A Horizontal Slice Versus A Vertical Slice: Which is better?

The difference between horizontal versus vertical slicing is trivial for some people. Some believe that delivering perfect engineered masterpiece is the best option. Others recognize the sacrifice of essential functionality. At times, a Product Owner will work closely with the engineering team. Other times, do it alone. They must break down user stories and prioritize them. A common mistake is to horizontal slice instead of vertical slice. Your ability to be transparent will decrease. A long-term impact on teamwork, innovation, and efficiency is very possible.

As covered in Part 3: Why Split User Stories, Anyway?, there are a variety of reasons to split user stories. Hence, a team must understand the difference between horizontal and vertical slice.

What Is A Horizontal Slice?

A horizontal slice is a sliver of work. It is a breakdown of the technical layers, components, tasks, and type of work. It is like slicing a cake horizontally, which does not allow for the perfect bite. Sometimes, the engineering team feels that it is the most logical thing to do or they run out of viable choices. From their perspective, it could be the most efficient way to get things done. Horizontal splitting prioritizes work over the user.

Here are a few reasons why horizontal slices are inefficient.

- Our goal is to deliver working product early and often. Yes, horizontal splits break work down into increments. Yet, we leave the ability to prioritize functionality on the cutting room floor. Demonstrable progress and feedback go out the window. Be mindful of splitting functionality instead of engineering work.

- The impact on high performing team is evident. They possess all the skills needed to achieve success. A horizontal slice isolates the work, which will decrease collaboration and self-organization. Why do you ask? The focus changes from end-to-end functionality, which requires the skill set of everyone. It requires the skillset a few to get the horizontal slice done.
Yes, the designer can get straight to designing. They might deliver a steady stream of quality designs. The independent designer becomes a false bottleneck holding up everyone else. Work becomes a constant set of handoffs and interfaces with others team members. Be careful of this very common outcome.

- Furthermore, horizontal slices leave an all or nothing mentality. In addition, Pareto’s Law states that the majority of the value is a result of a small amount of the work. It is common that not all requirements get completed. A horizontal slice paralyzes your Product Owner. While everyone is working, the Product Owner aimlessly waits. And wait! And wait until any functionality is complete. Again, be careful.

- Remember, horizontal slices eliminate the need for the innovation and the teamwork. The product owner’s domain knowledge is a well of information that is valuable to the entire team. By working together under a shared vision, the end user is better served.

**What Is A Vertical Splice?**

Understanding that horizontal slicing will never completely die. Remember the goal is not to split into technical layers or tasks, but by improving user function. A vertical slice requires the team to cut items based on value to the user. The slicing a cake analogy becomes appropriate. Each slice provides perfect bite of functionality across all layers. Also, the engineering team concerns itself with only the pieces that impact the bite. Therefore, this increases the need for a collective conversation.

Don’t forget your training wheels in Part 1: User Story – Start With Training Wheels.

As a <type of user>, I want <feature> so that <reason>.

Let’s use this standard as an example:

“As an online shopper, I want to pay for my order so that my item will be shipped directly to me.”

However, this user story can be split many times. A few examples may be:

- I want to pay by credit card...
- I want to pay by PayPal...
- I want to pay by cryptocurrency...
- I want to pay by electronic check...
Not only can this one story be split into many ways. If most customers use a credit card, we will prioritize that first. Or if we are selling to millennials, cryptocurrency will go first. Maybe there is a promotion to drive more sales through PayPal. Our Product Owner has more flexibility and options. As a result, everyone is on the same page. Engineering Team has a more manageable problem to solve. They all get feedback sooner. As a result, the user or stakeholder may want to play with it.

**What If A Complex User Story Can’t Be Split?**

User stories can be complicated for no reason. Others cannot be split into smaller vertically sliced user stories. Or, there is too much uncertainty behind the story to create a split. If our goal is to deliver early and often, this is not a problem.

You may need to spread it out over time. Hence, a few healthy timeboxes will get some level of feedback sooner. Otherwise, the best course of action is to create a spike. Remember to timebox these as well. Yet, this should be infrequent or get less frequent but is sometimes unavoidable. Once the desired research is complete, the team can revisit vertical splits.

**What If You Accidentally Horizontally Splice?**

Call it out! Transparency is key. Consequently, there are many reasons why focus can shift from user value to technical tasks. Mature teams catch these things faster. New teams will need a tad bit more hand-holding. Keep the product owner in the room. Allow them to help drive us in the right direction. Bring the user in the room. Not physically, but with a persona or two. Do whatever it takes to re-focus everyone.

On that note, there is the rare user story that you can’t do anything with. Get over it. Deal with it. Make sure that the horizontal splice is intentional and strategic. There is no need to dilute the teamwork. Take the opportunity to get the perfect technical solution in place.

**Conclusion**

Now that you understand the difference, the value of vertical slicing will become clear. It’s time to take a look at some splicing strategies.

Happy Trails!

This article is Part 4 of 6. Next up, splitting strategies. Take a look at the other parts of this series.

Part 1 – User Story – Start With Training Wheels
Part 2 – 7 Common Mistakes Splitting Stories
Part 3 – Why Split User Stories, Anyway?