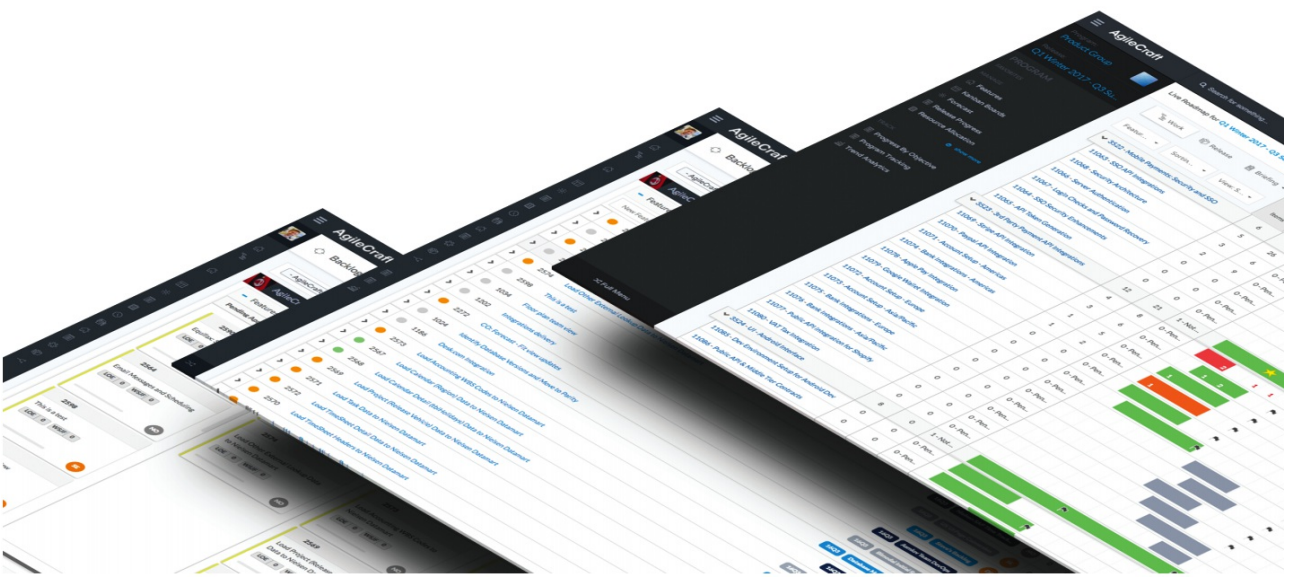
# Jira Align

## Azure DevOps integration guide

v.11.1.0 and above, August 2024



# Copyrights and Trademarks

## Copyrights

All of the content included in this documentation including all of the text, graphics, photographs, graphs, sounds, data, and images are the property of Atlassian or its licensors or content suppliers and is protected by US and international copyright laws. The compilation, collection, selection, arrangement, assembly, and coordination of all content is the exclusive property of Atlassian and protected by US and international copyright laws. You may not copy, publish, distribute, create derivative works of, or commercially exploit the content of this documentation, or use this documentation for any other purpose.

## Trademarks

“Jira Align” is a trademark of Atlassian. All other graphics, logos, product, and service names used in this documentation by Atlassian are the trademarks of Atlassian. Trademarks may not be used in connection with any third-party products or services or in any manner that disparages or discredits Atlassian. All other trademarks, brands, names, and logos appearing in this documentation are the property of their respective owners.

# Table of contents

# Integrations

# Azure DevOps integration

## Jira Align and Azure DevOps integration overview

The *Azure DevOps integration guide* is intended primarily for system administrators that set up tool integration. The guide includes sections that require you to examine your current Azure DevOps hierarchy to document your configuration and help us understand how it is structured. It may be beneficial to collaborate with a Jira Align Solutions Architect when conducting procedures and exercises contained within this guide.

Jira Align's Azure DevOps integration enables enterprises to plan their strategic work in Jira Align and have that work sent to Azure DevOps for execution by teams. As teams complete their work in Azure DevOps, detailed information, such as story and task completion, is sent back to Jira Align for reporting at the Program, Portfolio, and Enterprise levels. The bidirectional integration connection is managed by Jira Align Azure DevOps Connector.

This guide will help system administrators to seamlessly integrate the two tools, allowing mappings between the applications to facilitate data and workflow synchronization.

## Integration approach

Azure DevOps provides three default process templates: Agile, Scrum, and CMMI. Jira Align integration with Azure DevOps accommodates all three default processes. The default processes differ mainly in the work item types they provide for planning and tracking work as well as differences in work states. However, the processes share enough commonalities that a singular integration approach, defined in this guide, will work for all default processes. The integration will also support customized process templates, however, certain fields must be included to ensure the necessary data is copied between systems. See the fields section below.

For templates that do not inherit from the Agile, Scrum, or CMMI template, we cannot guarantee appropriate data will be available for reporting and other uses. With some older templates that do not inherit from the three templates or from the Basic template, the ADO API will not provide the data necessary for the connector to work. This problem is limited to ADO on-premise instances that were upgraded from TFS that used a non-inherited template.

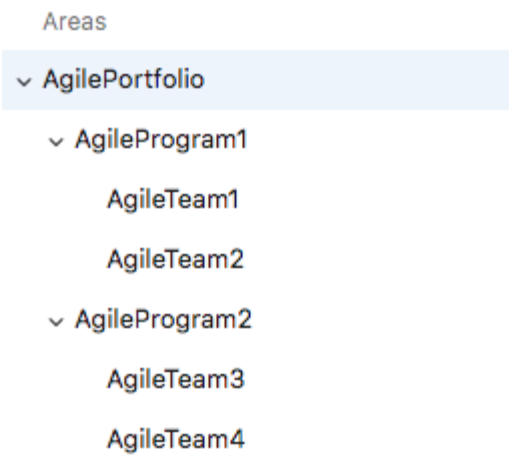
## Jira Align and Azure DevOps integration

Azure DevOps allows for hierarchical team area and iteration configurations. While there are an infinite number of ways these could be arranged and combined in order to successfully scale your organization and integrate with Jira Align, your organization must follow the Microsoft-recommended approach for scaling Agile teams in Azure DevOps.

© 2024 Atlassian — Confidential and Proprietary

At a high level this means:

- Areas
  - A project for portfolio-level work. Establish a portfolio-level area to aggregate your epics in.
  - Child areas below the project for program-level work. Establish a program-level area to aggregate your features in.
  - Child areas one level below the program for team-level work. Establish a team-level area to aggregate your stories in.




The configuration of the connector allows for programs to begin more than one level below the project level (named AgilePortfolio in the example above). The important things to keep in mind for configurations like this are:

  - All area paths that represent programs must exist at the same number of levels down from the top (e.g., AgileProgram1 and AgileProgram2 are both in level 2 in the example above.)
  - All area paths that represent agile teams must exist at the same number of levels down from the top (e.g., AgileTeam1 and AgileTeam 3 are both in level 3 in the example above)
  - All areas that represent agile teams must be below the area path that represents the program to which the teams belong.  For example, AgileTeam3 will be placed in the AgileProgram2 program in Jira Align.
  - More than one area path that represents a program may be mapped to a single program in Jira Align.

© 2024 Atlassian — Confidential and Proprietary

- Iterations
    - An iteration to represent program increments
        - The ADO path must have an end date that is the same (or within configured buffer days) of a Jira Align program increment
    - Iteration paths to represent sprints within program increments
        - Iteration paths that represent sprints must be descendants of the iteration path for the related program increment
        - Iteration paths that represent sprints must have end dates that are the same (or within configured buffer days) of a Jira Align program increment
        - Iteration paths that represent sprints within the same program increment should be on the same cadence and no two sprint iteration paths can have the same end date.
        - It is possible to use a staggered cadence for teams on the same anchor sprint by making cadence-oriented iteration paths at a level below the “anchor” sprints. When using this strategy, items created in Jira Align will initially be placed in the iteration path that is directly connected to anchors and teams can move them into the iteration path that represents the proper cadence without affecting the synchronization with Jira Align.
    - Program increment levels and anchor (sync) sprint levels must include end dates.

As with area paths, the important configuration need for iteration paths is consistency. Paths that will represent iterations in Jira Align must be at the same number of levels down the tree and be under the iteration path that will be linked to a program increment in Jira Align.

For more detailed information and setup instructions, refer to these articles:

- https://docs.microsoft.com/en-us/azure/devops/boards/plans/scaled-agile-framework?view=vsts
- https://docs.microsoft.com/en-us/azure/devops/boards/plans/configure-hierarchical-teams?view=vsts&tabs=previous-nav

When the Jira Align Azure DevOps Connector is configured and enabled, data is mapped automatically between Azure DevOps and Jira Align during a synchronization process. The connector works with the Azure DevOps .NET Framework API to deliver a near real-time, bidirectional integration.

# Key concepts

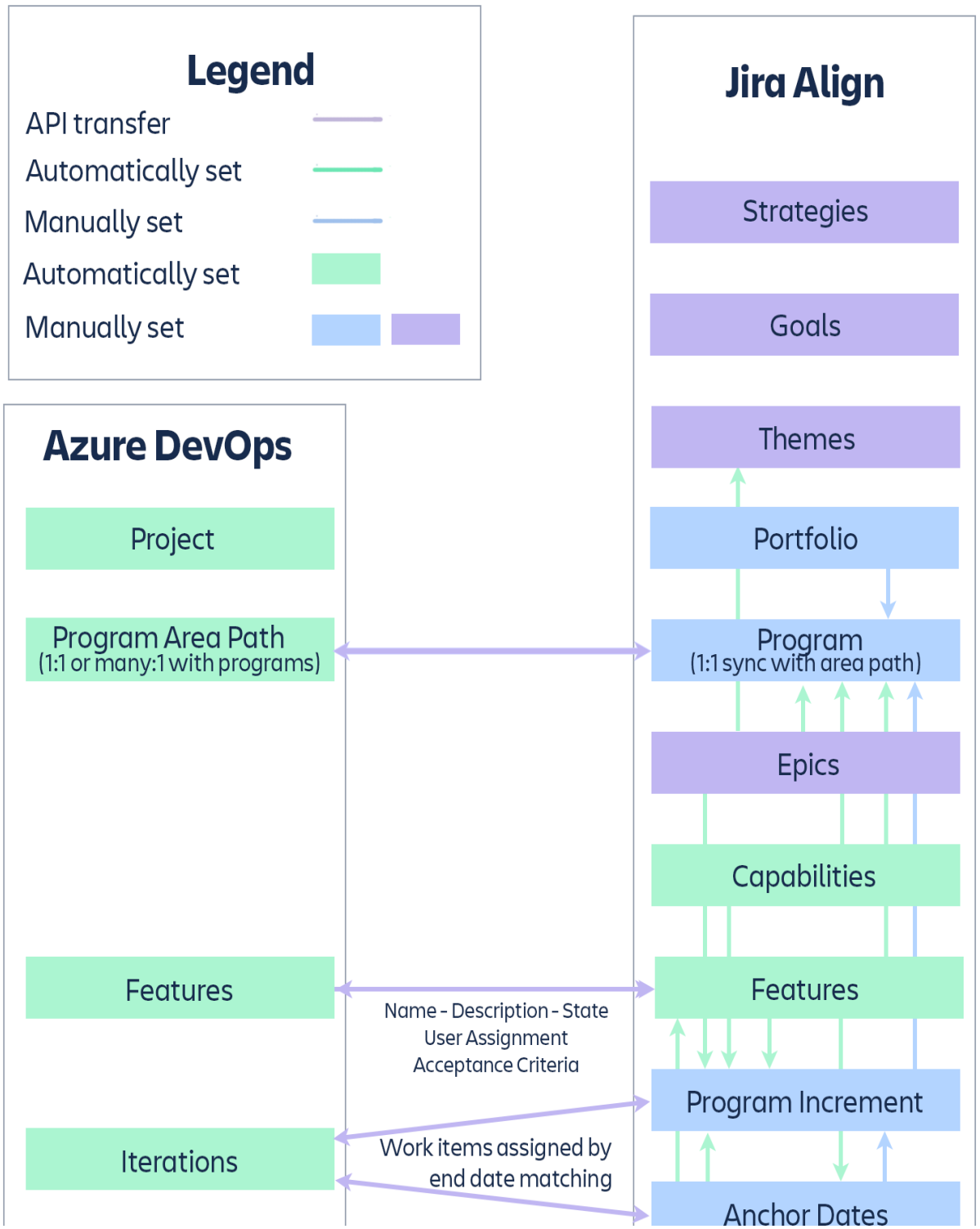
The following concepts are key to the Jira Align and Azure DevOps integration:

- Jira Align Azure DevOps data integrations are defined at the Project level within Azure DevOps.
- Portions of the area or iteration path hierarchy can be omitted from the integration using an “exclude” function during the source configuration.
- Azure DevOps area *paths* map to Jira Align teams no matter how Azure DevOps *teams* are configured.

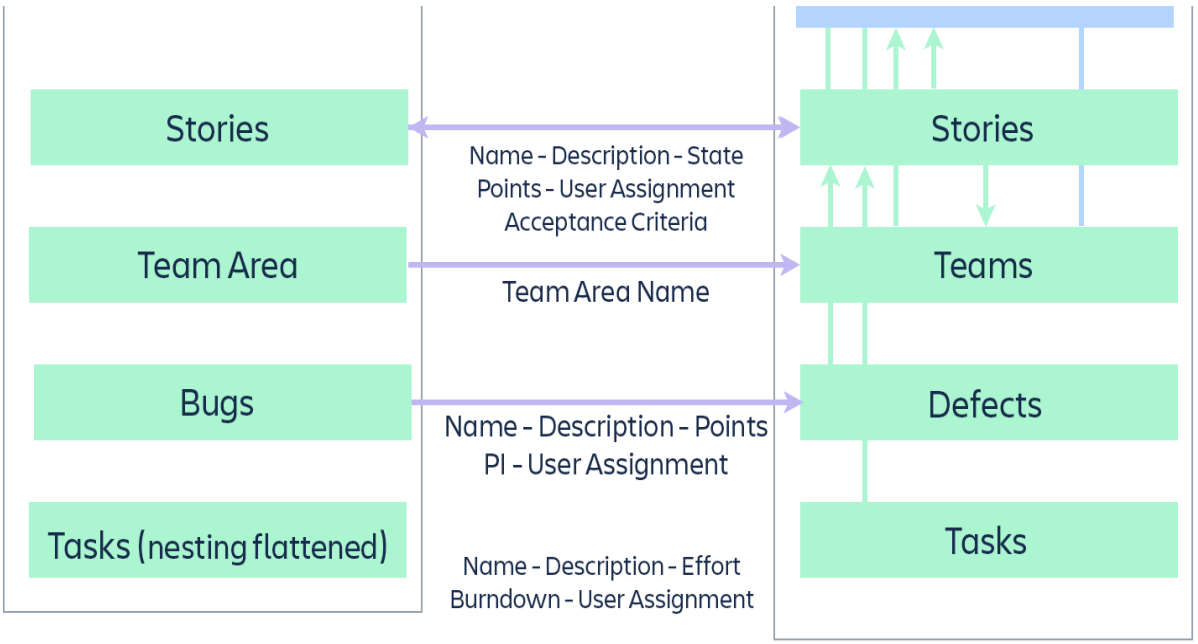
© 2024 Atlassian — Confidential and Proprietary

# Azure DevOps integration data mapping

The following diagram illustrates the general scheme of field mappings for mapping a Azure DevOps project to a Jira Align program.




© 2024 Atlassian — Confidential and Proprietary




Additional diagrams for team and portfolio mappings are located in the Appendix.

**State mapping and state movement restrictions**

Jira Align allows a transition to/from any state. Azure DevOps, on the other hand, can implement state transition rules about which states can be targeted from source states, and about fields that might be conditionally required. For example, if there have been defined some state transition rules that the Jira Align application violates, this can cause unexpected behavior related to replication.

Consider the following scenario:

A Product Manager moves a feature in Jira Align from state A to state B, and the connector attempts to replicate this in Azure DevOps. Azure DevOps doesn't allow this transition, so the state transition is rejected, but no message is sent to the user. Later, when the Jira Align connector discovers the feature with state A with Jira Align out of sync, it returns the feature in Jira Align to its prior state, state A. A Product Manager will discover that the feature returned to its prior state on the next screen refresh.

**In Progress state roll-ups**

Moving work that has not begun to the **In Progress** state will roll-up in Jira Align between related stories and features. When allowed, these changes will also sync to Azure DevOps.

For example, consider a story that has an associated feature. Both work items are in a **Not Started** state. When the story is changed to the **In Progress** state, the change rolls-up to the associated feature. If the story in this example is moved from **In Progress** backwards to **Not Started**, the feature remains in the **In Progress** state.

© 2024 Atlassian — Confidential and Proprietary

These roll-ups only take place up from child items to their parents. For example, changing a feature from **Not Started** to **In Progress** will not affect any associated stories. The audit log for a work item will indicate when edits or roll-ups take place.

## Work items

When you create a feature, story, or defect in Jira Align and select a mapped program, a new field will display — **Azure DevOps Area Path**. Select the Azure DevOps area path that represents your program in the field in order for the work item to sync. You will also need to specify a team in Jira Align for the work item to sync to the team-level area path in ADO.

When you create a work item in Azure DevOps, the connector will map the item to the correct program in Jira Align, based on your configuration settings. The **Azure DevOps Area Path** field will be set and displayed as read-only in Jira Align.

Moving a work item in either Azure DevOps or Jira Align affects these fields as well. When you move a work item into a different program-level area path in ADO, the **Azure DevOps Area Path** field will update in Jira Align upon the next sync. We recommend moving synced items in ADO instead of Jira Align to maintain or properly change mapped programs and area paths.

When you cancel a work item in Jira Align, the synced item in ADO will be marked as deleted and can be found in the Recycle Bin.

**Field mappings**

The following tables detail how data is mapped between Jira Align and Azure DevOps for each of the integrated objects represented in the data mapping diagram above. In addition to the fields below, all work items copied into Jira Align will have a field for External ID set to the Azure DevOps' key.

| **Azure DevOps features become features in Jira Align** | | | | |
|---|---|---|---|---|
| **Jira Align Field \| Azure DevOps Field** | **Create** | | **Update** | |
| | **JA to Azure DevOps** | **Azure DevOps to JA** | **JA to Azure DevOps** | **Azure DevOps to JA** |
| **Name \| Title** | Yes | Yes | Yes | Yes |
| **Description \| Description** | Yes | Yes | Yes | Yes |
| **Tags \| Tags** | Yes | Yes | Yes | Yes |
| **Owner \| Assigned To** | Yes | Yes | Yes | Yes |

© 2024 Atlassian — Confidential and Proprietary

| | | | | |
|---|---|---|---|---|
| **Status \| State**<br><br>**(Mapped via configuration setting)** | Yes | Yes | Yes | Yes |
| **Program Increment \| Iteration Path**<br><br>**(Mapped via end date)** | Yes | Yes | Yes | Yes |
| **Program \| Area Path**<br><br>**(Mapped via configuration setting)** | Yes | Yes | Yes | Yes |
| **Start/Initiation Date \| Start Date** | Yes | Yes | Yes | Yes |
| **Target Completion Date \| Target Date** | Yes | Yes | Yes | Yes |
| **Capitalized \| <Custom Field>** | Yes | Yes | Yes | Yes |
| **Type** | Yes | Yes | Yes | Yes |
| **Product \| <Custom Field>** | Yes | Yes | Yes | Yes |
| **Acceptance Criteria \| <Custom Field>***<br><br>**(Mapped via configuration setting)** | Yes | Yes | Yes | Yes |
| **Custom text input fields \| <Custom Field>**<br><br>**(Mapped via configuration setting)** | Yes | Yes | Yes | Yes |
| **Custom single-select dropdown fields \| <Custom Field>**<br><br>**(Mapped via configuration setting)** | Yes | Yes | Yes | Yes |

© 2024 Atlassian — Confidential and Proprietary

| | | | | |
|---|---|---|---|---|
| **Custom text area fields \| <Custom Field>**<br><br>**(Mapped via configuration setting)** | Yes | Yes | Yes | Yes |

* In programs that sync with ADO, you'll be limited to a single acceptance criteria text field in Jira Align since the default ADO acceptance criteria is a text field. If you already have multiple acceptance criteria in Jira Align for work items in these programs, the connector will automatically merge them into a single, formatted text field.

| **Azure DevOps stories and configured work item types* become stories in Jira Align** | | | | |
|---|---|---|---|---|
| **Jira Align Field \| Azure DevOps Field** | **Create** | | **Update** | |
| | **JA to Azure DevOps** | **Azure DevOps to JA** | **JA to Azure DevOps** | **Azure DevOps to JA** |
| **Name \| Title** | Yes | Yes | Yes | Yes |
| **Description \| Description** | Yes | Yes | Yes | Yes |
| **Estimate (Points) \| StoryPoints or Effort or Size**<br><br>**(Depending on template)** | Yes | Yes | Yes | Yes |
| **Team \| Area Path**<br><br>**(Case-sensitive name match)** | Yes | Yes | Yes | Yes |
| **Sprint \| Iteration Path**<br><br>**(Mapped via end date)** | Yes | Yes | Yes | Yes |
| **Parent Feature \| Parent Link** | Yes | Yes | Yes | Yes |

© 2024 Atlassian — Confidential and Proprietary

| | | | | |
|---|---|---|---|---|
| **Owner \| Assigned To** | Yes | Yes | Yes | Yes |
| **Status \| State**<br><br>**(Mapped via configuration setting)** | Yes | Yes | Yes | Yes |
| **Program Increment \| Area Path**<br><br>**(Mapped via end date)** | Yes | Yes | Yes | Yes |
| **Program \| Area Path**<br><br>**(Mapped via configuration setting)** | Yes | Yes | Yes | Yes |
| **Tags \| Tags** | Yes | Yes | Yes | Yes |
| **Type \| Value Area**<br><br>**(Mapped via configuration setting)** | Yes | Yes | Yes | Yes |
| **Acceptance Criteria \| <Custom Field>****<br><br>**(Mapped via configuration setting)** | Yes | Yes | Yes | Yes |
| **Custom text input fields \| <Custom Field>**<br><br>**(Mapped via configuration setting)** | Yes | Yes | Yes | Yes |
| **Custom single-select dropdown fields \| <Custom Field>**<br><br>**(Mapped via configuration setting)** | Yes | Yes | Yes | Yes |

© 2024 Atlassian — Confidential and Proprietary

| | | | | |
|---|---|---|---|---|
| **Custom text area fields \| <Custom Field>**<br><br>**(Mapped via configuration setting)** | Yes | Yes | Yes | Yes |

* Azure DevOps work item type Stories = Product Backlog Item in the Scrum process template, User Story in the Agile process template, and Requirement in the CMMI process template.

** In programs that sync with ADO, you'll be limited to a single acceptance criteria text field in Jira Align since the default ADO acceptance criteria is a text field. If you already have multiple acceptance criteria in Jira Align for work items in these programs, the connector will automatically merge them into a single, formatted text field.

| **Azure DevOps bugs become defects in Jira Align** | | | | |
|---|---|---|---|---|
| **Jira Align Field \| Azure DevOps Field** | **Create** | | **Update** | |
| | **JA to Azure DevOps** | **Azure DevOps to JA** | **JA to Azure DevOps** | **Azure DevOps to JA** |
| **Name \| Title** | Yes | Yes | Yes | Yes |
| **Description \| System Info** | Yes | Yes | Yes | Yes |
| **Parent Story \| Parent Link** | Yes | Yes | Yes | Yes |
| **Team \| Area Path** | Yes | Yes | Yes | Yes |
| **Sprint \| Iteration Path** | Yes | Yes | Yes | Yes |
| **Owner \| Assigned To** | Yes | Yes | Yes | Yes |
| **Status** | No | Yes | No | Yes |

© 2024 Atlassian — Confidential and Proprietary

| | | | | |
|---|---|---|---|---|
| **Program Increment \| Iteration Path** | Yes | Yes | Yes | Yes |
| **Program \| Area Path** | Yes | Yes | Yes | Yes |
| **State \| State**<br><br>**(Mapped via configuration setting)** | Yes | Yes | Yes | Yes |
| **Priority \| Priority** | Yes | Yes | Yes | Yes |
| **Severity \| Severity**<br><br>**(Mapped via configuration setting)** | Yes | Yes | Yes | Yes |
| **Steps to Reproduce \| Repro Steps** | Yes | Yes | Yes | Yes |
| **Tags \| Tags** | Yes | Yes | Yes | Yes |
| **Estimate (Points) \| Story Points** | Yes | Yes | Yes | Yes |
| **Estimate (Hours) \| Original Estimate** | Yes | Yes | Yes | Yes |
| **Remaining \| Remaining** | Yes | Yes | Yes | Yes |

In addition, when a defect in Jira Align is set to a state that maps to Closed in Azure DevOps, the Closed By field of the bug in Azure DevOps is set to the owner of the Jira Align defect; when a bug in Azure DevOps is closed, the Fixed By field of the Jira Align defect is set to the Azure DevOps Resolved By user.

If the Jira Align defect's State field changes to Fixed, the connector will also change its Status to Closed.

© 2024 Atlassian — Confidential and Proprietary

| Azure DevOps tasks become tasks under Jira Align stories<br>(tasks only synchronize from Azure DevOps into Jira Align) | | |
|---|---|---|
| **Jira Align Field \| Azure DevOps Field** | **Create** | **Update** |
| | **Azure DevOps to JA** | **Azure DevOps to JA** |
| **Name \| Title** | Yes | Yes |
| **Description \| Description** | Yes | Yes |
| **Effort (Hours) \| Original Estimate** | Yes | Yes |
| **Owner \| Assigned To** | Yes | Yes |
| **Status \| State**<br><br>**(Mapped via configuration setting)** | Yes | Yes |
| **Parent Story \| Parent Link** | Yes | Yes |

## Agile Teams

For each area path that matches the level set in the configuration to represent teams, the connector will attempt to find an Agile Team with a name that matches exactly (including case and spaces). If the connector cannot find a matching team, it will create one. If teams don't yet exist, we recommend letting the connector create the teams for you, so that you don't have to worry about matching names.

**Note**: if the connector finds a team with the same name but under a different program, it will move the team.

Users are added to teams in Jira Align when the user is added as the assignee of a story, defect, or child task that is assigned to a team and a sprint.

## Automatic user creation and team assignments

© 2024 Atlassian — Confidential and Proprietary

Users may be created and added to a team in Jira Align automatically, based on their association with a work item in Azure DevOps. The following rules apply:

- When an Azure DevOps user creates or updates a synced story, defect, or child task, a matching user — if not present — is created in Jira Align.
    - The new user is not added to a team in Jira Align.
- When an Azure DevOps user is added as an Assignee to a synced story, defect, or child task that is assigned to a team, a matching user — if not present — is created and added to the matching Jira Align team.
    - If the user already exists in Jira Align, they are added to the matching team.
    - Users added to a Jira Align team are given the role of Developer.
    - Team membership for tasks is determined through the team associated with the parent story in ADO.
    - Team allocations are not set in Jira Align for agile teams.
    - Team allocations are set to 100% for tasks and defects in Jira Align for Kanban teams.
    - When team assignment changes for a work item in ADO, team members are not removed from the previous team in Jira Align.

## Time increments

A program increment in Jira Align matches with an Iteration Path in Azure DevOps, where the iteration path is at the number of levels of hierarchy identified in the connector configuration setting and the Iteration Path has an end date that matches the end date of the Jira Align program increment. It is possible for the connector administrator to set a number of buffer days to allow the match to be fuzzy.

A sprint in Jira Align is **created** automatically by the connector for each program's Agile Team for Azure DevOps iteration paths where:

- The iteration path is at the number of levels of hierarchy identified in the connector configuration setting
- The iteration path is below an iteration path that has automatically mapped to a program increment in Jira Align
- There is a **sync** sprint defined in Jira Align at the program increment that matches the end date
- A team exists within the appropriate program

For example, the iteration path /project/2019/pi-1/sprint 7 will only create a team sprint if /project/2019/pi-1 maps to a program increment in Jira Align and that program increment has a sync sprint with an end date that will map to the full iteration path's end date. The team sprint name in Jira Align will be created or changed to the same name.

When work items are copied from Azure DevOps, the iteration and program increment will be set based on finding those that match the iteration path's date criteria as above. It is possible for a work item to be in an iteration path that is below the defined levels and the connector will still map it properly. For example, a feature in /project/2019/pi-1/sprint 7 will still be placed in the program increment pi-1 if the program increment level is set to 3, and a story in /project/2019/pi-1/sprint 7/sprintlet a/monday would still find its way to the program increment related to pi-1 and the sprint related to sprint 7 if the sprint level is set to 4.

© 2024 Atlassian — Confidential and Proprietary

**Note**: The count of levels for iteration paths and for area paths always includes the root (which is the project name in Azure DevOps), so /project/21stc/2019/H1/Q2/sprint 2.3 is at level 6.

Users are assigned to sprints in Jira Align when the user is added as the assignee of a story, defect, or child task that is assigned to a team and a sprint.

**Notes:**

- Sprint assignments for tasks are determined by the parent story's iteration path.
- The connector does not remove users from sprints; it only adds them.

# Assessment tables: document your Azure DevOps hierarchies

The following tables show examples of how you might have Azure DevOps configured for a portfolio, program, and team; the tables also contain blank cells for you to pencil in a sampling of your hierarchies. Please examine your Azure DevOps hierarchy during this exercise and write down your configuration to help us understand how it is structured and what levels are represented. It may be beneficial to collaborate with your Jira Align Solutions Architect for these exercises.

## Team area hierarchy

If multiple portfolios exist, capture each one and set up a separate project connector for each.

| Level # | Portfolio level example | Your configuration |
|---|---|---|
| 1 | Portfolio | |
| 2 | Program | |
| 3 | Team | |

## Iteration hierarchy

© 2024 Atlassian — Confidential and Proprietary

| Level # | Portfolio level example | Your configuration |
|---|---|---|
| 1 | Fiscal Year | |
| 3 | Program Increment | |
| 4 | Sprint | |

### Work item hierarchy

| Level # | Portfolio level example | Your configuration |
|---|---|---|
| 1 | Feature | |
| 2 | Story | |
| 3 | Task | |

# Azure DevOps integration prerequisites

Before configuring the tool integration, you need to decide on the basic setup in Jira Align and collect the following information on your Azure DevOps projects.

© 2024 Atlassian — Confidential and Proprietary

| Organization/ Collection | Project | Project Template | Custom Workflows | Custom Objects | Jira Align Scope | Jira Align Name |
|---|---|---|---|---|---|---|
| https:// srv/collect | Example | Agile | Yes | No | Program | CMS |
| | | | | | | |

1. Create a list of Azure DevOps projects that you plan to synchronize to Jira Align. Determine which Jira Align portfolio, program, or team to map to each Azure DevOps project. See the table above.
2. Access for data integration.
   **Note**: The Jira Align Azure DevOps connector is installed with the Jira Align server and makes its calls from Jira Align to the Azure DevOps system using a combination of Azure DevOps APIs. Data is always pulled and pushed by the Jira Align server.

   • What type of authentication does your server use?
   - Basic Authentication
   - NTML Authentication
   • If using Azure DevOps, you need only the organization/collection URL(s).
   • If Azure DevOps is behind a firewall and Jira Align is hosted in the cloud, SSL encryption on your server with certificates from a public certificate authority is required, along with a path through the firewall for the connector to call from the Jira Align server.
   • If both Azure DevOps and Jira Align are to be on-site behind the firewall, then VPN access for both the Jira Align engineer and the Solutions Architect is required. If no VPN access is provided to the Jira Align team, there is an additional license fee.
3. In each organization/collection, create an Azure DevOps user specifically for the integration and provide the user names and passwords to the Solutions Architect. These user accounts must be assigned to the following permission groups in Azure DevOps:

   • Project Organization/Collection Service Accounts
   • Team Foundation Service Accounts
   • Project Organization/Collection Administrators
4. What process template do you use? There are different object names, properties, and default workflows based on the template. If you use multiple or customized templates, be sure to note that on the project list in Step 1 of this section.

   • CMMI
   • Agile
   • Scrum
   • Other
5. Provide an overview of your structures including any customizations.
6. Provide XML for any customized epic, feature, story, task, and any custom-created items.
7. From what date would you like data to be pulled?
   Start date: (e.g. 03/01/2015)

© 2024 Atlassian — Confidential and Proprietary

8. Provide PI dates to be mapped to replicate the exact dates in Jira Align. If you change the dates of the PI in Azure DevOps too far, they will not be mapped anymore. Mapping is driven by dates. There is a configurable range for date overlap in the configuration file.

| PI # (e.g. PLI-P1; PLI – P2) | Start Date | End Date | Program Name | Sprint Length (e.g. 1wk, 2wk, 3wk) |
|---|---|---|---|---|
| | | | | |
| | | | | |
| | | | | |

9. Are there any areas of the Area Hierarchies that should be specifically included or excluded? The real-life situation here is a ragged hierarchy. Sometimes areas are used to represent teams and sometimes they are used for other things, like triage or backlog.

| Project Name | Area Paths | Include/Exclude? |
|---|---|---|
| Example | Example\Bad\Area | Exclude |
| Example2 | Example2\ProgramArea | Include |
| | | |

© 2024 Atlassian — Confidential and Proprietary

10. Are there any areas of the Iteration Paths that should be specifically included or excluded? The real-life situation here is a ragged hierarchy. Sometimes iteration paths are used to represent timeline work, and sometimes they are used for other things, like triage or backlog. Iteration Paths are excluded explicitly one-by-one.  Iteration paths can be limited to include only  a single branch of the tree. For example, it is possible to configure the system to only review iteration paths that start with Example\2019.  Note, however, that this must be a valid path: Example\20 would not be sufficient to capture all paths in this century (unless the path is actually Example\20\2019\...

| **Project Name** | **Iteration Paths** | **Include/Exclude?** |
|---|---|---|
| Example | Example\Bad\Iteration | Exclude |
| Example2 | Example2\2019 | Include |
| | | |

## Type Mappings

For work items that are synchronized, the connector can map values between the Value Area field in Azure DevOps and the Type field in Jira Align. For each record type, determine the value(s) that will match the values in Jira Align.

**Features**

| **Jira Align Value** | **Azure DevOps Value(s)** |
|---|---|
| Business | |
| Enabler | |
| Non Functional | |
| Architectural | |

**Stories**

© 2024 Atlassian — Confidential and Proprietary

| Jira Align Value | Azure DevOps Value(s) |
|---|---|
| User | |
| Automation | |
| Blocker | |
| Database | |
| Defect | |
| Design Spike | |
| Documentation | |
| Enablement | |
| End To End | |
| Feature Finisher Testing | |
| Performance & Sizing | |
| Project | |
| Ready To Publish | |
| Regression Test | |
| Tech Stack | |
| Template | |

© 2024 Atlassian — Confidential and Proprietary

# Severity Mappings

For work items that are synchronized, the connector can map values between the Value Area field in Azure DevOps and the Type field in Jira Align. For each record type, determine the value(s) that will match the values in Jira Align.

| Jira Align Value | Azure DevOps Value(s) |
|---|---|
| 1 - Crash / Data Loss | |
| 2 - Major Problem | |
| 3 - Minor Problem | |
| 4 - Cosmetic | |

# State Mappings

For work items that are synchronized, the connector can map values between the state fields in Azure DevOps and Jira Align.  For each record type, determine the value(s) that will match the values in Jira Align.  Be sure to have a home for all possible states in both systems.

**Features**

© 2024 Atlassian — Confidential and Proprietary

| Jira Align Value | Azure DevOps Value(s) |
|---|---|
| 0 - Pending Approval | |
| 1 - Ready To Start | |
| 2 - In-Progress | |
| 3 - Dev Complete | |
| 4 - Test Complete | |
| 5 - Accepted | |

**Stories**

| Jira Align Value | Azure DevOps Value(s) |
|---|---|
| 0 - Pending Approval | |
| 1 - Ready To Start | |
| 2 - In-Progress | |
| 3 - Dev Complete | |
| 4 - Test Complete | |
| 5 - Accepted | |

**Defects**

© 2024 Atlassian — Confidential and Proprietary

| Jira Align Value | Azure DevOps Value(s) |
|---|---|
| Active | |
| In-Work | |
| Pending Test | |
| Verifying | |
| Fixed | |
| Not a Bug | |
| CNR By Design | |
| Postponed | |
| Duplicate | |

**Tasks**

| Jira Align Value | Azure DevOps Value(s) |
|---|---|
| 1 - Not Started | |
| 2 - In Progress | |
| 3 - Done | |

© 2024 Atlassian — Confidential and Proprietary

# Configure and activate Azure DevOps integration

You should configure Azure DevOps settings in Jira Align (ADO) for the correct data synchronization. You can customize many settings according to your needs.

To configure and turn on integration:

- Set up connector information
- Configure your projects
- Configure work item sync
- View logs

Prerequisites:

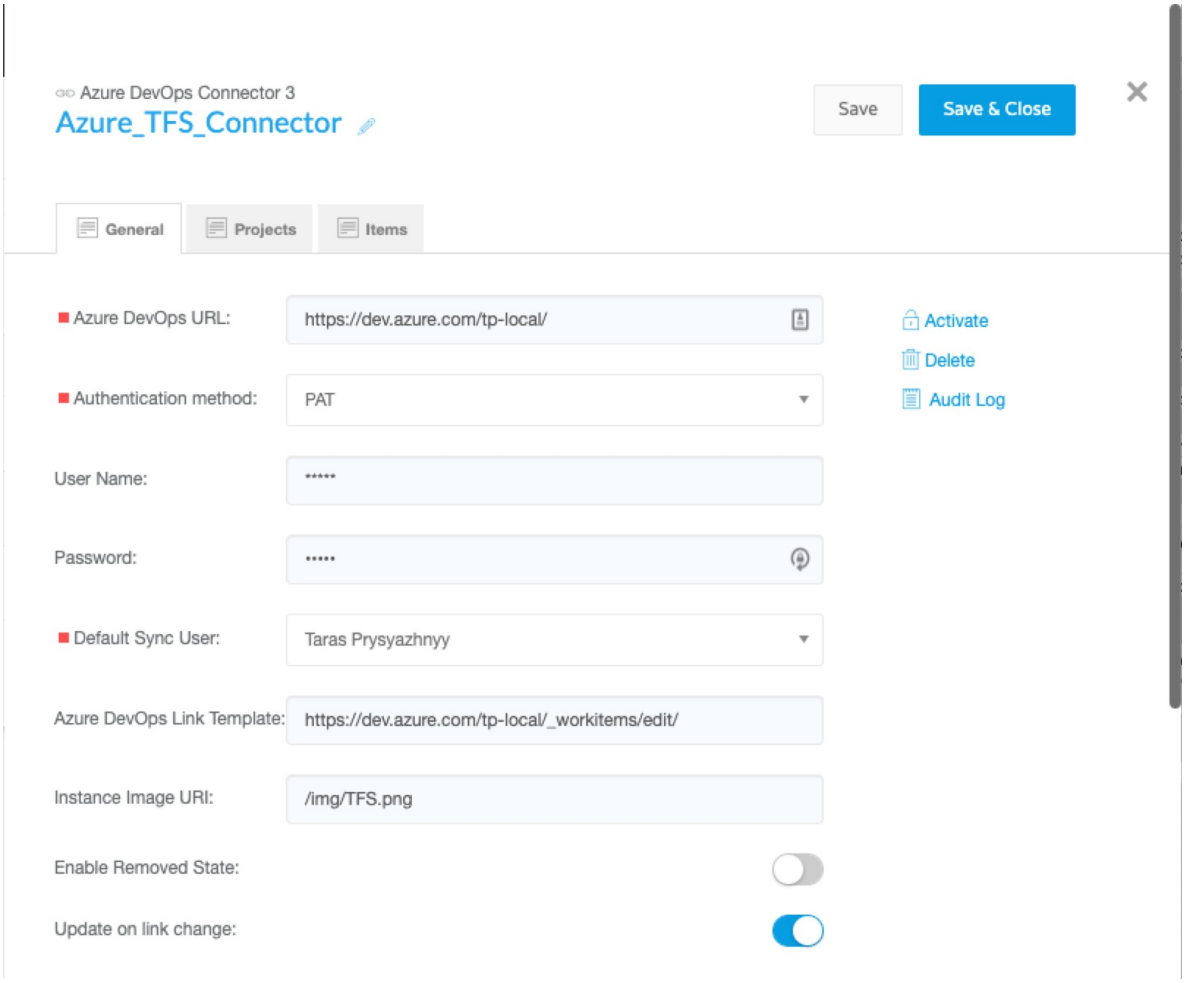
1. In Jira Align, set up portfolios and programs.
2. Create the PI dates you identified in Step 8 in the Azure DevOps Integration Prerequisites section.
3. For each Jira Align PI, create sync sprints for Azure DevOps iterations.

## Set up connector information

1. Select the **Settings gear**  in the top navigation bar.
2. On the left side of the page, select **Azure DevOps Settings** in the **Connectors** section.

© 2024 Atlassian — Confidential and Proprietary

3. Click **Add Connector**. The **New Azure DevOps Connector** slide-out appears.




4. Type the connection name and the URL to connect to Azure DevOps.
   **Note**: This configuration supports both the https://visualstudio.com and https://dev.azure.com.
5. Select the authentication method you want to use: PAT (personal access token), NTLM (NT LAN manager), or OAuth (only available for Cloud instances), and then enter the necessary credentials. Read more on authentication methods here.
6. In the **Item Link Back URL Template** box, type the URL template used when you want to present a link in Jira Align in the upper-right corner of an item that will take you to the linked item in Azure DevOps.
7. Enter a value between 60 and 1440 minutes in the **Project sync timer** field. This value controls how often the connector syncs mapped program area paths, team area paths, and iteration paths.
8. Enter a value between 3 and 60 minutes in the **Work item sync timer** field. This value controls how often the connector runs to sync new and updated work items.
9. Enter a value between 240 and 1440 minutes in the **Recycle bin sync timer** field. This value controls how often the connector syncs items in the ADO recycle bin with the Jira Align recycle bin.
10. Switch on the **Update on Link Change** toggle to allow the connector to perform a separate search to identify and sync items in AzureDevOps for which the only change was the identity of the item's parent. ADO's standard query for changed items only provides items that were directly changed; changing or removing an item's parent does not update the timestamp of the item and the standard query will not find these changes. However, the separate search enabled by this setting is an extra step in the sync process and connector performance may be improved by disabling the setting.
11. In the **Default Jira Align system role for ADO users** dropdown, select a system role. All users added to Jira Align through the connector will receive this system role.

© 2024 Atlassian — Confidential and Proprietary

12. Enter a value between 1 and 30 days in the **Logs Retain Days Limit** field. This value controls how long Jira Align should save information in the error log on the Jira Align page.

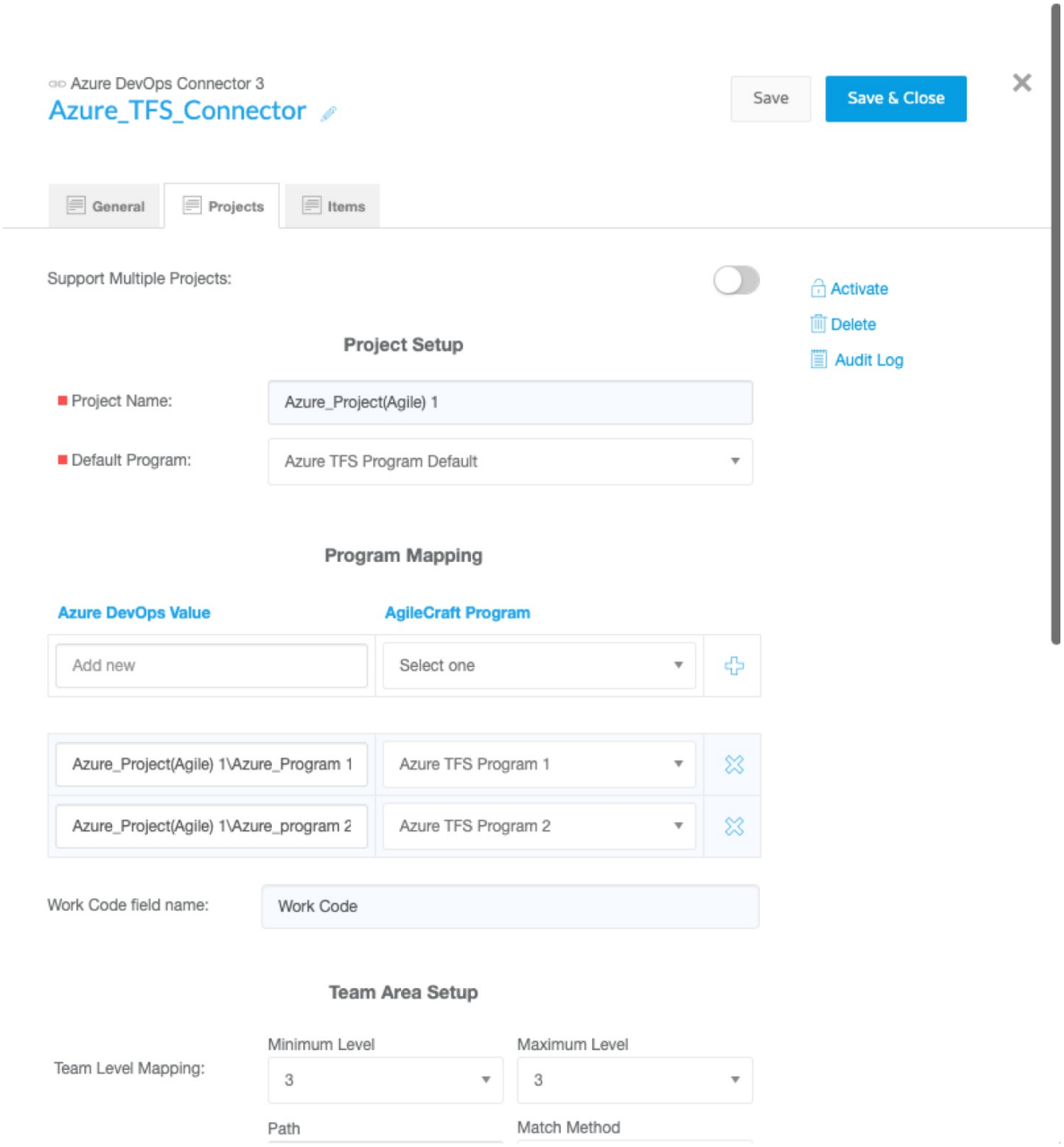
    **Note:** This setting does not change the internal log retention for server-side and Atlassian-visible logs.
13. Select **Save**. The connector information updates and is deactivated by default; additional **Projects** and **Items** tabs display at the top of the slide-out.
14. Activate the connector by selecting **Activate** on the right of the connector slide-out. When active, the **Activate** option will change to **Deactivate** so that the connector can be disabled if desired.

Remove the connector permanently by selecting **Delete** on the right of the slide-out.

# Configure your projects

© 2024 Atlassian — Confidential and Proprietary

1. On the connector slide-out, select the **Projects** tab.




2. Switch on the **Support Multiple Projects** toggle if you would like to map multiple Azure DevOps projects with this single connector. We recommend enabling this option when first setting up the connector, even if you have only one project. This makes future expansion easier.

**Important:** When this toggle is switched on or off, the connector's previous project setup is cleared.

If mapping a single project with the connector:

1. Enter the Azure DevOps **Project Name** for the project you'd like to map.
2. Select a Jira Align program to use as a **Default Program** if a program cannot be found in the mapping for an item's area path.

If mapping multiple projects with the connector:

© 2024 Atlassian — Confidential and Proprietary

1. Enter a **Process Template Name.**
   **Note:** All projects using the same connector must have the same process template name.
2. Add Azure DevOps projects to be synced by entering the project name and then selecting the **plus (+)** icon to add it to a list of projects to sync.
3. Remove a path from the list by selecting the **X** icon next to its name.

Regardless of how many projects you're mapping, continue to configure your projects:

1. In the **Program Mapping** section, enter the **Azure DevOps Value** that you wish to map, and select its corresponding **Jira Align Program.**
2. Add additional programs for mapping by selecting the **Add** button. You can also delete program mappings by clicking the **X icon** next to the mapping.
3. In the **Work Code Field Name** text field, enter the name of the Azure DevOps custom field you'd like to use to set Jira Align work codes for features.
4. Under **Team Area Setup,** set up your **Team Level Mapping** by selecting the minimum and maximum team area hierarchy levels in the **Minimum Level** and **Maximum Level** drop-downs, respectively.
5. If you're only mapping a single project with this connector, set a **Scope Path.** Enter an Azure DevOps **Path** to define the team sync scope in the text field, and then select a **Matching Method** in the drop-down:
   1. Begins with - all paths that starts with the specified value will be synced as teams.
   2. Includes - all paths that includes the specified value will be synced as teams.
   3. Exact only - only the path with the specified value will be synced as a team.

   If you're mapping multiple projects, leave the scope path blank.
6. Optionally, add **Exclude Paths** to denote team area paths, including children, that will not be synced. All items with paths set in the **Exclude Paths** section will not be synced. For each path you'd like to exclude, enter the name of the path to exclude, and then select the **plus (+)** icon to add it to a list of excluded paths.
   - If you're mapping a single project, list every path in the project that you do not want to synchronize (unless using scope path).
   - If you're mapping multiple projects, it is only necessary to list paths that you want to exclude that are also under paths mapped to programs.


7. Remove a path from the **Excluded Path** list by selecting the **X** icon next to its name.

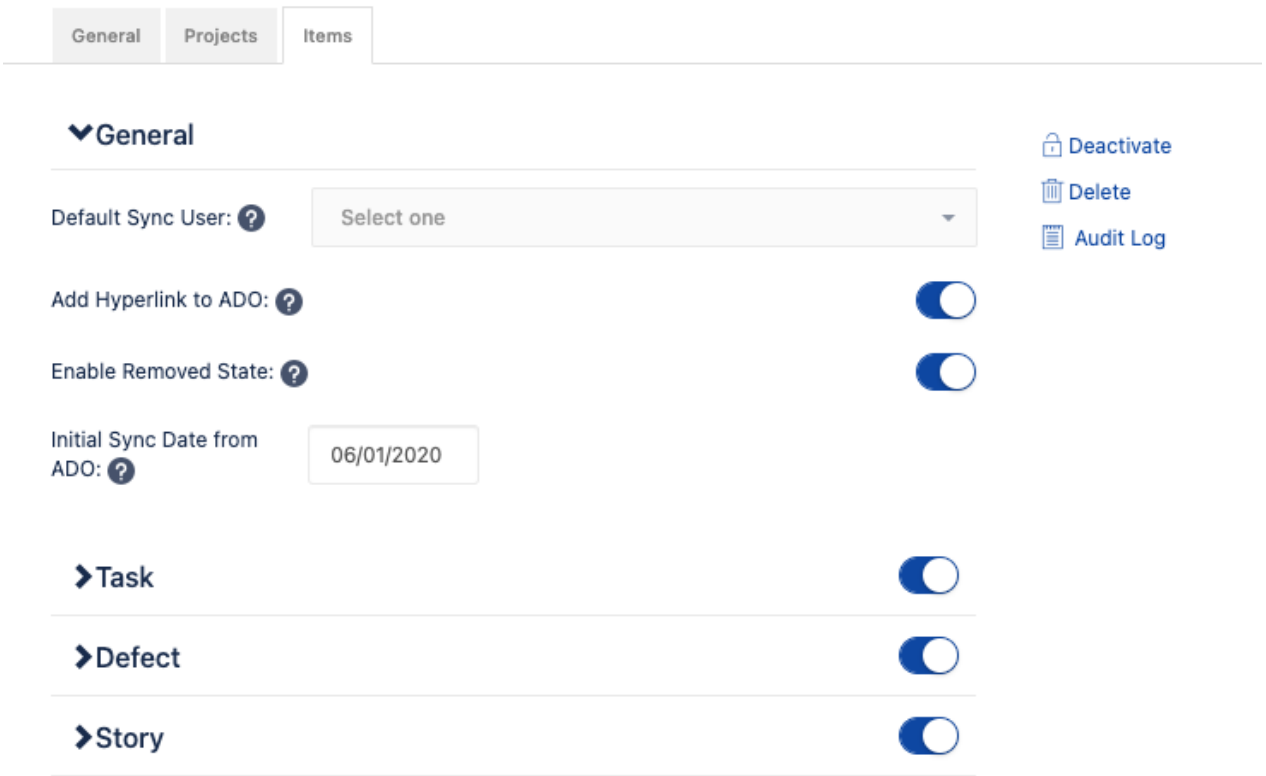
**Note:** When syncing a single-project, exclude paths should be within the team area scope set in **Scope Path**.

© 2024 Atlassian — Confidential and Proprietary

1. Under **Iteration Area Setup,** set up your **Release Level Mapping** by selecting the minimum and maximum release area hierarchy levels in the **Minimum Level** and **Maximum Level** drop-downs, respectively.
2. Set up your **Iteration Level Mapping** electing the minimum and maximum number of iteration area hierarchy levels in the **Minimum Level** and **Maximum Level** drop-downs, respectively.
3. If desired, set a **Scope Path.** Enter an Azure DevOps **Path** to define the sync scope in the text field, and then select a **Matching Method** in the drop-down:
    1. Begins with - items associated with all paths that starts with the specified value will be synced.
    2. Includes - items associated with all paths that includes the specified value will be synced.
    3. Exact only - items associated with the path with the specified value will be synced.
4. Optionally, set a number of **Buffer Days** to automatically map Azure DevOps iteration paths that have end dates to Jira Align sprints and PIs on a day range.
5. Optionally, add **Exclude Paths** to denote iteration area paths that will not be synced. All items with paths set in the **Exclude Paths** section will not be synced. For each path you'd like to exclude, enter the name of the path to exclude, and then select the **plus (+)** icon to add it to a list of excluded paths.
6. Remove a path from the **Excluded Path** list by selecting the **X** icon next to its name.
7. Select **Save** to save your changes to the connector.

# Configure work item sync

A variety of work items can be synced between Azure DevOps and Jira Align, including tasks, defects, stories, and features. On the **Items** tab, you can toggle syncing of each work item type on and off, as well as configure a variety of other settings.



Defects, stories, and features can be synced forward from Azure DevOps to Jira Align or bi-directionally; tasks can only be synced forward (from Azure DevOps to Jira Align).

Start by configuring general work item settings:

1. Expand the **General** section at the top of the **Items** tab.
2. From the **Default Sync User** drop-down menu, select the user name that will be used by default to synchronize the unassigned items in Jira Align. If you don't select anyone from the list, the item will stay unassigned.
3. The **Add Hyperlink to ADO** toggle controls whether Azure DevOps items will contain web links back to their corresponding Jira Align work items. Switch off the toggle to turn off these links in Azure DevOps. **Note:** Jira Align work items will always contain links to their corresponding Azure DevOps item.
4. Switch on the **Enable Removed State** toggle to assign objects in the Recycle Bin in Jira Align to the Removed state in Azure DevOps, and vice-versa. When the toggle is switched off, the connector will attempt to use ADO's recycle bin when an item is moved to the recycle bin in Jira Align. If the toggle is switched on, ensure that the Removed state is enabled for all record types that are synchronizing in both directions.
5. Optionally, in the **Initial Sync Date from ADO** field, select a date. The initial connector run will search for items in Azure DevOps that have changed since this date.

   **Note:** This setting applies to combinations of record types and projects individually. There is a separate timestamp maintained for each item type in each project so that issues encountered with one project or type do not affect other item types in other projects. For example, the search timestamp for a product backlog item in Project X might be this morning, and a product backlog item in Project Y might still be in 2020. This setting can be useful after additional projects and record types are added to the connector configuration after the initial run. Otherwise, changing this setting will not affect the connector unless no items were found in the initial sync.

Next, determine which work items will be synced. To sync a work item:

1. Switch on the toggle next to the work item's type to turn sync on.

© 2024 Atlassian — Confidential and Proprietary

2. Enter the Azure DevOps work item types to be synced:
   a. For stories, use the **Story type mappings** section to map Jira Align story types to Azure DevOps work item types. Map Jira Align values on the left to Azure DevOps work items on the right. You can add additional mappings by selecting **Add story type mapping** at the bottom of the table. Each Jira Align value can only be mapped once.
   If needed, you can remove a mapping by selecting the **X** icon to the right of a row.
   **Notes:**
   - When editing story type mappings, keep in mind that any unmapped Jira Align story types will stop syncing with Azure DevOps. This may result in stories remaining in Jira Align, but not updating in Azure DevOps because the mapping was removed.
   - If the connector has been syncing before setting up or making changes to story type mappings, and work items previously only existed in one product (e.g. work items only existed in Azure DevOps), those work items won't immediately display in the other product. You'll need to make a change to the relevant work items in either Jira Align or Azure DevOps to trigger a sync. A quick way to do this is to go to Azure DevOps, run a query on all relevant work items, then perform a mass action (e.g. add a tag) to the results. Any action that causes the Azure DevOps last updated date to change will suffice. This will cause those Azure DevOps work items affected to sync to Jira Align.
     b. For all other work item types, enter a single Azure DevOps work item type in the **Azure DevOps Item Type** field.
3. Select a **Delete Action** to determine what will happen to the work items in Jira Align when the associated work items are deleted in Azure DevOps.
4. For features and stories, enter the name of the Azure DevOps field that syncs with Jira Align acceptance criteria in **Acceptance criteria field name**.
   **Note:** In programs that sync with ADO, you'll be limited to a single acceptance criteria text field in Jira Align since the default ADO acceptance criteria is a text field. If you already have multiple acceptance criteria in Jira Align for work items in these programs, the connector will automatically merge them into a single, formatted text field.
5. For features and stories, fields to map Jira Align custom text input, text area, and single-select dropdown fields will display. Enter the names of the equivalent Azure DevOps fields in the corresponding fields.
6. In the **State Mapping (Azure DevOps to Jira Align)** section, enter the **Azure DevOps Value** that you wish to map, and select its corresponding **Jira Align State.** Optionally, add one or more **Other Allowed States** in Jira Align for the mapping.
7. For all items except tasks, in the **State Mapping (Jira Align to Azure DevOps)** section, enter the **Jira Align State** that you wish to map. Then, set its corresponding **Azure DevOps Value.** Optionally, add one or more **Other Allowed States** in AzureDevOps for the mapping.
   1. Add additional states for mapping by selecting the **Add** button. You can also delete state mappings by clicking the **X icon** next to the mapping.
8. For defects, in the **Severity Mapping** section, enter the **Jira Align Severity** that you wish to map. Then set its corresponding **Azure DevOps Value**.
9. For stories and features, in the Type Mapping section, enter the Jira Align type that you wish to map to a Azure DevOps Value Area. Optionally, add one or more Other Allowed Types for the mapping.

© 2024 Atlassian — Confidential and Proprietary

10. For stories, features, and bugs where the sync direction is bidirectional, in the **Azure DevOps Field Defaults (Jira Align to Azure DevOps)** section, type the **Azure DevOps Field** you'd like to set a default value for. Then, set the **Default Value**. If there's no value already set on the field in Azure DevOps, and no value supplied from Jira Align, the connector will set the Azure DevOps field to the default value.
    1. Add additional field default values by selecting the **Add** button. You can also delete field default values by clicking the **X** icon next to the mapping.

# Authentication methods

**PAT (personal access token)**

For PAT authentication, you need to generate an access token for the appropriate organization in Azure DevOps first. Read the Authenticate access with personal access tokens for Azure DevOps Services and TFS article to learn how to create personal access tokens to authenticate access. For proper functioning of the connector, select **Project and team (read, write, and manage)**, **Work item (full)**, and **Identity (manage)** as scopes. Finally, enter your user name (email) and password (your access token) in Jira Align.

**NTLM (NT LAN manager)**

NTL is used for server authentication. Type a user name, which is a domain name/the user added to the Azure DevOps server, for example, {local/admin}, and a password, which is a password of this user.

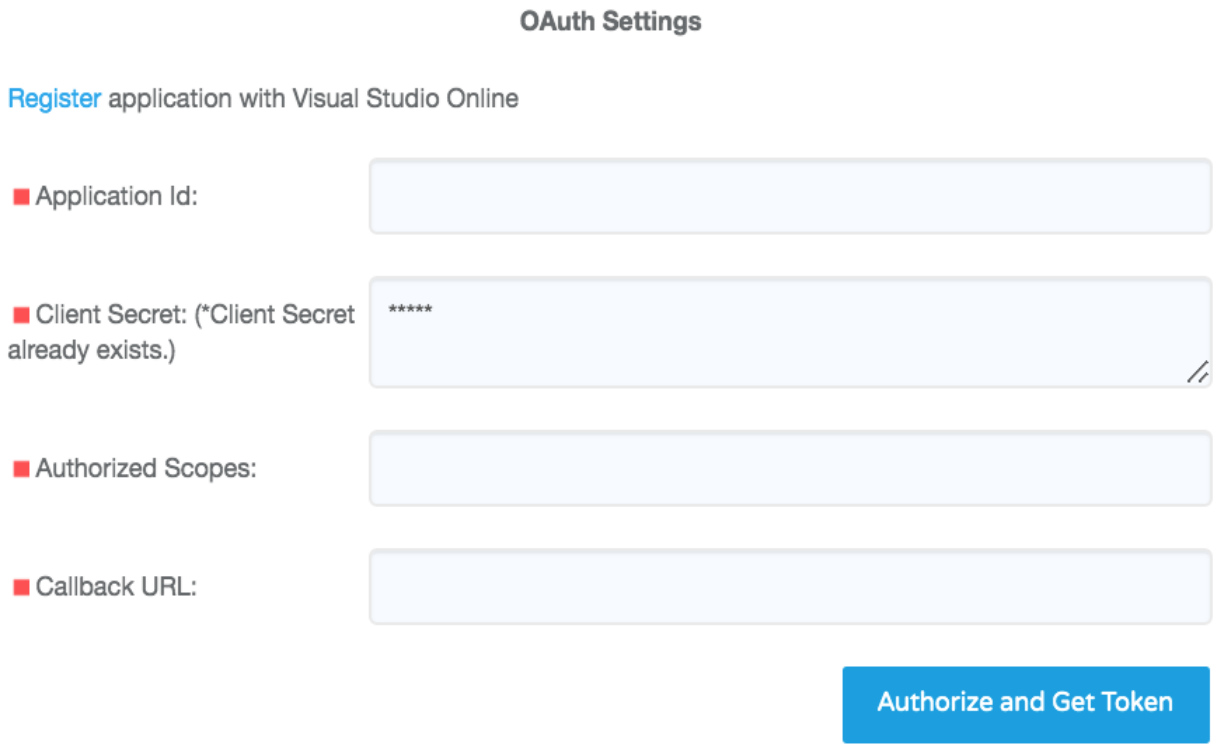
**OAuth**

OAuth is an authentication method that Cloud users can use. First, click **Register** to register your application with Visual Studio Online. Read more on how to register your app here.

Required fields to fill in are your **Company name**, **Application name**, **Application website**, and **Authorization callback URL**. For authorization callback URL, use https://{yourinstancename}/privateapi/tfsConfig/oauth/callback/. For proper functioning of the connector, select **Project and team (read, write, and manage)**, **Work item (full)**, **Identity (manage)**, and **MemberEntitlement Management (read)** as authorization scopes.

After the registration, you will get a generated **App ID**, **Client Secret**, **Authorized Scopes**, and **Callback URL**. Copy and transfer them to Jira Align. Click **Authorize and Get Token**.

© 2024 Atlassian — Confidential and Proprietary




# View logs

On the right of the table on the **Azure DevOps Settings** page, select the **View Logs** button to view error logs on the synchronization process. The log captures information on issues that occurred during the sync processes, specifically:

- **ID**: A unique identification number for the entry.
- **Level**: The type of issue. **Warnings** are issues that don't stop items from copying. **Errors** are issues that prevent items from copying.
- **Message**: Detailed information about the warning or error.
- **Item type**: Name of the type of item where the issue occurred. The name varies based on the direction of the sync process.
- **Jira Align Item ID**: The work item ID in Jira Align. This may be blank if the item does not exist in Jira Align yet, and the sync fails completely.
- **ADO ID**: The work item ID in Jira Align. This may be blank if the item does not exist in Azure DevOps yet, and the sync fails completely.
- **Date**: The date and time when the issue occurred.

**Note**: In addition to filtering by date and message contents, the table can be filtered by one or more values in the Level, Item Type, Jira Align Item ID, or ADO ID fields. To filter the table:

1. Click the column name.
2. In the list of items, select the checkbox next to one or more options to narrow the filter.
3. The filter criteria appear above the list of items, and the table filters to display items that meet the filter criteria.

To remove a filter:

© 2024 Atlassian — Confidential and Proprietary

1. Select the column that contains the filter you want to remove.
2. In the list of items, select the option or options to remove from the filter criteria.

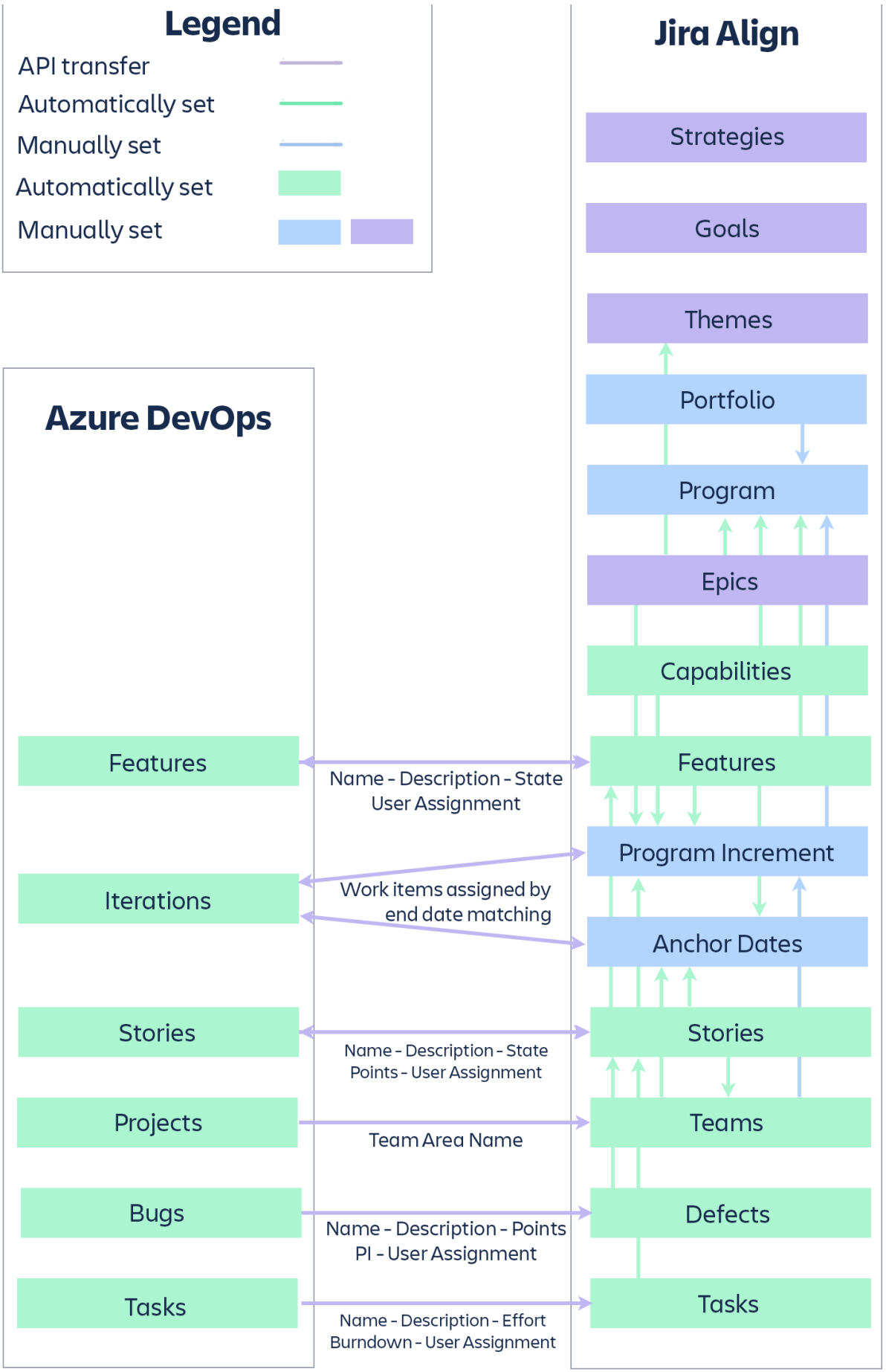
From the **Azure DevOps Settings** page, you can also view a queue report of recent connector activity. Select the **View Queue** button to access the Queue Report. This report captures connector processes as it synchronizes batches of 100-200 items, up to 25 batches (6,000 items) per connector run. On the queue report, you can view the following information:

- **ADO Project**: The Azure DevOps project being synchronized. In multi-project mode, a single connector can synchronize data from more than one Azure DevOps project as long as they are in the same Azure DevOps organization and have the same process template.
- **Item Type**: The Jira Align work item type being synchronized.
- **Last Modified**: The date that the connector uses to determine which items to synchronize from Azure DevOps. This is not the timestamp of the last run. Instead, it is the timestamp of the item that was last successfully copied, so it may be earlier than the Sync End timestamp. This timestamp is updated at the end of each batch of 100-200 items.
- **Sync Start**: The date and time that the connector started the most recent set of batches for the project and work item type.
- **Sync End**: The date and time that the connector last finished a set of batches for the project and work item type. In the vast majority of instances, if this is blank, it means that this is the record type and project that is currently being synchronized. However, it also can be blank if there was an error during processing and the cycle didn't complete.
- **Initial Count**: The count of work items that were found to need synchronization from Azure DevOps, captured at the beginning of the connector run cycle.
- **Last Count**: The count of the work items that were found to need synchronization from Azure DevOps that was captured at the beginning of the most recent batch of items:
    - If the number is less than 100, then the cycle has likely completed and all of the items have been synchronized.
    - If the number is greater than 200, then the cycle likely included over 25 batches and didn't finish copying all the items that were ready. This can happen because the cycle started with a large number of items or because additional items were changed by users in Azure DevOps during the cycle. The items will be synced the next time the connector runs.
    - If the number is between 100 and 200, then either of the above could be in play (or both).

# Azure DevOps integration guide appendix

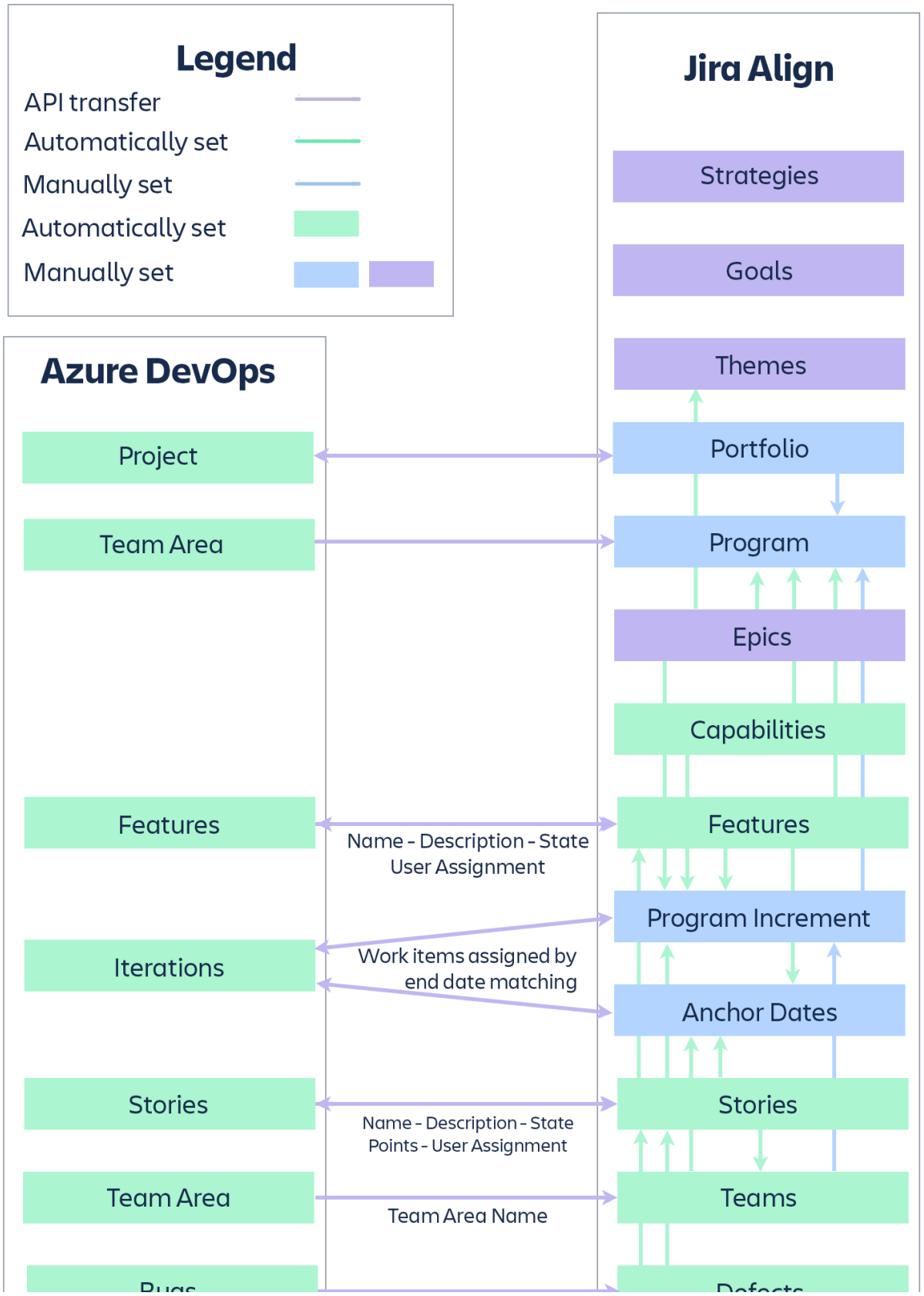
The following diagram illustrates the general scheme of field mapping when an Azure DevOps Project is mapped to a Jira Align team.

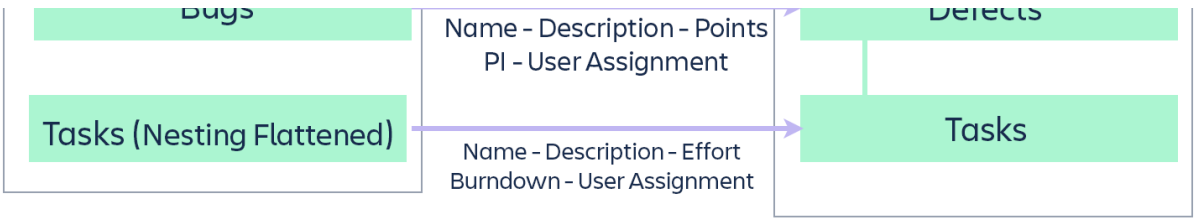
© 2024 Atlassian — Confidential and Proprietary




© 2024 Atlassian — Confidential and Proprietary

The following diagram illustrates the general scheme of field mapping when an Azure DevOps Project is mapped to a Jira Align portfolio.






All users added to Jira Align through the connector automatically receive the Developer team role and the system role selected in the Default Jira Align system role for ADO users dropdown.

Users added to Jira Align sprints through the connector automatically receive the Developer role.

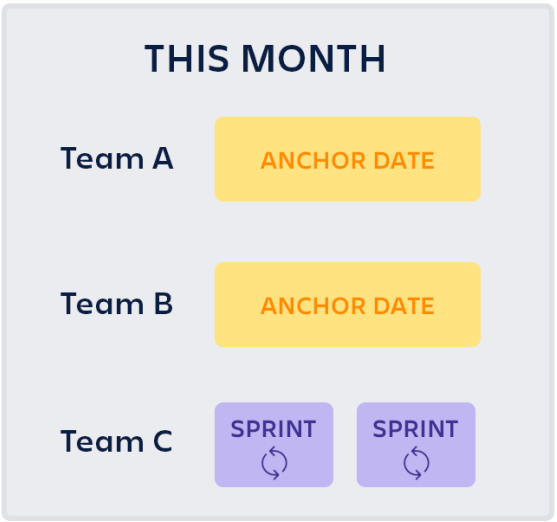
**Allocations**

If a user is a member of only scrum (Agile) teams, they will receive an allocation of 100% for each sprint. If a user is a member of two or more sprints within one anchor date, their allocation of 100% will be equally distributed across these sprints.

When a user is added to a Kanban team, allocations are re-calculated to be equally distributed across all Kanban teams and team sprints within the present anchor date range, for a total of 100%.

As a simple example, a user is a member of two Kanban teams, Team A and Team B. Both teams have an anchor date range for the current month.

She's also part of an Agile team, Team C. This team has two sprints during the current month.




When the user is added through the connector, her allocations for each Kanban team and Agile team sprint will be calculated as:

100% / (2 sprints user is added to within the current anchor date + 2 Kanban teams) = 25%

© 2024 Atlassian — Confidential and Proprietary

© 2024 Atlassian — Confidential and Proprietary