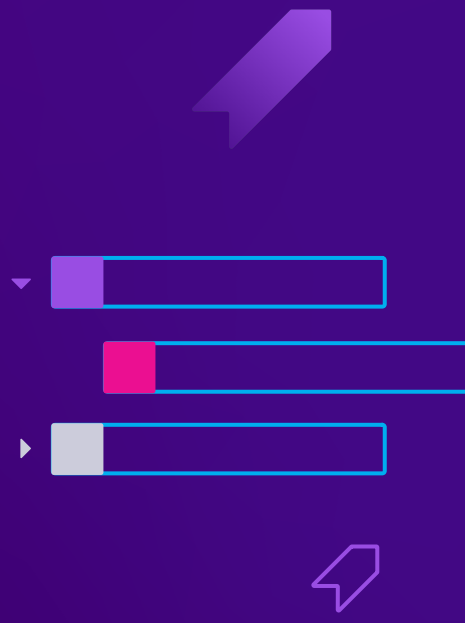
ALM WORKS



Flexibility: The Key to Managing Complex Projects in Jira

almworks.com

Table of Contents

Scaling starts with flexibility **4**

How Structure works **6**

- Example #1: Visualizing a portfolio of projects **7**
- Example #2: Sprint planning across teams **11**

Using Gantt charts for Agile teams **14**

- Example #1: Gantt portfolio roadmap **16**
- Example #2: Gantt for Agile teams **19**

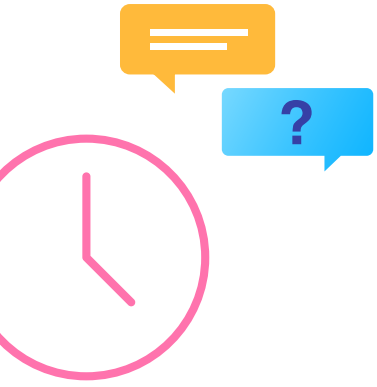
Flexible solutions for complex PPM **21**



No matter where they work, project managers often face the same types of challenges. Tracking and planning, managing resources, executing projects on strict schedules, balancing efforts between workstreams — these things come up over and over.

But while the problems are similar, the solutions are not.

Each project manager has to wrangle those tasks within the context of their own industries and companies. Regulatory requirements, business priorities, or workplace culture quirks make it impossible to prescribe a one-size-fits-all solution. Everyone has to find their own way to organize and communicate.



Scaling starts with flexibility

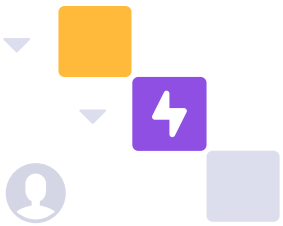
Right out of the box, Jira offers highly configurable workflows and custom fields, so you can set up projects to suit each team's unique process. Plus, Scrum and Kanban boards included with Jira Software offer planning and tracking for Agile teams.

But for many companies, there's a tipping point — too many teams, tasks, and dependencies between them — where organizations may outgrow Jira's standard capabilities. Project managers realize that they've gradually come to rely on a patchwork of additional Excel spreadsheets, Google Docs, and Post-it notes just to keep project details accessible and understandable. The whole apparatus starts to feel creaky and in desperate need of shoring up and streamlining.

That's the driving idea behind Structure for Jira — it expands upon Jira's functionality in ways that organizations can tailor to their own requirements. Here, we will detail how this works in practice with a few common use cases, so you can see how to clarify Jira data and communicate those findings more easily. We'll also show how the Structure.Gantt extension can introduce timeline views to make dependencies and resources easier to visualize.

How Structure works

Example #1:
Visualizing a portfolio of projects

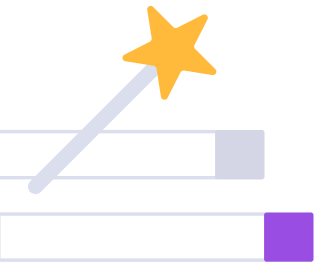


Structure helps you assemble Jira data into a clear configuration that makes complex portfolios of projects easier to digest.

Here's an example. You are running a large program, which spans across multiple teams and Jira projects. The teams are using Agile boards, where they manage their epics, stories and sub-tasks. Teams' epics contribute to larger initiatives, which are shared between multiple teams. Initiatives are linked to the corresponding themes that represent the highest-level requirements.

We can easily configure such a hierarchy in Jira using issue links and epic links, but out of the box there is no way to visualize it. With Structure you can easily build overviews of such programs and show all the data you need for planning and tracking.

You start by building a hierarchy of issues that we refer to as "structures." These can represent whatever you want, but in this example we will have them represent a portfolio of projects. You can build as many structures as you want, for any purpose you want.



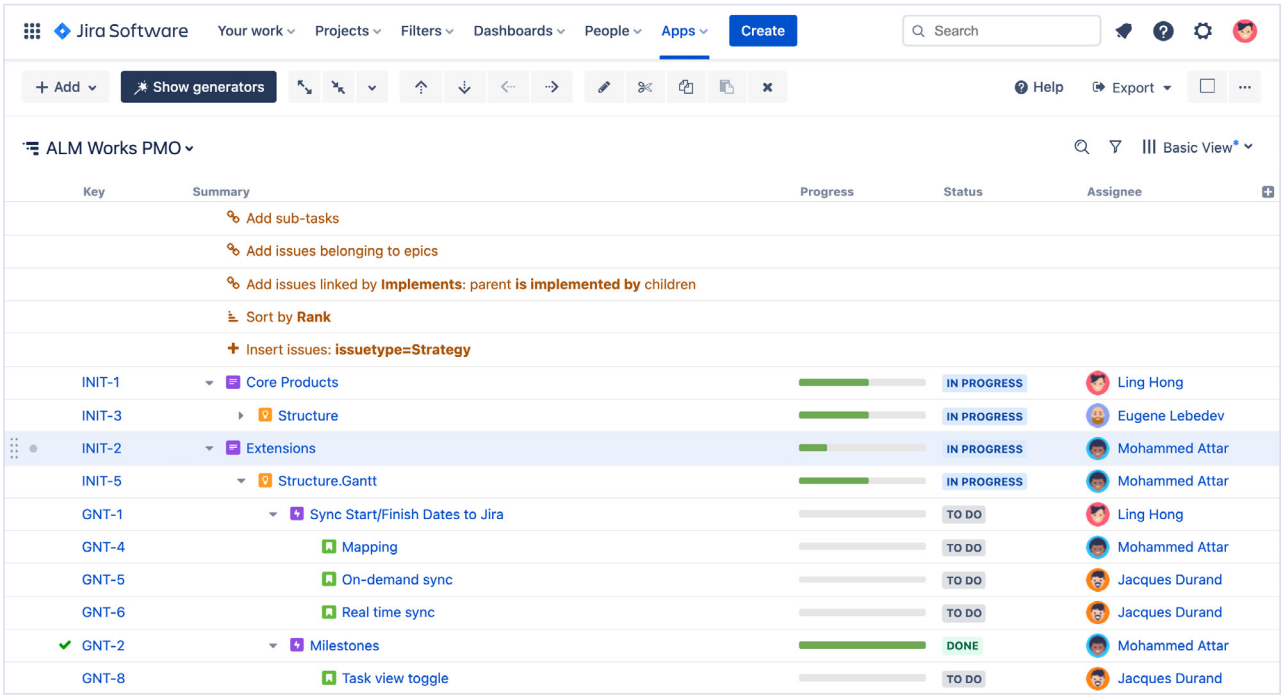
You can build structures manually, but the real power of the app lies in the ability to build dynamic structures based on different types of rules that you have at your disposal. We call these rules “Generators.”

Think of Generators as building blocks. They let you create an issue hierarchy for your specific use case, without having to change how your teams work.

Here’s how you’d use generators to visualize a portfolio of projects:

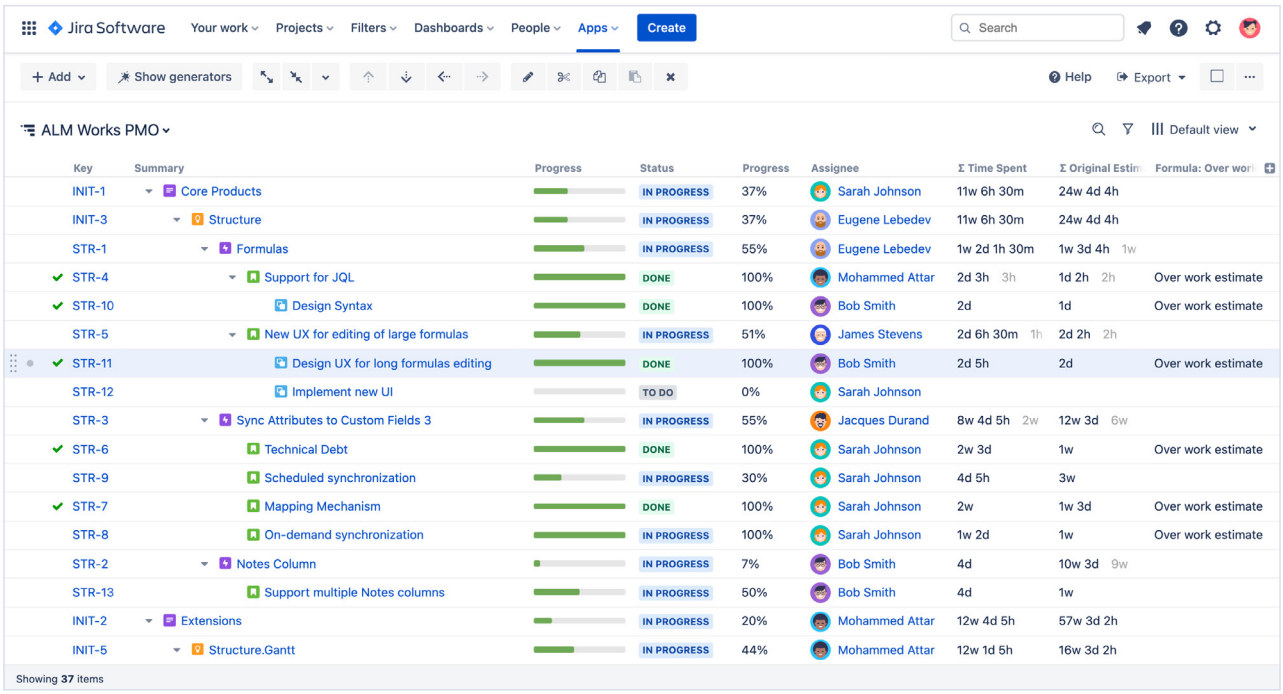
- Add all top-level items, in accordance with how your organization names them. In our example, we have themes at the top level.
- Bring in linked child items, based on issue links, following the convention of your organization.
- Add stories under epics, and sub-tasks under those, by following a similar principle.
- Use the sort Generator to arrange issues based on Jira rank, so they appear in the order set in your Jira Agile boards.

This will give us a full top-down view of all the work in this portfolio of projects and effectively rank issues by their importance.



A structure is a spreadsheet-like view of your Jira data based on rules you define

Structure allows you to add Jira fields as columns in your structures, allowing you to see the values of those fields and in many cases edit them in-line. It also allows you to aggregate data. For example, a custom progress column might aggregate data like work logged. In this way, Structure helps you work with a number of issues from different projects at the same time, allowing you to quickly turn a variety of data points into actionable insights.



Key	Summary	Progress	Status	Progress	Assignee	Σ Time Spent	Σ Original Estim	Formula: Over work
INIT-1	Core Products		IN PROGRESS	37%	Sarah Johnson	11w 6h 30m	24w 4d 4h	
INIT-3	Structure		IN PROGRESS	37%	Eugene Lebedev	11w 6h 30m	24w 4d 4h	
STR-1	Formulas		IN PROGRESS	55%	Eugene Lebedev	1w 2d 1h 30m	1w 3d 4h 1w	
STR-4	Support for JQL		DONE	100%	Mohammed Attar	2d 3h 3h	1d 2h 2h	Over work estimate
STR-10	Design Syntax		DONE	100%	Bob Smith	2d	1d	Over work estimate
STR-5	New UX for editing of large formulas		IN PROGRESS	51%	James Stevens	2d 6h 30m 1h	2d 2h 2h	
STR-11	Design UX for long formulas editing		DONE	100%	Bob Smith	2d 5h	2d	Over work estimate
STR-12	Implement new UI		TO DO	0%	Sarah Johnson			
STR-3	Sync Attributes to Custom Fields 3		IN PROGRESS	55%	Jacques Durand	8w 4d 5h 2w	12w 3d 6w	
STR-6	Technical Debt		DONE	100%	Sarah Johnson	2w 3d	1w	Over work estimate
STR-9	Scheduled synchronization		IN PROGRESS	30%	Sarah Johnson	4d 5h	3w	
STR-7	Mapping Mechanism		DONE	100%	Sarah Johnson	2w	1w 3d	Over work estimate
STR-8	On-demand synchronization		IN PROGRESS	100%	Sarah Johnson	1w 2d	1w	Over work estimate
STR-2	Notes Column		IN PROGRESS	7%	Bob Smith	4d	10w 3d 9w	
STR-13	Support multiple Notes columns		IN PROGRESS	50%	Bob Smith	4d	1w	
INIT-2	Extensions		IN PROGRESS	20%	Mohammed Attar	12w 4d 5h	57w 3d 2h	
INIT-5	Structure.Gantt		IN PROGRESS	44%	Mohammed Attar	12w 1d 5h	16w 3d 2h	

Add Jira fields you need as columns, and view the information you need in one place

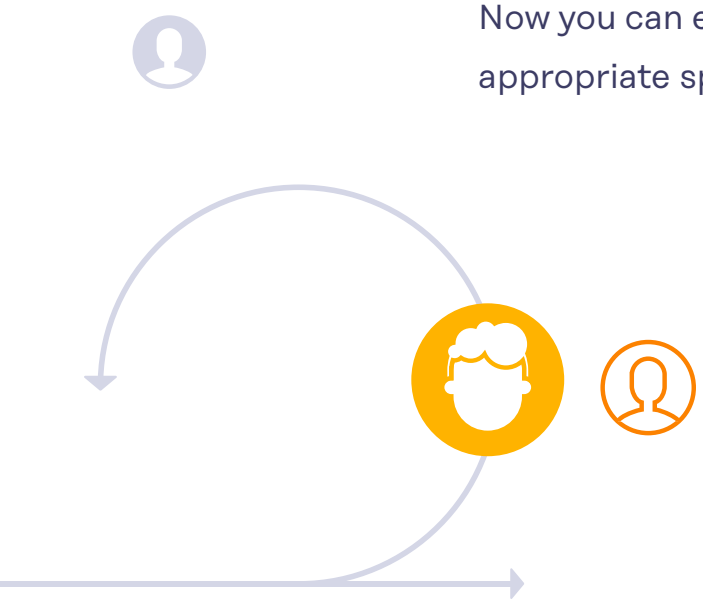
Example #2:
Sprint planning across teams

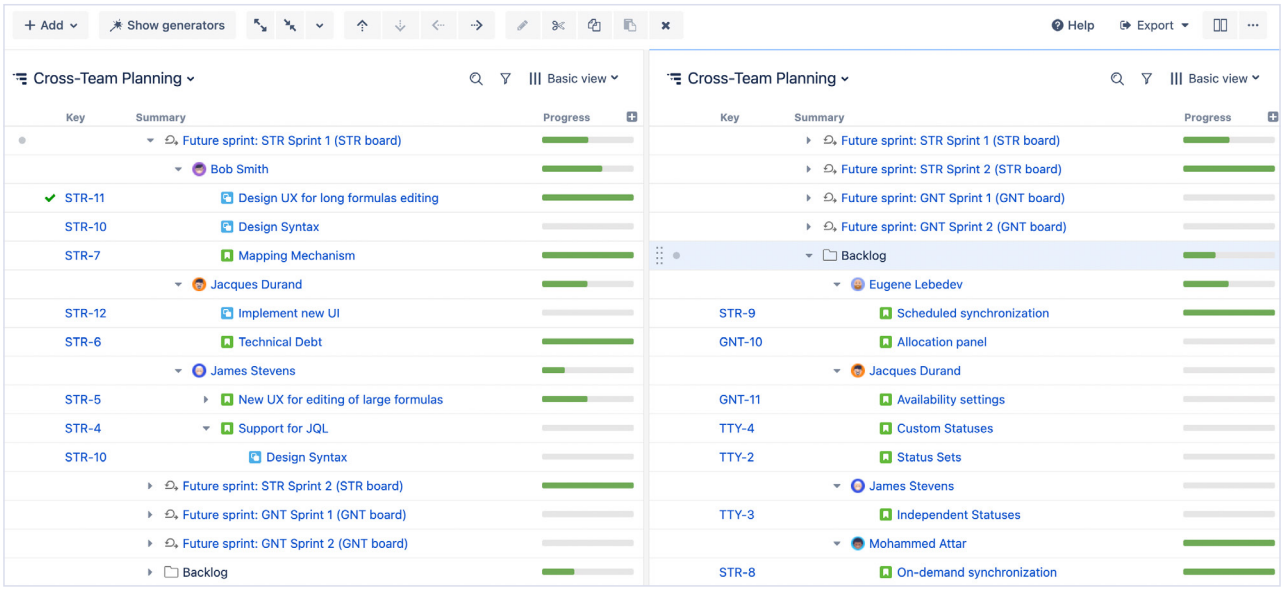
When you have a single Scrum team, it's relatively easy to stay focused on the current sprint, and maybe the next. However, in many real-life situations with cross-team dependencies you may need to plan for several iterations.

How to manage this? Create a structure specifically for sprint planning.

In this case, use JQL to add multiple teams' stories to your structure first. Use the Group Generator to see how they're distributed between different sprints — now, you can look at the Story Points totals column or time estimates (or both) to see how much work has already been committed to for each sprint.

Now you can easily move stories from the backlog to the appropriate sprints without overcommitting.



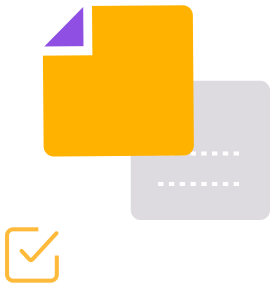


Choose Double Grid layout to get the most out of your data: the left view is for planning; the right view is the backlog

If you want to go to a level below the stories and check the work allocation by team members, you can add another level of grouping and see if anyone is overbooked. You can also go a level above and group your stories by the fixVersion field to see how much work is planned for each release.

This setup allows you to simplify and speed up your sprint planning sessions, giving you access to all the information you need in a compact view. It also gives you the opportunity to change fields easily, using in-line editing or drag-and-drop.

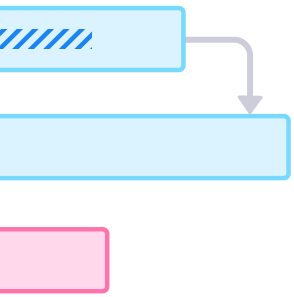
As users become more familiar with Structure, they start discovering new applications for it and new ways to improve their processes. It may even become one of the primary ways in which they interact with Jira. It establishes a new home base to work from.



Using Gantt charts for Agile teams

Many organizations find themselves split between Agile development teams' just-in-time sprint planning and business teams' longer-term planning needs. Even though Agile methodologies are the new standard in many industries, some projects simply require strict schedules.

To address scenarios like these, Structure has an extension called Structure.Gantt — a flexible tool that allows you to tailor traditional Gantt charts to your specific development model, including sprint-based Agile development. These charts help everyone visualize the projects' scope and identify bottlenecks and dependencies.



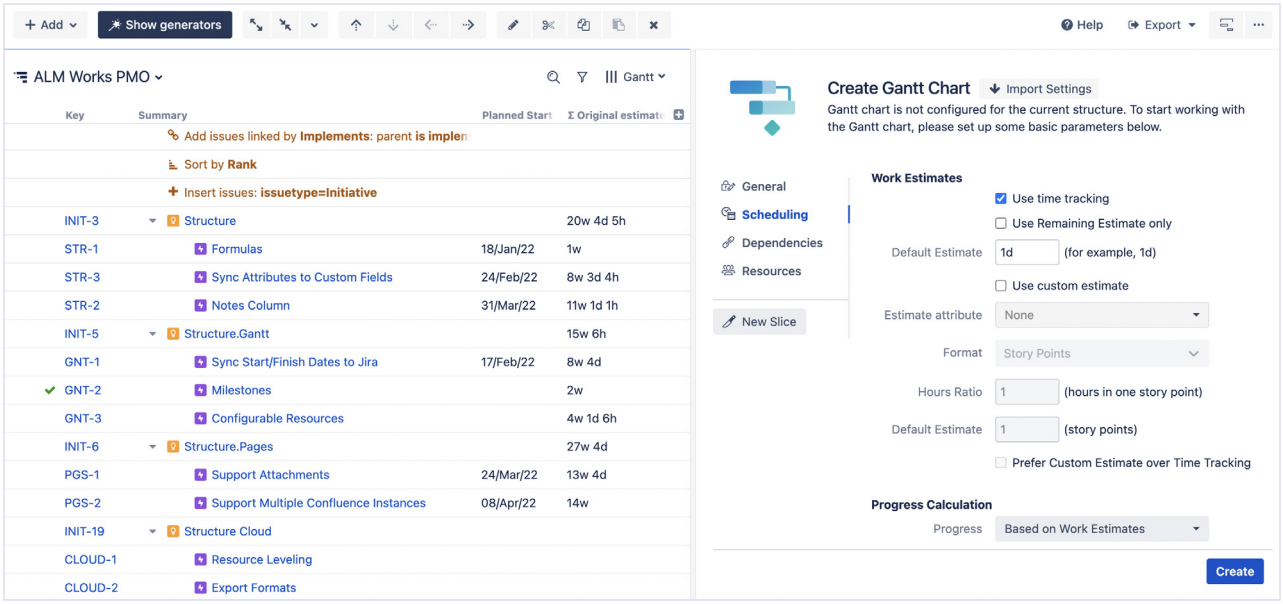
Structure.Gantt allows you to create timelines directly in Jira, without having to export things to external tools. That real-time connection to Jira helps stakeholders track progress and see potential delays quickly, build more realistic plans, and make scope or team adjustments to ensure you meet your deadlines. Gantt charts are also very helpful in terms of building reports that aid decision-making.

For this example, we'll start by showing how to build a portfolio roadmap for our projects. Afterward, we can dig into some Agile-specific details.

Example #1:
Gantt portfolio roadmap

First, build a hierarchy to represent a work breakdown structure similar to the examples mentioned above. We want to see the initiatives we'll be working on and the epics that implement them.

Next, configure the Gantt chart. Define the start date and choose which calendar to use. Then, select which Jira fields to use for the tasks' work estimates, along with start and end dates.



Structure.Gantt is flexible enough to adapt to your Jira project configurations, even if they differ across projects

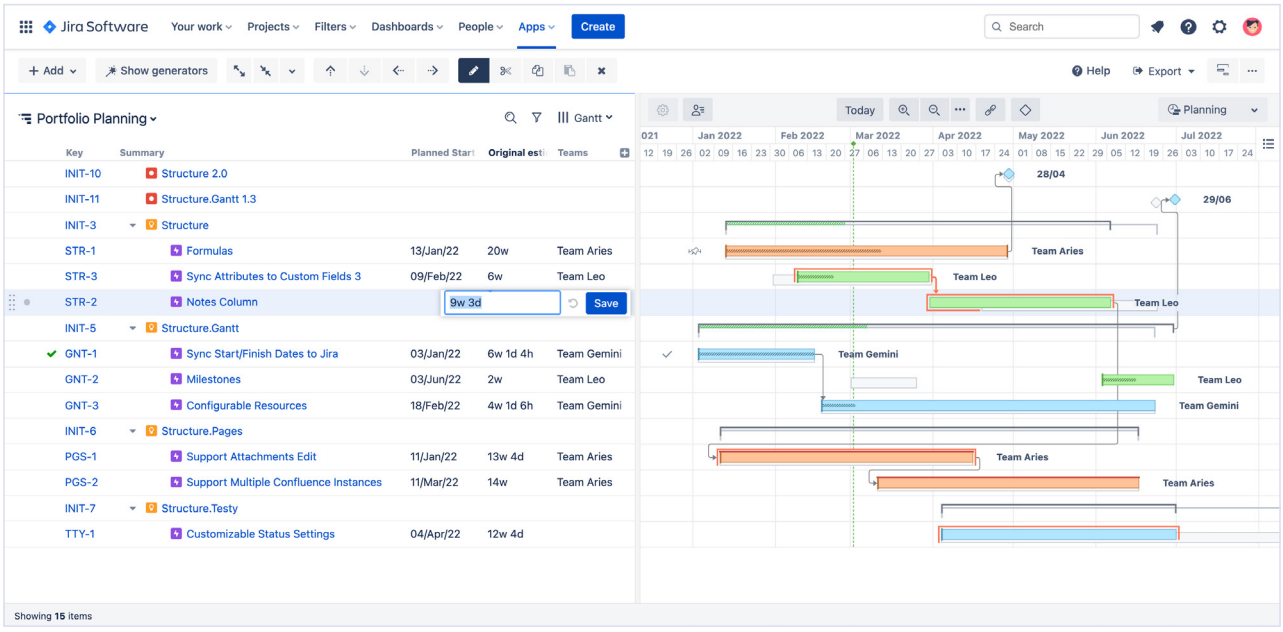
Visualize dependencies. Structure.Gantt uses Jira issue links for dependencies. You can choose which link types represent which types of dependencies. In this example, we've used the Blocks link for our Finish-to-Start dependencies.

In the resulting roadmap, you can make adjustments and calculate data in a variety of ways:



- Adjust the roadmap by making changes in the chart itself, or by changing the fields and links in Jira.
- Calculate task duration based on the estimate and the capacity of the resource assigned to the task, or simply have fixed-duration tasks defined by start and end dates.
- Resolve scheduling conflicts (conflicts are highlighted in the chart).
- View milestones to better understand whether your work is on track or see projected finish dates.
- View resources assigned to the tasks in your Gantt chart, as well as their total allocation. You can define an individual resource's capacity, schedule, time zone, and

availability, and the Gantt chart will calculate resource usage based on these settings and the tasks assigned to the resource, helping you avoid over-allocation of your resources.



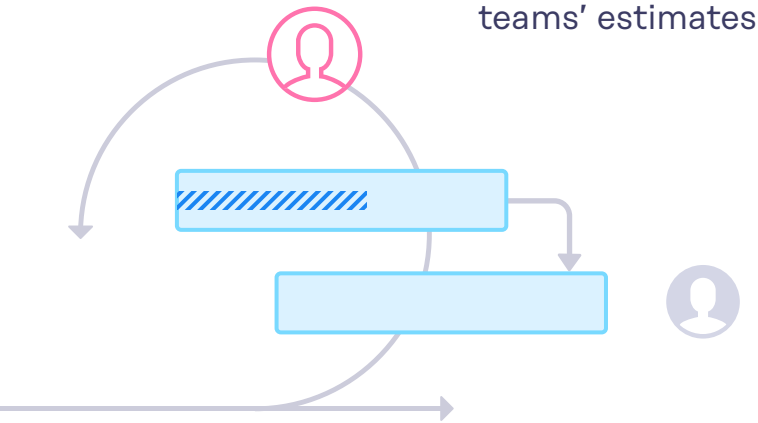
View issues alongside project timelines: Edit Jira issues in-line and watch the Gantt charts update in real-time

Example #2: Gantt for Agile teams

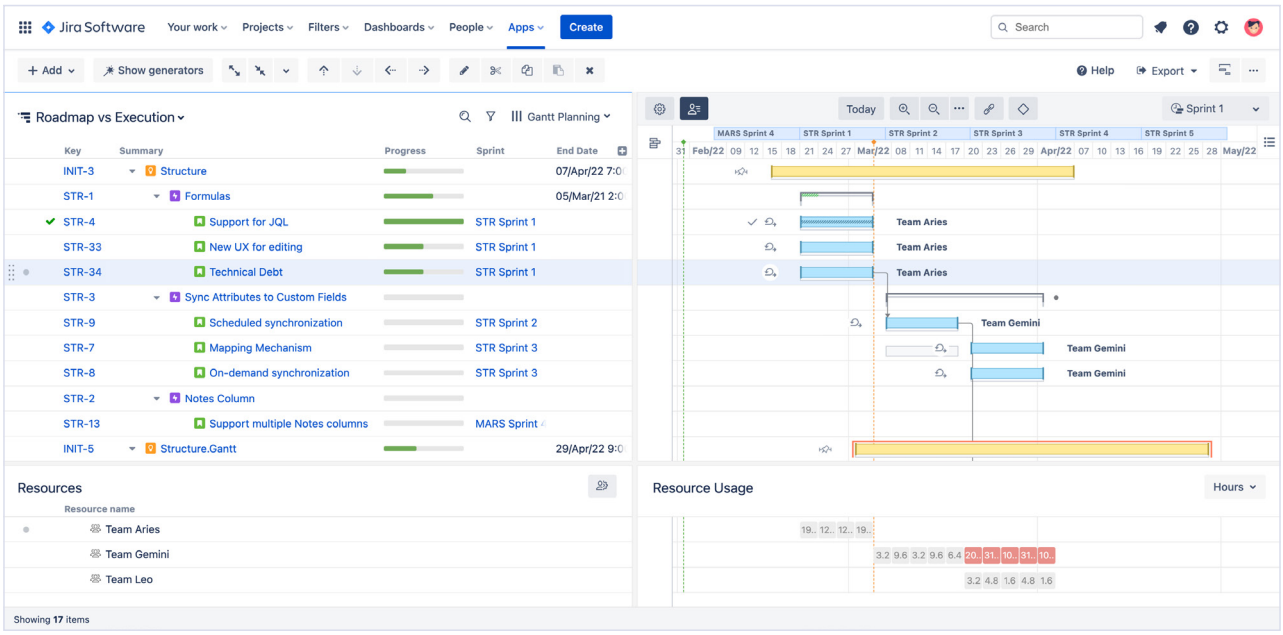
Now, let's bring the Agile teams into the picture and build a roadmap that will serve as a useful guide for all of our teams. There they will be able to see what they're working towards, which high-level items are next and what dependencies they need to keep in mind. These are useful bits of information the team can access as needed to make decisions about the breakdown and development of the stories that make up the epic.

As teams break down epics into stories, we can tweak some of the settings in the Gantt chart to enable in-flight tracking vs future planning.

Because Jira functions as the single source of truth for projects, teams can use both Structure and their regular Jira boards to groom the epics into their constituent issues, schedule sprints and perform their regular tasks. Project managers can track stories coming from those sprints asynchronously and proactively see whether the teams' estimates line up with the high-level plan.



They can now start to manage stakeholder engagement, adjust plans and deadlines, add more resources and reprioritize where necessary without needing to schedule update meetings with the team. The Gantt chart always shows you the current state of the project — you can always see any downstream impacts from changes you make.



Perform high-level and individual retrospectives to better spot improvements for future projects

Finally, once a project is done we can compare actuals against the estimates (which are still attached to the epics). We can perform high-level and individual retrospectives to learn where things went well and where improvement is needed.

Flexible solutions for complex PPM

As projects grow in complexity, they require additional support to aid project managers' efforts. But whatever tooling you use, choose apps that are flexible enough to adapt to your organization's specific needs. Structure for Jira — along with Structure.Gantt — were created with this flexibility and scalability in mind.

If you'd like to try Structure for yourself, install an evaluation and give it a try.

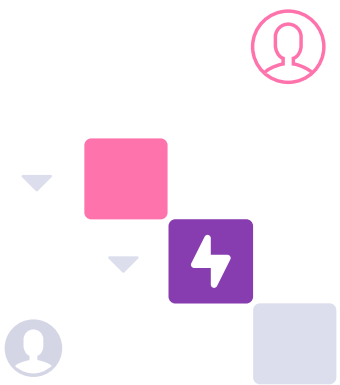
alm.works/structure

And you can always contact our support team if you have any questions or need help optimizing this solution for your organization.

Note: All images in this document show the cloud version of Structure for Jira, but the information applies to both cloud and on-prem platforms.

ALM WORKS

ALM Works empowers project managers with visibility and control. The company's flagship product — Structure for Jira — is a project portfolio management solution for Atlassian customers, currently used by organizations such as Apple, Daimler, NASA, Netflix, and Visa. ALM Works became part of Tempo Software, Inc. in December 2021.

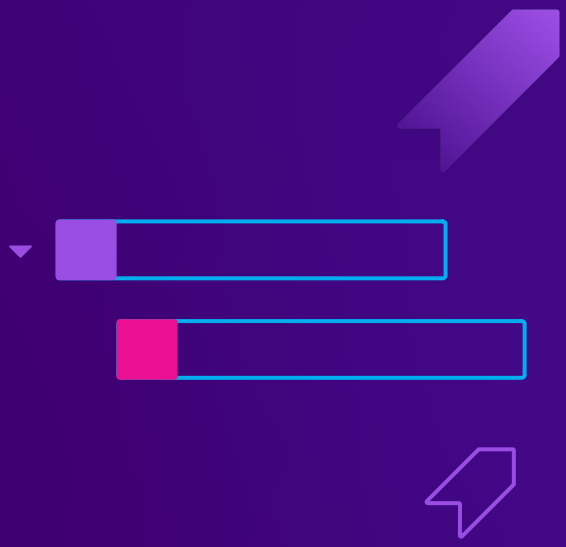


Contact Us

almworks.com

info@almworks.com

+1 617 600 4369



ATLASSIAN Platinum Top Vendor

Inc. 5000

Software Reviews GOLD MEDAL 2021 Application Lifecycle Management

SCALED AGILE PARTNER PLATFORM