

AGILE AT SCALE:

**HOW TO MANAGE YOUR
INVESTMENT IN SOFTWARE
DEVELOPMENT**

February 2020

Overview

This white paper looks at the challenges of adopting Agile at scale with Atlassian tools: how to foster the adoption of SAFe within a large organisation; how to configure and support Jira and Confluence to work at scale; and some apps which assist Jira and Confluence to work well at scale.

Currently, most large-scale Jira and Confluence environments use a Data Center deployment, which is an on-premise hosting option that permits multiple parallel application servers. The cloud versions of Jira and Confluence are growing in capability but are not yet optimal for large scale deployments. Data Center is the main deployment option discussed in this paper, although many of its points apply to Cloud as well.





1 Challenges of Agile at scale

In the last decade, investment in software development by non-technology organisations has approximately doubled¹. Common challenges with any large investment is how to control the money spent, how to ensure that it is spent well and, crucially, delivers the desired return.

Agile software development was designed for smaller teams; the daily stand-up and fast customer feedback from a single Product Owner do not sit as easily in multi-billion-dollar organisations with global teams, as with the small teams that initially adopted Agile.

Atlassian's suite of tools, particularly Jira and Confluence, in conjunction with the array of apps available in the Atlassian Marketplace go a long way toward enabling teams to do their best work, even over long distances; however, co-ordinating a large number of Agile teams remains a significant challenge – and it can be difficult to know where to start. No journey to succeeding with Agile at Scale is the same, and there is no one-size-fits-all approach.

Automation Consultants meets you where you are on that journey and will design a solution to meet your organisation's unique challenges. Here we present an approach to support the Scaled Agile Framework for Enterprise (SAFe®), complemented by selected Atlassian tools and apps.

1. IDC ICT Spending Forecast <https://www.idc.com/promo/global-ict-spending/forecast> based on 7% annual growth.

2 SAFe[®] Implementation

SAFe is the most popular of several methodologies which have emerged to manage large numbers of Agile teams. It provides structures to manage Agile development from the basic development team all the way up to the C-suite.

SAFe can be implemented at three different levels, which cater for this range.

SAFe Level	Scale
Essential	A set of development teams that comprise one or more Agile Release Trains (ARTs).
Large Solution	A Solution Train (ST) consisting of multiple ARTs capable of building and maintaining a large solution.
Portfolio	A portfolio of large solutions.

An implementation of SAFe is not simply a matter of training the required number of people and expecting them to practise SAFe overnight. Further to training, individuals at every level of an organisation require coaching to have an Agile mindset throughout their daily work. To make this possible, management should exercise Lean-Agile Leadership in providing time and support to reorganise their teams around customer-centric value streams, rather than more traditional function- or department-based structures.

Adopting SAFe requires significant cultural change which can only be nurtured over time. In fact, a phased approach is hugely beneficial. It enables development teams and ARTs to form and refine their approach without disrupting existing teams not yet practising SAFe. Fortunately, even before the whole organisation achieves a level of maturity where all teams can benefit from the framework, the benefits of large-scale agility will be evident even if implemented in a few teams.



Stage 1: Team-level agility

For an organisation new to SAFe, Automation Consultants' approach is to work initially on a small scale. For teams new to Agile, we offer two-day intensive workshops where our consultants will work with a handful of your development teams to get the most out of Jira's excellent team-level planning and reporting capabilities.

Once these teams are succeeding in their sprints, managing dependencies between teams and utilising Jira's standard Agile team reports effectively, they'll be in good stead to take advantage of the benefits of the foundational Essential SAFe configuration.

Stage 2: Launching an Agile Release Train

For organisations that have already adopted Agile methodologies confidently at the team level, the next step is to launch a single Agile Release Train (ART). We can help your programme managers to identify a set of 6 to 12 development teams, who together, build and maintain a single value stream, most likely a product or service, such as a banking website.

Once an ART has been identified, we can assist in leveraging Jira and Confluence, along with the most appropriate apps, to make sure you get the most out of your first Program Increment (PI), and those thereafter. As we would be implementing the Essential level of SAFe at this stage, one app you should consider for your needs is *Portfolio for Jira*². It is suited to the scale of Essential level, and can be used to set up and prioritise Epics and stories, map dependencies and plan releases across multiple teams (Note: *Portfolio for Jira* is less well suited for larger scales than Essential).

2. We refer to the Data Center deployment option of Portfolio for Jira.

You may prefer a more flexible hierarchy, in which case *Structure for Jira* might be a more appropriate tool for your ART. This can be used to filter and sort custom hierarchies of issues with a myriad of options tailored to your approach. *Structure for Jira* also scales better to support the SAFe Large Solution and SAFe Portfolio level than *Portfolio for Jira*. To ease the process of PI planning and execution itself, we can help your Release Train Engineer and Product Managers get the most out of the Easy Agile suite of apps, which provide intuitive user story, program and roadmap planning. This suite includes *Easy Agile User Story Maps for Jira*, and *Easy Agile Roadmaps for Jira*.

Throughout the exciting process of conducting your first PI, Confluence truly comes into its own. We'll show you how to create templates for all your Agile ceremonies, pre-PI, PI and post-PI planning sessions, and pull through live Jira data to show the connection between Feature-level objectives and team level user stories.

Step 3: More ART launches, and beyond

Once your first ART is consistently and effectively leveraging the benefits of Essential SAFe, the methodology can be rolled out to other teams, starting with, perhaps, two more ARTs. These new teams can draw on the experience of the first ART to speed up the adoption process. Once the second set of ARTs is operating within the Essential SAFe configuration effectively, a third set of ARTs can be set up, where this time each set contains a greater number of ARTs. In this way, the usage of SAFe can grow organically within the organisation until all teams that can benefit are using it.

Remember, the work of some teams is inherently waterfall in its approach, and there is nothing wrong with this! A hybrid approach, especially in sectors with stringent regulatory requirements, is to be expected. The benefit of a scaled approach to implementing SAFe is that it will not disrupt the work of non-Agile teams, and two approaches can co-exist harmoniously.



Step 4: Large Solution, Portfolio and Beyond

Depending on the size and nature of your organisation, either the Large Solution or Portfolio configurations of SAFe may be the next step on your journey. A Large Solution, coordinated in the activity of a Solution Train directed by a Solution Train Engineer (STE), represents a value stream complex enough to require the work of several ARTs working in close coordination from one PI to another. The Portfolio configuration, or Full SAFe if it incorporates the Large Solution level, might represent the development activities of an entire division or even the entirety of your organisation. AC can help implement these stages of your journey to Agile at scale.



3 Atlassian Tool Configuration

3.1 Configuring Jira to support SAFe

As discussed above, an ART is a group of development teams organised around delivering the same solution.

While multiple teams can build backlogs and manage Scrum or Kanban boards in a single project, we recommend a dedicated project be created per team of 6 – 10 users. This is the simplest way to avoid confusion and devolve project administration. Seek to create these projects with shared configuration where possible or desirable – you may wish to encourage teams within an ART to utilise certain custom fields or follow some common workflows.

On the other hand, wherever possible you should empower each team to customise some aspects of the project's configuration as it sees fit. No two teams are the same and users tend to perform their best work when free to innovate in their own ways of working!

The launching of an ART will necessitate the creation of a new, dedicated Jira project to create and manage the Program Backlog. By default, stories in Jira are collected into Epics via the Epic Link field. However, this hierarchy is slightly at odds with a SAFe approach for two reasons:

1. It can be reasonably interpreted that, in Jira terms, the Issue Type directly above Story in a SAFe-aligned hierarchy is a Feature, and

2. an Epic as it is defined and used in Jira is not to be confused with an Epic in SAFe terminology, which represents a high-level strategic objective managed by the Portfolio team, as opposed to a large user story as it is understood in Jira.

We propose two equally valid approaches to reconciling this. The first solution is to replace all references to the word **Epic** in Jira with **Feature**. This can be achieved with a series of manual translation steps, but we strongly recommend the use of *Cprime's SAFE Epic to Feature Translator for Jira*, an app which can be used to safely execute the necessary steps without any risk to the integrity of your Jira instance.

The approach above may not be possible in your organisation, for example due to a multi-tenant environment with some teams not using SAFe methodology. Alternatively, your teams may find organising their work into Epics a beneficial intermediate way of breaking down complex Features into sizeable pieces of work to be managed at the team level. In either case the Epic issue type can be accommodated, as-is, between the Story and Feature levels of your work hierarchy.

Whichever approach is taken, the newly created Program-level project for your ART should have a Kanban board configured to manage the Program Backlog, and also track the progress of Feature translation into user stories leading to their implementation in Program Increments.



3.2 Configuring Confluence to support SAFe

Confluence is the perfect collaborative environment to capture the outcomes of Agile ceremonies such as pre- and post-PI planning, and the PI planning event itself. It is also an ideal platform for documenting and tracking Strategic Themes, and the breakdown of high level SAFe artifacts, such as Portfolio Epics, Capabilities and Features into their constituent work items, all enriched by live data from Jira.

We recommend the creation of a Confluence space for each Portfolio, Solution Train, Agile Release Train and Agile development team. The creation of a series of templates for each entity and process will ensure methodical capture of proceedings from Agile ceremonies, and clear and consistent definition of Strategic Themes, Capabilities and Features. Some recommended space and page templates are summarised in the table below.

Space Template	Page Templates
Portfolio	Strategic Theme Portfolio Epic
Large Solution / Solution Train	Capability
Program / Agile Release Train	Feature Program Increment PI Objective Pre-PI Planning Event PI Planning Event Post-PI Planning Event
Team	Team PI Objective Iteration Plan Iteration Retrospective

Automation Consultants offers a set of ready-made Confluence space and page templates for adaptation to individual customer needs.



3.3 Agile Reports at Scale

The reporting of progress and the delivery of value from a large organisation can be one of the biggest challenges of an Agile transformation. SAFe is an organisational and cultural framework for scaling lean and Agile practices but, being tool-agnostic, does not offer a prescribed template for reporting.

Similarly, Atlassian does not have a one-size-fits-all approach to visualising and measuring success; instead, it leverages a diverse ecosystem of third-party apps. The Atlassian Marketplace's **Agile at Scale** collection offers a flexible array of options to organisations wanting to manage and report their Portfolios at scale, with some of the apps being use case-specific, and others covering multiple aspects of portfolio management.

In our experience, organisations we've worked with implement a combination of these apps to meet their unique needs and preferences. Our SAFe qualified consultants can discuss your particular requirements and advise on which combination of solutions can work best for you as your Agile adoption grows and matures.

A key thing to bear in mind is that metrics and reports which are useful at one level and for one audience, won't necessarily be instructive to another. Team level metrics such as sprint velocity are terrific for helping your team consistently plan and deliver on their increment objectives, but make little sense when summed up across an entire Portfolio. We recommend you choose the right planning and reporting apps to separately serve your development teams, programme and Portfolio management teams, as well as other stakeholders.

4 Performance and Security at Scale

4.1 Performance

When scaling Jira, the only sensible deployment option to use is Jira Data Center, which allows you to run a distributed set of nodes which stay synchronised and failover in case one node goes down.

This infrastructure can be on-premise or hosted by third party hosting or cloud providers, such as AWS.

Even with the increased performance and reliability of a distributed solution, performance can be severely hampered by the solution being poorly maintained or configured. Poor maintenance or configuration can at best be reduced by increasing computational power, however, it will still cause a significant degree of slowdown.

The most common causes of poor performance are:

- scripts/automation;
- Jira misconfiguration;
- the number of issues; and
- an abundance of configuration items.

Each configuration item comes with a performance cost, but that cost can be increased unnecessarily by poor configuration of the item itself. In most cases the configuration types causing most problems are Screens, Workflows, Custom Fields, and Issue Types.

The number of issues cannot be reduced by a large amount in most cases, so efforts will have to be focused on optimising the configuration items.

The main approach here is:

- reduce the number of items;
- reduce Custom Fields' scope (within certain bounds);
- reduce the number of scripts; and
- avoid certain behaviours in needed scripts and scripted objects.

In scripts and scripted objects, you want to avoid the loading of data from sources such as REST APIs due to the amount of delay it will add to the page data loading. This delay will occur on any page containing the script until the data has been loaded, which might freeze pages for very long periods of time and can cause major problems if that data source goes offline. You should also avoid any recursive scripts or scripts that modify issues which are currently being displayed on the same page.

The reduction of configuration items is a multi-levelled problem. No solution is suitable for all instances due to the optimal standardisation between different teams and projects not always being in place. A set of guidelines and approaches is therefore given below for how to tackle this problem.

1. Find standards

What things are common between your teams/projects in the work process and in reporting? If nothing is common in your company then you will not be able to work in an Agile manner as inter-team co-operation will not be easy, efficient or reliable, as each team works to its own metrics and with its own definitions.

Therefore, it is recommended that standards are put in place for communication and reporting to ensure that everything is easily understandable across the entire business. With standards it is easy to go overboard and to be too strict, it is important to strike the right balance between flexibility and inter-operability to

avoid limiting teams which have special requirements, some of which might be regulatory in nature.

2. Put a small helpdesk in place

The helpdesk should be the gateway for users who wish to make configuration changes in Jira. All configuration change requests should go through the helpdesk, even those from C-level staff. This stops the admins from having to go through emails and keeping track of requests manually.

We recommend using Jira Service Desk for your helpdesk as it will allow you to write automation scripts which can create, modify or remove any objects you require. A set of clever scripts or scripted fields could also look for similar previously created configuration items and suggest those instead of creating new ones, or suggest switching to a new one and deleting the old one if modification request comes in. By automating this process you can reduce the amount of human error while at the same time reducing admin overhead.

3. Start doing regular maintenance and oversight.

A tool which can greatly assist you with this and with the above two guidelines is *Optimizer for Jira*.

Optimizer for Jira allows you easily to get an overview of your configuration items, how they are linked and their utilization. With its built in Health Check system, it will also highlight any major performance or other concerns with your Jira system. By using *Optimizer for Jira* to check the state of your system weekly or even daily, you can notice trends to nip issues in the bud, and easily clean up derelict configuration items and projects.

When these three guidelines are followed, the system should end up with a set of standardised core configuration items with simplified management and increased efficiency.



4.2 Secure handling of large volumes of data in Confluence

Large-scale Agile requires a large volume of documentation, typically stored in Confluence Data Center. Handling large volumes of Confluence data can be difficult from a security and compliance perspective. Technical details such as designs, IP addresses and more may need to be protected, as may budgeting and planning documents.

The app *Compliance for Confluence* helps protect sensitive data. It allows a customisable security scheme to be created, and then applied to all pages, either at the page or paragraph/content level. The default classifications are 'Public', 'Internal', 'Restricted' and 'Highly Restricted', but they can be customised to your organisation's standards or needs. Classification can be enforced on page creation or applied in bulk to large sets of pages.

Compliance for Confluence helps detect breaches of policy. It can automatically detect certain types of sensitive data, e.g. IP addresses, and it can be configured to take action on detection of the sensitive data. Actions include redaction of the data and reclassification of the page, but the action can also be customised by running a script. The sensitive data to be detected can also be customised, by using a regular expression. Any data that fits the regular expression will then be detected.

Compliance for Confluence helps securely manage the large volumes of documentation that go with large projects, and improves compliance by detecting sensitive data and acting on breaches of policy automatically.

5 Support

When running Atlassian applications at scale, good support is especially important. An outage in a large system can prevent 10,000+ people from working, and thus be very expensive – see our paper on Data Center [here](#). Consequently, a well-supported Jira deployment often has a much lower TCO (Total Cost of Ownership) than a poorly supported one, as the cost to the business of outages, performance problems and security issues grows with increasing size which outweighs the cost of support. With an excellent support service, unplanned outages can be prevented, the effects of any errors or issues can be minimised and your Agile teams' confidence in the Jira deployment will grow.

An effective support team will provide troubleshooting, typically through a helpdesk or service portal. Highly responsive troubleshooting of issues will help you to mitigate their effects quickly, and get your teams back to their work as soon as possible.

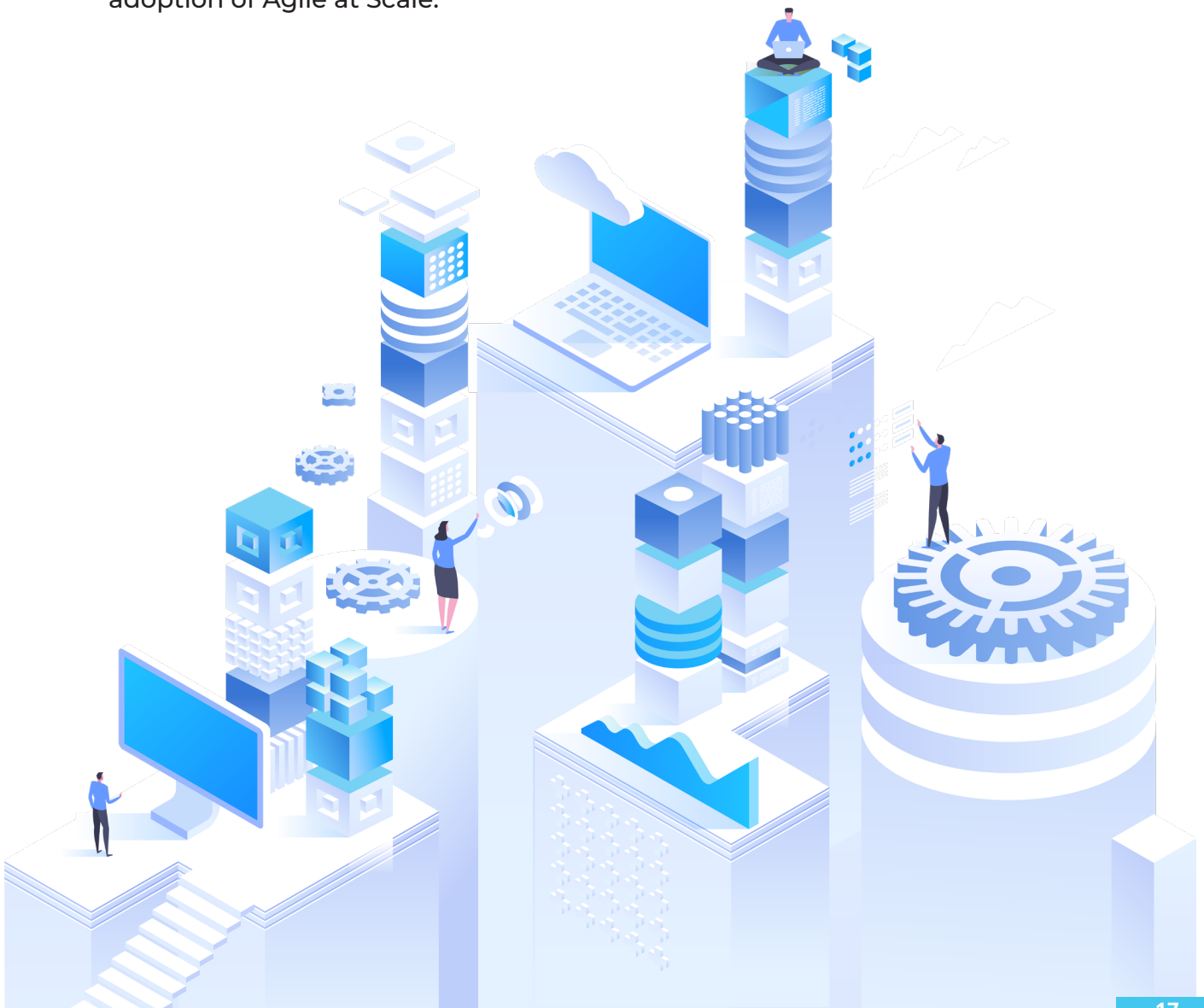
A support service should proactively monitor the system, giving alerts if signs of problems are detected. Well configured monitoring can help the support team spot issues before they affect users and prevent outages that would otherwise have negatively impacted the system. Monitoring also helps the support team optimise the application's 'under the hood' configuration and improve system wide performance.

Another critical function for a support team is performing application upgrades and maintenance. Atlassian regularly releases new versions of its software that introduce new features, bug fixes and security patches. Many of the new features and fixes can boost your team's productivity, such as the improved scrum and Kanban board load time introduced in Jira 8.0.



As more of your Agile teams adopt Jira and Confluence, the ever-growing quantity of sensitive data stored in the applications will make security a primary concern. Good support will ensure that security vulnerabilities are patched quickly and that your business data remains secure.

AC offers Enterprise-ready support that aims to keep your applications performing to the level that adoption of methodologies such as SAFe demands. We provide a 15-minute response time for critical issues and deploy a comprehensive suite of monitoring tools to resolve issues proactively before they impact your teams. We aim to help you build your team's confidence in the tools that support your adoption of Agile at Scale.



6 Conclusion

Successful Agile at Scale leads to Enterprise Agility, and with it benefits like faster time to market and less expensive, higher-quality software development.

Implementing Agile at scale is best done through an Agile framework such as SAFe, and it requires not only training, but cultural change within the organisation. Cultural change is best achieved on a small scale to begin with, followed by organic growth through the organisation, rather than by trying to make large numbers of teams adopt the change at the same time.

Successful Agile at scale requires scaled-up tools, such as the Data Center options of Atlassian Jira and Confluence. Configurations in Jira Data Center should be implemented to support SAFe, so that the artifacts in Jira match those of the SAFe process. Also, care must be taken to manage the creation of new configuration items in Jira by development teams so as to ensure a minimum degree of standardisation, and to control unnecessary or obsolete configurations. The controls should not be so tight as to stifle initiative within the teams, but tight enough to prevent a mass of configuration impeding reporting and adversely affecting performance.

Confluence is well-suited as a repository for the documentation associated with large scale Agile development. Templates can be developed to support the high-level artifacts in SAFe, such as Portfolio Epics. At the same time, the documentation should be kept compliant with relevant legal, security and regulatory standards, and tools exist to automate some of the tasks involved.



Crucial to successful Enterprise Agility is reporting. Effective reporting enables senior management to track the progress of large-scale Agile development and apply resources accordingly. Jira and Confluence can be configured to provide appropriate reports, provided that the necessary groundwork of standardisation has been done beforehand.

Finally, tools to support Agile at Scale should themselves be supported. Good support reduces total cost of ownership by pre-empting performance and security problems and preventing costly outages.

About Automation Consultants

We help organisations improve their business performance and enhance their competitive edge through digital transformation, automation and DevOps. We deploy the latest automated products and techniques with an emphasis on quality and deep expertise. We are uniquely placed to provide a complete solution for your organisation from strategic review, managed services and a leading array of Atlassian Marketplace Apps including Optimizer for Jira, all focussed on delivering the most value from your Atlassian platform.

Whether you require consultancy, training or support for your team, automation of your existing tools, we can help. To discuss our services and solutions, [get in touch](#) with one of our specialists today and transform your business at scale.

15+

Years improving
business
performance

600+

Happy
Customers
worldwide

Visit us at: www.automation-consultants.com

Get in touch: info@automation-consultants.com

Or follow us: [@AutoConsultUK](https://twitter.com/AutoConsultUK)

Automation Consultants
Merlin House, Brunel Road, Reading, RG7 4AB
+44 (0)118 932 3001
Automation-consultants.com
@AutoConsultUK

