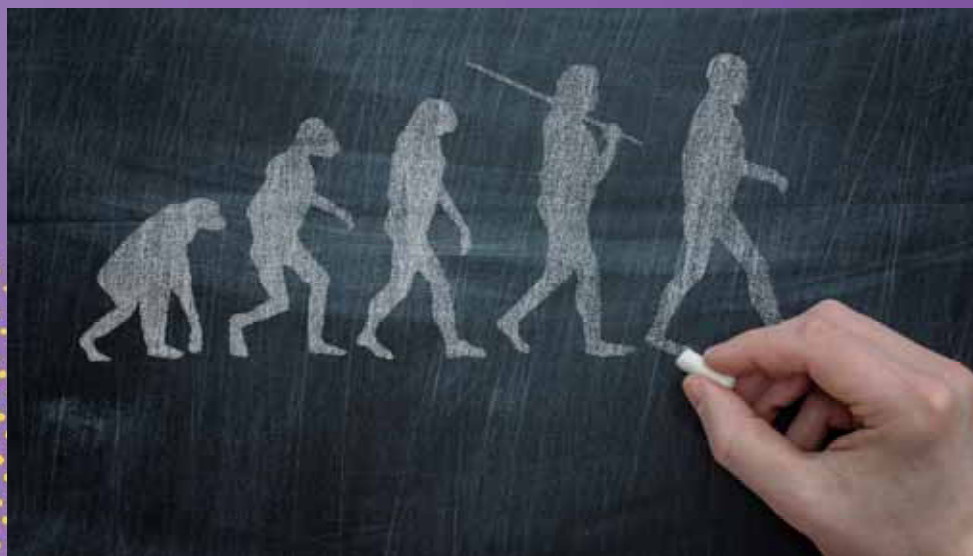




Evolutionary Change Workshop

The Foundation of the Kanban Method



Todd Little
Chairman, Kanban University

Gene Lege
Founder, Agile Earth

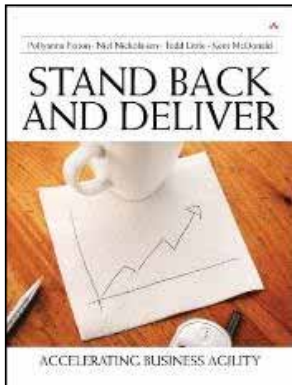
About Todd



Chairman, Kanban University

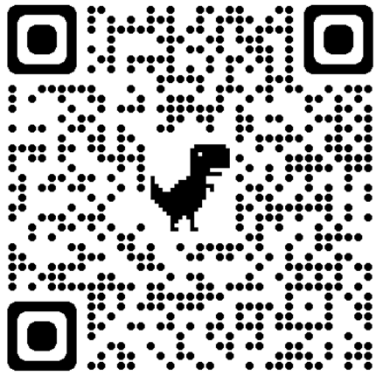
Executive roles as VP Product Development, Director of Software & Technology

@toddelittle

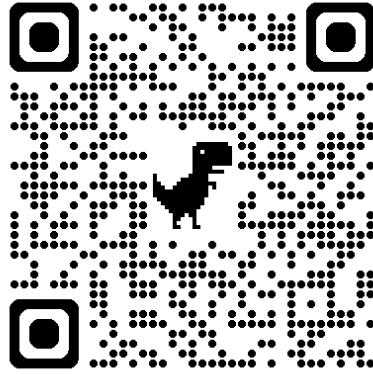








[linkedin.com/in/toddelittle/](https://www.linkedin.com/in/toddelittle/)



[linkedin.com/in/gene-lege/](https://www.linkedin.com/in/gene-lege/)



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> 

Poll – How did it go?

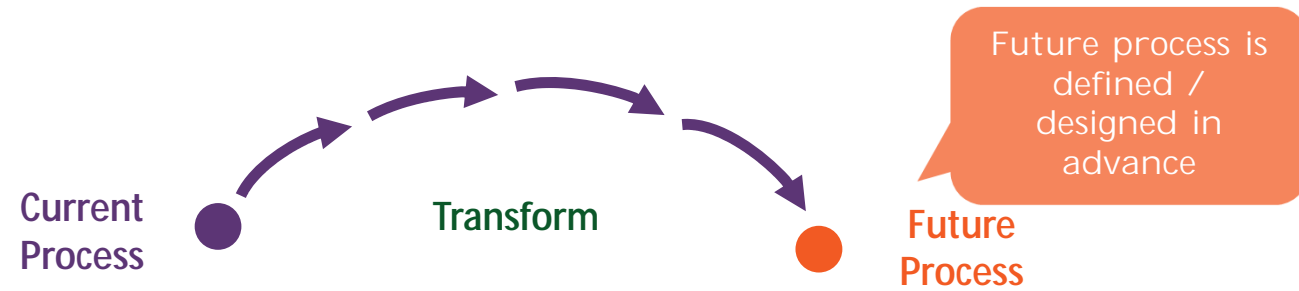
- Agile worked for us, and we continue to get better
- Agile helped a bit, but we haven't gotten much better lately
- Agile didn't really help, but didn't really hurt either
- Agile has been a disaster
- I haven't used Agile
- I hate Agile



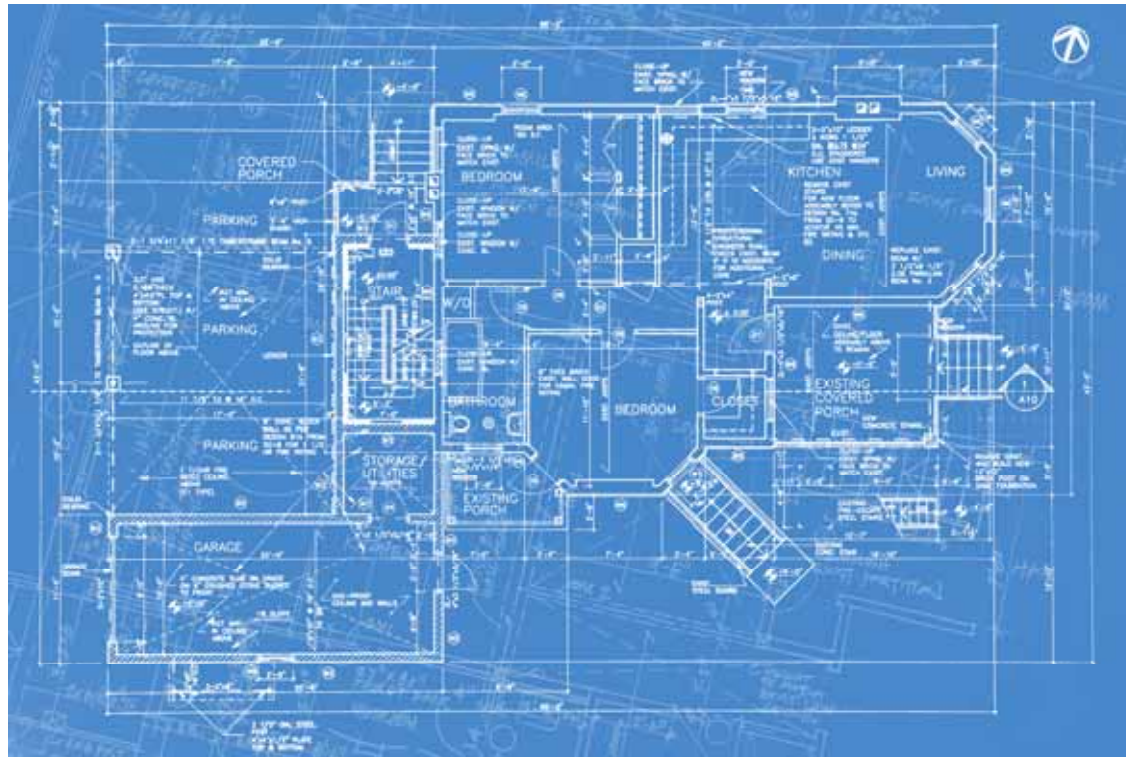
"58% of Scrum implementations are late, over budget, with unhappy customers."

— Jeff Sutherland

Traditional Transformation is an A to B Process



Big Design Up Front



Start with what you do now

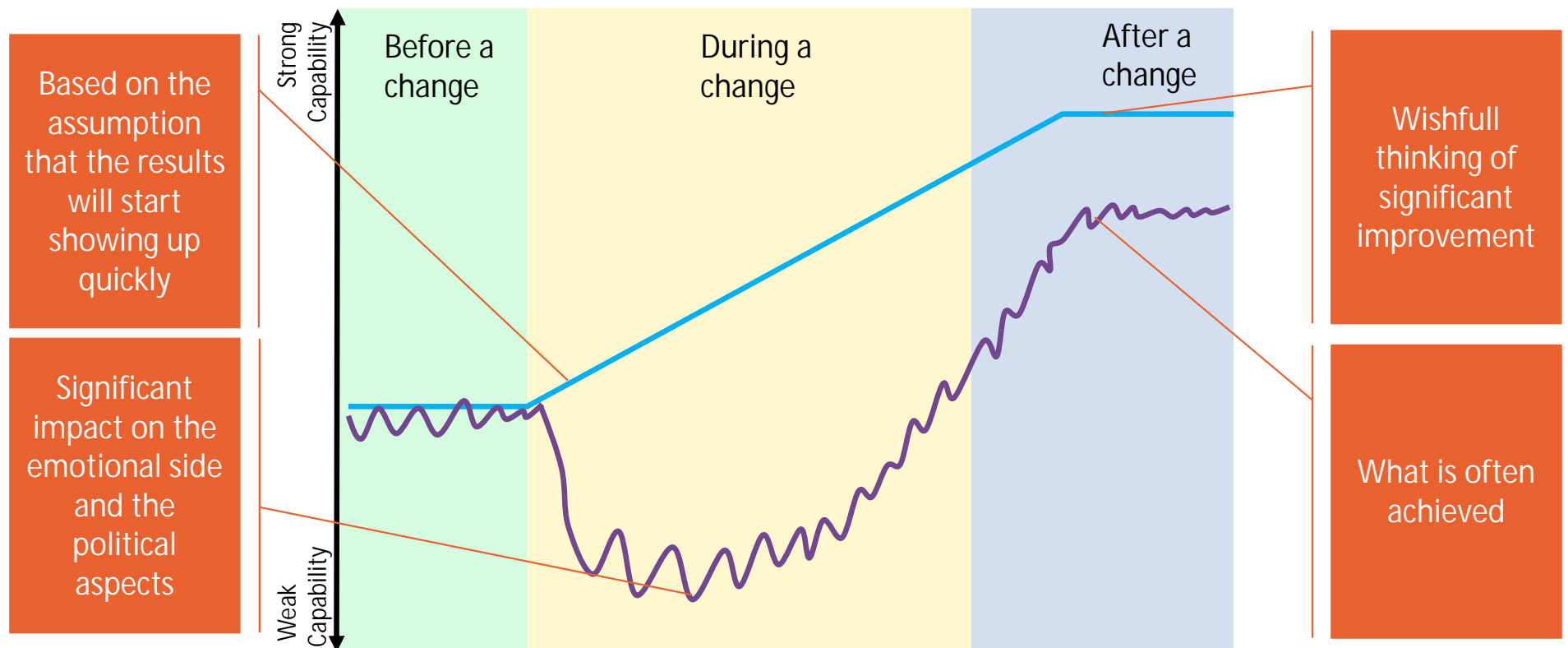
- Add Kanban principles and practices on top of your existing process or framework

Existing Process or Framework

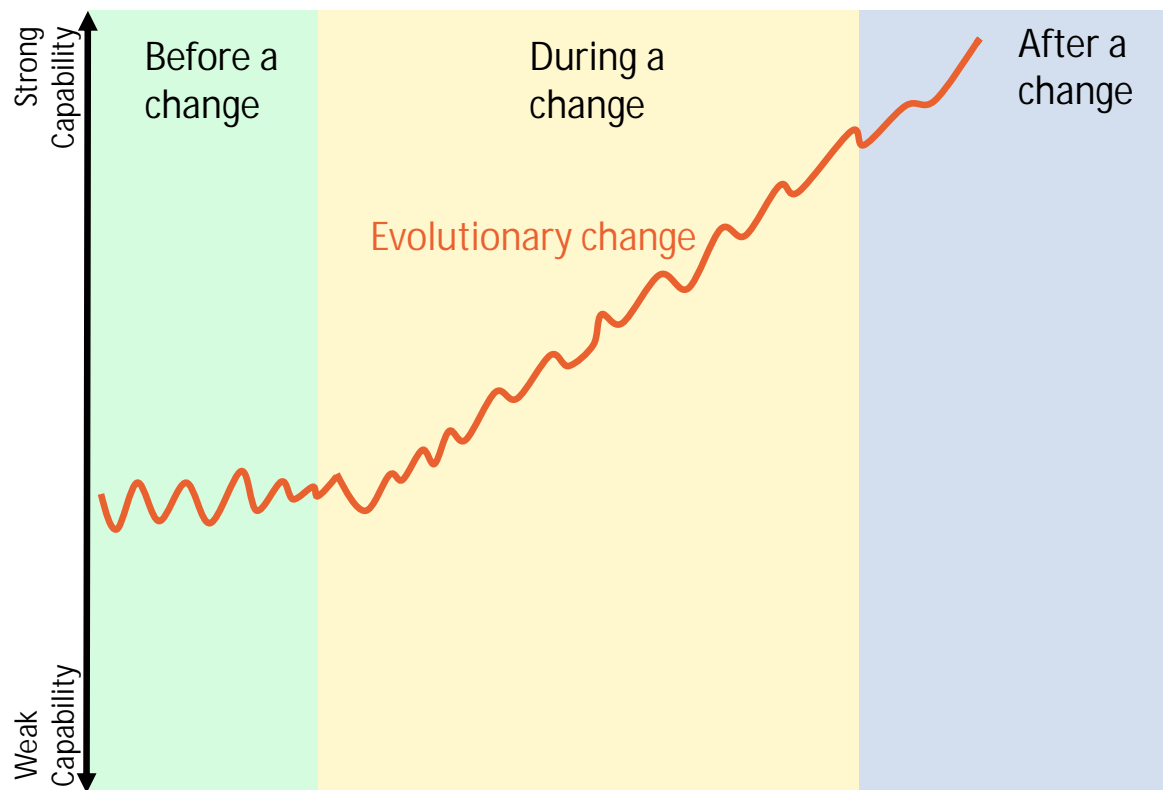


- Kanban is NOT a Process Framework!
- Kanban IS a method for managing and improving knowledge work service delivery through evolutionary change

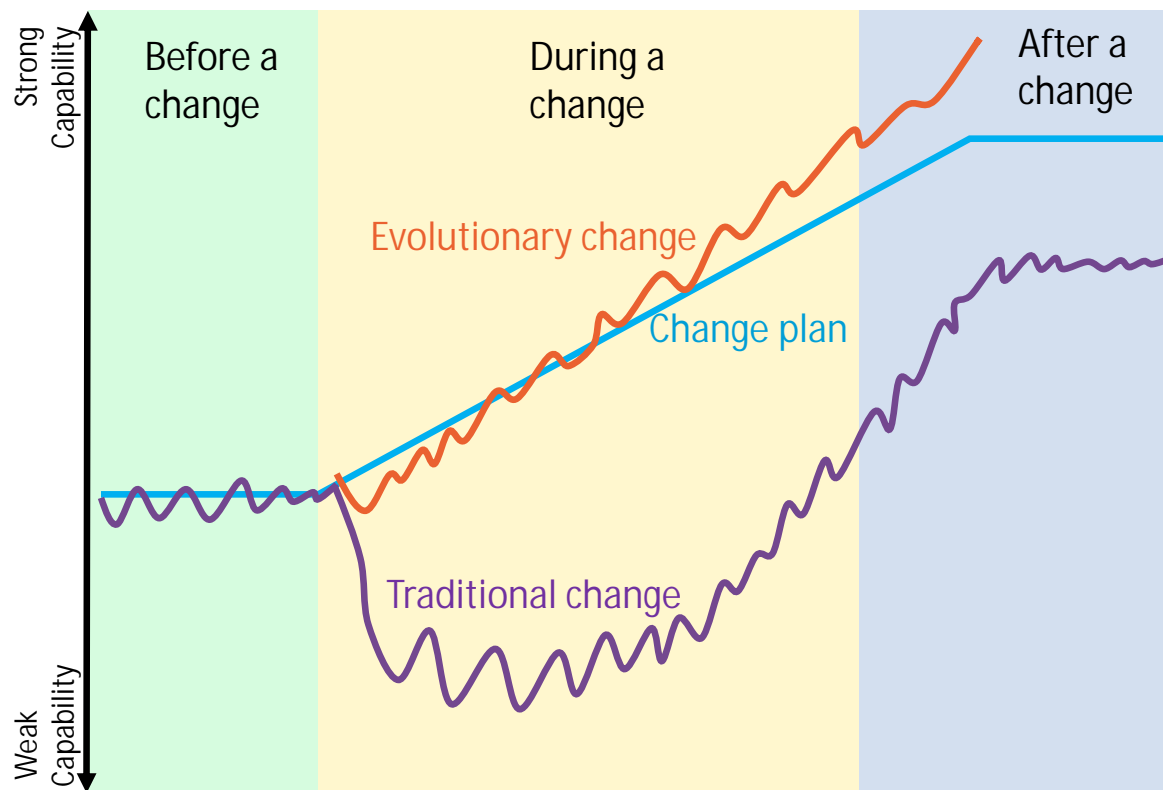
Impact of change plan on capability



Impact of evolutionary change on capability



The three different progressions of 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



“

“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.

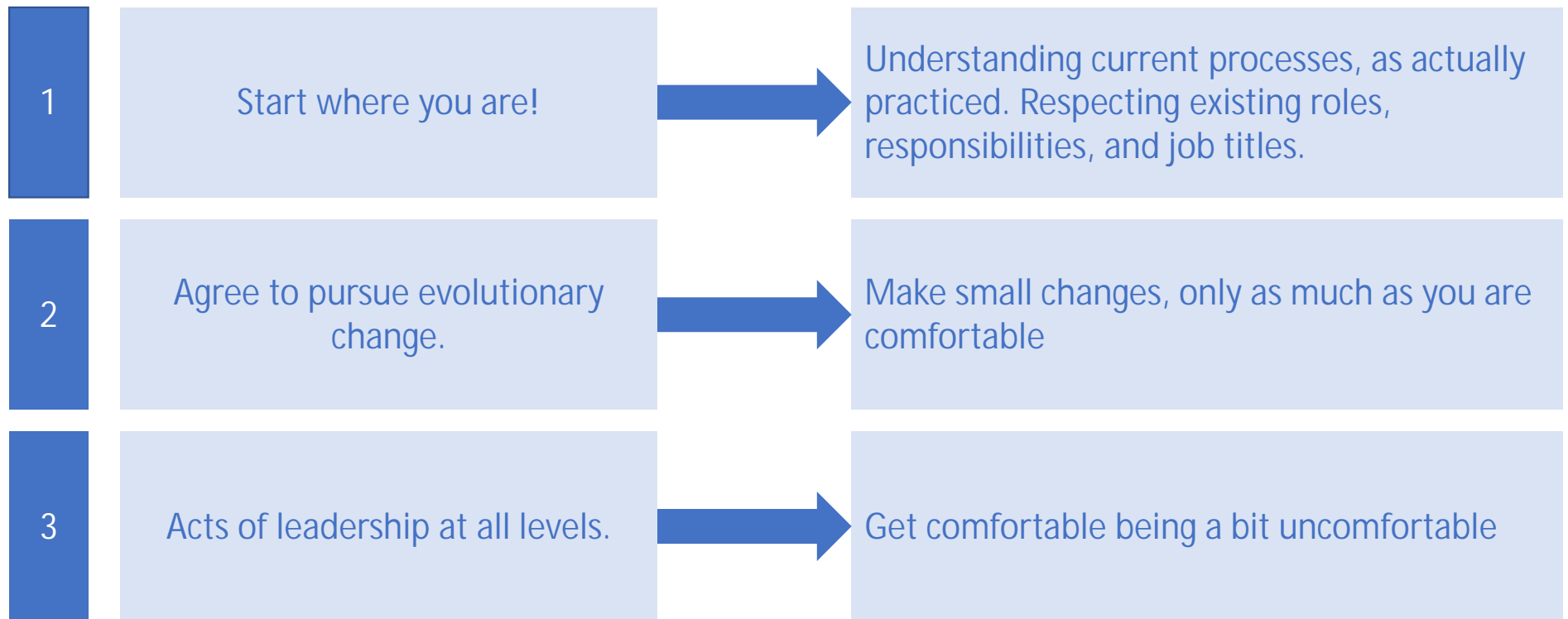
3

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 without action
is not a feedback loop.

Feedback Loop?

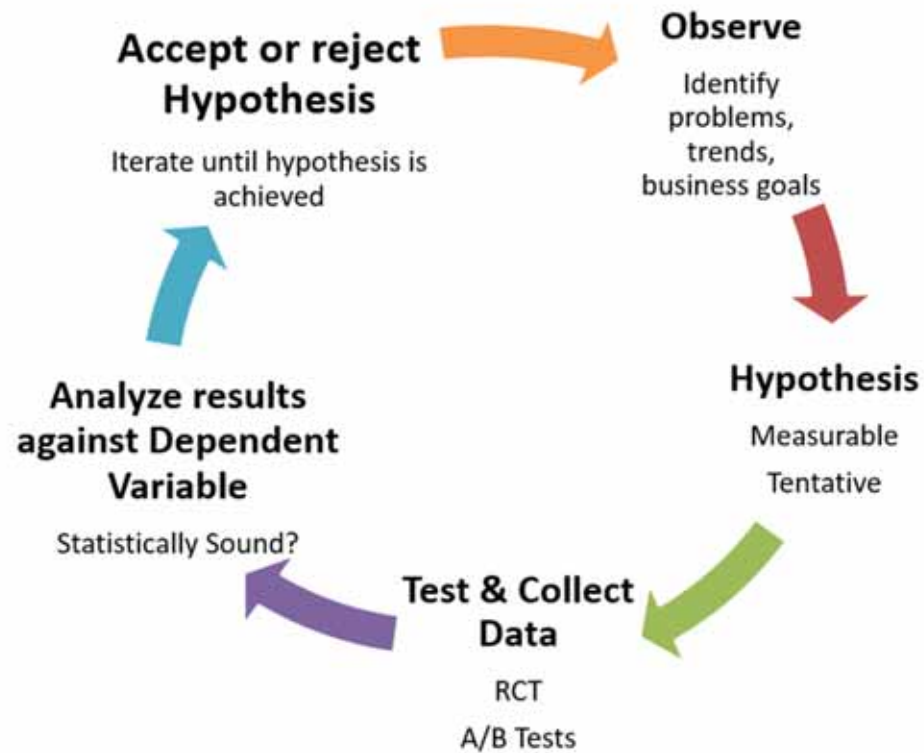


Don't leave it to chance: **MANAGED** Evolution

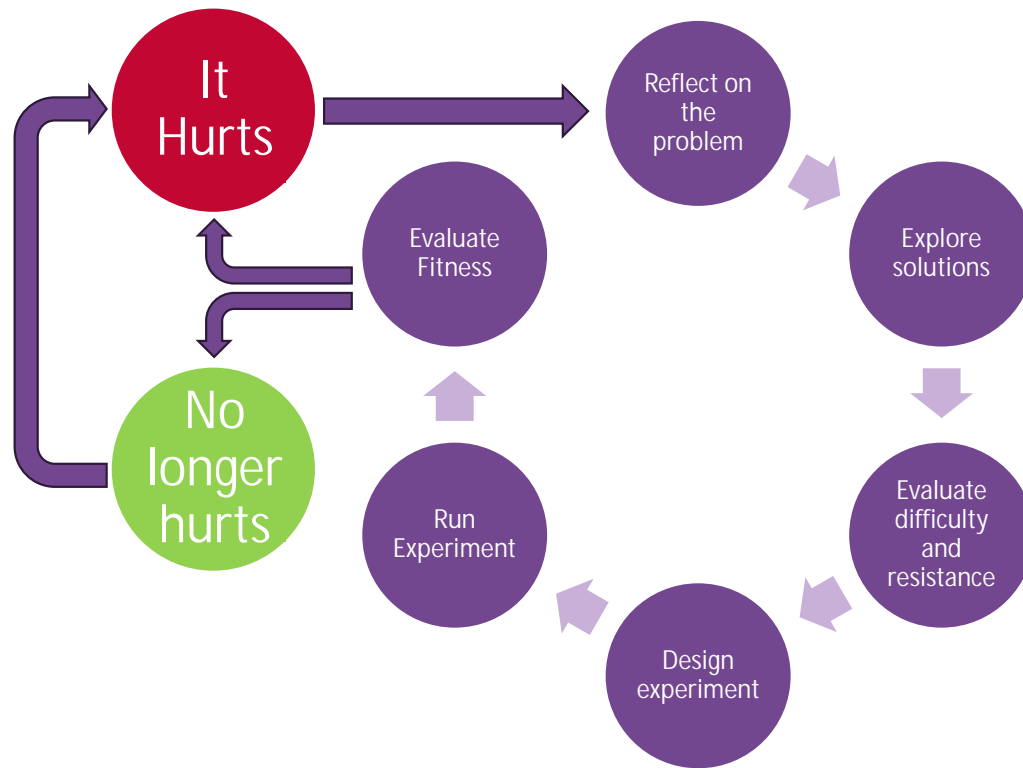
Only mutate that which isn't working well. Preserve that which is.




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!

It hurts - pain

Brainstorm pains that you have encountered. Write one pain per sticky.

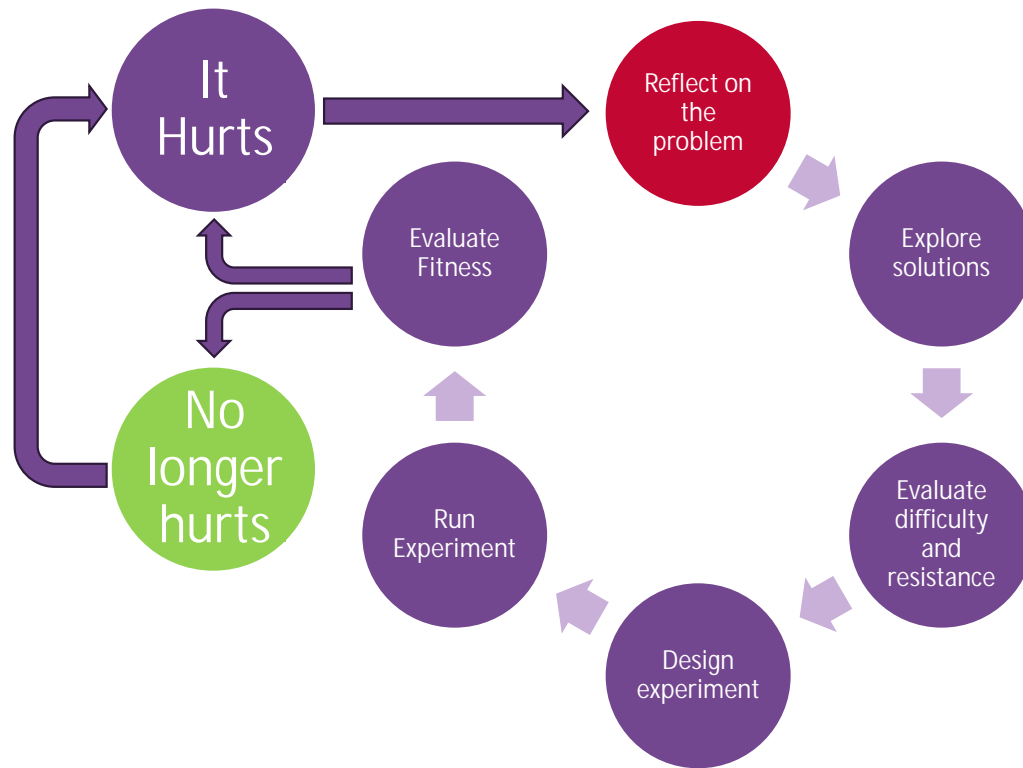
Are there additional aspects to include when describing the problem?

Is the pain clear, unique, and understandable?

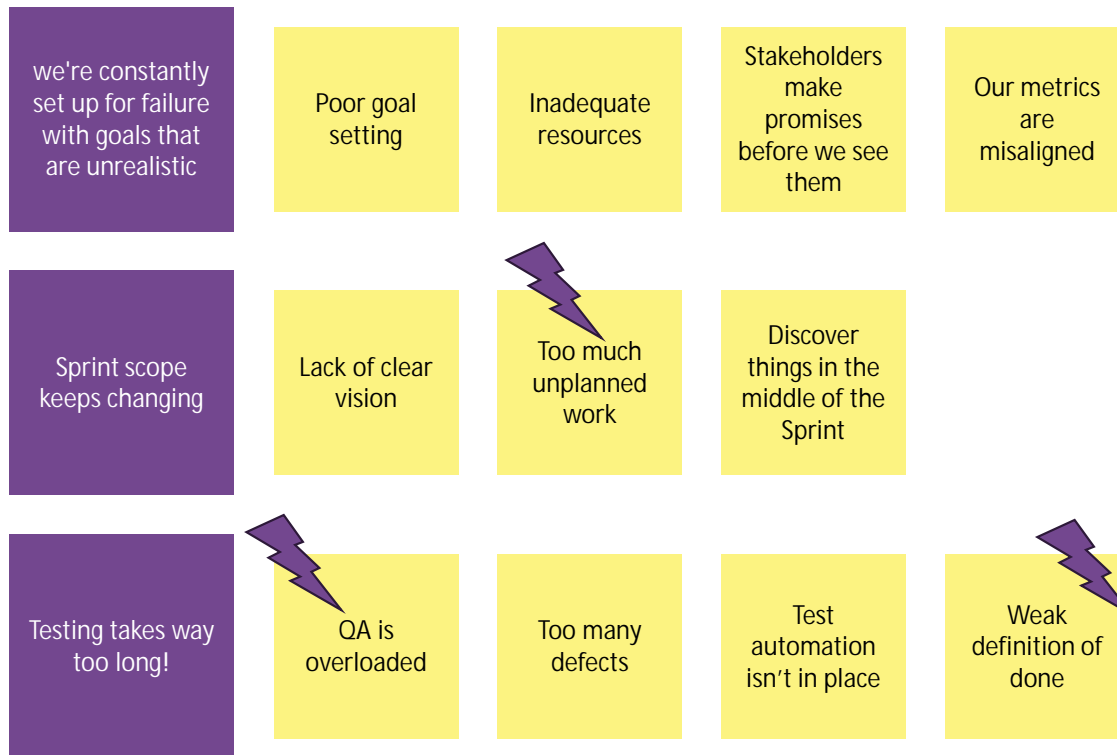
Within your group(s), identify the top 3 pains.



Improve collaboratively, evolve experimentally



Reflecting on the problems



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



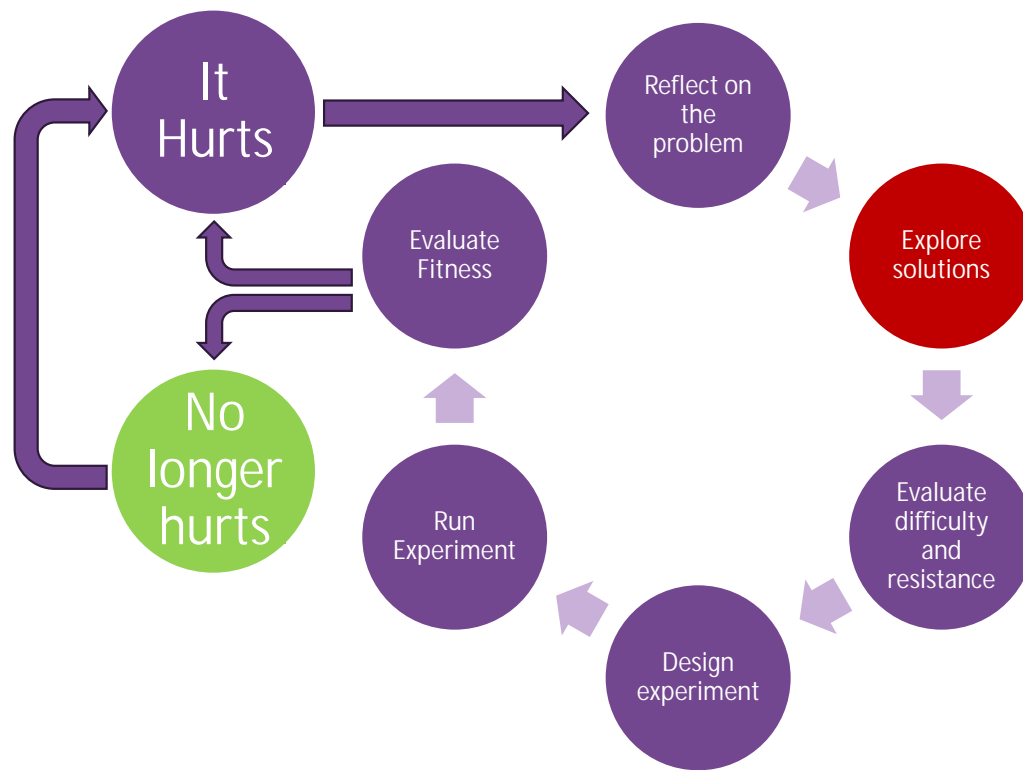
Choose 1 problem and reflect in and identify the top pains

What are possible causes of the pains that could be a real problem?

Within your group(s), identify the top 3 causes that you think might be actionable.



Improve collaboratively, evolve experimentally



Visualize

Show work and its flow.
Visualize risks.
Build a visual model that reflects how you actually work.



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



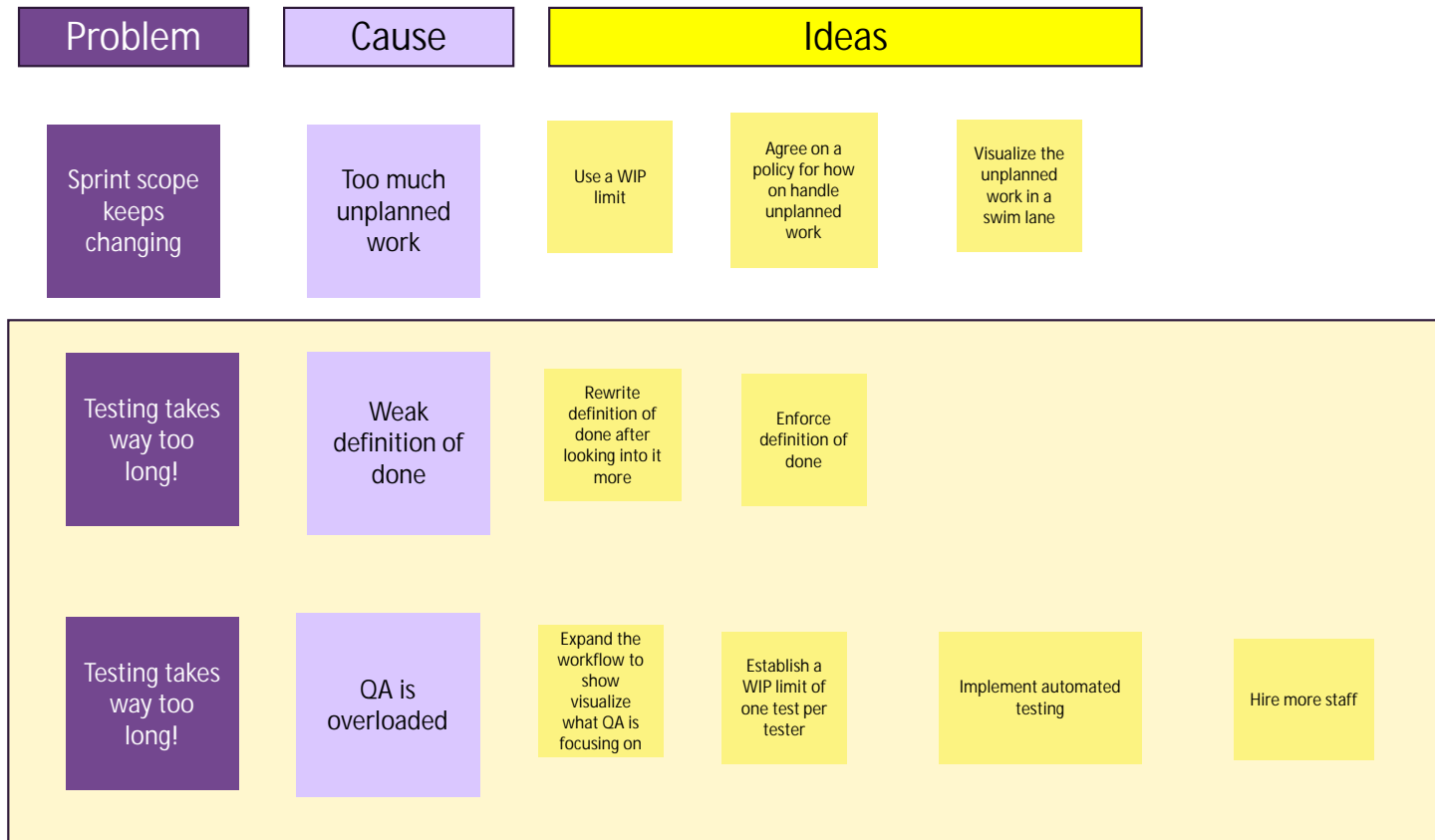
Establish feedback loops at an appropriate cadence.
Foster collaboration, learning, and improvements.
Data-driven.

Improve collaboratively, evolve experimentally

Using the scientific method.
Hypothesis-driven change.
Run safe-to-fail experiments.



The Scrum Teams' ideas



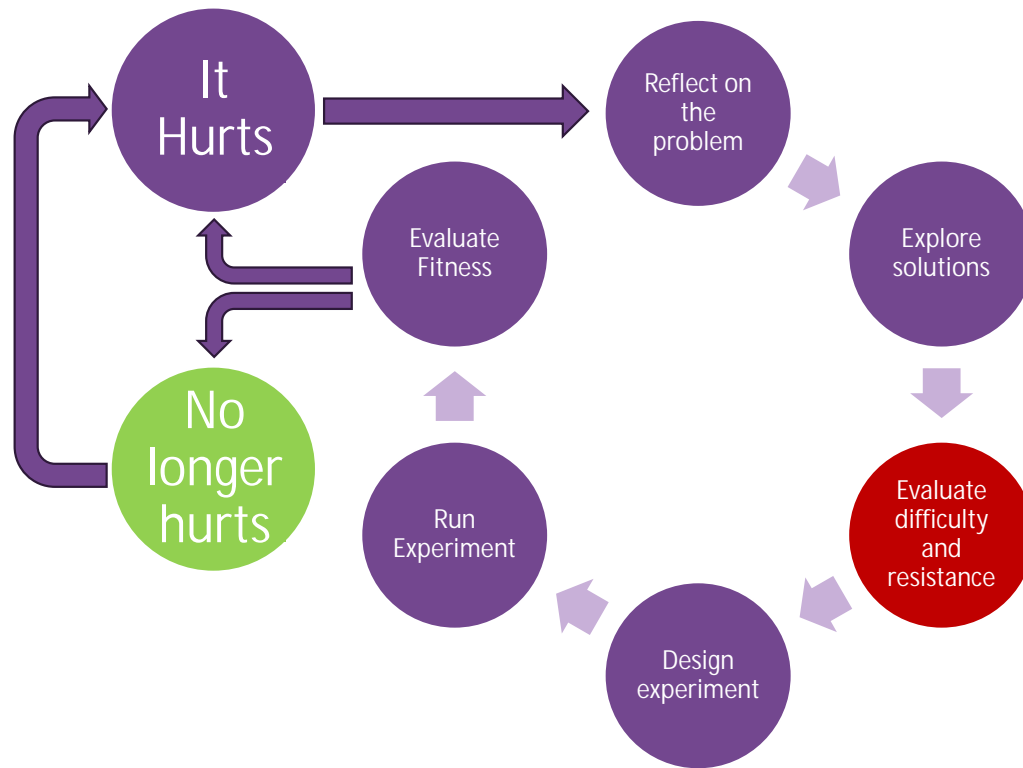
Explore ideas

Brainstorm solutions that may address your problems. One solution per sticky. As many solutions as time permits.

Within your group, identify the top 2-3 solutions per problem-cause.



Improve collaboratively, evolve experimentally





*"be like
water"*

*the rock
represents
resistance*

Water flows around the rock

Fists of Fury



Be Water my Friend



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.

“

Bruce Lee died from
drinking too much water,
new study claims

Too much water



Water runs in the family



Be Like Water – College Chem E Version

Input – Output = Accumulation



Accumulation



Water can flow



Patterns in Water



Dam it



Water can crash!



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

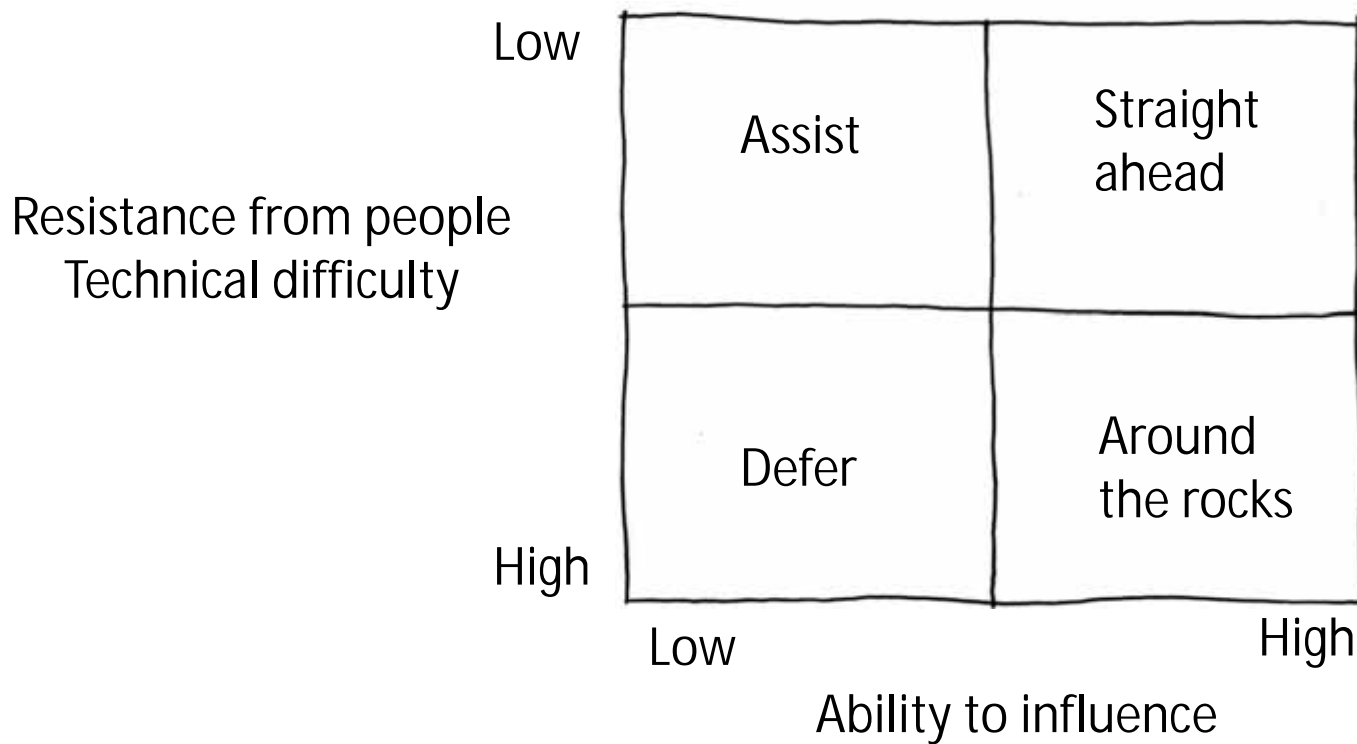
Resistance to the Implementation

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

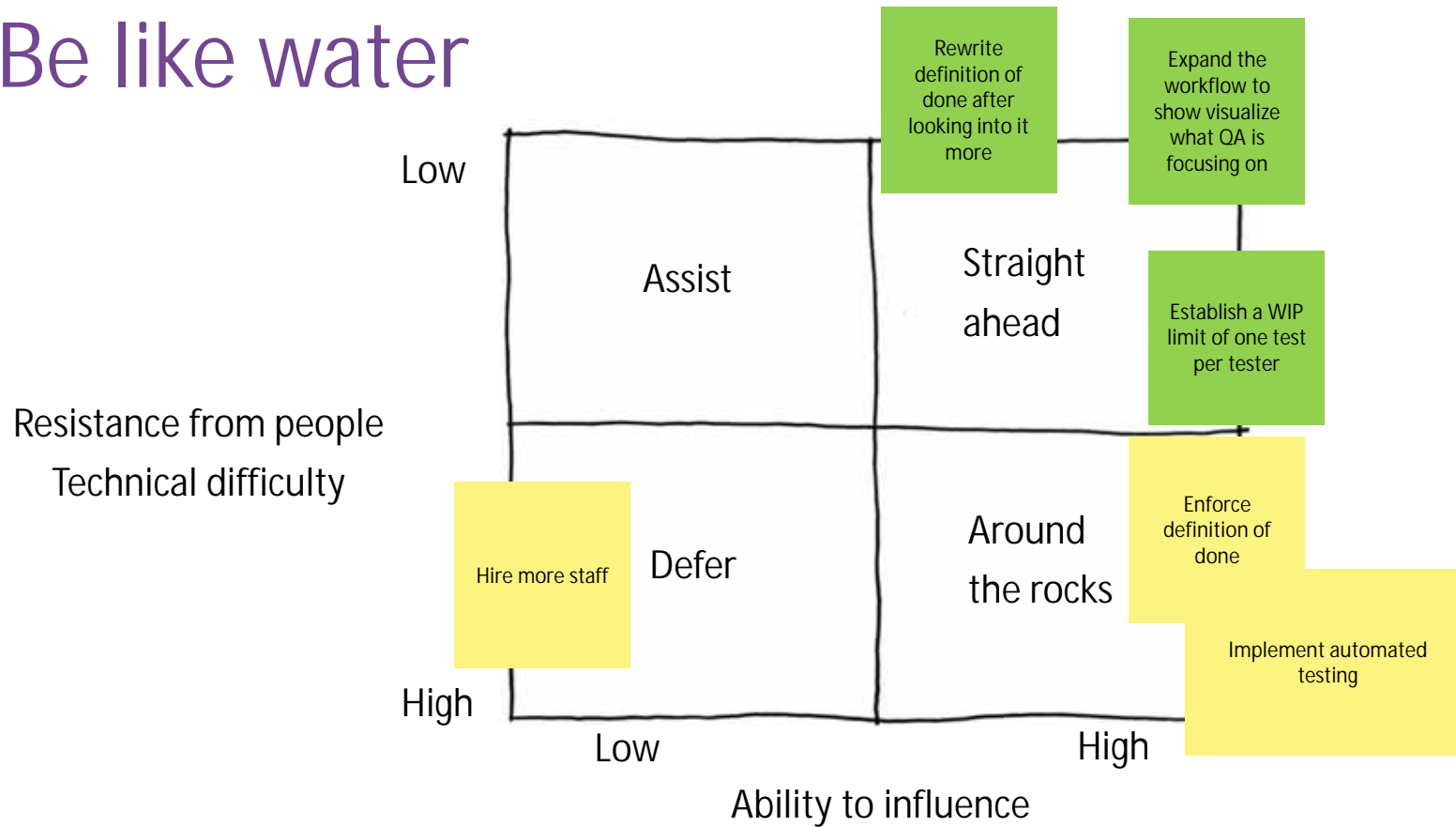
Deeper Dive into Resistance

	Disagree	Pain
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 how you propose to implement the solution, or I have a better implementation	The implementation harms me

Be like water



Be like water



Be like water

Review your ideas and ask three questions.

How likely is the idea to be resisted? How difficult is the idea, technically? How much autonomy do we have to execute the idea?

Map the ideas on the chart provided.



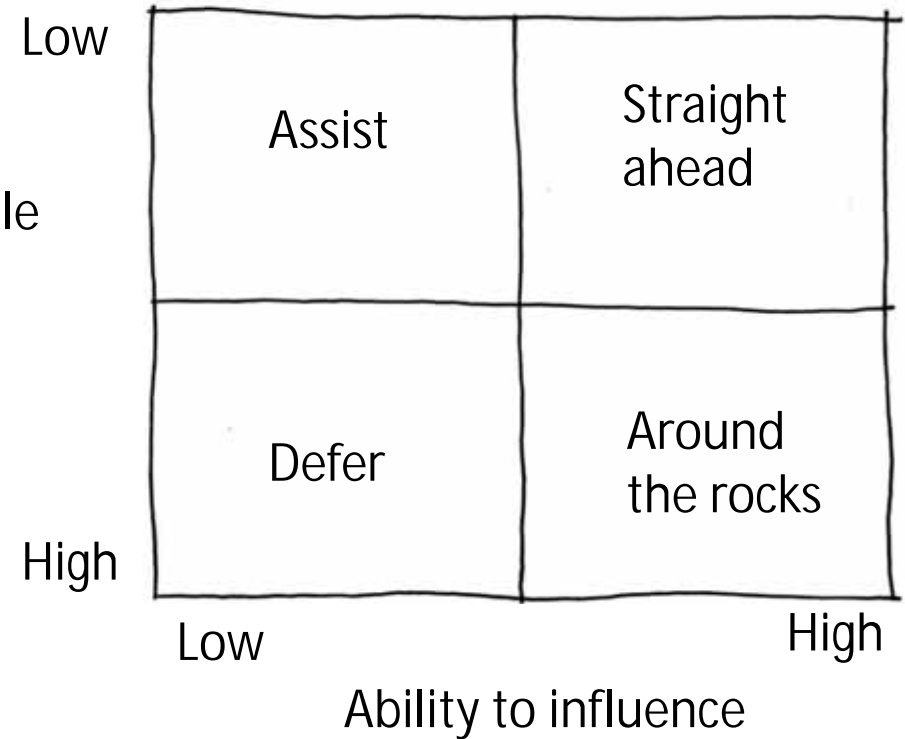
Be like water

Resistance from people
Technical difficulty

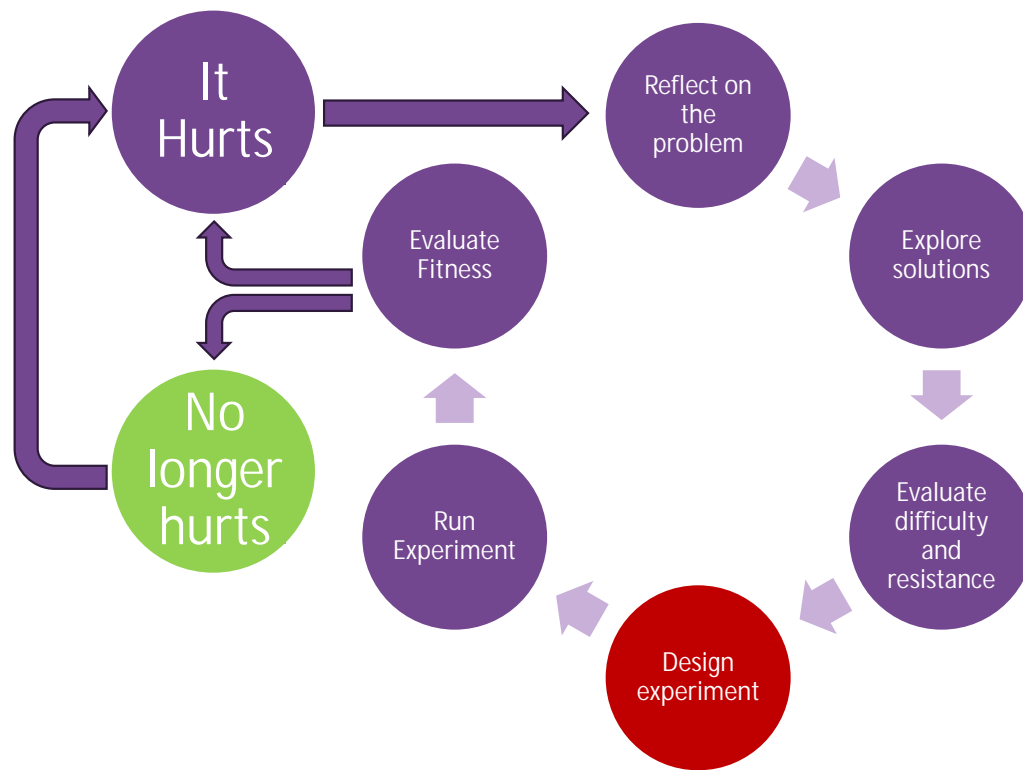
Review your ideas and ask three questions.

How likely is the idea to be resisted? How difficult is the idea, technically? How much autonomy do we have to execute the idea?

Map the ideas on the chart provided.



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!

What is your experiment?

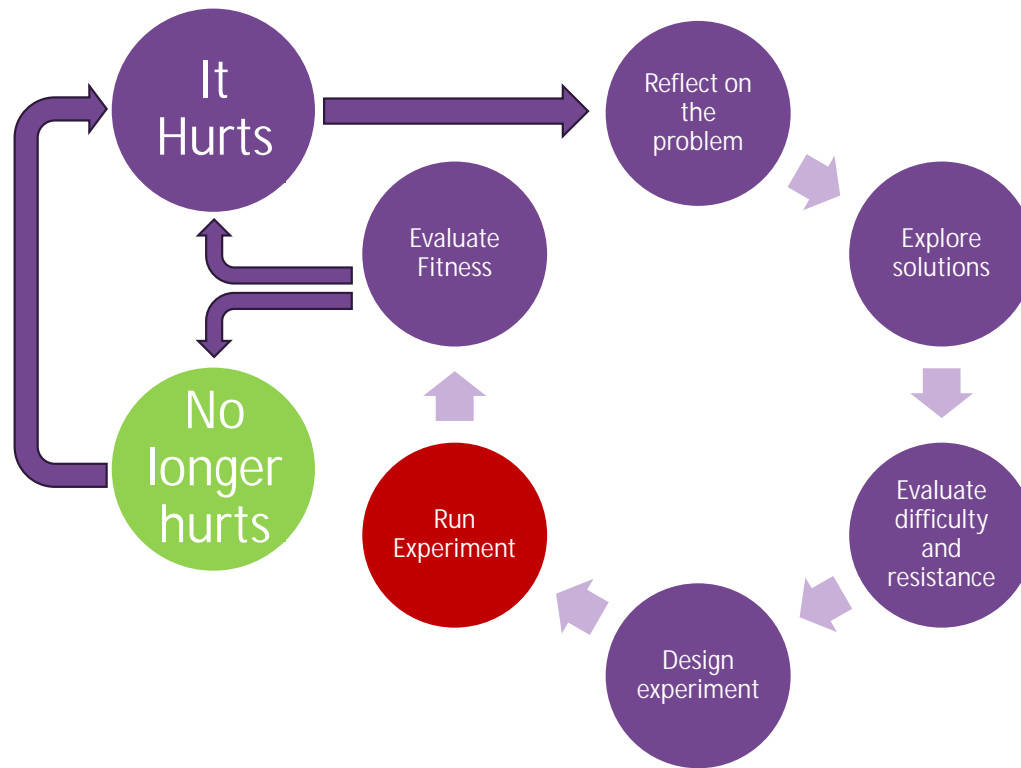
Copy your problem and ideas into the “experiments” area below.

Be prepared to explain your experiment and how you will know if it is succeeding or not.

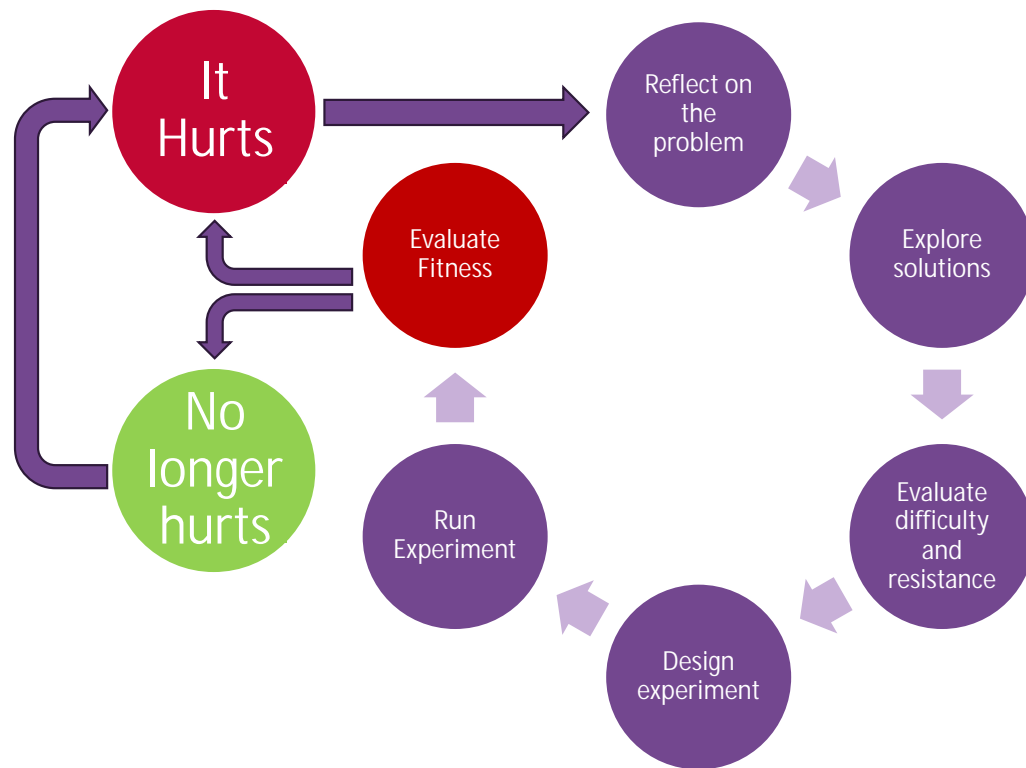
Be prepared to call out any Kanban Practices that are used.



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.

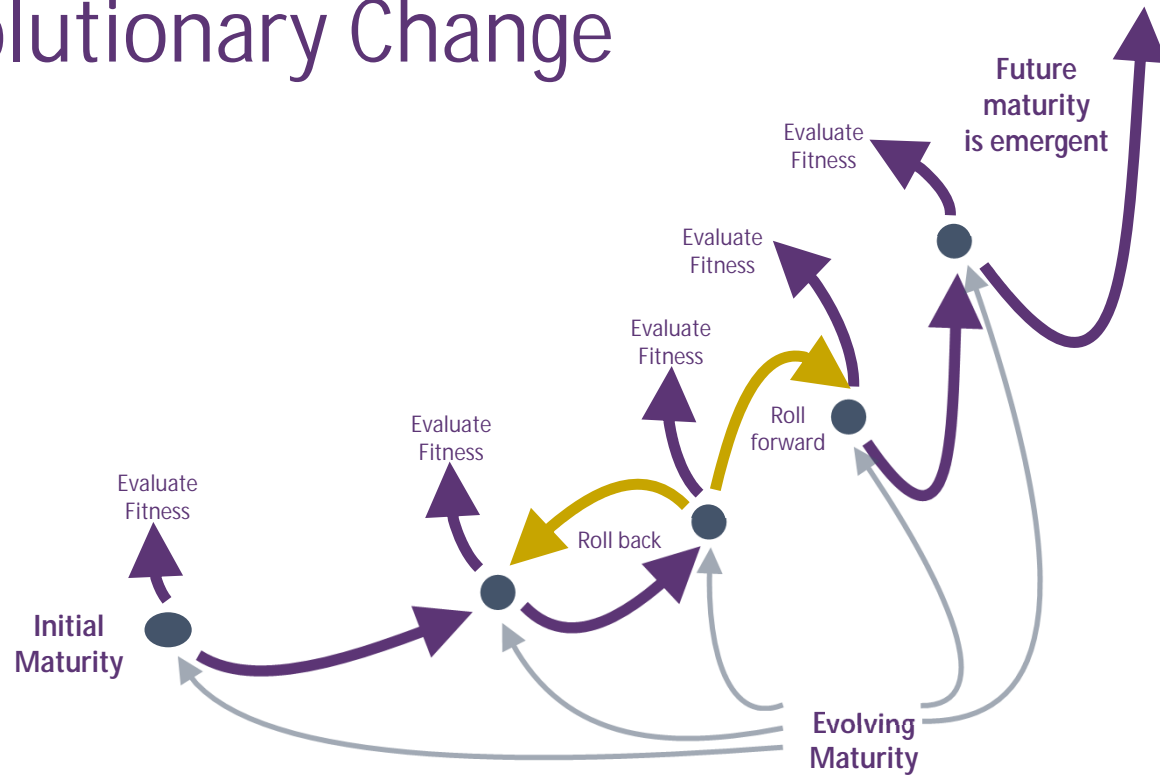
If there doesn't appear to be an obvious pain, consider taking a KMM class to further expand on evidence of hidden pains in your team and organization.

“

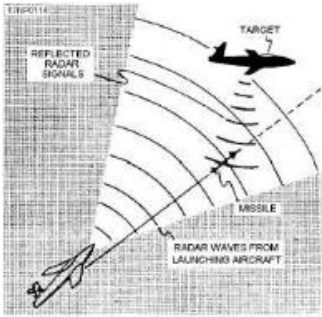
Feedback is the breakfast
of champions.

Ken Blanchard

Evolutionary Change

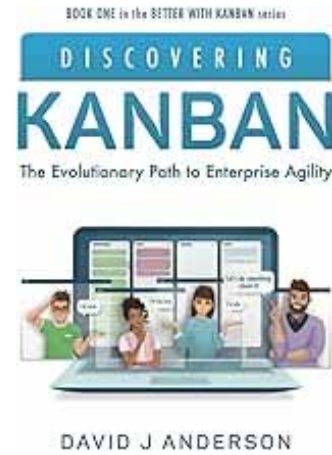
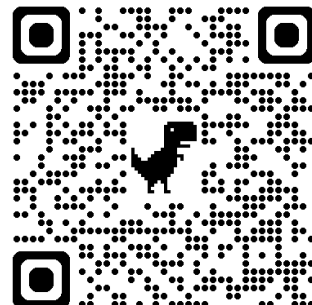


Gordon the Guided Missile





FREE!



30% off

