

Managing Jira at Scale

How to Optimize Jira as Your Company and Usage Grows



Rodney West
Professional Services
Lead Consultant
Isos Technology



Platinum
Solution Partner
ENTERPRISE

Managing Jira at Scale

How to Optimize Jira as Your Company and Usage Grows

Contents

- Jira Instances at Scale
 - Determining the Right Environment: Cloud vs. Data Center
 - A Single Instance vs. Multiple Instances
- Jira Governance at Scale
 - Establishing a Governance Oversight Committee
 - Creating Jira Template Projects
 - Evaluating Feedback and Change Requests
- Jira Administration at Scale
 - Establishing a Team of Administrators
 - Requesting Changes: Creating a Jira Admin Requests Project
 - Scaling Jira Administration with Jira Automation
- Managing Third-Party Marketplace Apps at Scale
 - General App Considerations
 - Evaluating New Apps
 - Adding a New App to Your Jira Environment
 - Maintaining the App: Life After Installation
- Jira at Enterprise Scale
 - Advanced Roadmaps
 - BigPicture
 - Jira Align
- Resources and Recommended Reading
- About Isos Technology

Introduction

As organizations grow and mature, they often find that their business and operational needs change, and because of this, they need to adapt the way they use critical, underlying technology platforms. For companies that use the Atlassian platform, this often means scaling their use of Jira Software. At Isos Technology, we see a couple of scenarios fairly often. In one, leadership is seeking the strategic benefits of visibility into work, and synchronization in support of marketplace success. This may be part of a focused journey to adopt similar practices across their organization using scaled agile, or simply an effort to get better organizational reporting across their teams. In the other, teams that work in Jira every day grow frustrated by the functional and performance issues resulting from organic growth.

With organic growth, what often occurs is that different divisions, work groups, and teams spin up their own instances, or develop their own processes. Over time, users, projects, and custom fields proliferate. Decentralized administration is also common, contributing to the differences in the way teams work. Mergers and acquisitions add to the complexity too, as companies take on new instances, often in different environments, and new teams with divergent ways of working start to develop their own processes.

Eventually, even though the tools were designed to help teams collaborate, the abundance of seemingly small differences in process means day-to-day work is negatively impacted. Teams cannot collaborate, managers cannot report on the work their teams are doing, and the systems start to have performance issues. Companies reach a point at which they must transform their Atlassian tools, and Jira Software in particular, to meet their greater organizational needs.

Beyond scaling Jira to handle an increased number of users and projects or scaling the volume of work their teams can accomplish by leveraging efficiencies within Jira, many organizations also want to scale their agile practices. Whether they are using a scaled agile framework like SAFe, LeSS, or a homegrown version of these, Jira Software, as well as Advanced Roadmaps and Jira Align, can be an essential part of the underlying technology that makes this type of growth feasible.

Over the course of our many years as an Atlassian Platinum and Enterprise Solution Partner, Isos Technology has helped hundreds of organizations of all sizes optimize and scale their Jira instances. The process can seem overwhelming, and a good place to start is to determine what scaling means to your company and what your desired outcomes are. These may include:

- Gaining operational efficiencies by managing the tooling complexities that come along with a growing number of users and projects.
- Getting more value from your investment in Jira by maximizing use of its features and functionality, including automation.
- Synchronizing work efforts across a growing number of teams to support collaboration and reporting.
- Gaining visibility into what your organization is working on so you can make data driven decisions about where to focus your teams' efforts.

Once the goals are clear, there are several key decisions companies must make to put a solid foundation in place to future-proof their instance and set themselves up for long-term growth.

These include:

- Choosing the right environment and develop a roadmap for putting it in place
- Establishing a governance board and put governance processes in place
- Establishing a Jira admin team and change management processes
- Establishing practices for evaluating new apps and maintaining existing apps
- Configuring their instance for visibility across teams, effective planning, and reporting
- Preparing for and support adoption of agile at scale

In this whitepaper, we will explore each of these topics and how best to achieve success in your overall quest to scale Jira.

Jira Instances at Scale

Whether you are catching up with growth that has already taken place, or planning for future growth, the first thing to look at is whether your existing Atlassian environment meets both your current and future business needs. Then, you need to make sure that your organization is in the right environment, and that the environment is optimally architected for scaling. With support for Server ending in February 2024, the decision comes down to two choices: Cloud and Data Center. While most organizations will opt for Cloud, which is purpose-built for scalability, there are some key reasons that certain organizations will choose Data Center.

Determining the Right Environment: Cloud vs. Data Center

There's a lot to consider when deciding whether Atlassian Cloud or Data Center is right for your organization. While more companies than ever are opting for Cloud, some may have unique compliance, security, or data residency issues that mean they must remain in a self-managed environment. We should note that Atlassian Cloud is designed from the ground up for scalability, and it's



where the company is putting the lion's share of its development focus, so capabilities are advancing regularly.

If your organization is currently in a Data Center environment, and you think you may need to remain there, it's worth having a discussion with an Atlassian Solution Partner. Cloud capabilities may have advanced significantly since you last looked, or there may be workarounds a partner can put in place that would make Cloud a viable option for your organization. Even if it's not the right fit now, a partner can help you identify the determining factors for reaching the point at which Cloud is a viable solution. Then, the Solution Partner can set up your Data Center environment in a way that makes you cloud-ready when the time comes to make the big move.

Next, we'll take a high-level look at some of the ways Atlassian has architected its Cloud offering to support scalability and growth, and some of the things you will have to be on the lookout for if you opt for Data Center. Additionally, if your organization determines that Data Center is the best environment, there are steps you can take to architect your instance to support scalability. These are covered in our companion piece [How to Architect Your Atlassian Data Center Instance to Scale for Growth](#).

Following are several key points specifically related to scalability, which are important to consider when determining whether Atlassian Cloud or Data Center is right for you.

Infrastructure Management

Atlassian Cloud has several scalability-related benefits over Data Center. Not only do you not have to provision new virtual infrastructure to scale your instance, you also don't have to staff resources around maintaining or expanding that infrastructure. With Data Center, rolling out Jira updates is a manual process that requires planning and hands-on implementation. However, with Cloud this is handled by Atlassian, so you'll always be on the latest version and have the most recent patches and security updates.

Another Cloud infrastructure benefit is that it is easier to come up with different, more flexible instance network topology. For example, you can

do split provisioning with multiple instances, which can have security and cost benefits that we'll investigate in greater detail a bit later in this section. As a reminder, split provisioning is not an option with Data Center.

User Management

Atlassian [Access](#) is the company's purpose-built solution for user management in Cloud, and it streamlines user management in ways that make scaling more efficient and less risky from a security standpoint. Access integrates with your identity provider solution for managing user identity, like Okta, so it can serve as a single source of truth for your organization. Once you set up your access policies in the active directory, those structures populate in for Access, so you don't have to manually assign new users to specific groups, or manually delete users who have left the company. This makes onboarding and offboarding new users very straightforward, helps to minimize errors, and saves your admins an enormous amount of time.

Access also works across instances, which is important at the enterprise level, as some companies make a strategic decision to maintain multiple, distinct Cloud instances. We'll explore this in greater detail later on, but the advantage here is twofold in that both admins and users can work across multiple sites, and those users only need a single license to do so.

Atlassian has a separate product for user management in Data Center, [Atlassian Crowd](#).

Third-Party App Functionality

The lack of parity in functionality between Data Center and Cloud versions of third-party Marketplace apps used to be a significant barrier, one which companies would have to overcome when migrating to Cloud. In the early days of Atlassian Cloud, some companies discovered that the lack of feature parity, particularly around apps, would negatively impact productivity and labor costs. Fortunately, this is no longer the case, as the overall number of Cloud apps available has increased, feature parity has improved, and native functionality in Cloud has advanced. It might not be a 1:1 transition, and you may need to access the desired functionality in different ways, but in almost all cases, you can bridge the parity gap.



Compliance, Data Residency, and Security

Industry-specific and organizational compliance, data residency, and security policies remain one of the key reasons a company might choose to operate in Data Center instead of Cloud.

- **Compliance:** In certain regulated industries like finance or healthcare, there may be instances where either Atlassian or third-party Marketplace app vendors do not yet have the appropriate certifications. Every organization will need to undertake its own research to determine if Atlassian Cloud and/or any business-critical, third-party apps meet its compliance parameters.

To learn more, we recommend exploring the [Atlassian Trust Center](#), which includes a comprehensive [Compliance Resource Center](#).

- **Data Residency:** Data residency requirements may be industry- or organization-specific. Atlassian does offer [data residency](#) for its own applications within the U.S., E.U., Australia, and Germany at the Standard, Premium, and Enterprise Cloud levels. This is an evolving situation, and per Atlassian's [Cloud Roadmap](#), the company plans to add more regions. However, it cannot guarantee data residency for third-party Marketplace apps at this time, so you have to work directly with each app vendor to understand where its data is housed. However, this is just one of the challenges Atlassian's new app development platform, [Forge](#), is designed to resolve. Over time, as more apps are built on Forge, we anticipate that data residency will become less of a barrier.
- **Security:** Atlassian Cloud offers Enterprise-level security and monitoring, and for many companies, this will ease a significant burden, and oftentimes info-sec teams welcome it. However, security policies vary widely from organization to organization, and some info-sec teams will mandate that third-party applications like Jira be housed within their own firewall or, alternatively, some companies have contractual obligations they must meet with their customers that limit the use of cloud applications. Before deciding on Cloud vs. Data Center, you will have to coordinate with your organization's info-sec team to get a better understanding of their policies.

A Single Instance vs. Multiple Instances

As companies scale, they often find themselves with multiple instances. This is often due to “under the desk” implementations in which different divisions, departments, and teams each spin up their own instance, or because of mergers and acquisitions (M&A) that bring together multiple organizations, each with its own instance or instances. The downside to multiple instances arises when they are set up differently, in different environments, without common processes, workflows, and nomenclature. These instances take more effort to manage, they limit visibility into work being done, and teams cannot collaborate across them. If team members need to work in multiple instances, they may need to have seats in each one, adding costs, and they need to learn multiple ways of working. It can be costly, chaotic, time-consuming, and it's virtually impossible to scale.

Most companies will quickly reach a point where they need to either migrate to a different type of environment, for instance from Data Center to Cloud, and they need to merge multiple instances that are already in the same environment. Alternatively, they will need to both merge multiple instances and migrate them to the desired environment. While it may seem counterintuitive, there can be some advantages to remaining on multiple instances as organizations scale, which we will discuss later.

At this stage of setting your Jira instance up for scaling, you will need to roadmap your instance, which entails determining what your end state should look like. Whether that means merging multiple instances, or making a strategic decision to remain on multiple separate instances. Regardless of which direction you go, you will need to establish a governance process, which includes determining what your instance configuration needs to be and how you will maintain it going forward. After that, you'll need to clean up your instances, either so you can merge and migrate, or so you can get your existing instance operating smoothly.



Roadmapping Your Jira Instance(s)

Once you have determined what environment is right for your organization, either Jira Cloud or Data Center, you will need to envision your end state and create a roadmap for how you will get there. If you have multiple instances, this may entail merging them into a single instance, or migrating them to a new environment. This is a critically important phase of the process, as your end state must be aligned with your long-term, strategic business objectives. In other words, the way the technology is architected must support how you want to do business, both in the present and in the future. In addition, merging or migrating are complex, time-consuming, and full of risk. Because of these two factors, we strongly recommend working with an Atlassian Solution Partner who has deep domain expertise in these areas, because this is not an area where we recommend you go it alone.

There are other options beyond pulling everything into a single instance: **Atlassian Cloud has advanced to a point at which it is possible, and sometimes advantageous, to have multiple instances.** While it may not make sense to have teams who need to collaborate in separate instances, particularly if you need to have Portfolio-level visibility and reporting into the work, some companies make strategic decisions to have different divisions, different regions, or even different departments in different instances. Reasons for maintaining multiple instances may include the need to allow different groups to work with a high level of autonomy, greater flexibility, or increased control over data residency. We see multiple instances work well for different divisions or business units with distinct portfolios, or for engineering teams to be on one instance and corporate functions to be on another. As long as there is not a need for these teams to function in a synchronized manner for cross-functional visibility, separate instances can work well.

There can also be cost advantages to multiple Cloud instances, particularly at scale. In Cloud Enterprise, user management is done in Atlassian Access, and Access allows you to manage users across instances. This means that each user has

a single seat, but if needed, they can work across multiple instances while the organization pays for only one seat for that user. However, with third-party apps, licensing is based on the total number of users in your instance, whether or not that user has permission to use that app. So, if you have thousands of users on a single instance, but only a few dozen use a particular app, you still must pay license fees for that app for all thousand users on the instance. Split provisioning provides a work around for this, by allowing you to add apps only to instances that truly need them.

Whether you opt for a single instance, multiple Cloud instances, or even a hybrid of Cloud-Data Center set of instances, we encourage you to think creatively about how your instance is architected, and work with an Atlassian Solution Partner to roadmap it. This will help ensure that your strategic business needs are aligned with the outcome, and that your instance is configured in a way that supports scaling for the future.

Establishing Jira Governance

Once you have developed a roadmap for your instance, you will need to establish a governance plan, which is the process of determining what your configuration should look like, how it should function, and what type of information you need to get from it through reporting. Governance also refers to establishing processes around how changes to your instance are evaluated and made. Additionally, Implementing governance also typically entails aligning teams that do similar or shared work around consistent processes and nomenclature.

Activities associated with governance include establishing a governance steering committee, creating template projects so that new projects are set up properly, creating a project where users can request changes, determining how requested changes will be evaluated and implemented, and making sure changes are communicated to impacted users. Because this is an in-depth topic, we have dedicated the second section of this paper to its discussion. We introduce it here to help you understand it as a necessary step in bringing together multiple instances.



Cleaning Up Your Jira Instance

Once governance has been established, and prior to merging or migrating, every organization will need to do some cleanup of its existing instances. The reason is that over time, obsolete projects accumulate, and seemingly small differences in workflows and custom fields increase. Eventually, organizations start having system performance issues, teams struggle to collaborate, there is limited visibility into work being done, and reporting becomes challenging and inaccurate. Even though the tools were designed to help teams collaborate, the abundance of seemingly small differences mean day-to-day work is negatively impacted, and makes it nearly impossible to scale.

In cleaning up Jira, companies may need to complete an extensive list of preparatory tasks, including: deleting inactive users, consolidating user groups, and editing permission schemes; deleting or archiving old projects or issues; deleting inactive workflows and consolidating the remaining ones; deleting unused and duplicate custom fields; and auditing and cleaning up apps. You will also want to consider what data you need to keep and what can be archived, particularly if you are merging or migrating. Clearly, instance cleanup is an in-depth subject, and requires a lot of forethought and planning prior to execution. While some companies tackle cleanup on their own, many benefit from working with an Atlassian Solution Partner that can bridge the gap between high-level strategy and day-to-day tactics, ensuring that cleanup efforts are reflective of the governance you want to establish. A partner also brings best practices to the table and can get the work done quickly and efficiently, while also minimizing the burden on your admin team and disruption of day-to-day responsibilities.

You can find detailed guidance about how to go about this in our companion whitepaper, [Practical Recommendations for Cleaning and Organizing Jira, JSM, and Confluence](#).

Scaling Jira Data Center

If you are looking for additional information about how to ready your Data Center instance to scale, read our companion article: [How to Architect Your Jira Data Center Instance to Scale for Growth](#).

Jira Governance at Scale

Governance is an imposing word, one that may conjure up thoughts of layer upon layer of hierarchy, and rigid mandates about how work gets done. However, at its core, governance is simply about figuring out what your configuration should look like, how it should function, and what types of information you need to get out of it through reporting. Importantly, governance also provides a way to effectively manage change in your instance.

There is also a component of governance that necessitates aligning teams that do shared work around consistent processes and nomenclature. This is necessary because as companies scale, they typically need to align work more consistently around company-wide goals and initiatives. When teams are working more consistently, companies gain improved, high-level visibility, including trends across groups, and are better able to report on completed work.

For organizations that are currently or planning to go down the path of agile at scale, effective governance practices serve as guardrails that support visibility and synchronization while enabling teams to be self managing, all of which are necessary to successfully scale agile.

Implementing governance practices necessitates some compromise, but it is well worth the effort. It makes work more straightforward and efficient, increases collaboration and visibility, and minimizes system performance issues, all while being essential to scaling.

Establishing a Governance Oversight Committee

In implementing a governance plan, companies must first establish a governance oversight committee. Every organization is different, but in general, the committee should be comprised of a diverse group of people who reflect the different departments and teams who will be using the software, including operations, engineering, product, and Jira administration. There should be



a balance of high-level, strategic perspectives, with people who understand that Jira is a critical component of organizational agility. Additionally, they know that Jira underpins critical portfolio-level insights and processes, and they have a vested interest in keeping it that way. It's also important to bring a group of people who have a more tactical, day-to-day understanding of its use.

This committee is responsible for determining what the company's processes and workflows should look like, and making sure that its Jira instance is configured to reflect this. It is a significant undertaking, and many companies benefit from the guidance of a third-party advisor, like an Atlassian Solution Partner, who can provide guidance on best practices and help them align their processes to their overarching business goals. Once the committee has determined what your governance practices should be, it is their responsibility to build out template projects, and ensure that teams are set up to work within the appropriate processes. The committee is also responsible for reviewing and approving change requests, documenting changes, and communicating upcoming changes to the appropriate teams.

At this point, we want to note that users will need a place to request changes. Our recommendation for this is to set up a Jira project specifically for this, as it will be done at the administrative level, which we cover in-depth in Section 3: Jira Administration at Scale.

Governance Committee Responsibilities

- Establishing organizational processes and workflows
- Ensuring Jira is configured to support organizational processes and workflows
- Developing project templates and making them available
- Reviewing and approving change requests
- Documenting and communicating change requests

Creating Template Projects (with Reporting in Mind)

Atlassian has done a great job of creating a tool that allows users to create Projects in whatever way they want to. The loose coupling of Issue Types, Custom Fields, Screens, Workflows, and the other core components are what make Jira so flexible. Unfortunately, an asset can also turn into a liability. Without careful guidance, all the options available in Jira can quickly lead to a situation lacking in consistency, causing problems around cross-functional communications, reporting, and resource management. Additionally, large numbers of unneeded objects like Custom Fields are often created, which can degrade performance, increase the burden on administrators, and make migrations and upgrades much more complex.

The best prevention and mitigation strategy is for the governance oversight committee to implement project standards, and support their use with template projects. Template projects use Jira schemes to codify organizational processes, and **Template projects also inform the type of reporting you can do.** Because of this, it's important to consider up front what information your company needs from the portfolio level, down to day-to-day project management, and set up your projects accordingly.

There is upfront time to create template projects, and you will need to determine what Issue Types, Fields, and Workflows are core to your business. It is best not to distill everything into one or two project types, since different organizational units will have different ways of completing projects, with different needs throughout. While you will need to differentiate your objects, try to minimize replication, especially where custom fields are concerned.

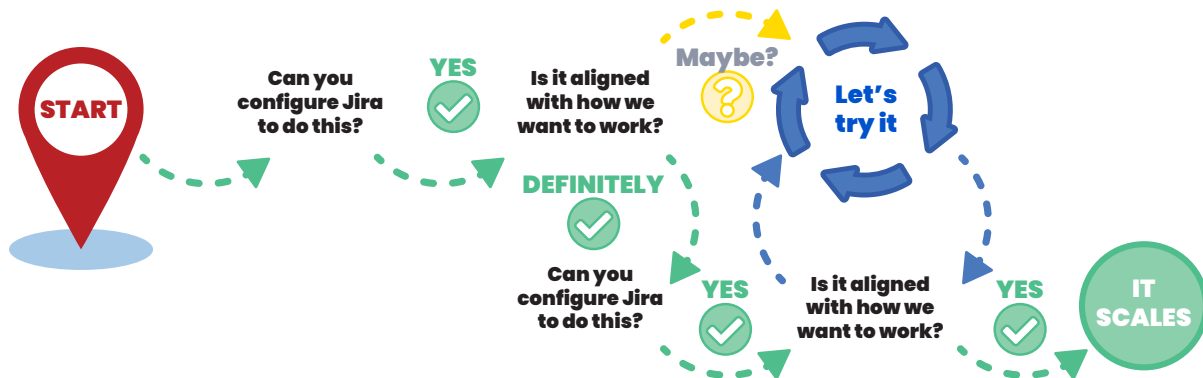
When you create new projects using these templates, in most cases you will want to choose the option to share the configuration with another project, usually the template project. This can be a double-edged sword in terms of creating and modifying project structures, as using this option makes it easier to create and modify project structures, but users will invariably ask for changes to their project. They may, for instance, need a custom field required or a new step in their



workflow. This can lead to either having to create a new set of schemes for a single project, or needing to poll the project teams using the shared schemes for impacts the change will have on their project.

This is where the governance oversight committee comes in. It's the committee's responsibility to make sure the guardrails stay on by determining and balancing what is beneficial to the teams that use the template project, while also ensuring that usable data can continue to be gathered from it by the people who need it.

Once the template projects are completed, be sure to add documentation to your knowledge base describing their purposes, the workflows they use, and the issue types and custom fields associated with them. This, especially in combination with proper training, will help users choose the proper project types when requesting a new project, and minimize the number of questions your Jira administrators will need to answer.



Evaluating Feedback and/or Change Requests

The governance committee is responsible for deciding whether feedback should be addressed or requests for changes should be made, and they need to develop criteria for evaluating those requests in order to do that effectively. The criteria should account for who is making the request, how invested they are in its impact, as well as what that overall impact is.

Evaluating Change Requests

The governance oversight committee will have given a good deal of thought to the configuration of your instance, and the setup of template projects. Because of this, changes to projects should not be made lightly. When evaluating change requests and determining whether to make them, there are several questions the committee needs to ask, either of the user making the request, or of themselves. These include:

1. Why is the change needed? What is the rationale for it?
2. How significant is the problem described by the user that this change will address?
3. How many people does it impact and how significant will its impact be on their work?
4. What is the impact of this change to the overall system?
5. If there is significant impact to the overall system, is there another way of resolving the problem with less impact?

Likely, the user who submitted the change request will provide an explanation of why it is needed and what benefit it will bring to their work. However, while they may also be able to explain why their immediate team will benefit from it, it will be up to the governance committee to determine if there are benefits to or negative impacts on other teams within the organization. One potential negative impact to be aware of is cross-project reporting that leadership might rely on, as this could be negatively affected.



Similarly, the governance committee will need to determine the impact of the change to the overall system and weigh that relative to its benefits. Jira admins on the governance committee can provide valuable input on the complexity of the change and its impact on the system, while other committee members can assess reporting and workflow impacts. A change with significant impact to the system and significant benefits might be approved, but it will have to be thoughtfully executed with change management top of mind. A change with low impact and strong benefits would likely be approved and implemented quickly.

Once the governance committee has done a thorough analysis of the impacts vs. benefits of the change request, a decision can be made to approve or deny it. Approved change requests should then be passed along to the admin team to implement and the change should be communicated to the impacted users. Keep in mind that it is worthwhile to establish timelines around both evaluating and implementing changes, as this will allow you to better manage user expectations.



Jira Administration at Scale

The flexible coupling of Jira objects is what allows for the high degree of customization and reuse that makes Jira such a powerhouse, but that level of customization can quickly get out of hand. Because of that, it will become important to limit administrative access to the correct roles as you scale.

Establishing a Team of Jira Administrators

Once you have your Jira instance environment up and running properly, and you have established a governance committee, one of the most important things you can do to maintain it is rein in the number of Jira administrators who can make changes to it.

Unfortunately, many companies have gone down the path of granting ad hoc Jira administrative privileges so team members can, “make just one change that I can do myself,” and end up with too many users with admin permissions, a growing list of system issues, and multiple high-priority system incidents (PIs). This is a recipe for disaster in any environment, even more so in a scaled one.

Companies need to develop and document clear policies on who can have Jira administrator access, and Jira administrators need to be well trained in their administrative capabilities so that they can be deliberate in how they evaluate requested changes to higher-level functions like modifying workflows and schemes, or adding custom fields.

Also of note: **as your company scales, you will need to dedicate a team to Jira administration**, as one single person cannot be available all the time, and will quickly become overwhelmed with user requests. You should also make sure to invest in continued training for the admin team since Jira is constantly evolving, and it’s important to keep teams up to date.



When identifying the Jira administration team, companies often stick to technical resources. While it is wise to have someone who can diagnose system problems on the team, it is also helpful to have people with backgrounds in business analysis. They can help Project teams design the best workflows, custom fields, and schemes, since this is where the majority of changes to your Jira instance will occur. It may also be helpful to have team members who have a background in managed services and QA.

Requesting Changes: Creating a “Jira Admin Requests” Project

With Jira administrator privileges restricted, your users will need a way to ask for changes, and Jira is the perfect place to do this. While you may create a couple of test projects to make sure your Jira production environment is configured properly, if you are implementing Jira for the first time or launching a migrated instance, your first “real” project should be a **Jira admin request project**. The reason for doing this before making Jira available to the larger user base is due to the fact that you will likely be getting a lot of requests early on.

This project can be used to handle all aspects of Jira administration, such as requesting changes and reporting problems. Users should, for example, be able to request new projects, custom fields, workflows, and any other functionality that falls under exclusive Jira administration functions. It is also a good idea to include training as an option users can request, since having readily available access to training will speed team adoption, help users work more efficiently, and minimize errors.

While a normal Jira project will suffice, a Jira Service Management project with simple, intuitive request types is even better. It will help users request the right functionality, so your Jira admin team can fulfill requests more efficiently and effectively.

For more information about Jira Service Management and its benefits, read our whitepaper: [A Modern ITSM Solution for an Evolving IT Landscape](#).

Scaling Jira Admin with Jira Automation

As Atlassian tools are adopted across teams within your company, both Jira admins and users often find themselves bogged down by high volumes of manual tasks, like closing issues once the last subtask has been closed, or closing all issues and notifying customers once a bug fix has shipped. [Jira automation](#) is a no-code rule builder designed to automate time consuming, repeatable tasks just like these. Global or project admins can use automation to create global and multi-project rules, so they don't have to create and edit rules on a project-by-project basis. It is easy to use, has unlimited applications, and can lessen the admin team's workload as adoption increases, freeing them to focus on higher level work.

Note: Jira automation has native functionality built into Jira Software Cloud, Jira Software 9.0 and later and Jira Service Management 5.0 and later. If you are not yet on a Jira Data Center version with this native automation functionality, and you can't upgrade to a version that includes it, you can download the equivalent [Automation for Jira](#) app from the Atlassian Marketplace to get it. It is free of charge and your Data Center license will be accepted.

However, please be aware that Atlassian has a stronger focus on enhancing and expanding automation functionality for their Cloud product, so that offering is more robust.

Managing Third-Party Marketplace Apps At Scale

Out of the box, Jira's rich feature set and configurability give it the ability to meet a wide range of organizational needs. However, as companies scale, they inevitably reach a point where they can benefit from Atlassian's extensive ecosystem of third-party apps to extend their Atlassian tools. Alternatively, if your instance has



grown rapidly, either through organic growth or M&A, you may find yourself in a situation where you have too many apps. If that's the case, you'll want to clean up your app environment by consolidating apps with duplicate functionality, or getting rid of those that aren't business-critical.

Once that's done, it is important to follow a deliberate process when evaluating and installing new apps in Jira, since **there is both a financial cost and performance impact to adding new, third-party apps.**

General Third-Party Marketplace App Considerations

Cost is an important consideration when evaluating new apps, and it can vary widely based on functionality, vendor, and number of users. Note that pricing is based on the total number of Jira users, not just the users who will be using the app. So, for example, if you have 1,000 licensed Jira users, but the HR department requests an app that will only be used by 20 people, the pricing will be based on the 1,000-user number, not just the 20. This is one reason why some companies, as they scale, make a strategic decision to run multiple Jira instances and avoid unnecessary costs, as we mentioned in Section 1 of this paper.

Evaluating Marketplace Apps

- Why is the app needed?
- Can Jira be configured to meet the need without an app?
- Can an existing app meet this need?
- Who else can use this app?
- Who will pay for the app?
- Who will own the app?

A second consideration to make when evaluating apps is the overall effect on **system performance**. Since every app you add to Jira uses the same system resources as the base Jira installation, they collectively impact overall system performance. Over time, you may experience degraded performance, whether from the impact of a few

resource-intensive apps or the thousand cuts of an overabundance of smaller apps. When looking into performance issues in a Jira instance, apps should always be one of the first parts of the environment that is scrutinized.

Another vitally important consideration when purchasing Apps is the need for **Jira administration**. While many of the features found in apps may be exposed to project administrators and normal Jira users, most apps require Jira administrator privileges to configure. Every new app that is installed has the potential to drain away time from your Jira administrator team over the course of the app's existence in your ecosystem.

Evaluating New, Third-Party Marketplace Apps

To ensure your Jira instance does not become cluttered with apps, it's important to wrap a good governance process around evaluating new ones. Requests for new apps should be submitted through the dedicated change request project, although as you scale, you may find you benefit from a separate project set up specifically for app requests.

When evaluating requests for new apps, there are several questions the governance oversight committee and requester should work together to address:

1. Why is the app needed?
2. Are there other apps already in use that might meet this need?
3. Can the app be used elsewhere in the organization?
4. Where is the budget for the app coming from?
5. Who will own the app?

The first question is often the easiest to answer, since a request for a new app wouldn't have been submitted without a business need. That said, there may be a way to configure Jira to provide the requested functionality without an app, or there may be similar apps that meet the same business need already in use by other teams. While the



requester may not be aware of these solutions, someone on the oversight committee should be. Likewise with the fourth question, as the requester may not know if other areas of the business can benefit from the new app, but the oversight committee will have greater insight.

In terms of budget, will the business unit be responsible for the cost of the app for the first licensing year and/or for the entire lifecycle of the app in your environment? Alternatively, is there a general bucket for app costs across the organization? The model varies by company, and there isn't a single standard. While having a business unit pay for the app may make them consider more carefully what they are requesting, it may also prevent them from asking for vital features they need in Jira. This can lead to lack of adoption of Jira and potential fragmentation of information across disparate tools.

Finally, establishing the owner of an app is important but often overlooked. While the easy answer is to say that the Jira admin team are the owners of all apps, this is seldom correct. While Jira administrators will be responsible for administering, they may not always know whether an app is still in use and how it is being used. The owner of an app should be a point of contact outside the admin teams who the admins can contact when they have questions on current business usage of an app. Owner information should be maintained, transitioning to a new owner when the current owner leaves the team or organization.

Adding an App to Your Jira Environment

Once the business case for an app has been approved, it is time to begin the process of adding the app to your Jira environment. We recommend that you **test the app in a non-production environment**. There are three primary things you need to look for: whether the app breaks anything else in your environment, the overall performance impact of the app, and whether it works for users the way it is intended to.

- **Does your app break anything?**
While vendors go to great lengths to minimize any negative impact their apps might have, it is inevitable that some apps may have unexpected interactions with other apps. You don't want to break existing vital functionality with the addition of a new app, so be on the lookout for anything you might need to troubleshoot before pushing the app into production.
- **Does the app impact system performance?**
All apps will have a performance footprint in your Jira environment. Carefully monitor your testing environment for any performance variations and be sure the impact is worth the gains before pushing the app into production.
- **Does the app meet the users' needs?**
Be sure to conduct **User Acceptance Testing (UAT)** from the business unit requesting the app. They will need to verify that it actually provides the functionality they're looking for, and it's not uncommon to find out that it doesn't.

Once UAT has been completed and everyone is comfortable with the results, it is time to **roll the app out to production**. As with anything else in a large-scale environment, make sure to have a common plan for this. You will need to let users know what app is going in, why it is going in, and which business unit requested it. If you are using Confluence or another system for internal documentation, it is also a good idea to add a page dedicated to it with information on functionality, the business owner, and a link to the Atlassian Marketplace listing. Having a page like this can be incredibly important over the app's lifecycle, both for its potential users and for anyone who needs to understand the full functionality of your Jira system. Additionally, including this information can even minimize requests for new apps over time, since users can search Confluence to see if there are existing apps already in use that may meet their needs.

A final word of caution when adding an app to production: **we do not recommend broad use of trial licenses of apps**. Though it is tempting to use a trial license while waiting for procurement or other administrative signoff, it can lead to issues down the road. If you install an app into production with a



trial license and approval or funding doesn't come through, the app will expire after 30 days. Then, the people who used it will raise tickets, resulting in administrative and communication issues.

Maintaining the App – Life After Installation

Once your app is in production, your users will need to follow the appropriate governance processes for configuration changes to it. They should create tickets in the Jira project where you manage apps, since only Jira admins will be able to make changes. Avoid the temptation to temporarily grant Jira admin access to users so they can configure their own apps. It is also a good idea to make sure that all tickets related to one or more apps use either components, labels, or a custom field. Labels are the easiest to use, but are the most error-prone without some form of label-fencing app to control which labels are available. A custom field is easy to set up and works well if the apps are part of the project used to govern other parts of your Jira environment, but it's not optimal. Project components are a great choice for a project dedicated solely to app governance.

It is best to see if you can align the licensing of new apps with existing apps, as it will make renewals simpler across the board. Prior to renewal, verify with the app owner that the business unit is still using it, and if they are not, check across the business to see if it is in use by any other users before deprecating it.

Before upgrading or removing an app, make sure to test in a non-production environment. This is especially important when upgrading, as there may be performance or functionality changes. Also be sure to communicate this to your Jira users and include an update in Confluence.

Jira Software At Enterprise Scale

As companies scale in size and complexity, and especially as they approach enterprise scale, they typically need more insight into work being done. They need visibility both vertically, in terms of how

work rolls up to key initiatives and value streams, and horizontally, across teams, so they can plan sprints and quarters, allocate resources, manage dependencies, and release software updates predictably. Atlassian has two great solutions, Advanced Roadmaps and Jira Align, that extend the functionality of Jira, help with planning and managing work across teams and at scale, and provide greater ability to track and report on progress. We also often recommend a third-party Marketplace app called BigPicture.

The first one, [Advanced Roadmaps](#), is a feature of Jira Software, and is a good solution for companies that are smaller or earlier in their agile journeys but need some high-level visibility into how work relates to key initiatives, how work being done by different teams is related, and how that cross team work is progressing. It also provides the ability to plan using what-if scenarios.

[BigPicture](#) works well for organizations that are transitioning their engineering and business teams toward an agile mindset and thinking and looking for a way for those teams to stay in sync. BigPicture provides a very flexible portfolio structure and is useful for organizations as they transition toward adoption of a scaled agile framework.

[Jira Align](#), on the other hand, is a separate Atlassian product, built on top of Jira. It's a more powerful tool that allows for deeper, more granular insights. It is better suited for organizations that are larger and/or further along on their agile journey, and it's especially useful for companies who are organizing their work around creating value, planning to adopt or have adopted a Scaled Agile Framework, and need underlying technology to support their agile practices.

There are a few other things that are important to note about both Advanced Roadmaps, BigPicture, and Jira Align. First, these tools pull live data directly from Jira, so the information you access is updated in real time. Second, your Jira instance will need to be properly configured to take advantage of the functionality these tools provide and so it provides the necessary, useful data. In the roadmapping section of this paper, we mentioned that when you are thinking through your instance and determining what your configuration should look like and how it should function, one of the things you need to think



through is what type of information you want to get out of it. If you need the type of high-level and/or granular visibility that Advanced Roadmaps, BigPicture, and Jira Align provide, you'll want to configure your instance so that you can use these tools effectively.

Advanced Roadmaps for Jira Software

[Advanced Roadmaps for Jira Software](#) is a feature within Jira Cloud Premium and Jira Data Center for strategically planning and tracking work across multiple initiatives, projects, and teams. In our view, Advanced Roadmaps is a strong solution for organizations in the earlier stages of their scaled agile journeys: ones that have not yet adopted a Scaled Agile Framework but have adopted agile practices across some teams and need greater visibility and planning capabilities than Jira alone allows for.

With Advanced Roadmaps, organizations can create plans that outline organizational goals and include milestones and key results. Plans show how multiple teams are working across multiple projects—they are a good way to understand how each team contributes to organizational goals. Once plans are in place, Advanced Roadmaps is useful for tracking and managing work: you can allocate resources, manage capacity, and see dependencies between teams. It also functions a bit like a sandbox for Jira in that you can run different scenarios to see the impact different choices have on reaching milestones or completing projects.

BigPicture

[BigPicture](#) can be used in two ways. There is a third-party Marketplace app that provides much of the functionality of a full BigPicture implementation, as well as a Marketplace app that connects to a separate, full instance of BigPicture that an organization may already have in place. In the latter case, a separate BigPicture license is required.

Both of these options provide organizations undergoing agile transformation. It can be used with a variety of scaled agile frameworks and is designed to optimize portfolio management by providing visibility into and reporting on all work being done across the company. It's both customizable and flexible, so it's a good fit for companies with complex needs, even those who may be using hybrid methodologies. With BigPicture, companies can plan and manage initiatives, set priorities, coordinate resources, track work progress, and identify and respond to issues. When your BigPicture instance is connected to your Jira instance, you get even greater flexibility, as well as the ability to plan by running "what-if" scenarios before updating your Jira instance.

Jira Align

[Jira Align](#) is an enterprise-level solution for organizations that are on a true scaled agile path and are using a Scaled Agile Framework like SAFe, Scrum@Scale, LeSS, DisciplinedAgile, etc. It is designed to help companies organize work around value streams and map high-level strategic objectives from programs and portfolios all the way down to the Jira story level. One of the great advantages of it is that everyone with access—executive leadership, managers, and engineers alike—can see in real time how their work fits into the big picture and what its current status is.

Jira Align has a broad range of uses for everyone at every level. Executives can see how departments are executing against business strategy; product managers can prioritize features in the backlog; program managers can see dependencies and manage risks; release engineers can allocate resources; and product delivery teams can execute sprints.

For either of these solutions, we recommend working with an Atlassian Solution Partner to make sure your Jira instance is configured to roll up correctly into these tools, and that Advanced Roadmaps, BigPicture or Jira Align is configured to give you the enterprise-wide insights you're looking for. In our experience, people, practices, and tools must all be in alignment to support scalability and enable your organization to achieve the best outcomes.



Resources And Recommended Reading

Atlassian Cloud

- [The Demystification of Atlassian Cloud: How to Choose the Right Plan](#)
- [The True Cost of Migrating to Atlassian Cloud](#)
- [The Essential Guide to Merging Instances When Migrating to Atlassian Cloud](#)
- [How to Migrate to Atlassian Cloud in Eight Steps](#)
- [Migrating to Atlassian Cloud? Seven Key Decisions to Make about Apps](#)
- [Cloud is Now! Overcome Migration Obstacles and Accelerate Your Transition to Atlassian Cloud](#)

Atlassian Data Center

- [How to Architect Your Jira Data Center Instance to Scale for Growth.](#)

Agile at Scale

- [The Roadmap to Enterprise Business Agility](#)
- [Leading through Change: the C-Suite's Role in Enterprise Agile Transformation](#)
- [Navigating Scaled Agile with SAFe and Jira Align](#)

Other Topics

- [Practical Recommendations for Cleaning and Organizing Jira, JSM, and Confluence](#)

About Isos Technology

Isos Technology accelerates the innovation curve for companies that are changing the world. Since 2005, we've partnered with hundreds of organizations to help them tackle tough business challenges and achieve their strategic goals. In November 2022, Isos became a portfolio company of The Acacia Group, joining forces with the Atlassian practice of MajorKey Technologies to form one of the largest Platinum Solution Partners in the Atlassian ecosystem. As an Atlassian Platinum Solution Partner with ITSM, Cloud, and Agile at Scale specializations, Isos helps organizations optimize their technology tools, develop effective practices, and guide people towards agility and resilience. Headquartered in Tempe, Arizona, and with offices across the U.S., Isos has been recognized as an Atlassian Partner of the Year in the ITSM and Enterprise categories for the last four years, an Inc. 5000 Fastest-Growing Private Company, and a CIOReview Most Promising Agile Consulting Company.

For more information, visit isostech.com.

