# Kelverion

# INTEGRATION PACK FOR AZURE DEVOPS

*For Microsoft System Center Orchestrator*

For System Center 2016 and 2019, you must use the 32-bit version of the integration pack, which has the name **Kelverion_Integration_Pack_for_Azure_DevOps_2.0**

For System Center 2022 and later, you must use the 64-bit version of the integration pack, which has the name **Kelverion_IP_Azure_DevOps_x64_2.0**

# User Guide
Version 2.0

# Kelverion Integration Pack for Azure DevOps

Published: May 2023

## *Feedback*

Please send suggestions to support@kelverion.com

# Contents

# Kelverion Integration Pack for Azure DevOps

The Integration Pack for Azure DevOps is an add-on for System Center Orchestrator that enables you to integrate with Azure DevOps to automate work item tasks, such as creating, updating, and monitoring work items.

*Important:* The Kelverion Integration Pack for Azure DevOps was formally named the Integration Pack for Azure DevOps and TFS. The integration pack was renamed to emphasize that it only supports Azure DevOps and no longer supports Microsoft TFS.

## System Requirements

The Integration Pack for Azure DevOps requires the following software to be installed and configured prior to implementing the integration. For more information about installing and configuring Orchestrator and  Azure DevOps, please refer to the respective product documentation.

### *Kelverion_Integration_Pack_for_Azure_DevOps (32-bit)*

- Microsoft System Center Orchestrator *2016, 2019*
- Microsoft .NET Framework 4.7.2

### *Kelverion_IP_Azure_DevOps_x64 (64-bit)*

- Microsoft System Center Orchestrator 2022
- Microsoft .NET Framework 4.7.2

* Please see [Kelverion.com/orchestrator](Kelverion.com/orchestrator) for the latest Orchestrator support information.

### *The following versions of Azure DevOps are supported:*

- Azure DevOps Server 2022
- Azure DevOps Server 2020
- Azure DevOps Online

*IMPORTANT:* This version of the integration pack introduces breaking changes in the configuration options. If you are upgrading to this version, make sure to update your configuration options as indicated in [Configuring the Integration Pack](Configuring the Integration Pack) section.

*IMPORTANT:* This version of the integration pack **does not support Microsoft TFS 2017 and 2018**. If you are building runbooks for TFS 2017 or 2018, you should not upgrade to this version, and instead use Kelverion Integration Pack for Azure DevOps and TFS 1.6.

## Registering and Deploying the Integration Pack

After you download the integration pack file, you must register it with the Orchestrator management server and then deploy it to Runbook Servers and Runbook Designers. For more information about how to install integration packs, see the [How to Install an Integration Pack](How to Install an Integration Pack) in the online documentation for System Center Orchestrator.

*IMPORTANT:* Ensure that you are deploying the correct version of the Integration Pack.

- For System Center 2016 and 2019, you must use the 32-bit version of the integration pack, which has the name **Kelverion_Integration_Pack_for_Azure_DevOps**
- For System Center 2022 and later, you must use the 64-bit version of the integration pack, which has the name **Kelverion_IP_Azure_DevOps_x64**

*To register the integration pack:*

1. On the management server, copy the **.OIP** file for the integration pack to a local hard drive or network share.

2. Confirm that the file is not set to **Read Only** to prevent unregistering the integration pack later.

3. Start the **Deployment Manager**.

4. In the navigation pane of the Deployment Manager, expand **Orchestrator Management Server**, right-click **Integration Packs** to select **Register IP with the Orchestrator Management Server**. The **Integration Pack Registration Wizard** opens.

5. Click **Next**.

6. In the **Select Integration Packs or Hotfixes** dialog box, click **Add**.

7. Locate the **.OIP** file that you copied locally from step 1, click **Open** and then click **Next**.

8. In the **Completing the Integration Pack Wizard** dialog box, click **Finish**.

9. On the **End User Agreement** dialog box, read the Kelverion License Terms**,** and then click **Accept**.

10. The **Log Entries** pane displays a confirmation message when the integration pack is successfully registered.


*To deploy the integration pack:*

1. In the navigation pane of the **Deployment Manager**, right-click **Integration Packs**, click **Deploy IP to Runbook Server or Runbook Designer**.

2. Select the integration pack that you want to deploy, and then click **Next**.

3. Enter the name of the runbook server or computers with the Runbook Designer installed, on which you want to deploy the integration pack, click **Add**, and then click **Next**.

4. Continue to add additional runbook servers and computers running the Runbook Designer, on which you want to deploy the integration pack. Click **Next**.

5. In the **Installation Options** dialog box, configure the following settings.

6. To choose a time to deploy the integration pack, select the **Schedule installation** check box, and then select the time and date from the **Perform installation** list.

7. Click one of the following:

    a. **Stop all running runbooks before installing the integration pack** to stop all running runbooks before deploying the integration pack.

b. **Install the Integration Packs without stopping the running Runbooks** to install the integration pack without stopping any running runbooks.

8. Click **Next**.

9. In the **Completing Integration Pack Deployment Wizard** dialog box, Click **Finish**.

10. When the integration pack is deployed, the **Log Entries** pane displays a confirmation message.

# Licensing the Integration Pack

After you register and deploy the integration pack you must provide a valid Kelverion license before running any runbooks that contain activities from the integration pack.

*To deploy the integration pack license file to System Center Orchestrator 2019 or earlier:*

1. Copy the .KAL license file to %PROGRAMFILES(X86)%\Kelverion Automation\Licenses

2. Repeat for each Orchestrator Runbook Server and Runbook Designer host system.

*To deploy the integration pack license file to System Center Orchestrator 2022 or later:*

1. Copy the .KAL license file to %PROGRAMFILES%\Kelverion Automation\Licenses

2. Repeat for each Orchestrator Runbook Server and Runbook Designer host system.

# Configuring the Integration Pack

A configuration establishes a reusable link between Orchestrator and an Azure DevOps server. You can create as many connections as you require specifying links to multiple Azure DevOps servers. You can also create multiple connections to the same server to allow for differences in security permissions for different user accounts.

## To setup an Azure DevOps Online configuration:

1. In the Client, click the **Options** menu, and select *KA Azure DevOps*. The **KA Azure DevOps** dialog box appears.

2. On the **Configurations** tab, click **Add** to begin the configuration setup. The **Add Configuration** dialog box appears.

3. In the **Name** box, enter a name for the configuration. This could be the name of the Azure DevOps server or a descriptive name to distinguish the type of configuration.

4. Click the ellipsis button (…) next to the **Type** box and select *Azure DevOps.*

5. In the **Server URL** box, type the name or URL of the **Azure DevOps Online** web service. For example: https://dev.azure.com/{organization}.

6. In the **Personal Access Token** box, enter the personal access token of the user that Orchestrator will use to connect to **Azure DevOps Online**. You do not need to specify **User Name** and **Password**.

7. **Optional:** In the **Custom Config File** box specify a custom configuration file used by Create Work Item (Custom) and Update Work Item (Custom) activities. See Custom Activities for file specifications.

8. **Optional:** In the **Skip Certificate Validation** box, specify if you want the integration pack to perform server certificate validation or not.

   *IMPORTANT:* When connecting to **Azure DevOps Online**, server certificate validation should be enabled, and **Skip Certificate Validation** should be set to **False.**

9. Add additional configurations if applicable.

10. Click **OK** to close the configuration dialog box, and then click **Finish**.

## To setup an Azure DevOps Server (on-premises) configuration:

1. In the Client, click the **Options** menu, and select *KA Azure DevOps*. The **KA Azure DevOps** dialog box appears.

2. On the **Configurations** tab, click **Add** to begin the configuration setup. The **Add Configuration** dialog box appears.

3. In the **Name** box, enter a name for the configuration. This could be the name of the Azure DevOps server or a descriptive name to distinguish the type of configuration.

4. Click the ellipsis button (…) next to the **Type** box and select *Azure DevOps.*

5. In the **Server URL** box, type the name or URL of the Azure DevOps web service. For example: https://{server:port}.

6. In the **User Name** and **Password** boxes, type the credentials that Orchestrator will use to connect to the **Azure DevOps Server**.

7. **Optional:** In the **Personal Access Token** box, enter the personal access token of the user that Orchestrator will use to connect to **Azure DevOps Server**. When **Personal Access Token** is specified, **User Name** and **Password** are ignored.

8. **Optional:** In the **Custom Config File** box specify a custom configuration file used by [Create Work Item (Custom)](#) and [Update Work Item (Custom)](#) activities. See [Custom Activities](#) for file specifications.

9. **Optional:** In the **Skip Certificate Validation** box, specify if you want the integration pack to perform server certificate validation or not. This applies only when connecting to the server over HTTPS.

   - When set to **True**, the IP will not perform certificate validation. This is typically used in secure on-premises environments, when working with trusted Azure DevOps Servers and self-signed certificates.

   - When set to **False**, the integration pack will validate the server certificate. The Azure DevOps Server must be configured with a valid certificate signed by a valid certificate authority and the server's name in the **Server URL** must be listed on the server certificate.

10. Add additional configurations if applicable.

11. Click **OK** to close the configuration dialog box, and then click **Finish**.

# Azure DevOps Activities

This integration pack adds the KA Azure DevOps category to the **Activities** pane in the Client. This category contains the following activities:

- Add Attachment
- Create Work Item
- Create Work Item (Custom)
- Delete Attachment
- Delete Work Item
- Get Attachment

- Get Project
- Get Work Item
- Monitor Work Item
- Save Attachment
- Update Work Item
- Update Work Item (Custom)

## Common Configuration Instructions for All Activities

The following configuration instructions apply to all activities in this integration pack. Links to this section are included in the configuration instructions for each activity.

### Activity Properties

Each activity has a set of required or optional properties that define the configuration of that activity. This includes how it connects to other activities or how the activity performs its actions. You can view or modify activity properties in the Orchestrator Client.

*To configure the properties for an activity:*

1. Double-click the activity. Alternatively, you can right-click the activity, and then click **Properties**.

2. To save your configuration entries, click **Finish**.

In the activity properties dialog box, several tabs along the left side provide access to general and specific settings for the activity. Although the number of available tabs for activity properties differs from activity to activity, all activities will have a **General** tab, a **Properties** tab and/or **Filters** tab, and a **Run Behavior** tab. Some activities may have additional tabs.

### General Tab

This tab contains the **Name** and **Description** properties for the activity. By default, the **Name** of the activity is the same as its activity type, and the **Description** is blank. You can modify these properties to create more descriptive names or provide detailed descriptions of the actions of the activity.

### Properties/Filters Tab

These tabs contain properties that are specific to the activity.

All activities in this integration pack have the **Configuration Name** property at the top of the **Properties** tab. This property is used to specify the connection to an Azure DevOps server.

1. Click the ellipsis **(…)** button next to the **Name** field, and then select the applicable configuration name. Connections displayed in the list have been previously configured as described in [Configuring the Integration Pack](#).

## Filter Behavior

The Monitor and Get activities use filters to determine the values that will invoke a runbook or retrieve activities. Property values of potential candidates are compared to the values of the filters to determine if they meet the criteria. When matching against values, you select one of the available methods of comparison. An option is provided to either match or not match the filter using each method. For example, the "Does not" version of a method causes alerts that do not match the filter to trigger the runbook. The Monitor activity will only trigger if all filters match and the Get activity will only return Alarms that match all filters.

- **Equals**: the field of the record exactly matches the text or number specified in the filter.

- **Does not equal**: the field of the record does not exactly match the text or number specified in the filter.

- **Is less than**: the field of the record is less than the number specified in the filter.

- **Is less than or equal to**: the field of the record is less than or equal to the number specified in the filter.

- **Is greater than**: the field of the record is greater than the number specified in the filter.

- **Is greater than or equal to**: the field of the record is greater than or equal to the number specified in the filter.

- **Contains**: the field of the record contains the exact text specified in the filter. Unlike the Equals behavior, there can be other text surrounding the matching text.

- **Does not contain**: the field of the record does not contain the exact text specified in the filter. Unlike the Equals behavior, there can be other text surrounding the matching text.

- **Starts with**: the field of the record starts with the exact text specified in the filter. Unlike the Equals behavior, there can be other text following the matching text.

- **Ends with**: the field of the record ends with the exact text specified in the filter. Unlike the Equals behavior, there can be other text preceding the matching text.

- **Matches**: the field of the record matches the specified pattern expression. You can use wildcard expression characters to define pattern expressions.

- **Does not match**: the field of the record does not match the specified pattern expression. You can use wildcard expression characters to define pattern expressions.

# Run Behavior Tab

This tab contains the properties that determine how the activity manages multi-value published data and what notifications will be sent if the activity fails or runs for an excessive period.

## Multi-Value Published Data Behavior

The Get activities retrieve information from another activity or outside source and can return one or more values in the published data. For example, when you use the Get Collection Member activity, the data output from that activity might be a list of computers that belong to the specified collection.

By default, the data from the Get activity will be passed on as multiple individual outputs. This invokes the next activity as many times as there are items in the output. Alternatively, you can provide a single output for the activity by enabling the **Flatten** option. When you enable this option, you also choose a formatting option:

- **Separate with line breaks**. Each item is on a new line. This format is useful for creating human-readable text files for the output.

- **Separate with _** . Each item is separated by one or more characters of your choice.

- **Use CSV format**. All items are in CSV (comma-separated value) format. This format is useful for importing data into spreadsheets or other applications.

The activity will produce a new set of data every time it runs. The **Flatten** feature does not flatten data across multiple instances of the same activity.

## Event Notifications

Some activities are expected to take a limited amount of time to complete. If they do not complete within that time they may be stalled or there may be another issue preventing them from completing. You can define the number of seconds to wait for completion of the action. After this period, a platform event will be sent, and the issue will be reported. You can also choose whether to generate a platform event if the activity returns a failure.

*To be notified when the activity takes longer than a specified time to run or fails to run:*

1. In the **Event Notifications** box, enter the **number of seconds** of run time before a notification is generated.

2. Select **Report if activity fails to run** to generate run failure notifications.

For more information about Orchestrator events, see the "Event Notifications " topics in the [Runbook Properties](https://technet.microsoft.com/en-us/library/hh489610.aspx#Event) (https://technet.microsoft.com/en-us/library/hh489610.aspx#Event Notifications).

# Published Data

Returned data is the foundation of a working Runbook. It is the data produced because of the actions of an activity. This data is published to an internal data bus that is unique for each Runbook. Subsequent activities in the Runbook can subscribe to this data and use it in their configuration. Link conditions also use this information to add decision-making capabilities to Runbooks.

An activity can only subscribe to data from the activities that are linked before it in the runbook. You can use published data to automatically populate the property values needed by activities.

1. Right-click the property value box, click **Subscribe**, and then click **Published Data**.

2. Click the **Activity** drop-down box and select the activity from which you want to obtain the data.

3. To view additional data elements common to all activities, select **Show Common Published Data**.

4. Click the published data element that you want to use, and then click **OK**.

For a list of the data elements published by each activity, see the Published Data tables in the activity topic. For information about the common published data items, see the Published Data (http://technet.microsoft.com/en-us/library/hh403821.aspx).

# Custom Activities

Create Work Item (Custom) and Update Work Item (Custom) require a custom configuration file to define the Work Item fields. The custom configuration file defines the fields using JSON and the location of the file is specified in the KA Azure DevOps configuration using the KA Azure DevOps options menu.

## *JSON fields*

| fields | JSON array of fields |
|---|---|
| name | Name of field |
| optional | If **optional** is specified, the field is optional. Valid values are: <br>• create <br>• update <br>• createandupdate |
| values | JSON array of browser values |
| isdatefield | If **isdatefield** is specified, the field has a Date/Time browser. Valid values are: <br>• true <br>• false |

*Example*

```
{
    "fields": [
        {
            "name": "State",
            "optional": "update",
            "values": ["New", "Closed", "Open"]
        },
        {
            "name": "Title"
        },
        {
            "name": "Due Date",
            "optional": "createandupdate",
            "isdatefield": "true"
        }
    ]
}
```

# Add Attachment Activity

The **Add Attachment** activity is used in a runbook to add an attachment to a work item. The following sections outline the properties used to configure the activity and the data that it publishes.

## Required Properties
You must configure the following properties.

| | |
|---|---|
| **Collection** | The Team Project Collection that contains the Work Item |
| **Work Item ID** | The Work Item ID |
| **File Path** | The File to add to the Work Item |
| **Comment** | The comment describing the attachment |

## Optional Properties
The activity does not have any optional properties.

## Published Data
The activity publishes the following data.

| | |
|---|---|
| **Collection** | The Team Project Collection that contains the Work Item |
| **Work Item ID** | The Work Item ID |
| **Attachment ID** | The ID of the attachment |
| **Server URL** | The Server URL |
| **User Name** | The User Name |

# Create Work Item Activity

The **Create Work Item** activity is used in a runbook to create a Work Item. This activity integrates with Azure DevOps during config. The following sections outline the properties used to configure the activity and the data that it publishes.

## *Required Properties*

If the selected **Work Item Type** has any required fields, the activity will provide properties that you can use to provide the required details. You must also configure the following properties.

| | |
|---|---|
| **Collection** | The Team Project Collection the project belongs to. |
| **Project** | The Team Project the new Work Item will be created in. |
| **Work Item Type** | The Work Item Type of new Work Item. |

## *Optional Properties*

If the selected **Work Item Type** has any optional fields, the activity will provide properties that you can use to provide additional details for the work item.

## *Published Data*

The activity publishes the following data.

| | |
|---|---|
| **Collection** | The Team Project Collection the project belongs to. |
| **Project** | The Team Project the new Work Item will be created in. |
| **Work Item Type** | The Work Item Type of new Work Item. |
| **Work Item ID** | The Unique ID of the new Work Item. |
| **Server URL** | The Server URL |
| **User Name** | The user name |

# Create Work Item (Custom) Activity

The **Create Work Item (Custom)** activity is used in a runbook to create a Work Item. The following sections outline the properties used to configure the activity and the data that it publishes.

*Important:* This activity requires a custom activity configuration file that outlines the work item fields that you want to support. The configuration file used by the activity is specified in the configuration that you select when you configure the activity. For more information see the Custom Activities section.

## Required Properties

If the selected **Work Item Type** has any required fields, the activity will provide properties that you can use to provide the required details. You must also configure the following properties.

| | |
|---|---|
| **Collection** | The Team Project Collection the project belongs to. |
| **Project** | The Team Project the new Work Item will be created in. |
| **Work Item Type** | The Work Item Type of new Work Item. |

## Optional Properties

If the selected **Work Item Type** has any optional fields, the activity will provide properties that you can use to provide additional details for the work item.

## Published Data

The activity publishes the following data.

| | |
|---|---|
| **Collection** | The Team Project Collection the project belongs to. |
| **Project** | The Team Project the new Work Item will be created in. |
| **Work Item Type** | The Work Item Type of new Work Item. |
| **Work Item ID** | The Unique ID of the new Work Item. |
| **Server URL** | The Server URL |
| **User Name** | The User Name |

# Delete Attachment Activity

The **Delete Attachment** activity is used in a runbook to delete an attachment. The following sections outline the properties used to configure the activity and the data that it publishes.

## Required Properties

You must configure the following properties.

| | |
|---|---|
| **Collection** | The Team Project Collection that contains the Work Item |
| **Work Item ID** | The Work Item ID |
| **Attachment ID** | The Attachment ID |

## Optional Properties

The activity does not have any optional properties.

## Published Data

The activity publishes the following data.

| | |
|---|---|
| **Collection** | The Team Project Collection that contains the Work Item |
| **Work Item ID** | The Work Item ID |
| **Attachment ID** | The ID of the attachment |
| **Server URL** | The Server URL |
| **User Name** | The User Name |

# Delete Work Item Activity

The **Delete Work Item** activity is used in a runbook to delete a work item. The following sections outline the properties used to configure the activity and the data that it publishes.

## Required Properties

You must configure the following properties.

| Collection | The Team Project Collection the Work Item belongs to. |
|---|---|
| **Work Item ID** | The ID of the Work Item to delete |

## Optional Properties

The activity does not have any optional properties.

## Published Data

The activity publishes the following data.

| Collection | The Team Project Collection the Work Item belongs to. |
|---|---|
| **Work Item ID** | The ID of the Work Item to delete |
| **Server URL** | The Server URL |
| **User Name** | The User Name |

# Get Attachment Activity

The **Get Attachment** activity is used in a runbook to retrieve attachment details. The following sections outline the properties and filters used to configure the activity and the data that it publishes.

## Required Properties
You must configure the following properties.

| Collection | The Team Project Collection that contains the Work Item |
|---|---|
| **Work Item ID** | The Work Item ID |

## Optional Properties
The activity does not have any optional properties.

## Filters
You can use the following filters to control which attachment records to retrieve.

| Attached Time | Attached time |
|---|---|
| **Attached Time UTC** | Attached time in UTC |
| **Comment** | Attachment comment |
| **Creation Time** | Create time |
| **Creation Time UTC** | Create time in UTC |
| **Extension** | File extension of attachment |
| **ID** | Unique ID of attachment |
| **Last Write Time** | Last write time |
| **Last Write Time UTC** | Last write time in UTC |
| **Length** | Attachment size |
| **Name** | Attachment name |

## Published Data
The activity publishes the following data.

| *Collection* | *The Team Project Collection that contains the Work Item* |
|---|---|
| *Work Item ID* | *The Work Item ID* |
| **Attached Time** | Attached time |
| **Attached Time UTC** | Attached time in UTC |
| **Comment** | Attachment comment |
| **Creation Time** | Create time |

| Creation Time UTC | Create time in UTC |
|---|---|
| Extension | File extension of attachment |
| ID | Unique ID of attachment |
| Last Write Time | Last write time |
| Last Write Time UTC | Last write time in UTC |
| Length | Attachment size |
| Name | Attachment name |
| Server URL | The Server URL |
| User Name | The Username |

# Get Project Activity

The **Get Project** activity is used in a runbook to retrieve project details. The following sections outline the properties and filters used to configure the activity and the data that it publishes.

## *Required Properties*
You must configure the following properties.

| | |
|---|---|
| **Collection** | The Team Project Collection |

## *Optional Properties*
The activity does not have any optional properties.

## *Filters*
You can use the following filter to control which projects to retrieve.

| | |
|---|---|
| **Name** | The name of the project |

## *Published Data*

| | |
|---|---|
| **Collection** | The Team Project Collection |
| **Areas** | Area Paths. Semi colon delimited list of area paths |
| **Iterations** | Iteration Paths. Semi colon delimited list of iteration paths |
| **Categories** | Categories. Semi colon delimited list of categories |
| **Work Item Types** | Work Item Types. Semi colon delimited list of work item types |
| **Name** | Project name |
| **Server URL** | The Server URL |
| **User Name** | The User Name |
| **Count** | Number of projects found |

# Get Work Item Activity

The **Get Work Item** activity is used in a runbook to retrieve work items. The following sections outline the properties and filters used to configure the activity and the data that it publishes.

## *Required Properties*

You must configure the following properties.

| | |
|---|---|
| **Collection** | The Team Project Collection to query. |

## *Optional Properties*

The activity does not have any optional properties.

## *Filters*

The activity will provide filters, based on the fields of the **Collection** you selected, that you can use to control which work items to retrieve.

## *Published Data*

The activity will publish work item data based on the **Collection** you selected. The activity also publishes the following data.

| | |
|---|---|
| **Collection** | The Team Project Collection queried |
| **Server URL** | The Server URL |
| **User Name** | The User Name |

# Monitor Work Item Activity

The **Monitor Work Item** activity is used in a runbook to monitor new and/or updated work items. The following sections outline the properties and filters used to configure the activity and the data that it publishes.

*Note:* The activity will log a warning event with Orchestrator if it is unable to connect to Azure DevOps. The activity will attempt to connect to Azure DevOps in subsequent polling cycle, logging additional warnings until a connection is made.

## *Required Properties*

You must configure the following properties.

| | |
|---|---|
| **Collection** | The Team Project Collection to query. |
| **Monitor Type** | Monitor Type determines when the monitor triggers |
| **Polling Interval** | The number of seconds the monitor polls Work Items |

## *Optional Properties*

The activity does not have any optional properties.

## *Filters*

The activity will provide work item filters, based on the **Collection** you selected, that you can use to control which work items will trigger the monitor.

*Tip:* Filters should be used to reduce the number of work items that can trigger the monitor.

## *Published Data*

The activity will publish work item data based on the **Collection** you selected. The activity also publishes the following data.

| | |
|---|---|
| **Collection** | The Team Project Collection queried |
| **Server URL** | The Server URL |
| **User Name** | The User Name |

# Save Attachment Activity

The **Save Attachment** activity is used in a runbook to save an attachment to a local folder. The following sections outline the properties used to configure the activity and the data that it publishes.

## *Required Properties*
You must configure the following properties.

| | |
|---|---|
| **Collection** | The Team Project Collection that contains the Work Item |
| **Work Item ID** | The Work Item ID |
| **Attachment ID** | The Attachment ID |
| **Folder Path** | The location to save the attachment. If the folder does not exist, it will be created. |
| **If Attachment Exists** | Overwrite – Overwrite the local file. |
| | Ignore – Ignore if local file exists. |
| | Rename – Rename file if local file exists. |

## *Optional Properties*
The activity does not have any optional properties.

## *Published Data*
The activity publishes the following data.

| | |
|---|---|
| **Collection** | The Team Project Collection that contains the Work Item |
| **Work Item ID** | The Work Item ID |
| **Attachment ID** | The ID of the attachment |
| **File Path** | The path of the saved file |
| **Server URL** | The Server URL |
| **User Name** | The User Name |

# Update Work Item Activity

The **Update Work Item** activity is used in a runbook to update a work item. The following sections outline the properties used to configure the activity and the data that it publishes.

## *Required Properties*
You must confiture the following properties.

| | |
|---|---|
| **Collection** | The Team Project Collection the project belongs to. |
| **Project** | The Team Project the Work Item belongs to. |
| **Work Item Type** | The Work Item Type of new Work Item. |
| **Work Item ID** | The ID of the Work Item to update |

## *Optional Properties*
The activity will provide properties, based on the **Work Item Type** that was selected, that you can use to update details of the work item.

## *Published Data*
The activity publishes the following data.

| | |
|---|---|
| **Collection** | The Team Project Collection the project belongs to. |
| **Project** | The Team Project the new Work Item will be updated in. |
| **Work Item Type** | The Work Item Type of the Work Item. |
| **Work Item ID** | The Unique ID of the updated Work Item. |
| **Server URL** | The Server URL |
| **User Name** | The User Name |

# Update Work Item (Custom) Activity

The **Update Work Item (Custom)** activity is used in a runbook to update a work item. The following sections outline the properties used to configure the activity and the data that it publishes.

*Important:* This activity requires a custom activity configuration file that outlines the work item fields that you want to support. The configuration file used by the activity is specified in the configuration that you select when you configure the activity. For more information see the [Custom Activities](#) section.

## Required Properties
You must confiture the following properties.

| | |
|---|---|
| **Collection** | The Team Project Collection the project belongs to. |
| **Project** | The Team Project the Work Item belongs to. |
| **Work Item Type** | The Work Item Type of new Work Item. |
| **Work Item ID** | The ID of the Work Item to update |

## Optional Properties
The activity will provide properties, based on the **Work Item Type** that was selected, that you can use to update details of the work item.

## Published Data
The activity publishes the following data.

| | |
|---|---|
| **Collection** | The Team Project Collection the project belongs to. |
| **Project** | The Team Project the new Work Item will be updated in. |
| **Work Item Type** | The Work Item Type of the Work Item. |
| **Work Item ID** | The Unique ID of the updated Work Item. |
| **Server URL** | The Server URL |
| **User Name** | The User Name |