

Chapter 7 - Unleashing the True Power of Kanban

Having gotten to this point in the book, you could be forgiven for thinking that Kanban is only applicable to development teams. However, we would argue that the true power of Kanban is unleashed only when we start scaling our implementation along the dimensions described in this chapter. While the implementation and optimization of a Kanban system at the team level will help you walk, the scaling of these practices will help you run. The good news is any scaling of Kanban does not require any new principles or practices. Scaling Kanban is simply the application of the same practices that you have already read about in this book, along new dimensions.

Dimensions of Scaling

There are multiple dimensions of scaling along which we can apply the practices of Kanban. Here are some ways that Kanban can be applied to achieve great results beyond the confines of a single development team.

Multiple Teams Working on the Same Product

Products are often too large for a single team to develop and maintain. Teams would often pick up features of the product to

develop by themselves. Soon, we realize that there are multiple dependencies across these teams as they are contributing to the same product. This leads to blocked work and often more features getting started. We end up in a case where the individual teams might be able to maintain their work item WIPs, but the overall Feature WIP for this set of teams keeps growing. A very common pattern that leads to this problem is teams that are split by specialization, for example teams focused on Front End, Back End, Database, QA, etc. This problem can also exist with a set of cross functional teams working on the same product.

The observant reader would probably already have guessed the authors' answer to this problem, based on earlier chapters. This is the perfect opportunity to look at these teams as a single system. We can do this by having a single representative model for the work on the product, probably a Kanban board with the features for the product laid out. The Kanban board forces two very important practices in this context - The limiting of WIP, at the feature level across these teams and the establishment of an SLE for these features.

If we actively manage the work on this board, we can ensure that none of the features age too long. We can also facilitate teams helping each other by unblocking work or even working on some features together. This usually leads to a single product backlog, instead of a backlog for each team. As a result, new work and collaboration patterns might start to emerge. Teams might start picking up work that was initially slated for another team, allowing for greater degrees of flexibility and knowledge transfer. The boundaries between the teams might start to blur. We might even realize that it might benefit the overall flow, to combine some teams and create larger teams that can better harness the newfound flexibility. Having a Feature or Epic level Kanban board across teams working on the same product can ensure the success of all teams and as a consequence, the product.

Multiple Teams and Multiple Products

It is easy to see the benefits of a single board for teams working on the same product. Their work is probably related and they have the ability to help each other finish work items. What if we have multiple teams but sets of them are focussed on multiple products? Should we still attempt to implement Kanban at a higher level? The answer is still 'Yes'.

A portfolio level Kanban board can ensure that our execution is in line with our current strategy. If we lay out all the initiatives/features/epics (whichever level makes sense here) that the organization is currently working on, what information might we gain? We will very quickly find out what is our current WIP on these high level items across the board. We will also find out if the focus for the teams matches the strategy for the organization. When each product has its own backlog it is very likely that the highest priority item in one product's backlog is not only lower priority for the org but also runs counter to the overall strategy. Furthermore, if the team working on this product is very efficient, the org will produce more of what it does not want than what it does consider strategically important.

Having a portfolio-wide Kanban implementation helps us become both efficient and effective at the org level. We are able to make the flow of work efficient by monitoring flow metrics and improving them over time. We are also able to produce more of the right things, by making sure that the distribution of that active and upcoming work aligns with our overall strategy. The frequency at which a board at this level is managed might not be the same as a team level board. It might be less often, but often enough so we can make adjustments in time.

Higher Levels of The Organization

Similar to how multiple teams can get out of sync with the overall strategic direction, entire departments can fall into the same trap. The strategic goals and priorities of the organization can often be mismatched from the priorities of individual departments. Scaling Kanban to the organization level where high level objectives are represented on a board can help avoid this. A board at every level can help tie the high-level organization objectives to department level work items, which can in turn be tied to product and team level features and consequently to work items on the team boards. The first and immediate result of this is the understanding of whether our current activities align with our organizational strategy and goals.

A Kanban board is not complete without a Service Level Expectation and explicit policies. This is where the true benefits of having boards at higher levels start coming in. The consistent problem with organizational objectives is how long running and vague they can be. Applying explicit exit criteria and identifying a 'finished' stage can help make sure that the overall objectives are not vague. An SLE can also ensure that we do not have long running objectives that are outdated, yet still active. Actively managing items on a board at any level of the organization requires the same discipline as team level boards. The higher the level of the board, the greater impact it would have on efficiency, effectiveness and predictability of the organization.

Active management of work items on higher levels can extend the benefits of Kanban across the board. Aging of the work items on these boards can reveal WIP, sizing or dependency issues. At higher levels of the organization, we often have the right people involved to solve these issues. We are able to bring teams and departments together to work on our collective priorities and ensure delivery of value to customers.

Extending Upstream and Downstream

So far we have talked about scaling up to include more teams, products and departments in our Kanban system. Another dimension is scaling out to include more activities. Many teams would often start implementation of Kanban from the point someone starts implementing a solution to the point at which the solution is ready to be delivered. This is a great place to start, but if left alone, this system might end up being suboptimal. We are likely to create a highly optimized sub-system which produces output faster than we can deliver it to customers. It might also encourage upstream processes to try and front load the system and create waste.

When we have stabilized Kanban between the lines, it is time to scale the system out to include upstream and downstream activities. We need to redefine the start and finish points within which our system runs. Once the process of creating the solution for customer problems is optimized, we should include the processes of understanding the problem (upstream) and delivering the solution (downstream) as well as getting feedback on the delivered solution (downstream which informs upstream). As we move the start and finish lines out, we apply the same practices as described in this book - Identifying work items, defining stages, limiting WIP, explicit policies and having an SLE. Over time we are able to extend our Kanban practice from the point anyone in the org starts to understand the problem to the point we get feedback from customers that the problem has or has not been solved.

Other Departments in the Organization

Very often within an organization only one department will start with Kanban. Another mode of scaling Kanban is to cross-pollinate the learnings to other departments. If the IT department has seen great benefits from the implementation of Kanban systems, some of the departments they regularly interact with are great places to

spread the benefits of Kanban. We have seen firsthand, departments like customer support, client activations and product strategy utilize the methods that were first adopted by development teams. These adoptions have led to greater efficiency, effectiveness and predictability for these departments.

Scaling in this manner often coincides with the other scaling dimensions already mentioned. Having an organization level Kanban system can act as a vehicle for spreading the benefits of Kanban to the various departments.

Conclusion

When it comes to scaling Kanban, there are multiple dimensions to consider. We have described a few of these here. We can unleash the true power of Kanban by scaling along these dimensions. Often one of these dimensions leads to the need to work along a different one. We recommend that the organization pick the one that seems easiest to make progress with and observe some success before embarking down a different dimension. Kanban is a strategy that is applicable across multiple types of teams and levels of an organization. Do not be afraid to play outside the lines.

An example of scaling is the Ultimate Software Scaling Kanban case study. This excerpt describes how the organization combined some of the scaling approaches described here -

‘Adopting Agile techniques has provided the benefits of increased productivity and predictability. For an overall perspective though, Ultimate Software is in a waterfall sandwich. The Agile development organization sits in the middle of traditional sales and support organizations and traditional deployment and activation organizations. As a part of the next evolution of Agile and flow based thinking at Ultimate Software, we are expanding out to organizations that flank development. Ultimate’s culture that en-

courages managers and employees to experiment and make the right decisions for Ultimate, has aided greatly in spreading the principles outside of core development. Departments within Ultimate Software have started pulling the services of the Agile coaches within development to help them with the same principles.

Our closer engagement with Product Strategy and the ability to give them a higher degree of predictability has vastly improved Development's ability to assist with support issues without interrupting active work. Tier 3 support has also adopted Kanban practices in order to improve their ability to support our customers. Product Strategy is able to utilize the predictability and productivity gains of Development to provide better guidance to Sales on upcoming products and features. As we continue to improve the predictability that we can provide Sales, we can start creating feature requests and priorities in conjunction with Sales. Features can then be pulled all the way through the value stream and tracking of Cycle Time and Throughput can allow us to make and keep more accurate commitments to our customers.

While the upstream expansion helps us get better at the creation of value, expanding downstream to deployment and activations is where we can improve the delivery of value to our customers. As Ultimate Software has started working on new products, we have pulled deployment activities onto the teams. For our older products, we have always done a handoff to our Sass deployment group. We broke the 'over the wall' mentality by embedding deployment engineers on the development teams for new products and helping them educate the rest of the team on maintaining their own deployment pipelines. The teams were initially concerned about taking on the additional responsibility. Those fears have abated as the teams have realized the support that is available to them from the rest of the organization. This practice has also greatly reduced the occurrences of production environment surprises. Since the teams help build the environments that they deploy code to, the code does not behave unexpectedly when pushed to production. These teams are

supported by three groups outside of Product Engineering. Groups that manage the Build and Deployment infrastructure for the products being developed have also adopted Kanban principles and started measuring Cycle Times for making infrastructure available to teams. They have established SLAs for different types of requests and have become predictable with these metrics.

We can now see a feature make its journey all the way from a request generated in Sales to Product Strategy, to Development and finally to Production. Once we are able to track the progress of a feature in this manner, we can start identifying opportunities for improvement in the inception-to-delivery cycle. The organization as a whole can identify where features get stuck and apply our understanding of flow to eliminate the time features have to wait in queues across the entire organization.

Another aspect that is downstream from the development and even the deployment group is activations. Activations is the group that helps a new customer go live with Ultimate Software's products. The activation process can take up to a year and can involve multiple teams. Every day that a customer is in the activation phase, Ultimate Software is investing time, but not receiving full revenue. This is an area that can use the benefits that the Development Organization has gained from flow and Agile practices. Development has started working with Activations to share the principles and practices that have made a positive difference in the predictability and speed of completion for deliverables.

Moving Kanban outside the lines is the next large step for Ultimate Software. We have already started moving in this direction through our work with support and deployment teams. Ultimate continues to scale out its Agile implementation without using any of the established frameworks. Setting up the right channels of communication and visualizing our work in a manner that is easily understood by all is at the crux of how Ultimate has been able to successfully adopt and evolve Agile at scale.'