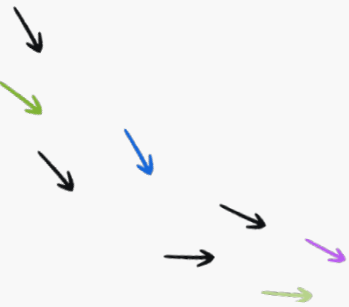
ATLASSIAN

Teamwork that drives ROI: 4 steps to product-centric program management

TABLE OF CONTENTS

Teamwork that drives ROI: 4 steps to product-centric program management



ATLASSIAN

Keeping a single project on track is hard – Program Managers need to coordinate work across an entire portfolio of projects! And honestly? It's incredibly challenging.

That means keeping:

- Teams on track through four stages: goal-setting, planning and prioritizing, monitoring work progress, and controlling dependencies
- Empowering multiple teams to work at their best – from setting a strategic vision to anticipating those unexpected conflicts and dependencies
- Juggling capacity constraints across teams while managing competing priorities and resource limitations

No matter what, all Program Managers share an ultimate goal: helping teams work better to deliver strategic outcomes.

That's a big job, and many Program Managers feel like they're struggling and all on their own when it comes to breaking it down into actionable steps.

But there's a new way of doing things, and it's inspired by how the best product teams work. We'll spell this out for you in the rest of this guide in a 4-part journey covering:

#1

Establishing goals as the foundation for product work

#2

Planning and prioritizing across a project portfolio

#3

Monitoring progress to keep work on track at a program level

#4

Mapping dependencies and capacity constraints across projects

We've also tapped three Program Management experts at Atlassian to share practices they've seen set modern Program Managers up for success in this field:

CAITLIN EWALD
Principal Marketing Program Manager

ASHOKE CHAKRABARTI
Senior Director, Technical Program Management

JESSICA IVES
Sales Senior Program Manager

Wherever you are in your program management journey, you can start elevating your skills to deliver incredible product outcomes today. We're here to support you with tips and expert advice.

STEP #1

Keeping teams aligned around goals

You can't run a race if there's no finish line – and the same is true of product work. For teams to be productive, they need to know where they're going. It's the responsibility of leaders to set that finish line, and Program Managers can help make sure it's always visible.

It's not enough to show up and deploy every day. What are the actual outcomes work is trying to create? Without clear goals, product teams' hard work won't actually drive the business forward. Teams might work toward their own conflicting or irrelevant objectives, or end up overfocusing on output metrics to "prove" they've been working hard.

Outputs vs Outcomes: Shipping a feature on time is an output: a specific task your team's agreed to achieve. Providing value is an outcome: for example, a better user experience, proven by a reduction in support tickets. Focusing on outcomes over outputs keeps work aligned with delivering value – *why* teams are working, not *what* they're doing day-to-day.

Examples of outputs

- Ship 10 features by a specific date
- Publish 3 long-form content assets per quarter
- Launch a product on time for a specific conference

Examples of outcomes

- Increase conversions from trial to paid from 15% to 25%
- Reduce support tickets for a feature from 50/week to 5/week
- Prove that x customers solved a specific problem with a solution.

Creating goal alignment has two parts:

1. Clarifying leadership's strategic goals
2. Communicating them well across teams

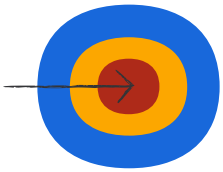
Program Managers are the glue that makes everything happen across the organization, which puts them in a uniquely influential position to support both parts of goal

alignment. They can advocate for strategic clarity by helping leadership decide on strategic goals–for example, by sharing information about user needs or the competitive landscape. Or, they might spend more time building processes to share those priorities, so they drive daily work and decision-making. Either way, Program Managers are positioned to bridge the gap between strategy and execution.

Clear, well-communicated goals are the foundation that enables fast, focused, and autonomous teamwork. "When teams are well-connected to strategy, they don't need as much managerial oversight, because they have an overarching vision they're marching towards," explains Jessica Ives.

How to create goal alignment

Define goals through strategic vision sessions



On a cadence that works for your organization, hold regular strategy alignment meetings with your executive suite. Consider starting with one foundational session to focus on long-term (3-5 year) strategic vision, followed by check-ins each month or quarter to break these goals down into team-level priorities.

These discussions can be led by a strategist or strategy-focused Program Manager, who will work with leadership to define a hypothesis for each desired outcome, and choose metrics to measure success.

Break strategic goals down into outcome-focused OKRs

As leadership determines strategic goals, consider the OKR (objectives and key results) format to translate them into measurable, outcome-oriented results. OKRs are especially useful during periods of growth, change, or when there's a need to improve transparency and accountability across the organization.

For example, if "improving customer satisfaction" is a strategic goal, you might choose "increase NPS from 30 to 50" as an OKR. Instead of "implement Feature Y by Q4," connect the feature to a goal like "increase conversion by 5% with Feature Y."

Unlike top-down goal-setting, OKRs should be collaborative. Once leadership sets the direction, teams help define OKRs that are meaningful and achievable. *Consider running an **OKR Play** session with your teams.*

Here's why OKRs work so well for Program Managers:

- **Determine what matters:** OKRs make it clear what is a priority and how your work contributes to those organizational goals
- **Determine what doesn't matter:** OKRs clarify what is not a priority, empowering teams to focus on a limited set of priorities and reject tasks that distract from core work

STEP #2

Planning and prioritizing across a project portfolio

Established, program-wide frameworks are the only way to keep planning and prioritization on track at a program-wide scale. These frameworks need to be applied iteratively and consistently, so teams can adapt to changing factors like customer needs, market conditions, and leadership priorities. Rigid annual or quarterly planning cycles can lead to teams working on things that no longer matter.

Beyond keeping work on track, standardized frameworks help everyone understand why decisions were made. Teams feel more motivated and engaged when they know why they're working on certain tasks – and it creates explainability should they need to walk leadership through their process later.

How to plan and prioritize at the program level

Balance autonomy and consistency

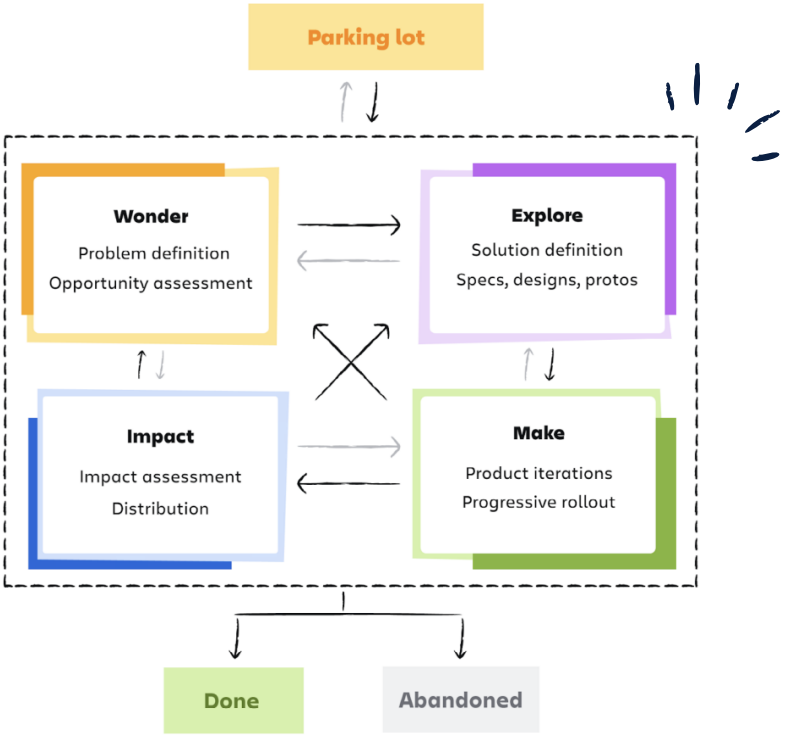
Teams work in different contexts, so you need to empower them to work in the ways that make them most successful. But you also need to make sure this doesn't lead to a breakdown when different teams try to work together. How do you do this? You need to balance autonomy with consistency.

To do this, you need to create a shared language that lets teams understand what work is happening and why certain choices are made. At Atlassian, we focus on:

- OKRs (we already discussed this above)
- 4-stage idea lifecycle
- Boulders, rocks, and pebbles for categorizing effort

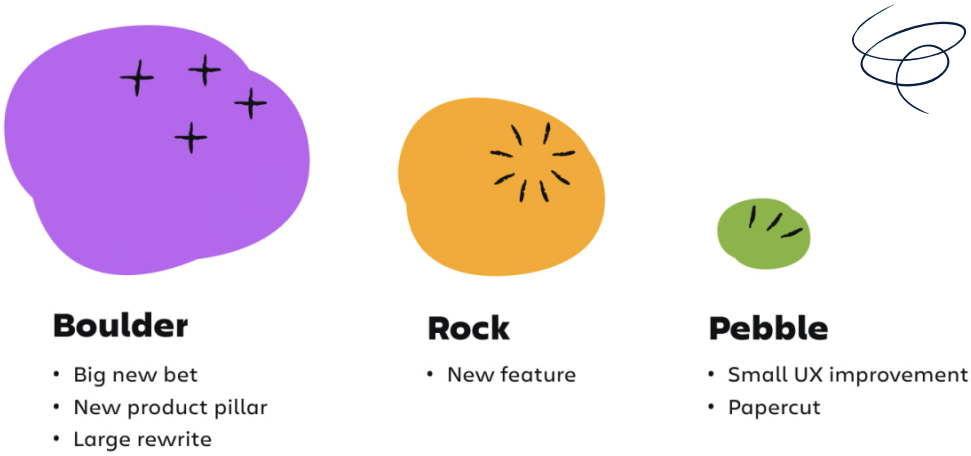
4-stage idea lifecycle

At Atlassian, we use a four-stage idea lifecycle: Wonder, Explore, Make, and Impact. We use this cycle to vet, ideate on, and execute all product ideas. Because our entire company understands this framework, people know exactly where an idea stands when they hear which stage it's in. Having a common language for the idea lifecycle makes prioritization conversations more productive, both within teams and with stakeholders.



Categorize ideas by effort into boulders, rocks, and pebbles

'Effort' refers to the level of investment or resources required to execute an idea. At Atlassian, we categorize ideas into Boulders, Rocks, and Pebbles



- Boulders: A large investment with potentially big payoff but high uncertainty
- Rocks: Medium-sized investments with fewer risks
- Pebbles: Small, typically straightforward investment

Striking this balance between individual-team autonomy and cross-organization frameworks is key to planning and prioritizing across the entire portfolio.

Use a collaborative, iterative planning cadence

Instead of top-down planning, hold regular cross-functional collaborative sessions where teams can review priorities, identify dependencies, and commit to a plan for the next increment. When the actual people involved in work are responsible for planning it, it's easier to catch dependencies and conflicts early, adapt to change, and accurately estimate the potential impact and effort of different bets.

Atlassian 'Rolling 4' (R4)

At Atlassian, we use a 'Rolling 4' planning cadence. Every quarter, we look back and review results, and look forward and plan 4 quarters ahead. At each R4 session, we reflect on progress toward our strategic targets, refresh our goals to stay focused on our strategy, and realign work and resources (that is, people and dollars) to deliver on our goals.

Rolling 4 is a company-wide ritual designed to share and broaden knowledge, make better and faster decisions, and allocate resources more effectively. Here's how it works:

The R4 ritual includes 6 key activities:

01
Metrics & Financials Review - Review quarterly business metrics and financials that feed into resource allocation decisions

02
OKR Refresh - Teams reflect on pivots based on strategic changes, market conditions, or competitive landscape shifts

03
R4 OKR Review - Quarterly review of progress against OKRs and plans to achieve targets

04
'State of' - Leadership reviews what's going well, challenges, key decisions, strategic shifts, and anticipated resource needs

05
Allocation Meeting - Review and approve changes to focus areas, portfolios, initiatives, and resource allocations (both dollars and positions)

06
Delivery Planning - Create milestones for the next 4 quarters with high fidelity for the current quarter and lower fidelity for future quarters

What teams actually do during R4:

Ahead of each session, each team receives a template to fill out with a summary of their progress, a breakdown of investments (we use the $10 resourcing game), a retrospective template for the last quarter (including key win and delivery hit rate), and a roadmap view of what we're planning to pursue in the coming quarter and rolling 12-month year.

The beauty of R4 is that it's both a review mechanism and a planning process. Teams don't just report status–they actively participate in shaping strategy based on what they've learned, and they commit to specific milestones and dependencies for the quarters ahead. This creates alignment from strategy all the way down to individual team execution while maintaining the flexibility to adapt as conditions change.

Communicate not just what **prioritization decisions were made, but** why

Part of prioritization means communicating not just what was decided but why. For example, if a feature was de-prioritized, Program Managers need to explain why they prioritized something, and why it's expected to deliver superior value. This builds trust, and ensures that both teams and leadership know *why* they're working on certain tasks.

> The element of value for a Program Manager is understanding your audience and communicating effectively to them," says Ashoke. "Even if you run a great program, if you can't explain what's going on in context, you're not going to be effective in this role."

These choices should also be documented, so they can be revisited later if factors affecting the program change. Stakeholders can also be invited into these conversations to voice their concerns and understand trade-offs.

In practice, this might look like publishing the scoring results of your prioritization framework, along with any adjustments made after discussion.

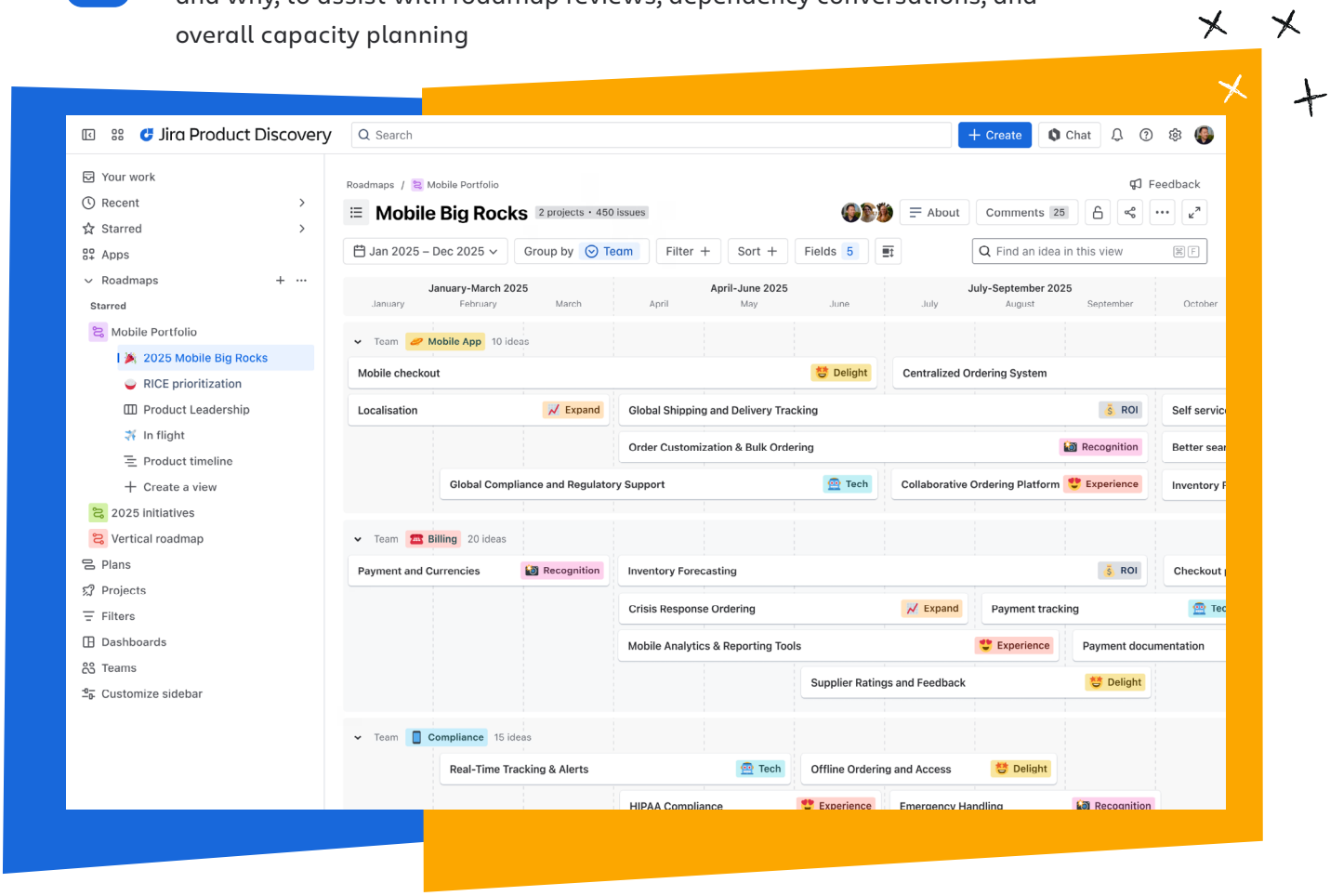
Share the program roadmap

Once you decide which work you'll prioritize, Program Managers need to put this work into a program roadmap to ensure everyone on the team knows which work will be done in what order.

Program Managers can tailor these roadmap to provide different views to different stakeholder groups. Program Managers should use *live* roadmaps,

because, unlike static slide decks, live roadmaps stay updated as things change (as they inevitably will!).

- Leadership roadmaps can provide a big-picture overview that highlights the most critical projects or releases first
- Go-to-Market team roadmaps would show when new features are being released, so they can plan marketing and launch activities.
- Research and Development roadmaps would show what is being prioritized, and why, to assist with roadmap reviews, dependency conversations, and overall capacity planning



Here's a big-picture, timeline-style roadmap view. Key investments (called Big Rocks here) are shown for the year, grouped by different product teams. This kind of view shows leadership what teams have planned, and when they expect to achieve it.

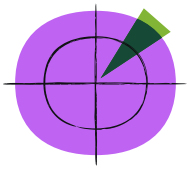
STEP #3

Tracking work progress on a program-wide level

As priorities are decided on, Program Managers need to support progress towards them. That's a unique challenge, because Program Managers don't usually hold direct management power. Rather, they guide teams in discovering how to do their best work, and give leadership visibility into cross-team progress.

> As a Program Manager, I don't hold performance authority – I simply facilitate better ways of working," says Jessica Ives. "I do that by gathering data on how teams deliver, and sharing it with other stakeholders so they can have productive conversations about performance and outcomes."

If Program Managers can only do this through manual data-collection and reporting, they'll end up with a fragmented view of portfolio status that goes out of date quickly. This can be especially problematic if teams are using many different tools which Program Managers need to pull data from, or if they're all defining success with different metrics, making it challenging to put together a coherent picture of where the project's at.



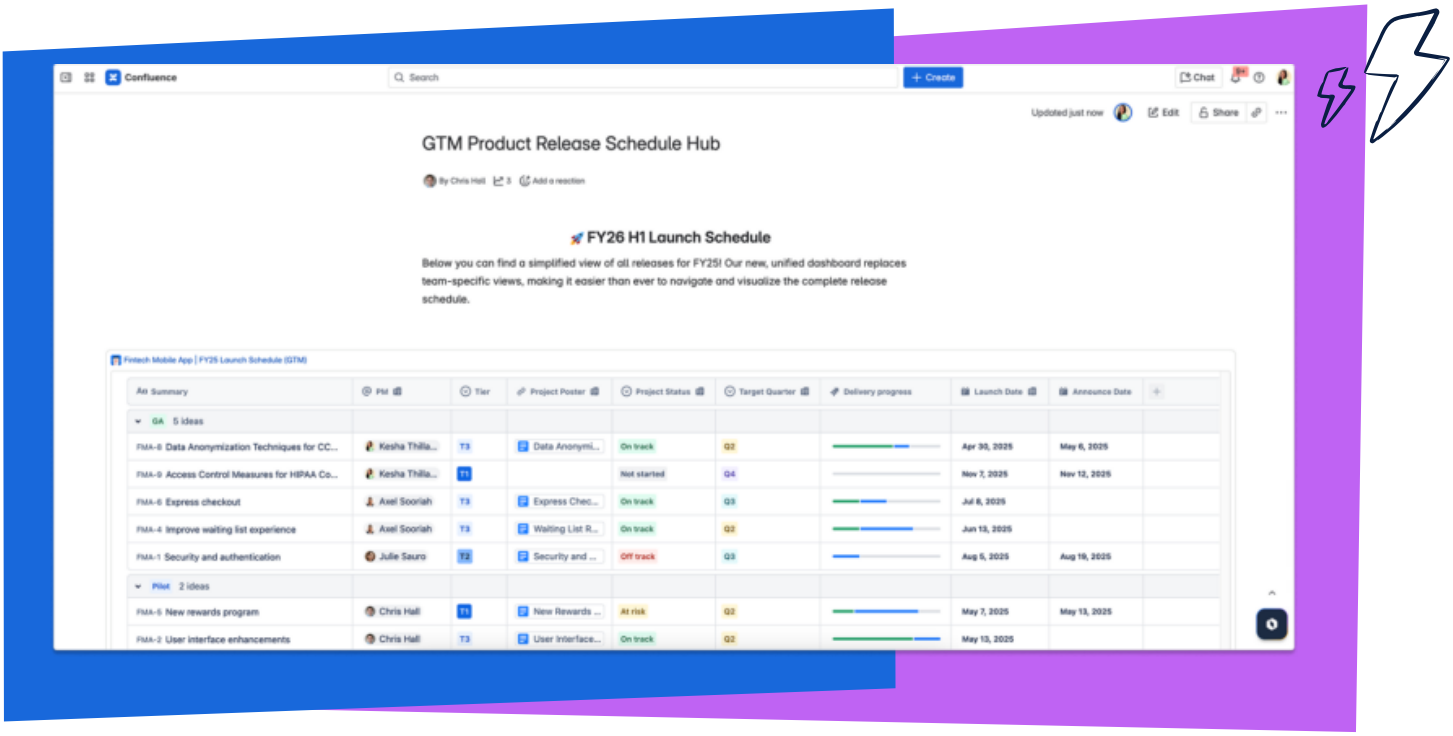
Instead, the goal is real-time, transparent tracking that links all the way from tasks to strategic outcomes. That allows Program Managers to spend their time empowering teams to work better, rather than gathering and sharing updates via email, spreadsheets, and meetings.

How to manage work across teams

Implement a project portfolio tool

If you choose the right tool, your single source of program truth can also give Program Managers and stakeholders an at-a-glance view into the health of the entire program. The tool should aggregate key data: milestones, current sprint progress, risks, dependencies, and metrics for each project, and then roll them up to program-level indicators.

This real-time visibility ensures that if something goes off-track, it's immediately apparent, and decisions can be made quickly.



Here's an example of a hub where stakeholders can get a full overview of multiple teams and projects – or narrow in on only the roadmaps they need. By standardizing how different teams enter information into the dashboard (like using common fields for idea priority and status, etc.), a Program Manager can easily merge or switch between views to get different looks into the portfolio.

Standardize status and success metrics across projects

If every team tracks success differently, it will be impossible to see the program's progress. Even if teams use different methods internally, they should be converted to a unified scale for program reporting.

This includes qualitative status as well as numerical progress scores, like story points, or % complete. For example, teams could use Red-Yellow-Green scoring to track scope, schedule, budget, and key risks.

Just ensure all teams define these qualitative categories the same way. For example, Green could mean on track, Yellow could mean at risk and currently being managed, and Red could mean a critical outcome will be missed without intervention.

Consistent metrics don't just make it easier to understand program-wide status – they also identify patterns. For example, if multiple projects are at a Yellow risk level, that can be escalated as a program-level issue.

How Deliveroo coordinates 100+ product teams around strategic goals

To provide the best food delivery experience in the world, Deliveroo works with more than 175,000 merchants and 150,000 delivery riders across 10 worldwide markets. This ambitious mission requires collaboration from 80+ product teams and 20+ platform teams.

Szymon Kikla, Head of Product Platform Engineering, knew excellent product work started with program-level planning. So, he introduced Jira Product Discovery for Deliveroo's annual planning exercise.

Before, planning and forecasting required dozens of complex spreadsheets. The 100+ teams would submit ideas, which Deliveroo's Technical Program Management Team would vet and add manually to the appropriate sheet. This exercise took two to three months every year, directly affecting the work of thousands of people.

Instead, Deliveroo needed one source of truth for strategic product goals, directly connected to prioritization and planning. The right system would have the flexibility to support teams' varied working styles, but be structured enough to give leadership full visibility.

Szymon customized Jira Product Discovery with Deliveroo's own prioritization model, and launched it for their 2025 annual planning process.

Szymon's team created a single program-level project with possible ideas for Deliveroo's annual roadmap. Within the project, 17 boards act as group-level roadmaps for different business and product functions. Business units across Deliveroo create tickets and submit them to the relevant board with support material like summaries, product requirements documents, and technical design documents. Every submission must be linked to a strategic goal.

From their primary view, leadership can see all initiatives submitted across teams. Within Jira Product Discovery, teams still have the autonomy to roadmap how they want to. For example, the Design Team uses a Kanban board with Now, Next, and Someday buckets, while others use a timeline.

Jira Product Discovery also helps Deliveroo manage the complex web of dependencies behind every new initiative. When creating an initiative, teams must specify all groups that need to contribute. These contributors then get an automated notification, asking them to sign off and provide their requirements, like effort estimates or headcount.

In the first year alone, Deliveroo saved three months of work by using Jira Product Discovery for their annual product planning exercise. Today, more than 100 technical and business teams at Deliveroo use Jira Product Discovery for planning and prioritization.

STEP #4

Tracking dependencies and capacity constraints across teams

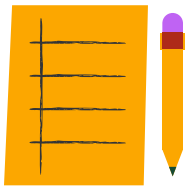
Interdependencies and resource-sharing are inevitable when multiple teams are working towards connected goals. This is one major way Program Managers' jobs can be even more challenging than product or program managers'. When coordinating multiple projects, there are exponentially more dependencies and capacity constraints to juggle.

In a siloed environment, it can be very difficult to visualize how teams' work connects to each other. Program Managers need a good way to track dependencies and stay within capacity. Otherwise, they risk discovering over-commitments and dependencies once they're already holding back work.

With the right tools and practices, Program Managers can proactively account for these constraints – for example, flexibly allocating resources across projects where they're needed. This could be the difference between the whole project grinding to a halt because everyone's waiting on the same overloaded specialist, and things moving ahead fluidly because the Program Manager anticipated this bottleneck and brought in contractors ahead of time.

How to track dependencies and manage capacity

Make dependency mapping part of early-stage planning

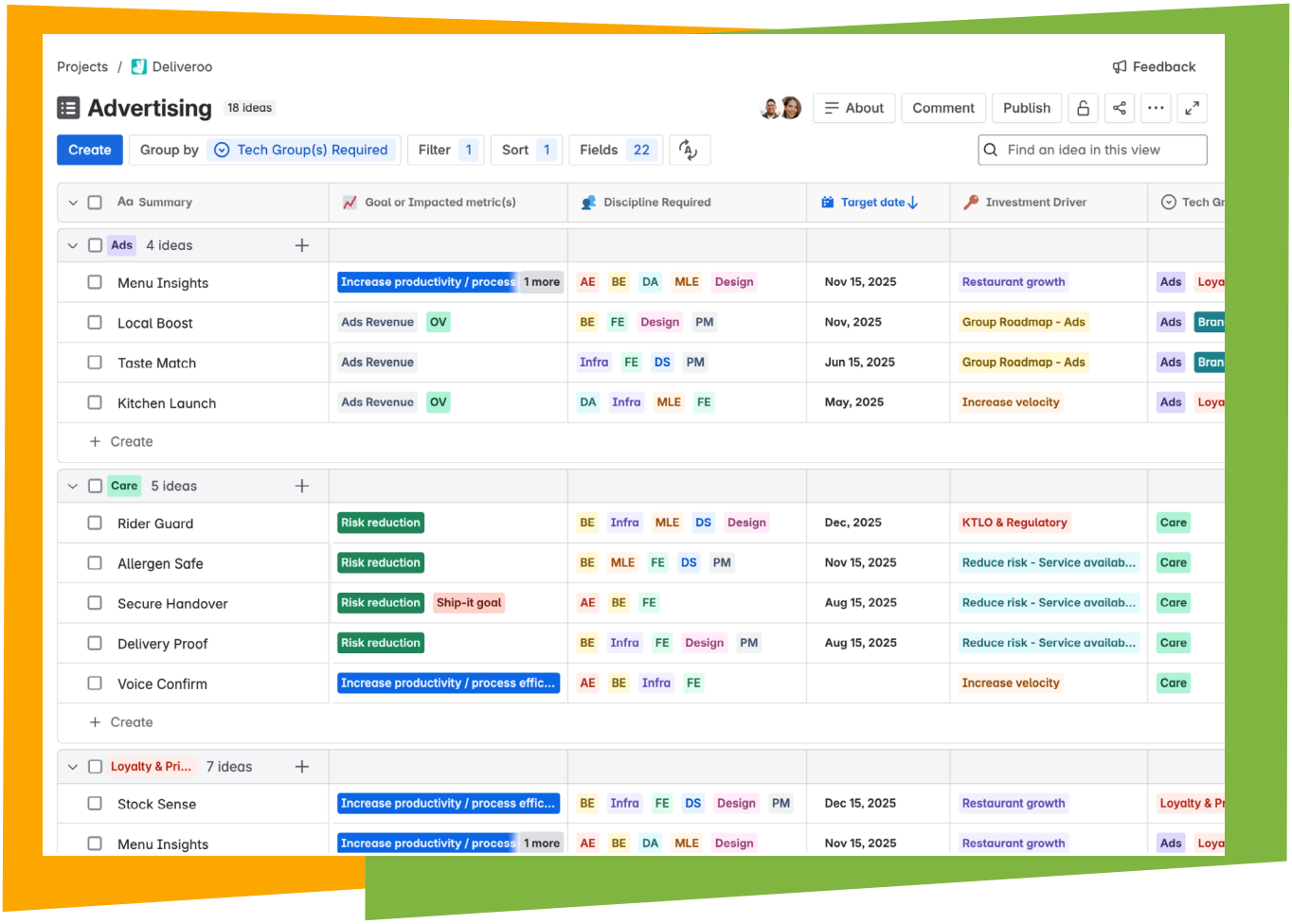


During initial planning, Program Managers should work collaboratively with teams to create a dependency map – essentially, a list of what teams need from each other to complete their work.

To manage capacity, identify scarce resources that are most likely to bottleneck the program. It could be a specific expert (say, the only security architect), a component (like a core database that everyone needs to modify), or an external vendor delivery.

Teams can then brainstorm how to reduce the dependencies, and develop contingency plans. Perhaps you could cross-train another person to cover the security architect's work, or prioritize getting tasks for the database done first.

Where dependencies are unavoidable, treat them as first-class work items. Assign each dependency an owner, and include it in your plan with clear deliverables and dates.

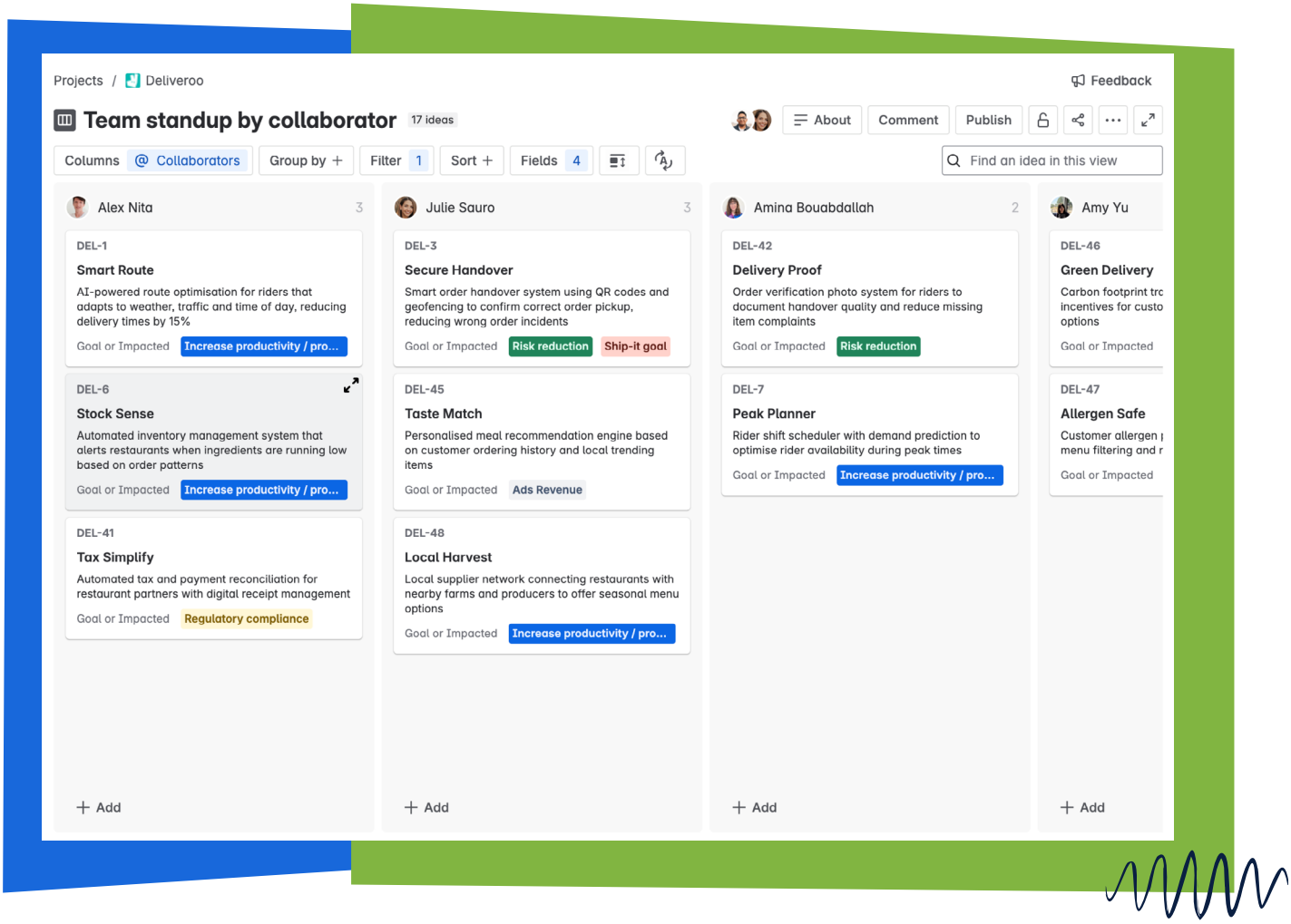


Here's a view that can be used to identify cross-team dependencies and potential capacity issues. By adding "Discipline required" and "Tech Groups required" fields and a "Target Date" for planned release, Program Managers can then group by tech groups and sort by target dates to quickly spot which teams are overloaded, under-resourced, or waiting on each other.

Hold regular cross-team syncs to manage dependencies

It's not enough to identify dependencies at the start of the project – teams need to proactively stay ahead of them. Dependency management is about ongoing communication. Teams should be encouraged to flag dependencies and risks right away. This is seen as contributing to program success, not complaining.

Hold regular check-ins where teams can update each other on dependency status and compare team's planned and actual output. Optimizing workload is an ongoing effort: if one group finishes early, redeploy them to assist others; if one is late, see if another can pick up slack.



This view shows upcoming work, sorted by team member. This could be used in standup meetings to help teammates visualize each others' workload and re-allocate to support each other as needed. This kind of view can be used when work is well underway, or early in planning to identify dependencies early. For example, leads and engineers could flag issues like "this will require significant work from the Platform team, which is already at capacity this half."

Pairing human and technical skills to guide teams towards success

Big dreams are **impossible alone**; that's why Atlassian is all about teamwork. Program Managers have a special role to play in supporting that collaboration, so organizations can accomplish their goals. With the skills and techniques we've shared in this guide, you're well on your way.

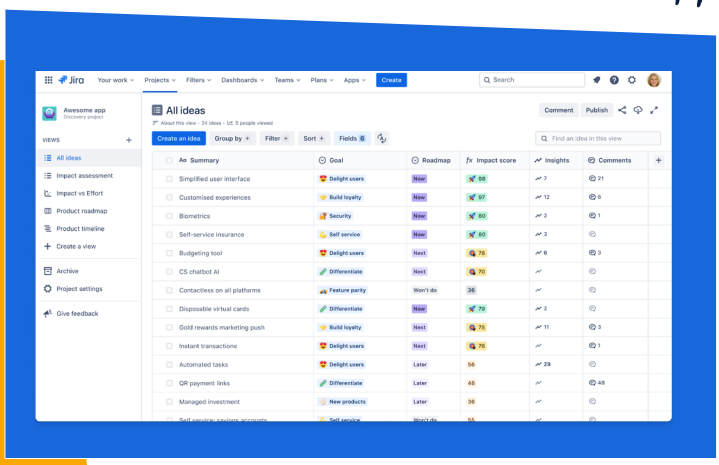
Here's the TL;DR of our roadmap to program management excellence:

- Get crystal clear on goals – so you can empower teams to deliver on them
- Plan and prioritize – in a big-picture, program-level way
- Monitor and track progress – so you can support teams in achieving greatness
- Map dependencies and capacity – so teams can collaborate and support each other

Get Jira Product Discovery for free and elevate your program management!

Tools won't make-or-break a strong Program Manager – but they can certainly make life easier.

In **Jira Product Discovery**, everyone from leadership to engineering to sales and marketing gets a clear view of progress, and a space to collaborate towards a bold vision of your product's future.



Learn more and try it for free at atlassian.com/product-discovery

ATLASSIAN

ATLASSIAN