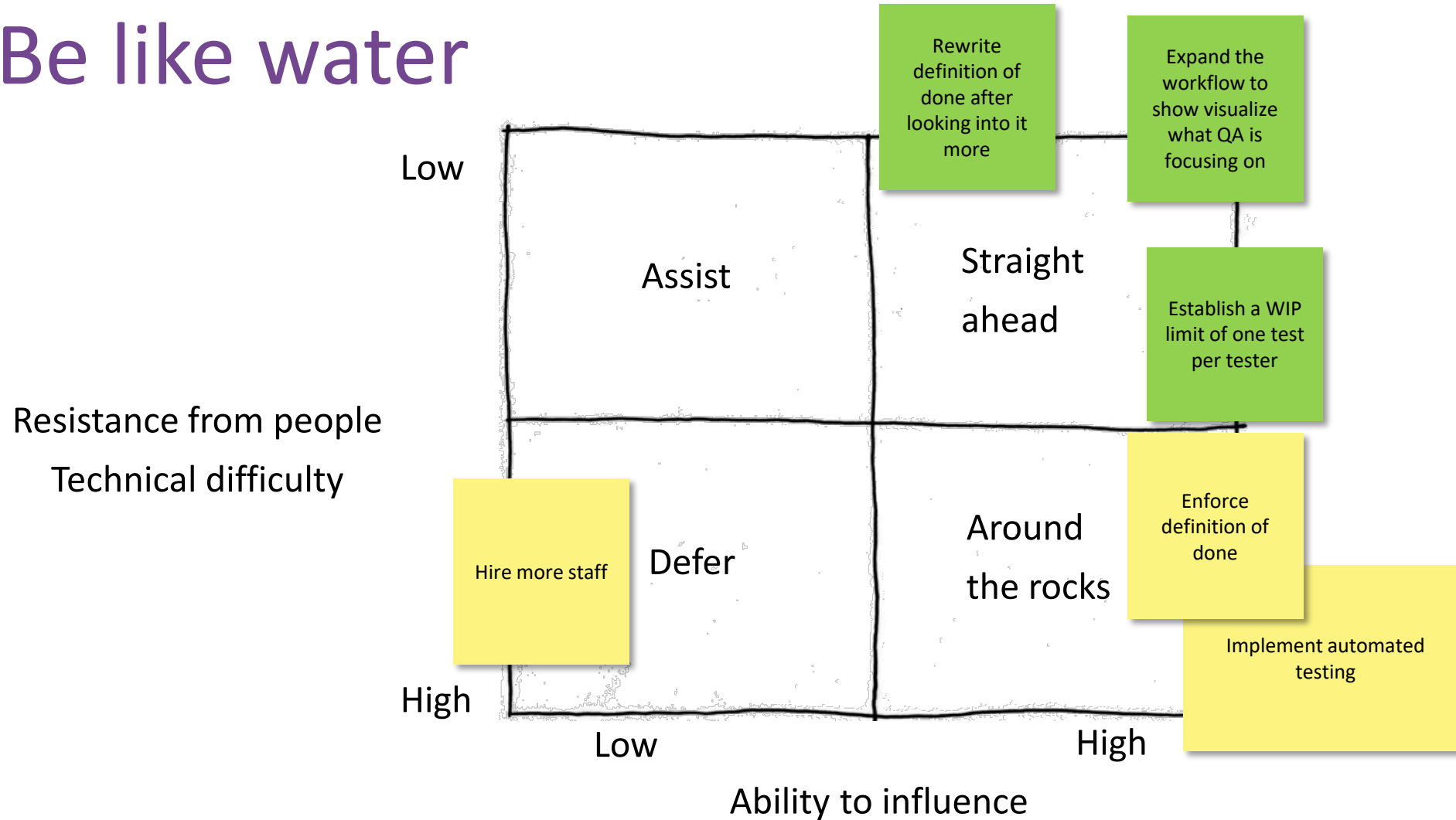
## Resistance to the Implementation

- I don't agree with your proposed implementation, or I have a better implementation
- The implementation harms me

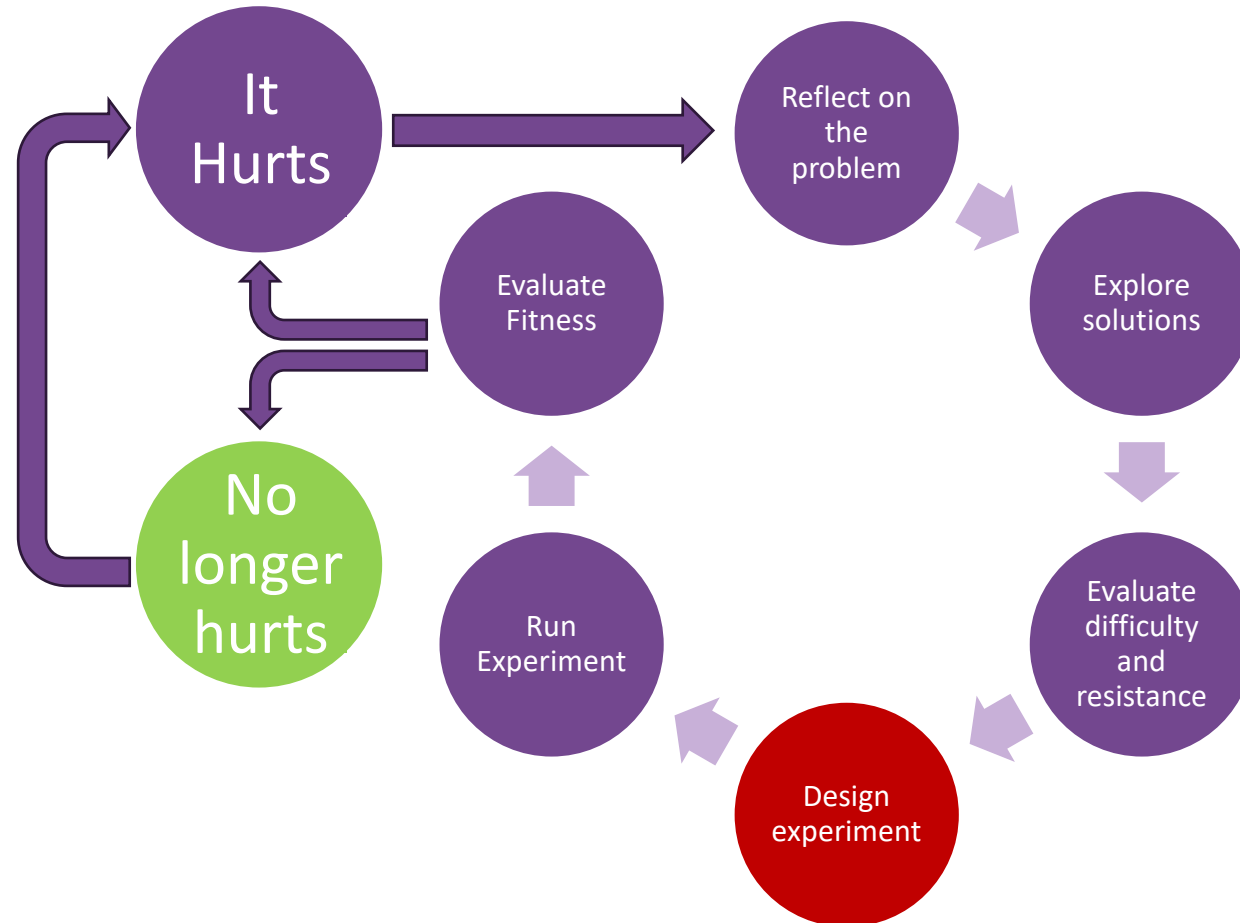
# Be like water



# Be like water



# Improve collaboratively, evolve experimentally



# The Scrum Team's experiment

We believe that if we ...

Expand the workflow to show visualize what QA is focusing on

Rewrite definition of done after looking into it more

Establish a WIP limit of one test per tester

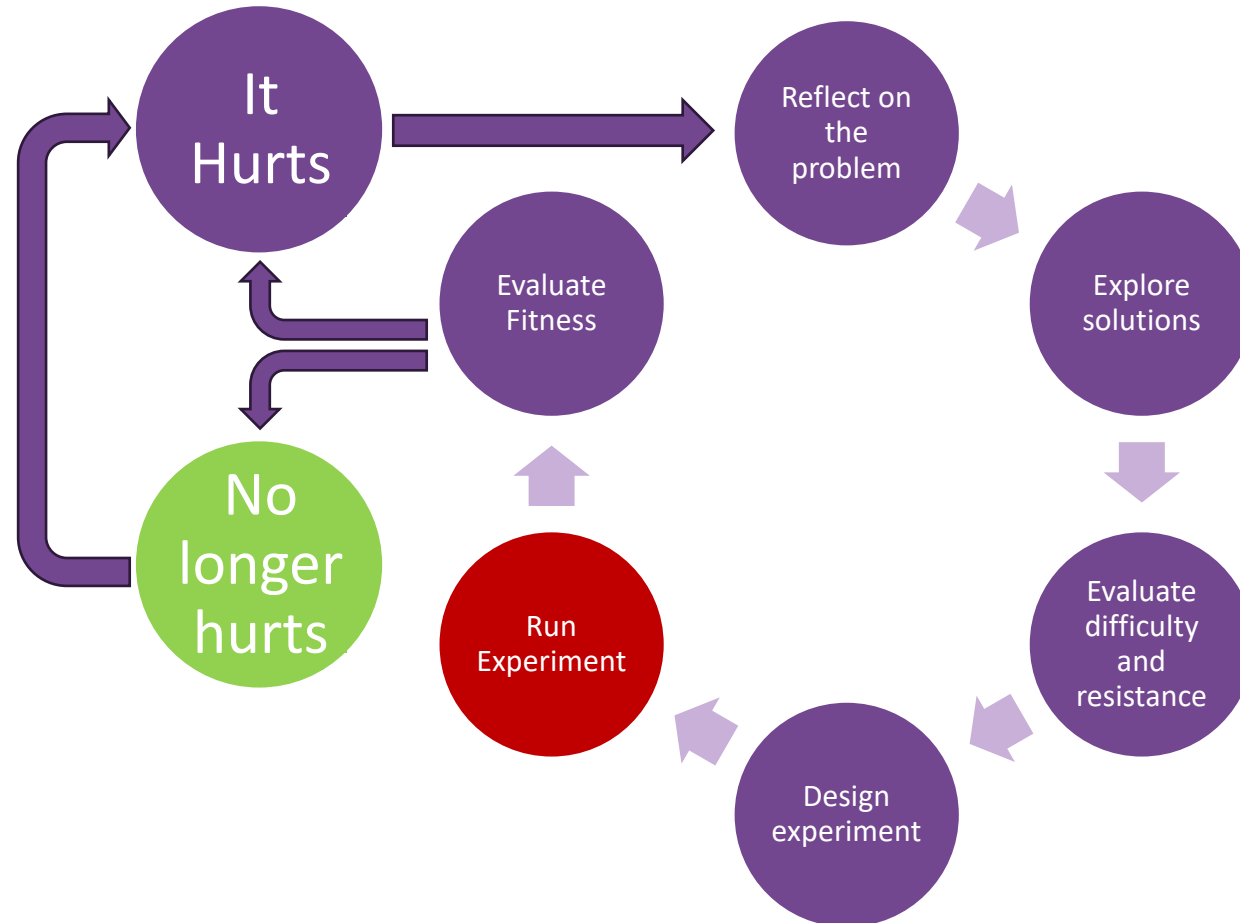
Rosie's team felt they were comfortable taking on all three of these experiments at the same time.

Your team may want to choose one at a time. That's OK!

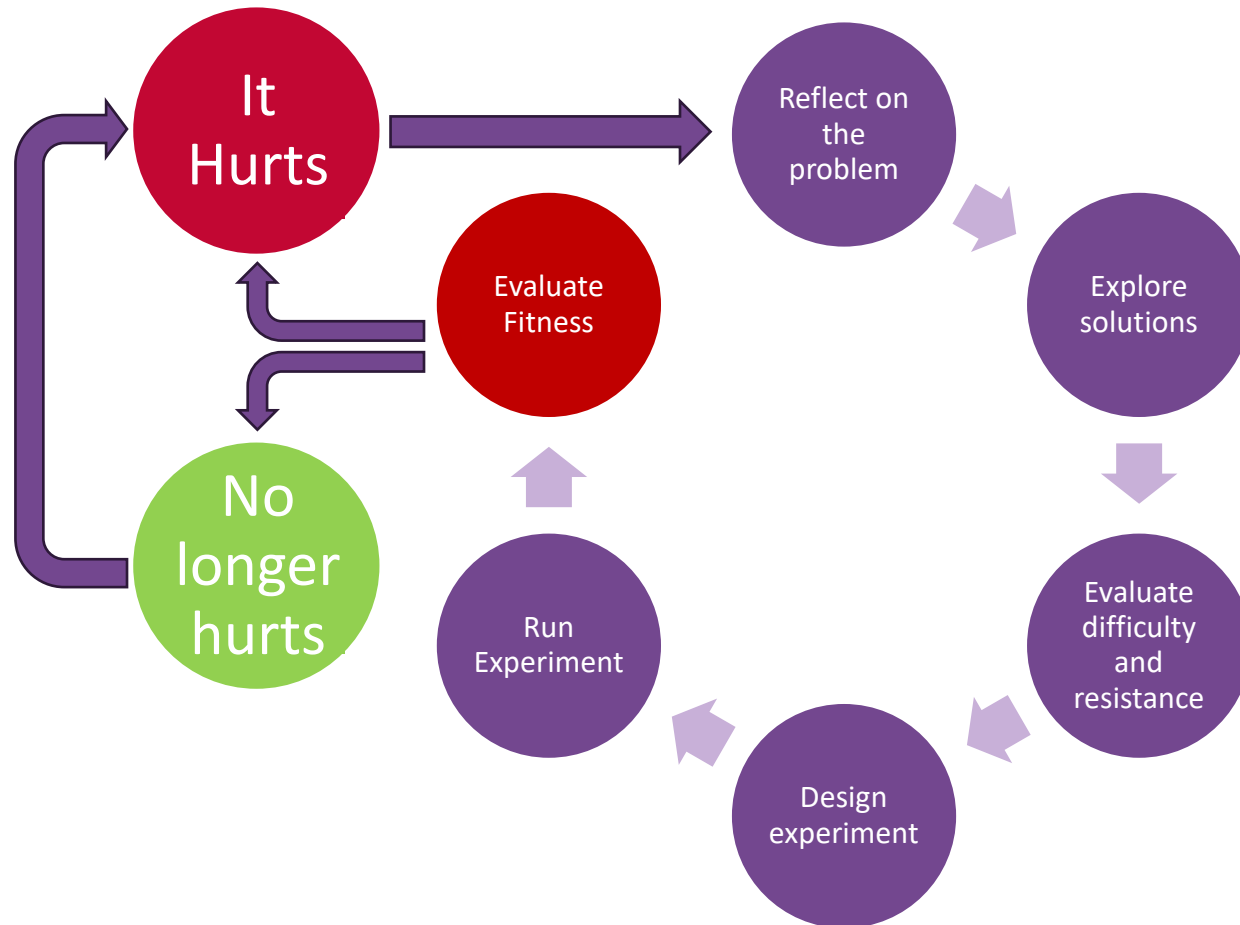
We will be able to address ...

Testing takes way too long!

# Improve collaboratively, evolve experimentally



# Improve collaboratively, evolve experimentally



## Options to consider

Do we roll back a change if it's not fit?

Even if the pain is there, do we want to keep the change and try something new?

If the pain is gone, we want to explore for another pain.

If there doesn't appear to be an obvious pain, consider taking a KMM class to further expand on evidence of hidden pains in your team and organization.

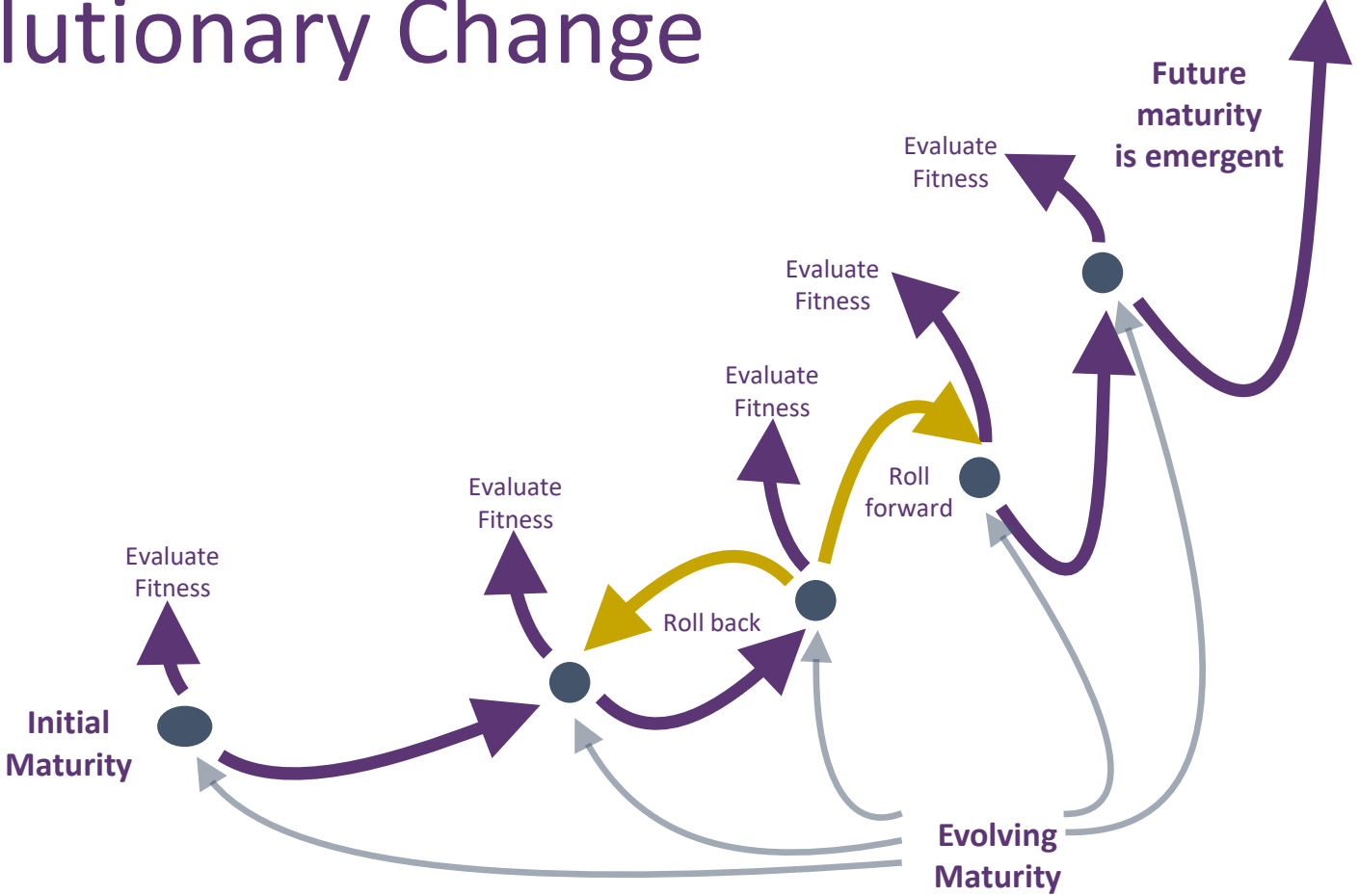


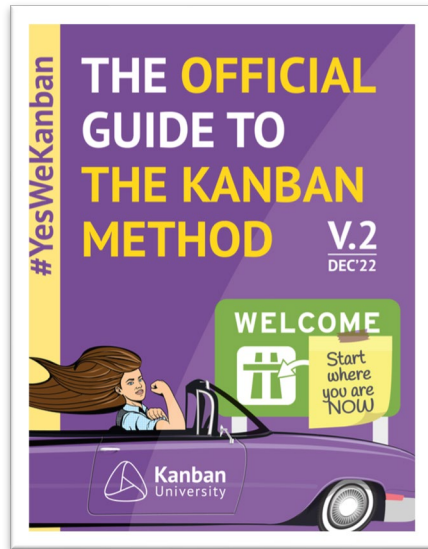
“

Feedback is the breakfast  
of champions.

Ken Blanchard

# Evolutionary Change





**FREE!**

