19

PORTFOLIO-LEVEL PLANNING

EVEN IF YOU'RE the product owner of an individual team, it's rarely wise to focus solely on your own backlog. You're usually part of a product ownership team and often expected to do planning at the roadmap level or portfolio level as part of your role. You probably are providing feedback to a product manager or chief product owner in this process, but you still need hierarchical thinking when it comes to products, roadmaps, workflow, and backlogs.

N-LEVELS OF AGILE PLANNING

Another typical scaling challenge is cross-team and cross-product planning. In chapter 2, as part of the 4 Quadrants discussion, I mentioned that the product owner is part project manager. I believe that the product backlog is the new planning vehicle in agile teams. It is the one agile artifact that revolves around workflow, dependencies, risks, milestones, formal testing cycles, and iterative delivery. It's the one-size-fits-all project plan for agile teams.

There are many references in agile writings to the *agile planning onion* (see table 8).[86] I've chosen to display the onion as a table, so don't be confused. I just think it's a better way to explore it.

TABLE 8. The Agile Planning Onion

	Activities, drivers, focus points	**Frequency**
1. Strategy	Vision, ideation, creative, customer	Periodically, continuously
2. Portfolio	Program, forecasts, valuation, PMO, governance	Quarterly, continuously
3. Release	Project, charter, release train, hardening, feature burn-up	Release cadence
4. Iteration	Sprint, cadence, S-curve, risk, dependencies, story/task burn-down	Sprintly
5. Daily	Scrum, standup, transparency	Daily
6. Continuous	Integration, deployment, stop-the-line	Daily

This common metaphor typically describes five or six layers. I like to group them based on frequency. For example, the continuous, daily, and iteration levels are part of basic agile execution. I consider them the heartbeat of our construction efforts.

The release level is unique. It's the glue between iterative execution and the higher-level strategic and portfolio planning functions required in the agile enterprise. As we discussed in chapter

[86] This book references the model from the VersionOne video (https://youtu.be/cmqmNWWQ5-4), but you can find many others. Some define it as five levels and drop the continuous planning part. Others have suggested extending it; for example, Mishkin Berteig has suggested that Culture is missing above Strategy ("The Agile Planning Onion Is Wrong," *Agile Advice* blog, April 24, 2011, http://www.agileadvice.com/2011/04/24/agilemanagement/the-agile-planning-onion-is-wrong/). I happen to agree with Berteig, although culture isn't necessarily a clear aspect of "planning."

11, release planning is fundamentally important to composing higher-level business goals into tangible release packages for customer consumption.

Strategy and portfolio management are at the highest layers of the onion. These are typically the domain of your executive, product organization, creative envisioning design, and software architecture and design teams.

These ideas need to be ordered, validated, and explored well enough to be release planned by their respective teams. While your agile teams might get pulled into some of this collaboration, this is usually a leadership-led activity.

As an engaged product owner, it's important to understand the overall planning activities within your agile teams. You may not fully participate in all of these layers, but you do want someone in your organization to be handling each of these in succession. Otherwise, you'll find that your team's planning, execution, and delivery are disconnected from your higher-level organizational expectations and unaligned with business priorities.

Mike Cottmeyer has a wonderful model for the three levels of orchestration that funnel work from the top of the organization (leadership) down to the bottom (teams).[87] It alludes to using different agile methods to orchestrate this flow. You see the same levels in SAFe, and most of the ALM tools have added the same idea to the portfolio management capabilities in their products. In these models, Kanban is used to orchestrate the validation flow of enterprise-level project work intended for multiple teams.

Kanban's pull model, which is based on visual workflow dynamics, is an ideal tool for senior leadership and other core

[87] Cottmeyer has his own agile consulting firm called LeadingAgile. Their general resource page has more information on this concept: https://www.leadingagile.com/the-company/resources/. In particular, look for their presentation about Exploring Agile Transformation and Scaling Patterns.

functions (architecture, analysis, design, business case development, portfolio costing, etc.) to filter work into "digestible chunks" for the organization's release processes. It also implies a pre-preparation or readiness model, where work items (epics) are pre-worked to whatever degree is required to get them into a stream of work that teams can analyze, plan, and execute for delivery in releases.

TABLE 9. Planning Onion mapping to levels and management methods.

	Story granularity and activities	**Method**
1. Strategy	Portfolio-level stories (epics), ideation, high-level product roadmaps, technology roadmaps	Kanban
2. Portfolio	Project-level stories (features or themes), business case, high-level sizing, valuation, design look-ahead	Kanban
3. Release	Features and MVPs, packaging themes into a release, mid-level sizing	Scrum
4. Iteration	Executable stories that fit within sprint boundaries	Scrum
5. Daily	Scrum, standup, transparency	Scrum
6. Continuous	Integration, deployment, stop-the-line	Scrum & XP practices

Table 9 illustrates the three-phase model for pulling work into teams from the Portfolio level and through the Release and Execution levels in larger-scale enterprise agile adoptions.

The key is for the upper-level structures to have just enough work thoughtfully prepared for execution to generate an even workflow into the releases and teams. That encompasses both ends of the pipeline: the inflow pipe for pre-work and the outflow pipe for deploying to customers.

There might even be some queuing going on at both ends as the organization is dealing with more traditional waterfall preparation and deployment capabilities. In some cases, this is advantageous as an agile migration phasing strategy while the organization ramps up to a leaner, more continuous flow model.

PORTFOLIO-LEVEL PRIORITIZATION

Throughout this book I've shared various methods and techniques for ordering your product backlogs. Most of those methods were targeted at team-level backlogs and more finely grained PBIs (stories and small features). But the same techniques can often work for high-level prioritization too.

I particularly like using theme screening and priority poker with product leadership and executive teams when ordering high-level epic backlogs or roadmaps. (Chapter 9 provides an overview of both techniques.) I usually try to get every stakeholder in a room and create as much interaction and discussion as possible around the potential opportunities and must-do work.

It's important to bring pre-sized epics and a clear understanding of the organization's capacity, using actual data. I usually post the data where everyone can see it during the meeting, as a way to bring everyone back to the reality when comparing the perception of work to the organization's capacity.

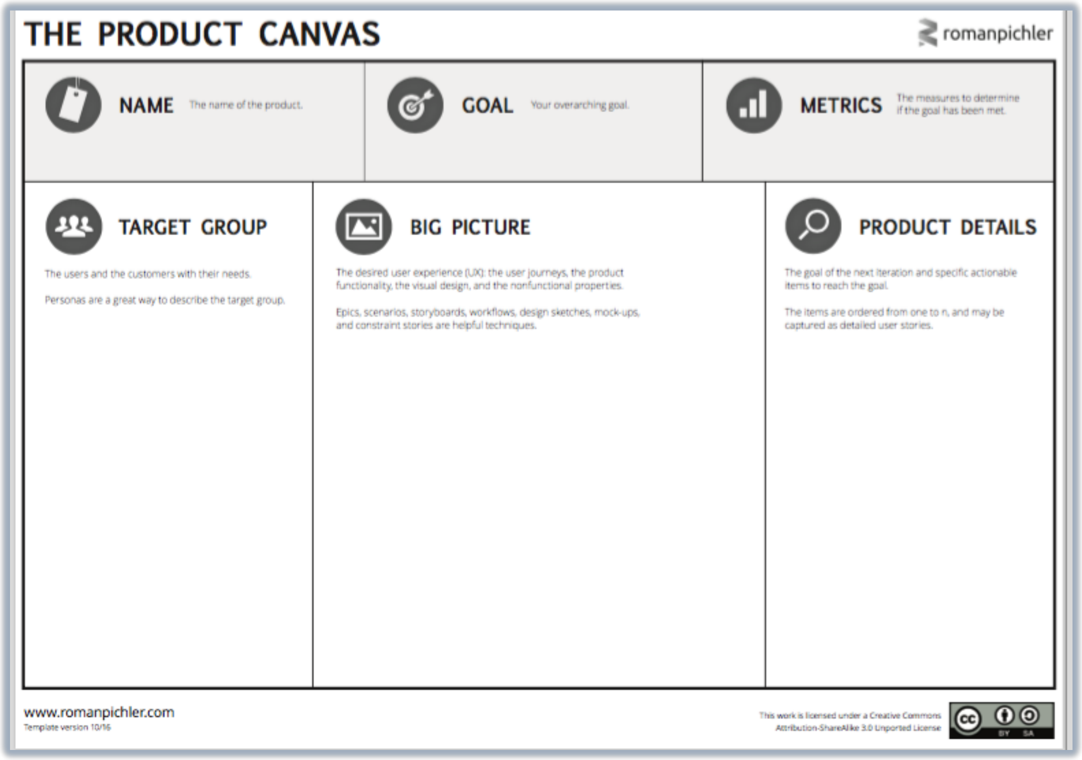
Since I've spent so much time on scaling and SAFe in the book, I need to bring up SAFe's default prioritization method: WSJF, or *weighted shortest job first*. SAFe recommends using WSJF at all three levels of the backlog as work flows from portfolio to team. It's particularly useful at the portfolio level for several reasons:

- It takes an economic view of the work.
- It ignores sunk costs.
- It quantifies the cost of delay.
- It supports continuously reevaluating the economics.

- It relies on an algorithm, which helps avoid emotional decisions.[88]

Effectively, WSJF is calculated using the *cost of delay* divided by the *duration* or *job size*. SAFe recommends using relative, Fibonacci-based values for estimating both variables, so the approach should be familiar and comfortable to many agile organizations.

There are many ways to prioritize (order) your backlogs at all levels. I've shared a few that I've personally found to be the most valuable and useful, but there are many other techniques.



THE PRODUCT CANVAS

romanpichler

NAME The name of the product.

GOAL Your overarching goal.

METRICS The measures to determine if the goal has been met.

TARGET GROUP

The users and the customers with their needs.

Personas are a great way to describe the target group.

BIG PICTURE

The desired user experience (UX): the user journeys, the product functionality, the visual design, and the nonfunctional properties.

Epics, scenarios, storyboards, workflows, design sketches, mock-ups, and constraint stories are helpful techniques.

PRODUCT DETAILS

The goal of the next iteration and specific actionable items to reach the goal.

The items are ordered from one to n, and may be captured as detailed user stories.

www.romanpichler.com
Template version 10/16

This work is licensed under a Creative Commons Attribution-ShareAlike 3.0 Unported License

FIGURE 14. Roman Pichler's Product Canvas. CC BY-SA 3.0, https://www.romanpichler.com/tools/product-canvas.

[88] For more on WSJF, see the abstract on the Scaled Agile website: https://www.scaledagileframework.com/wsjf/.

VISUALIZATION

Visualizing your portfolio is a powerful way to take a step back and review your work plans. I've found the following three methods incredibly useful for portfolio-level envisioning, planning, and decision making. All three can also be used for lower-level types of planning.

PRODUCT CANVAS

Canvases seem to be all the rage recently in the agile community. There are business model canvases and Lean canvases. There are several coaching canvases. Roman Pichler described and released the Product Canvas in 2012 (see
figure **14**).[89]

These are the key components of the product canvas:

- **Name –** Your name for the product
- **Goal** – Your overarching goal for the product
- **Metrics** – Your measures to determine whether the goal has been met
- **Target Group** – The users, customers, or personas for the product, and their needs
- **Big Picture** – The desired UX experience, user journeys or flows, visual design, product functions, and non-functional work; often articulated as sketches, mockups, storyboards, scenarios, epics, and constraint stories
- **Product Details** – The goal of the next iteration or release and actionable items to reach the goal; often an ordered backlog of user stories, refined for the period of time in question

[89] Roman Pichler, "The Product Canvas," *Roman Pichler* (blog), July 16, 2012, https://www.romanpichler.com/blog/the-product-canvas/.

CANVAS CREATION WORKSHOP

If you've seen a theme in this book, it's collaborating with the team when creating elements of your products. Pichler follows that trend by recommending a product canvas creation workshop as the method for initial envisioning and population of your canvas. He goes so far as to recommend the following framework for the meeting:

1. Establish the attendees—product owner, team, others/stakeholders
2. Bring input items for vision, business model canvas, etc.
3. Follow these basic steps:
   a. Create personas
   b. Outline the UX and the features
   c. Select goal and create actionable items (backlog)
4. Leave with outcomes:
   a. Initial sprint and/or release goal
   b. Initial product canvas
   c. Initial metrics triggered
5. Exit Criteria: good enough to start sprinting[90]

The duration of the workshop is usually four to eight hours.

This is another way to drive high-level views across your organization. You would engage people at the portfolio level to establish goals, metrics, personas, and key early epics. Once you establish those, you can drill down into individual content releases. I could easily see doing this on a quarterly tempo, tied to your release train.

[90] Roman Pichler, "The Product Canvas Creation Workshop," *Roman Pichler* (blog), May 23, 2013, https://www.romanpichler.com/blog/the-product-canvas-creation-workshop/.

Pichler's focus for the product canvas is at the team level. I think it's even more useful at the portfolio or organizational level, so please don't limit your use.

Impact Mapping

Another potentially useful envisioning tool is *impact mapping*, created by Gojko Adzic and explored in his book by the same name.[91] He's been describing this approach since 2011.

Essentially, impact mapping is a directed mind map with specific areas of interest, or a directed network of concerns.[92] Let's explore the map areas first:

- **Goal** – Every impact map starts with your goal. In this case, it's a business-focused goal.
- **Actor** – Next you list the various actors necessary to achieve the goal. Consider them personas as well, or direct customer references.
- **Impact** – This is the impact you're looking for the actors to drive toward your goal.
- **Deliverable** – Finally, what deliverable, from an epic or feature perspective, will initiate the impact that drives the actor toward the goal?

[91] Adzic has put up a wonderful website that fully explores impact mapping: https://www.impactmapping.org. The book is *Impact Mapping: Making a Big Impact with Software Products and Projects* (Provoking Thoughts, 2012).

[92] There are many available mindmapping tools that you can use for impact mapping. One of my favorites as of this writing is MindMup 2: https://discover.mindmup.com/.

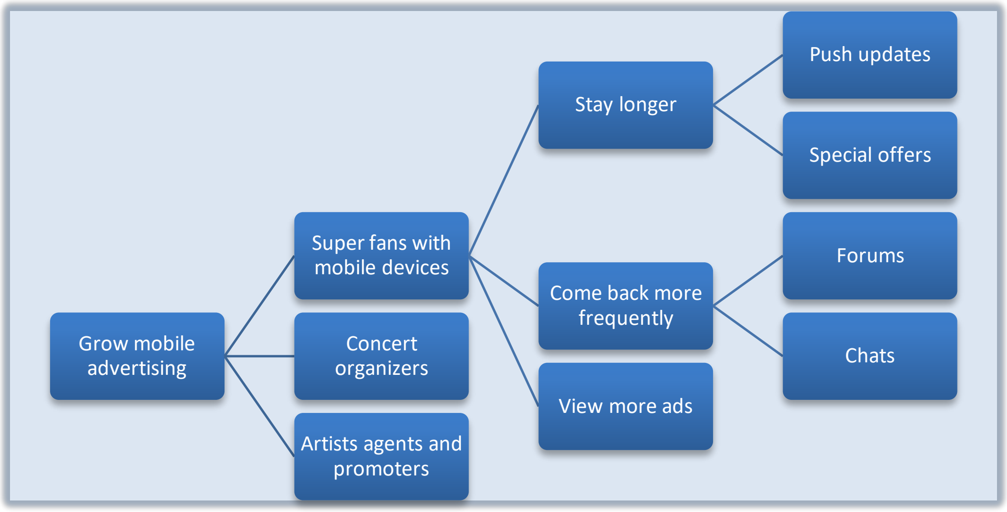


FIGURE 15. An impact map example from Adzic's website, https://www.impactmapping.org/drawing.html. CC BY 4.0.

The example in

figure 15 shows the mind map, or the hierarchical relationship from the goal to the deliverables. Beyond each deliverable, you would develop a set of user stories representative of implementing that deliverable.

It's important that you work left to right: First you define your goal, then your persona, then your impacts. Everything is goal-driven based on customer use. From that point of view, you can see strong similarities between impact mapping and the product canvas.

You can leverage impact mapping at several levels in your organization. At the portfolio level you would not decompose the deliverables into stories. In fact, you would probably keep them at the epic or feature level of granularity. Once they were prioritized at the portfolio level and made their way to your teams, you could extend the impact maps by sharing them and exploding more details from the deliverables.

The advantage here is that your teams can directly see the history and driving forces behind each deliverable (epic or feature) and use this knowledge to effectively decompose it into the appropriate stories.

BIG WALL

Mitch Lacey shared a technique he called "Big Wall" on his blog and in his book *The Scrum Field Guide*.[93] It's essentially a variant of release planning approaches discussed in chapter 11.

One of my clients used this technique as their sole portfolio planning tool, with fifty or sixty stakeholders participating in a two-day event. You can see the setup for their event in figure 16. They set up a Big Wall with all of the relevant epics that are in play for the near term—let's say for the next six to nine months or so.

The epics are clustered by theme and color-coded by type. (This is a SAFe shop, so their epics follow the SAFe format for epic-level descriptions.[94]) Their layout included *business epics*, *enabler epics*, and *architectural runway epics* that need to be considered within the portfolio-level decisions. Each epic included a lightweight business case written by the individual product owner.

[93] Mitch Lacey discusses Big Wall as a product backlog estimating *and* prioritizing technique at scale in this blog post: "The Big Wall: Prioritizing and Estimating Large Backlogs," https://www.mitchlacey.com/blog/the-big-wall-prioritizing-and-estimating-large-backlogs. See also chapter 29 of *The Scrum Field Guide: Agile Advice for Your First Year and Beyond* (2nd Edition), (Upper Saddle River, NJ: Addison-Wesley, 2016).

[94] See "Epic Abstract" on the SAFe website for more on this format: https://www.scaledagileframework.com/epic/.

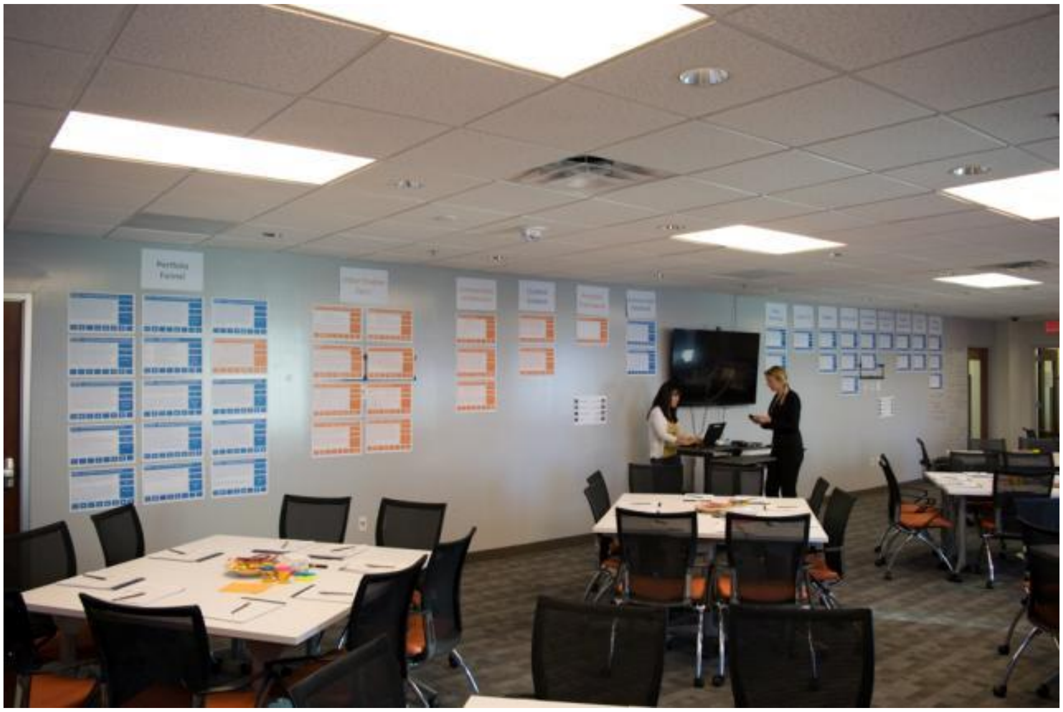


FIGURE 16. Big Wall preparation for portfolio-level planning. Photo courtesy of Ipreo.

What was interesting is the number of epics that enter and exit this event.

Inevitably the stakeholders ask for far more than the organization is capable of in the forecasted time frames. In this particular case there are over forty epics on the wall, but the organizational capacity is only ten to fifteen epics per one or two release trains.

In other words, the business ask far exceeded the organization's capacity. Nonetheless, it is useful to have that level of transparency and discussion across all of the stakeholders. It forces everyone to carefully decide how to use the precious capacity they do have.

In essence, Big Wall planning served as a funneling mechanism that help the organization visually move from what they *wanted* to what they could actually *achieve*. And they did this across all of the stakeholders, so the prioritization was ultimately thoughtful and balanced.

Big Wall is another wonderful way to leverage release-level planning techniques for portfolio analysis, discussion, and decision making. In fact, it's one of my favorite techniques because of the visual clarity and stakeholder interaction.

Finally, this story illustrates the power of making things visual and visible, and then collaborating around the wall.

Stakeholder Meeting with Kanban Roadmaps

Keeping your stakeholders and clients up to date is a challenge in and of itself. The most pressing issue of the day, other necessary work, and long-term direction and plans all have to be juggled.

I've been moderately successful with holding a regular session for stakeholders to discuss the roadmap. By keeping the focus on "what's coming up"—with a side of "here's what we learned from the last release"—these sessions help change the way stakeholders think about what's important. We occasionally drift into a side conversation ("What about Thing A that you're doing currently?"), but always return quickly to the upcoming priorities and the preparations and activities needed to make them successful.

Showing that roadmap and teaching stakeholders that it's open to change, and is not a fixed plan, keeps the conversation open. After reading several things from people much smarter than me on communicating with roadmaps, I have been using both short-term (three-month) and long-term versions to guide these conversations.

Best of all, some of the products have been successfully transitioned to using that roadmap as a live kanban board for the epics and themes that they talk about.

—Cory Bryan

Cory makes an important point in his story. It's not good enough simply to lay out your portfolio and communicate your plans. Did you notice that he "circled back" on progress to plan? That is, he created a feedback loop where progress and discovery were factored back into the portfolio planning and roadmapping discussions.

That sort of agile and lean thinking should start to pervade your planning and communication as you gain agile maturity as a product owner.

PORTFOLIO FLOW AND BALANCE

All of the at-scale agile approaches rely on a natural flow that we've been discussing directly and indirectly in this chapter. It's lean by nature, in that you use just-in-time decomposition, understanding, valuation, and priority decision making. Using story terminology, we often talk about *epics* at the portfolio level, *features* at the program or project level, and *user stories* at the execution, release train, or sprint level.

SAFe tries to insulate teams from the upper two levels of analysis and engagement. It focuses architects and designers and defines groups to do the pre-analysis so you don't interrupt the team by doing too much look-ahead work.

This is a valid concern at the team level, because they do have things to do in their current release focus. But let me be clear: I consider it incredibly rude for the stakeholders and product organization to dump a set of PI goals or features the team has never seen before into the team's lap during PI planning. Sure, others in the organization have thoughtfully written, decomposed, architected, designed, coupled into themes, and estimated all of the work associated with the stories. But none of those people are actually on the team that will be expected to do the work.

A significant part of your portfolio planning needs to include, one way or another, members of the teams who will be implementing the work, and it should happen as soon as possible in the flow from portfolio to team. Don't think of it as wasting their time. Think of it as a portfolio investment in de-risking the product work and in the team's health and morale.

SO MANY LEVELS, SO LITTLE TIME

I chose to end this book at the highest levels of product backlog planning—the portfolio level. But in many organizations, there are two challenges for product owners at all of these levels.

The first, and I can almost hear every product owner reading the book, is time; specifically, not having enough of it. This is often a factor when product owners decide to leave the upper-level planning to someone else.

The other challenge is that often there are people doing this for the product owner—usually a product manager or another senior product leader. It's hard for the product owner to get involved because they're not being invited to participate.

Both of these challenges need to be identified and overcome. Resist the urge to be "spoon fed" your work from upper levels in your portfolio or organization structure. It creates the wrong flow for collaboration, and it frequently does you and your team a disservice. I'm not saying you have to be involved with everything, but please stay aware and lightly engaged with the genesis of work as it moves from portfolio-level decision making to landing in your backlog and your team's lap. Most importantly, represent the technical skill and raw capacity of your team upward and be able to effectively articulate the *why* downward.