



# Agile roadmaps: build, share, use, evolve

Going agile doesn't mean not knowing where you're going. It means being flexible about the path you take.



BY DAN RADIGAN

## BROWSE TOPICS

- Agile manifesto
- › Scripting for new product managers
- **Roadmaps**
- Tips for presenting product roadmaps
- Requirements
- How to prioritize features using NPS
- Product analytics
- Remote product management
- › Agile at scale
- › Software development
- › Design
- › The agile advantage
- DevOps
- › Agile Teams
- › Agile tutorials
- › About the Agile Coach
- All articles

**Summary:** A product roadmap is a plan of action for how a product or solution will evolve over time. Product owners use roadmaps to outline future product functionality and when new features will be released. When used in agile development, a roadmap provides crucial context for the team's everyday work and should be responsive to shifts in the competitive landscape.

The idea that agile development discards long term planning may be the biggest myth since the Loch Ness Monster. A roadmap is every bit as important to an agile team as it is to a waterfall team because it provides context around the team's every-day work, and responds to shifts in the competitive landscape. But unlike a certain legendary Scottish water beast, an agile roadmap done right is easy to find and easy to understand.

## What is an agile product roadmap?

A product roadmap is a plan of action for how a product or solution will evolve over time. Product owners use roadmaps to outline future product functionality and when new features will be released. When used in agile development, a roadmap provides crucial context for the team's everyday work, and should be responsive to shifts in the competitive landscape. Multiple agile teams may share a single product roadmap.

## Building the roadmap

To build a roadmap, [product owners](#) take into account market trajectories, value propositions, and engineering constraints. Once these factors are reasonably well-understood, they are expressed in a roadmap as initiatives and timelines. Below is a very

### PRODUCT DISCUSSED

#### Jira Software

Plan smarter, communicate better, and release more predictably with Jira Roadmaps

[Get it free →](#)

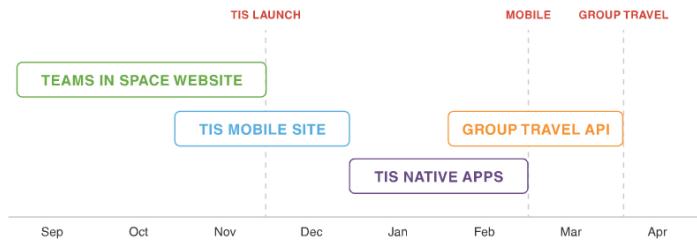
### SUBSCRIBE

Sign up for more articles

Email

[Subscribe](#)

A simple roadmap for a product team. The initiatives are in blue and timelines are indicated by the milestone-markers in red.



## Sharing the roadmap

Once a roadmap is built, it needs to be shared with the entire product team so everyone understands the vision and direction. In many organizations, product owners create their roadmaps in PowerPoint and spreadsheets, and then email the slides and spreadsheets out to the team. While well-intentioned, this strategy is flawed from the start. Each team member has their own copy of the roadmap, and keeping everyone up to speed when and if the roadmap changes is cumbersome (to say the least).

So how can the product owner keep the team better informed? Simple.



Post the roadmap online and keep it current so the team has a single source of truth.

Most collaboration tools built for this sort of thing will automatically notify all participants of a project letting them know the roadmap has changed. (If yours doesn't, it may be time to go shopping.)

When adding an initiative to the roadmap, consider the following questions:

Before we talk about dynamic forecasting solutions, let's talk about the steps to build a long-term agile plan using the metaphor of building a house:

- What are the relative priorities of each initiative?
  - When do we intend to work on each initiative?
    - Are there particular dates the team needs to hit?
    - What dependencies does the program have—either internal, or on other teams
  - Which **teams** are working on each initiative?

- Do the current teams have availability in their schedules and enough capacity?
- Can we keep the current agile teams stable?
  - If not...
    - How will teams be re-organized?
    - Are we accounting for ramping-up newly-formed teams in the project's timeline?

## Using the roadmap

It's important to link your team's work back to the roadmap so you get that whole "context" thing I mentioned above. A tried-and-true way of doing this is to break initiatives down into [epics](#) in the product backlog, then further decompose them into [requirements](#) and user stories. Tying it all together like that makes it easier for product owners and the development team to make near-term decisions that don't compromise future work. Let's look at an example to see how this plays out.

Say, for instance, that we roll out an extensive user profile feature on our website. If we find that our customers don't engage with the feature, should we continue to invest in it? Maybe, maybe not. We need to understand why engagement is low before we make this decision. So instead of pressing forward, we might opt to implement some A/B tests in the hopes of gaining some insight on the low engagement rate—which may point us in a direction that would have been far more difficult (or impossible) had we simply plowed ahead adding more bells n' whistles.

The ability to take a step back and research before we make a pivotal decision is the essence of an agile roadmap. It gives the team the ability to evolve features as they learn more about a product, and the market.

### ANTI-PATTERNS TO WATCH OUT FOR

- Future planning is completely ignored – we're shootin' from the hip!
- The "rest of the business" is kept in the dark as to what the team is up to.
- The roadmap is continuously updated (or never updated).
- Detailed requirements are weighing down the roadmap.

## Evolving the roadmap

Waterfall projects require a huge upfront investment. As a result, team members become emotionally attached to the roadmap and sacrifice making the right decision because it's too painful to undo the work they have done—a "human" sin, if ever there was one.

For its part, agile development runs into three different risks:

- The team may lose confidence in the ability of the leadership to make strategic decisions if the roadmap is updated too frequently.
- The product might arrive too late on the market and miss out on pent-up demand if the roadmap is not updated frequently enough.
- Long-term efforts can seem "too big and too difficult" for shorter iterations. The team over-compensates by breaking the work down into powder-fine granularity, and ends up focusing too heavily on short-term results.

To combat "thrashing", staleness, and nearsightedness, keep the roadmap evenly focused on short-term tactics and strategic, long-term goals. A great way to do this is to review roadmaps quarterly, adjust as needed, and share. This works well in any size organization, but remember: a single roadmap can span multiple agile teams, so inspect, adapt, and communicate accordingly.

Keep reading *The Agile Coach* to learn about special considerations for larger teams who manage [agile portfolios](#) which have roadmaps across several teams. Or learn more about [Jira Software's roadmap offerings](#).

SHARE THIS ARTICLE



DAN RADIGAN

Agile has had a huge impact on me both professionally and personally as I've learned the best experiences are agile, both in code and in life. You'll often find me at the intersection of technology, photography, and motorcycling.



TUTORIAL



ARTICLE

## Learn versions with Jira Software

Learn how to use versions to organize your work around milestones you can aim for. Learn to assign issues in your project to a specific version.

[Try this tutorial →](#)

## How to get buy in for your product roadmap

Win the development team over with your roadmap. Learn which top 10 tips will help product managers get buy-in from technical teams.

[Read this article →](#)

### Agile Topics

Agile project management

Scrum

Kanban

Design

Software development  
Product management

Teams

Agile at scale

DevOps

Sign up for more agile articles and tutorials.

Email

[Subscribe](#)



Up Next

[Tips for presenting product roadmaps →](#)