Facilitate Cleaning your Code (without being a Technical Coach)

@DigDeepRoots

Arlo Belshee
Principal Technical & Leadership Coach

Marian Willeke
Principal Learning Designer
Why am I doing this?

- Reduce time for each story
- Reduce bugs
- Reduce cost of future stories
- Make it easier and less frustrating for developers
How does THAT work?

Teach them to refactor code as they read it.
OK, so what is refactoring, really?

- Editing Code
- Refactoring and Testing Code
- Transforming Code
So when is the best time to refactor during story development?
Well, where do we spend all the time in story development?
When is it safe to refactor?

- Editing Code
- Refactoring and Testing Code
- Transforming Code
Let’s tackle the big number with the Read by Refactoring method.
What does RbR do?

Traditional Code Reading

New Code Reading
So now you’ll learn how to facilitate a ½ day mob that introduces RbR skills!
...and don’t forget why!

- Reduce time for each story
- Reduce bugs
- Reduce cost of future stories
- Make it easier and less frustrating for developers
Let’s establish trust.

- Authority Supported
- Legitimate Project
- Empathy for Devs
- Expected Results

- Write silently what you would do
- Discuss
- Pick what you will take home
Let’s establish your ability to do this.

• You will have recorded technical expertise to share with them at the right moments.

• You will have the flow for guiding them naturally through the habit shifts and first major mind shift of Read by Refactoring.

• You only need to know the questions that their default responses can’t solve.

...and we are here today to give you all of those resources.
Facilitation Demo

Preparing you for your first mob facilitation!
What is your plan?

Authority Supported
Legitimate Project
Empathy with Devs
Expected Results

Update your plan
Discuss
You can do this!

What are YOUR reasons that you can?
You can facilitate a mob through the Insight Loop

The team is now aware of a better way to read code

The team has a structure to change sustainably
Cleaning your Code Easier Every Day (without being a Technical Coach)

@DigDeepRoots

Arlo Belshee
Principal Technical & Leadership Coach

Marian Willeke
Principal Learning Designer
You can facilitate a mob through the Insight Loop

The team is now aware of a better way to read code

The team has a structure to change sustainably
Understanding the Insight Loop

Shifting from awareness to behavioral change

Applying the shift to the Insight Loop

Making the Insight Loop a common practice for your team
Mobbing was a sequence of tasks.

One of these Things is Not Like the Other

Tiny Commits

Honest Names

Bug-for-Bug Compatibility
<table>
<thead>
<tr>
<th>Habit Shift</th>
<th>One of these Things is Not Like the Other</th>
<th>Tiny Commits</th>
<th>Honest Names</th>
<th>Bug-for-Bug Compatibility</th>
</tr>
</thead>
<tbody>
<tr>
<td>Old Behavior</td>
<td>Recognize one chunk of a method &amp; try to hold in brain.</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>New Behavior</td>
<td>Recognize one chunk of a method &amp; record it in the code.</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Value</td>
<td>Explicitness: The chunk boundary.</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Habit Shift</td>
<td>One of these Things is Not Like the Other</td>
<td>Tiny Commits</td>
<td>Honest Names</td>
<td>Bug-for-Bug Compatibility</td>
</tr>
<tr>
<td>-------------</td>
<td>------------------------------------------</td>
<td>--------------</td>
<td>--------------</td>
<td>--------------------------</td>
</tr>
<tr>
<td>Old Behavior</td>
<td>Commit each movement.</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>New Behavior</td>
<td>Commit each safe action, merge each movement</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Value</td>
<td>Explicitness: Each commit does exactly one thing.</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Habit Shift</td>
<td>One of these Things is Not Like the Other</td>
<td>Tiny Commits</td>
<td>Honest Names</td>
<td>Bug-for-Bug Compatibility</td>
</tr>
<tr>
<td>------------</td>
<td>------------------------------------------</td>
<td>--------------</td>
<td>--------------</td>
<td>---------------------------</td>
</tr>
<tr>
<td>Old Behavior</td>
<td>Read method definition to understand it &amp; try to hold in brain.</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>New Behavior</td>
<td>Scan to understand one thing it does &amp; record it in the code.</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Value</td>
<td>Explicitness: Name is honest about what is and is not known.</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Habit Shift</td>
<td>One of these Things is Not Like the Other</td>
<td>Tiny Commits</td>
<td>Honest Names</td>
<td>Bug-for-Bug Compatibility</td>
</tr>
<tr>
<td>--------------------</td>
<td>------------------------------------------</td>
<td>--------------------------------</td>
<td>---------------------------------------------------</td>
<td>-----------------------------------------------</td>
</tr>
<tr>
<td>Old Behavior</td>
<td></td>
<td></td>
<td>Commit message states what you did.</td>
<td></td>
</tr>
<tr>
<td>New Behavior</td>
<td></td>
<td></td>
<td>Add tag indicating level of risk, then state what you did.</td>
<td></td>
</tr>
<tr>
<td>Value</td>
<td></td>
<td></td>
<td>Explicitness: Commit message states the level of risk caused by that commit.</td>
<td></td>
</tr>
</tbody>
</table>
Mindset Shift: The Insight Loop

- Insight is now much more explicit in coding.
- Making each part explicit makes the process explicit.

The Insight Loop:

1. Have a single insight
2. Write it down in the code
3. Commit explicitly
Ways to shift mindsets

Pick behaviors that don’t align or build up to a mindset shift

Pick behaviors that do align to a mindset shift, but are INTERDEPENDENT

Pick behaviors that do align to a mindset shift, but are INDEPENDENT
Understanding the Insight Loop

Shifting from awareness to behavioral change

Applying the shift to the Insight Loop

Making the Insight Loop a common practice for your team
You can now facilitate a mob for introducing and practicing the Insight Loop.

**STEP TWO**

Now we can create daily habits to make the Insight Loop a common practice.

Awareness

? 

Behavior Shift
The Path to Long Term Change

TRAINING

Awareness Increased!

*We are told. We now know something better!*

*(but don’t usually listen)*
The Path to Long Term Change

TRAINING
Awareness Increased!
We are told. We now
know something better!
(but don’t usually listen)

EXPERIENCE
Embodied Realization!
We resonate with the experience and got excited!
(but we later forgot)
The Path to Long Term Change

TRAINING
Awareness Increased!
We are told. We now know something better!
(but don't usually listen)

EXPERIENCE
Embodied Realization!
We resonate with the experience and got excited!
(but we later forgot)

DAILY PRACTICE
Do Without Thinking!
Our default has changed!
(getting a new default result)
The Path to Long Term Change

TRAINING
Awareness Increased!
We are told. We now know something better!
(but don’t usually listen)

EXPERIENCE
Embodied Realization!
We resonate with the experience and got excited!
(but we later forgot)

DAILY PRACTICE
Do Without Thinking!
Our default has changed!
(getting a new default result)
The Path to Long Term Change

- TRAINING: Awareness Increased!
  *We are told. We now know something better!*
  *(but don't usually listen)*

- EXPERIENCE: Embodied Realization!
  *We resonate with the experience and got excited!*
  *(but we later forgot)*

- REFLECTION: Motivations and Triggers!
  *Why do I really care and what reminds me?*
  *(going meta for a sec...)*

- DAILY PRACTICE: Do Without Thinking!
  *Our default has changed!*
  *(getting a new default result)*

*where change happens!*
FOR EXAMPLE
The Path to Long Term Change

TRAINING
Given data and stats of what happens with speeding

EXPERIENCE
Used cruise control as a tool to stay aware

REFLECTION

DAILY PRACTICE

where change happens!
Let’s Reflect...

What was the pain point that caused my old behaviour?

What is the trigger that reminds me there is a better way?

What is the trigger that reminds me to do the better way?

Panic that I was always going to be late.

Assurance that I would always know exactly the time I would arrive.

Putting my phone (already addiction) on dash and setting a route.
The Path to Long Term Change

TRAINING

Given data and stats of what happens with speeding

EXPERIENCE

Used cruise control as a tool to stay aware

REFLECTION

Motivation: Get there faster.

Trigger: Google Maps kept showing me the arrival time

DAILY PRACTICE

Indulge my phone addiction at start of drive by setting a route that keeps me informed
NOW you can facilitate sustained behavior change!
The Path to Long Term Change

TRAINING
You have been trained that the best way to manage is to walk around.

EXPERIENCE
You go through the steps of becoming a walking manager instead of emailing.

REFLECTION
What makes you want to email?
What reminds you there's a better way?
What reminds you to DO the better way?

DAILY PRACTICE
What existing practice are you going to use to make sure that better way is easy and daily?

where change happens!
“This matters because business results depend on what they **DO** every day instead of what they are **DESCRIBING** to you.”
Understanding the Insight Loop

Shifting from awareness to behavioral change

Applying the shift to the Insight Loop

Making the Insight Loop a common practice for your team
The Path to Long Term Change

TRAINING
- Read Arlo’s blogpost series on *Naming as a Process* and watch a short video on the Insight Loop.

EXPERIENCE
- Go through the mobbing cycles for the workshop.

REFLECTION

DAILY PRACTICE
- Devs running the insight loop as many times as you can with explicit awareness.
<table>
<thead>
<tr>
<th>Pain Point of Old Behavior</th>
<th>One of these Things is Not Like the Other</th>
<th>Tiny Commits</th>
<th>Honest Names</th>
<th>Bug-for-Bug Compatibility</th>
</tr>
</thead>
<tbody>
<tr>
<td>Pain Point of Old Behavior</td>
<td>It’s simply how they’ve learned.</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Trigger that Reminds of Better Way</td>
<td>Having a process to easily understand hard to read code.</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>What Trigger to Use as Reminder</td>
<td>When you’re looking at a large chunk of code or lots of indents.</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Daily Practice</td>
<td>Extracting as many chunks as they can that day.</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Pain Point of Old Behavior</td>
<td>Tiny Commits</td>
<td>Honest Names</td>
<td>Bug-for-Bug Compatibility</td>
<td></td>
</tr>
<tr>
<td>----------------------------</td>
<td>-------------------------------</td>
<td>---------------------------------------</td>
<td>---------------------------</td>
<td></td>
</tr>
<tr>
<td>Transaction cost of a commit is high.</td>
<td>Code review will be easier and not get lost.</td>
<td>Having just finished an extraction and naming.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Trigger that Reminds of Better Way</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>What Trigger to Use as Reminder</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Daily Practice</td>
<td>Checking in after every refactoring and maximizing the number of commits.</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Pain Point of Old Behavior</td>
<td>Tiny Commits</td>
<td>Honest Names</td>
<td>Bug-for-Bug Compatibility</td>
<td></td>
</tr>
<tr>
<td>---------------------------</td>
<td>--------------</td>
<td>--------------</td>
<td>---------------------------</td>
<td></td>
</tr>
<tr>
<td>Trigger that Reminds of Better Way</td>
<td>Following social norms (keeping it short).</td>
<td>You can now trust the names and when to do read the code for info or when to ask somebody for it.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>What Trigger to Use as Reminder</td>
<td></td>
<td>Any time you make a new name or renaming something that was confusing.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Daily Practice</td>
<td></td>
<td>Renaming anything that does not immediately tell you what it is.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>One of these Things is Not Like the Other</td>
<td>Tiny Commits</td>
<td>Honest Names</td>
<td>Bug-for-Bug Compatibility</td>
<td></td>
</tr>
<tr>
<td>------------------------------------------</td>
<td>--------------</td>
<td>--------------</td>
<td>---------------------------</td>
<td></td>
</tr>
<tr>
<td>Pain Point of Old Behavior</td>
<td></td>
<td>Proving safety is hard with large commit.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Trigger that Reminds of Better Way</td>
<td></td>
<td>You know where the risk is during code review so it’s faster and easier.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>What Trigger to Use as Reminder</td>
<td></td>
<td>You already leave log message / you’re just reflecting the new thing that happened.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Daily Practice</td>
<td></td>
<td>Practice to write a formatted log message for every single commit.</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
RESULTS

Running the insight loop as many times as you can with explicit awareness.
- Understanding the Insight Loop
- Shifting from awareness to behavioral change
- Applying the shift to the Insight Loop
- Making the Insight Loop a common practice for your team
Turn this into value!