

The
Pragmatic
Programmers

Agile Retrospectives

Second Edition

A Practical Guide
for Catalyzing Team
Learning and Improvement



Esther Derby
Diana Larsen
David Horowitz
edited by Adaobi Obi Tulton

This extract shows the online version of this title, and may contain features (such as hyperlinks and colors) that are not available in the print version.

For more information, or to purchase a paperback or ebook copy, please visit <https://www.pragprog.com>.

Copyright © The Pragmatic Programmers, LLC.

Shape the Focus

Focus implies that you will give attention to some topics and not to others, at least for now. It provides a sense of why people are investing their time. The focus might be a pattern over time, an unusual event, an improvement goal, interactions within or outside the team, the flow of work, a technical practice, or any other topic that's specific enough to provide a lens for the conversation.

More Than One Focus?



Can you have more than one focus in a retrospective? Yes, of course. We'd still advise following the same flow: attention, data, insights, and decisions. Finish one topic before you start another. Talking about multiple topics at the same time reduces the chance of making progress on any of them, and often it simply leaves people confused.

Methods to Pick a Focus

You have several ways to choose a focus. You can either choose a focus yourself as the retrospective leader or you can ask the team to choose a focus. In either case, it's best to work on topics where the team expresses energy for improving. This is true even if the topic doesn't seem like the most important topic to you.

The following table describes the various methods of picking a focus:

Who picks the focus	When	Method
The retrospective leader (you)	Before the retrospective	Propose a focus based on your own observations. Or, ask the team a day or two prior to the retrospective for ideas and make the decision yourself.
The team	Before the retrospective	Keep a running list of potential topics during the iteration. Then ask the team to vote a day or two before.
The team	During the retrospective	Have your team brainstorm a list of potential topics and ask the team to vote on the one they want to focus on. Or, pick a few candidate topics, place them on a Team Radar, and use the

Who picks the focus	When	Method
		resulting data to pick a focus. See the Team Radar activity on page ? .

All of these methods can work. We generally recommend having the team choose during the retrospective only after the team has some experience with retrospectives and has learned how to have productive discussions.

Choosing the focus at the beginning of the retrospective implies that you can't prepare up front, and it might require more facilitation skill on your part, at least until the team is in the groove and essentially self-facilitates.

Tip For Larger Groups: Discovery Questions

More preparation is needed for a larger group retrospective than for a typical team retrospective. Consider interviewing participants ahead of time to understand their perspectives as you shape the focus.



For a list of potential interview questions, see [Appendix 1, Potential Prework Questions, on page ?](#).

If you do interviews, be aware that some participants may believe that once they've written the issue down or mentioned it in a conversation or prework, it's no longer their job to bring it up. Make it clear that issues belong to the people who have them, and you're relying on them to raise topics in the retrospective.

Broad or Narrow Focus?

Retrospectives with a broad focus allow a team to explore, identify trends, and surface issues for future consideration. Those with a more narrow focus enable the team to dive deeper into a specific topic or concern. Either can be valuable, and some teams alternate between the two.

In general, start with a narrow focus when tangible progress is both necessary and possible until possibilities for progress are exhausted (or people need a break), then go broad. You will probably discover something new to focus on!

Writing a Useful Focus

Once you have a potential focus, ask yourself whether the focus follows these four criteria:

1. *Explore the issue with a systems mindset*, rather than restrict thinking to a single part of the problem.

2. *Stay curious and open to learning*, rather than blame a person or a group.
3. *Consider all possible causes and solutions*, rather than assume the source of the problem.
4. *Focus on a small enough topic such that action becomes likely*, rather than broaden the lens so wide that focus is difficult.

To better understand how to apply these criteria in the real world, let's look at some examples. For each example, we'll describe the scenario, share the team's retrospective focus, and describe how the focus could be improved.

Example 1: A Team That Struggled with Testing

One team was working for a while on integrating more tests into its development process. Despite their focus on testing, the team hadn't yet seen a positive impact on the quality or efficiency of its work. When it came time for the team's next retrospective, the retrospective leader picked the following focus:

Determine what we are doing wrong with testing

What are the issues with the way this focus is written?

First, this focus asks the team to look exclusively at itself ("what *we* did wrong"), in contrast with criteria 1 which promotes systems thinking.

Second, this focus biases the team toward looking for blame since it asks what the team "did wrong," whereas criteria 2 encourages openness. Perhaps no one did anything wrong at all!

We'd suggest rewriting the focus as follows:

Find ways to improve our testing practices

This version of the focus is blame-free and helps the team to broaden its perspective.

Example 2: A Team That Missed Its Iteration Goals

For the past few iterations, this team had been missing its goals. The team decided to focus its next retrospective on this problem. The team wrote its focus as:

Determine how to meet iteration goals

What are the issues with the way this focus is written?

This focus assumes the team *should have* met its iteration goals in the first place. Perhaps...but perhaps not! In contrast, criteria 3 asks us to consider all possibilities.

We'd suggest rewriting the focus as follows:

Understand the reasons behind missed iteration goals

Notice how this focus primes the team to have a systems mindset. Could the issue be the work itself? Could it be how the team is setting its goals? Or could it be something about the way work is structured in the organization more broadly? The team doesn't know ahead of time, and that's the point.

Example 3: A Team Whose Stories Were Getting Rejected

Another team had a recurring issue that was causing plenty of frustration: towards the end of many iterations, the product manager would reject the work being done with feedback such as, "The quality just isn't there!" A lot of time, energy, and effort was being wasted on work that would never be released, and the team was growing resentful. The retrospective leader wrote down this focus:

Improve the quality of our work

What are the issues with the way this focus is written?

This focus is so broad it's unlikely to lead to learning and action, while criteria 4 encourages us to focus on smaller topics. "Improving quality" is an admirable goal, but it presents too wide of a lens to enable a focused conversation.

We'd suggest rewriting the focus as follows:

Increase our understanding of the product manager's needs

Notice how this rewritten topic encourages the team to have a focused conversation. At the same time, it's not so restrictive that it discourages curiosity and creativity.