Look Back in Agile: The Sprint Retrospective

By Alan Bustamante, Senior Agile Consultant
Contents

The End of the Trail .................................................. 2
Why a Sprint Retrospective?...................................... 3
The Last Inspection..................................................... 4
Who’s Involved? ....................................................... 5
How Long Should the Retrospective Last? ................. 6
Facilitating the Retrospective ................................. 7
Conducting the Retrospective ................................. 8
Plan to Play ........................................................... 11
Carry It Forward ..................................................... 13
Retrospective Challenges ....................................... 14
The Sprint Retrospective in a Nutshell ..................... 16
The End of the Trail

In the words of a popular GPS navigation system, “You have reached your destination.” You’re at the end of the trail, standing on the big X on the map. The sprint is complete, the product has been demonstrated in the sprint review, and everyone can relax, right?

Not exactly.

Even though the sprint review takes place on the last day of each sprint, the sprint is not closed until the team has held their sprint retrospective. (While we could call it the “lessons learned” or “postmortem,” the term most often associated with Agile in general, and Scrum in particular, is “retrospective.”)

The sprint retrospective is usually held after the sprint review, allowing the team to reflect on the entire sprint, including the review. However, sometimes scheduling conflicts make it necessary to have the retrospective before the sprint review.

**DANGER!** Regardless of which way your team decides to order the retrospective and sprint review, they should occur on the same day to keep the team’s cadence. Completing both on the same day allows the team to mentally close the sprint and focus on what’s coming next.
Why a Sprint Retrospective?

Traditionally, most projects have included some sort of retrospective at the end of a project. While well intentioned, reviewing lessons learned at the end of a project can have several disadvantages:

1. **It’s too late to change anything.** The project has been completed, for better or worse. All you can do is capture the lessons learned, write the wrap-up report, and file it away. If you’re lucky, future project teams might benefit from your findings, but it’s too late to help your project.

2. **There are large memory gaps.** Team members have a hard time remembering what they did last week, let alone what they did over the past year or more. Waiting until the end of a project often results in large gaps between what the team remembers and what actually happened.

3. **Energy levels are low.** Everyone knows the disadvantages already described, and this drains team members of their enthusiasm for a retrospective conducted at the end of the project.

By contrast, retrospectives conducted at the end of every sprint give the team an opportunity to reconnect and improve the way the project is delivered.
The Last Inspection

For Agile teams, the retrospective is the last in a long line of inspections that occur during a sprint. Unlike the other inspections, the retrospective provides the best opportunity for the team to identify ways to adapt and improve their Agile processes going forward. Following are the main advantages of the retrospective:

1. **There’s still time to make a difference.** What you discover in the sprint retrospective will be used in future sprints, making a positive impact on the project team, the customer, and product quality.

2. **You have almost total recall.** Team members only need to remember things that happened over a period of weeks instead of months or years, which means the gap between what they remember and what actually happened will be small.

3. **Energy levels are high.** Teams are engaged and energized because they know the actions coming out of the retrospective will be implemented. In other words, they know the time is well spent and not wasted on a fruitless exercise.
Who’s Involved?

At a minimum, the retrospective should include the core team (the Product Owner, development team, and Scrum Master) and the supporting teams—people who were involved in fulfilling sprint objectives, but not assigned to the entire project.

Optional participants include other stakeholders, such as managers.

DANGER! Many teams new to Agile only include the delivery team in the retrospective. Project stakeholders should be given the option to attend because they are often consumers of the information that comes out of the sprint, such as metrics and reports. For example, if there is a problem with the way the burn down chart is structured, stakeholders will want to use the retrospective to discuss the issue and identify the actions necessary to change the chart.
How Long Should the Retrospective Last?

As a rule of thumb, a retrospective for a 30-day sprint should last about four hours. However, the retrospective should be tailored to the needs of the project and the experience of the team. For teams new to Agile, the sprint retrospective will initially take longer but will gradually take less time as they get used to the activity. Alternatively, as they start to realize the benefits of the retrospective, the time may actually increase because the team will be more engaged in subsequent retrospectives.

DANGER! The more participants there are in the sprint retrospective, the more time the team will need to gather and process the information generated during the meeting. This can cause problems if there are time constraints—an inefficient retrospective that does not produce actionable results will yield little value. You don’t want your team to feel like the sprint retrospective was a waste of time. If time is an issue, consider splitting up-in to groups and then coming back together to share ideas.
Facilitating the Retrospective

If your team is new to Agile, have the Scrum Master facilitate the retrospective because they typically have the most retrospective experience. As the team becomes familiar with the methods used to conduct a sprint retrospective, other team members should have the opportunity to facilitate the retrospective. This encourages buy-in from the team, which results in a more productive retrospective.
In *Agile Retrospectives: Making Good Teams Great*, Esther Derby and Diana Larsen identify several steps the facilitator should follow when conducting a retrospective. The basic steps are:

1. Set the stage  
2. Gather data  
3. Generate insights  
4. Decide what to do  
5. Close the retrospective

Retrospectives should be fun and informative, and they don’t always need to take a lot of time to put together. Often, just having candy in the room to keep the blood sugar up, a white board, markers, some sticky notes, and the ability to facilitate a good conversation is enough to get the job done and make it worthwhile for everyone.

For example, take a look at Figure 6: Retrospective results.
<table>
<thead>
<tr>
<th>What Worked Well for Us</th>
<th>What Did Not Work Well for Us</th>
<th>Actions for Improvement</th>
</tr>
</thead>
<tbody>
<tr>
<td>Unit Test Coverage</td>
<td>Engaged Architecture too Late</td>
<td>Everyone pair programs at least 1 day out of the week</td>
</tr>
<tr>
<td>Product Owner Very Responsive</td>
<td>Remote team members felt disconnected</td>
<td>Research TDD Practices and Give 1 hour report to team</td>
</tr>
<tr>
<td>Security Audit Passed!</td>
<td>Story/Defect Traceability</td>
<td>Introduce Projector At Daily Stand-up</td>
</tr>
<tr>
<td>All User Stories Accepted</td>
<td>Dependent Teams not notified about changes</td>
<td>Meet with Auditor by Day 2 of Sprint</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Include IT Security in Daily Stand-up</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Integrate Builds every other day</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Upgrade the Build Server to V2.5</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Start using burn up chart</td>
</tr>
</tbody>
</table>

Figure 6: Retrospective results
While sticky notes are used in Figure 6, a white board could easily be used to track the following:

1. What worked well in the sprint? List the processes, interactions, and events that the team found helpful and would like to continue.

2. What didn’t work well in the sprint? List the delays, impediments, and broken processes that the team would like to either improve or discontinue.

3. What actions can we take to improve our process going forward? List the actions that volunteers from the team (including the Scrum Master and Product Owner) agree to see through to completion in future sprints. In rare cases, actions may be picked up by the team as a whole.

**DANGER!** It's not enough to only capture what worked well and what didn't. Actions for improvement must be captured too. It's the only way to make sure improvements are rolled into future sprints and releases. Remember Agile teams don't just inspect—they also adapt!
Plan to Play

Games can also be used to foster communication and help identify issues during a retrospective, but they usually require advanced planning. One such game is Speed Boat from Innovation Games. (To learn more about playing Speed Boat, visit http://innovationgames.com/speed-boat.)

In this game, the speed boat represents the project and the items represent activities in the project. The placement of the sticky notes also has meaning:

1. Above the waterline-these sticky notes represent wind, which are items that are propelling the project forward. In other words, they are the wind in the sails. (We realize that speed boats don’t have sails, but go with it.) The farther ahead of the sail, the stronger the item.

2. Below the waterline-these sticky notes represent anchors, which are the impediments and other items holding the project back. The deeper the anchor, the heavier the item.

Notice that actions have not been identified. While items representing wind might lead to future actions, anchors are generally where the team should focus their time. Lead the team in a discussion about the specific actions that can be generated from these anchors.
Note: In the variation of Speed Boat played in the video, the participants concerned themselves only with the anchors to focus on the impediments that need to be fixed. Because retrospectives should also capture the things that went well, however, it is better to use the version in our example.

DANGER! Depending on your corporate culture, the word “game” should be used cautiously. While these games produce a useful outcome for the organization, the word “game” can derail a well-intentioned and valid activity. Make sure senior management and stakeholders understand the benefits of the game beforehand, or use a different word to describe the activity.
The team decides which improvements and actions they will carry forward into future sprints based on the lessons learned in the retrospective. It’s good to open subsequent retrospectives with a review of how the actions from the previous retrospective played out.

If actions require more than an hour of time to complete, the team member who volunteered for the action should carry it with them into sprint planning (which should be the next day) and then add it to the task board in the next sprint.

**DANGER!** Don’t be discouraged if actions that were carried forward from a previous retrospective aren’t successful. The actions are often experiments to see if a certain process will work better than an existing process, and all experiments are not a success. A team may also identify an item as something that was done well in one sprint, yet mark the same item as something that was done poorly in a subsequent sprint. It’s OK when this happens, but if the same item seesaws regularly, dig a little deeper to figure out the cause of this fluctuation.
Retrospective Challenges

Again, retrospectives should be fun. Teams new to Agile will often struggle in the beginning when it comes to participating in retrospectives. The reasons for this struggle are varied, but a few of the possible culprits include the following:

- **Fear of blaming.** If a story wasn’t completed in the sprint or something happened in a team member’s area that delayed the rest of the team, there may be a fear that the retrospective will result in blaming. To combat this, make sure you focus the team’s attention on the group goals and on continuous improvement as a team. Work with team members who might cause problems before the retrospective to understand their issues.

- **Fear of wasting time.** If a team member thinks the retrospective is a waste of time, try to find out why. Maybe they didn’t benefit from the actions (or lack of actions) coming out of the previous retrospective, or the team member might be overloaded with other projects and is frustrated at the thought of spending time looking backward.
• **Fear of speaking up.** Team members who are unsure of what to say or are shy will refrain from speaking. Handle this by introducing a speaking token that gets passed from team member to team member, but make it OK for team members to skip their turn if they have nothing to say. If a team member is unusually quiet over several sprints, speak with them individually to find out why.

---

**DANGER!** Don’t put quiet members on the spot in the retrospective. Singling out quieter team members may only intensify their unwillingness to talk, and may come off as an attempt to lay blame. Instead, talk to the person outside of the retrospective, one on one.
The Sprint Retrospective in a Nutshell

The sprint retrospective is held after the sprint review, allowing the team to reflect on the entire sprint, including the review. Here’s what we learned:

- Even though the sprint review occurs on the last day, the sprint is not closed until the team has held their retrospective.

- Retrospectives conducted at the end of every sprint provide an opportunity for the team to reconnect and improve the way the project is delivered.

- The retrospective is the last in a long line of inspections that occur during a sprint, and it provides the best opportunity to identify ways to adapt the Agile process going forward.

- At a minimum, the Product Owner, development team, and Scrum Master should participate in the retrospective. Supporting teams and other stakeholders may also be included.

- Retrospectives are usually four hours long for a 30-day sprint, but should be tailored to the needs of the project and team.

- The Scrum Master typically facilitates the retrospective in the beginning, but other team members are welcome to take on the role of facilitator.

- Games can help make the retrospective meeting be more productive, interactive, and fun.